Development of a Feature Extraction Methodology for Prognostic Tasks of Aerospace Structures and Systems

Antonio Orrù[1], Thanos Kontogiannis[2], Francesco Falcetelli[3], Raffaella Di Sante[4], Nick Eleftheroglou[5]

[1,3,4] *Department of Industrial Engineering, University of Bologna, Italy*
antonio.orru@studio.unibo.it
francesco.falcetelli@unibo.it
raffaella.disante@unibo.it

[1,2,5] *Intelligent and Sustainable Prognostics Group, Aerospace Structures and Materials Department, Faculty of Aerospace Engineering, TU Delft, Delft, 2629 HS, Netherlands*
a.kontogiannis@tudelft.nl
n.eleftheroglou@tudelft.nl

Abstract

The performance of prognostic models used for prognostic health management (PHM) applications heavily depend on the quality of features extracted from raw sensor data. Traditionally, feature extraction criteria such as monotonicity, prognosability, and trendability are selected intuitively. However, this intuitive selection may not be optimal.

This research introduces an innovative approach to transform raw data into 'high-scoring' data without the need for predefined feature extraction criteria. Our methodology involves generating a set of synthetic high-scoring time series. These synthetic time series, resembling the length and amplitude of target features, are created through Monte Carlo sampling (MCS) of a user-defined hidden semi-markov model (HSMM). We pair these synthetic time series with raw data/features from the signals and use them as targets to train a convolutional neural network (CNN). As a result, the trained CNN can convert input features into high-scoring ones, irrespective of their initial characteristics. So, this study provides the following contribution to PHM frameworks: it transforms raw data/features into high-scoring ones without relying on predefined criteria, rather on stochastically generated labels that resemble the nature of the degradation processes. It is worth noting, that the proposed FE technique is independent of the prognostic model that will be utilised, thus making it applicable to the established prognostic models.

We demonstrate and validate the effectiveness of this approach using acoustic emission (AE) sensor data for remaining useful life (RUL) estimation of open-hole CFRP specimens. We compare prognostic performance using cumulative AE features with their transformations via our proposed framework. The transformed features exhibit superior prognostic performance, underscoring the value of our innovative feature extraction framework.

1. Introduction

The current state-of-the-art feature extraction (FE) for prognostics relies heavily on deterministic targets chosen based on intuitively defined metrics such as monotonicity, prognosability, and trendability (Coble & Hines, 2009). These targets have shown efficacy in transforming raw data from sensors, providing a foundational approach for modelling degradation histories (Eleftheroglou, 2020; Moradi, Broer, Chiachío, Benedictus, & Zarouchas, 2023).

The literature-standard procedure for FE for prognostics includes the transformation of the noisy sensor data to high-scoring ones by utilising predefined deterministic labels. These labels are usually derived from simple functions such as second-degree polynomials, exponential and logarithmic. However, the inherent limitation of these deterministic labels lies in their assumption of certainty during the transformation process. This simplification potentially restricts their predictive accuracy and applicability, especially in scenarios characterised by complex and stochastic behaviours of system deterioration and noisy sensor measurements. Additionally, setting deterministic targets for transforming inherently stochastic signals, may significantly increase the complexity and computational time of the applied models (Xu et al., 2023; Chen, Qin, Wang, & Zhou, 2021; Ye, Zhang, Shao, Niu, & Zhao, 2022). These critical deficits motivate our research, highlighting a significant gap in existing methodologies. There is an evident need for enhanced feature extraction techniques that account for the

inherent noise associated with the sensor measurements.
Our research aims to fill this gap by introducing a novel approach integrating a stochastic labelling approach for noisy sensor signals with convolutional neural networks (CNNs). This method leverages Monte Carlo sampling (MCS) of predefined hidden semi-Markov models (HSMMs) to generate high-scoring degradation trajectories that serve as labels for transforming the sensor data utilizing the CNN. By doing so, the model is trained to transform the raw condition monitoring (CM) data into features suitable for prognostic health management (PHM) frameworks while accounting for the inherent noise of sensor measurements. The CNN is also trained on time windows of the data rather than the entire sequences. This attribute enables the model's applicability in real-world scenarios, allowing it to operate online.
The main contribution of the present study is the integration of MCS of HSMMs with CNNs to transform raw sensor data into stochastic degradation trajectories, thus incorporating randomness while being able to operate online in real-world use cases. By alleviating the ill-posed dependency of the FE methods on deterministic labels that stem from intuitive pre-defined metrics, we aim to create a simple and efficient online FE methodology able to transform raw sensor data into features with enhanced prognostic performance. This will ensure the accuracy and higher certainty of the applied PHM frameworks.
The remaining of this study is organised as follows:

- Section 2 delves into the core methodology of this research, providing insight into all of the different components of the proposed transformation methodology as seen in Figure 1.
- Section 3 presents a case study involving acoustic emission (AE) data and demonstrates the proposed methodology's practical application and efficacy. This section is crucial for illustrating the model's ability to handle real-world data.
- Section 4 discusses the research results, by first looking at the transformation of the data and then focusing on the prognostic outcomes. This section highlights the improved prognostic performance achieved using the transformed data by comparing these results against the baseline cumulative transformation of the raw data. This comparative analysis underscores the proposed transformation's enhanced performance in PHM tasks.
- Section 5 concludes the paper with a discussion of the implications of the research findings and proposes ideas for future works.

