Failure Prognosis with Uncertain Estimation Based on Recursive Models Re-sampling Bootstrap and ANFIS

Li Dongping

Abstract—**Effective prognostic tools are crucial for maintenance to predict failure before system completely damage and ensure systems reliability. Under the condition that the only available information is degradation observations from the forecasted system of interest, to perform fault prediction with uncertainty quantification, a recursive models re-sampling bootstrap (RMRB) associated with adaptive neural fuzzy inference system (ANFIS) predictor is presented. In addition with point prediction, the proposed RMRB could provide prediction interval (PI), thus reduce risks of misleading decision to maintenance. Faulty feature sensitive to system degradation is selected to compose time-series. Then the future time instance faulty feature prediction and its PI are provided by RMRB in two steps: firstly, a set of ANFIS predictors are updated dynamically by hybrid learning algorithm of recursive least squares and gradient descent (RLS-GD) according to temporal order; secondly, the candidate ANFIS predictors are re-sampled as bootstrap replications to estimate prediction mean and prediction interval. Case studies based on two real failure datasets show that the proposed approach can effectively estimate uncertainties coupled with fault pre-diction results.**

Key words—**fault prediction; prediction interval; adaptive neural fuzzy inference system (ANFIS); bootstrap; time-series**

I. Introduction

With an increasing demand to maintain system high reliability, safety and availability, besides effective fault detection and isolation when a failure has already occurred, it is usually required to predict system degradation trend before malfunctions or even catastrophic failures happen. As a result, failure prognosis has attracted considerable attention in maintenance and indemnification, especially in avionics [1], aircraft [2], machine [3] and manufacture industry [4] fields.

The existing fault prediction algorithms can be broadly categorized into two classes: analytical model based approaches and non-analytical model based approaches [5]. Given a proper mathematic model for representing the specific forecasted system, analytical model based algorithms tend to provide high prediction accuracy. Unfortunately, it is difficult to derive accurate mathematic model in some complex nonlinear systems, and this limits the application of analytical model based methods. In contrast, non-analytical model based methods have been more popular in fault prediction for their free of system mathematic functions. Data-driven based approaches, as an important part of non-analytical model approaches, employ data sensitive to system degradation to construct fault prediction model, which can be performed more easily compared with analytical model based methods. A variety of data-driven techniques have been proposed in the literature, including auto-regressive moving-average [6], grey model [7], hidden Markov model [4], bayesian networks [8], neural networks [9, 10], and support vector machine [11] etc.

Commonly data-driven methods are performed as point forecast without any indication of prediction accuracy. As there inherently exist some sources of uncertainties in data-driven predictors (such as the error between true system and data-driven model, measurement noise, dynamic process noise), it is more important to predict failure with quantitative uncertainty information. Such information can provide the estimation of faulty system best and worst degradation status, i.e., to what extents the predicted results could be trusted, and reduce risks of misleading decision to maintenance. Clearly, prediction interval (PI), composed of upper and lower bounds that contain a future unknown value with a prescribed probability called a confidence level $(1\text{-}\alpha)\%$ [12], offers us a convenient way to indicate the uncertainty associated with point prediction. PI estimation algorithms include but not limit to, statistical method [13], mean-variance estimation [14], delta[15], bayesian [16], and bootstrap [17, 18] etc.

Among the existing PI algorithms, bootstrap has become a popular method for its advantages in simplicity for constructing PI and independence of specified predicting models. Malhotra et al. [19] apply bootstrap for fault prediction of software quality. In some references, bootstrap is utilized to train neural networks for construct PI [20, 21]. Baraldi et al. [22] use bootstrap to predict turbine blade RUL (remaining useful life) with PI estimation. Noticeably, the aforementioned bootstrap algorithm is based on the assumption that data is independent and identically distributed (IID). However, time-series methods, that faulty feature indicating system degradation is selected to compose temporal sequence to be predicted, can not be assumed to be IID for its time dependence relationship [23]. Consequently, the ordinary bootstrap method would distort the underlying temporal dependence structure of the time-series, since it is incapable of preserving the serial correlation among data. To avoid this pitfall, some improved bootstrap techniques are proposed, such as moving block bootstrap (MBB) [24], models re-sampling bootstrap (MRB) [25], etc. Application

Manuscript received Jan 2, 2016; revised Apr 10, 2016.

Li Dongping is with institute of Information Science and Technology, Kunming University, 650214, China (e-mail: ynkm1971@sina.com).

of MBB has been documented in the literature among the fields of spacecraft telemetry data prognosis [26], electricity price prediction [27], power generation forecasts [28], etc. Whereas MBB is primarily suitable for specified prediction horizon H, which is due to its underlying strategy of re-sampling directly from the original time-series data[29]. Thus for another specified prediction horizon, new prediction models should be constructed to estimate PI for MBB. In contrast, MRB, instead of re-sampling directly from the time-series data, selecting bootstrap replications from the set of candidate data-driven prediction models, is befitting to estimate PIs online, i.e., calculating 1 to H steps-ahead PIs by iterating a one-step-ahead predictor without training different time horizon predicting model.

However, MRB is performed under the assumption that a set of observations similar to the one to be predicted is available. With regard to fault prediction in the actual system, it is hard to obtain a whole life cycle failure degradation observation sequences from other similar systems, especial for long life cycle and high reliability systems, i.e., the information available is just the degradation observations up to the current time of the predicted system. In this condition, the existing MRB with insufficient dataset would offer low quality PI. To provide effective PI with the limited available dataset, a recursive models re-sampling bootstrap (RMRB) algorithm is proposed based on the theory framework of MRB. With new coming observation according to temporal order, RMRB dynamically updates candidate data-driven predictors used for bootstrap replicating, which could adapt system failure changing trends timely without disturbing the time dependence structure of the data. In this paper, adaptive neural fuzzy inference system (ANFIS), using hybrid learning algorithm of recursive least squares and gradient descent (RLS-GD), is applied for building data-driven predictor to forecast time-series composed of faulty features. Then the proposed RMRB is utilized to quantify uncertainties associated with the predicted faulty features. A confidence level of (1-α)% is considered for constructing faulty feature PI. Experiments are conducted with two real failure datasets. The validation of the constructed PI is demonstrated from semi-quantitative analyzing and comparison of different PI estimating methods.

The rest of this paper is organized as follows. The construction of PI and models re-sampling based bootstrap are briefly introduced in Section 2. Section 3 presents the recursive models re-sampling bootstrap algorithm. The fault prediction approach based on the proposed RMSB is described in Section 4. In Section 5, we provide the experiment cases to verify the performance of the proposed method with two real failure datasets. Finally, the conclusive remarks are drawn in Section 6.

II. PROBLEM STATEMENT

A. Prediction interval

Consider the following time-series dynamic model

$$t_k = f_k(\boldsymbol{X}_k, \boldsymbol{W}_k) + \varepsilon_k \tag{1}$$

where t_k is measured target at k time instance, $f_k(\boldsymbol{X}_k, \boldsymbol{W}_k)$ is true regression model denoting mean of target distribution given input vector $\boldsymbol{X}_k$ and model parameters $\boldsymbol{W}_k$, ε_k is gaussian noise with zero mean and covariance $\sigma^2_{\varepsilon_k}$. To time-series prediction, $t_k = y_{k+H}$, $\boldsymbol{X}_k = [y_k, y_{k-H}, y_{k-2H}, \cdots, y_{k-(m-1)H}]$, here H is prediction horizon, m is embedding dimension and y_k is time-series variable at k time instance.

