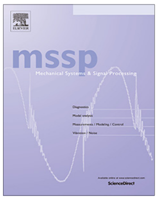
Mechanical Systems and Signal Processing 136 (2020) 106486

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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



Prediction of remaining useful life by data augmentation technique based on dynamic time warping ☆

Seokgoo Kim [a], Nam Ho Kim [b], Joo-Ho Choi [c,*]

[a] *Department of Aerospace & Mechanical Engineering, Korea Aerospace University, Goyang-si, 10540, Republic of Korea*
[b] *Mechanical & Aerospace Engineering, University of Florida, Gainesville, FL, 32611, USA*
[c] *School of Aerospace & Mechanical Engineering, Korea Aerospace University, Goyang-si, 10540, Republic of Korea*

ARTICLE INFO

Article history:
Received 28 January 2019
Received in revised form 18 October 2019
Accepted 27 October 2019

Keywords:
Dynamic time warping
Neural network
Prognostics
Run-to-fail
Data augmentation
Uncertainty

ABSTRACT

In the data-driven approaches for engineering prognostics, the lack of run-to-fail (RTF) data has been one of the bottlenecks that hinders practical applications in the field. In order to tackle this issue, the dynamic time warping (DTW) method is presented to augment the RTF data obtained from different operating conditions or systems to the current system, which plays the role of virtual RTF data under the current condition. Once the virtual RTF data are available, they are used to train a neural network model to predict the remaining useful life of the current system. When multiple RTF data are available with different behavior under different failure modes, an RMSE-based performance criterion is proposed that can adaptively choose the closest match to the current data and use it as the virtual RTF data during the prognosis process. Numerical examples are given to show that the proposed DTW-based data augmentation can predict the RUL with less uncertainty than the conventional neural network model without data mapping.

1. Introduction

Diagnosing the current health state of system and predicting the unexpected failure bring us not only a cost-effective operation but also a zero-downtime unscheduled maintenance. Nowadays, researches toward this objective, which is recognized as the area of prognostics and health management (PHM), are growing rapidly in many industrial fields. Two main streams of PHM are the diagnostics and prognostics. Diagnostics focuses on estimation of degradation level of component or system and identification of root causes of failure. Prognostics predicts the occurrence of fault or critical failure and estimate the remaining useful life (RUL) before the failure occurs.

There are several review papers which have introduced the state-of-the-art of PHM research [1–6]. Jardine et al. [1] summarized the researches and developments in the machinery diagnostics and prognostics. Pecht et al. [2] have addressed the research state in the PHM of information and electronics-rich systems. Lee et al. [3] provided a comprehensive review of the current PHM field and introduced commonly used algorithms and their characteristics. An et al. [4] suggested practical options for prognostics to choose appropriate algorithms for different applications. Several review papers have commonly pointed out that the prognostics is of more importance than the diagnostics in order to achieve effective asset management.

☆ Paper revised 2nd time for the publication in the Mechanical Systems and Signal Processing.
* Corresponding authors.
E-mail addresses: sgkim@kau.kr (S. Kim), nkim@ufl.edu (N.H. Kim), jhchoi@kau.ac.kr (J.-H. Choi).

https://doi.org/10.1016/j.ymssp.2019.106486

Prognostics is generally categorized into three approaches, namely model-based, data-driven and hybrid approach. Each approach has its own advantages and disadvantages [6]. Model-based approaches integrate a physical degradation model with measured data to predict the future trajectory of degradation and RUL. The most popular algorithms are Particle filter (PF) [7] and Kalman filter (KF) [8]. The model-based approaches provide accurate estimation of the RUL if the model is applied properly. In the case of complex system, however, it is difficult to characterize the physics of failure and construct an accurate degradation model. Data-driven approaches build a mathematical model or numerical algorithm to describe the degradation behavior using measured data, which is often called training data set. Many researches have employed data-driven approaches to predict the RUL of machineries. Numerous machine learning algorithms are employed in this field such as Multi-layer perceptron (MLP) [9], Recurrent neural network (RNN) [10]. Convolutional neural network (CNN) [11] and Long short-term memory (LSTM) [12]. The advantage of data-driven approaches is that they do not require the domain knowledge and understanding complex physical behavior of damage. However, data-driven approaches need a larger amount of run-to-failure data than the model-based approaches to train the mathematical model with many unknown parameters. Hybrid approaches attempt to leverage the advantages by combining the two approaches. For this purpose, selection of various prognostics algorithms is suggested depending on the data availability [13].

Among the three prognostics methods, according to the recent review paper [5], the data-driven prognostics is widely used in many applications. However, the lack of available field data describing the fault progression has been a major huddle, which is referred to as 'data deficiency'. There have been a few studies to overcome this huddle. For example, Widodo et al. [14] utilized censored and complete data obtained during the condition monitoring (CM) to assess the machine degradation. Hu et al. [15] proposed a co-training approach to predict the RUL by combining the censored data till the planned maintenance and a small number of run-to-fail (RTF) data. Even though the co-training approach shows an accurate RUL prediction, its long-term prediction is unclear because it uses the censored data at the last stage (90–100% of life cycle) of unit's lifetime. Sobie et al. [16] trained a machine learning classification model with training data which is generated from a simulation model for bearing diagnostics. Although the paper proposed interesting approaches, it requires a good system-level simulation model, which is rare in complex systems. An et al. [17] proposed prognostics approaches utilizing the degradation data in the accelerated life test (ALT) by converting them to those in the field loading conditions. However, this approach requires various ALT data under several different loading conditions in order to use them by mapping in the field loading condition. All the previous studies commonly mentioned that RTF data are rarely available in the in-service system, whereas proper RTF data should be used for the training in the data-driven prognostics.

In this paper, a dynamic time warping (DTW) method and artificial neural network (ANN) are utilized to predict RUL when a small number of RTF data are available. The future degradation data of the current system are simulated by mapping the current degradation data to the RTF data using DTW. The major assumption of the proposed approach is that the failure mode of the current data is the same with that of RTF data. If the RTF data under various failure modes exist, RUL prediction of current data can be accomplished by applying DTW to those RTF data. This is equivalent to exploring all possible failure modes from the past RTF data. DTW has been widely used to measure a similarity between two different sequence data. Barré et al. [18] estimated signal patterns from acquired data to observe the change in degradation behavior of battery, which was quantified using DTW. Tao et al. [19] developed the dynamic spatial time warping (DSTW) based on traditional DTW for the purpose of estimating the battery capacity degradation. Atamuradov et al. [20] focused on the phenomenon that similarity between the observed time series and control signal gives an information about the health state of the railway turnout system and used DTW to calculate the similarity. In addition to this, there were several studies to improve the performance of finding the global optimal mapping path. Yi et al. [21] proposed FastMap and a lower bounding function to improve the speed of DTW. Kim et al. [22] introduced a novel approach that supports high search performance without false dismissal under DTW by utilizing new distance function. Park et al. [23] employed a suffix tree for fast retrieval of similar subsequences without false dismissals in DTW. For the purpose of diagnostics and prognostics, most literature only used DTW to identify a similarity between two datasets by finding optimal mapping path between them. Unlike existing literature, this paper proposes the first attempt to generate virtual data by predicting the mapping path of DTW. In addition to this aspect, the original purpose of DTW is modified by adding the predictive interval to the mapping path. Therefore, the unique contribution of this paper is to use DTW not just for measuring the similarity but also overcoming the data deficiency by creating a number of virtual RTF data. Once, the virtual RTF data set is generated, the RUL prediction is performed by using an ANN model.

The paper is organized as follows. Section 2 introduces theoretical background of the proposed algorithms. In this section, basis of DTW is explained with numerical example. In Section 3, the authors introduce a methodology of data augmentation and mapping performance measure which is represented with root mean squared error (RMSE). Applications of the proposed algorithm are presented in Section 4 using simulation and real data of crack growth. The paper is concluded in Section 5.

2. Theoretical background

2.1. Dynamic time warping

DTW compares two sets of asynchronous signals and measures the similarity by finding the optimally matching path between them. DTW has been commonly used in the field of speech recognition, data mining and imaging [18,19,24]. Suppose that two time-series data, $X = \{x_1, x_2, \cdots, x_N\}$ and $Y = \{y_1, y_2, \cdots, y_M\}$ of length N and M, respectively, are given. In prognostics, these two data sets represent degradation data under different operating conditions.

The first step toward DTW is to construct the N-by-M cost matrix whose components are the difference $c(x_i, y_j)$ between two data x_i and y_j; that is, $c(x_i, y_j) = |x_i - y_j|$. Typically, $c(x_i, y_j)$ is small (low cost) when x_i and y_j are similar to each other, otherwise it is large (high cost). Then the goal is to find an alignment between X and Y having a minimal overall cost. A warping path is a sequence $p = (p_1, \cdots, p_L)$ with $p_l = (n_l, m_l) \in [1:N] \times [1:M], \forall l \in [1:L]$which defines an alignment between two sequences by assigning the element x_{n_l} of X to the element y_{m_l} of Y. Typically, the warping path satisfies the following conditions.

$$\begin{array}{l}\text{Boundary condition}: p_1 = (1,1) \text{ and } p_L = (N, M)\\ \text{Monotonicity condition}: n_1 \leqslant n_2 \leqslant \ldots \leqslant n_L \; and \; m_1 \leqslant m_2 \ldots \leqslant m_L\\ \text{Step size condition}: p_{l+1} - p_l \in \{(1,0),(0,1),(1,1)\} \text{ for } l \in [1 : L-1]\end{array} \tag{1}$$

The last condition means that the warping path can move either in the horizontal, vertical, or diagonal direction. Among the three possible paths, DTW finds the one that yields the lowest total cost. The total cost $c_p(X, Y)$ of a warping path p between X and Y is calculated by accumulating all the individual cost $c(x_{nl}, y_{ml})$ along the path as

$$c_p(X, Y) = \sum_{l=1}^{L} c(x_{nl}, y_{ml}) \tag{2}$$

where L is the total number of components in the path. Furthermore, an optimal warping path p^* between X and Y is determined by minimizing the total cost among all possible warping paths. The DTW distance, $d_{DTW}(X, Y)$, between X and Y is then defined as the total cost of p^*:

$$d_{DTW}(X, Y) = c_{p*}(X, Y) = \min\{c_p(X, Y) | p \text{ is an } (N, M) - \text{warping path}\} \tag{3}$$

The optimal path p^* can be found by defining an accumulated cost matrix, $D(N, M)$ with N rows and M columns. The accumulated cost matrix D satisfies the following conditions

$$\begin{array}{l}D(n,1) = \sum_{k=1}^{n} c(x_k, y_1) \text{ for } n \in [1:N]\\ D(1,m) = \sum_{k=1}^{m} c(x_1, y_k) \text{ for } m \in [1:M]\\ D(n,m) = c(x_n, y_m) + \min\{D(n-1, m-1), D(n-1, m), D(n, m-1)\}\\ \text{for } 2 \leqslant n \leqslant N \text{ and } 2 \leqslant \; < m \leqslant M\end{array} \tag{4}$$

The first row and the first column are calculated by accumulating cost along the row and column, respectively. The component $D(n, m)$ is calculated by adding $c(x_n, y_m)$ to the smallest accumulated cost in the previous adjacent elements; i.e., among $D(n-1, m-1), D(n-1, m), \text{and} D(n, m-1)$.

