IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 18, 2019, accepted October 23, 2019, date of publication November 4, 2019, date of current version November 14, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2951197

An Improved PF Remaining Useful Life Prediction Method Based on Quantum Genetics and LSTM

YANG GE[1,2,3], LINING SUN[2], AND JIAXIN MA[1,3]

[1]School of Mechanical Engineering, Changshu Institute of Technology, Changshu 215500, China
[2]School of Mechanical and Electric Engineering, Soochow University, Suzhou 215006, China
[3]Jiangsu Key Laboratory for Elevator Intelligent Safety, Changshu 215500, China

Corresponding author: Yang Ge (geyang@cslg.edu.cn)

ABSTRACT Remaining useful life (RUL) is the premise and basis of the equipment health management plan. As accurate as possible life prediction is of great significance to reliability and economy of equipment maintenance. In this paper, a data-driven improved particle filter (PF) RUL prediction method is proposed. A health indicator extraction method based on multi-feature fusion is introduced for the RUL prediction, which can visually show the degradation trend of the healthy state of the equipment. The degradation model and observation model of equipment health indicators are established, and the PF algorithm is used to track parameters of the model. A quantum genetic algorithm is employed to improve the problem of particle degradation in PF. On the basis of filter tracking, long short term memory (LSTM) network is used to predict the trend of model coefficients, which further improves the accuracy of RUL prediction. The experiment using the C-MAPSS data set shows the proposed method has a better prediction accuracy than other methods.

INDEX TERMS Remaining useful life, particle filter, quantum genetic algorithm, long short term memory.

I. INTRODUCTION

It is a key link for prognostics and health management to predict remaining useful life (RUL). The life prediction method based on data-driven can real-time reflect the health status of equipment. With the development of dynamic tracking and deep learning algorithms, it has become a hotspot in the research of condition-based maintenance (CBM). The first step of equipment life prediction often requires the extraction of indicators that can reflect the health of the equipment.

The first step to predicting the RUL of equipment is extracting features which can reflect the health of the equipment. The common feature types are time-domain feature, frequency-domain feature and time-frequency domain feature [1]. For example, wavelet packet [2] and root mean square [3] are commonly used features. Considering that single or few features extracted from sensor data may lose effective information, many researchers proposed multi-feature fusion methods. A health-weighted feature is constructed in literature [4] and [5], which integrates the mutual information of various features and has a good correlation with the degradation process of machinery. In addition, image feature extraction is also introduced into RUL prediction. A new fault diagnosis method for planetary gear based on image feature extraction and bag-of-words model is proposed in the literature [6]. Mishra and Huhtala [7] realized the fault detection accuracy of more than 90% of the elevator by extracting the depth features of sensor signals.

Based on feature extraction, the next step of RUL prediction is to track the changing trend of equipment health status. The main methods are as follows. (1) Mathematical fitting model. Jahani et al. [8] propose a B-spline based modeling method for non-parametric degenerate signals. Wang *et al.* [9] propose an improved RUL prediction wiener process model, both drift and diffusion parameters of the model can adapt to the updating of monitoring data. Bayesian methods are also often used to predict RUL. A two-stage degradation model is proposed in the literature [10], and the Bayesian method is used to estimate the model parameters. In addition to these, Cox proportional hazard model [11], nonlinear Wiener process model [12] and fractional Brownian motion model [13] have also been introduced into RUL prediction. (2) Dynamic tracking filter. Particle filter (PF) has been used by many researchers as a method of RUL prediction, which greatly improves the accuracy of life prediction. A genetic PF method is proposed for predicting the life of lithium-ion batteries [14]. Lei *et al.* [4] use the PF algorithm to predict the rolling bearing RUL. Because of the particle degradation

The associate editor coordinating the review of this manuscript and approving it for publication was Alberto Cano.

in the PF algorithm, many improved methods are proposed. An improved particle filter algorithm is proposed in the literature [15]. An improved method of unscented particle filtering (UPF) based on Markov chain Monte Carlo (MCMC) is proposed in literature [16]. Cui *et al.* [17] and Son *et al.* [18] propose a new switching noiseless Kalman filter algorithm respectively. A Wiener-Process-Model-Based Method is proposed for RUL Prediction Considering Unit-to-Unit Variability, and PF is used to update the model parameters [19]. An RUL prediction method based on the exponential model and PF is proposed to overcome the problem of nonlinear and non-gaussian characteristics in lithium-ion batteries capacity degradation [20]. Aiming at the nonlinear and non-gaussian characteristics of the system, a combination method of unscented Kalman filter and PF is proposed in the literature [21]. (3) Machine learning method. Machine learning is a method proposed in the field of speech recognition and image recognition in recent years. Due to its excellent self-learning function, it has also been introduced into the RUL prediction by many researchers, one of the most is the deep learning methods. A double convolutional neural network architecture is presented to predicted RUL in literature [22]. This method does not need any feature extractor, only needs to input the original vibration signal, and can predict RUL with high accuracy. A method for rapidly evaluating the reliability and predicting remaining useful life using a two-dimensional convolutional neural network with signal conversion is proposed in the literature [23]. An automatic two-stage estimation method of bearing robustness using deep neural networks (DNNs) is proposed in the literature [1]. A long short-term memory (LSTM) structure is proposed for predicting the robustness of short sequence monitoring with random initial wear [24]. A data-driven prediction method based on Elman neural network is proposed by Yang *et al.* [25]. A method that uses deep learning tools and curve matching technology is proposed to estimate the robustness of the system [26]. A framework for estimating the RUL of mechanical systems is proposed, which is composed of the multi-layer perceptron and multilayer perceptron and evolutionary algorithm for optimizing parameters [27]. Besides, there are many other machine learning algorithms, such as neural networks [28]–[30], capsule neural networks [31], dynamic Bayesian networks [32] and so on.

