Expert Systems with Applications 37 (2010) 2550–2562

Contents lists available at ScienceDirect

Expert Systems with Applications

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

System reliability prediction model based on evidential reasoning algorithm with nonlinear optimization

Chang-Hua Hu [a,*], Xiao-Sheng Si [a], Jian-Bo Yang [b]

[a] Department of Automation, Xi'an Institute of Hi-Tech, Hongqing, Xi'an, Shaanxi 710025, PR China
[b] Manchester Business School, The University of Manchester, Manchester M15 6PB, UK

ARTICLE INFO

Keywords:
Evidential reasoning
Reliability
Nonlinear optimization
Forecasting

ABSTRACT

In this paper, a novel reliability prediction technique based on the evidential reasoning (ER) algorithm is developed and applied to forecast reliability in turbocharger engine systems. The focus of this study is to examine the feasibility and validity of the ER algorithm in systems reliability prediction by comparing it with some existing approaches. To determine the parameters of the proposed model accurately, some nonlinear optimization models are investigated to search for the optimal parameters of forecasting model by minimizing the mean square error (MSE) criterion. Finally, a numerical example is provided to demonstrate the detailed implementation procedures. The experimental results show that the prediction performance of the ER-based prediction model outperforms several existing methods in terms of prediction accuracy or speed.

1. Introduction

The safe and reliable operation of technical systems is of great significance for the protection of human life and health, the environment, and the economy in general. The correct functioning of those systems also has a profound impact on production cost and product quality. The early detection of faults is critical in avoiding performance degradation and damage to machinery or human life. Such disasters as the explosion of the US space shuttle "*Challenger*", the crash of a Chinese early warning aircraft and the sinking of a Russian submarine could have been avoided if a robust system reliability forecasting mechanism had existed or had been strictly enforced.

System reliability can be defined as the probability that system will perform its intended function for a specific period of time under stated conditions (Gran & Helminen, 2001). In engineering practice, reliability is a fundamental attribute for the safe operation of any modern technological system. Focusing on safety, reliability analysis aims at the quantification of the probability of failure of the system (Zio, 2009). As such, reliability analysis has been a significant direction of safety management and attached great importance. Until now, many researchers have paid increasing attention to reliability analysis in safety management. It is necessary and important to forecast system reliability as decision makers are generally interested in estimating the future occurrences of system failures for resource planning, inventory management, developing realistic policies for age replacement and logistic support. In reality, practically most engineering systems are repairable and their reliability measures change with time. By considering this change as a time series process, the "*growth*" or "*deterioration*" of the system can be estimated (Ho & Xie, 1998). In recent years, reliability prediction techniques have developed by a large community of researchers.

An accurate system or product reliability prediction model not only can learn and track the reliability and operational performance, but also offer useful information for managers to take follow-up actions to improve the quality and cost of system (Yi-Hui, 2007). Presently, different reliability models were proposed to estimate and predict the reliability (EI-Sebakhy, 2008). Most of these modeling schemes were developed using linear or nonlinear multiple regression, Bayesian statistical, Autoregressive integrated moving average (ARIMA), neural networks, support vector machine (SVM). However, they often do not perform very well in terms of accuracy or speed, and suffer from a number of drawbacks such as lack of suitable models, exceptional assumptions of prediction model, and difficulty to validate. Furthermore, a fundamental issue in reliability analysis based on the failure data is the uncertainty in the failure occurrence and consequence (Zio, 2009; Wang & Elhag, 2007). As such, for a complex engineering system, many reliability analysis problems involve both quantitative data and qualitative information, as well as various types of uncertainties such as incompleteness and fuzziness. Under these circumstances, there is an urgent need to develop a new reliability analysis method which can deal with various types of uncertainties efficiently.

* Corresponding author. Tel.: +86 29 84743949; fax: +86 29 84744603.
E-mail addresses: hch6603@263.net (C.-H. Hu), sixiaosheng@126.com (X.-S. Si), jian-bo.yang@manchester.ac.uk (J.-B. Yang).

0957-4174/$ - see front matter Crown Copyright © 2009 Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.08.024

The evidential reasoning (ER) algorithm has been developed by Yang et al. for multiple attribute decision analysis (MADA) under uncertainty (Yang & Singh, 1994; Yang & Xu, 2002a, 2002b). This approach is developed on the basis of decision theory and the Dempster–Shafer (D–S) theory of evidence (Dempster, 1967; Shafer, 1976). Due to the power of the ER approach in handing and representing uncertainties (Yang, Wang, Xu, & Chin, 2006a), so far, it have been applied to many areas, such as environmental impact assessment (Wang, Yang, & Xu, 2006), pipeline leak detection (Xu et al., 2007), bridge condition assessment (Wang & Elhag, 2008), etc. In addition, the ER approach has been applied to conduct safety analysis. For instance, in safety analysis and assessment, system reliability can be described linguistically due to its easy interpretation. One may choose to use such linguistic terms as "*very low*", "*low*", "*average*", "*high*", and "*very high*". So it is common to assess a reliability level by degrees to which it belongs to such linguistic variables as "*very low*", "*low*", "*average*", "*high*", and "*very high*" that are referred to as reliability expression. For example, an expert may state that he is 20% sure system reliability is low and 80% sure it is high. In the statement, high and low denote distinctive evaluation grades, and the percentage values of 20 and 80 are referred to as the degrees of belief, which indicate the extents that the corresponding grades are assessed to. The above assessment can be expressed as the following expectation: $S(\text{reliability}) = \{(low, 0.2), (high, 0.8)\}$, where $S(\text{reliability})$ stands for the state of the system reliability and the real numbers 0.2 and 0.8 denote the degrees of belief of 20% and 80%, respectively. The above representation of reliability, $S(\text{reliability}) = \{(low, 0.2), (high, 0.8)\}$, is a belief distribution structure in fact. The ER algorithm is developed to model the belief distribution structure (Yang, 2001; Yang & Xu, 2002a; Yang, Liu, Wang, Sii, & Wang, 2006b; Yang, Liu, Xu, Wang, & Wang, 2007; Wang & Elhag, 2007). As such, we can conclude that the ER approach is applied to reliability analysis and further forecasting is possible. The analysis results can provide a panoramic picture about the system's reliability state. However, no attempt has been directly made to address the issue of how to deal with system reliability prediction problems. This research has been conducted to fill the gap.

Owing to the advantages of the ER approach in dealing with uncertainty, in this paper we develop a new model for reliability prediction referred as the ER-based reliability prediction model, which is targeted at the turbocharger in particular and at relevant fields in general. The main purpose of this model is focused on the analysis of the present system reliability and the prediction of system reliability at some future time points based on obtained test data. In the ER-based reliability prediction model, input data and attribute weights are combined to generate appropriate conclusions using the ER algorithm. In order to overcome the difficulty to accurately determine the parameters of the ER-based reliability prediction model, some nonlinear optimization models for training the parameters of the ER-based prediction model are investigated in some details. Finally, the proposed model is applied to forecast turbochargers engines' reliability. The experimental results demonstrate the validity and potential of the ER-based model for reliability prediction.