2. Aspects of the Feature Extraction Framework

In this section, we will introduce the method for enhancing the performance of prognostic algorithms by transforming raw data signals into forms similar to those generated through Monte Carlo simulated data. The methodology unfolds over four subchapters, detailing each critical phase of the transformation process:

Figure 1. Flowchart of the proposed methodology.

1. **Label Creation via MCS an HSMM:** This section describes how MCS of HSMM generates labels with the same time length as the target signal. These labels serve as a reference for the desired signal characteristics.
2. **Data Windowing for Online Operation:** The necessity of data windowing is explored here, underlining its importance in developing a system capable of online operation. Data windowing segments the continuous data stream, facilitating real-time processing and analysis.
3. **Support Model for EOL Estimation and time feature normalization:** This part of the method involves creating a model to predict an asset's end of life (EOL). Knowing the EOL is crucial for normalizing the time feature across

all signals, ensuring consistency in the data transformation process.

4. **Convolutional Neural Network (CNN) Model for Feature Transformation:** The technique's culmination is developing a model capable of converting raw data signals into idealized forms. This model leverages the insights gained from the previous steps to enhance the raw signals, making them more suitable for prognostics.

Together, these steps outline a comprehensive approach to data signal transformation that will boost the predictive accuracy and reliability of prognostic tools.

2.1. Label Creation via MCS of HSMM

Our framework's data transformation is based on a supervised learning algorithm. To that end, we need to provide labels for our samples. These labels need to encapsulate the stochastic nature of the assets' degradation. The way to achieve that is by modelling the degradation process with an appropriately initialized HSMM and then utilising it to generate degradation histories of equal length to our training samples. In order to generate degradation histories from a pre-initialized HSMM, we utilize MCS. It is worth noting that the initialization of the HSMM is a short procedure that is explained thoroughly in Sections 2.1.1 and 3.3, and the MCS is a simple algorithm presented in Algorithm 1. Adding to that, the label creation procedure, as explained below, is only required during the training step of the FE procedure. During the online deployment, the transformation of the sensor signals is done in a sub-second time manner since it includes only the windowing of the incoming data and the inference part of the trained CNN. Therefore, the proposed frameworks applicability and scalability is not an issue. In the following, the HSMM's initialisation is discussed, a short introduction to MCS is provided, and finally, its utilization for the required label generation is presented.

2.1.1. HSMM Initialisation

As discussed previously, the first step is to properly initialise the HSMM model to resemble the degradation process. To that end, it's needed first to declare the Initialisation topology ζ (Eleftheroglou, Zarouchas, & Benedictus, 2020):

- **The number of the hidden states ($\boldsymbol{N}$)**: representing the different levels of degradation.
- **Connectivity between hidden states ($\boldsymbol{\Omega}$)**: This parameter defines the connectivity between the states by defining the allowed transitions between them.
- **Condition Monitoring feature ($\boldsymbol{I}$)**: The observation of the values of a single CM feature is considered to be the sole indicator of damage in the system.
- **Number of discrete monitoring values ($\boldsymbol{V}$)**: In the case where the connection between the observation of the CM feature and the damage states is modelled in a non-parametric way, then it has to be converted to several discrete levels $\boldsymbol{V}$.
- **Transition rate function ($\boldsymbol{\lambda}$)**: This is the main characteristic of the degradation process since each transition will follow this function. This parameter can depend on the sojourn time of the current state, the transition between states, the total operating time or any combination of the above

So, in order to fully characterise the HSMM model, a set of parameters $\boldsymbol{\theta} = \{\boldsymbol{\Gamma}, \boldsymbol{B}\}$ are needed where $\boldsymbol{\Gamma}$ are the degradation process parameters and $\boldsymbol{B}$ are observation process parameters. Γ parameters consist of the parameters needed to define the chosen $\boldsymbol{\lambda}$ function, and $\boldsymbol{B}$ parameters consist of the emission matrix **B**. **B** is a matrix of dimension of $N * V$ containing the likelihood that every possible observation in the Z space will be emitted by a certain hidden state. After defining the HSMM, in order to generate the required sequences, MCS is utilized, which will be explained in Section 2.1.2.

2.1.2. Monte Carlo Simulated Data

Monte Carlo Sampling is a powerful statistical technique used across various fields. At its core, it leverages the power of randomness to solve complex problems, often too difficult or impossible to tackle with traditional deterministic methods. Monte Carlo Sampling operates on a simple yet profound principle: it uses randomness to approximate problems' solutions. Thanks to the law of large numbers, the more samples are used, the better the actual solution is estimated. Monte Carlo methods are useful when analytical solutions are complex or unavailable, providing a versatile tool for approximation and simulation across diverse applications (Lemieux, 2009). However, Monte Carlo sampling has two main disadvantages. Firstly, it can be computationally inefficient, especially when dealing with high-dimensional or complex problems. Since Monte Carlo methods rely on random sampling to estimate quantities, they may require a large number of samples to achieve accurate results, which can be computationally intensive and time-consuming. Secondly, Monte Carlo methods may struggle to estimate rare events or probabilities accurately with very low or very high values, leading to potential inaccuracies in the results.