Although the PF algorithm has a good trend tracking advantage, particle degradation is an inevitable phenomenon in the PF algorithm. Resampling attenuates the degradation of the particle, but the degradation still exists, and resampling also presents the problem of particle depletion and the operation of the restriction algorithm. Although there are many improved PF algorithms, they are basically at the cost of running time of the algorithm. A deep learning method can automatically extract effective information in signals, but in many cases, the signals contain a lot of noise. To achieve the ideal prediction effect, the number of layers of learning networks needs to be increased and a lot of computing time needs to be consumed. Therefore, in this paper, an improved PF algorithm based on quantum genetic algorithm is proposed to track the trend of equipment degradation, and the LSTM algorithm is employed into RUL prediction to improve the efficiency and accuracy of RUL prediction. The innovation of this paper is as follows. (1) An improved PF algorithm based on a quantum genetic algorithm is proposed, which shortens the running time of the algorithm and improves the tracking accuracy. (2) LSTM algorithm is combined with the proposed improved PF algorithm employed for RUL prediction. Compared with using a filter algorithm or LSTM algorithm alone to predict RUL, the prediction accuracy of the proposed method is higher. The rest of this paper is arranged as follows. Section 2 presents the methodology used in this paper, including the PF algorithm, quantum genetic algorithm and improved PF algorithm based on quantum genetic algorithm. Section 3 includes the experimental verification and analysis. The proposed is applied to the C-MAPSS data set for units RUL prediction. Section 4 is the conclusion of this paper.

II. METHODOLOGY

A. PF MODELS

PF is widely used in the field of visual tracking, signal processing, robotics, image processing, financial economy, as well as target positioning navigation, tracking, and other fields. In this paper, the PF model is applied to the RUL prediction of aeroengine. Let the state equation of the system be shown in equation (1).

$$\mathrm{X}_k = f\left(X_{k-1}, W_k\right) \tag{1}$$

where X_k denotes the state of the system at time k, $\mathrm{f}(\cdot)$ denotes the mapping function, W_k denotes system process noise. Assume that W_k obeys a Gaussian distribution with a mean of 0 and a variance of Q, i.e. $W_k \sim N\left(0, Q\right)$.

Let the state observation equation of the system be shown in equation (2).

$$\mathrm{Z}_k = h\left(X_k, V_k\right) \tag{2}$$

where Z_k denotes the measurement results of system state features at time k, $\mathrm{h}(\cdot)$ denotes the mapping function, and V_k denotes the measurement noise.

The specific steps of PF algorithm are as follows:

(1) Initialization, $\mathrm{k} = 0$, according to the prior density $\mathrm{p}\left(x_0\right)$ of the system state, collect particle sets $\left\{x_0^i, \omega_0^i\right\}$, $i = 1, 2, \ldots, \mathrm{N}$.

(2) Importance sampling. for $i = 1, 2, \cdots, N$, Cextract N particles from the proposal distribution $\mathrm{x}_k^i \sim q\left(x_k \left| x_{k-1}^i, z_{1:k}\right.\right)$.

(3) Update weights.

$$\omega_k^i = \omega_k^i \frac{p\left(z_k \left| \hat{x}_k^i\right.\right) p\left(\hat{x}_k^i \left| \hat{x}_{k-1}^i\right.\right)}{p\left(\hat{x}_k^i \left| x_{k-1}^i, z_{1:k}\right.\right)} \tag{3}$$

(4) Normalization of important weights.

$$\tilde{\omega}_k^i = \omega_k^i \Big/ \sum_{i=1}^{N} \omega_k^i \tag{4}$$

(5) Resampling. The new particles were resampled based on the weight of importance, with a mean weight of $1/N$.

(6) State estimation. The estimated state is obtained by the weighted sum of weights and extracted particles.

$$\hat{x}_k = \sum_{i=1}^{N} \tilde{\omega}_k^i \hat{x}_k^i \tag{5}$$

Particle degradation is an inevitable phenomenon in the PF algorithm. Resampling attenuates particle degradation, but the degradation still exists, and resampling also presents the problem of particle depletion and the parallel operation of the restriction algorithm.

B. QUANTUM GENETIC ALGORITHM

Quantum genetic algorithm (QGA) is an algorithm combining the concepts of quantum computing with the genetic algorithm, which can maintain good population diversity. It applies the probability amplitude representation of quantum bits to the coding of chromosomes so that one chromosome can express the superposition of multiple states and realize the chromosome update operation by using the quantum revolving gate and quantum non-gate, to realize the optimization of population.