The rest of this paper is organized as follows: Section 2 presents the related works to reliability prediction and analysis. Section 3 gives a brief description of the ER algorithm. In Section 4, the ER-based reliability prediction model is constructed. Section 5 investigates some optimal learning models for training the parameters of the ER-based prediction model. In Section 6, a numerical example is provided to demonstrate the detailed implementation procedures and the validity of the proposed approach in the areas of reliability prediction. Finally, some concluding remarks and recommendations for future works are drawn in Section 7.

2. Related works

At present, many prediction mechanisms and mathematical models have been proposed to accomplish reliability prediction by a large community of researchers. The existing prediction methods can be roughly classified into three categories as follows.

2.1. Model-based methods and reliability prediction

The model-based approach is to select an appropriate probability function as the reliability growth model in advance, then use this model to predict future system reliability (Ascher & Hung, 1987; MIL-HDBK-189, 1981). EI-Aroui and Soler (1996) used Bayesian statistical model to predict software reliability and demonstrated that the proposed method is useful for simulated failure data based on the numerical examples. Pham and Pham (2000, 2001) applied Bayesian approach to predict software reliability. The results show that proposed model outperforms other times-between-failures model in terms of the sum of square errors. Pham and Zhang (2003) proposed a prediction model based on Non-Homogeneous Poisson Process (NHPP) approaches for reliability prediction. The simulated results show that proposed model has better performance and fitness than other existing NHPP models.

However, in the model-based methods, predetermined probability function may fit the failure data well, but the prediction result may not be good in practice. Also, it is often difficult for a reliability engineer to select an appropriate probability function from many existing probability functions (Liang & Afzel, 2005, 2006; Tong & Liang, 2005).

2.2. Time series analysis and reliability prediction

Time series analysis, developed by Box and Jenkins (1970), has been applied to forecasting extensively, which involves utilizing time series analysis techniques to construct an equation representing reliability versus time, then to predict future reliability. The ARIMA model, developed by Box and Jenkins, has been one of most popular approaches in time series prediction. For instance, Singpurwala (1978) first employed the ARIMA model to analyze the failure data of repairable systems. Furthermore, Singh, Chrys, and Frishwick (1994) developed the failure prediction model for series-parallel complex systems using the ARIMA model. Singh supposed that the predictive power of the ARIMA model is superior to that of other traditional prediction models. Ho and Xie (1998) proposed an ARIMA model to analyze the failures of repairable systems. The experimental results indicated that the proposed model substantially outperforms the Duane model in terms of the mean absolute deviation. Liang and Tong (2001) applied SARIMA model to analyze and forecast the reliability of repairable systems. Ho, Xie, and Goh (2002) presented four models, including the ARIMA model, multilayer feed-forward neural network model (MFNN) and recurrent neural network (RNN) model, to predict failures of repairable systems. The experimental results indicated that both the RNN model and the ARIMA model are superior to the MFNN model in terms of forecasting accuracy.

2.3. Machines learning and reliability prediction

In recent years, machine learning techniques have been applied for predicting reliability. Liu, Kuo, and Sastri (1995) demonstrated how feed-forward multilayer perceptions (MLP) networks can successfully identify underlying failure distribution and estimate the parameters. Amjady and Ehsan (1999) presented an expert system based on neural networks for evaluating power system reliability. Cai, Cai, Wang, Yu, and Zhang (2001) proposed a neural network model to predict software reliability. They evaluated the performance of neural network model by different network architectures and further concluded that neural network model is better for a smooth reliability data set than a fluctuating one. A multilayer perceptron feed forward neural network has been proposed by Aljahdali, Sheta, and Rine (2001) as an alternative technique to build reliability growth models. The proposed model was used with a slightly different configuration in which the number of neurons in the input layer represents the number of delay in the input data. They made a comparison between regression parametric models and neural network models. Their results indicate that neural networks were able to provide models with small sum of squares errors (SSE) than the regression model. Xu, Xie, Tang, and Ho (2003) applied two techniques of neural networks: (i) multilayer perceptron feed forward neural networks and (ii) radial basis function to predict the reliability of engine systems. They found that the neural network approaches provide more accurate prediction results than the ARIMA model. Notably, Chang, Lin, and Pai (2004) applied a hybrid learning neural fuzzy system to forecast engine system reliability. Numerical results demonstrate that the proposed model is able to achieve more accurate forecasting results than ARIMA and generalized regression neural network model (GRNN). Costa, Vergilio, Pozo, and Souza (2005) used genetic programming (GP) to estimate the reliability growth. They carried out two experiments: one based on time and the other one based on test coverage. Also, they compared the results with other traditional and non-parametric ANN models. Zheng (2009) proposed a non-parametric software reliability prediction system based on neural network ensembles. The author conducted the comparative studies between the proposed system with the single neural network based systems and three parametric NHPP models. The experimental results demonstrate that the system predictability can be significantly improved by combining multiple neural networks.

SVM has been widely employed to deal with nonlinear forecasting problems. Recently, SVM has been applied to reliability prediction. Hong and Pai (2006) applied SVM to predict engine reliability. The simulated results show that the SVM model is a valid and promising alternative in reliability prediction. In order to determine the parameters of SVM model accurately, Pai and Hong developed a new SVM model with simulated annealing algorithms to predict software reliability (Pai & Hong, 2006). The results indicate that the SVM model with simulated annealing algorithms provides more accurate prediction results than the other prediction models. Furthermore, Pai (2006) proposed a SVM model with genetic algorithm to predict reliability. Two case studies were provided to demonstrate the performance of reliability foresting. The results reveal that the proposed model results in better predictions. Chen (2007) presented a new prediction model, known as GA-SVR, which searches for support vector regression (SVR) machine's optimal parameters using real-value genetic algorithm. Then the author applied the proposed model to predict reliability in engine systems. The experimental results demonstrate that the proposed model outperforms the existing neural network methods and ARIMA models based on the normalized root mean square error and mean absolute percentage error.

2.4. ER approach and safety analysis

The ER algorithm, developed by Yang and Singh (1994, 2002a) for MADA under uncertainty. This approach is developed on the basis of decision theory and the Dempster–Shafer (D–S) theory of evidence. Extensive research dedicated to the ER approach has been conducted in recent years. In addition, the ER approach has been applied to conduct safety analysis. For instance, Wang, Yang, and Sen (1995) employed fuzzy set theory and the ER approach to assess system safety. In order to prevent information loss in the process of system synthesis and aggregation of multi-persons' opinions, a methodology based on fuzzy set modeling and the ER approach is developed, which can make full use of available information to obtain safety synthesis without information loss (Wang, Yang, & Sen, 1996). Furthermore, in order to combine fuzzy logic and D–S models to deal with fuzziness and incompleteness in safety analysis, Liu, Yang, Wang, Sii, and Wang (2004) constructed a framework for modeling the safety of an engineering system using a fuzzy rule-based evidential reasoning (FURBER) approach. However, in this model, it is difficult to determine its parameters entirely subjectively, in particular for a large scale fuzzy rule base with hundreds of rules. Also, a change in model parameters may lead to significant changes in the performance of FURBER. To solve this problem, recently, the methods for self-tuning a FBRB for engineering system safety analysis have been developed based on a number of single and multiple objective nonlinear optimization models (Liu, Yang, Ruan, Martinez, & Wang, 2008).

