
A Prototype Similarity-based System for Remaining Useful Life Estimation for Future Industry by Singular Spectrum Analysis-Long Short Term Memory Neural Networks Algorithm

Prakit Intachai^{1,*} and Peerapol Yuvapoositanon²

¹*The Electrical Engineering Graduate Program, Faculty of Engineering, Mahanakorn University of Technology, 140 Cheumsamphan Rd., Nong-chok Bangkok 10530, Thailand*

²*Mahanakorn Institute of Innovation, Faculty of Engineering, Mahanakorn University of Technology, 140 Cheumsamphan Rd., Nong-chok Bangkok 10530, Thailand*

E-mail: prakit2554@gmail.com; peerapol@mutacth.com

**Corresponding Author*

Received 29 November 2019; Accepted 26 May 2020;
Publication 17 August 2020

Abstract

In this paper, we propose a prototype similarity-based approach of estimating the remaining useful life (RUL) of turbofan engine data using the singular spectrum analysis and the long-short term memory (SSA-LSTM) neural networks algorithm. The algorithm consists of two steps. First, the optimal window length of the trajectory matrix of the dataset is empirically determined from a prototype dataset. Second, the estimation of the RUL of the target datasets is performed using the window length parameter obtained from the first step. The validity of the proposed algorithm is verified by testing with 200 turbofan engine datasets. The results are shown to have a significant improvement in the performance of the RUL estimation over the existing LSTM algorithm.

Journal of Mobile Multimedia, Vol. 16.1-2, 181–202.

doi: 10.13052/jmm1550-4646.16129

© 2020 River Publishers

Keywords: Remaining useful life, Singular spectrum analysis, Long-short term memory, Optimal window length.

1 Introduction

Recently, predictive approaches for estimating remaining useful life (RUL) of a device or a machine have gained increasing interest in the maintenance perspectives [1, 2]. The reliability of these approaches depends on the accuracy of the RUL estimation. However, the RUL estimation is generally difficult in practice because some devices may not be suitable for physics-based estimation of true RUL when the real-life system is complex [3], or the characteristic of some time series data is described as non-linear [4]. Different techniques can be used in order to estimate the RUL, such as the data-driven methodology in Ref. [5], and similarity methods presented in Refs. [6–8]. The data-driven models are more suitable in some cases where supported models for the RUL are not achievable.

The singular spectrum analysis (SSA) is a decomposition technique for general time series [9]. The SSA decomposition process is data-adaptive and does not involve any harmonic function. Therefore, the SSA is suitable to perform on non-linear and non-stationary time series. The noise of time series in the process of the SSA can be described as definition of window length (L) [10] when L is an integer for creating the trajectory matrix. In Ref. [11], shown the estimation of $L \sim T/2$ is shown when T is the total number of data in the time series. The estimated window length (L_{est}) or $L \sim T/2$ is a simple method of choosing L . Elsner and Tsonis [12] present some dialogue and remark that choosing $L_{\text{est}} = T/4$ is a common method, and L from that method is not larger than $L_{\text{est}} = T/2$. A method of L selection states that a suitable parameter for L should depend on $L_{\text{est}} < T/2$ as presented in Ref. [13]. In Ref. [14], L selection is described as $L_{\text{est}} = \log(T)^c$ with $c \in (1.5, 3.0)$. Moment test for L selection in the SSA [15] is presented for the method of determining L_{est} by a statistical test, with $L_{\text{est}} > 2\sqrt{T}$ described as the estimated window length. Therefore, in order to do that each of the datasets must be estimated separately. For a set of time series, there is no existing approach suggesting an estimated window length optimally for the group of datasets. However, the length L is large when a time series is long. Also, for a system that has to deal with many time series, it is not practical for determining L for every time series.

In this paper, we propose an algorithm for the RUL estimation of multiple target time series with the LSTM network by means of the basis functions decomposed from the prototype time series using the SSA. The LSTM is adopted in our approach since its usages in the RUL estimation have been widely studied in the literature [16, 17]. By this approach, determining L for the target time series can be performed using the information derived from the prototype time series. The window length L optimal or L_{opt} for the prototype time series is achieved at the point where the root mean square error (RMSE) obtained from the LSTM is minimized. For the target time series, which will be shown later, that can be reconstructed from the prototype basis function with scaling factor, the same optimal window length L_{opt} can also be applied. The proposed system offers the advantage of determining the optimal parameters for a large number of time series using only those determined from the representative one called the prototype. The proposed technique is therefore a multi-step approach for the RUL estimation of a target time series using the LSTM with transformed basis functions obtained from the prototype time series. Since the subcomponents of the prototype time series are obtained from the SSA, this algorithm is called the SSA-LSTM algorithm.

The organization for the remaining of the paper is as follows. In Section 2, the definition of the trajectory matrix of a dataset, the window length and the basis functions are presented. In Section 3, the relationships between the prototype and target datasets via the basis functions and their coefficients are described. The construction of the SSA-LSTM algorithm and the determination of the window length are defined. Finally, Section 4 shows how the validity of the proposed algorithm in the prototype similarity-based RUL estimation is tested for the turbofan datasets acquired from Ref. [18]. Finally, conclusions are provided in Section 5.

2 Methodology

2.1 The Diagonal Averaging Parameters and their Coefficients

The ability of the SSA is mentioned in Ref. [10]. The SSA can be applied to the stationary and non-stationary time series data because it can describe what time series with no a priori information about the data structure. The SSA technique is based on the singular value decomposition (SVD) of the trajectory matrix of the SSA [10]. The model of the trajectory matrix

can be expressed as

$$\mathbf{X} = \begin{bmatrix} x(1) & x(2) & x(3) & \cdots & x(K) \\ x(2) & x(3) & x(4) & \cdots & x(K+1) \\ x(3) & x(4) & x(5) & \cdots & x(K+2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x(L) & x(L+1) & x(L+2) & \cdots & x(T) \end{bmatrix}, \quad (1)$$

where \mathbf{X} denotes the trajectory matrix and $\mathbf{x}(t)$, $t = 1, 2, \dots, T$ is the time series data of interest. L is the window length and $1 < L < T - 1$ defines the number of rows in \mathbf{X} . K is the embedding dimension and $K = T - L + 1$ where T is the time interval of the observed time series data. The trajectory matrix of \mathbf{X} is processed by the SVD, and the matrices \mathbf{U} , \mathbf{D} and \mathbf{V}^T are derived in the process of decomposition as

$$\mathbf{UDV}^T = \text{svd}(\mathbf{X}), \quad (2)$$

where \mathbf{D} denotes the eigenvalues matrix of $\text{svd}(\mathbf{X})$, and the value of eigenvalues is expressed in the diagonal matrix as $\mathbf{D} = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_L)$. \mathbf{U} and \mathbf{V} are the eigenvectors matrices. $\hat{\mathbf{X}}_i$ denotes the i^{th} eigentriple that is determined by $\hat{\mathbf{X}}_i = \lambda_i \mathbf{U}_i \mathbf{V}_i^T$.