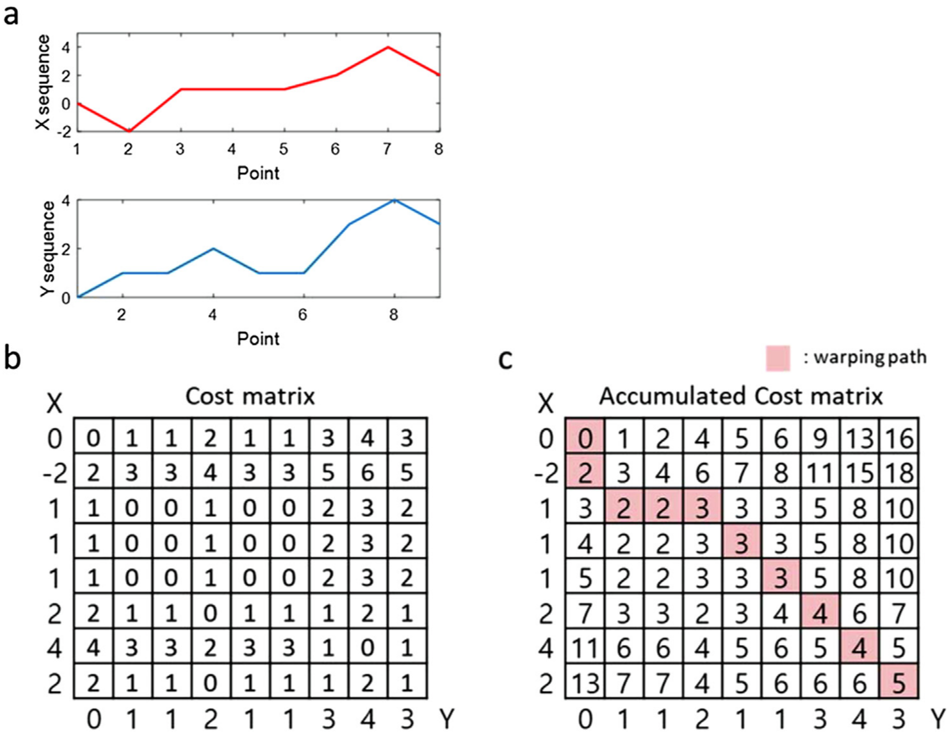
Once the accumulated cost matrix is available, the optimal path p^* is determined in the reverse order of the indices starting from $p_L = (N, M)$. Suppose $p_l = (n, m)$ has been determined. Then, the next point p_{l-1}in the path can be determined as

$$p_{l-1} := \begin{cases} (1, m-1) & \text{if} \quad n = 1 \\ (n-1, 1) & \text{if} \quad m = 1 \\ \operatorname{argmin}\{D(n-1, m-1), D(n-1, m), D(n, m-1)\}, & \text{otherwise}, \end{cases} \tag{5}$$

If "arg min" is not unique, the smallest pair is taken lexicographically. The process ends when $(n, m) = (1, 1)$. As a result, the mapping path that is found by DTW yields the best global match or alignment between two sequences as are mentioned in several literatures [25–27]. A simple example of DTW between X and Y sequence, is shown in Fig. 1(a), where the sequence length N and M are 8 and 9, respectively. As shown in Fig. 1(b), the cost matrix between two sequences can be calculated using the difference between data. Once the accumulated cost matrix is constructed, the warping path can be generated as depicted in Fig. 1(c). Table 1 specify the optimal warping path of Fig. 1(c). Using the optimal mapping path calculated by DTW, X sequence and Y sequence can be mapped each other.

2.2. Artificial neural network

In recent years, artificial neural network (ANN) has received much attention as a tool for solving complex problems. ANN is one of the most commonly used data-driven techniques in prognostics [28]. Fig. 2 shows a simple example of ANN, which consists of 3 layers: input, hidden and output layers. The inputs x_i, $i = 1, \ldots, I$ to the neuron are multiplied by weights W_{ij} and summed up together with the constant bias term of l^{th} layer $\theta_j^{(l)}$ at the hidden node j. The resulting n_j is the input to the activation function g [29]. The same process is conducted from a hidden node j to output node k to yield output data o_k



Cost matrix

X									
0	0	1	1	2	1	1	3	4	3
-2	2	3	3	4	3	3	5	6	5
1	1	0	0	1	0	0	2	3	2
1	1	0	0	1	0	0	2	3	2
1	1	0	0	1	0	0	2	3	2
2	2	1	1	0	1	1	1	2	1
4	4	3	3	2	3	3	1	0	1
2	2	1	1	0	1	1	1	2	1
	0	1	1	2	1	1	3	4	3 Y

Accumulated Cost matrix

X									
0	0	1	2	4	5	6	9	13	16
-2	2	3	4	6	7	8	11	15	18
1	3	2	2	3	3	3	5	8	10
1	4	2	2	3	3	3	5	8	10
1	5	2	2	3	3	3	5	8	10
2	7	3	3	2	3	4	4	6	7
4	11	6	6	4	5	6	5	4	5
2	13	7	7	4	5	6	6	6	5
	0	1	1	2	1	1	3	4	3 Y

Fig. 1. Application of DTW to X and Y sequence data: (a) X and Y sequence data, (b) cost matrix between X and Y, and (c) Accumulated cost matrix and warping path.

Table 1
Optimal mapping path example.

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
n	1	2	3	3	3	4	5	6	7	8
m	1	1	2	3	4	5	6	7	8	9

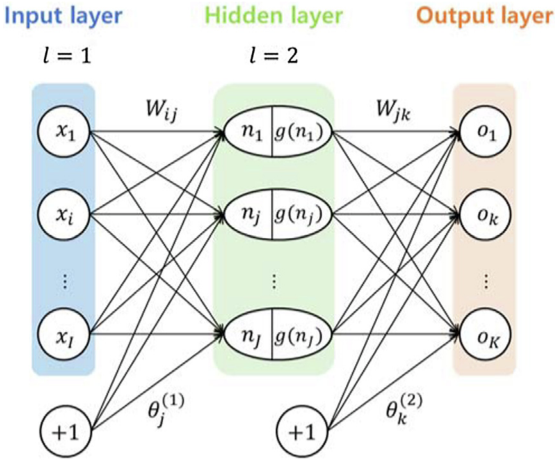


Fig. 2. ANN with one hidden layer.

of the network. It is possible that multiple hidden layers can be used for a complex network model. As more nodes and layers are used, more weights and biases need to be identified, which makes the model more flexible but the process of building the model becomes more challenging.

During the training process, the weights and the biases of the ANN model are estimated by minimizing the mean-squared-error (MSE) between the network model outputs and training data:

$$MSE = \frac{1}{N}\sum_{k=1}^{N} e_k^2 = \frac{1}{N}\sum_{k=1}^{N}(y_k - d_k)^2 \tag{6}$$

where N is the number of training data, and d_k, y_k, and e_k are the k-th training data, model output and corresponding output error, respectively [30]. Levenberg-Marquardt (LM) learning algorithm is one of the earliest and the most commonly used methods for the training algorithm [28,31].

There are different approaches of applying ANN in the prognostics. For example, An et al. [17] predicted the future degradation using a number of recent degradation data in the previous time as inputs, from which the RUL is estimated. On the other hand, Heimes [10] employed advance version of ANN called RNN and predicted the RUL of aircraft engine by training neural network model trained by multiple sensor signals and their RUL as inputs and outputs, respectively. In this paper, the latter is utilized where the RUL is directly estimated. For this purpose, this paper uses the current cycle (time) and degradation level as input nodes, while the output layer generates the RUL from the current cycle (time). The authors constructed a simple ANN model with two input nodes, two hidden nodes and one output node.

3. Methodology

3.1. Data augmentation

In this section, the key idea of the proposed approach, namely data augmentation, is presented. As mentioned in the introduction, utilization of preexisting data to be suitable for current operating condition is essential for successful data-driven prognostics. For this purpose, a simple example is considered using two degradation data sets generated from the same exponential model but with different rate as shown in Fig. 3. It is assumed that these two data sets represent degradation for different operating conditions. The reference data (star markers) and target data (circular markers) are generated every 0.2 time interval from exponential function, exp(0.5 t) and exp(0.3 t), respectively. The reference data are available until the failure occurs (i.e., RTF), whereas the target data are only available until the current time $t = 4$. The goal is to generate virtual target data beyond the current time using the data augmentation method. If the generated virtual data are accurate, they are supposed to follow the red dots, which is the exponential function, exp(0.3 t).