Several other methods, such as Markov processes, Kalman filter model, and times-between-failures model had been developed in predicting and analyzing reliability (Kimura, Yamada, & Osaki, 1995; Lyu, 1996; Lu, Lu, & Kolarik, 2001; Saranga & Knezevic, 2001; Xie, 1991). However, none of the above works, used and/or evaluated the capabilities of the ER approach in system reliability prediction.

3. Evidential reasoning algorithm

To begin with, suppose there are L basic attributes $e_i (i = 1, \dots, L)$ associated with system reliability state y. Define a set of L basic attributes as evidence source as follows:

$$E = \{e_1, \dots, e_L\}. \tag{1}$$

Suppose the weights of the attributes are given by $w = \{w_1, \dots w_i, \dots, w_L\}$ where w_i is the relative weight of the ith basic attribute e_i, and the weights of the attributes are normalized to satisfy the following constraints:

$$0 \leqslant w_i \leqslant 1 \quad \text{and} \quad \sum_{i=1}^{L} w_i = 1 \tag{2}$$

Define N distinctive reliability state evaluation grades as represented by

$$F = \{F_1, \dots F_n, \dots, F_N\} \tag{3}$$

where F_n is the nth reliability state evaluation grade. It is worth noting that F provides a complete set of standards for assessing attributes, referred to as the assessment framework.

Mathematically, a given assessment for $e_i (i = 1, \dots, L)$ may be represented as the following distribution:

$$S(e_i) = \{(F_n, \beta_{n,i}), n = 1, \dots, N\}, \quad i = 1, \dots, L \tag{4}$$

where $\beta_{n,i} \geqslant 0, \sum_{n=1}^{N} \beta_{n,i} \leqslant 1$, and $\beta_{n,i}$ denotes a degree of belief. The above distributed assessment reads that the attribute e_i is assessed to the grade F_n with the degree of belief $\beta_{n,i}$ at time, $n = 1, \dots, N$. A assessment $S(e_i)$ is complete if $\sum_{n=1}^{N} \beta_{n,i} = 1$ and incomplete if $\sum_{n=1}^{N} \beta_{n,i} < 1$. $\beta_{n,i}$ could be generated using various ways, depending on the nature of the attribute and data available such as a quantitative attribute using numerical values or a qualitative attribute using linguistic values (Yang, 2001).

Let β_n be a degree of belief to which the system reliability state is assessed to the grade F_n. The aggregation problem is to generate $\beta_n (n = 1, \dots, N)$ by aggregating the assessments for all the associated attributes $e_i (i = 1, \dots, L)$. To this end, the following evidential reasoning approach can be used (Yang & Xu, 2002a).

Let $m_{n,i}$ be a basic probability mass representing the degree to which the ith basic attribute e_i supports the hypothesis that the system reliability state is assessed to the nth grade. Let $m_{F,i}$ be a remaining probability mass unassigned to any individual grade after all the N grades have been considered for assessing the system reliability state as far as e_i is concerned. According to *Yang's view*, the basic probability mass can be calculated as follows:

$$m_{n,i} = w_i \beta_{n,i}, \quad n = 1, \ldots, N, \; i = 1, \ldots, L \tag{5}$$

$$m_{F,i} = 1 - \sum_{n=1}^{N} m_{n,i} = 1 - w_i \sum_{n=1}^{N} \beta_{n,i}, \quad i = 1, 2, \ldots, L \tag{6}$$

$$\bar{m}_{F,i} = 1 - w_i, \quad i = 1, 2, \ldots, L \tag{7}$$

$$\tilde{m}_{F,i} = w_i \left(1 - \sum_{i=1}^{N} \beta_{n,i}\right), \quad i = 1, 2, \ldots, L \tag{8}$$

$$m_{F,i} = \bar{m}_{F,i} + \tilde{m}_{F,i} \tag{9}$$

Note that the probability mass assigned to the whole set $F, m_{F,i}$ which is currently unassigned to any individual grades, is split into parts: $\bar{m}_{F,i}$ and $\tilde{m}_{F,i}$, where $\bar{m}_{F,i}$ is caused by the relative importance of the attribute e_i and $\tilde{m}_{F,i}$ by the incompleteness of the assessment on e_i for reliability state.

Next, the basic probability masses on the L basic attributes are aggregated into the combined probability assignments by using the following analytical formulae (Wang et al., 2006):

$$\{F_n\} : m_n = K_L \left[\prod_{i=1}^{L} (m_{n,i} + \bar{m}_{F,i} + \tilde{m}_{F,i}) - \prod_{i=1}^{L} (\bar{m}_{F,i} + \tilde{m}_{F,i})\right] \tag{10}$$

$$\{F\} : \tilde{m}_F = K_L \left[\prod_{i=1}^{L} (\bar{m}_{F,i} + \tilde{m}_{F,i}) - \prod_{i=1}^{L} \bar{m}_{F,i}\right] \tag{11}$$

$$\{F\} : \bar{m}_F = K_L \prod_{i=1}^{L} \bar{m}_{F,i}, \quad n = 1, \ldots, N \tag{12}$$

with

$$K_L = \left[\sum_{n=1}^{N} \prod_{i=1}^{L} (m_{n,i} + \bar{m}_{F,i} + \tilde{m}_{F,i}) - (N-1) \prod_{i=1}^{L} (\bar{m}_{F,i} + \tilde{m}_{F,i})\right]^{-1} \tag{13}$$

Finally, the combined probability assignments are normalized into overall belief degrees by using the following equations

$$\{F_n\} : \beta_n = \frac{m_n}{1 - \bar{m}_F} \tag{14}$$

$$\{F\} : \beta_F = \frac{\tilde{m}_F}{1 - \bar{m}_F} \tag{15}$$

β_n and β_F represent the belief degrees of the aggregated assessment, to which the general attribute is assessed to the grades F_n and F, respectively. The combined assessment can be denoted by $S(y) = \{(F_n, \beta_n), n = 1, 2 \ldots, N\}$. It provides a panoramic view about the assessment state of system reliability, from which one can tell which grades the system reliability is assessed to, and what belief degrees are assigned to the defined reliability assessment grades.

The analytical ER algorithm (10)–(15) offer the ER algorithm flexibility in aggregating a large number of basic attributes. Inheriting the nonlinear features of the original ER approach, the analytical ER algorithm clearly shows nonlinear features and provides a straightforward way to conduct sensitivity analysis for the parameters of the ER algorithm such as attributes weights. It also facilitates the estimation and optimization of these parameters.

4. Reliability prediction model based on the ER algorithm

4.1. Prediction model structure and representation

In this section, a reliability prediction model is investigated in the ER framework. It is assumed that a set of observed data is provided in the form of input–output pairs $(\mathbf{X}(t_m), y(t_m)), m = 1, \ldots, M$, with $\mathbf{X}(t_m)$ being an input vector of the actual system at time t_m and $y(t_m)$ being a scalar representing the corresponding output value or subjective distribution value of the actual system at time t_m.