In this framework, MCS is utilized to perform a "random walk" over the HSMM. Based on the random sampling and the predefined probability functions of the HSMM, a hidden state is picked for each time step, which in turn emits an observation. The observation is captured, and the "random walk" continues following the design of the model, until the transition to the final observed and terminating state occurs. So, in the context of the proposed methodology, Monte Carlo simulations take place in the training process, so the testing process does not require simulated data. Hence, computational inefficiency is not a concern for applicability. Additionally, rare samples (events) are not a concern of the feature extraction process since the domain of condition monitoring techniques is predefined in most cases.

Finally, it is worth highlighting that this approach reverses the traditional training process for HSMMs. Instead of relying on multiple observation sequences to estimate parameters, the predefined parameters are used to produce the observation sequences. This method is detrimental to the stochastic generation of trajectories, which are later used as labels for our transformations. By transforming the raw data with these sequences, the predictive accuracy of prognostic algorithms can be significantly enhanced. This is attributed to the fact that this method provides a transformation based on statistical characteristics rather than the traditionally used deterministic labels as explained in the previous. The pseudocode for the implementation of the MCS of the HSMM is adapted from (Eleftheroglou, 2020) and presented in Algorithm 1.

2.2. Data Windowing for Online Operation:

After acquiring the observation sequences that will be used as labels for the transformation of the raw data, our methodology strategically segments both the signals and their corresponding labels into fixed-length time windows. This division is essential for facilitating the model's operation in an online environment, where it is impractical to process the entire signal simultaneously due to the streaming nature of data.

2.3. Support Model for EOL Estimation and time feature normalization

However, the previously mentioned segmentation introduces a challenge: the absence of a definitive feature indicating the end of the signal complicates the transformation process, potentially affecting the accuracy of the model's predictions. To address this issue, we propose developing a secondary model capable of supporting the transformation by predicting the end-of-life (EOL) of the asset, based solely on information from the initial time window. In doing so, we aim to normalise the time feature on a scale from 0 to 1. However, in practice, this normalisation will yield values ranging from 0 to a number close to 1, as it perfomrs only a rough estimation of the EOL, rather than an accurate prediction. This approach aids our model in effectively adapting to and processing signals in real time, paving the way for more accurate and reliable predictions.

To this end, we opted for a fully connected neural network (FCNN) tailored with a specific architecture to meet our predictive objectives. The model is stacked in this order:

- A fully connected layer of 200 neurons is designed to process 200-time-steps inputs of the condition monitoring feature.
- A rectified linear unit (ReLU) function to introduce non-linearity, enhancing its learning capability.
- A dropout layer is then applied to mitigate the risk of overfitting by randomly omitting a subset of neurons during the training phase.
- For the output layer, a single neuron layer is employed to output the predicted time, encapsulating the RUL estimation.

The FCNN is trained only on the first window of the training signals. This model's outputs are then used to normalise the time feature by dividing all time steps of every window by the predicted EOL value. We employ the Mean Squared Error (MSE) as the loss function and Adamax as the optimiser. The training regimen extends over 1000 epochs, with an Early Stopping mechanism in place to monitor progress. This mechanism halts training if no improvement is observed after 50 epochs, simultaneously recovering the best weight combination observed. This approach ensures that the model remains efficient and effective, capturing the essential predictive dynamics without succumbing to overfitting or underfitting tendencies.

2.4. Convolutional Neural Network (CNN) Model for Feature Transformation

After the labels are generated, the signal is split into windows and normalized over its length, the final step is its transformation. This is handled by the primary model, whose architecture is outlined as follows:

- **Convolutional Layer:** This layer has filters, each with a kernel size of 1, facilitating distinct feature detection across time series data without the need for padding. The Glorot uniform method initializes kernel weights, with biases set to zero.
- **Activation Layer:** This layer utilizes the Rectified Linear Unit (ReLU) for non-linear activation, enhancing the model's learning capabilities.
- **Dropout Layer:** A dropout rate is applied in the training phase to reduce overfitting by randomly omitting connections from the previous layer to the next.
- **Fully Connected Layer:** Encapsulated in a time-distributed wrapper, this layer selects the most relevant outputs from the filters, culminating in the final output.

Figure 2 illustrates the architecture of the CNN model, providing an intuitive understanding of its design and flow. This model provides the transformed CM data, concluding the proposed framework.

3. Case Study

In this part, the methodology proposed in Chapter 2 will be applied, showcasing in detail the steps to transform raw acoustic emission data of CFRP specimens under fatigue loading for predicting their RUL. The chapter starts explaining how

Algorithm 1 Pseudocode of Simulated Monte Carlo data generator (Eleftheroglou, 2020)

Inputs:
$M = \{$
  ζ (int): model's initialization parameters
  θ (array): degradation and the observation parameters
$\}$

Procedure:
$X_0 = 1$
$T_0 = 0$
$T_{age} = 0$
for $(c = 0; c < N; c++)$ **do**
  $i = X_c$
  $s = T_c$
  $j = i + 1$
  $a = U(0, 1)$
  $T_j = \Lambda_{i,j}^{-1}(s, -log(1 - a))$ where $\Lambda_{i,j}(s, t) = \int_0^t \lambda_{i,j}(s, u)\, du$
  $T_{age} = T_{age} + T_j$
  for $(t = T_c + 1; t < T_{age}; t++)$ **do**
    $a = U(0, 1)$
    for $(f = 2; f <= V; f++)$ **do**
      if $\sum_{z=1}^{f-1} b_{X_j}(z) < a < \sum_{z=1}^{f} b_{X_j}(z)$ **then**
        $y_t = f$
      else
        $y_t = 1$
      end if
    end for
  end for
end for

Output:
X_i, T_i (array): respectively the hidden state and the time at the n^{th} transition.
y_t (array): condition monitoring indicator at time $t \in [1, D]$.