To perform data augmentation, DTW is applied between the reference and target data. First, the DTW finds the optimal mapping path by matching the levels of degradation as close as possible between the two datasets. When the data show a monotonic trend, which is the case for most degradation, it is better to use two datasets with the same range of degradation. As shown in Fig. 4(a), a total of 20 target degradation data (filled circular markers) are given until the current time $t = 4$, which has grown up to the level of 3.3, as shown by the dashed line. In order to match this with the reference data within the degradation range, 13 reference data (filled star markers) below the dashed line are selected, which are called mapping data here. The reference data (hollow star markers) above the dashed line will be used to generate the virtual target data after finding the optimal warping path.

The 13 mapping data below the dashed line are mapped into the 20 target data using the DTW as described in Section 2.1, with $N = 13$ and $M = 20$. Using these two sets of data, the optimal warping path, called current mapping, is generated as illustrated in Fig. 4(b), represented by the blue line with circular markers. Intuitively speaking, the mapping path between two sequence data will show a linear relationship as the two sequence data have similar trend. In the figure, the true mapping

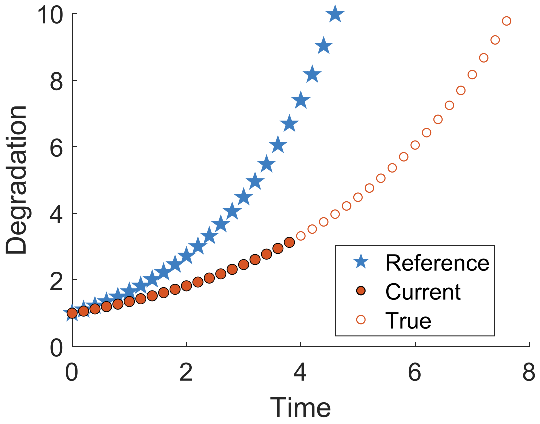


Fig. 3. Example data sets.

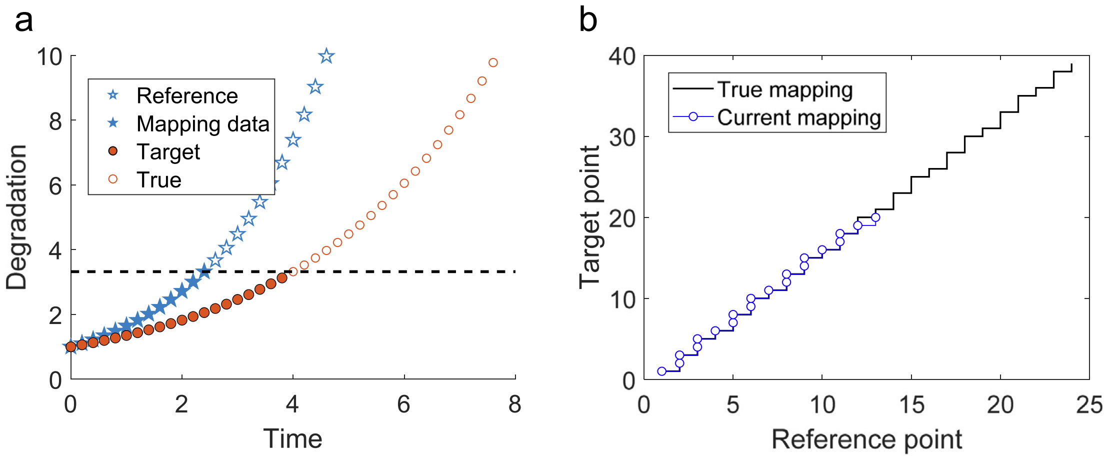


Fig. 4. Application of DTW to reference and target data: (a) Reference and target data, (b) mapping path plot between reference and target data.

path is generated from the entire reference and true data in Fig. 4(a). As the time progresses, current mapping path generated from the mapping and target data converges to the true mapping path. It is noted that the current mapping shows different trend from the true mapping near the last points because the DTW algorithm searches optimal path between the reference and target data given at different times.

The proposed data augmentation method is based on the assumption that the future mapping path will follow the same trend as the one obtained by DTW up to the current time. The linear relationship indicates that the data points from two curves can show one-to-one correspondence. Based on this assumption, the trend in the current mapping in Fig. 4(b) can be fitted by a linear function to predict or extrapolate into the future mapping path:

$$Y = \alpha + \beta X + \varepsilon \tag{7}$$

where X is the reference point, Y is the target point, and ε represents the error in the estimation of Y. α and β are the fitting parameters of linear regression, which are estimated by minimizing the mean-squared-error (MSE). For a linear regression model, the prediction interval (PI) for individual response Y_i at a given X_i can be calculated as follows:

$$PI(X_i) = \hat{Y}_i \pm t_{\alpha/2,n-2} S_{YX} \sqrt{1 + \frac{1}{n} + \frac{\left(X_i - \bar{X}\right)^2}{\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^2}} \tag{8}$$

where $\widehat{Y}_i$ is the estimated response at X_i, $\pm t_{\alpha/2,n-2}$ are the values of Student's t-distribution with $(n-2)$ degree of freedom evaluated at probabilities of $(1-\alpha/2)$ and $\alpha/2$. S_{YX} represents the standard error of estimation. Usually, the prediction interval is expressed with the upper and lower bounds, which can be considered as two different paths. Since the objective of proposed algorithm is to provide sufficient virtual data to train the data-driven model, the probability level α is varied between 0.01 and 0.99 to produce many mapping paths. As a result, 199 mapping paths which consist of median and 198 prediction intervals are generated. Note that the actual mapping path shows a zigzag pattern as shown in Fig. 4(b), while the virtual mapping paths are straight lines made by the regression in Eqs. (7) and (8).

Once virtual mapping paths are obtained, the next step is to generate the virtual target data by mapping the reference data above the dashed line (hollowed star markers) in Fig. 4(a) using the virtual mapping paths. Fig. 5 illustrates this process. In Fig. 5(a), three virtual mapping paths are generated, namely path 1, path 2 and path 3, which represent the upper 2.5% prediction bound, the median, and the lower 2.5% prediction bound, respectively. Based on these mapping paths, reference point 15, for example, is mapped into three different target points. That is, these three points have the same degradation level with the reference point 15 based on the three different mapping paths. The corresponding times on the reference point can be obtained by multiplying the predicted target point with the time interval (dt) of target data, whose results are depicted in Fig. 5(b) as the three dashed curves. Table 2 gives an example of this process for reference point 15. In the table, the degradation level is same for all paths, but the times to grow the degradation level to the specific threshold are different for different paths. It is noted that the length of virtual data depends on that of original reference data because the proposed algorithm focuses on stretching or shortening the time information of reference data.

By repeating this process, reference data is mapped into the current target data based on all 199 virtual mapping paths to generate virtual data. Fig. 6(a) illustrates a total of 199 mapping paths, which are converted into the virtual data in Fig. 6(b). These 199 sets of virtual data are used as the input data for the training of ANN model.

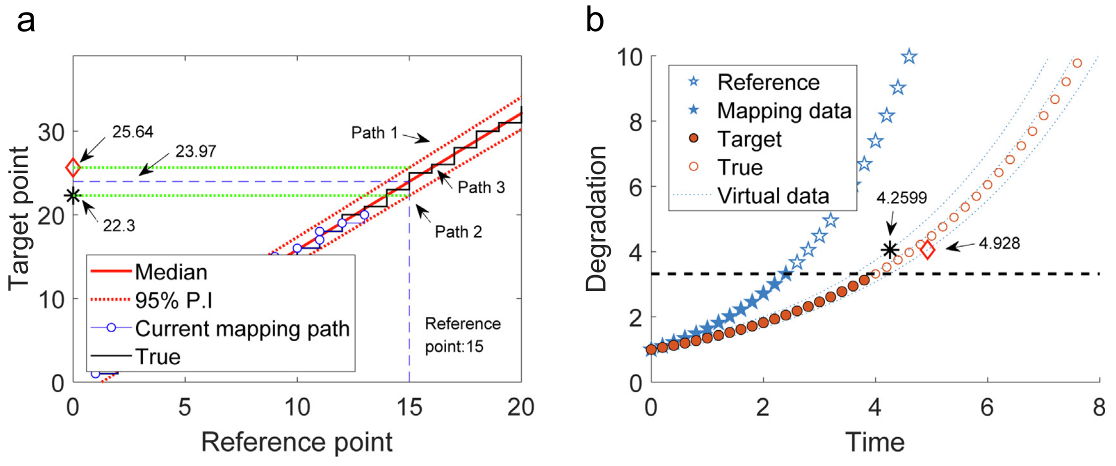


Fig. 5. Virtual target mapping generation: (a) mapping path of 95% P. I and (b) mapped 95% P. I of reference data.

Table 2
Mapping example for reference point 15.

	Reference	Path 1	Path 2	Path 3
Point	15	25.64	23.97	22.29
Time	2.8	4.928	4.594	4.259
Degradation	4.055	4.055	4.055	4.055

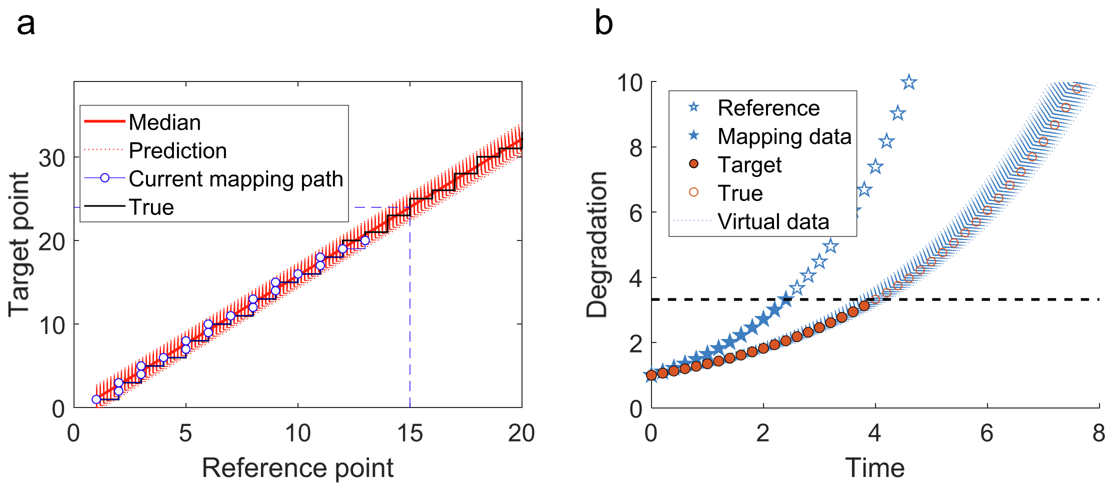


Fig. 6. Data augmentation: (a) mapping path prediction and (b) virtual data generation.