For prediction problems, the inputs used in a prediction model are the past input vector and the lagged observations of the current time, while the outputs are the future values. Each set of input patterns is composed of any moving fixed-length window within the time series of the input data. The general reliability prediction model can be represented as

$$\hat{y}(t + k - 1) = f(y_{t-1}, y_{t-2}, \ldots, y_{t-p}) \tag{16}$$

where $\hat{y}(t + k - 1)$ is a scalar representing the predicted future value at time $t + k - 1, (y_{t-1}, y_{t-2}, \ldots, y_{t-p})$ is a vector of lagged variables, and p represents the dimensions of the input vector (number of input nodes) or the number of past inputs related to the future value. In reliability prediction, it is reasonable to assume that the current output value y_t is related to the most recent input vector y_{t-1} or even extended into the past values $y_{t-2}, \ldots, y_{t-p}$. This is because the next output value is dependent on the current output value to certain extent which is in turn related to the current inputs.

In the above equation, if $k = 1$, the prediction is a one-step-ahead prediction, and if $k > 1$, the prediction is a multi-step prediction. Although multi-step forecasting may capture some system dynamics, the performance may be poor due to the accumulation of errors. In practice, one-step-ahead prediction results are more useful since they provide timely information for preventive and corrective maintenance plans (Xu et al., 2003). In addition, according to Wang (2007), the more the step ahead, the less reliable the forecasting operation is because multi-step prediction is associated with multiple one-step prediction operation. Thus, this study only considers one-step-ahead prediction. As such, the pattern of training data set can be described in Table 1 (Chen, 2007).

For simplicity, let $\mathbf{X}(t) = (y_{t-1}, y_{t-2}, \ldots, y_{t-p})$ denote the input vector of the prediction model in the following.

4.2. Reliability prediction model under the ER framework

In order to apply the ER approach, suppose $(y_{t-1}, y_{t-2}, \ldots, y_{t-p})$ are p basic attributes associated with system prediction output y_t, and the ER approach attempts to identify the appropriate internal representation between $(y_{t-1}, y_{t-2}, \ldots, y_{t-p})$ and $\hat{y}_t$. In this case, the number of the basic attributes L is equal to p, that is $L = p$. The key for solving the prediction problem is how to approximate the function $f(\cdot)$.

In reliability prediction problems, the ER-based prediction model can be trained first to learn relationships between past historical data and the corresponding targets, and then future output values can be predicted if the new inputs become available. Due to the fact that the input data $\mathbf{X}(t)$ may be a numerical value or a subjective distribution, there is a need to transform such data into the belief structure. Rule or utility equivalence transformation techniques can be used in this case, on which more discussions can be found in (Yang, 2001). As a result, each

Table 1
The pattern of training data set.

X				Y
y_1	y_2	$\cdots$	y_p	y_{p+1}
y_2	y_3	$\cdots$	y_{p+1}	y_{p+2}
y_3	y_4	$\cdots$	y_{p+2}	y_{p+3}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
y_{M-p}	$\cdots$	$\cdots$	y_{M-1}	y_M

input can be represented as a distribution on referential values using a belief structure. The main advantage of doing so is that precise data, random numbers, and subjective judgments with uncertainty can be consistently modeled under the same framework.

Based on the rule equivalence transformation techniques (Yang, 2001), the input vector $\mathbf{X}(t)$, that is, the p basic attributes $\mathbf{X}(t) = (y_{t-1}, y_{t-2}, \dots, y_{t-p})$ can each be represented using the following belief structure:

$$S(y_{t-i}) = \{(F_n, \beta_{n,i}(y_{t-i})), n = 1, \dots, N\}, \quad i = 1, \dots, p \tag{17}$$

Having represented each attribute as (17), the ER approach can be directly applied to combine all attributes and generate final conclusions. Using the ER analytical algorithm (5)–(15), the final prediction result $O(\hat{y}(t))$ that is generated by aggregating all attributes $\mathbf{X}(t) = (y_{t-1}, y_{t-2}, \dots, y_{t-p})$ can be represented as follows:

$$O(\hat{y}(t)) = \left\{(F_n, \hat{\beta}_n(t)), (F, \hat{\beta}_F(t)), n = 1, \dots, N\right\} \tag{18}$$

where $\hat{\beta}_n(t)$ can be obtained by the analytical ER algorithm as follows:

$$\hat{\beta}_n(t) = \frac{\prod_{i=1}^{L}(w_i\hat{\beta}_{n,i}(y_{t-i}) + 1 - w_i + w_i\beta_{F,i}(y_{t-i})) - \prod_{i=1}^{L}(1 - w_i + w_i\beta_{F,i}(y_{t-i}))}{D(t)} \tag{19}$$

$$\hat{\beta}_F(t) = \frac{\prod_{i=1}^{L}(1 - w_i + w_i\beta_{F,i}(y_{t-i})) - \prod_{i=1}^{L}(1 - w_i)}{D(t)} \tag{20}$$

$$D(t) = \sum_{n=1}^{N}\prod_{i=1}^{L}(w_i\beta_{n,i}(y_{t-i}) + 1 - w_i + w_i\beta_{F,i}(y_{t-i})) - (N-1)\prod_{i=1}^{L}(1 - w_i + w_i\beta_{F,i}(y_{t-i})) - \prod_{i=1}^{L}(1 - w_i) \tag{21}$$

where $L = p$, p denotes the number of lagged variables of the prediction model (16).

4.3. Rule based information transformation technique for quantitative data

In the above method, $\beta_{n,i}(y_{t-i}), n = 1, \dots, N; i = 1, \dots, L$ could be generated using various ways, depending on the nature of an attribute and data available such as a qualitative attribute using linguistic values, which is an important characteristic of the ER algorithm. The input information can be one of the following types: continuous, discrete, symbolic and ordered symbolic. In order to facilitate data collection, a scheme for handling various types of input information has been summarized by Yang (2001), Yang et al. (2006b, 2007). In the proposed scheme, there is an important technique, i.e., rule based information transformation technique (Yang, 2001), which is used to deal with the input information which includes qualitative assessment and quantitative data. In this paper, we will only consider the quantitative input. So we first review this technique for quantitative data in this subsection.