The population of QGA consists of quantum chromosomes encoded in quantum bits. Quantum bit is the smallest information unit in QGA. Different from the classical bit, it can not only be in the state 0 or 1, but also represent any superposition state of the two. Therefore, QGA has a lot of diversity compared with GA. For a population containing n individuals, the length of a quantum chromosome m is expressed as

$$\mathrm{P}(t) = \left\{p_1^t, p_2^t, \cdots, p_n^t\right\} \tag{6}$$

$$\mathrm{p}_j^t = \begin{vmatrix} \alpha_1^t \\ \beta_1^t \end{vmatrix} \begin{vmatrix} \alpha_2^t \\ \beta_2^t \end{vmatrix} \cdots \begin{vmatrix} \alpha_m^t \\ \beta_m^t \end{vmatrix} \quad (j = 1, 2, \ldots, n) \tag{7}$$

where p_j^t is an individual in generation t.α_i and β_i are both complex numbers, called probability amplitudes, denoting the probability amplitudes of state 0 and state 1 respectively, and satisfying the normalization condition that $\alpha^2 + \beta^2 = 1$. And t is the number of generations.

Quantum gate is the executor of the ultimate evolutionary operation in QGA. One of the keys of QGA is to construct a suitable quantum gate. The update of the quantum bit is realized through the quantum rotation gate. The specific formula is as follow:

$$\begin{bmatrix} \alpha_i' \\ \beta_i' \end{bmatrix} = U_i \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \tag{8}$$

where $U_i = \begin{bmatrix} cos\theta_i & -sin\theta_i \\ sin\theta_i & cos\theta_i \end{bmatrix}$ is the quantum rotation gate, $\begin{bmatrix} \alpha_i' \\ \beta_i' \end{bmatrix}$ is the i-th quantum bit in the updated chromosome, $\begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix}$ is the i-th quantum bit in the chromosome before the update, and θ_i is the rotation angle of the quantum gate.

In order to reduce the particle degradation of the PF algorithm, QGA is introduced into PF in this paper. Each particle produced by PF is regarded as a chromosome, and the sample set is optimized by QGA, and the sample set $\left\{\left(x_k^i, \omega_k^i\right), i = 1, 2, \ldots, N\right\}$ consisting of the best samples with weights can be obtained. At last, an estimated value of the state can be obtained using $\mathrm{E}(X_k) = \sum_{i=1}^{N} \omega_k^i x_k^i$.

C. IMPROVED PF BASED ON QGA

(1) Initialize the population. Regard each particle in the sample set importance sampled by PF as an individual, code them into chromosomes by a quantum code with length m, and they form the initial population $\mathrm{Q}(t)$.

(2) Construct $\mathrm{P}(t)$. According to the probability amplitude $\left|\alpha_i^T\right|^2$ and $\left|\beta_i^T\right|^2$ (i = 1, 2, . . . , m) of each individual in $\mathrm{Q}(t)$, generate binary chromosome population set $\mathrm{P}(t) = \left\{b_1^T, b_2^T, \cdots, b_N^T\right\}$. b_k^T is generated as: Randomly generate a number θ between [0, 1], if $\theta > \left|\alpha_i^T\right|^2$, $b_k^T = 1$, otherwise, $b_k^T = 0$.

(3) Evaluate the quality of the particles using fitness function, and retain the optimal individual in the generation. The fitness function is as:

$$\mathrm{f}(x_i) = \frac{1}{N}\left(\sum_{i=1}^{N} x_i^2 - N\bar{x}^2\right) \tag{9}$$

where x_i is the i-th individual of the population, $\bar{x}$ is the mean value of all the individuals in the population, and N is the number of individuals in the population.

(4) Measure each individual in the population $\mathrm{Q}(t)$, and calculate the fitness value of $\mathrm{Q}(t)$.

(5) update the population $\mathrm{Q}(t)$ to the offspring population $\mathrm{Q}(t+1)$ using the quantum rotation gate.

(6) Record the best individual and its fitness value.

(7) If meet the end condition, stop the optimization of the population, otherwise, jump to step (4) to continue the optimization of the population.

D. LSTM PREDICTION METHOD

LSTM is a special recurrent neural network (RNN). For a given sequence $x = (x_1, x_2, \ldots, x_n)$, a prediction sequence $y = (y_1, y_2, \ldots, y_n)$ can be iterative calculated by equation (10) and (11).

$$h_t = f\left(W_{xh}x_t + W_{hh}h_{t-1} + b_h\right) \tag{10}$$

$$y_t = W_{hy}h_t + b_y \tag{11}$$

where $h = (h_1, h_2, \ldots, h_n)$ is the hidden layer sequence, W_{xh} denotes the weight coefficient matrix from the input layer to the hidden layer, W_{hy} denotes the weight coefficient matrix from the hidden layer to the output layer, b_h denotes the offset vector of the hidden layer, b_y denotes the bias vector of the output layer, f denotes the activation function, which is generally non-linear, such as tanh or ReLU function.

The simple RNN is equivalent to the multilayer DNN expanded on the time series. This model is easy to appear gradient disappear or gradient explosion problem [33]. LSTM model can learn long-term dependence information while avoiding the gradient disappearance problem [34]. LSTM

TABLE 1. Information about C-MAPSS data sets.