The eigentriple $\hat{\mathbf{X}}_i$ can be applied to the process of eigentriple grouping and diagonal averaging of the SSA [10, 11]. $\theta_i(t)$ describes the result from the process of diagonal averaging as $\hat{\mathbf{X}}_i$ where L is the parameter to be determined.

3 Prototype and Target Time Series

3.1 The Basis Function and its Coefficient

In this section, we explain the idea of how to represent a time series, called the target, using the basis functions derived from another time series, called the prototype.

The hypothesis underlying this idea stems from the concept of basis functions in Fourier series of a periodic time series [19]. First, the estimates of the prototype and the target time series are determined from

$$\hat{\mathbf{x}}^{(pro)}(t) = \sum_{i=1}^{T-1} \theta_i^{(pro)}, \quad (3)$$

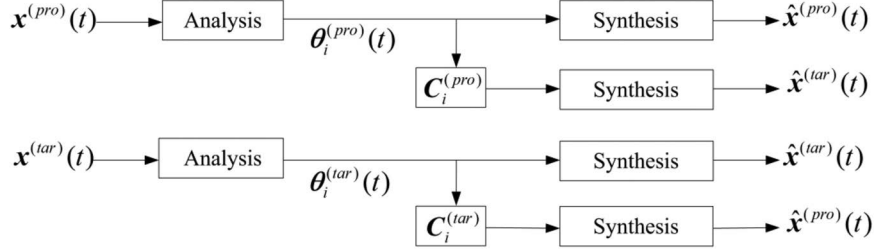


Figure 1 The concept of representing the target time series $\hat{\mathbf{x}}^{(tar)}(t)$ from the diagonal averaging parameters $\theta_i^{(pro)}(t)$, and their coefficients $\mathbf{C}_i^{(pro)}$ from the prototype time series through the analysis and synthesis parts of SSA, is illustrated. The reverse operation, i.e., representing the prototype from the target, is also shown.

$$\hat{\mathbf{x}}^{(tar)}(t) = \sum_{i=1}^{T-1} \theta_i^{(tar)}, \quad (4)$$

where $\theta_i^{(pro)}$ and $\theta_i^{(tar)}$ denote the diagonal averaging results from the basis function $\hat{\mathbf{x}}^{(pro)}$ and $\hat{\mathbf{x}}^{(tar)}$ of the prototype and the target, respectively.

Now the relationship between the prototype and the target can be described as

$$\hat{\mathbf{x}}^{(pro)}(t) = \sum_{i=1}^{T-1} \mathbf{C}_i^{(tar)} \theta_i^{(tar)}(t), \quad (5)$$

$$\hat{\mathbf{x}}^{(tar)}(t) = \sum_{i=1}^{T-1} \mathbf{C}_i^{(pro)} \theta_i^{(pro)}(t), \quad (6)$$

where

$$\mathbf{C}_i^{(pro)} = \text{diag}(\theta_i^{(pro)}(t))^{-1} \text{diag}(\theta_i^{(tar)}(t)) \quad (7)$$

is the coefficient matrix required for reconstructing $\hat{\mathbf{x}}^{(pro)}(t)$ from $\theta_i^{(tar)}(t)$ and

$$\mathbf{C}_i^{(tar)} = \text{diag}(\theta_i^{(tar)}(t))^{-1} \text{diag}(\theta_i^{(pro)}(t)) \quad (8)$$

is the coefficient matrix required for reconstructing $\hat{\mathbf{x}}^{(tar)}(t)$ from $\theta_i^{(pro)}(t)$. This relationship can be depicted through the analysis and synthesis operations of the SSA in Algorithm 1.

Algorithm 1 Algorithm for estimating the coefficients of the prototype and target time series.

Input: $\theta(t)$ // Diagonal averaging parameter.
Define: t // Index of time series.
 T // End index of time series.
 i // Index of window length.
 L // The window length.

Check value of input $\theta(t)$.
For $t=1$ to T
 IF $\theta(t) < \text{Threshold}$
 Set $\theta(t) = \text{Threshold}$
 else
 Set $\theta(t) = \theta(t)$
 End
End
 Diagonalization of $\theta(t)$ for $\text{diag}(\theta(t))$ and determine its inverse.
While $i \leq L$ **do**
 Calculate the i^{th} coefficient matrix of the prototype $\mathbf{x}^{(pro)}(t)$, $\mathbf{C}_i^{(pro)}$ as in (7).
End
While $i \leq L$ **do**
 Calculate the i^{th} coefficient matrix of the prototype $\mathbf{x}^{(tar)}(t)$, $\mathbf{C}_i^{(tar)}$ as in (8).
End
 Calculate $\hat{\mathbf{x}}^{(pro)}(t)$ from (5).
 Calculate $\hat{\mathbf{x}}^{(tar)}(t)$ from (6).
Results: $\mathbf{C}_i^{(pro)}$, $\mathbf{C}_i^{(tar)}$, $\hat{\mathbf{x}}^{(pro)}(t)$, $\hat{\mathbf{x}}^{(tar)}(t)$.

3.2 Reconstruction of a Time Series from a Group of Subcomponents

Operating on the separated groups of the SSA data can lead to an enhanced performance [20, 21]. Therefore, we divide the analysis components into three groups as follows

$$\hat{\mathbf{x}}_p(t) = \begin{cases} \hat{\mathbf{x}}_1(t) = \mathbf{C}_1\theta_1(t) \\ \hat{\mathbf{x}}_2(t) = \sum_{i=2}^B \mathbf{C}_i\theta_i(t) \\ \hat{\mathbf{x}}_3(t) = \sum_{i=B+1}^L \mathbf{C}_i\theta_i(t) \end{cases}, \quad (9)$$

where $\hat{\mathbf{x}}_p(t)$ denotes the p^{th} group of the sub-component $\hat{\mathbf{x}}(t)$ which in turn is the estimated version of $\mathbf{x}(t)$ and B is an integer designating the number

of components in the second group, $p = 2$. The choice of B can be changed as $B = 3, 4, 5$ or $B = 3, 4, \dots, L - 1$. From (9), the estimate of $\mathbf{x}(t)$ can be achieved by the summation of the three groups:

$$\hat{\mathbf{x}}(t) = \sum_{p=1}^3 \hat{\mathbf{x}}_p(t) \quad (10)$$

3.3 Estimation Model for Reconstructed Data

The combination of the SSA and neural networks has been studied in the literature [22, 23] and the results of estimation performance are improved as compared to those from using only the neural networks. It was shown that the LSTM is promising for the estimation of trendy time series [16, 17]. Since the RUL datasets of multiple turbo fan engines time series [18] is trendy, the LSTM is then adopted as the neural network algorithm of the estimation part in our proposed SSA-LSTM architecture for the RUL estimation.