Suppose that the input of a quantitative attribute is given by numerical values. In this case, equivalence rules need to be extracted from the decision maker. This can be used to transform a value to an equivalent expectation, thereby relating a particular value to a set of referential values (Yang et al., 2006b). Therefore, a quantitative value $\gamma_j (j = 1, \dots, N)$ can be judged to be a referential value F_j in the ER approach, or

$$\gamma_j \text{ means } F_j, \quad j = 1, \dots, N \tag{22}$$

Suppose that a larger value γ_{j+1} is preferred over a smaller value γ_j. Let γ_N and γ_1 be the largest and smallest feasible values, respectively. Then, an input value y_{t-i} is represented using the following equivalent expectation:

$$S(y_{t-i}) = \{(\gamma_j, \beta_{j,i}(y_{t-i})),\ i = 1, \dots, L;\ j = 1, \dots, N\} \tag{23}$$

where $\beta_{j,i}(y_{t-i})$ can be calculated by

$$\beta_{j,i}(y_{t-i}) = \frac{\gamma_{j+1} - y_{t-i}}{\gamma_{j+1} - \gamma_j} \quad \text{if } \gamma_j \leqslant y_{t-i} \leqslant \gamma_{j+1}, \quad j = 1, \dots, N-1 \tag{24}$$

$$\beta_{j+1,i}(y_{t-i}) = 1 - \beta_{j,i}(y_{t-i}) \quad \text{if } \gamma_j \leqslant y_{t-i} \leqslant \gamma_{j+1}, \quad j = 1, \dots, N-1 \tag{25}$$

$$\beta_{s,i}(y_{t-i}) = 0 \quad \text{for } s = 1, \dots, N,\ s \neq j, j+1 \tag{26}$$

The quantitative attribute, y_{t-i}, may also be a random variable and may not always take a single value but several values with different probabilities. In order to solve this problem, the corresponding rule based information transformation technique has also been proposed by Yang (2001).

4.4. Interpretation of prediction results

In the ER approach, evidence is represented by belief degrees firstly, and then all pieces of evidence are aggregated to obtain the final results. Halpern and Fagin (1992) proposed that there were two useful and quite different ways of interpreting belief functions. The first is that a belief function is interpreted as a generalized probability function and the second is that a belief function is used as a way for representing evidence (Halpern & Fagin, 1992).

Recently, Zhou, Hu, Yang, Xu, and Zhou (2008) extended Halpern's view and proposed that evidence was represented as belief distributions and belief was represented as probability in the ER approach. Furthermore, Zhou et al. draw a conclusion that the probabilistic representation of belief defined on the assessment framework F is the only appropriate representation of belief which acts correctly for evidence combination in the ER approach.

In order to give a reasonable interpretation of the predicted results in the ER-based prediction model, Lemma 1 is given firstly.

Lemma 1. *The probabilistic representation of belief is the only appropriate representation of belief which acts correctly under combination* (Halpern & Fagin, 1992).

In Lemma 1, belief is represented as probability and the combination of evidence represented by belief is referred to as the Dempster's combination rule. Apart from the probabilistic representation of belief, there are also other representation schemas, including Shafer's, Yager's and Smets' representation schemas. Lemma 1 shows that all other representation schemas for belief than the probabilistic representation cannot act correctly for evidence combination. For example, Shafer's representation does not act correctly for evidence combination (Halpern & Fagin, 1992).

Now we will give more Lemmas.

Lemma 2. *In the ER approach, the Dempster's combination rule is adopted, and the aggregated beliefs generated by combining multiple pieces of evidence in the ER approach are also represented as probability* (Zhou et al., 2008).

Lemma 3. *The probabilistic representation of belief defined on the power set F is the only appropriate representation of belief which acts correctly for evidence combination in the ER approach* (Zhou et al., 2008).

Based on Lemmas 1–3, Theorem 1 is given below.

Theorem 1. *According to (18), in the ER-based prediction model, the predicted output can be represented as* $O(\hat{y}(t)) = \{(F_n, \hat{\beta}_n(t)), (F, \hat{\beta}_F(t)), n = 1, \ldots, N\}$. *If* $\hat{\beta}_F(t) = 0$, *then the belief distribution* $O(\hat{y}(t))$ *can be interpreted as a probabilistic distribution function.*

Proof. It is well known that in the probability theory, the magnitude of probability assigned to the whole set F must be one. However, in the most cases, the belief mass assigned to F in evidence theory is not necessity to be one. As such, $\hat{\beta}_F(t) = 0$ is the premise that evidence is represented as belief distributions and belief is represented as probability in the ER approach.

In Section 4.3, from (22)–(26), we can see that $\beta_{F,i}(y_{t-i})$ is always zero, that is $\beta_{F,i}(y_{t-i}) = 0, i = 1, \ldots, p$. Then according to the complete synthesis theorem, proposed by Yang and Xu (2002a), we can easily obtain $\hat{\beta}_F(t) = 0$, this also can get by (20). So the premise $\hat{\beta}_F(t) = 0$ is satisfied.

From Lemma 2, we can conclude that belief is represented as probability on F and the Dempster's combination rule is adopted in an analytical fashion in the ER approach. According to Lemma 3, the probabilistic representation of belief is the only appropriate representation of belief which acts correctly under the ER approach, so the belief distribution $O(\hat{y}(t))$ can be interpreted as a probabilistic distribution function. This proves the Theorem 1.

Remark 1. Theorem 1 shows that, if pieces of evidence represented by belief (the inputs of the ER-based prediction model) are represented as belief distributions or generalized probability distributions, the outputs of the ER-based prediction model under belief outputs (the results of the ER algorithm) are also represented as belief distributions.

Remark 2. A belief distribution defined on F reduces to a conventional probability distribution defined on F if $\hat{\beta}_n(t) \geqslant 0$ hold only for the singleton hypotheses F_n for $n = 1, \ldots, N$, and $\hat{\beta}_F(t) = 0$, further there is $\sum_{n=1}^{N} \hat{\beta}_n(t) = 1$. This means that a conventional probability distribution is a special case of a belief distribution. In other words, a belief distribution is a generalized probability distribution.

Remark 3. According to Theorem 1, in the reliability prediction model, the predicted output $O(\hat{y}(t))$ can be read that the state of system reliability is to be assessed to grade F_n with certain value of probability, which is equal to $\hat{\beta}_n(t)$. That is, the probability assigned to grade F_n is $\hat{\beta}_n(t)$, which represents belief in evidence theory. As such, Theorem 1 facilitates the interpretation of the predicted results and provides a panoramic view about the system's reliability state.

4.5. Utility based construction of the numerical outputs

The aggregated distributed assessment $O(\hat{y}(t)) = \{(F_n, \hat{\beta}_j(t)), (F, \hat{\beta}_F(t)), n = 1, \ldots, N\}$ represents the overall assessment of reliability. It provides a panoramic view about the reliability state at time t, from which one can tell which assessment grades the reliability is assessed to, and what belief degrees are assigned to the defined reliability grades $F_n, n = 1, \ldots, N$. However, if the output value $\hat{y}(t)$ is a numerical value, it is desirable to generate numerical values equivalent to the distributed assessments (18) in a sense. The expected utility theory is used to define such values (Yang & Xu, 2002a; Yang et al., 2006b, 2007). Suppose $u(F_i), i = 1, \ldots, N$ are the utilities of the reliability grade $F_i, i = 1, \ldots, N$, respectively. If for a reliability prediction problem, F_j is preferred to F_i, then we have utility relations $u(F_i) < u(F_j), i < j$. In reality, $u(F_i), i = 1, \ldots, N$ may be estimated using prior expert knowledge or by constructing optimization models. In this paper, we investigate the latter approach.

