Journal of Manufacturing Systems 56 (2020) 359–372

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

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

Prognostic study of ball screws by ensemble data-driven particle filters

Yafei Deng[a,b], Du Shichang[a,b,*], Jia Shiyao[a,b], Zhao Chen[a,b], Xie Zhiyuan[a,b]

[a] State Key Lab of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, No. 800 Dongchuan Road, Shanghai, 200240, China

[b] Department of Industrial Engineering and Management, School of Mechanical Engineering, Shanghai Jiao Tong University, No. 800 Dongchuan Road, Shanghai, 200240, China

ARTICLE INFO

Keywords:
Prognostic
Particle filters
Hybrid model
Ball screw
Deep learning

ABSTRACT

The prognostic study of the ball screw is critical to increase the reliability of manufacturing system, which has drawn great attention in the field of Prognostics and Health Management (PHM). The particle filters (PF) method is a powerful tool for prognostic study because of its capability of robustly predicting the future behavior. However, lack of analytical ball screw measurement model limits the application of PF. In this paper, an ensemble GRU network is designed to extend PF to the case where the analytical measurement equation is not available. The proposed hybrid GRU-PF method integrates the data-driven model and the physical model into the particle filters network to realize the prognostic and remaining useful life (RUL) prediction of the ball screw. The effectiveness of the proposed method is validated by designing a ball screw accelerated degradation test (ADT), and the results of this experimental study demonstrate the satisfactory performances in terms of prognostic sensibility.

1. Introduction

The ball screw plays a key role in the computer numerical control (CNC) machine tools to convert rotational motion into linear motion with high accuracy [1]. In practice, the health state of the ball has significant impacts on the machining system in terms of processing performance, manufacturing efficiency and production costs [2–5]. For example, the preload loss of the ball screw will lead to decrease in the stiffness of the feed drive system, and consequently lead to aggravated vibrations and decrease in position accuracy, which degrade the machining quality and even cause severe machining errors. Therefore, the prognostic of ball screw gains great emphasis from academic and industrial field.

The degradation of the ball screw is mainly caused by the preload loss, and a great number of approaches have been developed to monitor the preload loss of the ball screw.

Generally, they can be concluded as the physical-model approach and the data-driven approach.

The physical-model based approach aims to detect the degradation of the ball screw through analyzing the dynamic behavior of the ball screw drive system. GH. Feng et al. develops a lumped dynamic model to investigate the relationship between preload loss and vibration signal spectrum, and it is found that the preload variation can be detected by peak frequency shift and the magnitude variation of its peak frequency [1]. Tsai et al. employs Angular Velocity Vold–Kalman Filtering Order Tracking (AVVKF-OT) to detect the ball passing frequency, which can determine the onset of ball screw degradation [6]. Wei et al. establishes the relationship between the sliding wear rate and degradation based on elastic-plastic microcontact model [7]. Nguyen et al. proposes a discrete dynamic model to characterize the degradation level by evaluating the axial natural frequency from the motor current signal and axial vibration signal [3]. These physical models mainly focus on exploiting the fault mechanism of the ball screw. However, these approaches may not be the most feasible for practical prognostic of ball screw, since the uncertainty in the machining process and the measurement noise are not incorporated in the physical models, and it is difficult to perform extensive experiments to identify some model parameters.

The data-driven approach investigates the mechanical components degradation from monitoring data instead of building physical models, which aims at learning the invisible mapping relationship [8–10]. For the ball screw prognosis problem, the data-driven approach solves the issues through two steps, the representative feature selection from raw sensory data and different fault modes classification. The Hilbert-Huang Transform (HHT) and Multiscale entropy (MSE) are used to extract features and evaluate the degradation levels of the ball screw in [11].

* Corresponding author at: State Key Lab of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, No. 800 Dongchuan Road, Shanghai, 200240, China.
E-mail address: lovbin@sjtu.edu.cn (D. Shichang).

https://doi.org/10.1016/j.jmsy.2020.06.009
Received 3 February 2020; Received in revised form 16 June 2020; Accepted 16 June 2020

The dynamic classifier selection (DCS) strategy and a novel local class accuracy technique (N-LCA) are proposed to differentiate the degradation levels of the ball screw in [12]. Li et al. [4] first proposes a systematic ball screw prognostic methodology, which effectively achieves the early diagnosis, degradation assessment and residual useful life (RUL) prediction of the ball screw. Since the deep learning (DL) method has become a rapidly growing research issue in PHM area [13,14], there are several data-driven approaches based on deep learning for the prognostic of the ball screw in the recent five years. Jin [15] proposes an enhanced Restricted Boltzmann Machine (RBM) method to reduce the effect of regimes on the extracted features which could better represent the ball screw actual degradation. The deep belief network (DBN) and deep residual neural network (DRNN) are utilized to evaluate the degradation condition of the ball screw with multi-sensor data [16,17]. Since the performance of the data-driven approaches highly relies on the quality of the training data, a large amount of run-to-failure experimental data is required to train the model. Another limitation of the data-driven approach is the lack of generality, the trained model is effective to monitor the preload loss of the ball screw under specific situation, such as the constant load and speed, but it may not cover the variance in actual machining operation.

Both the pure physical-model and pure data-driven method have their own limitations when applied to ball screw prognostic study. Therefore, hybrid models combining the physical-based knowledge and the monitoring data have been proposed for diagnostic and prognostic issues recently, which are expected to benefit from both and improve the performance by better exploiting all available information [18]. A key issue in hybrid models is that a proper integration framework should be constructed as the bridge process between the physical-based model and data-driven model. Various researches have been investigated to develop different integration structures. Zhao develops the Bayesian method to combine the physical wear model and condition monitoring data for gear remaining life prediction [19,20]. Yu proposes a physical-informed machine learning strategy for prognostic study of material removal rate [21]. Deng employs the latent Wiener process to integrate the machine learning regressor and physical parametric probabilistic models for RUL prediction [22]. Chang conducts the unscented Kalman filter (UKF) to combine the complete ensemble empirical mode decomposition (CEEMD) and relevance vector machine (RVM) to evaluate the degradation of lithium-ion battery [23].

Besides these hybrid methods, the particle filter (PF) shows great potential in construction of hybrid models since it achieves the real-time health assessment by combining the state transition function and the measurement updating function, where the former is suitable for physical model and the latter could be established by data-driven model. Many machine learning (ML) methods have been developed to build the measurement equation when the analytical formulation is not available in PF. Saha introduces the relevance vector machine (RVM) into PF to learn the non-linear patterns of measurement data for prognostic study of lithium-ion batteries [24]. Wang investigates autoregressive (AR) model and support vector regression (SVR) model to build the measurement equation between the fused features and tool conditions for RUL prediction of machine tools [25,26]. The Kullback–Leibler (KL) divergence is investigated as a general indicator to build the measurement model for machine tools in [27]. Zhang integrates the linear regression (LR) model as the observation function into particle filter for online diagnosis and prognosis of machine tools [28]. Jha proposes a hybrid particle framework for health monitoring, in which the measurement equation is obtained from nominal part of the interval valued analytical redundancy relations (I-ARR) [29]. Daroogheh develops three different types of neural network (NN) paradigms in the observation module of PF to enhance the robustness of proposed prognosis model [30]. Recently, the arising deep learning methods (DL) have been explored in constructing the measurement model for PF. Cadini designs a hybrid framework called as multi-layer perceptron-based particle filter (MLP-PF) to predict the RUL of lithium-ion batteries, in which the MLP network is utilized as measurement model to map the relationship between hidden health state and observation data [18]. Huang combines the bidirectional long short-term memory (BLSTM) networks and PF together for RUL estimation of engineering systems, and measurement model based on BLSTM could automatically learn temporal long-term dependencies [31].

However, the above hybrid data-driven particle filter method may suffer the instability problem cause by model uncertainty missing. This limitation is due to the fact that data-driven method usually provides point estimations without prediction intervals, and the lack of uncertainty assessment could seriously affect the measurement reliability in particle filter. Only a few researchers investigate the construction of model uncertainty when applying data-driven method into PF. Baraldi proposes a bagged ensemble artificial neural network (ANN) to estimate the model uncertainty in the measurement model, where the prediction interval is obtained from the variance of multiple prediction results [32]. Since the bagging ensemble process is time-consuming, this technique may be not feasible for real-time prognostic application.

In order to supplement the gaps of previous research, a novel ensemble data-driven model called as Monte-Carlo Dropout based Gated Recurrent Unit (MC-GRU) is integrated with particle filter in this paper. The intellectual merits of this paper rest on the following:

1) A novel hybrid MC-GRU-PF model is proposed to achieve real-time degradation monitoring and RUL prediction of the ball screw in which the analytical measurement model is not available.
2) The proposed ensemble particle framework MC-GRU could enhance the performance of particle filters in two aspects. Firstly, the proposed ensemble technique could alleviate the instability problem without bring much computational burden. Secondly, compared with the traditional GRU network, the proposed MC-GRU could provide interval estimations instead of point estimations by embedding the MC-Dropout layer. It could bridge the gap of lacking the model uncertainty quantification when utilizing data-driven model in PF and enhance the robustness of the measurement equation.
3) A ball screw degradation experiment bed is built up and the accelerated degradation test (ADT) is conducted to evaluate the performance of proposed model. The comparative results show that proposed method outperform other ensemble and hybrid methods in terms of real-time feasibility and uncertainty interpretation.

The rest of the papers is organized as follows: The theoretical background of PF and GRU is briefly introduced in Section 2. Then, the mathematical framework of the proposed hybrid model is formulated in Section 3. After that, experimental studies on the ball screw degradation are conducted and the comparative results are presented in Section 4. Finally, conclusions are provided in Section 5.

2. Theoretical background

2.1. Particle filter

In the prognostic field, the particle filter aims at tracking the evolution of an equipment degradation state based on a sequence of observations. Based on the Monte Carlo (MC) principle, the particle filter provides discrete random measures composed of numeral particles which are possible values of the unknown state $x(t)$ at time t [33]. When a new measurement is available, the poster distribution in the previous step is used as the prior information in the current step and the parameters in the particles are updated with the likelihood. The particle filter scheme for prognostics assumes that the following information are available:

(1) The knowledge of the system state equation characterizing the evolution of the monitoring equipment degradation with time:

$$x(t+1) = g(x(t), \omega(t)) \quad (1)$$

where g is a transform function and $\omega(t)$ is a possibly non-Gaussian noise vector.