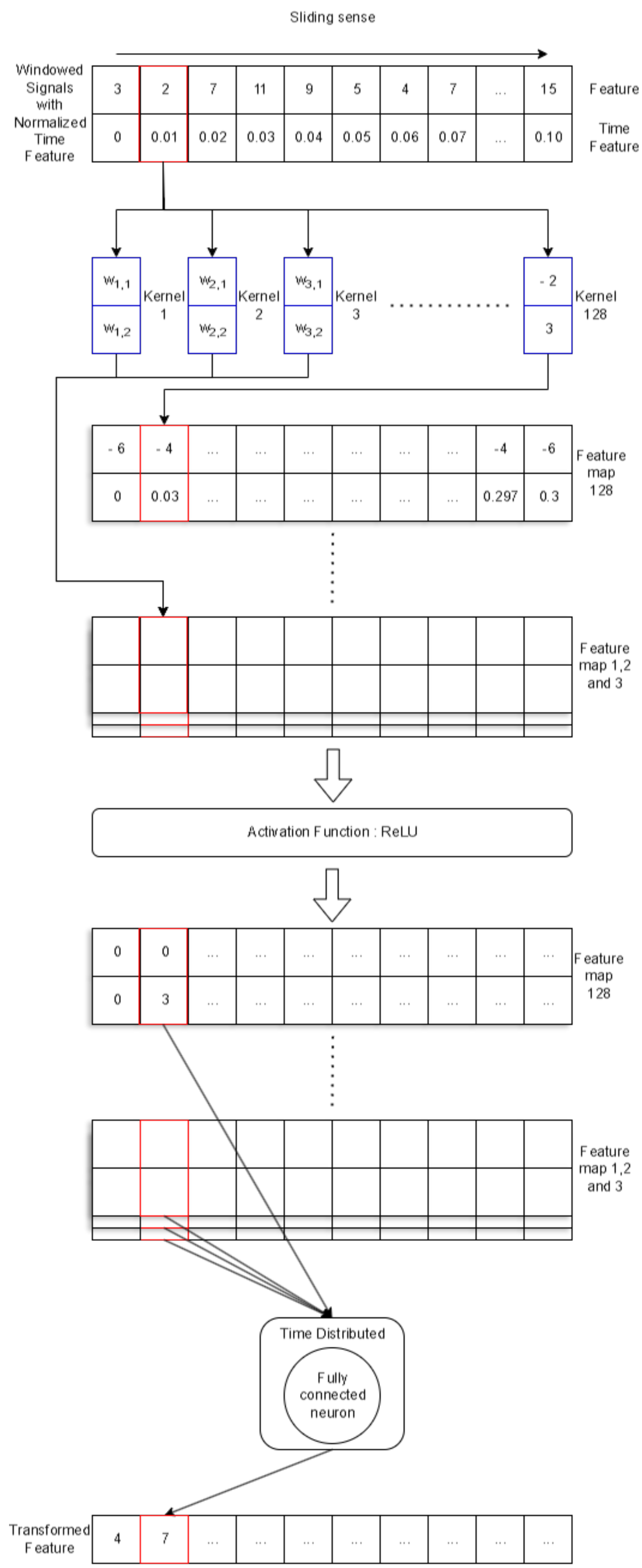


Figure 2. Architecture of the primary CNN model (2.4)

the data are created, preprocessed, labelled, sliced into time windows, and finally transformed.

3.1. Dataset

The acoustic emission (AE) dataset utilised in this research paper comes from the experiments performed in (Eleftheroglou et al., 2020). The dataset is derived from tests on open-hole carbon/epoxy samples, which were subjected to constant amplitude fatigue loading until failure. The specimens of dimensions $400x45mm^2$ are cut from plates manufactured from carbon/epoxy prepregs via the autoclave process. The stacking sequence is a quasi-static lay-up of $[0/45/90/-45]_{2S}$. A hole with a diameter of 10mm was drilled in the center of each specimen. One broadband piezoelectric transducer is attached to each specimen, and with the help of an AMSY-6 Vallen Systeme GmbH, 8-channel AE system, the acoustic emissions of the specimens are recorded and utilized as our CM feature. Hence, the training dataset consists of CM data obtained by utilizing AE sensors from seven samples under fatigue loading, and the testing dataset consists of an eighth specimen that is unseen during training.