In order to generate numerical output values equivalent to the distributed assessments (18), maximum, minimum and average utilities are introduced. Suppose the utility of an evaluation grade F_n is $u(F_n), n = 1, \ldots, N$, then the expected utility of the aggregated assessment $O(\hat{y}(t))$ is defined as follows:

$$u(O(\hat{y}(t))) = \sum_{n=1}^{N} \hat{\beta}_n(t) u(F_n) \tag{27}$$

The belief degree $\hat{\beta}_n(t)$ stands for the lower bound of the likelihood that $\hat{y}(t)$ is assessed to F_n, whilst the corresponding upper bound of the likelihood is given by $(\hat{\beta}_n(t) + \hat{\beta}_F(t))$, which leads to the establishment of a utility interval if the assessment is incomplete. Without loss of generality, suppose the least preferred assessment grade having the lowest utility is F_1 and the most preferred assessment grade having the highest utility is F_n. Then the maximum, minimum and average utilities of $\hat{y}(t)$ can be calculated by

$$u_{\max}(O(\hat{y}(t))) = \sum_{n=1}^{N-1} \hat{\beta}_n(t) u(F_n) + (\hat{\beta}_N(t) + \hat{\beta}_F(t)) u(F_N) \tag{28}$$

$$u_{\min}(O(\hat{y}(t))) = (\hat{\beta}_1(t) + \hat{\beta}_F(t)) u(F_1) + \sum_{n=2}^{N} \hat{\beta}_n(t) u(F_n) \tag{29}$$

$$u_{\text{avg}}(O(\hat{y}(t))) = \frac{u_{\max}(O(\hat{y}(t))) + u_{\min}(O(\hat{y}(t)))}{2} \tag{30}$$

If original distribution assessments $O(\hat{y}(t))$ in (18) are all complete, then $\hat{\beta}_F(t) = 0$ and $u_{\min}(\hat{y}(t)) = u_{\max}(\hat{y}(t)) = u_{\text{avg}}(\hat{y}(t))$. Having obtained the outcome shown in (18), the prediction outcome $\hat{y}(t)$ is calculated as follows:

$$\hat{y}(t) = u_{\text{avg}}(O(\hat{y}(t))) = \sum_{n=1}^{n=N} \hat{\beta}_n(t) u(F_n) + \frac{u(F_1) + u(F_N)}{2} \hat{\beta}_F(t) \tag{31}$$

where $\hat{y}(t)$ represents the prediction outcome. If the distributed reliability assessments are complete and precise, then $\hat{\beta}_F(t) = 0$ and the prediction outcome $\hat{y}(t)$ is reduced to

$$\hat{y}(t) = \sum_{n=1}^{N} \hat{\beta}_n(t) u(F_n). \tag{32}$$

The ER approach is a nonlinear aggregation method in nature (Yang & Xu, 2002b). Therefore, the functional relationship between outputs and inputs in the ER-based prediction model is nonlinear in reality.

From (19)–(21) and (31), it can be seen that the attribute weights w_i and the grade utilities $u(F_n)$ play a significant role in the final conclusion $O(\hat{y}(t))$ or $\hat{y}(t)$. The degree to which the final output can be affected is determined by the magnitude of the attribute weights and the grades utilities. On the other hand, if the parameters of the ER-based prediction model such as $w_i, u(F_n), i = 1, \ldots, L, n = 1, \ldots, N$ are not given a *priori* or only known partially or imprecisely, the predicted result will be reliable enough. However, it is difficult to determine the parameters of the ER-based prediction model accurately. In addition, a change in attribute weights or grades utilities may lead to changes in the performance of the ER-based prediction model. As such, there is a need to develop a method that can optimally learn parameters using observed input and output information. This is exactly the topic for the rest of this paper.

5. Optimal learning algorithm for training the ER-based prediction model

For a reliability prediction model discussed in the previous sections, the relative weights of attributes $w_i, i = 1, \ldots, L$, and the utility of assessment grades $u(F_n), n = 1, \ldots, N$ are all unknown in general. In order to train these parameters for a specific system from its input–output pairs, a generic optimization model can be constructed first.

5.1. A generic optimal framework

In this section, how to determine the attributes weights and expected utilities in the ER-based prediction model are investigated. In the ER framework, based on the optimization, the attributes weights and expected utilities of the ER-based prediction model are determined optimally and simultaneously. Fig. 1 shows how to get the additional parameters for dealing with the reliability prediction problems in the ER-based prediction model, where $\mathbf{X}(t)$ is a given input represented as $(y_{t-1}, y_{t-2}, \ldots, y_{t-p})$; $y(t)$ is the corresponding actual observed output, either measured using instruments or assessed by experts; $\hat{y}(t)$ is the simulated output that is generated by the ER-based prediction model, and $\xi(P)$ is the difference between $y(t)$ and $\hat{y}(t)$, as defined later.

It is desirable that $\xi(P)$ is as small as possible where P is the vector of training parameters including $w_i, i = 1, \ldots, L$, and $u(F_n), n = 1, \ldots, N$. This objective is difficult to achieve if the ER-based prediction model is constructed using expert judgments only. Several learning models are designed to adjust the parameters in order to minimize the difference between the observed output$y(t)$and the simulated output $\hat{y}(t)$, i.e., $\xi(P)$. Such an optimally trained ER-based model may then be used to predict the behavior of the system. In general, the optimal learning problem can be represented as the following nonlinear programming problem:

$$\begin{aligned} \min \quad & f(P) \\ \text{s.t.} \quad & A(P) = 0, B(P) > 0 \end{aligned} \tag{33}$$

where $f(P)$ is the objective function, P is the training parameter vector, $A(P)$ is the equality constraint functions, and $B(P)$ is the inequality constraint functions.

In the learning process, a set of observations on the system inputs and outputs is required. In the following, we assume that a set of observation pairs $(\mathbf{X}(t_m), y(t_m)), m = 1, \ldots, M$ is available, where $\mathbf{X}(t_m)$ is an input vector and $y(t_m)$ the corresponding output. Both $\mathbf{X}(t_m)$ and $y(t_m)$ can be either numerical, distributed, or both. The format of the objective function is important for the parameter optimization. Depending on the types of input and output, the optimal learning models can be constructed in different ways, as discussed in detail in the following sections.

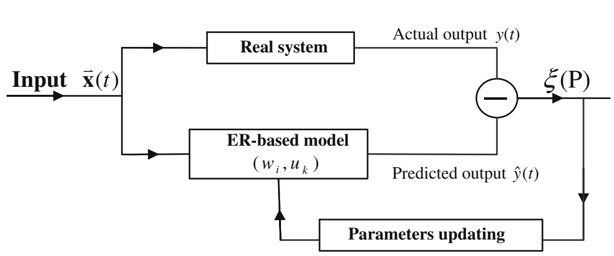

Fig. 1. Illustration of optimal learning process for the ER-based prediction model.

5.2. Optimal learning model based on numerical output

In this case, it is assumed that a set of observed training data is provided in the form of input–output pairs $(\mathbf{X}(t_m), y(t_m))$, $m = 1, \ldots, M$, with $y(t_m)$ being a numerical output value of the actual system at timet_m. The output that is shown in (18) is represented as a distribution, and its average utility is given by

$$\hat{y}(t) = \sum_{n=1}^{n=N} \hat{\beta}_n(t) u(F_n) + \frac{u(F_1) + u(F_N)}{2} \hat{\beta}_F(t) \tag{34}$$

where $\hat{y}(t)$ is considered to be the predicted output at time t from the ER-based prediction model.