Data set	C-MAPSS			
	FD001	FD002	FD003	FD004
Unit number of training set	100	260	100	249
Unit number of test set	100	259	100	248

adds a memory unit in the neural node of the RNN hidden layer to record historical information, and three gates (input, forget and output) are added to control the use of historical information.

Let i, f, c, o denote input gate, forget gate, unit state and output gate respectively. W is the corresponding weight coefficient matrix, and b is the offset vector. σ and tanh denote sigmoid and hyperbolic tangent activation function respectively. The forward calculation formula is as follows:

$$i_t = sigmoid\left(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i\right) \quad (12)$$

$$f_t = sigmoid\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right) \quad (13)$$

$$c_t = f_t c_{t-1} + i_t tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right) \quad (14)$$

$$o_t = sigmoid\left(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o\right) \quad (15)$$

$$h_t = o_t tanh\left(c_t\right) \quad (16)$$

A detailed introduction of LSTM can be found in reference [35].

III. EXPERIMENT AND ANALYSIS

A. EXPERIMENTAL DATA INTRODUCTION

C-MAPSS data set is used to verify the RUL prediction method proposed in this paper. C-MAPSS data set is the data generated by the simulation model developed by NASA based on MATLAB/Simulink. The data set contains 4 sub-datasets, composed of time series variables obtained from 21 sensors. Each sub-data set contains the training data set and the test data set. The training data set contains the life-span data of each engine unit running to failure. It should be noted that the initial state of each unit is different but can be considered a healthy state. The working conditions and failure modes of each sub-data set are different. The 4 sub-datasets are respectively denoted as FD001, FD002, FD003, and FD004. The specific information is shown in table 1.

In this paper, the training set of FD001 is the verification object of the proposed method. The data set consists of information collected by 21 sensors. Considering that the information of some sensors has almost no change during the whole life of the engine, which is of no value to the prediction of engine life. After comparison, 14 sensors data are selected from 21 sensors as the features of the engine. The numbers of the 14 sensors are [2, 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, 21].

B. FEATURE FUSION

To describe the performance degradation trend of the aero-engines, the establishment method of health indicator (HI) is adopted from the literature [36]. The features vector of unit I at time k is denoted as $\mathrm{X}_k^i = \left(x_{k,1}^i, x_{k,2}^i, \cdots, x_{k,14}^i\right)$. The HI of unit i at time k can be calculated using a linear weighted model, as shown in equation (17).

$$\mathrm{HI}_k^i\left(\mathrm{X}_k^i, \lambda\right) = \lambda_0 + \sum_{n=1}^{14} \lambda_n x_{k,n}^i \quad (17)$$

where $\lambda = (\lambda_0, \lambda_1, \ldots, \lambda_{14})$ denotes the weight vector. To make the HI have a significant degeneration trend, the value of HI is defined as shown in equation (18).

$$\hat{HI}_k^i\left(\mathrm{X}_k^i, \lambda\right) = \begin{cases} 0, & k \le T_i 5\% \\ \exp\left(\dfrac{\log(0.05)}{0.95T_i}(T_i - k)\right), & T_i 5\% < k < T_i 95\% \\ 1, & k \ge T_i 95\% \end{cases} \quad (18)$$

where T_i is the lifetime of unit i. It can be seen that the range of HI value of unit during the whole life is [0,1]. When the unit is new, HI = 0. As the unit's service time increases, the HI value increases, until the unit fails, HI = 1.

From the 100 units, select the first 40 units as training samples and the remaining 60 units as test samples. The features and collection time of each training sample are substituted into equation (17) and (18). An equation set about λ is formed by combining 40 training samples, and the value of λ can be calculated using the least square method. The HI of each test sample can be calculated by substituting the λ value and the collected data of 60 test samples into equation (17). The HI values of the 60 training units are shown in Figure 1.

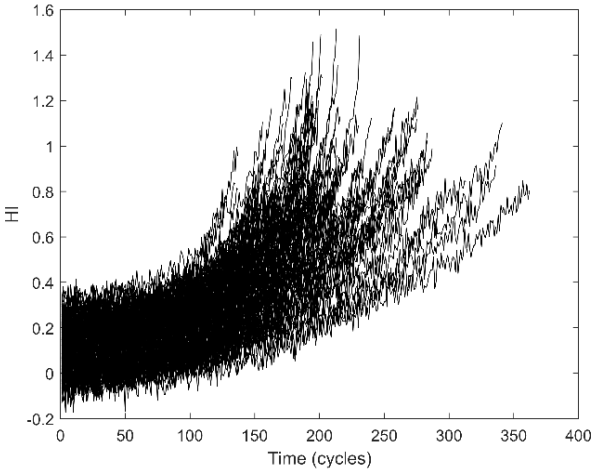

FIGURE 1. HI values of the 60 training units.