3.2. Dataset Preprocessing

The preprocessing phase is designed to enhance the efficiency and effectiveness of model training. The primary objective was to purify the data from noise and scale it appropriately, ensuring that the subsequent steps in our machine-learning pipeline could properly process it. The first required step is to discretize the data. This is achieved by employing the K-means clustering technique (Lloyd, 1982). The entire discretization of the dataset was done by training the K-means with the training dataset, setting the number of clusters to 49, and clustering both the training and test signals with the trained model. This method played a pivotal role in cleaning the data by effectively grouping values into clusters, thereby reducing noise. Each cluster represented a range of values, allowing for a more structured and less noisy dataset. However, the process required careful consideration regarding the number of clusters; an insufficient number could lead to the loss of significant information from the signal. Additionally, this clustering approach ensures that the dataset and the labels are aligned on the same scale. As a final step of the preprocessing phase, we establish a uniform fail value across all signals by setting the final value of each signal feature to 50. The discretized raw data can be seen in Figure 3. By observing the figure, the necessity of transforming the data becomes apparent. The raw feature is highly fluctuating and presents no monotonicity whatsoever. Thus, it cannot be directly used to convey the degradation characteristics of the specimens.

3.3. HSMM initialisation for the case study

This paragraph will discuss the HSMM initialisation required for the Monte Carlo sampling to be performed. To initialise the HSMM, we must define a topology ζ as discussed in chapter 2.1.1.

- **The number of the hidden states (N)** is set at 20 (19 hidden + 1 observed).
- **Connectivity between hidden states (Ω)**: Soft and only left-to-right transitions (meaning that no self-healing or repair actions are modelled) and the final state is observed rather than hidden.
- **Condition Monitoring feature** (I): The connection between the CM feature's values and the hidden states is modelled with a non-parametric discrete probability function, whose values are defined with the emission matrix in the following.
- **Number of discrete monitoring values (V)** are set at 50.
- **Transition rate function (λ)**: For the model of the degradation of the CFRP specimens under fatigue loading, the Weibull failure rate distribution is chosen as displayed in Equation 1.

$$\lambda(t) = \frac{\beta}{\alpha}\left(\frac{t}{\alpha}\right)^{\beta-1} \tag{1}$$

So, in order to fully describe the HSMM model, we need to define the $\theta = \{\Gamma, B\}$ parameters. Γ parameters consist of:

- **Matrix α**: a $(N-1)*N$ matrix of scale parameters for transitions between hidden states. Represented as a diagonal matrix with the first column as zeros. The diagonal is made with logarithmic spacing values from 12 to 6 into 19 values.

$$\alpha = \begin{bmatrix} 0 & 12 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 6 \end{bmatrix}$$

- **Matrix β**: a $(N-1)*N$ matrix of shape parameters for transitions between hidden states. Represented as a diagonal matrix with the first column as zeros as the previous one. The diagonal is made by linear spacing values from 64 to 25 into 19 values.

$$\beta = \begin{bmatrix} 0 & 64 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 25 \end{bmatrix}$$

To complete the initialisation process, we must also create the Emission Matrix **B**. This matrix has dimensions of $N*V$ and displays the probability of the i-th hidden state (rows) emitting