In Figure 2, the schema of the SSA-LSTM architecture is illustrated. The datasets, which are arranged as time series, are categorized as features of $\mathbf{x}^{(F)}(t)$ and their associated RULs $\mathbf{x}^{(RUL)}(t)$ have gone through the analysis and synthesis of the SSA resulting in groups of the reconstructed datasets $\hat{\mathbf{x}}_p^{(F)}(t)$, and $\hat{\mathbf{x}}_p^{(RUL)}(t)$ for $p = 1, 2, 3$ for feature and RUL datasets, respectively. $\hat{\mathbf{x}}^{(RUL)}(t)$ is the training datasets to the LSTM networks whose outputs $\mathbf{y}_p^{(RUL)}(t)$, $p = 1, 2, 3$ are the estimation results of the three LSTM networks.

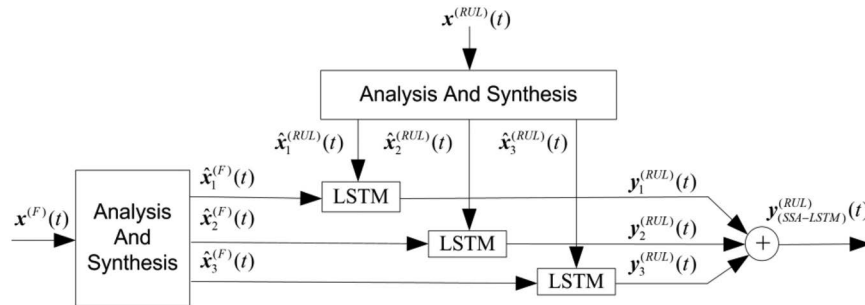


Figure 2 The proposed SSA-LSTM algorithm for RUL estimation is illustrated. Three sub-components of features, $\hat{\mathbf{x}}_p^{(F)}(t)$ $p = 1, 2, 3$ and $\hat{\mathbf{x}}_p^{(RUL)}(t)$ $p = 1, 2, 3$ from analysis and synthesis parts of SSA are used by the LSTM in order to generate the estimated RUL $\mathbf{y}_{(SSA-LSTM)}^{(RUL)}(t)$.

The estimate of the true RUL is simply the combination of $\mathbf{y}_p^{(RUL)}(t)$:

$$\mathbf{y}_{(SSA-LSTM)}^{(RUL)}(t) = \sum_{p=1}^3 \mathbf{y}_p^{(RUL)}(t) \quad (11)$$

The process of determining the optimal value of L , the L_{opt} , involves analysis and synthesis of both the features and true RUL of the prototype turbofan engine datasets [18]. The three LSTM networks are fed with the synthesized sub-components of the features $\hat{\mathbf{x}}^{(F)}(t)$ and are trained with the synthesized sub-components of the true RUL. The root mean squared errors (RMSE) of the prediction for all possible values of L up to $T/2$, i.e., $L_{est} = T/2$, are measured. The values of L at the minimum RMSE is then selected as the optimal data-dependent L , i.e., L_{opt} .

Algorithm 2 The Singular Spectrum Analysis-Long short Time Memory (SSA-LSTM) Algorithm

- Input:** $\mathbf{x}^{(F)}(t)$ // Normalized feature data of training and test data.
 $\mathbf{x}^{(RUL)}(t)$ // The RUL data set associating $\mathbf{x}^{(F)}(t)$ of training and testing data sets.
- Define:** $\hat{\mathbf{x}}_p^{(F)}(t)$ // The p^{th} normalized sub-component feature data.
 $\hat{\mathbf{x}}_p^{(RUL)}(t)$ // Sub-component of RUL data sets.
- Training:** Analysis and synthesis by SSA at $\mathbf{x}^{(F)}(t)$ and $\mathbf{x}^{(RUL)}(t)$ of training data.
Determine $\hat{\mathbf{x}}_p^{(F)}(t)$ and $\hat{\mathbf{x}}_p^{(RUL)}(t)$ of training data from (9).
Train the LSTMs with the RUL data sets by $\hat{\mathbf{x}}_p^{(F)}(t)$ and $\hat{\mathbf{x}}_p^{(RUL)}(t)$ of training data.
- Testing:** Analysis and synthesis by SSA at $\mathbf{x}^{(F)}(t)$ of test data.
Determine $\hat{\mathbf{x}}_p^{(F)}(t)$ of test data from (9).
Estimating RULs by $\hat{\mathbf{x}}_p^{(F)}(t)$ of test data.
Calculate $\mathbf{y}^{(RUL)}(t)$ from (11).
- Result:** $\mathbf{y}^{(RUL)}(t)$.
-

We also show that considering the similarity between the prototype and the targets, L_{opt} that is obtained from the prototype can be applied to the analysis and synthesis parts of the SSA of the targets as well.

4 Experimental Results

4.1 The Data Sets

The data sets for the experiments were the turbofan engine degradation simulation dataset provided by the Prognostics CoE at the NASA Ames Research

Algorithm 3 Algorithm for determining the optimal window length, L_{opt} .

Input: L // Window length.
 $\mathbf{x}^{(pro)}(t)$ // Prototype data.
Define: T // End index of time series.
 $L_{est} = T/2$ // Estimated window length.
 $\hat{\mathbf{x}}^{(pro)}(t)$ // Estimated time series data of prototype data.
 $RMSE_i$ // The i^{th} RMSE that is calculated from (15).
For $i=3$ **to** L_{est}
 Take $\mathbf{x}_i^{(pro)}(t)$ to the process of analysis and synthesis in SSA.
 Determine $\hat{\mathbf{x}}_i^{(pro)}(t)$ from (9).
 Estimate the RUL by LSTM and calculate $RMSE_i$ from (15).
End
 Find the index that produces the minimum of $RMSE_i$ and select the index for the optimal window length L_{opt} .
Result: L_{opt} .