C. OBSERVATIONAL EQUATION

In this paper, the exponential distribution is selected as the distribution model of the HI indicator, as shown in equation (19).

$$HI = a * exp\left(b * n\right) + c * exp\left(d * n\right) \quad (19)$$

where n is the number of cycles. HI, a, b, c, d contain Gaussian white noise, the mean is 0, and the variance is unknown. The state of the prediction model is denoted as equation (20).

$$\mathrm{X}\left(n\right) = \left[a\left(n\right), b\left(n\right), c\left(n\right), d\left(n\right)\right]^T \quad (20)$$

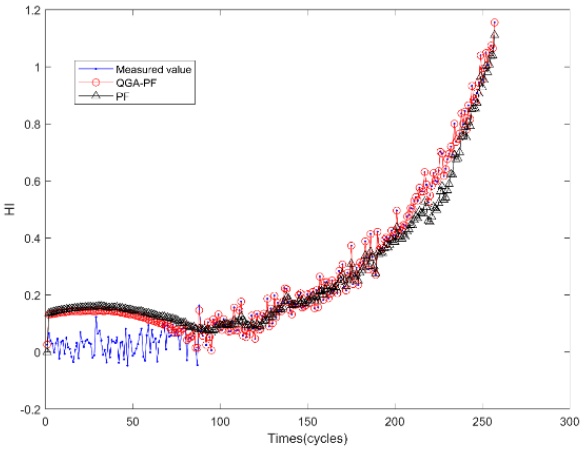

FIGURE 2. The tracking results of the 54th unit.

The state update equation can be expressed as equation (21).

$$\begin{cases} a(n+1) = a(n) + w_a(n), & w_a \sim N(0, \sigma_a) \\ b(n+1) = b(n) + w_b(n), & w_b \sim N(0, \sigma_b) \\ c(n+1) = c(n) + w_c(n), & w_c \sim N(0, \sigma_c) \\ d(n+1) = d(n) + w_d(n), & w_d \sim N(0, \sigma_d) \end{cases} \tag{21}$$

The observation equation is shown in equation (22).

$$\mathrm{HI}(n) = \mathrm{a}(n) * \exp(b(n) * n) + c(n) * exp(d(n) * n) + v(n) \tag{22}$$

where $v(n)$ is the measured noise, which is the gaussian white noise whose mean is 0 and variance is σ_v. i.e. $v(n) \sim N(0, \sigma_v)$.

D. HI TRACKING

In this paper, the 54th unit is selected to test the tracking effect of the PF algorithm and the proposed method, and the results are shown in figure 2. Before the state tracking, 40 training samples are substituted into equation (19) for numerical fitting, and 40 groups of a, b, c, d can be obtained, and the mean values of a, b, c, d are calculated. To improve the tracking effect, the mean values of a, b, c, d are set as the initial values to track the unit state indicators.

From figure 2, at the beginning of tracking, PF and QGA-PF both have large tracking errors. This is because of the initialization problem of a, b, c, d. The error of QGA-PF is smaller relatively. The root mean square (RMSE) of error between the tracing result and the measured value is adopted to specifically compare the tracking accuracy of the two methods, and the computational formula is shown as equation (23).

$$\mathrm{RMSE} = \sqrt{\frac{1}{T}\sum_{i=1}^{T}\left(\hat{HI}_i - HI_i\right)^2} \tag{23}$$

The experimental hardware is a desktop computer (Intel Core i7-8700 processor, 16G memory), and the experimental software environment is MATLAB 2018b. The particle number of PF is 100, the maximum iteration algebra of QGA is 100, and the length of quantum code is 10. The total running time and RMSE of the two methods are shown in table 2.

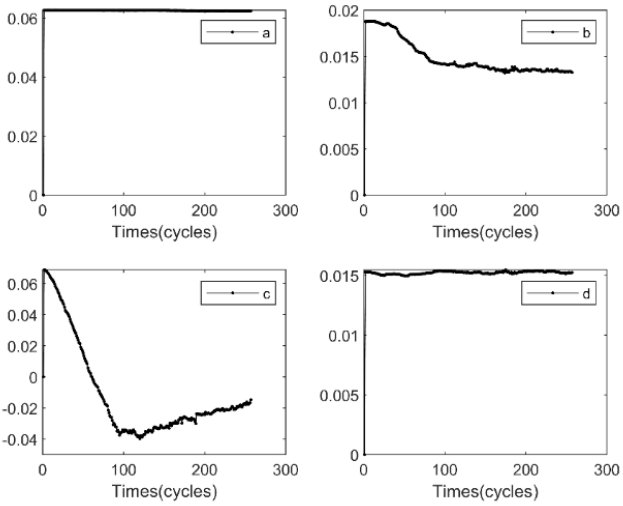

FIGURE 3. The trend of model coefficients.

TABLE 2. The tracking results of PF and QGA-PF.

Methods	Running time (s)	RMSE
PF	909.68	0.07
QGA-PF	612.32	0.0594

It can be seen that the running speed and tracking error of QGA-PF are both better than PF.

E. RUL PREDICTION

The classic RUL prediction method is to substitute the last tracked a, b, c, d values into equation (19) to predict the trend of unit HI. When HI reaches the predetermined threshold, the elapsed time is RUL. The dynamic tracking result curve of a, b, c, d in the state tracking process of unit 54 is shown in figure 3.