Actually, only the approximate model $\hat{f}_k(\boldsymbol{X}_k, \hat{\boldsymbol{W}}_k)$ toward system true model $f_k(\boldsymbol{X}_k, \boldsymbol{W}_k)$ can be constructed, where $\hat{\boldsymbol{W}}_k$ is model parameters for $\hat{f}_k(\boldsymbol{X}_k, \hat{\boldsymbol{W}}_k)$. Suppose $f_k(\cdot)$ and $\hat{f}_k(\cdot)$ to be the short form of $f_k(\boldsymbol{X}_k, \boldsymbol{W}_k)$ and $\hat{f}_k(\boldsymbol{X}_k, \hat{\boldsymbol{W}}_k)$ respectively, then we obtain,

$$t_k - \hat{f}_k(\cdot) = [f_k(\cdot) - \hat{f}_k(\cdot)] + \varepsilon_k \tag{2}$$

Assuming that $f_k(\cdot) - \hat{f}_k(\cdot)$ and ε_k are statistically independent, prediction error variance $\sigma^2_{p_k}$ of $t_k - \hat{f}_k(\cdot)$ can be decomposed into two terms,

$$\begin{aligned}\sigma^2_{p_k} &= E\left\{[t_k - \hat{f}_k(\cdot)]^2\right\} \\ &= E\left\{[f_k(\cdot) - \hat{f}_k(\cdot)]^2\right\} + E\left\{[t_k - f_k(\cdot)]^2\right\} \\ &= \sigma^2_{m_k} + \sigma^2_{\varepsilon_k}\end{aligned} \tag{3}$$

where $\sigma^2_{m_k}$ is variance between predicted regression model $\hat{f}_k(\cdot)$ and true regression model $f_k(\cdot)$, while $\sigma^2_{\varepsilon_k}$ denotes a noise term ε_k. The regression model uncertainty calculated by $\sigma^2_{m_k}$ and target uncertainty calculated by $\sigma^2_{p_k}$ are called confidence interval (CI) and prediction interval (PI) separately, as shown in Eq. (4) and Eq. (5).

$$\hat{f}_k(\cdot) - c_{CI}\sigma^2_{m_k} \le f_k(\cdot) \le \hat{f}_k(\cdot) + c_{CI}\sigma^2_{m_k} \tag{4}$$

$$\hat{f}_k(\cdot) - c_{PI}\sigma^2_{p_k} \le t_k \le \hat{f}_k(\cdot) + c_{PI}\sigma^2_{p_k} \tag{5}$$

where c_{CI} and c_{PI} represent the factor of CI and PI respectively. CI is corresponding to the accuracy of predicted regression model $f_k(\cdot)$, i.e., of the estimation of the probability $P(f_k(\cdot) \mid \hat{f}_k(\cdot))$. In contrast, PI deals with the accuracy of measured target t_k with the predicted value $\hat{f}_k(\cdot)$, i.e., the estimation of the probability $P(t_k \mid \hat{f}_k(\cdot))$. Obviously, PI should be wider than CI and enclose it. And prediction interval is preferable since it covers more sources of uncertainties and gives more accurate information about unknown measured target value.

B. Models re-sampling based bootstrap

Bootstrap, as a popular algorithm to estimate PI, is essentially a re-sampling method for calculating the distribution of the predicted targets t_k [17]. It regenerates a mass of samples via re-sampling the available observed dataset with replacement. Then the samples can be utilized to construct confidence interval (CI) and prediction interval (PI) for t_k. The conventional bootstrap algorithm is based on the assumption that data is independent and identically distributed (IID). However, time-series data can not be assumed to be IID, thus the conventional bootstrap method would distort time-series underlying temporal dependence structure for its incapacity to preserve the serial correlation among data. To avoid this pitfall, some improved bootstrap techniques are proposed. Of the bootstrap algorithms applied in time-series prediction, models re-sampling bootstrap (MRB) [25] has emerged as a popular method due to its facility in estimating PIs online, i.e., calculating 1 to H steps-ahead PIs by iterating a one-step-ahead predictor without training different time horizon predicting models. Unlike other bootstrap algorithms of re-sampling directly from time-series dataset, MRB selects samples from the set of candidate data-driven models, where each candidate model is built based on part of the original time-series dataset to predict target value at future time instance. The detailed re-sampling algorithm for MRB is described as [25].

Step 1 Generate n model samples $\{M^j\}_{j=1}^n$ for bootstrap re-sampling. Here M^j is one-step-ahead predictor trained according to time-series data of the predicted system, and M^j can be constructed based on neural network, support vector machine, etc.

1-a Define $j = 1$;

1-b Select two arbitrary integers N_j and t_j from the following equations,

$$N_j \sim U(S,T) \tag{6}$$

$$t_j \sim U(0, T-N_j) \tag{7}$$

where $U(\cdot)$ operator represents discrete uniform distribution, N_j denotes the length of data interval and t_j defines the starting time instance of training dataset, S and T is the minimum and the entire length of data to train model M^j separately;

1-c The jth model M^j is trained based on the time-series dataset $\{y_i\}_{i=t_j}^{t_j+N_j-1}$;

1-d $j = j + 1$, if $j > n$, go to step 2, else go to step 1-b;

Step 2 Obtain B bootstrap replications from the n trained model samples $\{M^j\}_{j=1}^n$.

2-a Set $b = 1$;

2-b Re-sample the model samples set $\{M^j\}_{j=1}^n$ n times with replacement. Then we obtain the bth bootstrap estimate $(y_b^*)_{k+H}$

$$(y_b^*)_{k+H} = \frac{1}{n}\sum_{j=1}^{n}(o_j^*)_{k+H} \tag{8}$$

where $(o_j^*)_{k+H}$ is the predicted output of the jth model M^j at $k+H$ time instance, k is the current time instance and H is prediction horizon. For multiple steps prediction, $(o_j^*)_{k+H}$ is calculated by iterating a one-step-ahead prediction $(o_j^*)_{k+1}$ H times.

2-c $b = b + 1$, if $b > B$, go to step 3, else go to step 2-b;

Step 3 Compute target prediction mean and PI [22] based on the $(y_b^*)_{k+H}$ obtained from step 2, $b = 1, 2,\ldots, B$.

III. RECURSIVE MODELS RE-SAMPLING BOOTSTRAP

The current MRB algorithm for PI estimation described in section 2.2 is processed mainly under the assumption that historical time-series data set $\{y_{1:N_q}^q\}_{q=1}^Q$ related to the degradation process of Q similar failed systems are available, whereas this assumption may not be satisfied in most actual system, namely the available system degradation sequence $\boldsymbol{y}_{1:k}$ is only k observations $\{y_i\}_{i=1}^k$ up to the current time instance k. In this case, ordinary MRB method, re-sampling the n model samples $\{M^j\}_{j=1}^n$ trained with insufficient amount of data set, may provide low-quality PIs, as the poor model samples are not fully representative of the "true" distribution of time-series prediction. In addition, N_j length training data is acquired simultaneously to construct bootstrap re-sampling model M^j, as described in section 2.2, which has two drawbacks: firstly, the training result is global optimization among N_j observations, yet good fitting accuracy does not guarantee an equally good prediction capability; secondly, constructing model M^j with N_j training data simultaneously is time consuming.

To overcome the aforementioned shortcomings, a recursive models re-sampling bootstrap (RMRB) algorithm is proposed under the theory framework of MRB. For each time instance PI estimation, instead of using the latest N_j training data $\{y_i\}_{i=k-N_j+1}^k$, the n model samples $\{M^j\}_{j=1}^n$ are updated with the latest one observation y_k based on recursive learning techniques, such as recursive least squares, gradient descent, bayesian and kalman filter, etc. Here adaptive neural fuzzy inference system (ANFIS) is chosen as the data-driven method to build n predicting models $\{M^j\}_{j=1}^n$. Further information and mathematical expression about ANFIS can be found in reference [30]. Hybrid recursive learning techniques [31-33] of recursive least squares and gradient descent (RLS-GD) is utilized to train ANFIS, where ANFIS consequent parameters are identified by recursive least squares and premise parameters are updated by gradient descent[34]. Detailed algorithm for RMRB is as the following steps.

(1) Initialize model samples.

