# Remaining Useful Life Prediction for a Roller in a Hot Strip Mill Based on Deep Recurrent Neural Networks

Ruihua Jiao, Kaixiang Peng, *Member, IEEE*, and Jie Dong

***Abstract*—Accurate estimation of the remaining useful life (RUL) and health state for rollers is of great significance to hot rolling production. It can provide decision support for roller management so as to improve the productivity of the hot rolling process. In addition, the RUL prediction for rollers is helpful in transitioning from the current regular maintenance strategy to conditional-based maintenance. Therefore, a new method that can extract coarse-grained and fine-grained features from batch data to predict the RUL of the rollers is proposed in this paper. Firstly, a new deep learning network architecture based on recurrent neural networks that can make full use of the extracted coarsegrained fine-grained features to estimate the heath indicator (HI) is developed, where the HI is able to indicate the health state of the roller. Following that, a state-space model is constructed to describe the HI, and the probabilistic distribution of RUL can be estimated by extrapolating the HI degradation model to a predefined failure threshold. Finally, application to a hot strip mill is given to verify the effectiveness of the proposed methods using data collected from an industrial site, and the relatively low RMSE and MAE values demonstrate its advantages compared with some other popular deep learning methods.**

***Index Terms*—Hot strip mill, prognostics and health management (PHM), recurrent neural network (RNN), remaining useful life (RUL), roller management.**

Manuscript received May 23, 2020; revised July 9, 2020; accepted August 19, 2020. This work was supported by the Natural Science Foundation of China (NSFC) (61873024, 61773053), Fundamental Research Funds for the China Central Universities of USTB (FRF-TP-19-049A1Z), and the National Key RD Program of China (2017YFB0306403). Recommended by Associate Editor Yanjun Liu. *(Corresponding author: Kaixiang Peng.)*

Citation: R. H. Jiao, K. X. Peng, and J. Dong, "Remaining useful life prediction for a roller in a hot strip mill based on deep recurrent neural networks," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 7, pp. 1345–1354, Jul. 2021.

R. H. Jiao is with the Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, and also with the AVIC Xi'an Aviation Brake Technology Co., Ltd., Xi'an 710065, China (e-mail: jorry0123@163.com).

K. X. Peng is with the Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, and also with the Institute of Artificial Intelligence, University of Science and Technology Beijing, Beijing 100083, China (e-mail: kaixiang@ustb.edu.cn).

J. Dong is with the Key Laboratory of Knowledge Automation for Industrial Processes of Ministry of Education, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China (e-mail: dongjie@ies.ustb.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2021.1004051

## I. Introduction

HOT rolling is one of the most widely used processes in steel production. The roller is the most important tool in hot rolling production, since its quality and service life directly affects the stability of strip production and can further influence the production efficiency. Once the roller is damaged, it will not only lead to the fracture of the strip, but can cause a fire due to the high temperatures of more than 1000 °C during the rolling process. The roller is a consumable part, so it should be maintained every day. However, as quality requirements of products continue to increase, higher demands are placed on the use of rollers. As a consequence, maintenance strategy should shift from traditional breakdown maintenance and preventive time-based maintenance to condition-based maintenance (CBM), which is also called predictive maintenance (PdM) or prognostics and health management (PHM) [1].

PHM is an engineering process of failure prevention, and remaining useful life (RUL) prediction [2]. The core process of PHM is to make reliable predictions of the RUL [3], [4]. The RUL of a roller refers to the lifetime left on the component from the current time to end-of-life (EoL). Accurate RUL prediction is able to provide helpful information for the management of rollers. It can not only guarantee the surface quality of the hot rolled products and avoid losses caused by accidental failure of the roller, but it also improves the operating rate of the rolling mill so as to increase production efficiency of the business. Therefore, accurate RUL prediction for rollers is of great significance for the production of hot rolling, and special attention needs to be paid to the design of a reliable and accurate RUL prediction approach.

Generally, the existing RUL prediction methods can be classified into three categories: model-based, data-driven, and hybrid approaches [5]. The basic idea of the model-based method is to develop a mathematical model that can describe the physical characteristics and failure modes of the system to realize the prediction of the RUL. It usually achieves more accurate results but cannot be applied to systems that lack prior knowledge about the physical degradation. The widely used model-based approaches include the Paris-Erdogan model [6], the Kalman filter [7], and the particle filter [8]. Data-driven methods are able to construct a mapping relationship between input and output based on a large amount of historical data and then predict the RUL, which can avoid

the disadvantages of model-based method. A large number of data-driven approaches, such as the autoregressive model [9], the proportional hazard model [10], the Wiener process [11], the support vector machine (SVM) [12], and the artificial neural networks [13], have been studied in the past few years and have made remarkable achievements. By combining several of the aforementioned methods, hybrid approaches are able to leverage advantages of different models while avoiding their disadvantages at the same time [14]. For instance, Wei *et al.* [15] proposed a framework that integrates SVM and particle filter to estimate the RUL of batteries, which not only provides a possible failure time range, but also improved the accuracy of prediction results compared with conventional methods. In [16], the particle filter algorithm is introduced to estimate the state of Wiener process to remove the influence of multisource variability and survival measurements. However, developing an effective and reliable hybrid approach still remains challenging, in particular the combination of model-based and data-driven methods. Accordingly, a new hybrid approach is proposed in this paper.

Recently, since the multilayer deep network architecture can fully capture the representative features from original measured data, deep learning methods have been widely used in many fields, such as speech recognition, image recognition, and fault diagnosis [17]. It has also been gradually applied to the field of PHM. In [18], an enhanced restricted Boltzmann machine was proposed to construct a heath indicator (HI), and the RUL of the machines were estimated through a similarity-based method. The experimental results demonstrate the advantages of the proposed method. Li *et al.* [19] proposed a data-driven method based on a deep convolution neural network for RUL prediction, where accurate prediction results can be obtained without prior knowledge and signal processing. Compared to the previously mentioned deep network architecture, the recurrent neural network (RNN) is a more effective approach for processing time-series data since they have an internal state that can represent aging information [20]. Heimes [21] presented a RUL prediction model based on the RNN for turbofan engines, which won the 2008 PHM competition. Guo *et al.* [22] proposed a HI construction method based on RNN for predicting the RUL of bearings, where the experiment showed that the approach has better performance than some other methods. These applications indicate that the RNN-based approach has enormous potential for PHM and RUL prediction.