As can be seen from figure 3, a, b, c, d are almost changing over time. If only the values of a, b, c, d at the last time point are used to predict the RUL of the unit, there may be a larger error. Therefore, in this paper, LSTM is employed to firstly predict the a, b, c, d values of future time, and then the predicted values of a, b, c, d are substituted into equation (19) to predict the RUL of the unit. The specific prediction process is shown in figure 4.

The specific steps are as follows:

(1) Input the training sample set and use equation (19) for data fitting to obtain the coefficients of each sample, i.e. the values of a, b, c, d.

(2) Input test samples, take the mean values of a, b, c, d obtained in step (1) as initial values, and use QGA-PF to track the HI of the unit and get the sequences of coefficients.

(3) Input the coefficient sequences obtained into LSTM for prediction.

(4) Input the predicted coefficients into equation (19) to calculate the HI of the unit. Judge whether the HI has reached

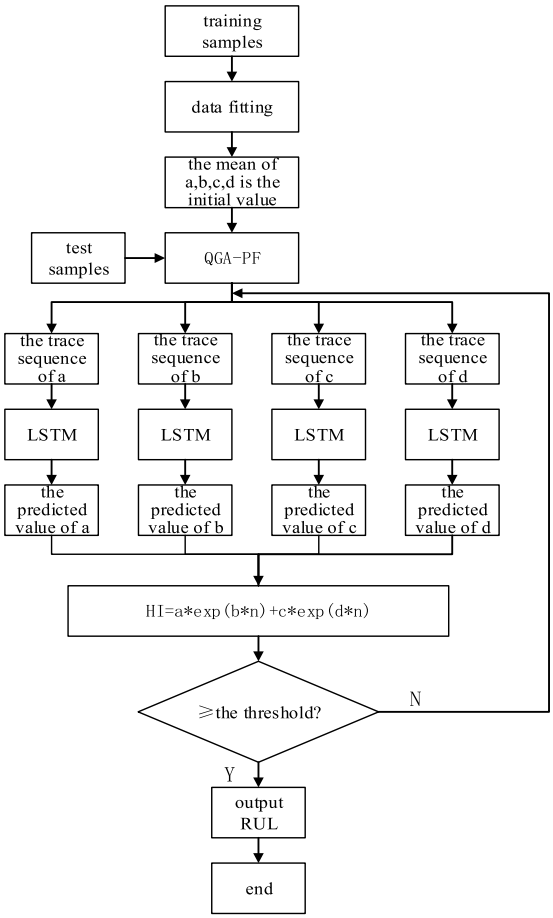

FIGURE 4. RUL prediction process.

the threshold value. If it is greater than or equal to the threshold value, the RUL of the unit is equal to the current time minus the prediction start time. Otherwise, add the prediction coefficients into the tracking sequences and turn to step (3).

To test the prediction effect of the method proposed in this paper, the observation time of HI = 0.5 is set as the beginning time of prediction, the observation time of HI = 0.8 is set as the end time of prediction, The time interval from the beginning to the end is the RUL. The unit samples with the maximum HI≥0.8 are selected from 60 test samples for the experiment. There are 52 units accords with the condition. The 52 units are ranked in order of their actual RUL. PF, PF-LSTM, LSTM, QGAPF, and QGAPF-LSTM are used to respectively predict the RUL of each unit, and the results are shown in figure 5. The method of PF prediction is to use the coefficients obtained from the last tracking of PF and substitute them into equation (19) for prediction. The approaches of PF-LSTM and QGAPF-LSTM are to input the coefficients after PF and QGAPF tracking into the LSTM network for prediction respectively and calculate the corresponding HI by substituting them into equation (19). The approach of LSTM is to directly input the HI sequence before the prediction beginning of each unit into the LSTM network for prediction.

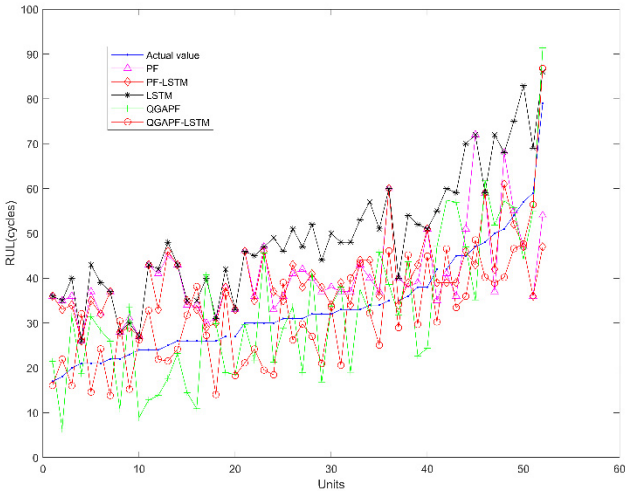

FIGURE 5. The RUL prediction results of the test samples.

It should be said that the parameters of all the LSTM networks in this paper are set the same. The number of hidden units is 100, and the maximum number of iterations is 100. The RMSEs of the above prediction methods are shown in table 3.

TABLE 3. The RUL prediction RMSE of the test samples.

Methods	RMSEs
PF	11.3578
PF-LSTM	9.9024
LSTM	16.6034
QGAPF	9.6436
QGAPF-LSTM	7.8919

As can be seen from figure 5 and table 3, the prediction accuracy of QGAPF is higher than PF, and the prediction method combined with QGAPF and LSTM has a better prediction accuracy.