Center [18]. In the datasets obtained from Ref. [18], there are 200 datasets recorded from several sensors under different conditions of both normal and fault modes. Features and RULs datasets are also provided. In order to cope with different scales experienced in the datasets, we apply the normalization process before performing the RUL estimation. Ref. [24] describes how the turbofan engines are used. Let $\mathbf{x}''^{(F)}(t)$ and $\bar{\mathbf{x}}''^{(F)}(t)$ be a raw dataset and its time average respectively and σ be the standard deviation of $\mathbf{x}''^{(F)}(t)$ from various sensor measurements, then its feature and normalized version can be calculated by

$$\mathbf{x}'^{(F)}(t) = \frac{(\mathbf{x}''^{(F)}(t) - \bar{\mathbf{x}}''^{(F)})}{\sigma}, \quad (12)$$

$$\mathbf{x}^{(F)}(t) = \frac{\mathbf{x}'^{(F)}(t) - \mathbf{x}'_{\min}(t)}{\mathbf{x}'_{\max}(t) - \mathbf{x}'_{\min}(t)}, \quad (13)$$

where $\mathbf{x}'^{(F)}(t)$ and $\mathbf{x}^{(F)}(t)$ are the features and its normalized version of the raw data $\mathbf{x}''^{(F)}(t)$, respectively. The maximum and the minimum of $\mathbf{x}'^{(F)}(t)$ are defined by $\mathbf{x}'_{\max}(t)$ and $\mathbf{x}'_{\min}(t)$, respectively. The normalized feature $\mathbf{x}^{(F)}(t)$ is fed to the analysis and synthesis operations by the SSA to produce the basis functions act as in Figure 2.

In Figures 3(a)–(c), an example of $\mathbf{x}''^{(F)}(t)$ is compared with its feature and normalized version, $\mathbf{x}^{(F)}(t)$. In this section, we explain the idea of how to represent a time series, called the target.

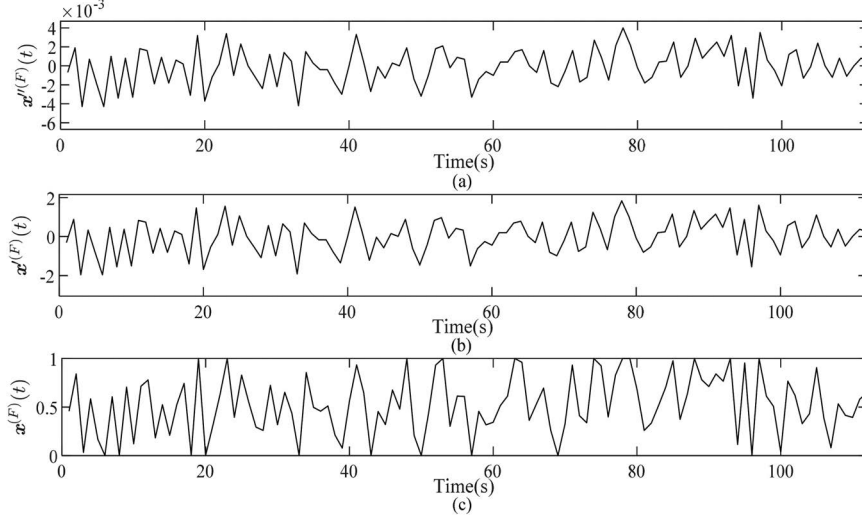


Figure 3 An example of a date set to be used as a feature which is categorized into (a) raw dataset, (b) the feature and (c) the normalized feature.

4.2 Optimal Window Length of Prototype Data and Performance Metrics

In this section, we provide an empirical approach in determining the optimal window length L_{opt} and the value of B of the prototype data. The performance metrics are the mean absolute error (MAE) and the RMSE defined respectively as

$$MAE = \frac{1}{T} \sum_{t=1}^T \mathbf{x}^{(RUL)}(t) - \mathbf{y}_{(SSA-LSTM)}^{(RUL)}(t), \quad (14)$$

$$RMSE = \sqrt{\frac{1}{T} \left(\sum_{t=1}^T \mathbf{x}^{(RUL)}(t) - \mathbf{y}_{(SSA-LSTM)}^{(RUL)}(t) \right)^2}, \quad (15)$$

where $\mathbf{x}^{(RUL)}(t)$ is the true RUL data and $\mathbf{y}_{(SSA-LSTM)}^{(RUL)}(t)$ is the RUL estimate of the SSA-LSTM algorithm as shown in Figure 2. In our similarity-based approach, L_{opt} is calculated from a dataset, namely the prototype data, selected from the 200 datasets. The true RUL datasets contain $T = 116$ records. From Ref. [11], the estimate maximum range of the window length to be used for the trajectory matrix X in (1), L_{est} , is determined by $L_{\text{est}} = T/2$,

i.e., $L_{est} = 58$. The RMSE of the estimated RUL of the prototype data for $3 \leq L \leq 57$ is plotted in Figure 4. It is noted that for this particular prototype data, L_{opt} is 24 since the minimal RMSE is attained at that point. This value of L_{opt} coincides with the suggested values of $L_{opt} < T/2$ of Ref. [13], $L_{opt} = T/4$ of Ref. [12], $L_{opt} = \log(T)^c$ of Ref. [14] and $L_{opt} > 2\sqrt{T}$ of Ref. [15]. In finding the value of B , we fixed $L_{opt} = 24$ and varied B from 3 to 23. However, as shown in Figure 5, the sensitivity of the RMSE of the estimation RUL to B is low. This means that we can choose from a wide degree of freedom.

In Figures 6 and 7, the estimation of the basis functions of the $\mathbf{x}_p^{(pro,RUL)}(t)$ and the estimation of true RUL $\mathbf{x}^{(pro,RUL)}(t)$ of the prototype are shown respectively using $L_{opt} = 24$ and $B = 9$.

In Figure 6, the thin lines are the true RUL basis functions $\mathbf{x}_p^{(pro,RUL)}(t)$ and the thick lines are the estimated true RUL outputs from the LSTM networks $\mathbf{y}_p^{(pro,RUL)}(t)$. As shown in Figure 6(a), the estimated RUL for $p=1$, i.e., $\mathbf{y}_1^{(pro,RUL)}(t)$ is able to track the true RUL $\mathbf{x}_1^{(pro,RUL)}(t)$. However, for the second and third estimated basis functions, i.e., $\mathbf{y}_2^{(pro,RUL)}(t)$ and $\mathbf{y}_3^{(pro,RUL)}(t)$, respectively, the tracking errors are higher than the first.

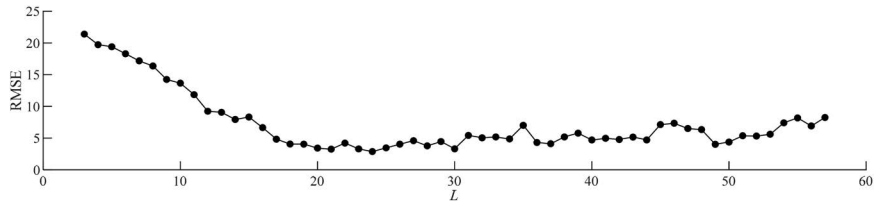


Figure 4 The RMSE of estimated RUL for prototype data for $3 \leq L \leq 57$.