For a reliability prediction problem, the relative weight of attribute $w_i, i = 1, \ldots, L$, the utilities of assessment grades $u(F_n), n = 1, \ldots, N$ are all unknown. In order to train and learn these parameters from input–output pairs, the optimization model is constructed to minimize the mean square error (MSE) criterion as follows:

$$\min J = \xi(P) = \frac{1}{M} \sum_{i=1}^{M} (y(t_i) - \hat{y}(t_i))^2 \tag{35}$$

where $y(t_i)$ denotes the actual output data of a system at time t_i and $\hat{y}(t_i)$ denotes the predicted output data of the system at time t_i from the ER-based prediction model, which can be obtained from (31). $w_i, u(F_n), i = 1, \ldots, L, n = 1, \ldots, N$ are the relative weights of attributes and the utilities of reliability assessment grades to be estimated respectively. P is the parameter vector including $w_i, u(F_n), i = 1, \ldots, L, n = 1, \ldots, N$. M is the number of training data in the input–output pairs. $(y(t_i) - \hat{y}(t_i))$ is the residual of training data at time t_i.

The construction of the constraints of the learning model is given as follows.

(1) An attribute weight is normalized, so that it is between zero and one and the total weights will be equal to one, e.g.,

$$0 \leqslant w_i \leqslant 1, \quad i = 1, \ldots, L \tag{36}$$

$$\sum_{i=1}^{L} w_i = 1 \tag{37}$$

(2) For numerical data, the utility of the reliability state evaluation grade is a nonnegative number, e.g.,

$$u(F_n) \geqslant 0, \quad n = 1, \ldots, N \tag{38}$$

And if F_j is preferred toF_i, then

$$u(F_j) > u(F_i) \tag{39}$$

Therefore, all generations of the optimization algorithm are used to obtain the minimal mean square error. To (35), it is a multi-variable constrained nonlinear single-objective optimization problem and can be solved using MATLAB optimization toolbox. Once the weights, the utilities of grades are learned, the ER-based prediction model can be used for reliability prediction.

5.3. Optimal learning model based on belief distribution output

In this case, a set of observed training data is assumed to be composed of M input–output pairs $(\mathbf{X}(t_m), y(t_m)), m = 1, \ldots, M$, with $y(t_m)$ being a belief distribution structure and represented using a distributed assessment with different degrees of belief as follows:

$$O(y(t_m)) = \{(F_n, \beta_n(t_m)), n = 1, \ldots, N\} \tag{40}$$

where F_n is a reliability state evaluation grade of system obtained by the ER-based prediction model, and $\beta_n(t_m)$ is the degree of belief to which F_n is assessed by the mth pair of observed data at time t_m.

Using the same referential terms as for the observed output $O(y(t_m))$, a belief distribution conclusion that is generated by aggregating all the input information can also be represented as follows:

$$O(\hat{y}(t_m)) = \{(F_n, \hat{\beta}_n(t_m)), n = 1, \ldots, N\} \tag{41}$$

where $\hat{\beta}_n(t_m)$ is generated by the ER-based prediction model using (19)–(21) for a given input. It is desirable that, for a given input $\mathbf{X}(t_m)$, the ER-based prediction model can generate an output, which is represented as (41) and can be as close to $O(y(t_m))$ represented as (40) as possible. In other words, for the mth pair of the observed data $(\mathbf{X}(t_m), y(t_m))$, the ER-based prediction model is trained to minimize the difference between the observed belief $\beta_n(t_m)$ and the belief $\hat{\beta}_n(t_m)$ that is generated by the ER-based prediction model for each referential term. Such a requirement is true for all pairs of the observed data. This multi-objective optimization is solved using the FMINIMAX function in MATLAB referred to the minimax problem defined as follows:

$$\begin{aligned} &\min_{Q} \max_{\{\varepsilon_n\}} \quad \{\varepsilon_n(Q), n = 1, \ldots, N\} \\ &\text{s.t.} \quad (36) \text{ and } (37) \end{aligned} \tag{42}$$

where

$$\varepsilon_n(Q) = \frac{1}{M}\sum_{m=1}^{M}(\beta_n(t_m) - \hat{\beta}_n(t_m))^2, \quad n = 1, \ldots, N \tag{43}$$

$(\hat{\beta}_n(t_m) - \beta_n(t_m))$ is the residual at the mth data point, and $\mathbf{Q} = \{w_1, \ldots, w_L\}$ is the training parameter vector without $u(F_n)$ because F_n does not need to be quantified in this case, that is, the training parameters include attribute weights and belief decaying factor only. The constraints are given by (36) and (37). Eq. (42) is an N-objective and multi-variable nonlinear optimization problem.

The minimax method (Coleman, Branch, & Grace, 1999; Marler & Arora, 2004; Yang et al., 2007) is to minimize a worst case objective function. In other words, the purpose of the minimax formulation strategy is to minimize the maximum relative deviation of the objective function from its minimum objective function value. The objective functions are assigned weights, $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_N)$ indicating the designer's subjective preferences, with $0 \leqslant \delta_i \leqslant 1$ and $\sum_{i=1}^{N}\delta_i = 1$. Suppose ξ_j^+ and ξ_j^* are the feasible maximum and minimum values for the jth objective.Then, the objective function can be formulated as follows:

$$\min_{\mathbf{Q}} \max_{j=1,\ldots,N}\{\varphi_j(\mathbf{Q})\} \tag{44}$$

with

$$\varphi_j(\mathbf{Q}) = \frac{\xi_j(\mathbf{Q}) - \xi_j^*}{\xi_j^+ - \xi_j^*} \tag{45}$$

$$\xi_j^+ - \xi_j^* > 0, \quad j = 1, \ldots, N \tag{46}$$

We take inspiration from Yang et al. (2007) and Liu et al. (2008) to solve the nonlinear multi-objective optimization problem above. The computational steps are summarized as follows.

Step 1. Based on the multi-objective function, solve the following single-objective optimization problem with the conventional methods individually (e.g., FMINCON).

$$\begin{aligned} &\min_{\mathbf{Q}} \quad \xi_j(\mathbf{Q}) \\ &\text{s.t.} \quad (36) \text{ and } (37) \end{aligned} \tag{47}$$

where $j = 1, \ldots, N, \xi_j(\mathbf{Q})$ is given by (45), and there are N single-objective and multi-variable nonlinear optimization problems to solve. Suppose ξ_j^* is the optimal solution $j = 1, \ldots, N$.

Step 2. Suppose a relative weight vector $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_N)$ is provided. In this paper all objectives are assumed to have equal weights.