(2) A set of measurements $\boldsymbol{z} = z(1), \cdots z(t)$ of the monitoring equipment. The measurement vector is related to the degradation state $\boldsymbol{x}(t)$, and it could be indirect sensory data or direct observation.
(3) A probabilistic measurement equation which links the measurement vector $\boldsymbol{z}$ and the equipment state $\boldsymbol{x}$.

$$z(t) = h(x(t), v(t)) \quad (2)$$

where h describes the relationship between the degradation state and the corresponding measurement values, and $v(t)$ is the measurement noise vector.

Generally, the process of particle filter contains three steps: prediction, updating and resampling. At the first step, a set of particles (i weighted random samples) representing the initial state of the equipment is drawn from the prior distribution. Then, the following three steps are conducted. In the prediction step, the current state of the equipment $x^i(t)$ is obtained according to the state transform model of Eq. (1), and the particles evolve independently on each other. Next in the updating step, a new measurement data is collected, and the likelihood of measurement data is employed to adjust the weights of the particles proportionally. For example, a good match between the collected measurement value and the particle state will lead to an increase of the corresponding particle's weight. The measurement equation (Eq. (2)) is required in this step to perform the probabilistic law which links the state of the particles and measurement data. Finally, the particles with small weights are eliminated and particles with larger weights are duplicated in the resampling step. A new set of particles is generated by resampling several times from $p(x^i(t)|z^i(t))$. The flowchart of the particle filter method is illustrated in Fig.1.

2.2. Gated recurrent unit

Gated recurrent unit (GRU) is a gating mechanism in recurrent neural networks (RNNs), which can handle the time-series forecasting with the ability of encoding temporal information. GRU networks employ their internal state (memory) to process the sequential data, and this makes GRU ideal for machine health monitoring issues [34]. Compared with another popular variants of RNNs, Long short-term memory (LSTM), GRU has fewer parameters. LSTM utilizes three different gates (forget gate, input gate and output gate) to decide what should be forgot or memorized. The schematic of LSTM is shown in Fig.2a. Roughly speaking, the forget gate determines what is relevant to reserve from previous steps. The input gate decides what information should be added from the current step. The output gate decides the next hidden state. In GRU networks shown in Fig.2b, the input gate and update gate are merged into the update gate, which decides what information to forget and what new information to add. The mathematic formulation of GRU is expressed as [35]:

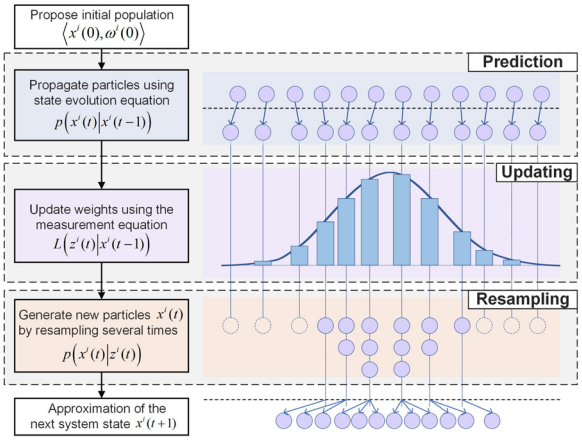


Fig. 1. Flowchart of the particle filter method.

$$z_t = \sigma(W_z \bullet [h_{t-1}, x_t]) \quad (3)$$

$$r_t = \sigma(W_r \bullet [h_{t-1}, x_t]) \quad (4)$$

$$\widetilde{h}_t = \tanh(W[r_t * h_{t-1}, x_t]) \quad (5)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \widetilde{h}_{t-1} \quad (6)$$

where z_t represents the update gate, which controls how much information should be added to the next state. If the value of update gate is close to 1, most of information should be brought to the next state. The reset gate r_t is employed to determine how much past information should be forgot. A smaller value of the reset gate leads to more information ignored from the previous cell. $\widetilde{h}_t$ represents the candidate hidden state, which could be understood as the information added to the current memory. The output hidden state h_t is a linear interpolation between $\widetilde{h}_t$ and h_{t-1}, and the output can be regarded as a combination of the prior memory and the current memory.

From the perspective of RUL prediction, the innovative gate structures in GRU are expected to improve the prognosis performance since each unit will learn to capture the characteristic of degradation dependencies over different time scales separately. For example, the reset gates which focus on the new information from the current health state, enable the network of learning short-term dependencies when they are frequently activated. On the other hand, the update gates, which focus on keeping the health state information from previous memories, make the prognosis model capable of capturing the longer-term degradation dependencies.

Another key improvement of both LSTM and GRU is that the gradient vanishing and explosion problems in traditional RNNs are solved. However, GRU could reach nearly equivalent performance in relative less training time compared with LSTM. In this paper, an improved GRU architecture called as MC-GRU is designed to enhance the model robustness through providing interval estimations instead of point estimations and incorporated into the particle filter model to formulate the measurement equation.

3. Methodology

In this paper, a hybrid model based on the modified particle filters is designed for RUL prediction of ball screws. The architecture of the proposed model is shown in Fig. 3.

First, the proposed MC-GRU network is employed to formulate the measurement model which is not available in an analytical form for prognosis of ball screws. And an ensemble process based on MC-Dropout layer enables the MC-GRU provide interval estimations for the construction of the measurement model uncertainty. Then the physical based model of ball screw is derived from the Archard wear law, thus the degradation state equation could be formulated. Finally, the measurement equation obtained from MC-GRU and state equation formulated based on the physical model are fused into the particle filter framework, where the health state of ball screw could be real-time updated when new observations are collected.

3.1. Measurement equation construction based on MC-GRU

During the degradation process of the ball screw, many measurement data including indirect measurements and direct measurements could be acquired. However, there are challenges hinder access of analytical measurement equation based on these data:

1) The indirect measurements, such as position accuracy and ball

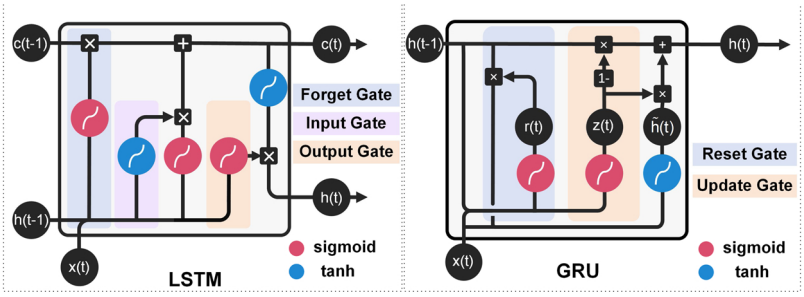


Fig. 2. The structure of a) LSTM b) GRU.

screws stiffness, are obtained under the manufacturing equipment downtime, which is an offline process and would affect the normal production schedule. Using the indirect measurement to construct analytical formulation would run counter to the idea of real-time prognostic based on particle filtering.

2) The direct measurements, such as monitoring vibration signals from ball screws, will be severely influenced by the low signal to noise ratio (SNR) of sensing measurement and other irrelevant components. These contaminated observations may lead to difficulty to construct a stable analytical measurement model.
3) When building the measurement model in an analytical form, parameters of different components should be obtained. It is hard to measure the value of these components once they have been installed in the CNC machine tools.

Although it is hard to obtain an explicit form of measurement equation, it is possible to build a ball screw dataset $\boldsymbol{T}\{[\boldsymbol{x}_j, \boldsymbol{z}_j], j = 1,\cdots,N\}$ which contains the state vector and the corresponding measurement vector, and the measurement model could be obtained by exploiting $\boldsymbol{T}$ with the data-driven approach.

The unknown measurement equation can be expressed in the form:

$$\boldsymbol{z}(t) = f[\boldsymbol{x}(t)] + v[\boldsymbol{x}(t)] \tag{7}$$

where $\boldsymbol{z}(t)$ and $\boldsymbol{x}(t)$ are the measurement vector and state vector at time t from the dataset $\boldsymbol{T}$ respectively, $f(\bullet)$ is an invertible mathematical function and $v[\boldsymbol{x}(t)]$ is a zero mean Gaussian noise.

Because the ball screw measurement function $f(\bullet)$ could not be expressed in an analytical form, an interpolator $\bar{\varphi}(\boldsymbol{x})$ is proposed to model the mapping relationship between the state vector $\boldsymbol{x}$ and measurement vector $\boldsymbol{z}$. The framework is shown as Fig. 4, including two modules: the dataset preparation, and measurement model construction based on MC-GRU network. The first module aims at building an informative dataset from raw monitoring data, which could extract representative features as the measurement vector. The second module conducts the MC-GRU network to construct a robust measurement model based on the ball screw dataset, where the uncertainty from model parameters could be obtained from the model ensemble process.

3.1.1. The dataset preparation

The performance of the dataset will be degraded if taking the raw sensory data as the measurement vector directly, since the raw sensory data may include too much redundancy, which are irrelevant to the current health state of the ball screw. In this section, a data preprocessing strategy is described to build the dataset.

The raw sensory data is first segmented into N slices and the length of segment window is $\frac{L}{N}$ (L is the sampling length of the raw measurement). Then the tri-domain features, which includes time domain, frequency domain and time-frequency domain are extracted from each window. The extracted features can be translated into a feature vector $\boldsymbol{z}_j (j = 1,2,\cdots,N)$. $\boldsymbol{z}_j \in \mathbb{R}^{m\times n}$ indicates the jth measurement vector including $m \times n$ features, where m and n are the sensor numbers and feature types respectively. Compared with the raw noisy measurement data, the new measurement vector $\boldsymbol{z}_j$ is much shorter and could reflect more discriminative information. Subsequently, the feature selection strategy based on the prognostic criteria is conducted to find more representative features. The feature ranking criteria is calculated according to the monocity, trendability and correlation coefficient with the wear value, since the ball screw degradation is normally characterized a monotonic process. On the other hand, $\boldsymbol{x}_j$ including the wear state of the ball screw at observation moment j can be updated through state equation. In the end, the newly constructed measurement vector $\boldsymbol{z}_j$ and the corresponding state vector $\boldsymbol{x}_j$ will be put into the dataset $\boldsymbol{T}$ for training the ensemble GRU network.