In the hot strip production process, the roller is in direct contact with the metal to be rolled in order to plastically deform the metal. Due to the large rolling force of the hot strip mill (HSM), the friction between the work roller and the rolled metal is extremely high. Accordingly, wear becomes the major factor affecting the service life of the rollers [23]. With the increase of rolling time, the wear degree gradually increases, which will cause the surface of the roller to become rough, requiring maintenance of the component. The traditional theory of roller RUL prediction is based on the prediction of wear. However, there are still some lacunae which are worth discussing. First, these theories are based on an empirical model and only consider central influencing factors such as rolling pressure, contact arc length, and rolling length, which is a simplification of the real situation. At present, there still does not exist an accurate prediction model based on the wear mechanism. Oike's equation [24] is a widely used empirical model. However, it cannot predict the wear precisely when applied to different industrial sites. In [25], a roller force model was integrated with Oike's equation to estimate the wear along the roller barrel, and the accuracy of prediction was improved. Secondly, the traditional model requires some important parameters, such as the coefficient of friction and the uneven friction coefficient of the steel plate, which are unmeasurable quantities in industrial cases. Although the empirical relationships between these coefficients and other measurable parameters, like rolled material, strip temperature, and lubrication, can be derived, these relationships are sometimes contradictory [26]. Thirdly, it is unreasonable that the RUL of the roller be determined only by the degree of wear. Roller deformation, roller fatigue, mechanical factors, and the processes of rolling also affect the service life of the roller. It is difficult to establish a mathematical model to describe the relationship between these factors and the RUL.

In order to overcome these shortcomings in predicting the RUL of rollers, a new method which can extract coarse-grained and fine-grained characteristics to estimate the health state and predict the RUL is proposed in this study. To form a feature set that contains various roller degradation characteristics for accurate RUL prediction, the coarse-grained and fine-grained characteristics are extracted from batch data based on the proposed deep RNN. Then, the features are used to construct a HI that is capable of indicating the health state of the roller. Following that, the RUL can be estimated by extrapolating the HI model, which is described by a double exponential function, to a predefined failure threshold (FT). Finally, the effectiveness of the proposed method is verified by the dataset collected from an industrial site.

For the approach proposed in this work, the main contributions and innovations are:

1) The proposed deep RNN architecture is able to extract coarse-grained features from monitoring data and fine-grained features from maintenance data to develop a comprehensive HI that can reflect the health state of the roller. Furthermore, the model parameters can be obtained automatically during network training, rather than the model parameters of traditional methods are sometimes unmeasurable.

2) The RUL of the roller is determined by the developed comprehensive HI rather than relying on just one factor as in the conventional method, which is more reasonable. In addition, the constructed HI has a value equal to one under fault state, so there is no need to artificially specify a FT.

3) A RUL prediction and health state estimation framework based on deep RNN is proposed for the RUL estimation of the roller of HSM, where many monitored variables related to the roller wear that are not considered by empirical models are taken into account in this work.

## II. Proposed Deep RNN Network Architecture

### A. Basic Theory of Recurrent Neural Network

As a kind of deep learning approach, RNN is well suited for

dealing with sequential data. This can be attributed to the special network structure that remembers previous information and applies it to the calculation of the current output. As shown in Fig. 1, at time $t$, the input of the hidden layer comes not only from the input layer $x_t$, but also from its own output at the previous moment $h_{t-1}$. As a result, the output $y_t$ is determined by both the present input information and the information at time $t-1$.

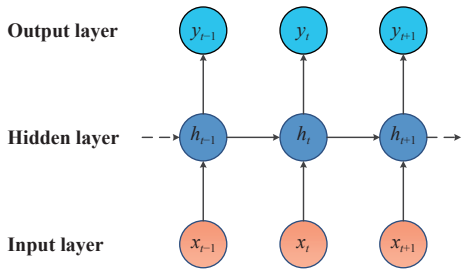


Fig. 1. Structural diagram of the RNN.

The RNN is generally trained using a back-propagation through time (BPTT) algorithm. However, as the time information stored in the RNN increases, the gradients tend to vanish or explode [27]. In order to overcome this problem, a long short-term memory (LSTM) network architecture including memory cells is proposed [28]. The memory cell is introduced to replace the hidden neuron in the hidden layer of a traditional RNN. It records the information for a long time through a forget gate, an input gate, and an output gate. There is an initial cell state in the memory cell, which is assigned with certain unit weight. All components of the LSTM are shown in Fig. 2.

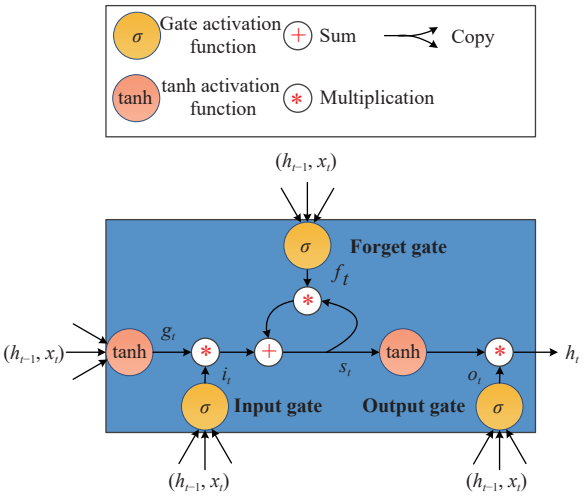


Fig. 2. The network architecture of the LSTM memory cell.

*Forget Gate:* The forget gate $f_t$ is used to decide how much information will be discarded. With a sigmoid layer, it outputs a number between 0 and 1, where 1 indicates the cell state value is fully reserved and 0 represents the value is completely forgotten. The forget gate can be calculated as

$$f_t = \sigma\left(w_{fx}x_t + w_{fh}h_{t-1} + b_f\right). \tag{1}$$

*Input Gate:* The input gate $i_t$ is used to determine what new information is to be stored in the cell state and to update the cell state. This step consists of two parts. First, a sigmoid layer is used to update the input value. After that, a tanh layer $g_t$ is used to create a candidate state, which can be multiplied by the input value to update the cell state. These can be implemented as

$$i_t = \sigma(w_{ix}x_t + w_{ih}h_{t-1} + b_i) \tag{2}$$

$$g_t = \tanh\left(w_{gx}x_t + w_{gh}h_{t-1} + b_g\right). \tag{3}$$

With (1)–(3), the current cell state can be updated as

$$s_t = f_t \times s_{t-1} + i_t \times g_t. \tag{4}$$

*Output Gate:* The output gate $o_t$ is introduced to decide which information will be output. First, a sigmoid layer is used to determine which part of the cell state will be output. Then, the remaining state value can be obtained by multiplying the outputs of a tanh layer and the sigmoid layer. This can be calculated as

$$o_t = \sigma(w_{ox}x_t + w_{oh}h_{t-1} + b_o) \tag{5}$$

$$h_t = \tanh(s_t) \times o_t \tag{6}$$

where $w_{fx}$, $w_{ix}$, $w_{gx}$, and $w_{ox}$ are the weights of the input layer to the hidden layer; $w_{fh}$, $w_{ih}$, $w_{gh}$, and $w_{oh}$ are hidden layer weights between time $t$ and $t-1$; and $b_f$, $b_i$, $b_g$, and $b_o$ are the biases.