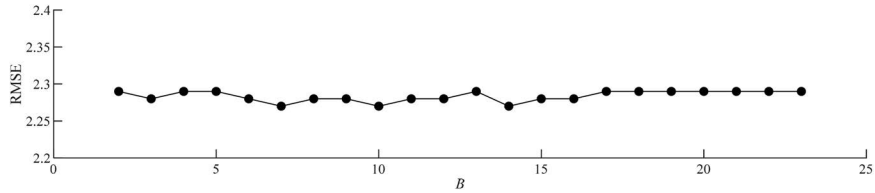


Figure 5 RMSE of the estimated RUL for various values of B .

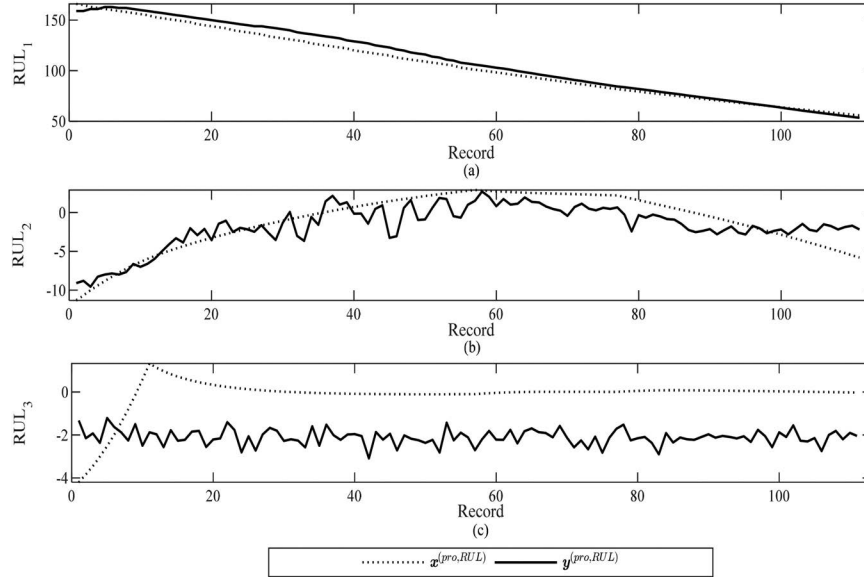


Figure 6 Estimated RULs of three sub-components for the prototype data, $\mathbf{y}_p^{(pro,RUL)}(t)$, are provided in (a) $\mathbf{y}_1^{(pro,RUL)}(t)$, (b) $\mathbf{y}_2^{(pro,RUL)}(t)$ and (c) $\mathbf{y}_3^{(pro,RUL)}(t)$.

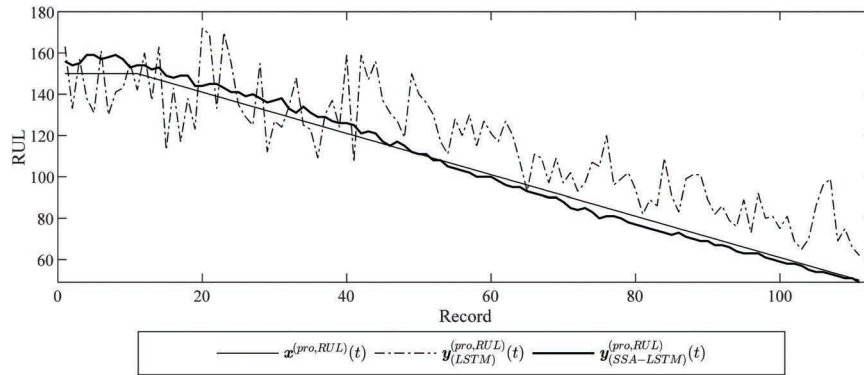


Figure 7 Estimated RUL of prototype data by SSA-LSTM using the optimal windows length $L_{opt} = 24$ and $B = 9$.

Nevertheless, after all the estimated RULs for the sub-components $\mathbf{y}_p^{(pro,RUL)}(t)$ are combined to form $\mathbf{y}^{(pro,RUL)}(t)$ to the estimated true RUL as described in (11), the RMSE in tracking RUL of the proposed SSA-LSTM algorithm was 2.285, whereas that of the LSTM [24] was 23.24.

4.3 Testing on the Target Datasets

In order to test the concept of the proposed similarity based RUL prediction system, we then tested whether the remaining datasets respond to the choice of window lengths in similar ways to the prototype. We selected 20 datasets from the remaining 199 datasets and measured the RUL estimation performance of each dataset at various choices of the window length, L .

In Figure 8, the normalized RMSE plots of the selected 20 datasets are illustrated. It is interesting to observe that the window lengths associating with low RMSE of all 20 datasets are approximately in the range of 10 to 30. The regions coincide with the suggested window lengths as stated in Refs. [11–15]. As the optimal window length obtained from the prototype data, L_{opt} was 24, it is legitimate for using L_{opt} as a representative for the window length for all the target datasets selected from the remaining datasets.

Proceeding to the evaluation of the SSA-LSTM algorithm, we tested the performance for 200 datasets as compared to the existing LSTM algorithm. We used the optimal window length $L_{opt} = 24$ for the SSA-LSTM algorithm. The additional parameter to be set for the SSA-LSTM was B which was set to nine. The performance metrics were the MAEs and the RMSEs of the

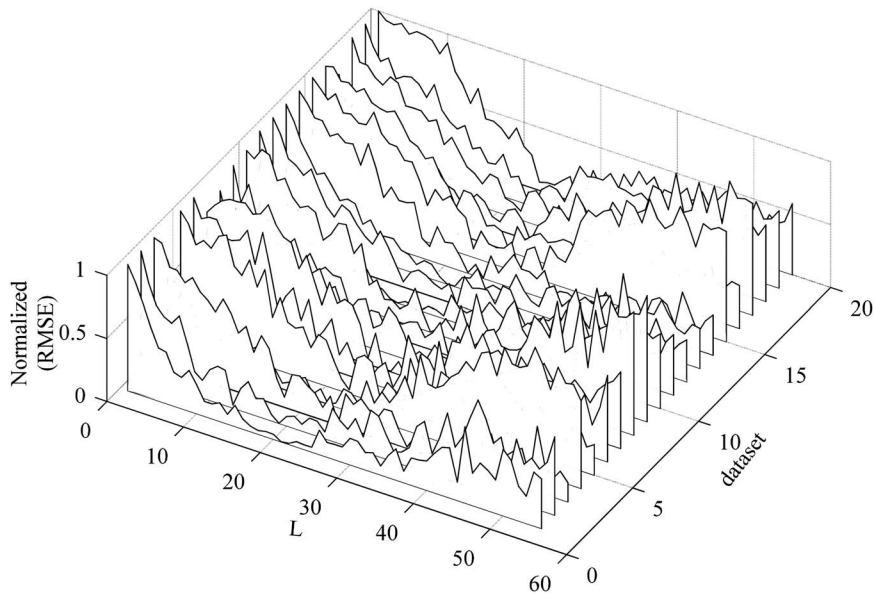


Figure 8 The ensemble of the normalized RMSEs in RUL estimations of 20 target datasets performed by the proposed SSA-LSTM algorithm.