Initialize n model samples $\{M_0^j\}_{j=1}^n$ with the beginning T length time-series observations $\{y_i\}_{i=1}^T$, where each model M_0^j is trained through a child data set $\{y_i\}_{i=t_j}^{t_j+N_j-1}$, N_j and t_j is selected according to Eq. (6) and Eq. (7). M_0^j built based on ANFIS is shown in Eq. (9). At last, set $k = 1$;

$$\hat{y}_{i+1} = \mathrm{ANFIS}(y_{i-m+1},\cdots,y_{i-1},y_i;\hat{\boldsymbol{W}}_i) \tag{9}$$

where $\hat{\boldsymbol{W}}_i = [\hat{\boldsymbol{W}}_i^{\mathrm{con}}\ \hat{\boldsymbol{W}}_i^{\mathrm{pre}}]$, $\hat{\boldsymbol{W}}_i^{\mathrm{con}}$ and $\hat{\boldsymbol{W}}_i^{\mathrm{pre}}$ are ANFIS consequent parameters and premise parameters separately.

(2) Generate models re-sampling based bootstrap replications.

① Update the ANFIS model M_k^j parameters $\hat{\boldsymbol{W}}_k^j = [(\hat{\boldsymbol{W}}_k^j)^{\mathrm{con}}\ (\hat{\boldsymbol{W}}_k^j)^{\mathrm{pre}}]$ with (t_j+N_j-1+k) time instance observation $y_{t_j+N_j-1+k}$ and previous model M_{k-1}^j parameters $\hat{\boldsymbol{W}}_{k-1}^j$ by RLS-GD as follows, where $j = 1, 2, \ldots, n$.

Consequent parameters $(\hat{\boldsymbol{W}}_k^j)^{\mathrm{con}}$ are identified by recursive least squares (RLS) when taking premise parameters $(\hat{\boldsymbol{W}}_k^j)^{\mathrm{pre}}$ as known ones. Then $(\hat{\boldsymbol{W}}_k^j)^{\mathrm{con}}$ can be updated recursively.

$$(\hat{\boldsymbol{W}}_k^j)^{\mathrm{con}} = (\hat{\boldsymbol{W}}_{k-1}^j)^{\mathrm{con}} + K_k^j[y_{t_j+N_j-1+k} - (\hat{\boldsymbol{W}}_{k-1}^j)^{\mathrm{con}}(\boldsymbol{\Psi}_{k-1}^j)^{\mathrm{T}}] \tag{10}$$

$$K_k^j = \frac{\boldsymbol{P}_{k-1}^j(\boldsymbol{\Psi}_{k-1}^j)^{\mathrm{T}}}{1+\boldsymbol{\Psi}_{k-1}^j\boldsymbol{P}_{k-1}^j(\boldsymbol{\Psi}_{k-1}^j)^{\mathrm{T}}} \tag{11}$$

$$\boldsymbol{P}_k^j = (\boldsymbol{I} - K_k^j \boldsymbol{\Psi}_{k-1}^j)\boldsymbol{P}_{k-1}^j \quad (12)$$

where $\hat{y}_{t_j+N_j-1+k} = (\hat{\boldsymbol{W}}_{k-1}^j)^{\text{con}}(\boldsymbol{\Psi}_{k-1}^j)^{\text{T}}$ is the estimation of true $y_{t_j+N_j-1+k}$ by ANFIS model M_k^j.

Premise parameters $(\hat{\boldsymbol{W}}_k^j)^{\text{pre}}$ are trained via gradient descent (GD) by fixing $(\hat{\boldsymbol{W}}_k^j)^{\text{con}}$.

$$(\hat{w}_k^j)_i^{\text{pre}} = (\hat{w}_{k-1}^j)_i^{\text{pre}} - \beta \frac{\partial E_k^j}{\partial \hat{w}_k^j},\ i = 1, 2, \cdots, N^{\text{pre}} \quad (13)$$

$$E_k^j = \frac{1}{2}[y_{t_j+N_j-1+k} - \text{ANFIS}(\boldsymbol{Y}_{k-1}^j; (\hat{\boldsymbol{W}}_{k-1}^j)^{\text{con}}\ (\hat{\boldsymbol{W}}_{k-1}^j)^{\text{pre}})]^2 \quad (14)$$

$$\boldsymbol{Y}_{k-1}^j = [y_{t_j+N_j-1+k-m}, \cdots, y_{t_j+N_j-1+k-2}, y_{t_j+N_j-1+k-1}] \quad (15)$$

$$(\hat{\boldsymbol{W}}_k^j)^{\text{pre}} = \{(\hat{w}_k^j)_i^{\text{pre}}\}_{i=1}^{N^{\text{pre}}} \quad (16)$$

where $(\hat{w}_k^j)_i^{\text{pre}}$ is one of premise parameters $(\hat{\boldsymbol{W}}_k^j)^{\text{pre}}$, N^{pre} is total number of $(\hat{\boldsymbol{W}}_k^j)^{\text{pre}}$, and β is the learning rate.

② Obtain B bootstrap replications from the n updated model samples $\{M_k^j\}_{j=1}^n$ as described in step 2 of section 2.2. Then the bth bootstrap estimate $(y_b^*)_{k+H}$ is calculated according to Eq. (8), b = 1, 2,…, B, where $(y_b^*)_{k+H}$ is the predicted output of the bth bootstrap replication at $k+H$ time instance, here H is the desired prediction horizon.

(3) Compute the prediction of mean value m_{k+H} and error variance $\sigma_{m_{k+H}}^2$ through the B bootstrap replications.

$$m_{k+H} = \frac{1}{B}\sum_{b=1}^{B}(y_b^*)_{k+H} \quad (17)$$

$$\sigma_{m_{k+H}}^2 = \frac{1}{B-1}\sum_{b=1}^{B}[(y_b^*)_{k+H} - m_{k+H}]^2 \quad (18)$$

where m_{k+H} is the prediction value of true target y_{k+H} at $k+H$ time instance, $\sigma_{m_{k+H}}^2$ corresponds to the first term in the right-hand side of Eq. (3).

(4) Estimate PI at $k+H$ time instance.

To estimate total variance $\sigma_{p_{k+H}}^2$ as shown in Eq. (3), we need to construct a model $\chi^2(\boldsymbol{X})$ that provides an estimation of measurement noise $\sigma_{\varepsilon_{k+H}}^2$ in correspondence of an input X. In general, $\chi^2(\boldsymbol{X})$ can be built based on neural network model. Since the true target of $\chi^2(\boldsymbol{X})$ is not known a priori, $\chi^2(\boldsymbol{X})$ can not be trained directly by minimizing the variance errors between true target and prediction output. As a result, a indirect way is needed to train $\chi^2(\boldsymbol{X})$.