### B. Proposed Network Architecture

The hot rolling process is a typical batch process, which produces strip steel coil-by-coil and each coil represents a batch [29]. Considering that the HSM production process has its own unique characteristics, it is necessary to design a targeted modeling method. The HSM can be used to produce a wide range of products in different materials, shapes, and sizes, and different products inevitably lead to different wear rates. The HSM processes a large number of products by continuously repeating the rolling operation. Thus, a certain periodicity may exist between each batch of the monitored data. In view of the above characteristics, a deep RNN model is proposed that can fully extract the fine-grained features from the monitoring data and the coarse-grained features from the maintenance data. The monitoring data refers to the data collected during the rolling process, which contains approximately 180 measuring points in each batch. Thus, fine-grained features can be used to model wear rate under different working conditions. The maintenance data refers to the data collected after rolling a batch of steel, where each batch of steel contains only one measurement point. Thus, coarse-grained features can be used to model the periodicity of rolling. The network architecture consists of three sub-structures, i.e., a multilayer LSTM for extracting fined-grained features which is represented as LSTM1, a multilayer LSTM for extracting coarse-grained features which is represented as LSTM2, and a fully connected layer for regression. Fig. 3 shows the network architecture of the proposed approach.

First, the collected data is prepared as input data for the

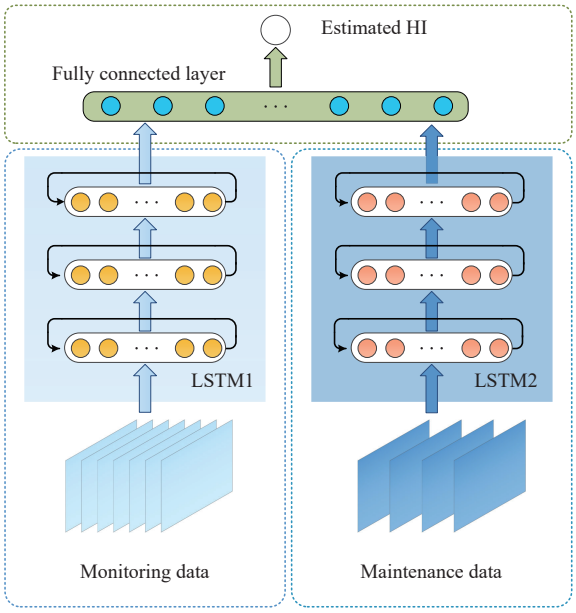


Fig. 3. The architecture of the proposed deep RNN.

LSTM in 3-dimensional (3D) format for the convenience of training. The dimension of the data is $N_s \times N_{tw} \times N_{id}$, where $N_s$ is the number of samples, $N_{tw}$ indicates the time window, and $N_{id}$ denotes the number of selected input dimension.

Afterwards, the multilayer LSTM1 is developed to extract fine-grained features from the monitoring data, and the multilayer LSTM2 is constructed to extract coarse-grained features from the maintenance data, which can both learn the most suitable feature representations from each raw input data. After model convergence, each vector of the outputs in the two multilayer LSTMs naturally represents the fine-grained and the coarse-grained features, respectively.

Finally, all the extracted features are connected to a fully connected layer, and one neuron as a representative of the output is added at the top of the network for HI estimation. To further improve the performance of the proposed deep RNN, the Adam algorithm [30] is used to optimize the neural network. In addition, the dropout [31] technique is used to regularize the hidden layers, which is able to alleviate overfitting and improve generalization ability. The parameters of the deep RNN model are optimized to minimize the error between the actual HI and the estimated HI.

## III. Remaining Useful Life Prediction

### A. Health Indicator Construction

In order to develop a comprehensive HI to assess the health state of the roller, the proposed deep RNN is used to fuse the fined-grained features and coarse-grained features as the HI. Specifically, in the training step, a label set indicating the percentage of roller degradation at time $t$ is attached to the input data set. Thus, the HI can be expressed as

$$HI_t = \frac{t}{T} \tag{7}$$

where $HI_t$ represents the label at moment $t$, and $T$ is the number of steel batches rolled during the last maintenance to the next maintenance. For example, assuming that the roller maintenance time is after rolling 380 batches of steel, and the current monitoring time point is the 228th batch of steel, then the label HI is 0.6. Accordingly, the deep RNN model is trained to minimize the cost function as follow:

$$J = \frac{1}{2}\sum_{t=0}^{T}\left\|HI_t - \widehat{HI}_t\right\|_2^2 \tag{8}$$

where $\widehat{HI}_t$ is the output of the deep RNN model. In the testing step, the monitored data are directly input into the trained deep RNN model for HI estimation. As can be derived from (7) that the FT of the constructed HI is expected to equal to one. Thus, the RUL of roller can be predicted based on the fixed FT value.

### B. Remaining Useful Life

For the purpose of estimating the RUL, the double exponential model is introduced to describe the HI model, which has been proved to be an effective model for RUL prediction [27]. Thus, the state-space model can be built as follow:

$$\begin{cases} z_{1,k} = z_{1,k-1} + N\left(0,\sigma_1^2\right) \\ z_{2,k} = z_{2,k-1} + N\left(0,\sigma_2^2\right) \\ z_{3,k} = z_{3,k-1} + N\left(0,\sigma_3^2\right) \\ z_{4,k} = z_{4,k-1} + N\left(0,\sigma_4^2\right) \\ HI_k = z_{1,k} \times \exp(z_{2,k} \times k) + z_{3,k} \times \exp(z_{4,k} \times k) + \eta_k \end{cases} \tag{9}$$

where $z_k = [z_{1,k}, z_{2,k}, z_{3,k}, z_{4,k}]$ is the state variable at time $k$; $N\left(0,\sigma^2\right)$ is the Gaussian noise with mean zero and variance $\sigma^2$; and $\eta_k$ is the independent measurement noise.

The particle filter (PF) algorithm is used to predict the future HI based on this state-space model. The main idea of PF is to sample a large number of random particles using the Monte Carlo method, and give each particle an importance weight to represent the posterior probability density. More details about PF can be found in [32]. The PF algorithm includes the following four steps:

*1) Particle Initialization:* At time $k = 0$, the initial particles $\left\{z_0^i\right\}_{i=1}^N$ are produced from a priori probability density function (PDF) $p(z_0)$, and the weights are assigned as $w_o^i = 1/N$.

*2) Particle Update:* Weights are calculated at time $k$ as

$$w_k^i = w_{k-1}^i \frac{p\left(HI_k|z_k^i\right)p\left(z_k^i|z_{k-1}^i\right)}{q\left(z_k^i|z_{k-1}^i, HI_{1:k}\right)} \tag{10}$$

where $q\left(z_k^i|z_{k-1}^i, HI_{1:k}\right)$ is the important density function which is selected as the prior probability density function in this paper, and normalized as

$$w_k^i = w_k^i \Big/ \sum_{j=1}^{N} w_k^j. \tag{11}$$

*3) Particle Resampling:* The new particle set $\left\{\hat{z}_k^i\right\}_{i=1}^N$ is generated by duplicating the particles with large weights and assigned with equal weight $1/N$.