3.1.2. The proposed MC-GRU network

The MC-GRU network is developed by three steps. First, a Bi-Directional GRU network is designed to construct the relationship between $\boldsymbol{x}_j$ and $\boldsymbol{z}_j$. Subsequently, the Monte-Carlo Dropout (mc-dropout) strategy is applied to regularize the model. Finally, the ensemble output $\bar{\varphi}(\boldsymbol{x})$ is obtained by running repeatedly with varied mc-dropout masks

(1) Bi-Directional GRU network

The Bi-Directional GRU (BD-GRU) network is constructed to build a multivariate regression model between the state vector and measurement vector. Compared with GRU model, the BD-GRU network has two

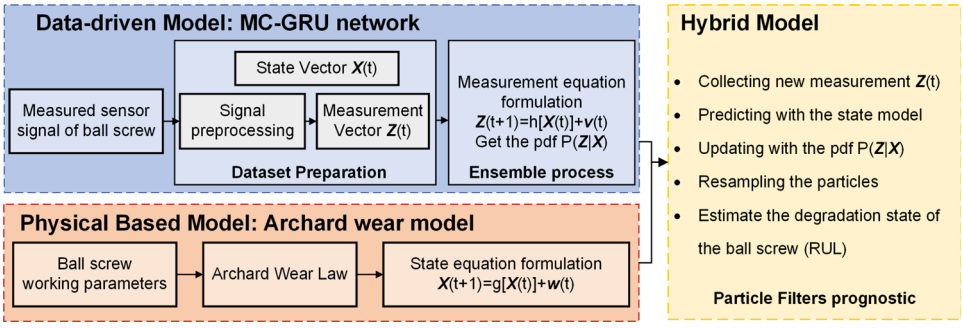


Fig. 3. Architecture of the proposed hybrid prognostic model for ball screw.

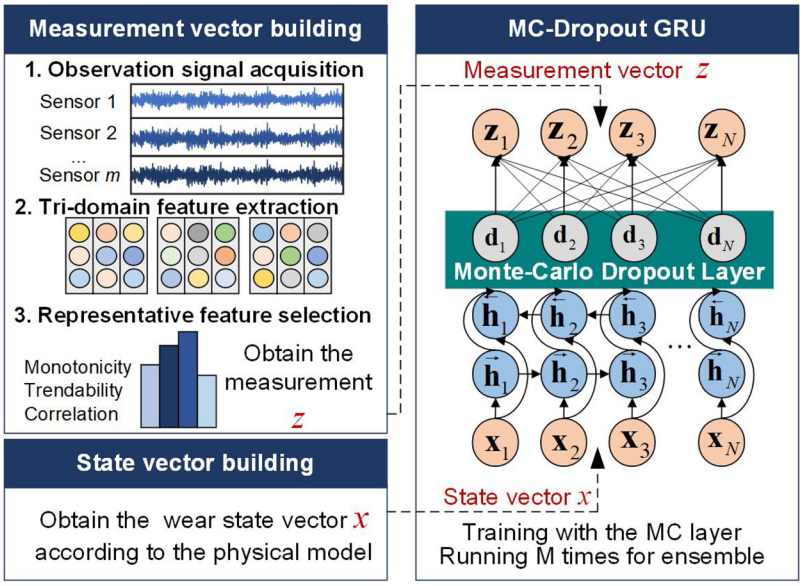


Fig. 4. Framework of the measurement equation building based on ensemble GRU.

individual hidden layers, which process the input sequence in two directions including forward and backward propagations. The bidirectional structure enables the hidden units to capture the past (forward direction) and the future (backward direction) information jointly, which is expressed as:

$$\bar{\boldsymbol{h}} = [\overrightarrow{\boldsymbol{h}_j}, \overleftarrow{\boldsymbol{h}_{N-j+1}}] j = 1,2\cdots,N \tag{8}$$

$$\overrightarrow{\boldsymbol{h}_j} = \overrightarrow{\mathbb{F}}(\boldsymbol{x}_j, \overrightarrow{\boldsymbol{h}_{j-1}}) \tag{9}$$

$$\overleftarrow{\boldsymbol{h}_j} = \overleftarrow{\mathbb{F}}(\boldsymbol{x}_j, \overleftarrow{\boldsymbol{h}_{j-1}}) \tag{10}$$

where $\bar{\boldsymbol{h}}$ is the completed hidden element at each time step j, which is the concatenated vector $\overrightarrow{\boldsymbol{h}_j}$ and $\overleftarrow{\boldsymbol{h}_j}$. $\overrightarrow{\boldsymbol{h}_j}$ indicates the hidden state vector of the last units in GRU in forward direction and $\overleftarrow{\boldsymbol{h}_j}$ is for backward direction. $\mathbb{F}$ denotes the transition function in GRU, which is defined by Eqs. (3–6).

(2) Ensemble through Monte-Carlo Dropout technique

The output of BD-GRU is defined as $\varphi(\boldsymbol{x})$ to model the mapping relationship between the state and measurement vector. Since the obtained output $\varphi(\boldsymbol{x})$ depends on the dataset being fed into the network, a large scale of dataset is important for the model's robust performance. In the ball screw residual life prediction problems, it is hard to obtain a large scale of training data and the BD-GRU may suffer from instability. To overcome this issue, the Monte-Carlo Dropout (mc-dropout) technique is developed for the proposed model. The mc-dropout is same as dropout in that they both randomly turn off some neurons in the network, so the influence of these masked neurons will be eliminated during the training phase. Generally, the mc-dropout technique as well as dropout can be seen as an approach to enlarge the scale of dataset, since the masking of random neurons will generate new variants in data samples. Unlike the dropout is turned off during the testing phase, the mc-dropout will be kept activated during the testing phase. This difference enables the mc-dropout technique to perform multiple forward propagation processes on the same input and to simulate the outputs of different network structures. The ensemble output $\bar{\varphi}(\boldsymbol{x})$ can be obtained through averaging the outputs $\varphi(\boldsymbol{x})$ after T times mc-dropout, which is expressed as Eq. (11). The mc-dropout enables the GRU model to build different network structures for testing, which could conduct the ensemble within one model. This could mitigate the computational burden compared with other parallel ensemble methods loading multiple models.

$$\bar{\varphi}(\boldsymbol{x}) = \frac{1}{T}\sum_{j=1}^{T} \varphi_j(\boldsymbol{x}) \tag{11}$$

It should be noticed that $f[\boldsymbol{x}(t)]$ could not be directly replaced by the obtained $\bar{\varphi}[\boldsymbol{x}(t)]$, since the model uncertainty should be considered. By subtracting the $\bar{\varphi}[\boldsymbol{x}(t)]$ from both sides of Eq. (7), the measurement model can be written as:

$$\boldsymbol{z} - \bar{\varphi}(\boldsymbol{x}) = [f(\boldsymbol{x}) - \bar{\varphi}(\boldsymbol{x})] + v(\boldsymbol{x}) \tag{12}$$

where the term $\boldsymbol{z}(t) - \bar{\varphi}(\boldsymbol{x})$ is the prediction uncertainty of $\bar{\varphi}(\boldsymbol{x})$ output with respect to the measurement vector $\boldsymbol{z}$. The prediction uncertainty includes two parts: the model uncertainty and the measurement uncertainty.

(1) Model uncertainty: $f(\boldsymbol{x}) - \bar{\varphi}(\boldsymbol{x})$ is a random difference between the unknown $f(\boldsymbol{x})$ and the ensemble output $\bar{\varphi}(\boldsymbol{x})$, which can be also called as the epistemic uncertainty. The model uncertainty accounts for uncertainty in the model parameters, and this type of uncertainty could be reduced by training the model with larger scale of input data. The variance of the model uncertainty is indicated as $\sigma^2(\boldsymbol{x})$.
(2) Measurement uncertainty: $v(\boldsymbol{x})$ is the noise inherent during the measurement process, and the variance of $v(\boldsymbol{x})$ is indicated as $\varepsilon^2(\boldsymbol{x})$.

The procedures of estimating the model uncertainty and measurement uncertainty is described in section 3.1.3.

3.1.3. Estimation of the model uncertainty and measurement uncertainty

Inspired by the idea of representing the uncertainty in deep neural network [36], the mc-dropout is employed to estimate the model uncertainty. To estimate the model uncertainty, a prior distribution has been put over the weights of the Bi-directional GRU network, for example a Gaussian prior distribution: $W \sim \mathcal{N}(0, I)$. The proposed Bi-directional GRU could be referred to a Bayesian neural network (BNN), which aims at estimating the distributions over the weight parameters rather than finding the optimum weight parameters. The output of BNN could be denoted as $\varphi^W(\boldsymbol{x})$ and the model likelihood is defined as $p[f(\boldsymbol{x})|\varphi^W(\boldsymbol{x})]$. Given the dataset $\boldsymbol{T}\{[\boldsymbol{x}_j, \boldsymbol{z}_j], j = 1,\cdots,N\}$, the posterior distribution of the weights $p(\boldsymbol{W}|\boldsymbol{T})$ could be updated. Since the posterior $p(\boldsymbol{W}|\boldsymbol{T})$ is difficult to perform on the heterogeneous GRU model,

the mc-dropout layer added in the model is employed to estimate the posterior distribution. The key idea of mc-dropout can be interpreted as a variational approximation inference to obtain a cluster with weight distributions $\widehat{\boldsymbol{W}}_j$, and the cluster $\widehat{\boldsymbol{W}}_j$ minimizes the Kullback-Leibler (KL) divergence to the true model posterior of $f(\boldsymbol{x})$. After multiple mc-dropout, the variance of the model uncertainty could be estimated by marginalizing over the approximated weight posterior distribution, which is expressed as:

$$\sigma^2(\boldsymbol{x}) \approx \frac{1}{T}\sum_{j=1}^{T} \varphi^{\widehat{\boldsymbol{W}}_j}(\boldsymbol{x})^T \varphi^{\widehat{\boldsymbol{W}}_j}(\boldsymbol{x}) - E[f(\boldsymbol{x})^T]E[f(\boldsymbol{x})] \tag{13}$$

with predictions in the model done by approximating the mean [36]: $E[f(\boldsymbol{x})] \approx \frac{1}{T}\sum_{j=1}^{T} \varphi^{\widehat{\boldsymbol{W}}_j}(\boldsymbol{x})$. Since the procedure of conducting multiple mc-dropout has been implemented in section 3.1.2, the variance of model uncertainty can be shortened to:

$$\sigma^2(\boldsymbol{x}) \approx \frac{1}{T}\sum_{j=1}^{T} [\varphi_j(\boldsymbol{x}) - \bar{\varphi}(\boldsymbol{x})]^2 \tag{14}$$

After obtaining the variance of the model uncertainty $\sigma^2(\boldsymbol{x})$, the variance of the measurement uncertainty can be calculated accordingly. From Eq. (12), one can derive [32]:

$$\begin{aligned} Var[\boldsymbol{z} - \bar{\varphi}(\boldsymbol{x})] &= Var + Var[v(\boldsymbol{x})] + 2E\{[f(\boldsymbol{x}) - \bar{\varphi}(\boldsymbol{x})]v(\boldsymbol{x})\} \\ &= \sigma^2(\boldsymbol{x}) + \varepsilon^2(\boldsymbol{x}) \end{aligned} \tag{15}$$