3.2. Mapping performance measure

As mentioned before, the reference data is critical for performing data augmentation. Therefore, there is no doubt that the selection of suitable reference data is an important step to generate virtual data. Inappropriate reference data may lead to a poor mapping performance. In practice, when several different reference data exist, the user should decide which data are suitable for mapping and augmenting into the current degradation problem. This paper proposes a mapping performance measure calculated from the root-mean-squared-error (RMSE). For this purpose, the current target data set is divided into training and validation data sets within itself. The training data set is used for mapping the reference data, while the validation data set is used to calculate the RMSE between the virtual data by the future mapping with the validation set. The RMSE can be calculated based on this Eqs. (9).

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N} e_k^2} = \sqrt{\frac{1}{N}\sum_{k=1}^{N} (y_k - \hat{y}_k)^2} \tag{9}$$

where N, e_k, y_k, and $\widehat{y}_k$ represent the number of samples, error, virtual data, and validation set, respectively. As shown in Fig. 7, for a given length of current data up to time 4, the reference data with the same range of degradation which are those under the dashed line is used for mapping. The training data is divided into training (circular mark) and validation (triangular mark) sets. Then, the RMSE value is calculated between mapped data and validation data.

To verify the proposed RMSE measure, an example problem is presented where there are three reference data made from different degradation functions, which may represent different failure scenarios or operating conditions. The objective is to determine which reference data is the most suitable for mapping the current data. Fig. 8 illustrates several different degradation trends by employing empirical functions that represent the degradation. As shown in Fig. 8(a), the function X = exp (0.3 t) represents the current data, while $Y1$ = exp(0.5 t), $Y2$ = 2 t + 1, and $Y3$ = 2 $t^{0.8}$ + 1 are the three reference data sets.

At a given time, the data before the time are used for the mapping, while the data after the time are used for calculating RMSE. Fig. 8(b) shows the validation results of mapping of X to the three reference data sets as a function of time. It is shown that the mapping results are not accurate for early time because of the lack of data. However, after time 3.5, the mapping between X and $Y1$ outperforms other two mappings. Indeed, the other two mappings do not show the linear relationship as was observed in Fig. 4(b), which means that they may represent different failure scenarios. Fig. 8(c) illustrates mapping paths between reference data sets and current data. As mentioned above, X and $Y1$ which were simulated from exponential function shows a linear relationship in the mapping path, whereas the other two reference data sets show a non-linear relationship [32].

4. Application

4.1. Dynamic time warping of fatigue crack growth

The crack growth example has been widely used in the field of prognostics. Paris law is well known physical degradation model, which represents the future trajectory of crack growth under repeated loading condition [33]:

$$\frac{da}{dN} = C(\Delta K)^m, \quad \Delta K = \Delta\sigma\sqrt{\pi a} \tag{10}$$

where N is the number of cycles, a is half crack size, ΔK is the range of stress intensity factor, $\Delta\sigma$ is the stress range, C and m are model parameters.

To verify the proposed dynamic time warping method, crack growth simulation data are generated at every $dN = 1000$ cycles following Eq. (10) with the 10 mm initial crack size and true parameters: $m_{true} = 3.5$, $C_{true} = 6.4 \times 10^{-11}$. Reference and current data are assumed to be operated under two different loading conditions, $\Delta\sigma = 70$MPa and 60 MPa. The reference data are generated until crack size reaches predefined threshold (0.1 m), while the current data are generated up to 13,000 and 27,000 cycles to compare the accuracy and uncertainty of the mapping. Fig. 9 shows the data augmentation results using reference data and their mapping path. Both cases show a linear relationship for the mapping path, and thus, the medians of the virtual data agree well with the true crack growth curve. As the number of cycles increases, virtual data shows a narrower interval of uncertainty. This is because the prediction interval in Eq. (8) is decreased as the number of mapping data increases.

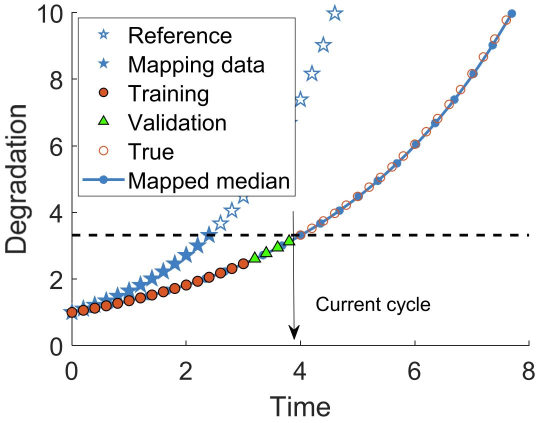


Fig. 7. Validation of mapping result.

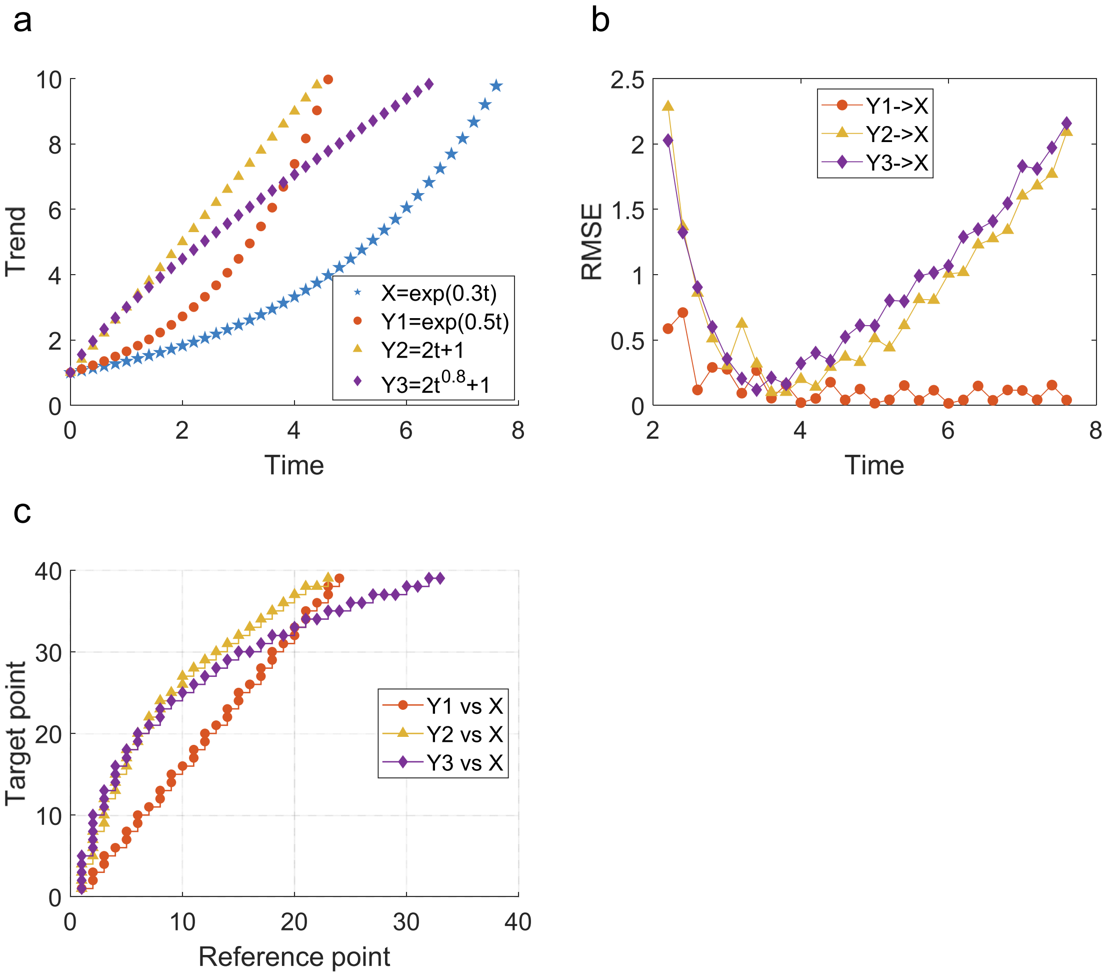


Fig. 8. Performance measure: (a) various reference data, (b) RMSE trend and (c) mapping path of reference data sets.

In the prognostics stage, three-layer ANN model with two input nodes (cycle and current crack size), two hidden nodes and one output node (RUL of virtual data) is trained to predict the RUL of current data, which can be obtained by calculating the residual cycle until crack size reaches the pre-defined threshold (0.1 m). Generally, optimal network structure of ANN (e.g. number of neurons and number of hidden layers) is searched by implementing several validation processes, but it is not the main scope of this paper. In addition, the ANN in this study employs only two input nodes which is not the complex network structure. In the training phase, virtual datasets are randomly divided into training, validation, and test data with the fractions of 0.7, 0.15, and 0.15, respectively. The model is first trained using the training data set to optimize the weights. The model is then examined using the validation data set to evaluate the degree of overfitting. Finally, the accuracy of the trained model is estimated using the test data set. The process is implemented by the ANN functions in MATLAB, in which the accuracy is calculated by R-square between the test and model output. The prediction is considered good when the R-square is close to one. In this study, R-squares of ANN in Fig. 9(b) and (d) are 0.9948 and 0.9998 respectively. ANN model training is repeated 30 times with different subsets of the training data and different initial weights. Finally, 95% predictive intervals are calculated from prediction results obtained by 30 outputs. RUL prediction results are illustrated in Fig. 10, which shows that the median converges to the true RUL curve with very narrow prediction intervals.