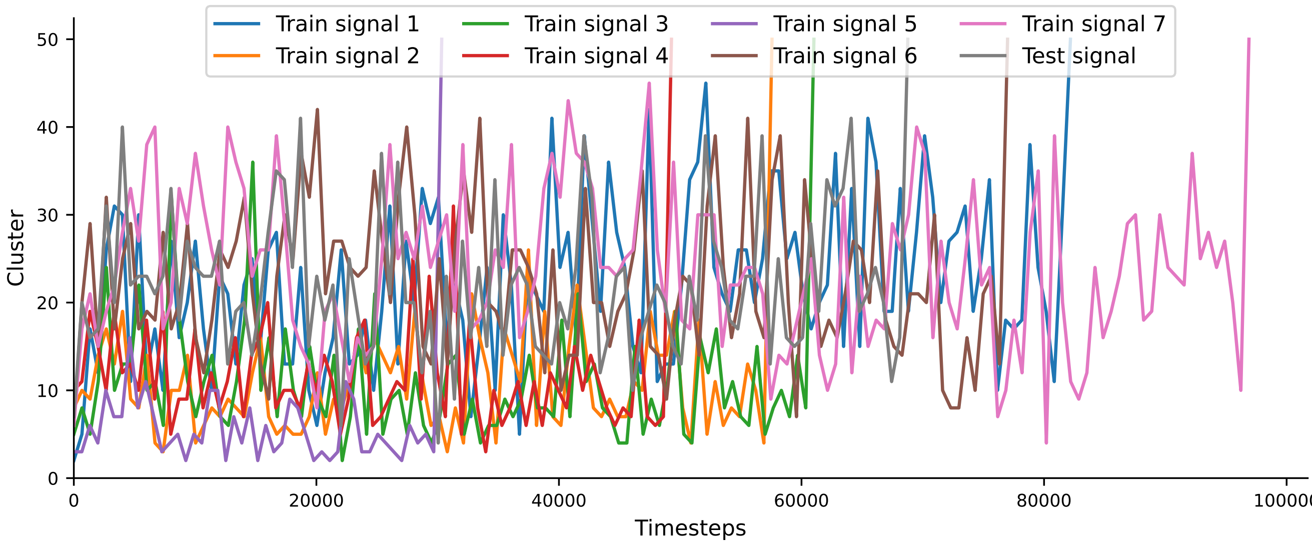


Figure 3. Raw discretized data of the real-world dataset

a specific observable state j-th (columns). The sum of the values in each row always adds up to 1. To fill the values, we have opted for the truncated Gaussian distribution $\mathcal{N}_{T[0,49]}(\mu, \sigma^2)$, which is truncated at 0 and 49 since the failure state is observed, rather than hidden. The standard deviation is equal to 3 for all states, and the mean of $\mathcal{N}_T$ is set in every row in such a way that it increases with the row index. Thus, the mean values of the rows are 19 linearly spaced values in the inclusive $[3, 49]$ range. We have made this decision to ensure that the first hidden states emit the first observable states and the last hidden states emit the last observable states, thus creating a monotonic observation sequence. The emission matrix is presented below:

$$B = \begin{bmatrix} \mathcal{N}^1_{T[0,49]}(3, 3^2) & \mathcal{N}^2_{T[0,49]}(3, 3^2) & \dots & 0 \\ \mathcal{N}^1_{T[0,49]}(5.42, 3^2) & \mathcal{N}^2_{T[0,49]}(5.42, 3^2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}$$

The aforementioned set of parameters is chosen empirically in order for the model to resemble the degradation process of composite specimens under fatigue loading. The user is free to use their own set of parameters that suit their application. So once the parameters are defined, the MCS Algorithm 1 can be run in order to generate the labels.

3.4. Data Windowing of the available data

The data are segmented into fixed-length time windows, as the user specifies. In this research paper, the selected time window is fixed at 200-time steps, corresponding roughly to 10% of the average total length. The size of the windows is chosen intuitively, but it can't be too small since the prediction of the EOL for the normalisation in the following steps will be inaccurate and not too big since doing so will remove the online applicability of the framework. The window advances by one step at each iteration, effectively creating an overlap of 199 timesteps. This allows us to increase the data available for training and testing our transformation model.

3.5. EOL estimations and time feature normalisation of the available data

As previously highlighted, the model's need to predict the EOL derives from our aim to normalize the time feature for the training of the transformation model. The results are presented in Table 1. It can be seen that the error of the estimations varies across the train specimens, with errors up to 20%. However, since this model is used to provide an initial rough estimation of the EOL for the normalization of the time feature, as presented in Section 4, its estimations are more than adequate.

Table 1. EOL Results with Corrected Error Percentages

Signal	Predicted	Actual	Error (%)
Train 1	84128	82176	2.38
Train 2	54304	57568	-5.67
Train 3	48320	61024	-20.82
Train 4	50624	49280	2.73
Train 5	34272	30336	12.97
Train 6	84416	76992	9.64
Train 7	96448	96896	-0.46
Test	77152	68768	12.19

3.6. CNN model architecture for the available data transformation

In developing our 1D CNN model, the choice of hyperparameters, loss function, and optimizer was deliberate and aimed at optimizing performance for our specific dataset characteristics.

- The decision to employ a kernel of dimension 1 ensures that the filter remains unaffected by padding, maintaining the feature map's dimensionality identical to the input. This approach guarantees continuity between successive windows, avoiding noisy spikes at the prediction's beginning and end. Given our transformation goal, this is a critical factor: the prognostic model can lead to wrong predictions.
- For our loss function, Mean Squared Error (MSE) was selected to precisely track the fluctuating nature of our labels, aiming for a regression model that closely mirrors the original data.
- After experimenting with various optimizers, Adamax emerged as the most effective, offering superior convergence properties for our scenario.
- The model architecture was kept minimal with a single CNN layer, a choice driven by the limited size of our signal dataset. This simplicity facilitated a more effective training process compared to deeper models.
- To counteract overfitting due to the high redundancy among the time windows (as detailed in Section 3.4), we implemented L1 regularization and dropout at standard values.
- Due to the non linearity of the labels, CNN is equipped with Rectified Linear Unit.

These choices collectively formed a robust framework for our model, tailored to the unique demands of our data.

3.7. Prognostics

To showcase the effectiveness of the proposed feature extraction for PHM tasks, the Remaining Useful Life (RUL) of the specimens will be predicted by utilising an HSMM. Any prognostic model can be utilized in this step since the data transformation framework is independent of the prognostic model. However, since the degradation process is modelled with an HSMM for the label generation, it is a straightforward choice to utilize a model from the same family. For the prognostics, the explicit duration modification to the HSMM is chosen. Thus, following the initialization procedure explained in Section 2.1.1, the following parameters are defined:

- **The number of the hidden states** (N): It's considered a hyperparameter of the model, and in order to pick a value, the elbow method utilizing the Bayesian information criterion (BIC) was utilized. The optimal number of states was found to be 7.
- **Transition between hidden states** ($\boldsymbol{\Omega}$): soft and left-to-right transitions, no self transitions are allowed.
- **Start probability matrix** ($\boldsymbol{\pi}$): the process always starts from the first state.
- **Transition rate function** ($\boldsymbol{\lambda}$): is assumed to be non-parametric and depends only on the current state.
- **Condition Monitoring feature** (I): The connection between the hidden states and the values of the CM features are assumed to be described by Gaussian distributions $\mathcal{N}(\mu, \sigma^2)$ and represented by a mean and a standard deviation observation vector.
- **CM indicator space** (Z): since the observation process is modelled with a continuous probability distribution (Gaussian), the indicator space consists of all the real numbers ($Z = \{z \in \Re\}$).