Consider the square of residuals,

$$r^2(\boldsymbol{X}) = \max\left([y(\boldsymbol{X}) - m(\boldsymbol{X})]^2 - \sigma_m^2(\boldsymbol{X}), 0\right) \quad (19)$$

where $m(\boldsymbol{X})$ is the prediction value of true target $y(\boldsymbol{X})$ with input $\boldsymbol{X}$. $m(\boldsymbol{X})$ and $\sigma_m^2(\boldsymbol{X})$ are computed according to Eq. (17) and Eq. (18). Suppose the residuals $r^2(\boldsymbol{X})$ obey a Gaussian distribution with zero mean, then the variance of residuals $r^2(\boldsymbol{X})$ can be written as,

$$\begin{aligned} E[r^2(\boldsymbol{X})] &= E\{[y(\boldsymbol{X}) - m(\boldsymbol{X})]^2 - \sigma_m^2(\boldsymbol{X})\} \\ &= E\{[y(\boldsymbol{X}) - m(\boldsymbol{X})]^2\} - \sigma_m^2(\boldsymbol{X}) \\ &= \sigma_p^2(\boldsymbol{X}) - \sigma_m^2(\boldsymbol{X}) \\ &= \sigma_\varepsilon^2(\boldsymbol{X}) \end{aligned} \quad (20)$$

Obviously, the variance of residuals $r^2(\boldsymbol{X})$ is equal to $\chi^2(\boldsymbol{X})$. Then a maximum likelihood criterion can be applied to train a neural network model $\chi^2(\boldsymbol{X})$ with the data set of input output pairs $\{\boldsymbol{X}_i, y_i\}_{i=1}^{N'}$ by minimizing cost-function,

$$LL = -\sum_{i=1}^{N'} \log\left(\frac{1}{\sqrt{2\pi\chi^2(\boldsymbol{X}_i)}}\exp\left(-\frac{r^2(\boldsymbol{X}_i)}{2\chi^2(\boldsymbol{X}_i)}\right)\right) \quad (21)$$

According to the above mentioned, estimating PI at $k+H$ time instance can be described as follows,

① Choose the latest $N^{\#}$ time-series data $\{y_i\}_{i=k-N^{\#}+1}^k$ to construct a M length validation data set $\{\boldsymbol{X}_i^{\#}, y_i\}_{i=k-M+1}^k$, where $N^{\#}= m+H+M-1$, $\boldsymbol{X}_i^{\#} = [y_{i-H-m+1}, \cdots, y_{i-H-1}, y_{i-H}]$ is an input vector and y_i is the corresponding H-steps-ahead prediction output.

② Calculate residuals $r^2(\boldsymbol{X}_i^{\#})$ with $\{\boldsymbol{X}_i^{\#}, y_i\}_{i=k-M+1}^k$ through Eq. (19). Then $r^2(\boldsymbol{X}_i^{\#})$ is linked by the set of corresponding input $\boldsymbol{X}_i^{\#}$ to form a new input-output dataset $\{\boldsymbol{X}_i^{\#}, r_i^2\}_{i=k-M+1}^k$.

③ Train a new model $\chi^2(\boldsymbol{X})$ based on dataset $\{\boldsymbol{X}_i^{\#}, r_i^2\}_{i=k-M+1}^k$ via minimizing the log likelihood function

$$LL = -\sum_i \log\left(\frac{1}{\sqrt{2\pi\chi^2(\boldsymbol{X}_i^{\#})}}\exp\left(-\frac{r^2(\boldsymbol{X}_i^{\#})}{2\chi^2(\boldsymbol{X}_i^{\#})}\right)\right) \quad (22)$$

where $\chi^2(\boldsymbol{X})$ is built by a separate neural network with an exponential transfer function for the output layer to keep the $\chi^2(\boldsymbol{X})$ to be positive.

④ Calculate prediction error variance. For input $\boldsymbol{X}_k$, the prediction error variance can be computed as

$$\begin{aligned} \sigma_{p_{k+H}}^2(\boldsymbol{X}_k) &= \sigma_{m_{k+H}}^2(\boldsymbol{X}_k) + \sigma_{\varepsilon_{k+H}}^2(\boldsymbol{X}_k) \\ &= \frac{1}{B-1}\sum_{b=1}^{B}[(y_b^*)_{k+H} - m_{k+H}]^2 + \chi^2(\boldsymbol{X}_k) \end{aligned} \quad (23)$$

⑤ Calculate PIs as follow,

$$m_{k+H} - c_{PI}\sigma_{p_{k+H}}^2 \le PI_{k+H} \le m_{k+H} + c_{PI}\sigma_{p_{k+H}}^2 \quad (24)$$

where the factor c_{PI} is the percentile of a Student's t-distribution with number of degrees of freedom equal to the number of bootstrap replications B.

(5) k = k + 1, go to step (2) for next time instance PI estimation.

IV. Fault Prediction

Faulty feature y representing faulty system degradation is selected as the time-series variable to be predicted. The fault prediction algorithm flow based on the proposed RMRB is organized as follows (shown in Fig. 1).

Step 1 Initialize parameters of RMRB, ANFIS and RLS-GD;

Step 2 Collect faulty feature y_k at current time instance k to construct time-series data set $\{y_i\}_{i=1}^{k}$, where y_k is obtained directly from instrument measurement or feature extraction method based on system output signal;

Step 3 Calculate prediction value $\hat{y}_{k+H}$ of $k+H$ time instance according to Eq. (17) and prediction error variance $\sigma^2_{p_{k+H}}$ by Eq. (23). Then estimate PI of $\hat{y}_{k+H}$ by Eq. (24), as a result, upper limit $(\hat{y}_{k+H})_{\text{up}}$ and lower limit $(\hat{y}_{k+H})_{\text{low}}$ of prediction $\hat{y}_{k+H}$ are obtained;

Step 4 Substitute $\hat{y}_{k+H}$, $(\hat{y}_{k+H})_{\text{up}}$ and $(\hat{y}_{k+H})_{\text{low}}$ into the following Eq. (25) to calculate the mean, upper limit and lower limit of system faulty probability [35].

$$\text{Prob}\{\hat{y} \in s_2\} = \frac{\hat{y} - U(s_1)}{U(s_2) - U(s_1)} \tag{25}$$

where s_1 and s_2 denote normal and faulty state of y separately, $U(s_1)$ and $U(s_2)$ are the corresponding utilities. The system is judged to be faulty when the predicted faulty probability exceeds the threshold.

Step 5 Return to step 2 for next time instance fault prediction.

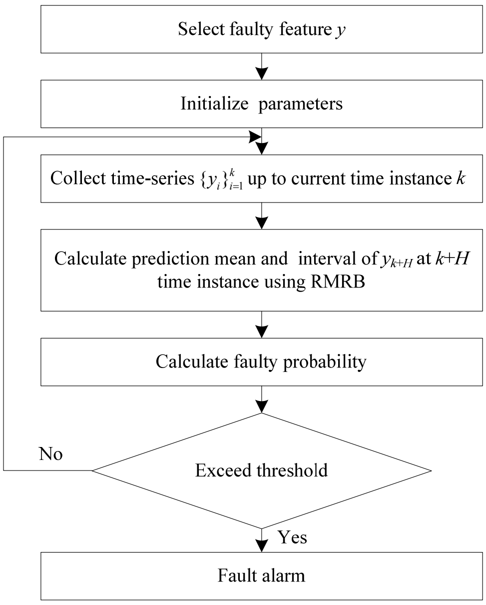


Fig. 1. Fault prediction flow chart

V. An Experiment Case Study

In this section, two real failure datasets from a certain navigation receiver is applied to demonstrate the implementation and validity of the proposed RMRB based fault prediction algorithm. The two test datasets, (a) frequency deviation failure of reference-clock and (b) gain attenuation failure of intermediate frequency amplifier, are gathered. The test platform is mainly composed of control computer, measuring instrument and navigation receiver with fault injection, where some devices in the prototype are replaced by tunable ones comparing with actual navigation receiver equipment. As a result, different faulty types and faulty degree is generated by changing such tunable devices value.

A. Frequency deviation failure of reference-clock

The frequency deviation failure of reference-clock can affect the output frequence. A total number of 205 frequence values are recorded to construct a time-series [fre_1, fre_2, …, fre_{205}], where the receiver is normal at the beginning, then the faulty degree grows up as time going on. fre_i is measured by frequency meter at $i\Delta$t time instance, Δt =1min. At last, time-series $\{fre_i\}_{i=1}^{205}$ is normalized and a 1 offset is added to the normalized $\{fre_i\}_{i=1}^{205}$ for keeping positive.

1. *Semi-quantitative analyzing*

In this section, semi-quantitative validation [25] is carried out to demonstrate the proposed PI estimation method. As shown in Eq. (3), PI are determined by variance σ_m^2 and σ_ε^2. Clearly σ_m^2 and σ_ε^2 are inherently associated with the data samples (namely ANFIS models in this case). Thus one way to validate the proposed PI estimation approach is to check for the correlation between the variability of data samples and PI. Three corresponding relationships are considered as follows: prediction error – PI, ANFIS models accuracy – PI and prediction horizon – PI.