To identify the robustness of the proposed DTW algorithm, three different levels of noise (u = 1 mm, 3 mm and 5 mm) are considered for the crack growth example. Uniformly distributed random noise between $\pm u$mm is added to crack growth simulation for both reference and mapping data. Fig. 11 shows the mapping result under the different levels of noise and RUL estimations. The results show that a high level of noise slows the convergence of the RUL, but the noise effect is compensated by increasing the number of mapping data. Based on this observation, once the target data shows a similar trend with the reference data, the DTW mapping result shows a good performance. It is also possible that a data point might be distorted due to measurement error. In such case, the standard outlier detection methods, such as the maximum normed residual test [34], can be used to eliminate those data.

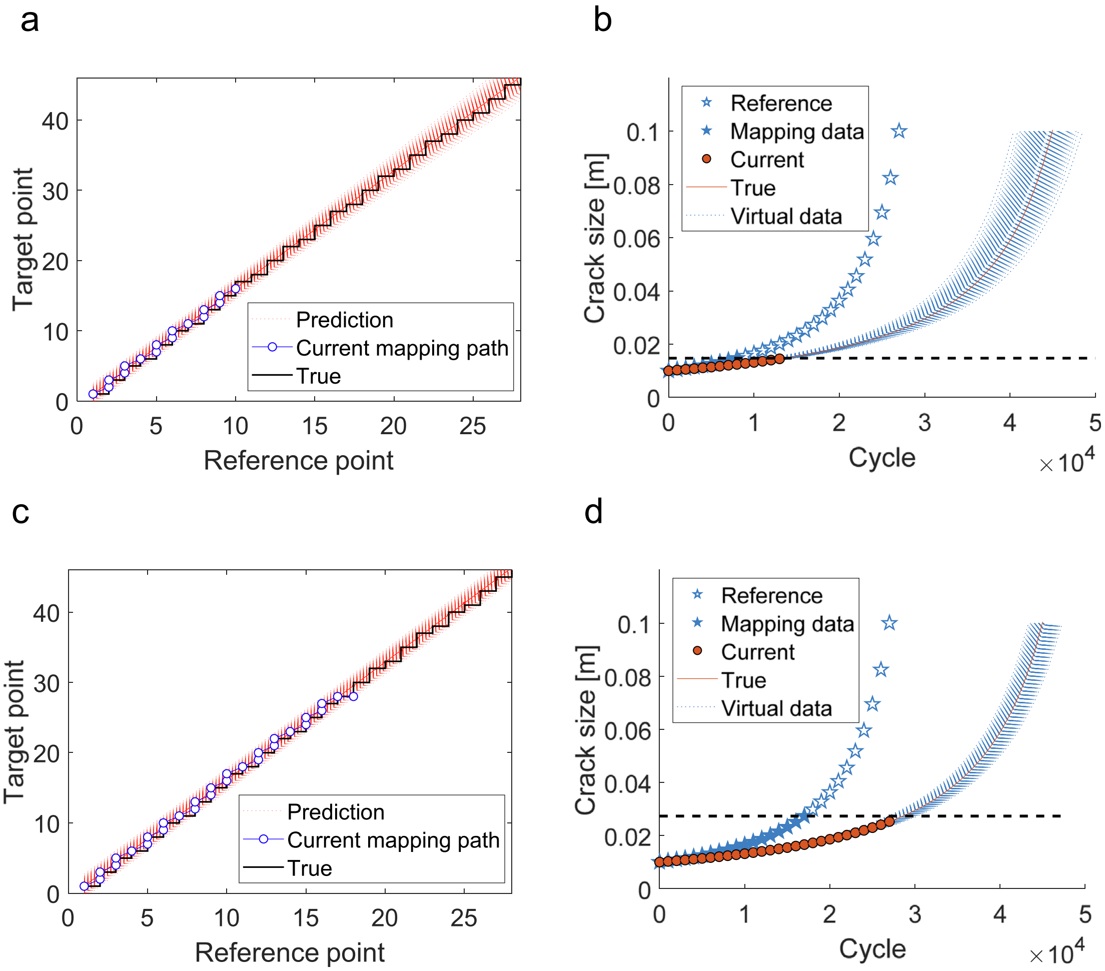


Fig. 9. Data augmentation of simulation crack growth data: (a) mapping path at 13,000 cycles, (b) mapping result at 13,000 cycles, (c) mapping path at 27,000 cycles and (d) mapping result at 27,000 cycles.

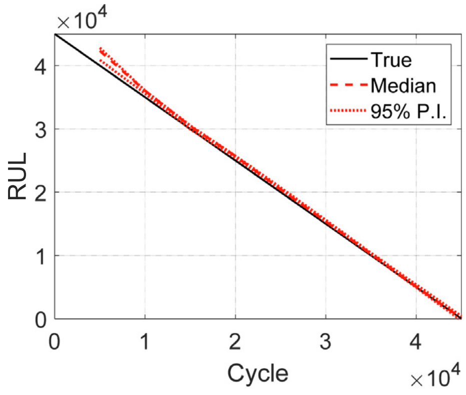


Fig. 10. Remaining useful life estimation of simulation crack growth data.

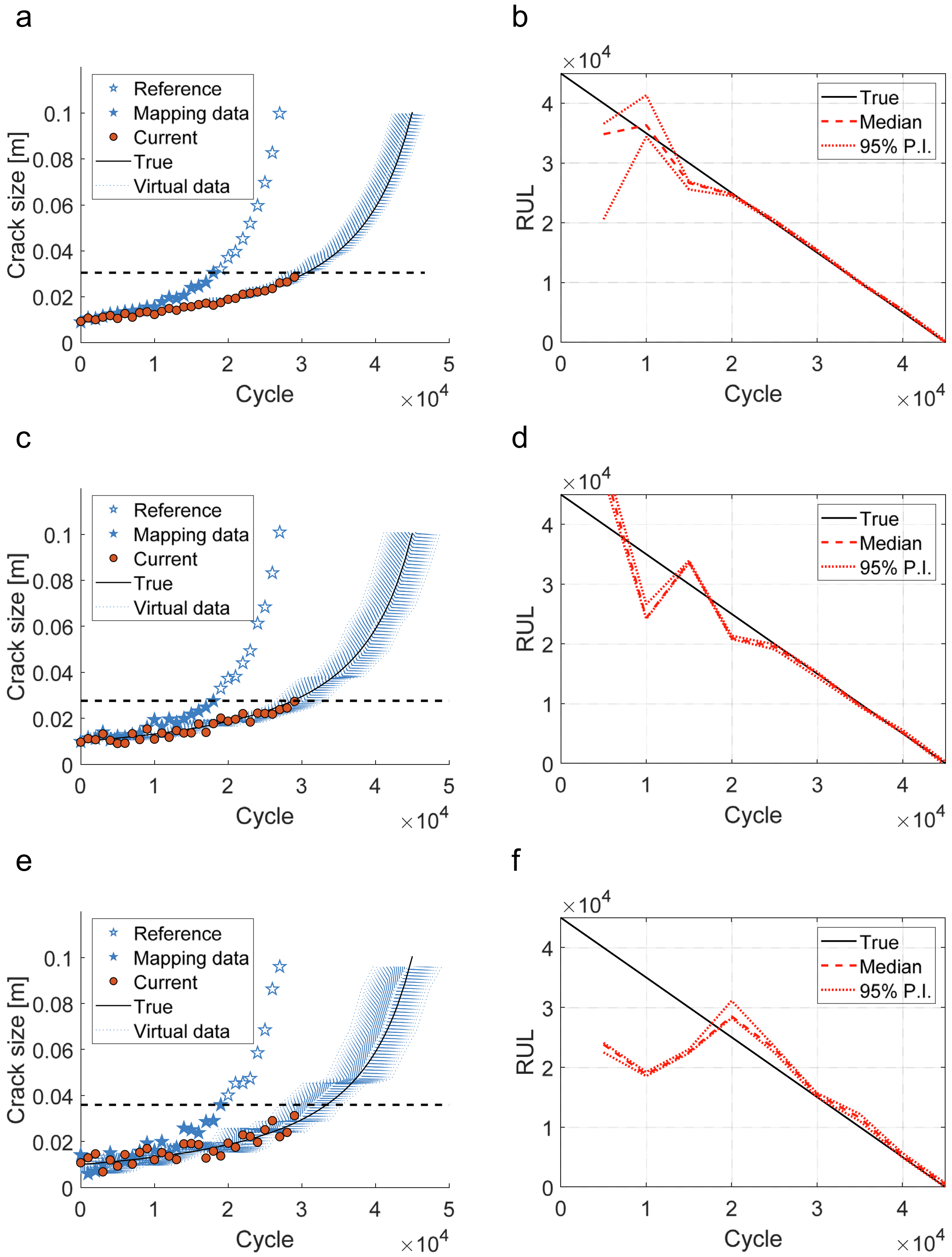


Fig. 11. Comparison of prognostics performance between different level of noise: (a) mapping result at 29,000 cycles (u = 1 mm), (b) RUL prediction under 1 mm noise, (c) mapping result at 29,000 cycles (u = 3 mm), (d) RUL prediction under 3 mm noise, (e) mapping result at 29,000 cycles (u = 5 mm) and (f) RUL prediciton under 5 mm noise.

4.2. Real crack growth data

Speaker et al. [35] performed fatigue crack growth tests to provide databases needed to evaluate and to demonstrate the durability analysis methodology. A large number of tests were performed using Al 7475-T7351 coupons with the 0.5-inch thickness. The tests were performed under various configurations, such as different load spectra, specimen, fastener type and load transfer. Among different configurations, this section uses two configurations, AFLR4 and AFHR4(A), and exploited two data sets from each of them, with a crack size of 0.35 in. to determine the end of life. The characteristics of the selected datasets are described in Table 3, and Fig. 12 shows the crack growth trend.