RUL estimation of the SSA-LSTM and LSTM algorithms, and are shown in Figures 9(a) and 9(b), respectively. The black circles denote the results from using the SSA-LSTM, whereas those from the LSTM are represented by the white ones. As shown in Figures 9(a) and 9(b), both the MAE and RMSE levels of the SSA-LSTM are lower in average across the 200 datasets as compared to those of the LSTM. The numerical results of the averaged MAEs and RMSEs for the 200 datasets of both the algorithms are shown in Table 1. The numerical results indicate that those of the SSA-LSTM are lower than those of the LSTM nearly by half.

We then investigated in detail regarding how the SSA-LSTM and LSTM perform comparatively in estimation of the RUL in each record. We selected three target datasets from 199 datasets and labeled them tar_1 to tar_3 . Similar to Figure 8, the RMSEs of three target datasets were tested against the window lengths as shown in Figures 10(a)–(c). It is shown that the window length associated with the minimum RMSEs of all tar_1 to tar_3 datasets are

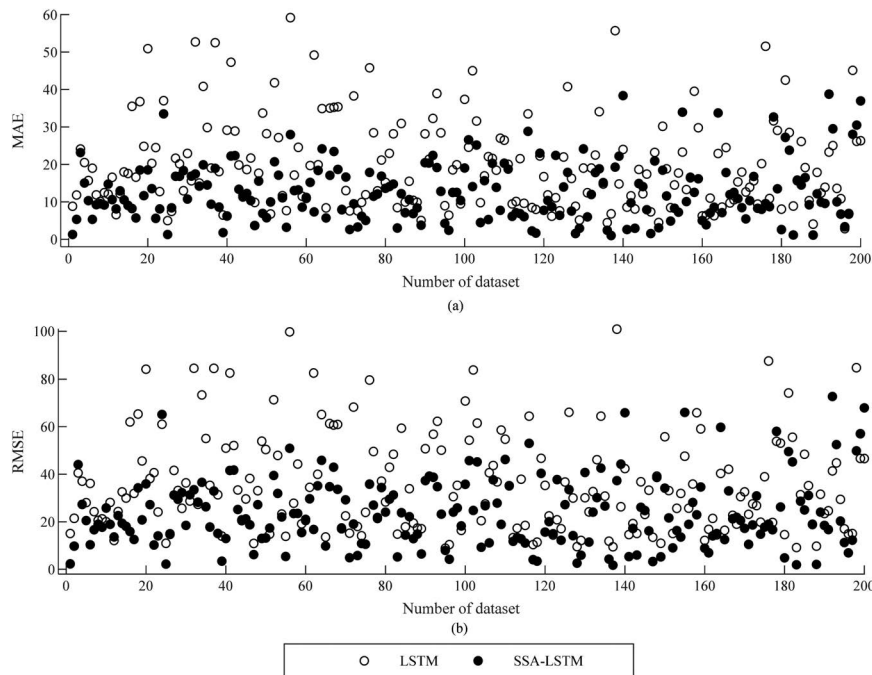


Figure 9 Performance plots of RUL estimation of 200 datasets by SSA-LSTM, (black circles) and LSTM (white circles) as measured by (a) the mean absolute errors (MAEs) and (b) the root mean squared errors (RMSEs).

Table 1 Averaged MAE and averaged RMSE for 200 datasets

Algorithms	MAE	RMSE
LSTM	23.828	35.284
SSA-LSTM	12.750	19.783

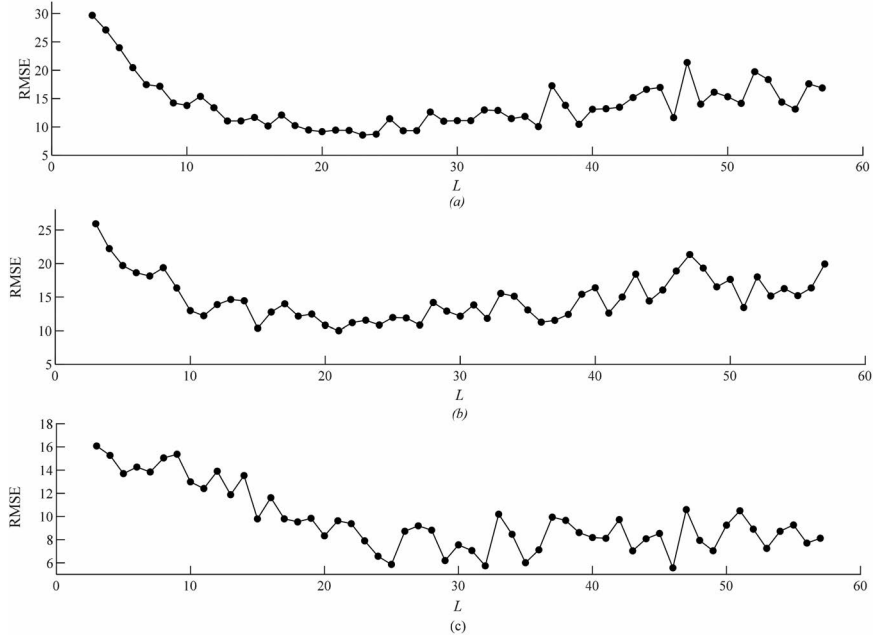


Figure 10 The RMSE at various window lengths of (a) tar_1 , (b) tar_2 and (c) tar_3

located in the vicinity of L_{opt} obtained from the prototype datasets, i.e., $L_{opt} = 24$.

Also, the parameter B was set to nine. In Figures 11(a)–11(i), the three sub-components of the true RUL, $\mathbf{x}_p^{(RUL)}(t)$, i.e., $p=1, 2, 3$ are estimated by the SSA-LSTM. Figures 11(a)–11(c) illustrate the estimation results for the first target datasets tar_1 of $\mathbf{x}_p^{(RUL)}(t)$ and $\mathbf{y}_p^{(RUL)}(t)$ where $p=1$ to 3, respectively. In a similar fashion, Figures 11(d)–11(f) are the results $p=1$ to 3 for the second target tar_2 and Figures 11(g)–11(i) are for the third target tar_3 .