(1) Relationship between prediction error and PI

It is obvious that the PI should be wider when the prediction error is bigger, whereas smaller prediction error relating to narrower PI. The normalized cross-correlation between 5-steps-ahead prediction error ε_{k+5} and corresponding prediction interval PI_{k+5} is shown in Fig. 2. From Fig. 2, it can be seen that the highest cross-correlation coefficient between ε_{k+5} and PI_{k+5} is not at the location of delay τ=0, actually the highest coefficient is at τ=-9. The reason is that training data for calculating PI_{k+5} is some latest observations [fre_k, fre_{k-1},…] up to current time instance k, which brings in some hysteresis to PI_{k+5} comparing with actual error ε_{k+5}. Meanwhile, the normalized cross-correlation coefficient between ε_{k+5} and PI_{k+5} is 0.8371 at τ=0, and this also indicates a high correlation between ε_{k+5} and PI_{k+5} at τ=0. Therefore, it demonstrates the relationship of prediction error to PI.

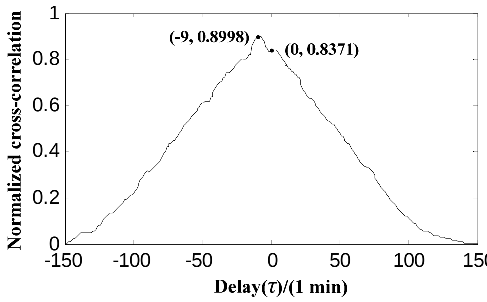


Fig. 2. The normalized cross-correlation between 5-steps-ahead prediction error ε_{k+5} and corresponding PI_{k+5}

(2) Relationship between ANFIS models accuracy and PI

In the proposed RMRB method, ANFIS models are treated as samples for bootstrap. It is obvious that high quality samples (exact ANFIS models) will improve the estimation accuracy of PI, i.e., a narrow width of PI. In order to yield poor samples (less accurate ANFIS models), the parameters of the trained ANFIS models are perturbed by adding noise to them. 5-steps-ahead PI widths with different noise added in ANFIS parameters are listed in Fig. 3, where different noises follow Gaussian distribution N(0,(0.002i)^2), i=0, 1, 2,…, 10, and i=0 represents no noise. The PI width is computed as the following equation.

$$MPIW = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (U_i - L_i) \tag{26}$$

where N_{test} is total number of test data, L_i and U_i are upper and lower limit of PI at i time instance. Fig. 3 shows that higher quality samples correspond to narrower PI, and this verifies the correlation between ANFIS models accuracy and PI.

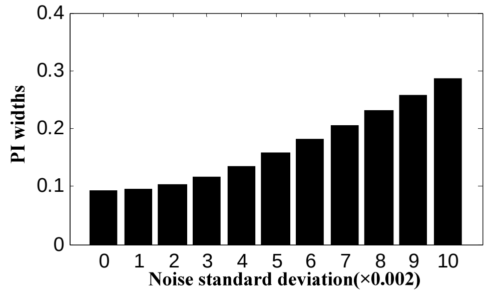


Fig. 3. 5-steps-ahead PI widths with different noise added in neural network parameters

(3) Relationship between prediction horizon and PI

For multiple-steps-ahead prediction, the one-step-ahead prediction error is accumulated to the next step prediction due to the underlying iterative prediction process. In this case, PI accuracy should be decreasing (PI width increasing) as prediction horizon growing up. 1 to 10 steps-ahead PI are shown in Fig. 4, which indicates that the prediction steps is related to PI.

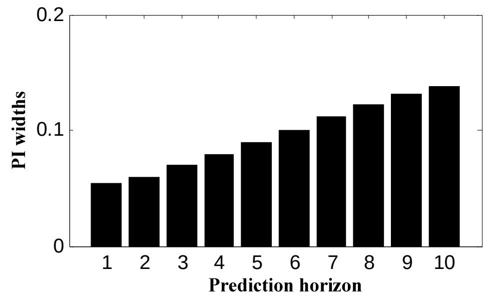


Fig. 4. 1~10 steps-ahead PI widths

2. *Contrast of different PI methods*

5-steps-ahead prediction of the normalized time-series $\{fre_i\}_{i=1}^{205}$ is carried out to verify a performance comparison between the proposed algorithm and three other methods: (a) prediction error covariance statistic (PECS) [36], (b) moving block bootstrap (MBB) [24], (c) model re-sampling bootstrap (MRB) [25] and (d) the proposed recursive models re-sampling bootstrap (RMRB). ANFIS with embedding dimension m=3 is used as neural network for all the four methods for training time-series. For PECS, the latest 10 prediction errors are utilized to estimate prediction error variance σ, then σ is substituted into Eq. (5) to calculate PI. The parameters are set as B=20, n=10, S=40, T=50, M=10. To MRB, a global optimization approach, combination of least-squares and back-propagation, is utilized to train ANFIS with the latest T length time-series data. The RLS-GD algorithm is applied to recursively update ANIFS with the newest observation for MBB and RMRB.

5-steps-ahead PIs with 95% confidence level of the four methods are plotted in Fig. 5. Fig. 6 shows the absolute value of 5-steps-ahead prediction error and corresponding half PI width of the four methods. 10 times Monte Carlo simulations are performed to calculate PI coverage probability (PICP), mean PI width (MPIW) and total time consuming of the four methods, as listed in Table 1, where MPIW is according to Eq. (26) and PICP can be computed as follows.

$$PICP = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} c_i \tag{27}$$

$$c_i = \begin{cases} 1 & fre_i \in [L_i, U_i] \\ 0 & fre_i \notin [L_i, U_i] \end{cases} \tag{28}$$