IV. CONCLUSION

In this paper, A HI extraction method based on multi-feature fusion is introduced for the RUL prediction. QGA is employed to improve the problem of particle degradation in PF. On the basis of filter tracking, LSTM is used to predict the trend of model coefficients, which further improves the accuracy of RUL prediction. The experiment using the C-MAPSS data set shows the proposed method has a better prediction accuracy than other methods. Of course, there are still some shortcomings in this paper that need further research and improvement. For example, the computing speed of LSTM is relatively slow, which requires further research on the computing speed.

REFERENCES

[1] M. Xia, T. Li, T. Shu, J. Wan, C. W. de Silva, and Z. Wang, "A two-stage approach for the remaining useful life prediction of bearings using deep neural networks," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3703–3711, Jun. 2019.

[2] A. R. Bastami, A. Aasi, and H. A. Arghand, "Estimation of remaining useful life of rolling element bearings using wavelet packet decomposition and artificial neural network," *Iranian J. Sci. Technol., Trans. Elect. Eng.*, vol. 43, pp. 233–245, Jul. 2019.

[3] N. Zhang, L. Wu, Z. Wang, and Y. Guan, "Bearing remaining useful life prediction based on Naive Bayes and Weibull distributions," *Entropy*, vol. 20, no. 12, p. 944, Dec. 2018.
[4] Y. Lei, N. Li, S. Gontarz, J. Lin, S. Radkowski, and J. Dybala, "A model-based method for remaining useful life prediction of machinery," *IEEE Trans. Rel.*, vol. 65, no. 3, pp. 1314–1326, Sep. 2016.
[5] Y. C. Liu, X. F. Hu, and W. J. Zhang, "Remaining useful life prediction based on health index similarity," *Rel. Eng. Syst. Saf.*, vol. 185, pp. 502–510, May 2019.
[6] H. Zheng, G. Cheng, Y. Li, and C. Liu, "A new fault diagnosis method for planetary gear based on image feature extraction and bag-of-words model," *Measurement*, vol. 145, pp. 1–13, Oct. 2019.
[7] K. M. Mishra and K. Huhtala, "Elevator fault detection using profile extraction and deep autoencoder feature extraction for acceleration and magnetic signals," *Appl. Sci.*, vol. 9, no. 15, p. 2990, Jul. 2019.
[8] S. Jahani, R. Kontar, S. Zhou, and D. Veeramani, "Remaining useful life prediction based on degradation signals using monotonic B-splines with infinite support," *IISE Trans.*, to be published.
[9] H. Wang, X. Ma, and Y. Zhao, "An improved Wiener process model with adaptive drift and diffusion for online remaining useful life prediction," *Mech. Syst. Signal Process.*, vol. 127, pp. 370–387, Jul. 2019.
[10] G. Prakash, S. Narasimhan, and M. D. Pandey, "A probabilistic approach to remaining useful life prediction of rolling element bearings," *Struct. Health Monit.*, vol. 18, no. 2, pp. 466–485, Mar. 2019.
[11] J. Man and Q. Zhou, "Remaining useful life prediction for hard failures using joint model with extended hazard," *Qual. Rel. Eng. Int.*, vol. 34, no. 5, pp. 748–758, Jul. 2018.
[12] J. Wen, H. Gao, and J. Zhang, "Bearing remaining useful life prediction based on a nonlinear Wiener process model," *Shock Vib.*, vol. 2018, Jun. 2018, Art. no. 4068431.
[13] H. Zhang, M. Chen, X. Xi, and D. Zhou, "Remaining useful life prediction for degradation processes with long-range dependence," *IEEE Trans. Rel.*, vol. 66, no. 4, pp. 1368–1379, Dec. 2017.
[14] L. Li, A. A. F. Saldivar, Y. Bai, and Y. Li, "Battery remaining useful life prediction with inheritance particle filtering," *Energies*, vol. 12, no. 14, pp. 1–18, Jul. 2019.
[15] C. Fangzhou, Q. Liyan, Q. Wei, and L. Hao, "Enhanced particle filtering for bearing remaining useful life prediction of wind turbine drivetrain gearboxes," *IEEE Trans. Ind. Electron.*, vol. 66, no. 6, pp. 4738–4748, Aug. 2019.
[16] X. Zhang, Q. Miao, and Z. Liu, "Remaining useful life prediction of lithium-ion battery using an improved UPF method based on MCMC," *Microelectron. Rel.*, vol. 75, pp. 288–295, Aug. 2017.
[17] L. Cui, X. Wang, Y. Xu, H. Jiang, and J. Zhou, "A novel switching unscented Kalman filter method for remaining useful life prediction of rolling bearing," *Measurement*, vol. 135, pp. 678–684, Mar. 2019.
[18] J. Son, S. Zhou, C. Sankavaram, X. Du, and Y. Zhang, "Remaining useful life prediction based on noisy condition monitoring signals using constrained Kalman filter," *Rel. Eng. Syst. Saf.*, vol. 152, pp. 38–50, Aug. 2016.
[19] N. Li, Y. Lei, T. Yan, N. Li, and T. Han, "A Wiener-process-model-based method for remaining useful life prediction considering unit-to-unit variability," *IEEE Trans. Ind. Electron.*, vol. 66, no. 3, pp. 2092–2101, Mar. 2019.