The term $[\boldsymbol{z} - \bar{\varphi}(\boldsymbol{x})]$ is assumed as a zero mean random variable, so the variance $Var[\boldsymbol{z} - \bar{\varphi}(\boldsymbol{x})]$ can be transformed to $E\{[\boldsymbol{z} - \bar{\varphi}(\boldsymbol{x})]^2\}$. Given the training dataset $\boldsymbol{T}\{[\boldsymbol{x}_j, \boldsymbol{z}_j], j = 1, \cdots, N\}$, $E\{[\boldsymbol{z} - \bar{\varphi}(\boldsymbol{x})]^2\}$ can be estimated by $[\boldsymbol{z}_j - \bar{\varphi}(\boldsymbol{x}_j)]^2$, and thus the variance of measurement uncertainty can be obtained, which is expressed as:

$$\varepsilon^2(\boldsymbol{x}) = max\{[\boldsymbol{z}_j - \bar{\varphi}(\boldsymbol{x}_j)]^2 - \sigma^2(\boldsymbol{x}), 0\} \tag{16}$$

Based on the assumption that $\bar{\varphi}(\boldsymbol{x})$ is an estimate of $f(\boldsymbol{x})$ ($E[f(\boldsymbol{x})] \approx \frac{1}{T}\sum_{j=1}^{T} \varphi^{\widehat{\boldsymbol{W}}_j}(\boldsymbol{x})$), the measurement distribution of $p[\boldsymbol{z}|f(\boldsymbol{x})]$ can be approximated by the distribution $p[\boldsymbol{z}|\bar{\varphi}(\boldsymbol{x})]$. According to Eq. (12), $p[\boldsymbol{z}|\bar{\varphi}(\boldsymbol{x})]$ can be derived from the distribution $p[\bar{\varphi}(\boldsymbol{x})|f(\boldsymbol{x})]$ and the distribution of measurement noise $v(\boldsymbol{x})$. Since the mean and variance of these two distributions have been estimated in section 3.1.3, $p[\boldsymbol{z}|f(\boldsymbol{x})]$ can be approximated as a Gaussian distribution, with mean $\bar{\varphi}(\boldsymbol{x})$ and variance $\sigma^2(\boldsymbol{x}) + \varepsilon^2(\boldsymbol{x})$. Finally, the distribution $p(\boldsymbol{z}|\boldsymbol{x})$ of the measurement vector $\boldsymbol{z}$ with the corresponding state vector $\boldsymbol{x}$ can be expressed as:

$$p(\boldsymbol{z}|\boldsymbol{x}) \approx p[\boldsymbol{z}|f(\boldsymbol{x})] \approx N(\bar{\varphi}(\boldsymbol{x}), \sigma^2(\boldsymbol{x}) + \varepsilon^2(\boldsymbol{x})) \tag{17}$$

3.2. State equation formulation based on physical wear law

The state equation characterizes the underlying ball screw degradation behavior evolving with time. Since the preload of the ball screw is mainly decreased with the wear depth, the Archard wear model considering dynamic working conditions is used to formulate the state equation of ball screw [37]:

$$V(t) = K_{b-s}\frac{F_{b-s}(t)S_{b-s}(t)}{H_{b-s}} + K_{b-n}\frac{F_{b-n}(t)S_{b-n}(t)}{H_{b-n}} \tag{18}$$

where $V(t)$ represents the wear loss at time t. K_{b-s} and K_{b-n} are the wear coefficient of the ball-screw contact surfaces and ball-nut contact surfaces respectively, which are determined by the material and lubrication condition. H_{b-s} and H_{b-n} are the hardness of the softer material of ball-screw and ball-nut respectively. $F_{b-s}(t)$ represents the contact force between the ball and screw at time t, and $F_{b-n}(t)$ represents the contact force between the ball and nut at time t. $S_{b-s}(t)$ represents the relative sliding displacement between the ball and screw at time t, and $S_{b-n}(t)$ represents the relative sliding displacement between the ball and nut at time t.

As shown in Fig.5a, the contact load F_{b-n} between the ball and nut contact surfaces can be calculated by the force balance equation:

$$Z \bullet F_{b-n}\sin\theta_N cos\alpha - F_a = 0 \tag{19}$$

where Z represents the ball numbers, θ_N is the contact angle of ball and nut, α is the helix angle of the ball screw and F_a is the external axial load. To simplify the wear state model, the ball screw is assumed running at relative low rotation speed and the centrifugal force can be ignored, so θ_N equals to the contact angle of ball and screw θ_S. The contact load can be calculated as:

$$F_{b-s}(t) = F_{b-n}(t) = \frac{F_a(t)}{Zsin\theta_N cos\alpha} \tag{20}$$

The relative sliding distance $S_{b-s}(t)$ between the ball and screw contact point S can be expressed as Eq. (21) [37].

$$S_{b-s}(t) = \sqrt{R^2 + \left(\frac{L}{2\pi}\right)^2}\left(\frac{R_{bs}}{1 + R_{bs}}\right)\omega_{s-b}(t)dt \tag{21}$$

As shown in Fig. 5b, R represents the pitch radius of the ball screw, L represents the lead of the screw and R_{bs} is the sliding-to-rolling ratio at the screw contact point S. $\omega_{s-b}(t)$ is the angular velocity of the screw relative to a ball at time t, which is expressed as:

$$\omega_{s-b}(t) = \frac{r_s(r + r_n cos\theta_N)cos\alpha}{r_s(r + r_n cos\theta_N) + r_n(r - r_s cos\theta_S)}\omega(t) \tag{22}$$

where r_s denotes the distance between the contact point S and ball center, and r_n denotes the distance between the contact point N and ball center. $\omega(t)$ is the rotation angular velocity of the screw.

Similarly, the relative sliding distance $S_{b-n}(t)$ between the ball and nut contact point N can be expressed as:

$$S_{b-n}(t) = \sqrt{R^2 + \left(\frac{L}{2\pi}\right)^2}\left(\frac{R_{bn}}{1 + R_{bn}}\right)\omega_{b-n}(t)dt \tag{23}$$

$$\omega_{b-n}(t) = \frac{-r_n(r - r_s cos\theta_S)cos\alpha}{r_s(r + r_n cos\theta_N) + r_n(r - r_s cos\theta_S)}\omega(t) \tag{24}$$

By substituting Eq. (20–24) into Eq. (18), the wear state equation can be rewritten as:

$$\mathrm{V}(t) = \mathrm{d}V \bullet \mathrm{d}t = K(K_1 + K_2)F_a(t)\omega(t)\mathrm{d}t \tag{25}$$

$$K_1 = \frac{\sqrt{R^2 + \left(\frac{L}{2\pi}\right)^2}}{Z \times H}\left(\frac{R_{bs}}{1 + R_{bs}}\right)\frac{r_s(r + r_n cos\theta_N)}{[r_s(r + r_n cos\theta_N) + r_n(r - r_s cos\theta_S)]sin\theta_N} \tag{26}$$

$$K_2 = \frac{\sqrt{R^2 + \left(\frac{L}{2\pi}\right)^2}}{Z \times H}\left(\frac{R_{bn}}{1 + R_{bn}}\right)\frac{-r_n(r - r_s cos\theta_S)}{[r_s(r + r_n cos\theta_N) + r_n(r - r_s cos\theta_S)]sin\theta_N} \tag{27}$$

Once the type of ball screw is determined, the intrinsic parameters of ball screw including H, R, L, Z, r_n and r_s are fixed, and the change of the contact angle θ_N, θ_S and sliding-to-rolling ratio R_{bn}, R_{bs} during the ball screw operation is fairly small, so the sum of K_1 and K_2 is assumed to be a constant m. The final wear state equation can be derived as:

$$x(t + 1) = [K \times m \times F_a(t)\omega(t)\mathrm{d}t] + x(t) \tag{28}$$

where K is the equivalent wear coefficient, which is assumed changing exponentially according to the lubrication condition.

4. Experimental case study

4.1. Experiment setup

The experimental test platform is first built up to investigate the degradation behavior of the ball screw. The specification of the test platform is list in Table 1, and the schematic and photograph of the ball screw test platform are shown in Fig.6.

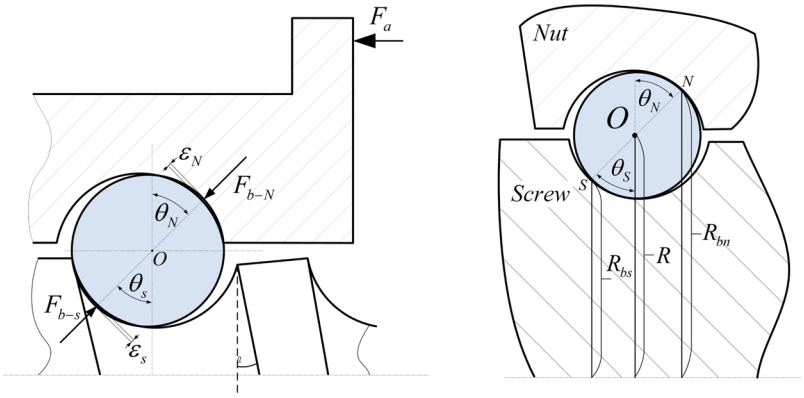


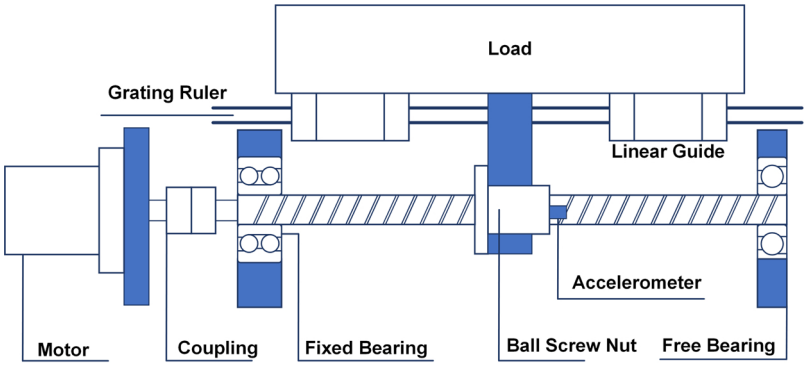
Fig. 5. The ball screw a) Contact load relationship b) sliding-to-rolling ratio.