Among four data sets, AFLR4-6 data (yellow triangular marker) is regarded as the target data, while others are used for the reference data. Fig. 13(a) shows the result of mapping validation, where AFHR4(A)-10 shows the best mapping result. As can be seen in Fig. 13, RMSE of AFHR4(A)-9 and AFLR4-9 continuously grow after 6000 cycles, whereas that of AFHR4(A)-10 does not. As a result, the performance of RUL estimation using data augmentation based on AFHR4(A)-10 scores the lowest RMSE value. It is interesting to note that even if AFLR4-9 is close to AFLR4-6, DTW predicts that AFHR4(A)-10 is the optimal mapping. This is related to the fact that both AFLR4-6 and AFLR4-9 show almost identical crack growth in early cycles but deviate

Table 3
Test matrix [28].

Material	Spectrum	Transfer	σ(ksi)	Fastener type	Dia. (in.)	Specimen type	Data set designation
7475-T7351	F-16 400 HR BLOCK	0%	32.0	MS-90353	1/4	1A	AFLR4
7475-T7351	F-16 400 HR BLOCK	0%	38.0	MS-90353	1/4	1A	AFHR4(A)

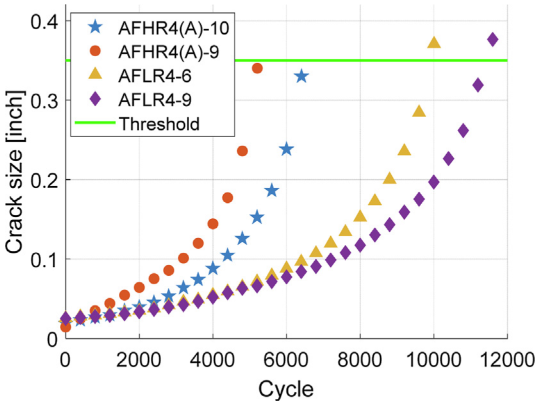


Fig. 12. Real crack growth data.

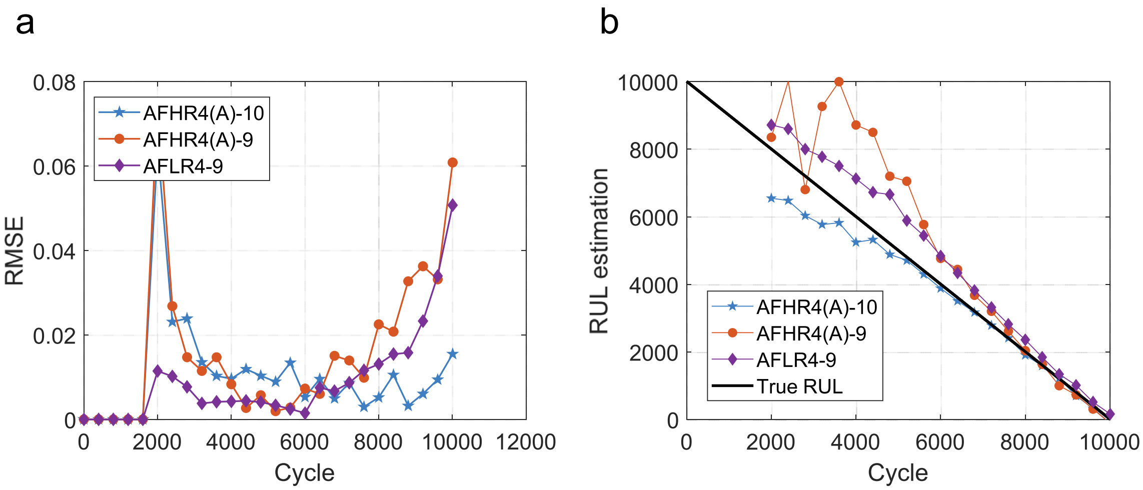


Fig. 13. Comparison of RUL prediction performance between reference data: (a) mapping performances of reference data and (b) RUL prediction result.

significantly by different growth rates, which indicates that the failure modes might be different. The accuracy in RMSE is also reflected in the RUL prediction in Fig. 13(b). When the current target data, AFLR4-6, is augmented using AFHR4(A)-10, the predicted RUL converges faster to the true RUL from the conservative side.

To verify the effectiveness of proposed algorithms, performance is compared with ordinary neural network based prognostics [11,12] which uses all the existing RTF data as the input for the training without mapping or augmentation. Considering the uncertainty in the training phase, the process is repeated 100 times with different initial weights. The same process is also applied to proposed approach which used AFHR4(A)-10 as the reference data. Fig. 14(a) shows the predicted RUL with 90% confidence intervals using the proposed method by augmenting with AFHR4(A)-10, while Fig. 14(b) shows the predicted RUL using all three datasets without augmenting.

As shown in the figure, prognostics using mapping and augmentation results in a narrower interval of uncertainty and higher accuracy in the RUL prediction. When all the RTF data are used as an input data for ANN without mapping and augmentation, not only the median fails to converge, but also the level of uncertainty does not reduce. This phenomenon implies that utilizing all the RTF may not always bring a good result.

4.3. Crack growth under variable amplitude loading

In practice, system is not always operated under constant loading condition. The proposed algorithm is also applicable to predicting the crack growth under variable amplitude loading conditions. Since the real data under variable loading are difficult to obtain, synthetic data of crack growth are generated using Huang's model that addresses variable loading condition, which is defined as follows [36]:

$$\frac{da}{dN} = C\left[\left\{\Delta K_{eq}(\beta, n, \sigma_Y)^m\right\} - \left\{\Delta K_{th}\right\}^m\right] \tag{11}$$

where, m, C, ΔK_{th}, β, n and σ_Y are model parameters. ΔK_{eq} is the range of equivalent stress intensity factor, considering the effect of crack tip plasticity and crack closure after overloading. Based on the above model, six crack growth data are generated under different loading spectrums with the threshold of 0.04 m to determine the end of life. As shown in Fig. 15, the loading block consists of a fixed minimum and nominal load, 5 MPa and 65 MPa, respectively. The amplitude of overload and the number of cycles vary for different loading conditions. In the following numerical example, five data sets are generated based on the configurations given in Table 4.

The results of crack growth simulation under different loading conditions are illustrated in Fig. 16. Predicting the behavior of crack growth is considered challenging because of crack retardation period when the magnitude of overload is decreased. Among the five data sets, dataset 3 is considered as target data, while other four datasets are used for reference data. Note that the target dataset is located not within the reference datasets, in which case the prediction may be inaccurate in the ordinary ANN approach than when within the reference datasets [17]. Prior to applying the proposed algorithm, the performance of an ordinary ANN is evaluated by training all four RTF data and applying to the target data, whose results are shown in Fig. 17. Since the crack growth rates of all training data are higher than the target data, the ANN underestimates the RUL of dataset 3. In addition, the level of uncertainty is not reduced even if the RUL is close to the true value near the end of life.

For the purpose of dynamic time warping, the RMSE is calculated as a function of time in order to find the best reference dataset. As shown in Fig. 18(a), it turns out that dataset 2 has the lowest RMSE and the error does not increase as the number

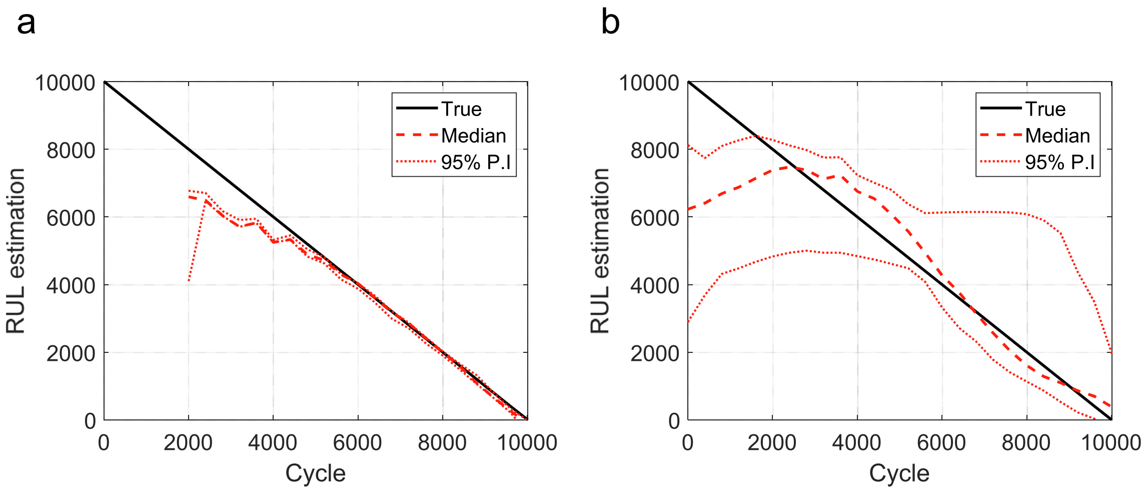
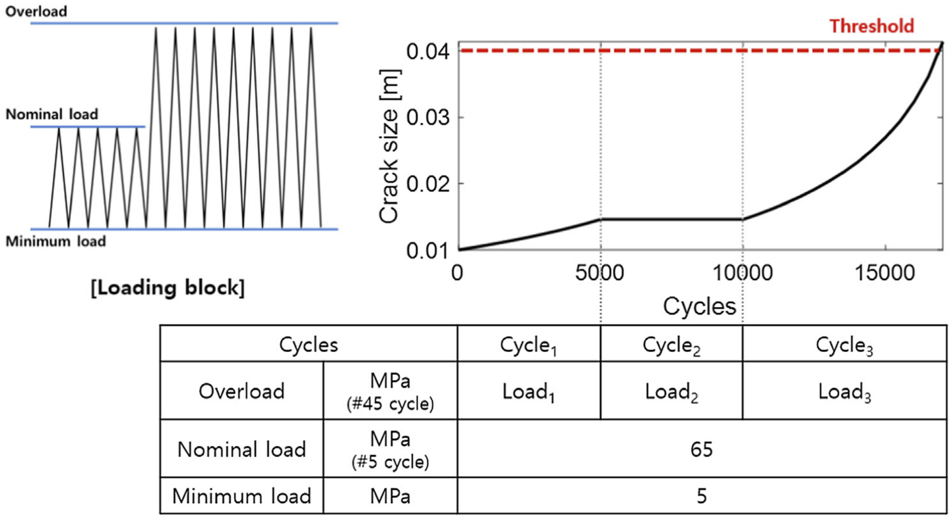


Fig. 14. RUL estimation performance comparison: (a) RUL prediction with mapping and augmentation and (b) RUL prediction without mapping.