*4) State Estimation:* The new state is calculated by using

new particles and weights as

$$\hat{z}_k^i = \frac{1}{N}\sum_{i=1}^{N}\hat{z}_k^i. \tag{12}$$

Once the $t$ measurements of the HI have been obtained, the posterior PDF of the state variable at time $t$ can be derived with the particle states as

$$p(z_t|HI_{0:t}) \approx \sum_{i=1}^{N} w_t^i \delta\left(z_t - z_t^i\right) \tag{13}$$

where $N$ is the number of particles; $w_t^i$ denotes the importance weight of particle; $z_t^i$ is the state variable estimated by the $i$th particle; and $\delta(\cdot)$ is the Dirac delta function. Based on (9), the $k$-step ahead prediction of the corresponding PDF is calculated using

$$p(z_{t+k}|HI_{0:t}) \approx \sum_{i=1}^{N} w_t^i \delta\left(z_{t+k} - z_{t+k}^i\right). \tag{14}$$

and the prediction value of the state is

$$\tilde{z}_{t+k} \approx \sum_{i=1}^{N} w_t^i z_{t+k}^i. \tag{15}$$

Accordingly, the HI can be predicted by substituting (15) into (9). Suppose that $L_t^i$ is the failure time that is estimated by the $i$th particle, which can be calculated by extrapolating the state-space model until the HI exceeds the predefined FT for the first time. Thus, the RUL associated with the $i$th particle is obtained as follows:

$$RUL_t{}^i = \left\{L_t^i - t\right\}_{i=1}^{N} \tag{16}$$

and the PDF of the RUL prediction is further can be estimated by

$$p\left(RUL_t{}^i|HI_{o:t}\right) \approx \sum_{i=1}^{N} w_t^i \delta\left(RUL_t - RUL_t{}^i\right). \tag{17}$$

The whole process of the proposed RUL prediction framework is shown in Fig. 4.

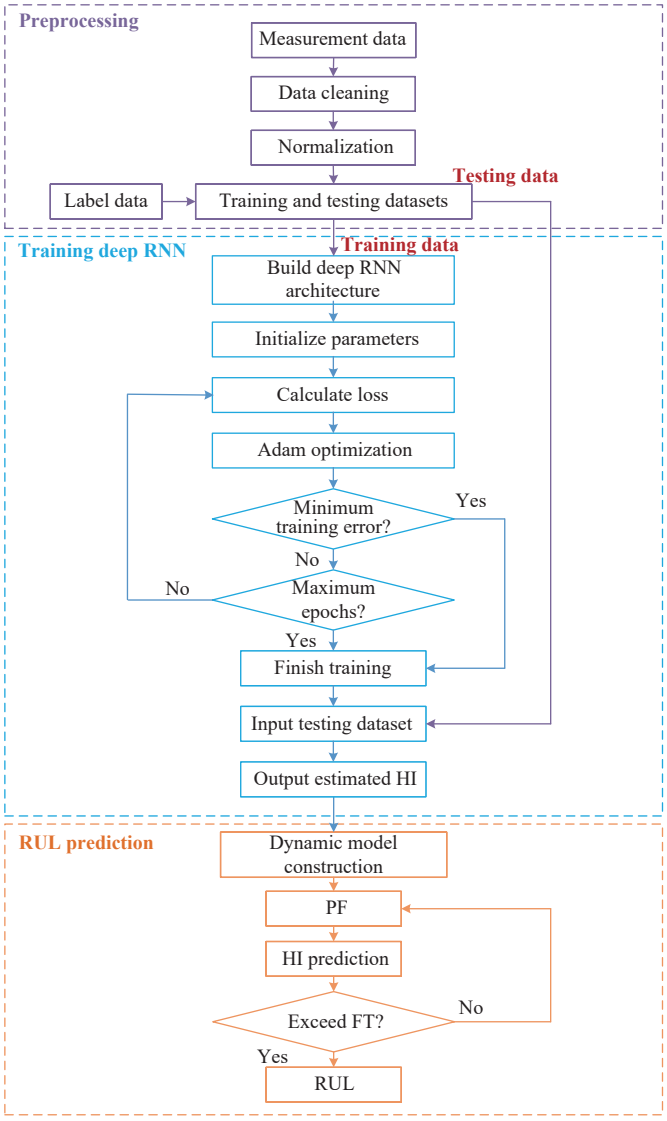


Fig. 4. Flowchart of proposed RUL prediction framework.

## IV. Experimental Study: Application to the HSM

### A. Descriptions of the HSM

The modern hot strip rolling process is a very complex process, and it is able to customize production based on order. In general, the hot strip rolling process consists of six sub-processes as shown in Fig. 5 [33]. The steel slab is first heated in the reheating furnace to about 1200 °C–1280 °C, then it is rolled back and forth in the roughing mill until the thickness of the slab reduces to the desired thickness for the finishing mill. After being passed through the transfer table, the thickness and width are shaped as expected in the finishing mill. Following that, the extremely hot strip steels are cooled by a laminar cooling system, and then the cooled strip is coiled for the convenience of storage. The finishing process is the core of the hot strip rolling line, and the control function of product quality is mainly concentrated in this area. Therefore, the work roller of the finishing mill will serve as the focus in this work.

The experimental data set was collected from the HSM of a 1700-mm hot rolling line in March 2019. It is generally believed that the rollers have reached the end of their service life and need to be replaced after rolling a large number of batches of steel. The work roller maintenance data of the last stand were collected during 12 cases of maintenance, and the corresponding rolling monitored data were also collected. Among them, 10 of the collected data sets were treated as the training set, and the other two were used to test the performance of the proposed method.

### B. Data Preprocessing

Fifteen variables were used in total. Ten of them were input to LSTM1 as shown in Table I, which are collected during the rolling process. The other five variables collected after rolling a batch of steel were used as the input for LSTM2, and the detailed descriptions are shown in Table II. Due to the difference in rolling time for each batch, as shown in Fig. 6,

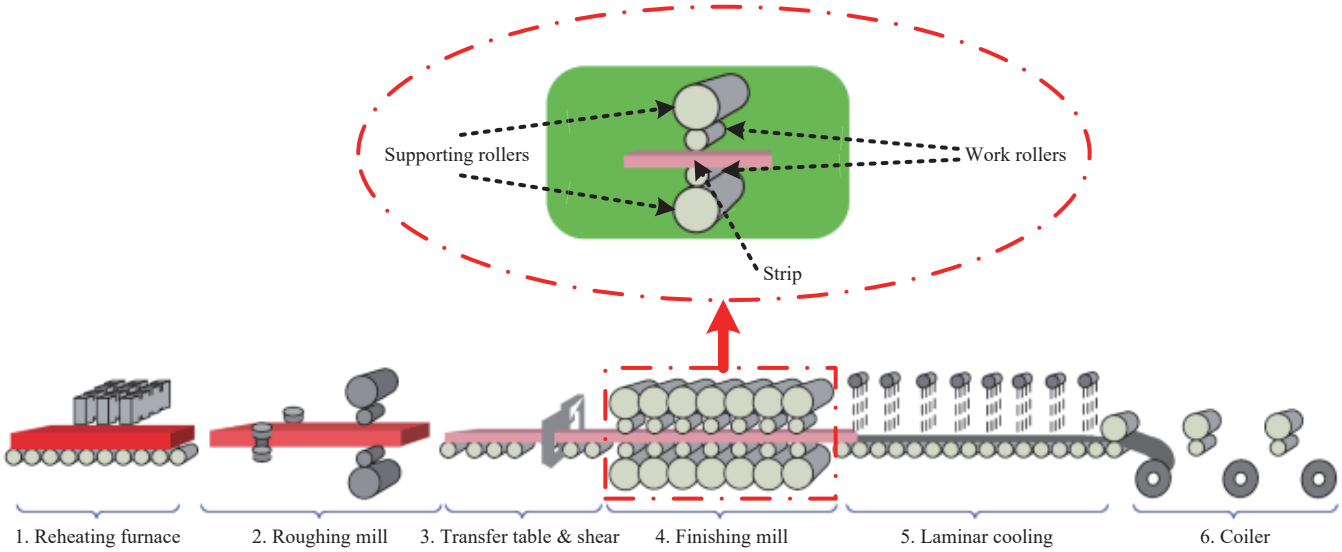


Fig. 5. Schematic of the hot strip rolling process.