Table 1
Specification of the ball screw test platform.

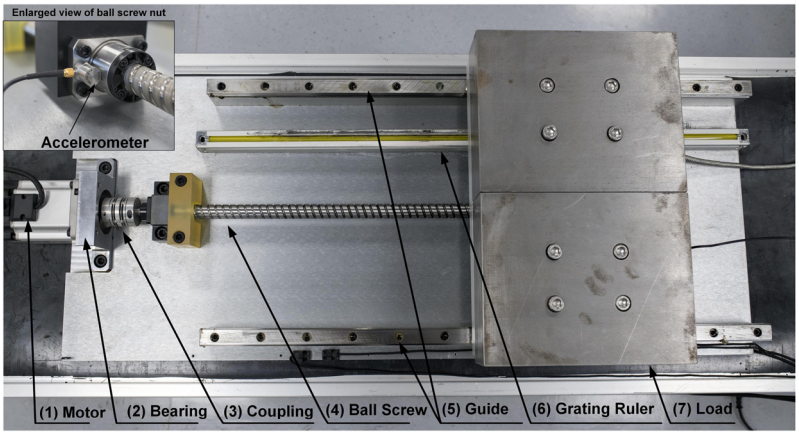
Test platform component	Specification	Main parameters
Ball screw	BNK2020 – 3.6G0-C5 – 742	Dynamic load: 7 kN
Linear guide	SV2R-MX33 – 760	Dynamic load: 11.7 kN
Motor	A5II-MSMJ082G1U	Power: 750 W
Controller	MCDKT3520E	/
PLC	FX3U-32MT/ES-A	/
Accelerometer	HD-YD-216	Axial sensitivity: 100 mV/g

To validate the performance of the proposed GRU-PF method, one accelerated degradation test (ADT) is designed based on a completely new ball screw. In order to accelerate the degradation process, the ball screw is kept running at 400 mm/s constantly with an 800 mm reciprocating stroke. Throughout the ADT, an external load (50 kg) is applied to the worktable, which is located on the ball screw nut. In addition, no further lubrication is applied to the ball screw except at the beginning stage, and the linear guides are lubricated every 50 h to ensure the ball screw wears faster than the guide.

In this experiment, one accelerometer is installed on the ball screw nut to collect the axial vibration signal. As shown in Fig.7, the vibration data is collected from the accelerometer at each hour, and each sampling data lasts 10 s. In addition, the positioning accuracy of the ball



(a) Schematic of the ball screw test platform



(b) Photograph of the ball screw test platform

Fig. 6. a) Schematic of the ball screw test platform. b) Photograph of the ball screw test platform.

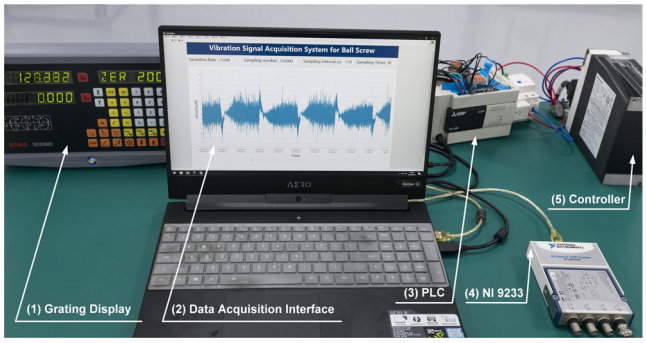


Fig. 7. Data acquisition of accelerated degradation process.

screw is recorded every 50 h by the grating ruler to evaluate the wear state quantitatively. At the end of ADT, totally 550 h vibration data of the ball screw is collected with sampling rate of 25,600 Hz.

4.2. Validation of the state function

During the ADT process, the true wear depth of ball screw is difficult to measure directly, so the proposed wear state equation is verified through the recorded positioning accuracy. The difference dh between the positioning accuracy with wear condition and original condition can be formulated by the cumulative wear depth [38], which can be express as:

$$dh = \frac{V(t)}{\pi ab L_s sin\alpha} dt \tag{29}$$

where πab is the contact ellipse area of screw and ball, a and b are the half lengths of the major axis and the minor axis, and L_s is the total sliding distance of ball screw during dt. The initial parameters of the testing ball screw are given in Table 2, which can be used to calculate the theoretical wear depth based on Eq. (25–27).

The theoretical wear value calculated according to the state equation and the recorded positioning accuracy are shown in Fig. 8. And the equivalent wear coefficient K at the initial point, middle point and final point during ADT is marked, which is obtained by [39]. It can be seen that the proposed state function could roughly reflect the degradation process of the ball screw.

4.3. Dataset preparation

The raw measurement data has a huge data size as 550 × 256000. Firstly, the sliding window with 2560 points is used to iterate over the raw data, so the sliced data could be reshaped into 550 × 100 × 2560. Subsequently the tri-domain features are extracted from the data slice and totally 16 types of features are obtained. The details of extracted features are given in Table 3.

It should be noticed that not all extracted features are suitable as the measurement vector, therefore the feature selection process is conducted finally to find more representative features. The score of features as shown in Fig.9 are calculated according to the selection criteria, and the top three features (RMS, wavelet energy#1 and wavelet energy#2 in the low frequency band) are selected as the measurement vector $\boldsymbol{z}_j$ with data size 550 × 100 × 3.

Table 2
Initial parameters of the ball screw.

Parameters	Value	Unit
Nominal radius R	10.375	mm
Helix pitch L	20	mm
Distance between the contact point and ball center r_s, r_N	1.651	mm
Semi-major axis in the contact ellipse a	1.3	mm
Semi-minor axis in the contact ellipse b	0.14	mm
Contact angle θ_N, θ_S	45	degree
Helix angle α	17.4	degree
Axial load F_a	150.3	N
Rotation angular velocity of the screw ω	20	rad/s
Ball number Z	36	/
Ball screw hardness	62	HRC
Sliding-to-rolling ratio at the screw contact point R_{bs}	0.23	/
Sliding-to-rolling ratio at the nut contact point R_{bn}	1.55	/
Wear coefficient K	3.3E-7	/

4.4. Results and discussions

In this section, three types of comparative experiments are conducted to evaluate the proposed hybrid prognosis framework. The first experiment mainly investigates the real-time feasibility and uncertainty quantification capacity of MC-GRU compared with other ensemble techniques. The second experiment explores the measurement model robustness under different ensemble parameters. The third experiment is employed to verify the prognosis performance based on MC-GRU-PF compared with other data-driven PF methods.

4.4.1. Comparison with other ensemble techniques

In this subsection, the model accuracy and computation efficiency of the ensemble GRU based on MC-Dropout is investigated to ensure that the proposed method could achieve effective real-time ball screw health state updating. Three commonly used ensemble techniques including stacking, boosting and bagging are chosen to make comparison with proposed ensemble GRU based on MC-Dropout model. These ensemble techniques will be briefly introduced subsequently, and the frameworks of these methods are shown in Fig. 10.

Stacking: The main procedure in the stacking-based ensemble is to develop multiple heterogeneous models to improve the robustness. The predictions of these models are given as an input to a combiner algorithm to obtain the final model.

Boosting: The core principle of boosting-based ensemble is to learn a sequence of weak regressors. At each boosting iteration, the training instances are reweighted according to the prediction error, and the subsequent network will put more emphasis on the wrongly predicted instances in the previous step. The ensemble output is obtained by combining the results of all the weak regressors.

Bagging: The key idea of the bagging-based ensemble is to conduct a bootstrap sampling, and each sample is treated as the whole population for model training. A series of parameter sets will be learned from the sub-training datasets and the ensemble output is obtained by averaging each single base model output. In this experiment, 10 % of the total training set is extracted randomly at each bootstrap sampling.

Totally six different ensemble learning models are conducted to compare the performance on the prediction accuracy and computation efficiency. The detailed information of these ensemble models is shown in Table 4. Both DL methods and ML methods have been developed in these ensemble models, and in order to guarantee the effective and fair performance comparison, the hyperparameter of deep learning models used in different ensemble techniques are set to be same. The iteration number is chosen as 1000 and the ensemble number is set as 50 in each ensemble model. The predictions of the measurement vector on testing data based on aforementioned ensemble methods are demonstrated in Fig.11.

To quantitively evaluate the prediction accuracy of the ensemble model as the measurement equation in particle filters framework, the root mean square error (RMSE) and mean absolute error (MAE) of testing data are calculated. And the running time of each model during training and testing is obtained to investigate the computational efficiency and real-time feasibility. The comparative evaluation results of these ensemble methods are summarized in Table 5.

It can be found that these ensemble methods, except ML-Stacking, all achieve high prediction accuracy on the measurement vectors

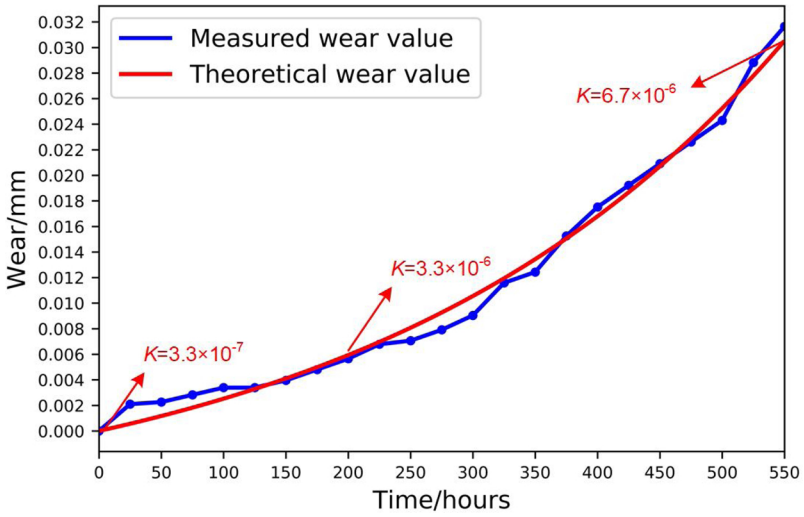


Fig. 8. Accumulated wear value of ball screw during ADT.

Table 3
Extracted tri-domain features.