Table 4
Five simulation data description (minimum load = 5 MPa, nominal load = 65 MPa).

Data 1	Cycles	5000	10,000	25,000
	Overload (MPa) (# 45 cycles)	125	100	140
Data 2	Cycles	5000	10,000	25,000
	Overload (MPa) (# 45 cycles)	125	100	120
Data 3	Cycles	5000	10,000	25,000
	Overload (MPa) (# 45 cycles)	125	100	110
Data 4	Cycles	4000	10,000	25,000
	Overload (MPa) (# 45 cycles)	120	90	120
Data 5	Cycles	4000	8000	25,000
	Overload (MPa) (# 45 cycles)	140	100	130



Cycles		Cycle$_1$	Cycle$_2$	Cycle$_3$
Overload	MPa (#45 cycle)	Load$_1$	Load$_2$	Load$_3$
Nominal load	MPa (#5 cycle)	65		
Minimum load	MPa	5		

Fig. 15. Crack growth simulation under variable amplitude loading.

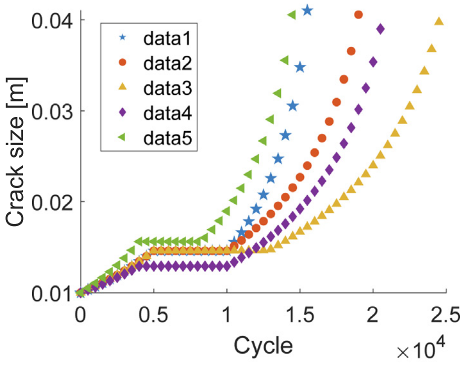


Fig. 16. Five different crack growth data under variable amplitude loading.

of cycles increases. Fig. 18(b) shows the RUL prediction results using four different reference data sets. Since dataset 2 shows the smallest RMSE, it also predicts the RUL best. Comparing Fig. 18(b) with Fig. 17, the proposed augmentation technique demonstrates that it can predict the RUL better and converges much faster.

4.4. NASA battery degradation

The lithium-ion battery is a very critical part for the performance of many engineering systems, whose unexpected failure can lead to the failure of entire system. Therefore, predicting the RUL of battery is crucial for the system safety. The proposed

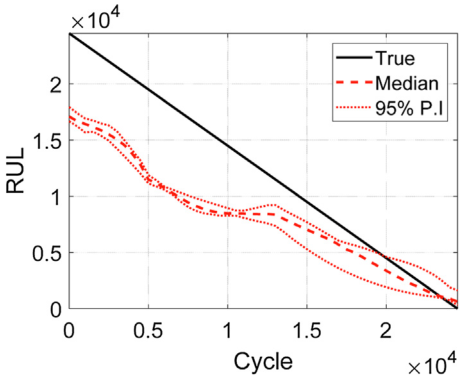


Fig. 17. RUL prediction of variable amplitude load case without mapping.

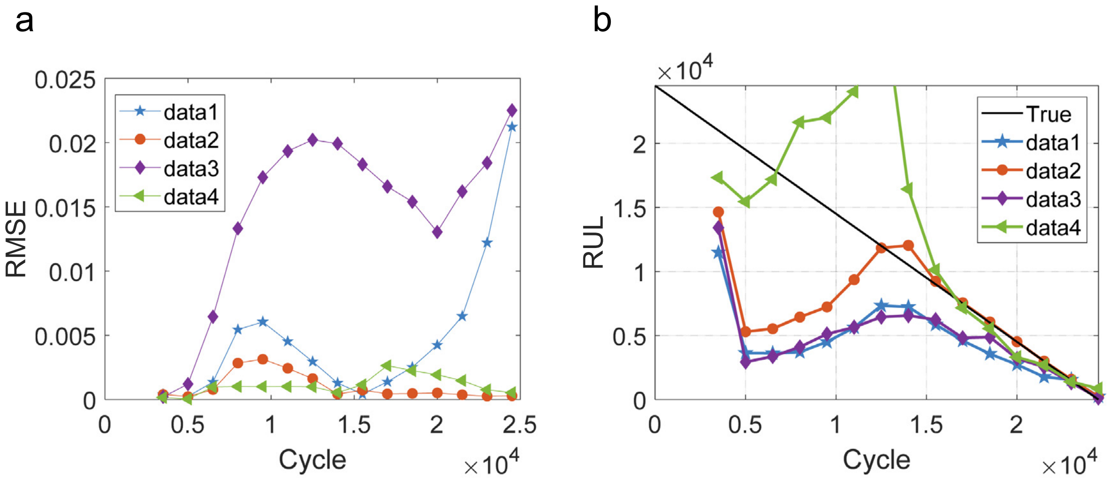


Fig. 18. RUL prediction using data augmentation: (a) RMSE trend of reference data and (b) results of RUL prediction using mapped reference data.

approach is applied to battery degradation data sets, which were obtained from data repository of NASA Ames Prognostics Center of Excellence (PCoE). Charging in a constant current (CC) mode at 1.5A was carried out until the battery voltage reached 4.2 V and then continued in a constant voltage (CV) mode until the charge current dropped to 20 mA. Discharge was carried out at a constant current (CC) level of 2A until the battery voltage fell to 2.7, 2.5 and 2.2 V for batteries 5, 6, and 7, respectively. Impedance was measured through an electrochemical impedance spectroscopy (EIS) frequency sweep from 0.1 Hz to 5 kHz. Repeated charge and discharge cycles result in accelerated aging of the batteries while impedance measurements provide insight into the internal battery parameters that change as gaining progress. The experiments were stopped when the batteries reached end-of-life (EOL) criteria, which was a 30% fade in rated capacity (from 2Ahr to 1.4Ahr) [37]. In the field of battery prognostics, State of Health (SOH) of battery is firstly estimated which is widely used as a measure for the battery health condition and predict RUL until it reaches to pre-defined threshold. Among several approaches determining battery SOH, this paper defines the battery SOH as follows [38]

$$SOH = \frac{C_i}{C_0} \times 100\% \tag{12}$$

where, C_i and C_0 represent the degenerated capacity at i-th cycle and the initial capacity, respectively. For a normal battery, the initial capacity was 2Ahr. Fig. 19(a) shows battery degradation data sets used in this paper. Among the three datasets, Batteries #5 and #6 are used as the past reference datasets, while Battery#7 is selected as the current target dataset. Note that the target dataset is located out of the two reference datasets, which can cause a difficulty in prediction. First, the two reference data sets are used to train ANN, from which the RUL of the target dataset is predicted without mapping, whose results are shown in Fig. 19(b). As the two reference datasets show faster degradation, the predicted RUL is significantly underestimated and failed to converge to the true RUL. On the other hand, after applying the proposed mapping method,

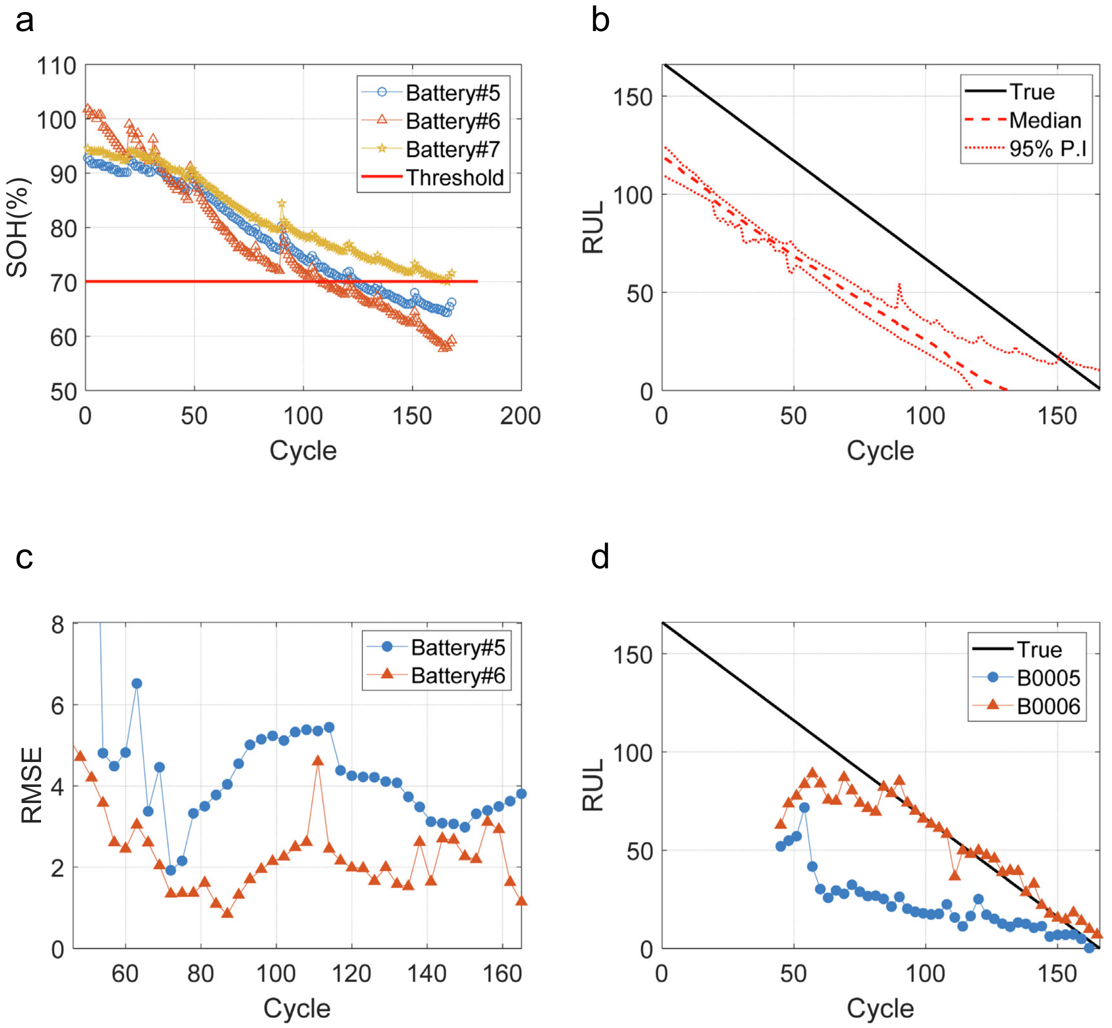


Fig. 19. Application to NASA battery data sets: (a) SOH trend of Battery data sets, (b) RUL prediction without mapping, (c) RMSE trend of the two reference datasets, and (d) RUL prediction after mapping.