TABLE I
DETAILED DESCRIPTIONS OF VARIABLES FOR LSTM1

| Variable | unit | Variable | unit |
|---|---|---|---|
| Strip thickness | mm | Rolling force | kN |
| Strip width | mm | Bending force | kN |
| Temperature | °C | Rolling speed | m/s |
| Crown | mm | Roller gap | mm |
| Flatness | I | Power | kW |

TABLE II
DETAILED DESCRIPTIONS OF VARIABLES FOR LSTM2

| Variable | unit | Variable | unit |
|---|---|---|---|
| Cumulative rolling length | km | Target thickness | mm |
| Weight of the steel | kg | Target width | mm |
| Carbon content | % | | |

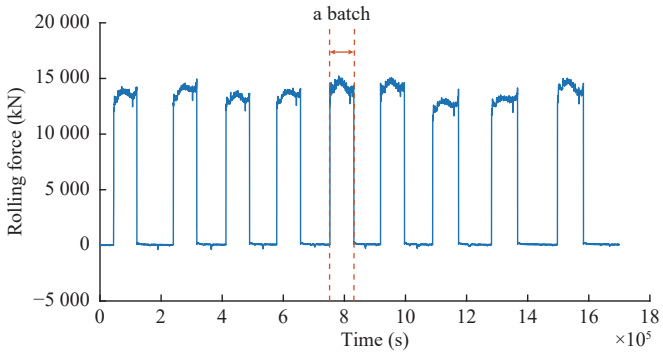


Fig. 6. An example of a monitoring sample of the rolling force.

the monitoring data for LSTM1 should be trimmed to the same length. Since the data lengths have little difference, the shortest batch data could be found out from all modeling data, and then the data of remaining batches were cut to this length. In addition, the value of each variable was normalized using min-max normalization to eliminate the dimensional effect between the variables.

### C. Experimental Setup

In the proposed deep RNN, the network structure parameters strongly depend on the complexity of the problem. Considering the complexity of the research problem and the number of available training samples, the LSTM1 for fine-grained feature extraction was set to have five LSTM layers which contain 128, 200, 256, 360, and 512 hidden nodes, respectively, and the LSTM2 for coarse-grained feature extraction was set to three LSTM layers with 32, 64, and 128 hidden nodes, respectively. The truncated BPTT with the minibatch stochastic gradient descent method was used to update the weights during training, and Adam optimization algorithm was used to optimize the neural network. In the testing phase, once the collected test data set is input into the deep RNN, the network outputs a HI corresponding to the health of the roller as the recognized result. A detailed configuration of this deep RNN is summarized in Table III, and all the parameters were selected based on a large number of trials.

TABLE III
DETAILS OF DEEP RNN

| Setting items | Detail |
|---|---|
| Learning rate adjustment | Adam |
| Number of LSTM layers for LSTM1 | 5 |
| Number of LSTM layers for LSTM2 | 3 |
| Number of hidden nodes for LSTM1 | 128; 200; 256; 360; 512 |
| Number of hidden nodes for LSTM2 | 32; 64; 128 |
| Output dimension of LSTM1 | 4 |
| Output dimension of LSTM2 | 2 |
| Time window of LSTM1 | 6 |
| Time window of LSTM2 | 12 |
| Mini-batch size | 64 |
| Dropout rate | 0.5 |

### D. Results and Discussion

*1) HI Construction Results:* In order to quantitatively evaluate the constructed HI, two metrics of monotonicity (Mon) and correlation (Corr) [34] are considered here. The monotonicity metric is used to evaluate the tendency of an HI, and a higher score means a better performance in monotonicity, which can be calculated by

$$Mon(HI) = \left| \frac{Num\ of\ dHI > 0}{T-1} - \frac{Num\ of\ dHI < 0}{T-1} \right| \quad (18)$$

where $T$ represents the length of the HI during the lifetime; $d/dHI = HI_{T+1} - HI_T$ is the difference of the HI sequence; $Num\ of\ d/dHI > 0$ and $Num\ of\ d/dHI < 0$ are the number of the positive differences and the negative differences, respectively.

The correlation metric is able to reflect the correlation property between HI and operating time, which is denoted as

$$Corr(HI) = \frac{\left| \sum_{t=1}^{T} \left(HI_t - \overline{HI}\right)\left(l_t - \bar{l}\right) \right|}{\sqrt{\sum_{t=1}^{T} \left(HI_t - \overline{HI}\right)^2 \sum_{t=1}^{T} \left(l_t - \bar{l}\right)^2}} \quad (19)$$

where $HI_t$ and $l_t$ denote the HI and the time value at moment $t$; $\overline{HI}$ and $\bar{l}$ are the means of $HI_t$ and $l_t$, respectively.

According to (18) and (19), the Mon and Corr of the proposed approach are obtained and listed in Table IV. To demonstrate the advantages of the proposed HI construction method, an RNN was modeled for comparison using the multilayer LSTM1 for fine-grained feature extraction, which is denoted as RNN1. Another RNN with multilayer LSTM2 for coarse-grained feature extraction was also used to construct a HI for comparison, and marked as RNN2.

TABLE IV
HI CONSTRUCTION RESULTS

| Approach | Mon | Corr |
|---|---|---|
| RNN1 | 0.4342 | 0.9628 |
| RNN2 | 0.3249 | 0.9137 |
| Deep RNN | 0.5973 | 0.9762 |

It can be seen from Table IV that the Mon and Corr values of deep RNN and RNN1 are both larger than RNN2, and the deep RNN achieves the highest values. The large values imply that the HI established by deep RNN has a better performance in monotonicity and correlation. Through the comparison, it is demonstrated that the proposed deep RNN network structure which can extract fine-grained and coarse-grained features has better performance than other RNN-based methods.