Time Domain Feature	Equations
Root Mean Square	$x_{\text{mean}} = \sum_{i=1}^{N} x(i)/N$
Peak Value	$x_{\text{peak}} = \max[x(i)] - \min[x(i)]$
Maximum Absolute Value	$x_{\text{ma}} = \max\lvert x(i)\rvert$
Skewness Factor	$x_{\text{skewenss}} = \sum_{i=1}^{N}[x(i) - x_1]^3/[(N-1)x_2^3]$
Kurtosis Factor	$x_{\text{kurtosis}} = \sum_{i=1}^{N}[x(i) - x_1]^4/[(N-1)x_2^4]$
Frequency Domain Feature	Equations
Central Frequency	$f_{\text{cf}} = \sum_{i=1}^{N} f(i)\cdot df$
Spectrum Kurtosis	$f_{\text{skewness}} = \sum_{i=1}^{N}\left[\frac{f(i)-\overline{f(i)}}{\sigma}\right]^3 S[f(i)]$
Spectrum Power	$f_{\text{power}} = \sum_{i=1}^{N} f(i)^3 S[f(i)]$
Time-Frequency Domain Feature	Equations
Wavelet Energy	$w_{\text{energy}} = \sum_{i=1}^{N} \text{wt}_{\phi}^2(i)/N$

according to the low RMSE and MAE value. Therefore, the main comparisons are focused on the running time and prediction type to evaluate the real-time feasibility and model robustness:

(1) For the bagging-based method and stacking-based method, the running time is relatively high because multiple trained models need to be loaded during the ensemble process. The poor performance on computational efficiency would limit the application on real-time measurement prediction.
(2) For the Boosting-based method, the computation time is the lowest among all models, which seems to be an efficient ensemble technique. However, the boosting-based model could only provide a point estimation without confidence intervals. The model uncertainty of the measurement equation would be missed.
(3) For the proposed MC-GRU ensemble network, it could provide interval estimations without sacrificing the prediction accuracy. What's more, the running time is greatly reduced compared with the bagging-based techniques which could also provide interval

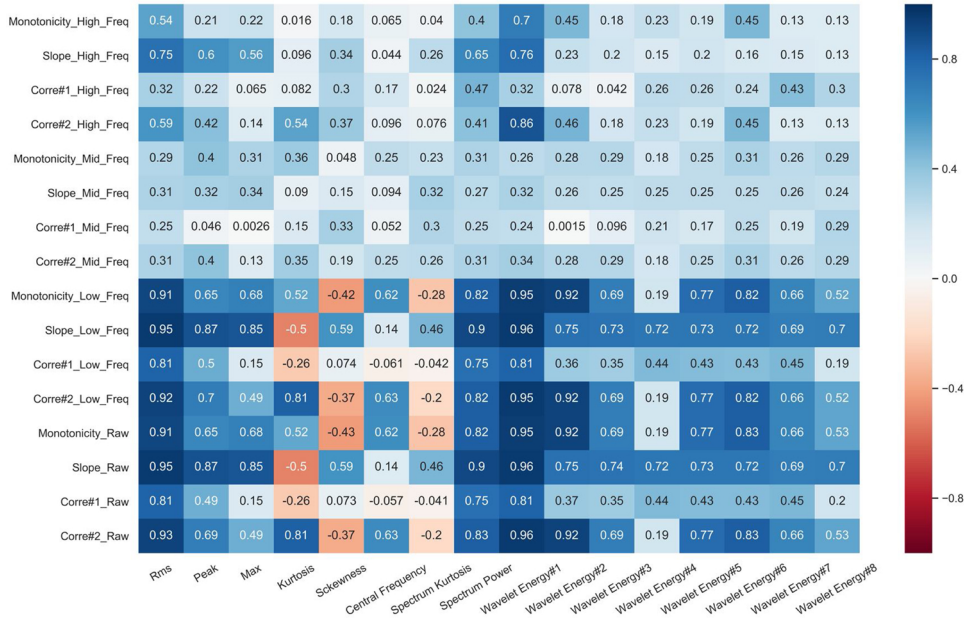


Fig. 9. Feature score according to the selection criteria.

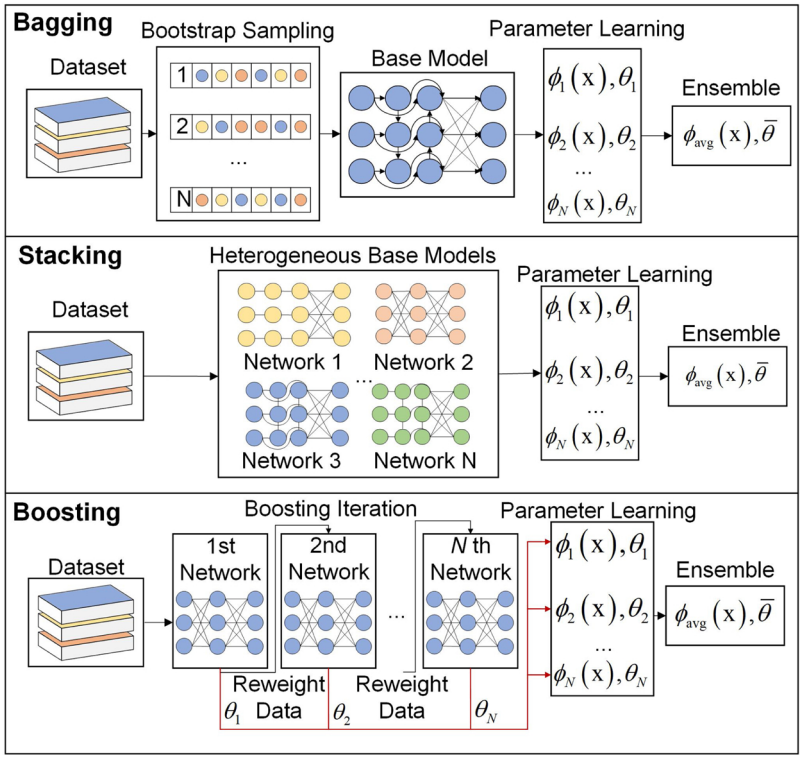


Fig. 10. Frameworks of three ensemble learning methods.

Table 4
Detailed information of different ensemble techniques.

Ensemble technique	Base model
Stacking	LSTM-GRU-DNN
	SVR-Random forest-Ridge regression
Boosting	AdaBoost
Bagging	GRU
	ANN
MC-Dropout	Proposed MC-GRU

estimations. The computational advantage is attributed to the running strategy of proposed MC-GRU. During the testing stage, partial neural nodes of the trained GRU network would be changed multiple times to achieve the same performance as building different structured models. Therefore, the ensemble process in MC-GRU is conducted on one trained model with multiple times, which could save much running time compared with loading multiple models for ensemble.

Based on the above comparison results, it can be concluded that the proposed ensemble MC-GRU network outperforms other ensemble techniques. It could provide interval estimations and save computational time without sacrificing model accuracy.

4.4.2. Discussions on different parameters of MC-GRU

In this sub-section, two indexes, prediction interval coverage probability (PICP) and mean prediction interval width (MPIW) [40], are introduced to comprehensively and quantitively evaluate the model robustness of MC-GRU.

The index PICP represents the correctness of a model, which is measured by counting the number of real target values covered by the estimate prediction intervals, the equation of PICP can be expressed as:

$$PICP = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} c_i,\ c_i = \begin{cases} 1,\ t_i \in [L_i,\ U_i] \\ 0,\ t_i \notin [L_i,\ U_i] \end{cases} \tag{30}$$

where n_{test} is the number of samples in the test dataset, and L_i and U_i are lower and upper bound of the ith prediction interval respectively. Ideally, PICP should be very close or larger than the nominal confidence level associated to the PIs.

Another evaluation index MPIW is constructed based on the width of prediction interval, which could evaluate the informativeness of a model. The expression of MPIW could be expressed as:

$$MPIW = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (U_i - L_i) \tag{31}$$

MPIW represents the average width of the prediction intervals, and a large MPIW may be not acceptable in practice as the variation in testing data could not be recognized.

Based on the aforementioned indexes, a series of comparative experiments of MC-GRU with different model parameters are conducted. At the same time, the running time of each case is also recorded to investigate the computational efficiency. The evaluation results under different ensemble and iteration times are given in Table 6.

It can be seen that the model with 1000 iteration times and 500 ensemble numbers shows the best performance, which has a relatively narrow prediction interval width and the PICP is close to the ideal PI 95 %. The model with 5000 iteration times seems to be overfitted since a few target values are not covered in the estimated confidence intervals. Noticed that the model real-time feasibility is mainly influenced by the ensemble times, and the running time of MC-GRU will increase linearly with the ensemble times. Since the improvement of PICP between model with 100 times ensemble and 500 times ensemble is not obvious, the ensemble number is set as 100 for better real-time feasibility.

4.4.3. Evaluation the performance on RUL prediction

After building the measurement model with proposed MC-GRU network, two types of predictions over different time scales have been conducted:

1) Short-term prediction: The first 450 h data are selected as the available information, and the next 100 h data are set to be updated.

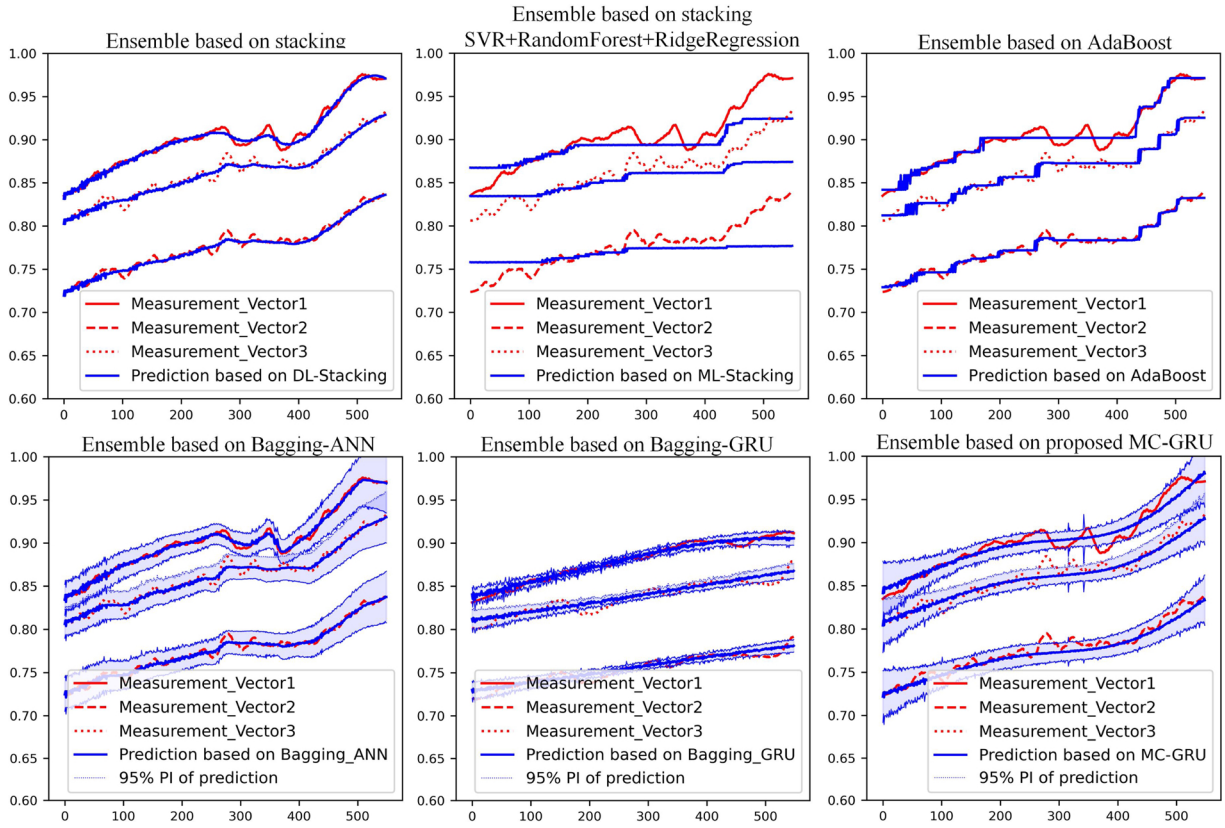


Fig. 11. Performance comparison of different ensemble techniques.