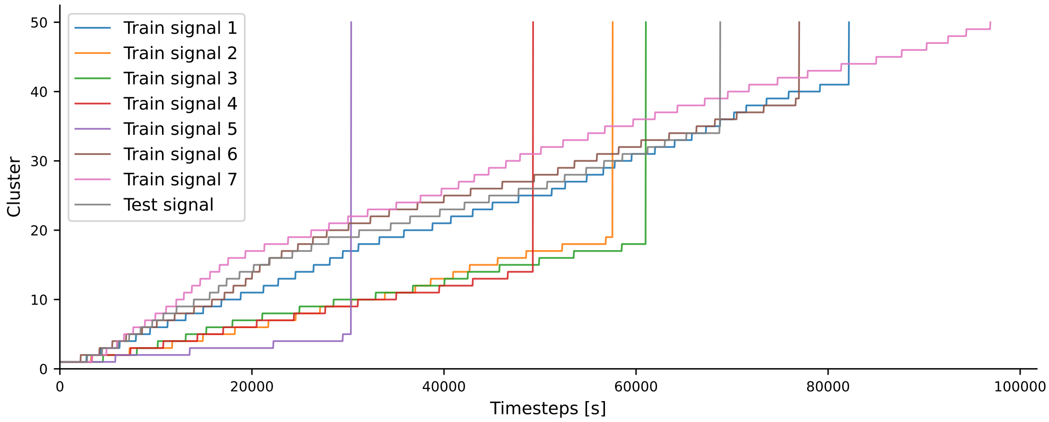
After the parameters are initialised, the parameter estimation can be performed as described in (Yu, 2010). When the optimal parameters have been estimated, they are utilised in order to predict the RUL of the asset, following the procedure in (Dong, He, Banerjee, & Keller, 2006)

4. Results

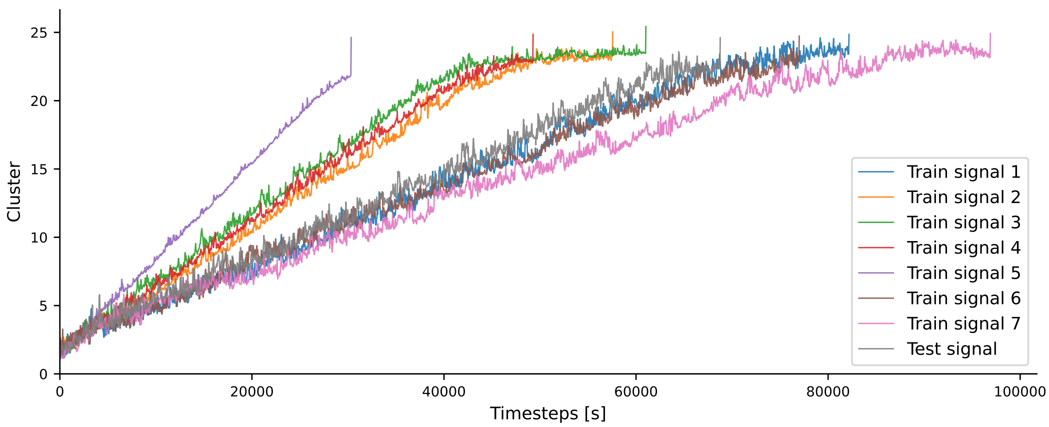
In the first part of the current section, the transformed data utilizing the proposed framework are presented and contrasted against the raw data and their cumulative transformation, highlighting the performance of the framework. Finally, the prognostic findings from the HSMM of the cumulative feature and the one obtained from our framework are contrasted.

4.1. FE results

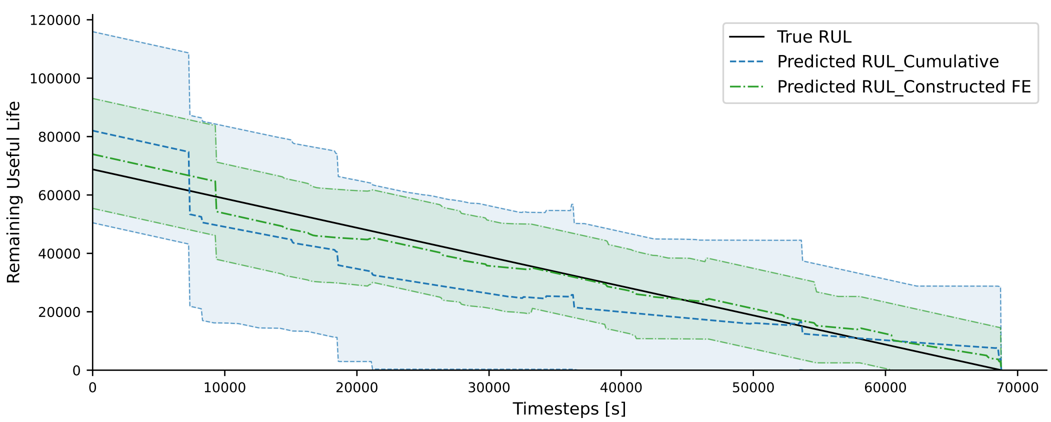
As a baseline for the proposed transformation, the cumulative transformation of the discretized raw data is calculated and presented in Figure 4a. This choice is justified as the authors consider it to be the most straightforward choice for transforming noisy and highly fluctuating data into monotonic ones. The necessity of transforming the data in the first place lies in the inability of prognostic algorithms to provide any meaningful results when applied to fluctuating data that present no monotonic behaviour. In Figure 4b, the transformed data utilizing the proposed framework are showcased. By comparing the two, it can be observed that even though the cumulative features present highly monotonic behaviour (as expected due to the cumulative summation function), there is a great uncertainty associated with the last value directly before failure (also known as prognostability). This is where an added contribution of the proposed transformation also lies. It can be seen (Figure 4b) that the transformed signals (both training and test) fail at values that are very close to each other. This is expected to, in turn, come with reduced uncertainty when it comes to the RUL prediction values, which remains to be seen in the following section.