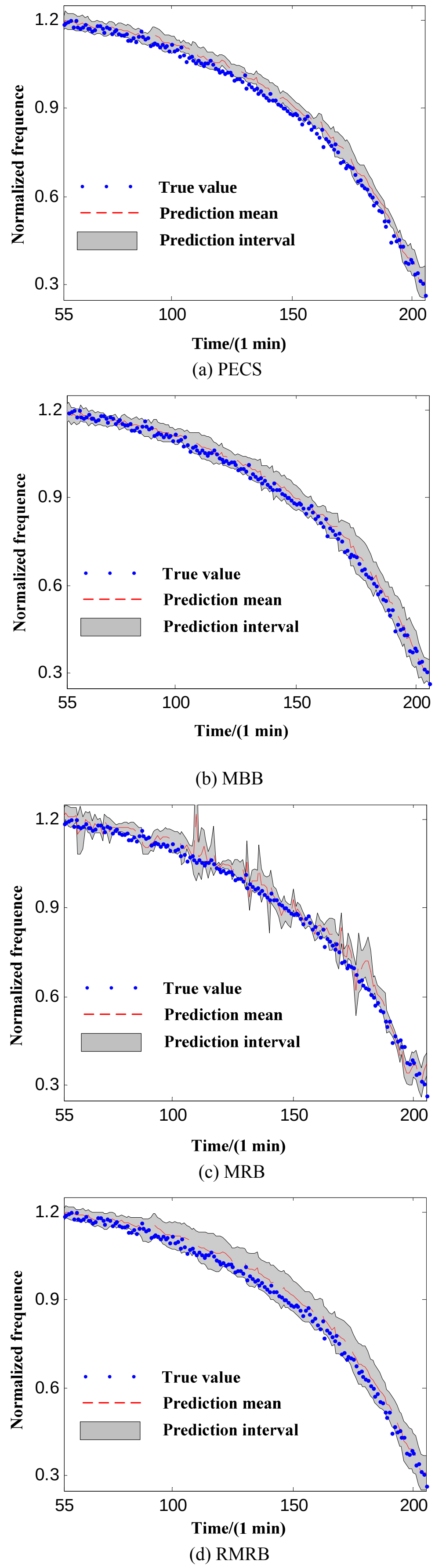
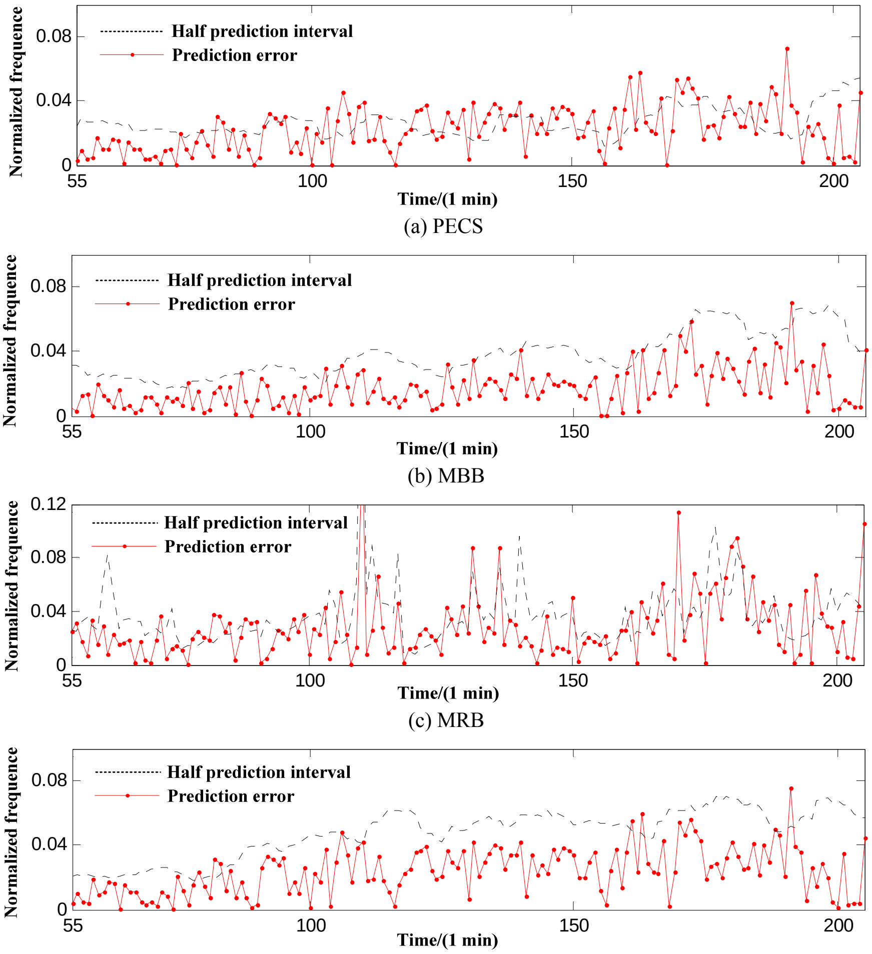


Fig. 5. 5-steps-ahead PIs of frequency deviation failure of reference-clock

Both PICP and MPIW estimate the quality of PI in only one aspect. In order to evaluate PI from the two aspects (PICP and MPIW) together, a combined index, coverage width-based criterion (CWC) [12], is calculated,



(a) PECS

(b) MBB

(c) MRB

(d) RMRB

Fig. 6. The absolute value of 5-steps-ahead frequence prediction error and corresponding half PI width under frequency deviation failure of reference-clock.

$$CWC = MPIW\left(1+\gamma(PICP)e^{-\eta(PICP-\mu)}\right) \tag{29}$$

$$\gamma(PICP) = \begin{cases} 0 & PICP \geq \mu \\ 1 & PICP < \mu \end{cases} \tag{30}$$

where μ corresponds to confidence level (1-α)%, here μ=0.95 and η=40 in this experiment. It is evident that small CWC indicates high quality PI estimation.

TABLE I
INDEXES OF 5-STEPS-AHEAD PREDICTION UNDER FREQUENCY DEVIATION FAILURE OF REFERENCE-CLOCK

Method	PICP	MPIW	CWC	Time Consuming(s)
PECS	0.5960	0.0339	4.7842×10^4	2
MBB	0.9316	0.0725	0.2238	123
MRB	0.6642	0.0613	5.6535×10^3	1330
RMRB	0.9417	0.0917	0.2195	365

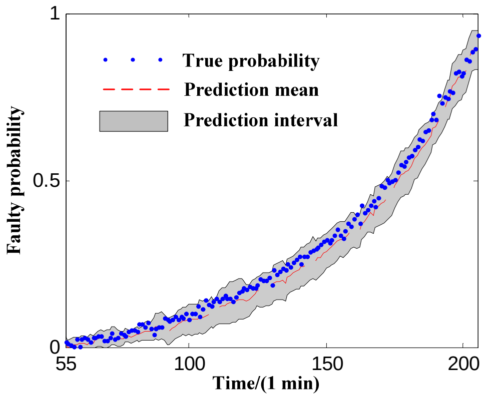


Fig. 7. Faulty probability mean and PI of 5-steps-ahead prediction by RMRB under frequency deviation failure of reference-clock

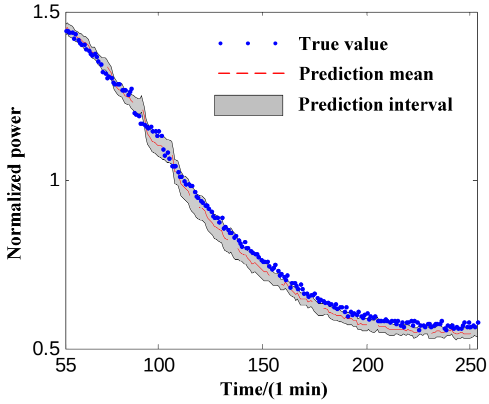
Fig. 5-6 and Table 1 show that misleadingly narrow PI width results in a low PICP for PECS, thus CWC for PECS is the highest. Apparently, the PI quality of PECS is much lower than the other three methods though time consuming is the shortest. CWC of MRB is the second highest, this is primarily due to the low prediction accuracy with global optimization learning algorithm for training ANIFS, since such learning algorithm fails to capture faulty system new dynamics without sufficient training dataset (only time-series data up to current time instance is available). MBB and RMRB, both adopting recursive learning algorithm for training ANIFS, exhibit good quality PIs for their small CWCs and high PICPs. Therefore the results of MBB and RMRB verify the effectiveness of the proposed recursive re-sampling algorithm. In addition, the PICP of the proposed RMRB is 0.9470, which approximately satisfies the prescribed 95% confidence level. It is observed that time consuming of RMRB is longer than MBB, the reason is that MBB re-samples directly from the time-series to calculate PI while RMRB re-samples from ANFIS models built by time-series data. However, comparing with MBB, the advantage of RMRB is its convenient to estimate PIs online, i.e., calculating 1 to H steps-ahead PIs by iterating a one-step-ahead predictor ANFIS$_{k+1|k}$ without constructing different time horizon predicting model ANFIS$_{k+i|k}$ as MBB does, i = 1, 2, …, H.

Suppose $U(s_1)$=1.2 and $U(s_2)$=0.2 denote the utilities of the normal and faulty state, respectively. We substitute the mean, upper limit and lower limit of 5-steps-ahead prediction by

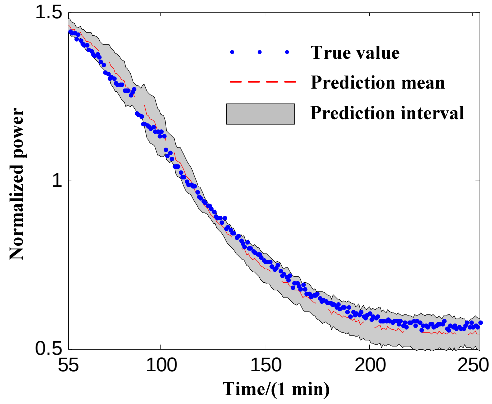
RMRB into Eq. (25) to calculate faulty probabilities, as shown in Fig. 7. It is shown that as the faulty degree expands, the faulty probability increases at the same time.