For all the three sub-component RUL data, the SSA-LSTM performs the best in the estimation of $\mathbf{y}_1^{(RUL)}(t)$ for the first sub-component RUL $\mathbf{x}_1^{(RUL)}(t)$, and the same are shown in Figures 11(a), (d) and (g). This is

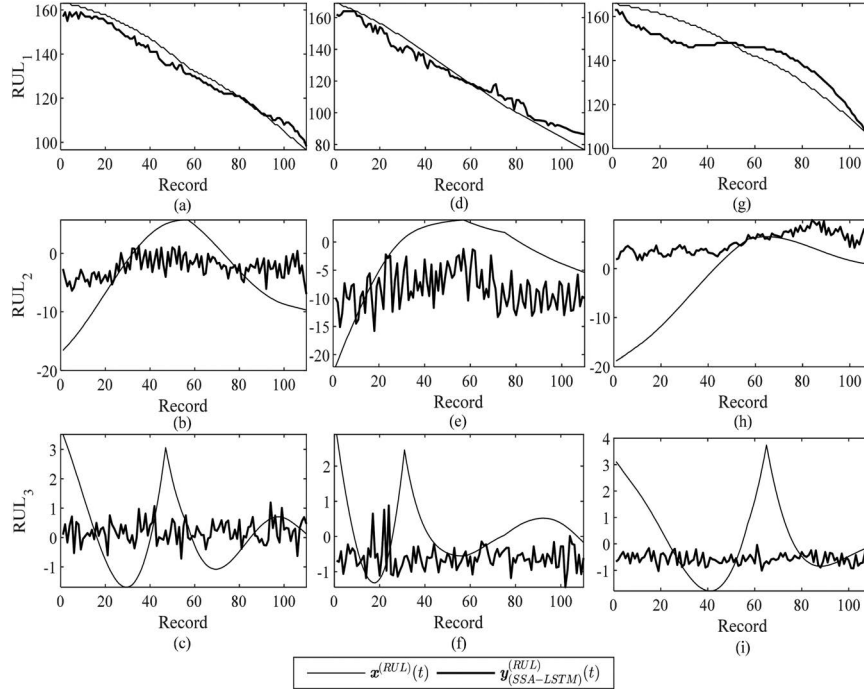


Figure 11 The estimated sub-components RULs of three target datasets by SSA-LSTM with $L_{opt} = 24$ and $B = 9$ for (a,b,c) tar_1 , (d,e,f) tar_2 and (g,h,i) tar_3 .

true for all the three targets, tar_{1-3} , even though the performance of tar_3 is not as good as those of the other two. For the sub-component $x_2^{(RUL)}(t)$ and $x_3^{(RUL)}(t)$, the estimation $y_2^{(RUL)}(t)$ and $y_3^{(RUL)}(t)$ witness the degradations in performance. However, after combining all the sub-components to produce the estimated RUL, $y_{(SSA-LSTM)}^{(RUL)}(t)$, the SSA-LSTM shows a more promising result in the RUL estimation than the LSTM. The results are shown in Figures 12(a)–12(c).

By inspection, it is apparent that for all the three target datasets, the SSA-LSTM algorithm is capable of estimating the RULs, whereas the LSTM has a difficulty in performing the task. For numerical results the MAEs and RMSEs of the RUL prediction of the SSA-LSTM and LSTM are compared in Tables 2 and 3, respectively.

It is apparent from the results of the MAE and RMSE for all the three target datasets that the SSA-LSTM has achieved a significant improvement over the LSTM.

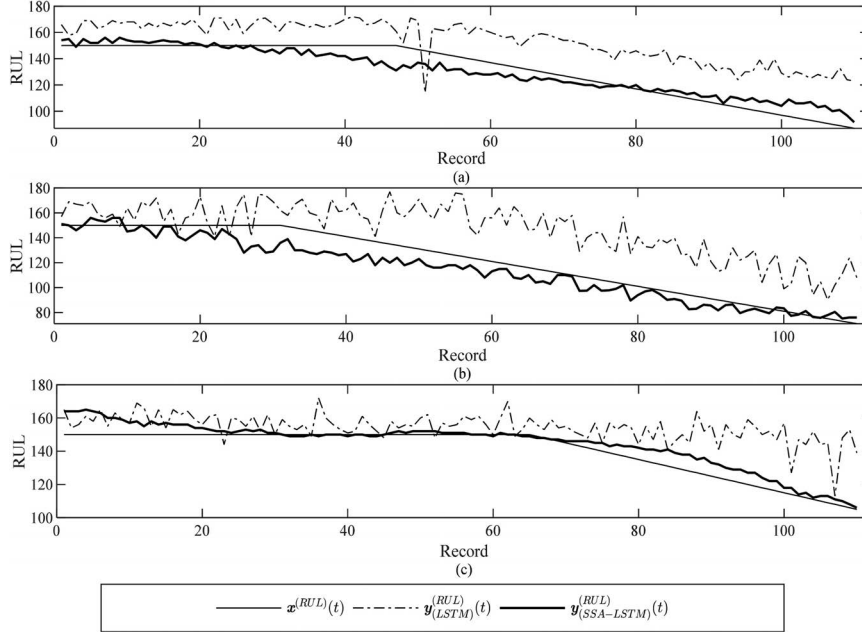


Figure 12 The estimated RUL of three target datasets by SSA-LSTM and LSTM for (a) tar_1 , (b) tar_2 and (c) tar_3 .

Table 2 MAE of RUL estimation by LSTM and SSA-LSTM for the three target datasets

Algorithm	$y_{(Algorithm)}^{(tar_1, RUL)}(t)$	$y_{(Algorithm)}^{(tar_2, RUL)}(t)$	$y_{(Algorithm)}^{(tar_3, RUL)}(t)$
LSTM	24.4638	26.8894	12.7456
SSA-LSTM	6.9426	9.7874	4.7874

Table 3 RMSE of RUL estimation by LSTM and SSA-LSTM for the three target data sets

Algorithm	$y_{(Algorithm)}^{(tar_1, RUL)}(t)$	$y_{(Algorithm)}^{(tar_2, RUL)}(t)$	$y_{(Algorithm)}^{(tar_3, RUL)}(t)$
LSTM	26.8622	28.5992	14.2143
SSA-LSTM	8.1145	10.8742	5.8742

5 Conclusion

In this paper, we have proposed the SSA-LSTM for the estimation of the RUL of turbofan engines’ datasets of Ref. [18]. The proposed system offers the advantage of determining the optimal parameters for a large number of datasets using only those determined from one representative dataset called the prototype. The algorithm is a two-step approach. First, the optimal window length of the trajectory matrix of the data is empirically determined

from a prototype dataset. This involves the analysis and synthesis operations of the SSA algorithm and the estimation part of the LSTM networks. Second, the target datasets are selected for the RUL estimation with the SSA-LSTM algorithm using the optimal window length obtained from the first part.

The experimental results show how the performance in RUL estimation of the proposed system has improved significantly over the existing LSTM. The validity of this concept is verified by testing on 200 datasets of features and true RULs of the turbofan engines.