*2) RUL Prediction Results:* To verify the effectiveness of the proposed RUL prediction approach based on deep RNN, the two collected test sets, represented as test01 and test02, were used for prediction and analysis. Due to the statistical properties of the PF algorithm, the estimation results are in distribution forms approximated by 5000 samples. The RUL of test01 was estimated on the 161th and 242nd batch, and the estimation and prediction results are shown in Fig. 7. The mean of the HI prediction is indicated by the blue line, and the 95% confidence interval (CI) for HI estimation and prediction are also given. Fig. 7(a) shows the EoL predicted at the 161th batch, which means that the data from the first 161 batches are used to update the model. Thus, the RUL and 95% CI can be derived based on (16) and (17). From Fig. 7(c), it can be seen that the median value and 95% CI of RUL are 37 batches and [31, 47], respectively, which indicates that roller maintenance is required after rolling 37 further batches of steel. As shown in Fig. 7(b), the HI estimated for test01 at the 242nd batch exceeded the FT at the 252nd batch. The RUL prediction results are shown in Fig. 7(d), and the 5th, 50th, and the 95th percentiles of the RUL are 5, 10, and 17, respectively.

The estimation and prediction results for test02 are shown in Fig. 8, which are estimated on the 195th and 263th batch. As shown in Fig. 8(a), the estimated HI at the 195th batch is higher than FT for the first time at batch 223, indicating that the EOL is 223 batches. The PDF of the RUL is shown in Fig. 8(c), the median value and 95% CI of RUL are 28 batches and [22, 35], respectively. As can be seen from Fig. 8(b) that the EoL predicted at 263th batch is 277 batches, and the PDF of the RUL is plotted in Fig. 8(d). The 5th, 50th, and the 95th percentiles of the RUL are 7, 14, and 21, respectively, which indicates that it is necessary to maintain the rollers after rolling 14 more batches of steel.

The comprehensive results of the prediction performance are listed in Table V. It can be seen that the RUL prediction results with long available batch data for both test01 and test02 are more accurate than those with shorter ones. This can be attributed to the limitation of the Bayesian method. If there is no new measurement information to update the model, the prediction accuracy will decrease as the prediction window increases. On the other hand, the uncertainty of the rolling schedule and the uncertainty in the actual rolling process can also lead to the inaccuracy of the initial predictions. However, once new measurement data are acquired, the parameters of state-space model are automatically updated to improve the prediction results. Although the prediction at the beginning is imprecise, the prediction results become more and more accurate as the measured data increases.

In order to further demonstrate the advantages of the proposed deep network architecture, convolutional neural network (CNN) and deep belief network (DBN) are used for comparison. The CNN has five convolution layers, and a fully connected layer is also added at the end so as to have the same architecture as deep RNN. The DBN is composed of four restricted Boltzmann machines, and the basic neural network is attached at the top layer. The approaches are carried out on the two test datasets, and root mean square error (*RMSE*) and mean absolute error (*MAE*) are used to assess the performance, which are defined as

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(RUL_a - RUL_e)^2} \quad (20)$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|RUL_a - RUL_e| \quad (21)$$

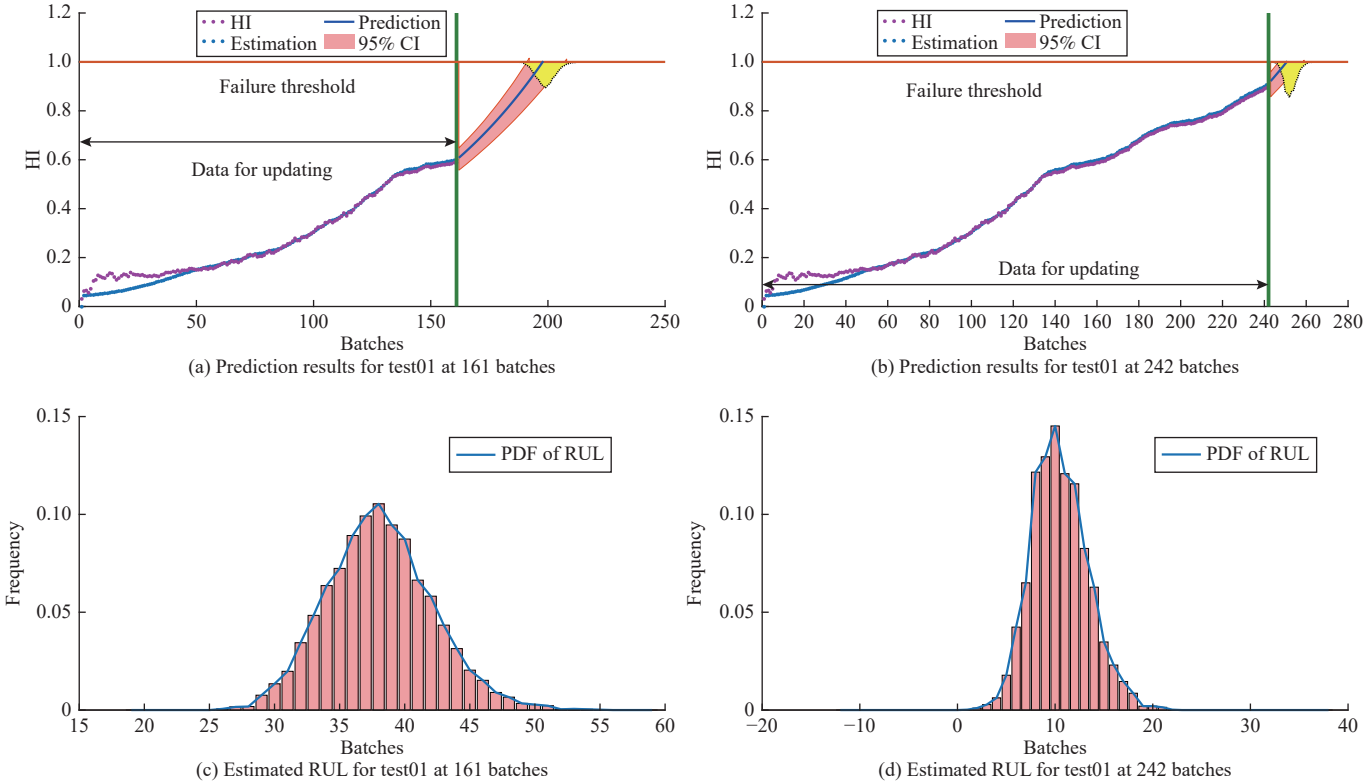


Fig. 7. Predicted results of test01.