B. Gain attenuation failure of intermediate frequency amplifier

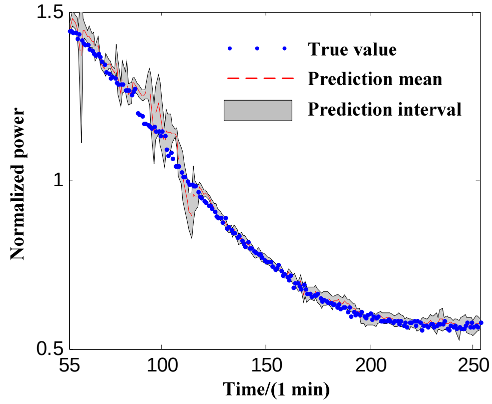
The receiver output power is corresponding to the gain attenuation failure of intermediate frequency amplifier. A time-series $[pow_1, pow_2, \ldots, pow_{253}]$ are measured by power meter, where the receiver is normal at the beginning, then the faulty degree of gain attenuation failure expands as time going on. Also time-series $\{pow_i\}_{i=1}^{253}$ is normalized and a 1 offset is added to the normalized $\{pow_i\}_{i=1}^{253}$. We can obtain the same result for semi-quantitative validation as analyzing in section 5.1.1. Fig. 8 shows the 5-steps-ahead PI with 95% confidence level of the four methods. The absolute value of 5-steps-ahead prediction error and corresponding half PI width of the four methods are shown in Fig. 9. Four methods PICP, MPIW, CWC and total time consuming are averaged across a Monte Carlo simulation consisting of 10 runs, shown in Table 2. Table 2 indicates the superior PI quality of the proposed algorithm. Fig. 10 shows fault probability of 5-steps-ahead prediction with PI.



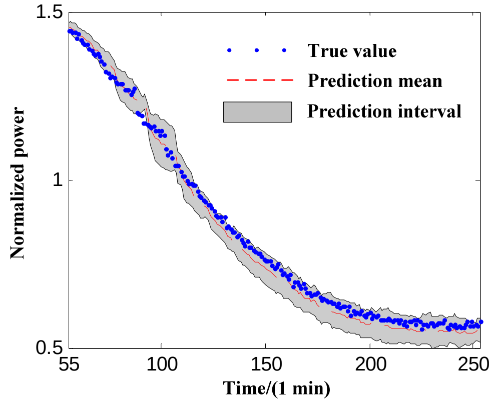
(a) PECS



(b) MBB

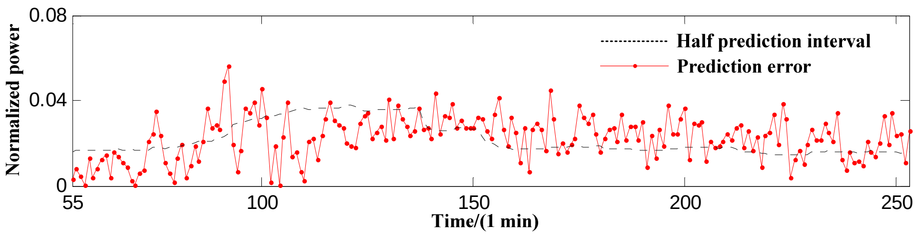


(c) MRB

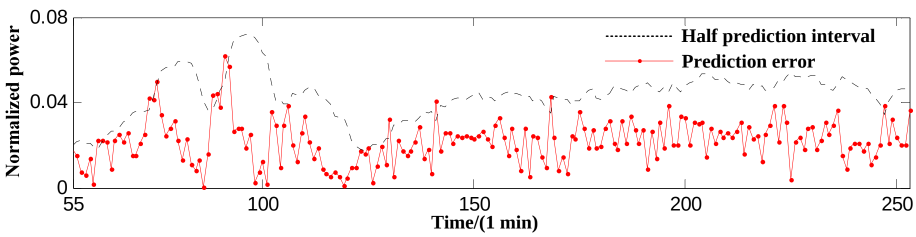


(d) RMRB

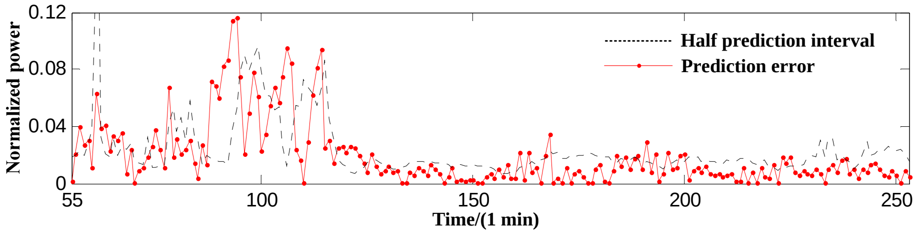
Fig. 8. 5-steps-ahead PIs of gain attenuation failure of intermediate frequency amplifier



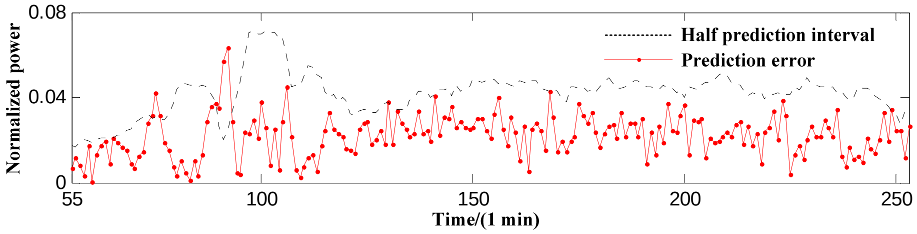
(a) PECS



(b) MBB



(c) MRB



(d) RMRB

Fig. 9. The absolute value of 5-steps-ahead power prediction error and corresponding half PI width under gain attenuation failure of intermediate frequency amplifier

Table II
INDEXES OF 5-STEPS-AHEAD PREDICTION

Method	PICP	MPIW	CWC	Time Consuming(s)
PECS	0.4523	0.0224	9.9125×10^{6}	2
MBB	0.9409	0.0812	0.1981	162
MRB	0.7085	0.0362	5.6757×10^{2}	1809
RMRB	0.9548	0.0810	0.0810	483

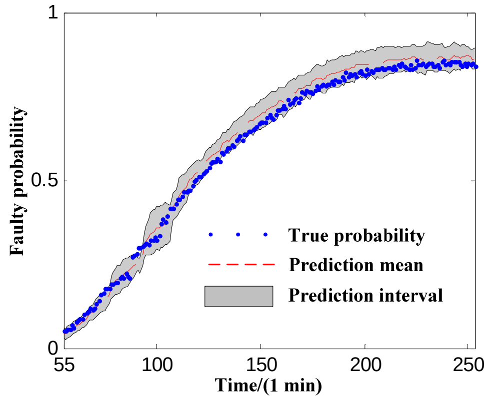


Fig. 10. Faulty probability mean and PI of 5-steps-ahead power prediction by RMRB under gain attenuation failure of intermediate frequency amplifier

VI. Conclusion

In this paper, a recursive models re-sampling bootstrap (RMRB) based fault prediction with PI estimation is presented. The proposed RMRB could provide PI associated with point prediction, when there is lack of the whole life cycle failure degradation sequences from the predicted system or other similar systems. The estimation of PI allows decision maker to efficiently quantify the level of uncertainty towards the predicted system degradation, which would reduce false alarm and risk of inaccurate maintenance decision. Using two real failure datasets, the case study results illustrate validity and potential applications for fault prediction. Moreover, the proposed RMRB can be conveniently applied to other data-driven models.