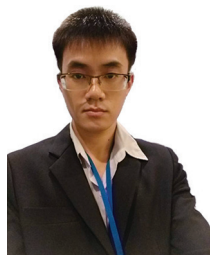
References

- [1] L. Ren, Y. Sun, H. Wang, and L. Zhang. Prediction of bearing remaining useful life with deep convolution neural network. *IEEE Access*, 6:13041–13049, 2018.
- [2] Z. Zhao, B. Liang, X. Wang, and W. Lu. Remaining useful life prediction of aircraft engine based on degradation pattern learning. *Reliability Engineering & System Safety*, 164:74–83, 2017.
- [3] Y. Liu, X. Hu, and W. Zhang. Remaining useful life prediction based on health index similarity. *Reliability Engineering & System Safety*, 185:502–510, 2019.
- [4] X.S. Si, W. Wang, C.H. Hu, D.H. Zhou, and M.G. Pecht. Remaining useful life estimation based on a nonlinear diffusion degradation process. *IEEE Transactions on Reliability*, 61(1):50–67, 2012.
- [5] E. Sutrisno, H. Oh, A. S. S. Vasan, and M. Pecht. Estimation of remaining useful life of ball bearings using data driven methodologies. In *3rd IEEE Conference on Prognostics and Health Management, Denver, Colorado, 18-21 June, 2012*, pages 1–7, June 2012.
- [6] F. Liu, B. He, Y. Liu, S. Lu, Y. Zhao, and J. Zhao. Phase space similarity as a signature for rolling bearing fault diagnosis and remaining useful life estimation. *Shock and Vibration*, 2016(5341970):1–12, 2016.
- [7] Q. Zhang, P. W.T. Tse, X. Wan, and G. Xu. Remaining useful life estimation for mechanical systems based on similarity of phase space trajectory. *Expert Systems with Applications*, 42(5):2353–2360, 2015.
- [8] L. Zeming, G. Jianmin, J. Hongquan, G. Xu, G. Zhiyong, and W. Rongxi. A similaritybased method for remaining useful life prediction based on operational reliability. *Applied Intelligence*, 48(9):2983–2995, 2018.

- [9] P. Bonizzion and J.M.H. Karel. Singular spectrum decomposition: A new method for time series decomposition. *Advances in Adaptive Data Analysis*, 6(4):1450011, 2014.
- [10] N. Golyandina and A. Zhigljavsky. *Singular Spectrum Analysis for time series*. Springer Science & Business Media, 2013.
- [11] N. Golyandina and A. Korobeynikov. Basic singular spectrum analysis and forecasting with r. *Computational Statistics & Data Analysis*, 71:934–954, 2014.
- [12] J.B. Elsner and A.A. Tsonis. *Singular spectrum analysis: a new tool in time series analysis*. Springer Science & Business Media, 2013.
- [13] M. Atikur Rahman Khan and D.S. Poskitt. A note on window length selection in singular spectrum analysis. *Australian & New Zealand Journal of Statistics*, 55(2):87–108, 2013.
- [14] M.A.R. Khan and D. Poskitt. Window length selection and signal-noise separation and reconstruction in singular spectrum analysis. *Monash Econometrics and Business Statistics Working Papers*, 23(11):2011-23, 2011.
- [15] M.A.R. Khan and D. Poskitt. Moment tests for window length selection in singular spectrum analysis of short-and long-memory processes. *Journal of Time Series Analysis*, 34(2):141–155, 2013.
- [16] W. Mao, J. He, J. Tang, and Y. Li. Predicting remaining useful life of rolling bearings based on deep feature representation and long short-term memory neural network. *Advances in Mechanical Engineering*, 10(12):1687814018817184, 2018.
- [17] J. Liu, Q. Li, W. Chen, Y. Yan, Y. Qiu, and T. Cao. Remaining useful life prediction of pemfc based on long short-term memory recurrent neural networks. *International Journal of Hydrogen Energy*, 44(11): 5470–5480, 2019.
- [18] NASA. Prognostics Center of Excellence Data Repository: Turbofan engine degradation simulation data set. <https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/>, last accessed on May 2020.
- [19] F.J. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *IEEE*, 66:51–83, 1978.
- [20] A. Miranian, M. Abdollahzade, and H. Hassani. Day-ahead electricity price analysis and forecasting by singular spectrum analysis. *IET Generation, Transmission & Distribution*, 7(4):337–346, 2013.

- [21] C. Yu, Y. Li, and M. Zhang. Comparative study on three new hybrid models using elman neural network and empirical mode decomposition based technologies improved by singular spectrum analysis for hour-ahead wind speed forecasting. *Energy Conversion and Management*, 147:75–85, 2017.
- [22] C. Lima, V.L. de Oliveira Castellani, J.F.M. Pessanha, and A. Soares. A model to forecast wind speed through singular spectrum analysis and artificial neural networks. In *Proceedings on the International Conference on Artificial Intelligence (ICAI), The Steering Committee of The 2017 World Congress in Computer Science, Computer Engineering and Applied Computing, Las Vegas, Nevada, USA, 17-20 July, 2017*, pages 235–240, July 2017.
- [23] P. Intachai and P. Yuvapoositanon. A singular spectrum analysis pre-coded deep learning architecture for forecasting currency time series. In *6th International Electrical Engineering Congress (iEECON), Krabi, Thailand, 7-9 March, 2018*, pages 1–4. IEEE, Mar 2018.
- [24] The Mathworks Inc.. Sequence-to-sequence regression using deep learning. <https://www.mathworks.com/help/deeplearning/ug/sequence-to-sequence-regression-using-deep-learning.html>, last accessed on May 2020.

Biographies



Prakit Intachai received his bachelor's and master's degrees in telecommunication engineering in 2010 and 2012, respectively. Currently, he is pursuing a doctor of engineering degree in electronic engineering with signal processing as the subject of study at the Mahanakorn University of Technology, Bangkok, Thailand. His main research interests include technical analysis and synthesis of time series data and estimation of remaining useful life in industrial systems.



Peerapol Yuvapoositanon received his B.Eng degree in electronics from the King Mongkut's Institute of Technology, Ladkrabang, Thailand in 1991 and the M.Sc., degree and DIC in physical science and engineering in medicine in 1995 and Ph.D. degree and DIC in electrical engineering (digital signal processing for wireless communications) in 2002, both from the Imperial College, London, United Kingdom. Since 1992, he has been with the Mahanakorn University of Technology, Bangkok, Thailand where he is now serving as an associate professor in electronic engineering. Throughout the years, he has served as a reviewer and a technical program committee member for multiple international conferences and journals. He has also been involved in the automation industry as a consultant for image and signal processing tasks. His current research interests include signal processing theory for time series analysis in predictive maintenance and machine learning algorithms for industrial applications.