[20] L. Zhang, Z. Mu, and C. Sun, "Remaining useful life prediction for lithium-ion batteries based on exponential model and particle filter," *IEEE Access*, vol. 6, pp. 17729–17740, Mar. 2018.
[21] Y. Hu, S. Liu, H. Lu, and H. Zhang, "Online remaining useful life prognostics using an integrated particle filter," *Proc. Inst. Mech. Eng. O, J. Risk Rel.*, vol. 232, no. 6, pp. 587–597, Dec. 2018.
[22] B. Yang, R. Liu, and E. Zio, "Remaining useful life prediction based on a double-convolutional neural network architecture," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9521–9530, Dec. 2019.
[23] Q. Wang, B. Zhao, H. Ma, J. Chang, and G. Mao, "A method for rapidly evaluating reliability and predicting remaining useful life using two-dimensional convolutional neural network with signal conversion," *J. Mech. Sci. Technol.*, vol. 33, no. 6, pp. 2561–2571, Jun. 2019.
[24] A. Elsheikh, S. Yacout, and M.-S. Ouali, "Bidirectional handshaking LSTM for remaining useful life prediction," *Neurocomputing*, vol. 323, pp. 148–156, Jan. 2019.
[25] L. Yang, F. Wang, J. Zhang, and W. Ren, "Remaining useful life prediction of ultrasonic motor based on Elman neural network with improved particle swarm optimization," *Measurement*, vol. 143, pp. 27–38, Sep. 2019.
[26] W. Yu, I. Y. Kim, and C. Mechefske, "Remaining useful life estimation using a bidirectional recurrent neural network based autoencoder scheme," *Mech. Syst. Signal Proc.*, vol. 129, pp. 764–780, Aug. 2019.
[27] D. Laredo, Z. Chen, O. Schütze, and J.-Q. Sun, "A neural network-evolutionary computational framework for remaining useful life estimation of mechanical systems," *Neural Netw.*, vol. 116, pp. 178–187, Aug. 2019.
[28] X. Li, F. Elasha, S. Shanbr, and D. Mba, "Remaining useful life prediction of rolling element bearings using supervised machine learning," *Energies*, vol. 12, no. 14, p. 2705, Jul. 2019.
[29] Y. Shen, L. Shen, and W. Xu, "A Wiener-based degradation model with logistic distributed measurement errors and remaining useful life estimation," *Qual. Rel. Eng. Int.*, vol. 34, no. 6, pp. 1289–1303, May 2018.
[30] S. Zhao, Y. Zhang, S. Wang, B. Zhou, and C. Cheng, "A recurrent neural network approach for remaining useful life prediction utilizing a novel trend features construction method," *Measurement*, vol. 146, pp. 279–288, Nov. 2019.
[31] A. R.-T. Palazuelos, E. L. Droguett, and R. Pascual, "A novel deep capsule neural network for remaining useful life estimation," *Proc. Inst. Mech. Eng. Part O-J. Risk Rel.*, to be published, doi: 10.1177/1748006X19866546.
[32] J. S. Nielsen and J. D. Sørensen, "Bayesian estimation of remaining useful life for wind turbine blades," *Energies*, vol. 10, no. 5, p. 664, May 2017.
[33] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. 30th Int. Conf. Mach. Learn. (ICML)*. Atlanta, GA, USA: IMLS, Jun. 2013, pp. 1310–1318.
[34] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[35] Y. Ge, L. Guo, and Y. Dou, "Remaining useful life prediction of machinery based on K-S distance and LSTM neural network," *Int. J. Performability Eng.*, vol. 15, no. 3, pp. 895–901, 2019.
[36] M. S. Hamada, A. G. Wilson, B. P. Weaver, R. W. Griffiths, and H. F. Martz, "Bayesian binomial assurance tests for system reliability using component data," *J. Qual. Technol.*, vol. 46, no. 1, pp. 24–32, 2014.

YANG GE received the B.S. degree in mechatronic engineering from the North University of China, in 2004, and the M.S. and Ph.D. degrees from Army Engineering University, in 2010 and 2016, respectively. He is currently a Lecturer with the Changshu Institute of Technology. His current research interests include fault diagnosis, signal processing, and machine learning.

LINING SUN received the B.S., M.S., and Ph.D. degrees in mechanical engineering from the Harbin Institute of Technology, in 1985, 1988, and 1993, respectively. He is currently a Professor with Soochow University. His current research interests include nanoscale micro-actuated and micro-operated robot, high speed and high precision mechanism, industrial robot technology, parallel robot, medical robot, miniature robot, humanoid arm, and robot mechanism and control.

JIAXIN MA was born in Suzhou, Jiangsu, China, in 1988. He received the B.S. degree from the School of Environment and Safety Engineering, Jiangsu University, in 2010, and the M.S. and Ph.D. degrees from the School of Mechanical Engineering, Southeast University, China, in 2012 and 2019, respectively. His main research interests include fault diagnosis, structural damage identification, and time series analysis and nonlinear system identification.