References

[1] Xu YL, Sun JZ, Chen XH, Wang GM, "Electronic equipment fault prediction method based on accelerated degradation testing and particle filter", *Acta Aeronautica et Astronautica Sinica*, vol. 33, no. 8, pp. 1483-1490, 2012.

[2] YAN Cl, LIU K, "Theory of Economic Life Prediction and Reliability Assessment of Aircraft Structures", *Chinese Journal of Aeronautics*, vol. 24, no. 2, pp. 164-170, 2011.

[3] Chen C, Vachtsevanos G, "Bearing condition prediction considering uncertainty: An interval type-2 fuzzy neural network approach", *Robotics and Computer-Integrated Manufacturing*, vol. 28, pp. 509-516, 2012.

[4] Wang M, Wang J, "CHMM for tool condition monitoring and remaining useful life prediction", *The International Journal of Advanced Manufacturing Technology*, vol. 59, pp. 463-471, 2012.

[5] Si XS, Hu CH, Yang JB, Zhang Q, "On the dynamic evidential reasoning algorithm for fault prediction", *Expert Syst Appl*, vol. 38, pp. 5061-5080, 2011.

[6] Pham HT, Yang BS, "Estimation and forecasting of machine health condition using ARMA/GARCH model", *Mechanical Systems and Signal Processing*, vol. 24, pp. 546–558, 2010.

[7] Zhang L, Wang Z, Zhao S, "Short-term fault prediction of mechanical rotating parts on the basis of fuzzy-grey optimising method", *Mechanical Systems and Signal Processing*, vol. 21, pp. 856–865, 2007.

[8] Wagner S, "A Bayesian network approach to assess and predict software quality using activity-based quality models", *Information and Software Technology*, vol. 52, pp. 1230–1241, 2010.

[9] Wei Xing S, Yun Long Z, Fang L, Kun Yuan H, "An online outlier detection method based on wavelet technique and robust RBF network", *Transactions of the Institute of Measurement and Control*, vol. 35, no. 8, pp. 1046-1057, 2013.

[10] Mnasser A, Bouani F, "Centralized predictive controller design based on an uncertain neural network for multivariable non-linear systems", *Transactions of the Institute of Measurement and Control*, vol.28, no.1, pp., 2014.

[11] Chen XZ, Yu JS, Tang DY, Wang YX, "A Novel PF-LSSVR-based Framework for Failure Prognosis of Nonlinear Systems with Time-varying Parameters", *Chinese Journal of Aeronautics*, vol. 25, no. 5, pp. 715-724, 2012.

[12] Khostavi A, Nahavandi S, Creighton D, Atiya AF, "Comprehensive Review of Neural Network-Based Prediction Intervals and New Advances", *IEEE Transactions on Neural Networks*, vol. 22, no. 9, pp. 1341-1356, 2011.

[13] Odgaard P, Stoustrup J, Mataji B, "Prediction of lacking control power in power plants using statistical models", *Proc Euro Control Conf*, p. 4679-4684, 2007.

[14] Nix DA, Weigend AS, "Estimating the mean and variance of the target probability distribution", *Proc IEEE int Conf Neural Netw*, pp. 55-60, 1994.

[15] Hwang JTG, Ding AA, "Prediction intervals for artificial neural networks", *J Amer Stat Assoc*, vol. 92, no. 438, pp. 748-757, 1997.

[16] MacKay DJC, "The evidence framework applied to classification networks", *Neural Computation*, vol. 4, no. 5, pp. 720-736, 1992.

[17] EFRON B, TIBSHIRANI RT, "An introduction to the bootstrap", *Chapman & Hall Inc*, 1993.

[18] Khosravi A, Nahavandi S, Creighton D, "A neural network-GARCH-based method for construction of Prediction

Intervals", *Electric Power Systems Research*, vol. 96, pp. 185-193, 2013.
[19] Malhotra R, Jain A, "Fault prediction using statistical and machine learning methods for improving software quality", *Journal of Information Processing Systems*, vol. 8, no. 2, pp. 241-262, 2012.
[20] Heskes T, "Practical confidence and prediction intervals", *Advances in neural information processing systems*, vol.15, no.1, pp. 176-182, 1997.
[21] Sonmez R, "Range estimation of construction costs using neural networks with bootstrap prediction intervals", *Expert Syst Appl*, vol. 38, pp. 9913-9917, 2011.
[22] Baraldi P, Cadini F, Mangili F, Zio E, "Model-based and data-driven prognostics under different available information", *Probabilist Eng Mech*, vol. 32, pp. 66-79, 2013.
[23] Li H-y, Teng J, Li Z-h, Zhang L, "Nonlinear Dynamic Analysis Efficiency by Using a GPU Parallelization", *Engineering Letters*, vol. 23, no. 4, pp. 232-238, 2015.
[24] Polltis DN, "The Impact of Bootstrap Methods on Time Series Analysis", *Statistical Science*, vol. 18, no. 2, pp. 219-230, 2003.
[25] Oleng NO, Gribok A, Reifman J, "Error bounds for data-driven models of dynamical systems", *Computers in Biology and Medicine*, vol. 37, pp. 670-679, 2007.
[26] Luan JH, Tang J, Lu C, "Prediction Interval on Spacecraft Telemetry Data Based on Modified Block Bootstrap Method", *Lecture Notes in Computer Science*, vol. 6320, no. 1, pp. 38-44, 2010.
[27] Khosravi A, Nahavandi S, Creighton D, "Quantifying uncertainties of neural network-based electricity price forecasts", *Appl Energ*, vol. 112, pp. 120-129, 2013.
[28] Khosravi A, Nahavandi S, Creighton D, "Prediction Intervals for Short-Term Wind Farm Power Generation Forecasts", *IEEE Transactions on Sustainable Energy*, vol. 4, no. 3, pp. 602-610, 2013.
[29] Reis TSd, Gomide W, Anderson JADW, "Construction of the Transreal Numbers and Algebraic Transfields", *IAENG International Journal of Applied Mathematics*, vol. 46, no. 1, pp. 11-23, 2016.
[30] Cheng CH, Wei LY, Liu JW, Chen TL, "OWA-based ANFIS model for TAIEX forecasting", *Economic Modelling*, vol. 30, no. pp. 442-448, 2013.
[31] Farokhnia A, Morid S, Byun H-R, "Application of global SST and SLP data for drought forecasting on Tehran plain using data mining and ANFIS techniques", *Theoretical and Applied Climatology*, vol. 104, no. 1, pp. 71-81, 2011.
[32] Hu YB, "Iterative and recursive least squares estimation algorithms for moving average systems", *Simulat ion Modelling Practice and Theory* vol. 34, no.1, pp. 12-19, 2013.
[33] Xua Y, Zeng XQ, Han LX, Yang J, "A supervised multi-spike learning algorithm based on gradient descent for spiking neural networks", *Neural Networks*, vol. 43, no.1 pp. 99-113, 2013.
[34] Suprapto, Wardoyo R, Widjaja BH, Pulungan R, "A Formal Proof of Correctness of Construct Association from PROMELA to Java", *IAENG International Journal of Computer Science*, vol. 42, no. 4, pp. 313-331, 2015.
[35] Zhou Z-J, Hu C-H, Xu D-L, Chen M-Y, Zhou D-H, "A model for real-time failure prognosis based on hidden Markov model and belief rule base", *European Journal of Operational Research*, vol. 207, no.1, pp. 269-283, 2010.
[36] Odgaard PF, Stoustrup J, "Estimation of Uncertainty Bounds for the Future Performance of a Power Plant", *IEEE Transactions on Control Systems Technology*, vol. 17, no. 1, pp. 199-206, 2009.

Li Dongping , female, was born in the city of Kunming, Yunnan Province, China on 1971. She acts as an associate professor in the institute of Information Science and Technology, Kunming University. Her interest is data mining.