2) Long-term prediction: The first 250 h data are selected as the available information, and the next 300 h data are set to be updated.

Considering the real-time performance of the algorithm, the particle numbers is set as 100 and the ensemble times is set as 100. And to make comprehensive comparisons, the LSTM-PF [31] and Autoregressive particle filters (AR-PF) [25] are both tested for short-term and long-term predictions. The experimental results based on these three models for short-term and long-term prognosis of ball screws are given in Fig. 12.

From Fig.12, it can be seen that the LSTM-PF prunes to achieve the failure threshold prematurely both for short-term and long-term predictions, and the AR-PF performs well for short-term prediction but shows delay for long-term predictions. Compared with these two integrated models, the proposed model shows superiority on prediction accuracy on different time scales. In order to further explain what makes MC-GRU-PF outperform the above models in prediction accuracy, the same batch particles with random initial values are fed into these models for long-term predictions, and the measurement equations and updated particle weights are shown as Fig.13.

From Fig.13, it can be seen that both AR-PF and LSTM-PF provide a wider confidence interval compared with MC-GRU-PF. They only employ the measurement noise as the uncertainty of the measurement model, so the confidence interval will be too wide if the noise level is high. Noticed that the too wide intervals could make the measurement model lack of informativeness, because the particles with different initial values will be assigned approximate weights based on the likelihood function with excessive variance. For MC-GRU-PF model, the model uncertainty has been taken into consideration and the fused interval is narrower compared with the interval only constructed by measurement noise. The added model uncertainty could govern the width of confidence interval and suppress the influence caused by excessive noise. Therefore, the particles closer to the true state would be assigned higher weights rather than same weights, which could

Table 5
Evaluation comparison of different ensemble techniques.

Method	Measurement 1	Measurement 2	Measurement 3	Running time (second)	Prediction type
DL-Stacking	RMSE:0.00405 MAE:0.00325	RMSE:0.00319 MAE:0.00252	RMSE:0.00481 MAE:0.00378	30.83	Point estimation
ML-Stacking	RMSE:0.0204 MAE:0.0148	RMSE:0.0182 MAE:0.0130	RMSE:0.0198 MAE:0.0143	12.64	Point estimation
AdaBoost	RMSE:0.00542 MAE:0.00427	RMSE:0.00339 MAE:0.00269	RMSE:0.00547 MAE:0.00437	0.462	Point estimation
ANN-Bagging	RMSE:0.00324 MAE:0.00255	RMSE:0.00286 MAE:0.00230	RMSE:0.00416 MAE:0.00334	113	Interval estimation
GRU-Bagging	RMSE:0.00391 MAE:0.00315	RMSE:0.00418 MAE:0.0033	RMSE:0.00625 MAE:0.00490	518	Interval estimation
MC-GRU	RMSE:0.00571 MAE:0.00460	RMSE:0.00357 MAE:0.00275	RMSE:0.00574 MAE:0.00452	1.71	Interval estimation

Table 6
Performance evaluation of MC-GRU with different ensemble parameters.

Ensemble	10	100	500	10	100	500	10	100	500
Iteration time	200	200	200	1000	1000	1000	5000	5000	5000
PICP	0.903	0.978	0.990	0.842	0.930	0.943	0.736	0.797	0.803
MPIW	0.050	0.0539	0.0573	0.0246	0.0344	0.0352	0.0114	0.00822	0.00849
Running time	0.41s	3.8 s	20.2 s	0.39 s	3.7 s	19.9 s	1.7 s	3.94 s	19.3 s

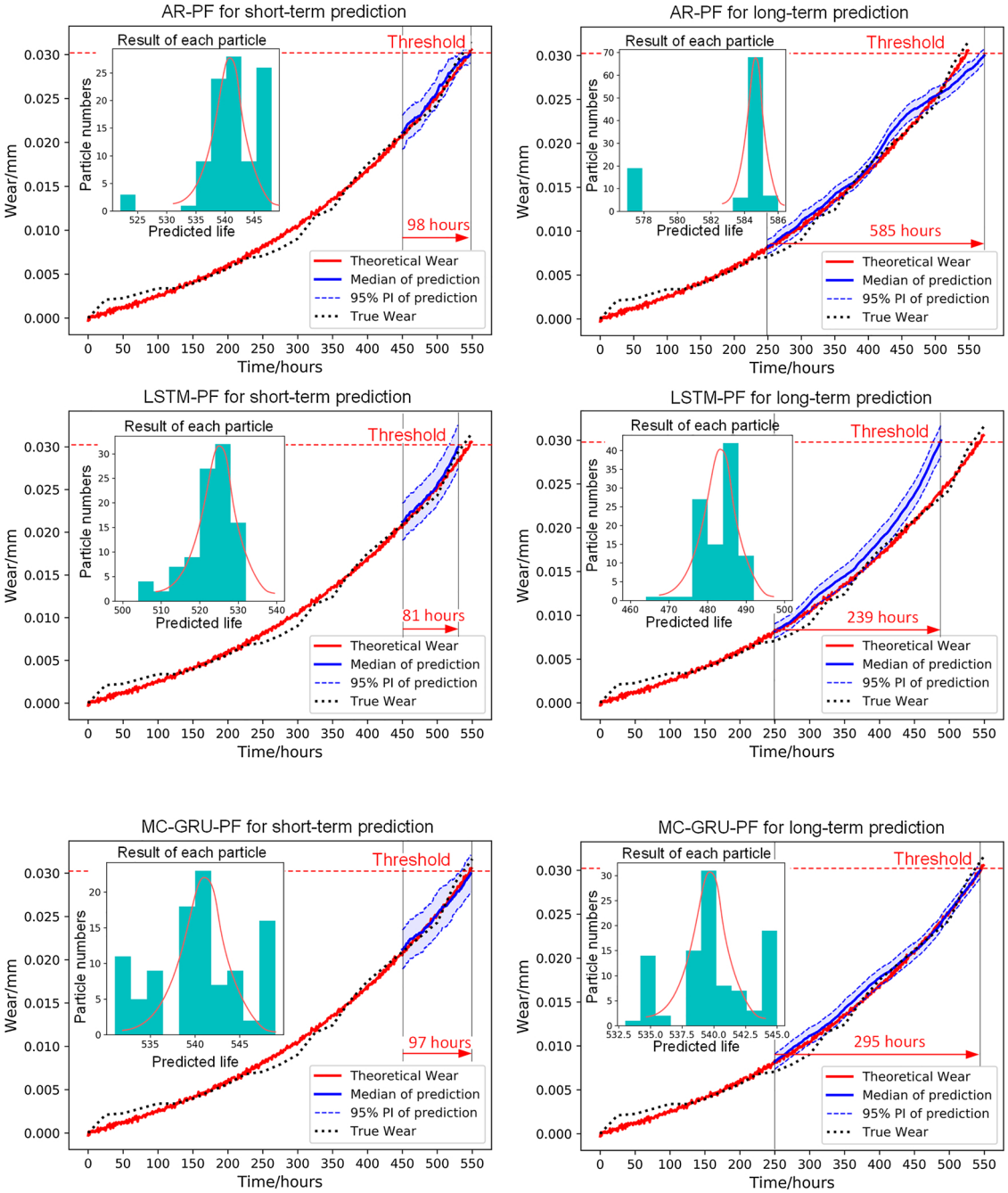


Fig. 12. Comparative results for different time scales prediction.

enhance the updating process in particle filters.

5. Conclusions

Particle filters model is often adopted as a typical prognostic technique for estimating the degradation state and predicting the RUL of a system. For the ball screw prognostic issue, the measurement relationship between sensor signal and wear state could not be formulated in an analytical form, which limits the application of particle filters in ball screw prognostic investigation. In this paper, a hybrid model called as MC-GRU-PF is designed to address this problem, and a ball screw ADT experiment has been conducted to validate the