Battery #6 shows a lower RMSE (see Fig. 19(c)) than that of Battery #5, and because of this, the predicted RUL using the mapping path of Battery #6 is much better than that of Battery #5 as shown in Fig. 19(d), which illustrates that the degradation trend of #7 is closer to that of #6 than #5.

5. Conclusions

Data-driven prognostics requires a lot of run-to-fail (RTF) data, which is not easy to obtain in real applications. To overcome this difficulty of 'data deficiency', this paper has proposed a data augmentation technique that searches the optimally matching with the existing reference data available from several sources: (1) scarce RTF data, (2) accelerated life test data, and (3) RTF data under different operating conditions. The algorithm generates virtual RTF data useful for the current operation condition using the reference data by searching for the closest match. ANN model is then trained using the virtual RTF data to predict the RUL. We have also proposed RMSE-based performance criterion to find the best RTF data among different sets, which might represent different failure modes or different degradation characteristics. Crack growth simulation data that are generated from Paris law and Huang model are employed to verify the effectiveness of the proposed algorithm. In addition, the real RUL of crack growth under F-16 loading spectrum has been successfully predicted by using proposed algorithm. Furthermore, the proposed algorithm, has been applied to the NASA battery degradation data with good accuracy. For more practical application, future works may include the quantification of uncertainty in the virtual data to find the optimum size of virtual data. In addition, exploitation of the censored data may be another issue for the RUL prediction, since the failure occurrence is rare in the real practice.

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.

Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2019R1A2C2010028).

References

[1] A.K.S. Jardine, D. Lin, D. Banjevic, A review on machinery diagnostics and prognostics implementing condition-based maintenance, Mech. Syst. Signal Process. 20 (7) (2006) 1483–1510.
[2] M. Pecht, R. Jaai, A prognostics and health management roadmap for information and electronics-rich systems, Microelectron. Reliab. 50 (3) (2010) 317–323.
[3] J. Lee, F. Wu, W. Zhao, M. Ghaffari, L. Liao, D. Siegel, Prognostics and health management design for rotary machinery systems – reviews, methodology and applications, Mech. Syst. Signal Process. (2014).
[4] D. An, N.H. Kim, J.H. Choi, Practical options for selecting data-driven or physics-based prognostics algorithms with reviews, Reliabil. Eng. Syst. Saf. (2015).
[5] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, J. Lin, Machinery health prognostics: A systematic review from data acquisition to RUL prediction, Mech. Syst. Signal Process. (2018).
[6] T. Xia, Y. Dong, L. Xiao, S. Du, E. Pan, L. Xi, Recent advances in prognostics and health management for advanced manufacturing paradigms, Reliabil. Eng. Syst. Saf. (2018).
[7] D. An, J.H. Choi, N.H. Kim, Prognostics 101: A tutorial for particle filter-based prognostics algorithm using Matlab, Reliab. Eng. Syst. Saf. (2013).
[8] P. Lall, R. Lowe, K. Goebel, Prognostics health management of electronic systems under mechanical shock and vibration using Kalman filter models and metrics, IEEE Trans. Ind. Electron. (2012).
[9] L. Peel, Data driven prognostics using a Kalman filter ensemble of neural network models, 2008 International Conference on Prognostics and Health Management, 2008.
[10] F.O. Heimes, Recurrent Neural Networks for Remaining Useful Life Estimation, Progn. Heal. Manag. 2008. PHM 2008. Int. Conf., 2008.
[11] G.S. Babu, P. Zhao, X.-L. Li, Deep convolutional neural network based regression approach for estimation of remaining useful life, in: International Conference on Database Systems for Advanced Applications, 2016, pp. 214–228.
[12] S. Zheng, K. Ristovski, A. Farahat, C. Gupta, "Long short-term memory network for remaining useful life estimation, in: Prognostics and Health Management (ICPHM), 2017 IEEE International Conference on, 2017, pp. 88–95.
[13] L. Liao, F. Köttig, Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an application to battery life prediction, IEEE Trans. Reliab. (2014).
[14] A. Widodo, B.S. Yang, Application of relevance vector machine and survival probability to machine degradation assessment, Expert Syst. Appl. 38 (3) (2011) 2592–2599.
[15] C. Hu, B.D. Youn, T. Kim, P. Wang, A co-training-based approach for prediction of remaining useful life utilizing both failure and suspension data, Mech. Syst. Signal Process. 62 (2015) 75–90.
[16] C. Sobie, C. Freitas, M. Nicolai, Simulation-driven machine learning: Bearing fault classification, Mech. Syst. Signal Process. 99 (2018) 403–419.
[17] D. An, J.-H. Choi, N.H. Kim, Prediction of remaining useful life under different conditions using accelerated life testing data, J. Mech. Sci. Technol. 32 (6) (2018) 2497–2507.
[18] A. Barré, F. Suard, M. Gérard, D. Riu, A real-time data-driven method for battery health prognostics in electric vehicle use, in: Proceedings of the Second Second European Conference of the Prognostics and Health Management Society, 2014, pp. 1–8.
[19] L. Tao, C. Lu, A. Noktehdan, Similarity recognition of online data curves based on dynamic spatial time warping for the estimation of lithium-ion battery capacity, J. Power Sources 293 (2015) 751–759.
[20] V. Atamuradov, F. Camci, S. Baskan, M. Sevkli, Failure diagnostics for railway point machines using expert systems, 2009 IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives, SDEMPED 2009, 2009.
[21] Byoung-Kee Yi, H.V. Jagadish, C. Faloutsos, Efficient retrieval of similar time sequences under time warping, 2002.
[22] S.W. Kim, S. Park, W.W. Chu, An index-based approach for similarity search supporting time warping in large sequence databases, Proc. - Int. Conf. Data Eng. (2001).
[23] S. Park, W.W. Chu, J. Yoon, C. Hsu, Efficient searches for similar subsequences of different lengths in sequence databases, 2002.
[24] E. Keogh, C.A. Ratanamahatana, Exact indexing of dynamic time warping, Knowl. Inf. Syst. 7 (3) (2005) 358–386.
[25] R. Han, Y. Li, X. Gao, S. Wang, An accurate and rapid continuous wavelet dynamic time warping algorithm for end-to-end mapping in ultra-long nanopore sequencing, Bioinformatics (2018).
[26] H. Kaprykowsky, X. Rodet, Globally optimal short-time dynamic time warping application to score to audio alignment, ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing – Proceedings, 2006.
[27] F. Petitjean, A. Ketterlin, P. Gançarski, A global averaging method for dynamic time warping, with applications to clustering, Pattern Recognit. (2011).
[28] A.K. Mahamad, S. Saon, T. Hiyama, Predicting remaining useful life of rotating machinery based artificial neural network, Comput. Math. with Appl. 60 (4) (2010) 1078–1087.
[29] A. Messai, A. Mellit, I. Abdellani, A. Massi Pavan, On-line fault detection of a fuel rod temperature measurement sensor in a nuclear reactor core using ANNs, Prog. Nucl. Energy 79 (2015) 8–21.
[30] Z. Tian, L. Wong, N. Safaei, A neural network approach for remaining useful life prediction utilizing both failure and suspension histories, Mech. Syst. Signal Process. 24 (5) (2010) 1542–1555.
[31] F. Ahmadzadeh, J. Lundberg, Remaining useful life prediction of grinding mill liners using an artificial neural network, Miner. Eng. 53 (2013) 1–8.
[32] R.J. Meszlényi, P. Hermann, K. Buza, V. Gál, Z. Vidnyánszky, Resting state fMRI functional connectivity analysis using dynamic time warping, Front. Neurosci. (2017).
[33] P.C. Paris, F. Erdogan, A Critical Analysis of Crack Propagation Laws, J. Basic Eng. 85 (1960) 528–534.
[34] W. Stefansky, Rejecting outliers by maximum normed residual, Ann. Math. Stat. 42 (1) (1971) 35–45.
[35] S.M. Speaker, D.E. Gordon, W.T. Kaarlela, A. Meder, R.O. Nay, Durability Methods Development. Volume 8. Test and Fractography Data, GENERAL DYNAMICS FORT WORTH TX FORT WORTH DIV, 1982.
[36] X. Huang, M. Torgeir, W. Cui, An engineering model of fatigue crack growth under variable amplitude loading, Int. J. Fatigue, 2008
[37] B. Saha, K. Goebel, Battery data set, NASA ames prognostics data repository. NASA Ames, Moffett Field, CA; 2007.
[38] D. Liu, J. Pang, J. Zhou, Y. Peng, M. Pecht, Prognostics for state of health estimation of lithium-ion batteries based on combination Gaussian process functional regression, Microelectron. Reliab. (2013).