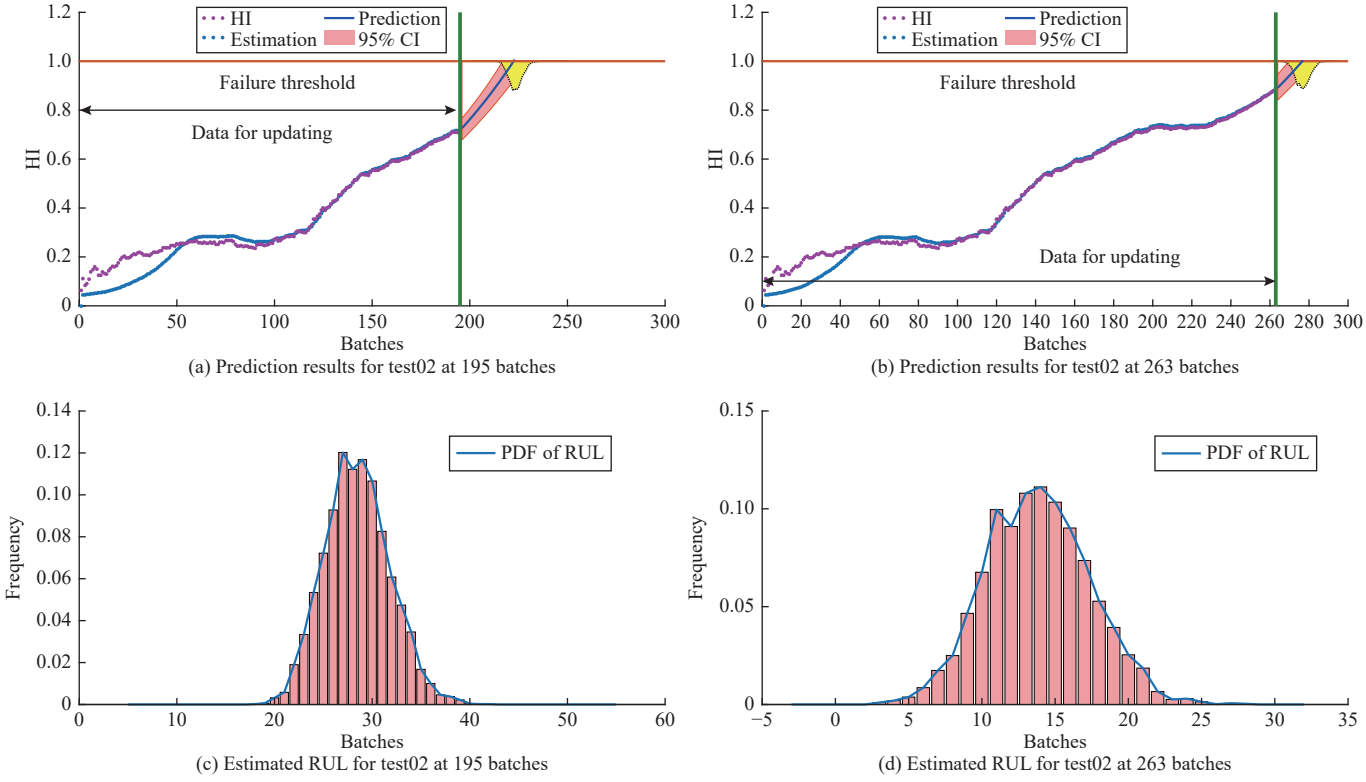


Fig. 8. Predicted results of test02.

where $RUL_a$ is the actual RUL value; and $RUL_e$ is the estimated value.

The comparison results with respect to the RMSE and MAE are summarized in Table VI. RNN2 achieves the highest RMSE value, which indicates its prediction error is the largest. The RMSE values of RNN1, CNN, and DBN are 17.67, 18.97, and 15.23, respectively, which are higher than the deep RNN. Deep RNN achieves relatively low MAE value

TABLE V
RUL PREDICTION RESULTS

| Testing data | Current time | Estimated EOL | Predicted RUL | 95% CI for RUL | Actual EOL | Actual RUL |
|---|---|---|---|---|---|---|
| Test01 | 161 | 198 | 37 | [31, 47] | 256 | 95 |
| | 242 | 252 | 10 | [5, 17] | 256 | 14 |
| Test02 | 195 | 223 | 28 | [22, 35] | 278 | 83 |
| | 263 | 277 | 14 | [7, 21] | 278 | 15 |

TABLE VI
COMPARISON OF THE PREDICTION RESULTS

| Approach | *RMSE* | *MAE* |
|---|---|---|
| RNN1 | 17.67 | 15.42 |
| RNN2 | 31.26 | 26.25 |
| CNN | 18.97 | 15.74 |
| DBN | 16.23 | 14.37 |
| Deep RNN | 11.05 | 9.85 |

as 9.85, while the MAE values of RNN1, RNN2, CNN, and DBN are 15.42, 26.25, 15.74, and 14.37, respectively. From the results, it is observed that the deep RNN has the smallest prediction error. This demonstrates that the deep RNN network structure proposed in this study is able to extract coarse-grained and fine-grained features to construct the HI, and can further accurately predict the RUL for rollers of HSM. Furthermore, the prediction result based on the proposed RUL framework include a time range of possible failures, which can provide more useful information for roller management decision.

## V. CONCLUSION

Rollers plays a significant role in the hot rolling process. With their wide and frequent application in steel production, the RUL prediction of rollers has become a crucial and challenging issue. To solve this problem, a deep learning architecture based on RNN was developed to estimate the RUL in this work. Firstly, a deep RNN model was established to extract the coarse-grained and the fine-grained features to construct the HI. Afterwards, the PF algorithm was introduced to iteratively update the parameters of a constructed state-space model, and the PDF of RUL could be estimated by extrapolating the dynamic model to the FT. In the end, the proposed methods were applied to the HSM to predict the RUL of the rollers. The proposed RUL prediction framework is helpful for the management of the roller, since it can effectively reduce the production cost and improve the operating rate of the HSM so as to improve the production efficiency of the business.

The following conclusions can be drawn from the experimental results. First, the proposed deep RNN network structure is able to extract coarse-grained and fine-grained features to create the HI. Second, the developed comprehensive HI is able to reflect the health of the roller, and it gradually increases with the deterioration of health state. Third, compared with some other popular deep learning methods, the proposed RUL prediction framework is able to accurately predict the RUL for rollers of HSM since it obtained relatively low RMSE and MAE values.

Despite the good experimental results achieved by the proposed approach, it still needs further optimization in the future. On one hand, the parameters in the deep network structure can be further optimized through some optimization algorithms, such as [35] and [36]. On the other hand, some efforts should be made to improve the PF so as to obtain higher prediction accuracy.

## REFERENCES

[1] A. Azadeh, S. M. Asadzadeh, N. Salehi, and M. Firoozi, "Condition-based maintenance effectiveness for series-parallel power generation system-a combined markovian simulation model," *Reliab. Eng. Syst. Saf.*, vol. 142, pp. 357–368, Oct. 2015.

[2] T. Sutharssan, S. Stoyanov, C. Bailey, and C. Yin, "Prognostic and health management for engineering systems: A review of the data-driven approach and algorithms," *J. Eng.*, vol. 2015, pp. 215–222, Jul. 2015.

[3] A. K. S. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mech. Syst. Signal Proc.*, vol. 20, no. 7, pp. 1483–1510, Oct. 2006.

[4] M. Pecht and R. Jaai, "A prognostics and health management for information and electronics-rich systems," *Microelectron. Reliab.*, vol. 50, no. 3, pp. 317–323, Mar. 2010.

[5] H. Meng and Y. F. Li, "A review on prognostics and health management (PHM) methods of lithium-ion batteries," *Renew. Sust. Energ. Rev.*, vol. 116, Article No. 109405, Dec. 2019.