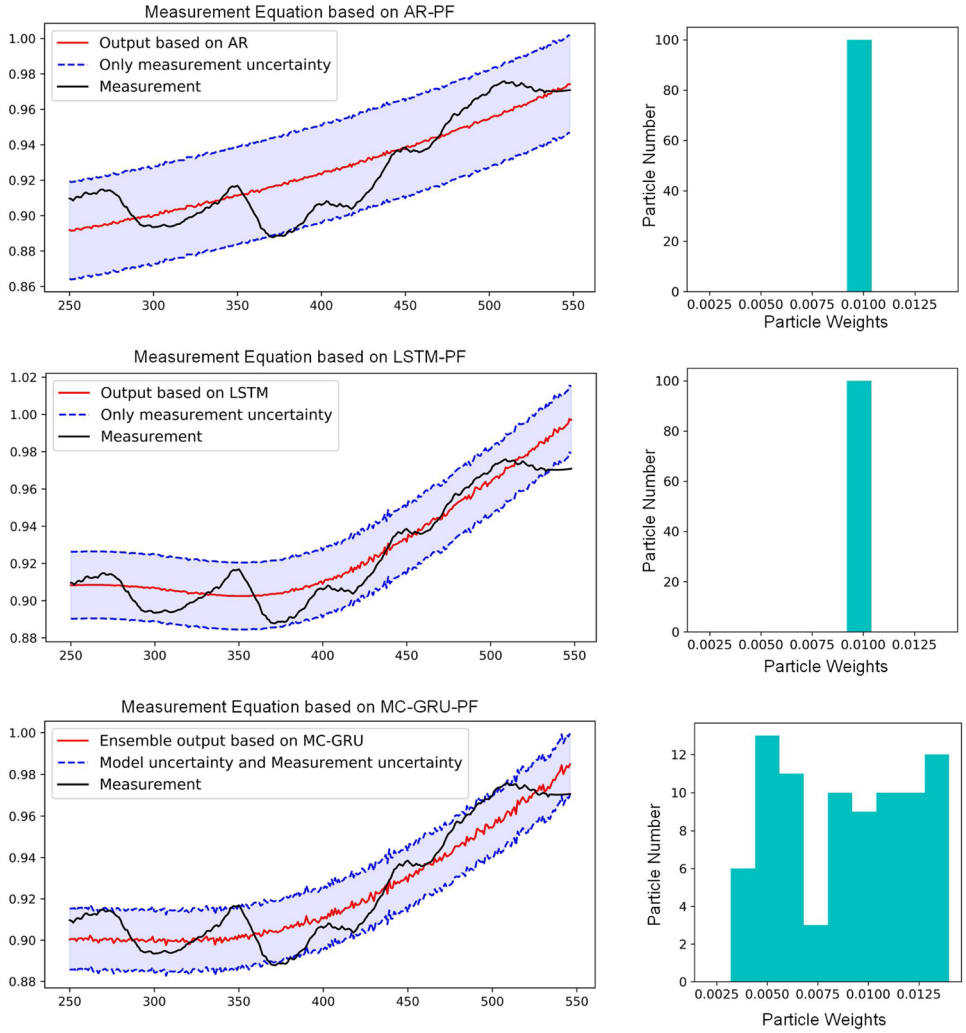


Fig. 13. Measurement equations for long-term prediction and particle weights.

performance. The conclusions are drawn as follows:

1) The proposed state function could reflect the ball screw degradation process with constant working speed. The main influence factor of the ball screw wear is the lubrication state.
2) Compared with the original GRU network, proposed MC-GRU network could enhance the measurement model from two aspects. First, the MC-GRU could conduct the ensemble process within one model to improve the prediction stability. Secondly, it could provide interval estimations instead of point estimations based. The prediction interval provided by MC-GRU could be employed as the model uncertainty to supplement the construction of the measurement distribution $p(\boldsymbol{z}\ |\boldsymbol{x})$, which solves the problem that only measurement noise is considered when building the hybrid particle filter framework.
3) In order to verify the performance of MC-GRU network on real-time feasibility, model prediction accuracy and uncertainty quantification quality, different ensemble techniques including stacking, boosting and bagging are investigated and the performance of MC-GRU under different parameters are also evaluated. The results show that proposed ensemble model could not only save computational time without sacrificing prediction accuracy compared with other ensemble techniques, but also provide reliable prediction interval estimations.
4) The proposed MC-GRU-PF framework shows superiority for RUL prediction on different time scales compared with the other hybrid PF models based on the comparative experimental results. The fused uncertainty quantification could govern the confidence interval and ensure the informativeness of the measurement model. Thus, the particle weights close to true state would be assigned more accurately during the PF update process and the prediction accuracy could be enhanced accordingly.

For future work, a variety of experiments under different ball screw operating conditions (changing load or rotary speed) will be investigated to evaluate performance of the hybrid model. The feasibility of applying the hybrid prognostic model into similar scenarios where the analytical measurement equation is not available will be also explored.

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.

Acknowledgements

This research acknowledges the financial support provided by NSFC51775343, the authors are also thankful for the finance support by Chinese Institute for Quality Research.

References

[1] Feng Guo-Hua, Pan Yi-Lu. Investigation of ball screw preload variation based on dynamic modeling of a preload adjustable feed-drive system and spectrum analysis of ball-nuts sensed vibration signals. Int J Mach Tools Manuf 2012;52(1):85–96.
[2] Han Chang-Fu, et al. Techniques developed for fault diagnosis of long-range running ball screw drive machine to evaluate lubrication condition. Measurement 2018;126:274–88.
[3] Nguyen Tung Lam, Ro Seung-Kook, Park Jong-Kweon. Study of ball screw system preload monitoring during operation based on the motor current and screw-nut vibration. Mech Syst Signal Process 2019;131:18–32.
[4] Li Pin, et al. Prognosability study of ball screw degradation using systematic methodology. Mech Syst Signal Process 2018;109:45–57.
[5] Jin Wenjing, Chen Yan, Lee Jay. Methodology for ball screw component health assessment and failure analysis. ASME 2013 International Manufacturing Science and Engineering Conference Collocated With the 41st North American Manufacturing Research Conference. 2013.
[6] Tsai PC, Cheng CC, Hwang YC. Ball screw preload loss detection using ball pass frequency. Mech Syst Signal Process 2014;48(1–2):77–91.
[7] Chin-Chung Wei, Liou Wei-Lun, Lai Ruei-Syuan. Wear analysis of the offset type preloaded ball–screw operating at high speed. Wear 2012;292:111–23.
[8] Ren Lei, et al. Bearing remaining useful life prediction based on deep autoencoder and deep neural networks. J Manuf Syst 2018:71–7.
[9] Zhang Yuyan, et al. Imbalanced data fault diagnosis of rotating machinery using synthetic oversampling and feature learning. J Manuf Syst 2018:34–50.
[10] Wu Dazhong, et al. A fog computing-based framework for process monitoring and prognosis in cyber-manufacturing. J Manuf Syst 2017:25–34.
[11] Chang Jun-Liang, et al. Prognostic experiment for ball screw preload loss of machine tool through the hilbert-huang transform and multiscale entropy method. The 2010 IEEE International Conference on Information and Automation. 2010.
[12] Wen Juan, et al. A new method for identifying the ball screw degradation level based on the multiple classifier system. Measurement 2018;130:118–27.
[13] Wang Jinjiang, et al. Deep learning for smart manufacturing: methods and applications. J Manuf Syst 2018:144–56.
[14] Adams Stephen, et al. The WEAR methodology for prognostics and health management implementation in manufacturing. J Manuf Syst 2017:82–96.
[15] Jin Wenjing. Modeling of machine life using accelerated prognostics and health management (APHM) and enhanced deep learning methodology. Diss. University of Cincinnati; 2016.
[16] Zhang Li, et al. A deep learning-based recognition method for degradation monitoring of ball screw with multi-sensor data fusion. Microelectron Reliab 2017;75:215–22.
[17] Hong Xin, et al. Ball screw stability degradation stages evaluation based on deep residual neural network and multi-sensor fusion. 2018 Prognostics and System Health Management Conference (PHM-Chongqing). 2018.
[18] Cadini Francesco, et al. State-of-life prognosis and diagnosis of lithium-ion batteries by data-driven particle filters. Appl Energy 2019:661–72.
[19] Zhao Fuqiong, Tian Zhigang, Zeng Yong. Uncertainty quantification in gear remaining useful life prediction through an integrated prognostics method. Ieee Trans Reliab 2013;62(1):146–59.
[20] Zhao Fuqiong, et al. An integrated prognostics method for failure time prediction of gears subject to the surface wear failure mode. Ieee Trans Reliab 2018;67(1):316–27.
[21] Yu Tianyu, Li Zhixiong, Wu Dazhong. Predictive modeling of material removal rate in chemical mechanical planarization with physics-informed machine learning. Wear 2019:1430–8.
[22] Deng Yingjun, Di Bucchianico Alessandro, Pechenizkiy Mykola. Controlling the accuracy and uncertainty trade-off in RUL prediction with a surrogate Wiener propagation model. Reliability engineering & system safety. 2020.
[23] Chang Yang, Fang Huajing, Zhang Yong. A new hybrid method for the prediction of the remaining useful life of a lithium-ion battery. Appl Energy 2017:1564–78.
[24] Saha Bhaskar, et al. Prognostics methods for battery health monitoring using a bayesian framework. IEEE Trans Instrum Meas 2009;58(2):291–6.
[25] Wang Jinjiang, Wang Peng, Gao Robert X. Enhanced particle filter for tool wear prediction. J Manuf Syst 2015:35–45.
[26] Wang Jinjiang, et al. A virtual sensing based augmented particle filter for tool condition prognosis. J Manuf Process 2017:472–8.
[27] Wang Peng, Gao Robert X. Adaptive resampling-based particle filtering for tool life prediction. J Manuf Syst 2015:528–34.
[28] Zhang Jianlei, et al. Particle learning in online tool wear diagnosis and prognosis. J Manuf Process 2017:457–63.
[29] Jha Mayank Shekhar, Dauphintanguy G, Ouldbouamama Belkacem. Particle filter based hybrid prognostics for health monitoring of uncertain systems in bond graph framework. Mech Syst Signal Process 2016:301–29.
[30] Daroogheh Najmeh, et al. Prognosis and health monitoring of nonlinear systems using a hybrid scheme through integration of PFs and neural networks. systems man and cybernetics 2017:1990–2004.
[31] Huang Chenggeng, et al. An enhanced deep learning-based fusion prognostic method for RUL prediction. Ieee Trans Reliab 2019:1–13.
[32] Baraldi Piero, et al. Ensemble neural network-based particle filtering for prognostics. Mech Syst Signal Process 2013;41(1):288–300.
[33] Jouin Marine, et al. Particle filter-based prognostics: review, discussion and perspectives. Mech Syst Signal Process 2016;72:2–31.
[34] Zhao Rui, et al. Machine health monitoring using local feature-based gated recurrent unit networks. Ieee Trans Ind Electron 2017;65(2):1539–48.
[35] Cho Kyunghyun, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 2014.
[36] Gal Yarin, Ghahramani Zoubin. Dropout as a bayesian approximation: representing model uncertainty in deep learning. International Conference on Machine Learning 2016.
[37] Cheng Qiang, et al. An accuracy degradation analysis of ball screw mechanism considering time-varying motion and loading working conditions. Mech Mach Theory 2019;134:1–23.
[38] Zhou Chang-Guang, et al. Investigation of the precision loss for ball screw raceway based on the modified Archard theory. Ind Lubr Tribol 2017;69(2):166–73.
[39] Tao Weijun, et al. Model for wear prediction of roller linear guides. Wear 2013;305(1-2):260–6.
[40] Khosravi Abbas, et al. Comprehensive review of neural network-based prediction intervals and new advances. IEEE Trans Neural Netw 2011;22(9):1341–56.