(a) Cumulative feature



(b) Constructed feature with the proposed methodology



(c) Comparative plot for the test sample

Figure 4. Plots for the results of the proposed methodology

4.2. Prognostic results

In Figure 4c, the prognostic results of the test sample utilising both the cumulative feature and the proposed one are presented on the same plot for comparison reasons. We can see that not only the mean value predictions of the RUL using the constructed feature are closer to the true RUL, but also that the 90% confidence intervals are reduced. This is the manifestation of the main contribution of the constructed feature, which is the reduced uncertainty of the final values of the constructed feature compared to the cumulative one. Hence, the goal of a non-complex FE method that aids in the realization of accurate and highly confident PHM frameworks is achieved.

5. CONCLUSIONS

This research introduced a methodology integrating Monte Carlo simulated data with CNN to enhance prognostic performance in predicting system degradation by incorporating the stochastic nature of system deterioration and the noisy measurements in the labels for transforming raw data. Its theoretical viability has been demonstrated, as well as its practical applicability, particularly in its ability to operate online, making RUL predictions for CFRP specimens under fatigue loads, based on noisy AE measurements. It is worth noting that due to the simplicity of the proposed framework, given enough data and a proper HSMM initialization for the MCS (following the procedure showcased in Section 3.3), the applicability to more complex systems is a straightforward procedure. The main contribution of the proposed framework lies in its simplicity of deliberately combining well-established and non-complex components in a novel way that alleviates the deterministic labelling based on intuitively picked metrics in extracting suitable features for PHM applications. This led to the creation of a simple and efficient model that effectively transforms raw and noisy data for accurate and high-confidence prognostics. Our motivation was simple: We consider the labelling of signals that are by nature stochastic with deterministic labels to be ill-posed. Rather, we proposed the generation of labels by sampling a stochastic model (HSMM) in a framework that is independent of the prognostic algorithms and can be applied online. We aim to expand this framework to be able to fuse different CM features and compare its performance against numerous traditional FE methods for prognostic tasks.

REFERENCES

Chen, D., Qin, Y., Wang, Y., & Zhou, J. (2021). Health indicator construction by quadratic function-based deep convolutional auto-encoder and its application into bearing rul prediction. *ISA transactions*, *114*, 44–56.

Coble, J., & Hines, J. W. (2009). Identifying optimal prognostic parameters from data: a genetic algorithms approach. In *Annual conference of the phm society* (Vol. 1).

Dong, M., He, D., Banerjee, P., & Keller, J. (2006). Equipment health diagnosis and prognosis using hidden semi-markov models. *The International Journal of Advanced Manufacturing Technology*, *30*, 738–749.

Eleftheroglou, N. (2020). *Adaptive prognostics for remaining useful life of composite structures* . Delft University of Technology. (Thesis (Ph.D.))

Eleftheroglou, N., Zarouchas, D., & Benedictus, R. (2020). An adaptive probabilistic data-driven methodology for prognosis of the fatigue life of composite structures. *Composite Structures*, *245*, 112386.

Lemieux, C. (Ed.). (2009). *Monte carlo and quasi-monte carlo sampling*. Springer.

Lloyd, S. (1982). Least squares quantization in pcm. *IEEE transactions on information theory*, *28*(2), 129–137.

Moradi, M., Broer, A., Chiachío, J., Benedictus, R., & Zarouchas, D. (2023). Intelligent health indicators based on semi-supervised learning utilizing acoustic emission data. In P. Rizzo & A. Milazzo (Eds.), *European workshop on structural health monitoring* (pp. 419–428). Cham: Springer International Publishing.

Xu, Z., Bashir, M., Liu, Q., Miao, Z., Wang, X., Wang, J., & Ekere, N. (2023). A novel health indicator for intelligent prediction of rolling bearing remaining useful life based on unsupervised learning model. *Computers & Industrial Engineering*, *176*, 108999.

Ye, Z., Zhang, Q., Shao, S., Niu, T., & Zhao, Y. (2022). Rolling bearing health indicator extraction and rul prediction based on multi-scale convolutional autoencoder. *Applied Sciences*, *12*(11), 5747.

Yu, S.-Z. (2010). Hidden semi-markov models. *Artificial intelligence*, *174*(2), 215–243.