[6] P. Paris and F. Erdogan, "A critical analysis of crack propagation laws," *J. Basic Eng.*, vol. 85, no. 4, pp. 528–533, Dec. 1963.

[7] G. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation," *J. Power Sources*, vol. 134, pp. 277–292, 2004.

[8] Z. Liu, G. Sun, S. Bu, J. Han, X. Tang, and M. Petch, "Particle learning framework for estimating the remaining useful life of lithium-ion batteries," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 2, pp. 280–293, Feb. 2017.

[9] B. Long, W. Xian, L. Jiang, and Z. Liu, "An improved autoregressive model by particle swarm optimization for prognostics of lithium-ion batteries," *Microelectron. Reliab.*, vol. 53, no. 6, pp. 821–831, Jun. 2013.

[10] J. Hu and P. Chen, "Predictive maintenance of systems subject to hard failure based on proportional hazards model," *Reliab. Eng. Syst. Saf.*, vol. 196, Article No. 106707, Apr. 2020.

[11] X. Si, W. Wang, C. Hu, and D. Zhou, "Estimating remaining useful life with three-source variability in degradation modeling," *IEEE Trans. Reliab.*, vol. 63, no. 1, pp. 167–190, Mar. 2014.

[12] R. Khelif, B. Chebel-Morello, S. Malinowski, E. Laajili, F. Fnaiech, and N. Zerhouni, "Direct remaining useful life estimation based on support vector regression," *IEEE Trans. Ind. Electron.*, vol. 64, no. 3, pp. 2276–2285, Mar. 2017.

[13] Z. Zhao, B. Liang, X. Wang, and W. Lu, "Remaining useful life prediction of aircraft engine based on degradation pattern learning," *Reliab. Eng. Syst. Saf.*, vol. 164, pp. 74–83, Aug. 2017.

[14] L. Liao and F. Kottig, "Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an

application to battery life prediction," *IEEE Trans. Reliab.*, vol. 63, no. 1, pp. 191–207, Mar. 2014.

[15] J. Wei, G. Dong, and Z. Chen, "Remaining useful life prediction and state of health diagnosis for lithium-ion batteries using particle filter and support vector regression," *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5634–5643, Jul. 2018.

[16] X. Si, T. Li, Q. Zhang, and C. Hu, "Prognostics for linear stochastic degrading systems with survival measurements," *IEEE Trans. Ind. Electron.*, vol. 67, no. 4, pp. 3202–3215, Apr. 2020.

[17] W. Liu, Z. Wang, X. Liu, N. Zeng, Y. liu, and F. E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, no. 19, pp. 11–26, Apr. 2017.

[18] L. Liao, W. Jin, and R. Pavel, "Enhanced restricted boltzmann machine with prognosabillity regularization for prognostics and health assessment," *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 7076–7083, Nov. 2016.

[19] X. Li, Q. Ding, and J. Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliab. Eng. Syst. Saf.*, vol. 172, pp. 1–11, Apr. 2018.

[20] Y. Zhang, R. Xiong, H. He, and M. G. Petch, "Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 5695–5705, Jul. 2018.

[21] F. O. Heimes, "Recurrent neural networks for remaining useful life estimation," in *Proc. Int. Conf. Prognostics and Health Management*, pp. 1–6.

[22] L. Guo, N. Li, F. Jia, Y. lei, and J. Lin, "A recurrent neural network based health indicator for remaining useful life prediction of bearings," *Neurocomputing*, vol. 240, pp. 98–109, May 2017.

[23] J. A. Schey, "Tribology in metalworking: Friction, lubrication, and wear," *J. Applied Metalworking*, vol. 3, no. 2, pp. 173–173, Jan. 1984.

[24] S. Puchhala, M. Franzke, and G. Hirt, "Interaction effects between strip and work roll during flat rolling process," in *Process Machine Interactions*, 2013, pp. 439–458.

[25] S. John, S. Sikdar, A. Mukhopadhyay, and A. Pandit, "Roll wear prediction model for finishing stands of hot strip mill," *Ironmak. Steelmak.*, vol. 33, no. 2, pp. 169–175, Jul. 2013.

[26] S. Rath, A. P. Singh, U. Bhaskar, B. Krishna, B. K. Santra, D. Rai, and N. Neogi, "Artificial neural network modeling for prediction of roll force during plate rolling process," *Mater. Manuf. Process.*, vol. 25, no. 1-3, pp. 149–153, Mar. 2010.

[27] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. Int. Conf. Machine Learning*, 2013, pp. 1310–1318.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[29] K. Peng, K. Zhang, B. You, J. Dong, and Z. Wang, "A quality-based nonlinear fault diagnosis framework focusing on industrial multimode batch processes," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2615–2624, Jan. 2016.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent*, 2015, pp. 1–15.

[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014.

[32] Z. Liu, G. Sun, S. Bu, J. Han, X. Tang, and M. Pecht, "Particle learning framework for estimating the remaining useful life of lithium-ion batteries," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 2, pp. 280–293, Nov. 2016.

[33] L. Ma, J. Dong, K. Peng, and C. Zhang, "Hierarchical monitoring and root cause diagnosis framework for key performance indicator-related multiple faults in process industries," *IEEE Trans. Ind. Informat.*, vol. 15, no. 4, pp. 2091–2100, Apr. 2015.

[34] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to RUL prediction," *Mech. Syst. Signal Proc.*, vol. 104, pp. 799–834, May 2018.

[35] W. Liu, Z. Wang, Y. Yuan, N. Zeng, K. Hone, and X. Liu, "A novel sigmoid-function-based adaptive weighted particle swarm optimizer," *IEEE T. Cybern.*, Jul. 2019. DOI: 10.1109/TCYB.2019.2925015.

[36] I. U. Rahman, Z. Wang, W. Liu, B. Ye, M. Zakarya, and X. Liu, "An N-state markovian jumping particle swarm optimization algorithm," *IEEE Trans. Systems*, *Man*, *and Cybernetics*: *Systems*, Jan. 2020. DOI: 10.1109/TSMC.2019.2958550.

**Ruihua Jiao** received the B.E. and Ph.D. degrees in automation from University of Science and Technology Beijing in 2012 and 2020, respectively. His research interests include fault diagnosis, prognostics and health management.

**Kaixiang Peng** (M'15) received the B.E. degree in automation, and M.E. and Ph.D. degrees from the Research Institute of Automatic Control, University of Science and Technology in 1995, 2002, and 2007, respectively. He is a Professor in the School of Automation and Electrical Engineering, University of Science and Technology Beijing. His research interests include fault diagnosis, prognosis, and maintenance of complex industrial processes, modeling and control for complex industrial processes, and control system design for the rolling process.

**Jie Dong** received the B.E., M.E., and Ph.D. degrees from University of Science and Technology Beijing in 1995, 1997, and 2007, respectively. She is currently a Professor of the School of Automation and Electrical Engineering, University of Science and Technology Beijing. From July to December in 2004, she visited University of Manchester as a visiting scholar. Her research interests include intelligent control theory and application, process monitoring and fault diagnosis, complex system modeling and control.