Step 3. The multi-objective optimization problem can be formulated as follows:

$$\begin{aligned} &\min_{\mathbf{Q}} \max_{j=1,\ldots,N} \quad \{\delta_j \cdot \varphi_j(\mathbf{Q})\} \\ &\text{s.t.} \quad (36) \text{ and } (37) \end{aligned} \tag{48}$$

with

$$\varphi_j(\mathbf{Q}) = \frac{\xi_j(\mathbf{Q}) - \xi_j^*}{\xi_j^+ - \xi_j^*} \text{ and } \delta_j = \frac{1}{N}, \quad j = 1, \ldots, N \tag{49}$$

Step 4. Arrange the problem into the standard form in MATLAB, and use the MATLAB function, FMINMAX, to solve it.

In this subsection, The training problem for the belief distribution output is a multiple objective nonlinear optimization problem with N objective that are defined as in (42) and (43), L training parameters as given (w_i) and $2L + 1$ constraints, as defined in (36) and (37).

6. Experiment study

Turbochargers are a critical component in turbo-charged diesel engines. Since reliability is one of the most important considerations in diesel engine system design, an accurate forecast of reliability can provide a good assessment of engine performance and help make correct decision for follow-up maintenance actions.

6.1. Problem description of the Turbochargers reliability prediction problem

In this study, the system reliability data are the same data sets as in Xu et al. (2003). The data sets include the time-to-failure data for 40 suits of turbochargers. When analyzing ungrouped failure data, the cumulative failure distribution can be estimated from generating the median plotting positions for the ith ordered failures (Xu et al., 2003). Since the cumulative failure distribution is skewed for values of i close to zero and close to the sample size n, the reliability estimates $(R(T_i))$ can thus be calculated based on the following formula via Benard approximation (Nelson, 1988; Xu et al., 2003; Chen, 2007).

$$R(T_i) = 1 - \frac{i - 0.3}{n + 0.4} \tag{50}$$

In this study, the reliability data of turbocharger were employed as the data set to train and test the ER-based prediction model.

6.2. The ER-based reliability prediction model for turbochargers

As analyzed above, the system reliability data of turbochargers is a time series. So it is reasonable to assume that the current reliability value y_t is related to the most recent y_{t-1} or even extended into the past values $y_{t-2}, \ldots, y_{t-p}$. This is because the next reliability value is dependent on the current level of system state to a large extent. As such, the proposed ER-based prediction model can be applied to predict turbochargers' reliability. Let $(y_{t-1}, y_{t-2}, \ldots,$

y_{t-p}) be the input vector of the ER-based prediction model, so (16) can be represented as follows:

$$\hat{y}(t) = \hat{y}_t = f(y_{t-1}, y_{t-2}, \ldots, y_{t-p}) \quad (51)$$

where $\hat{y}(t)$ is a scalar representing the predicted turbochargers reliability at time t.

In this study, the reliability data of turbochargers engine system were employed as the data set to train and test the ER-based prediction model. This study is conducted with a relatively larger number $p = 4$ for the order of autoregressive terms. Therefore, we can transform 40 observation values to 36 input patterns. So (51) can be reduced to

$$\hat{y}(t) = \hat{y}_t = f(y_{t-1}, y_{t-2}, y_{t-3}, y_{t-4}) \quad (52)$$

In (52), the original input and output data were all provided as numerical numbers, so there is a need to equivalently transform the numerical value into the belief distribution structures.

6.3. Referential points of turbochargers' reliability state

For the turbochargers' reliability prediction problem, suppose turbochargers' reliability state is classified into several categories like "*High*" (F_1), "*Average*" (F_2), and "*Low*" (F_3). Define the following discernment framework of turbochargers' reliability state as follows:

$$F = \{F_j, j = 1, 2, 3\} = \{\text{High}, \text{Average}, \text{Low}\}$$

All related factors to turbochargers' reliability may then be assessed with reference to this framework using the rule-based information transformation technique. Note that the referential values of an attribute and the types of input information are problem-specific. Thus, their definitions depend on the problems in hand.

Through analyzing the characteristic of turbochargers' reliability data, similar to Yang's method (Yang, 2001; Yang et al., 2007), it is assumed that equivalence rules can be acquired. Three equivalence rules could be acquired as follows:

1. If turbochargers' reliability is 1.0, then system reliability state is ranked to be "*High*" with belief degree 1.0, that is $F_1 = 1.0$.
2. If turbochargers' reliability is 0.75, then system reliability state is ranked to be "*Average*" with belief degree 1.0, that is $F_2 = 0.75$.
3. If turbochargers' reliability is 0.5, then system reliability state is ranked to be "*Low*" with belief degree 1.0, that is $F_3 = 0.5$.

The quantified results above are listed in Table 2 as follows:

Table 2
The referential points of turbochargers' reliability.

Linguistic terms	High (F_1)	Average (F_2)	Low (F_3)
Numerical values	1.0	0.75	0.5

6.4. Illustration of the ER-based prediction model under numerical outputs

For illustration purpose, the first 31 sets of data are used as the training data for parameters estimation. In this case, the output value $\hat{y}(t)$ in (31) is a numerical value. After training, the remaining 5 sets of data are used for testing the trained ER-based prediction model. The initial attribute weights of $y_{t-1}, y_{t-2}, y_{t-3}$ and y_{t-4} are all set to be equivalent. According to (31) and the characteristic of turbochargers' reliability data, in which the maximal value is 0.9930 and the minimal is 0.6045, the initial parameter vector P is set to:

$$P = [w_1, w_2, w_3, w_4, u(F_1), u(F_2), u(F_3)]$$
$$= [0.25, 0.25, 0.25, 0.25, 1.0, 0.8, 0.5]$$

In order to solve nonlinear optimization model (35), FMINCON function in MATLAB is used, the error tolerance is set to 0.000001, and the maximum iteration is set to 60. After training, the trained parameter vector P is obtained as follows:

$$P = [0.2511, 0.2500, 0.2495, 0.2494, 0.9782, 0.7761, 0.4804],$$

where the weight relations $w_1 > w_2 > w_3 > w_4$ denotes that y_{t-1} has the most influence on the prediction results $\hat{y}_t$ and y_{t-4} has the least influence on $\hat{y}_t$, and the utility relations $u(F_1) > u(F_2) > u(F_3)$ shows that F_1 is preferred to F_2 and F_2 is preferred to F_3.

After searching for the optimal parameter sets using the nonlinear optimization model, the ER-based prediction model was built. The forecasting simulation was performed against the testing data. The graphical comparisons between the actual and predicted reliability are shown in Fig. 2. We can see that the predicted and actual values of the system reliability are matched closely and relative prediction errors in the data sets are small. As such, we can say that the proposed ER-based prediction model fits this particular data set very well.

6.5. Comparison with the existing methods

It is of interest to compare the results generated by the ER-based prediction model with the existing reliability forecasting methods. In this section, the traditional ARIMA models, the neural network approaches, the SVM model and the support vector regression model with genetic algorithm (GA-SVR) are used for the study, in comparison with the experiment results given by Xu et al. (2003), Chang et al. (2004), Hong and Pai (2006) and Chen (2007).

Table 3 shows that the forecasting performances from the ER-based model, GA-SVR and various neural network approaches are all excellent. The mean absolute percentage error (MAPE) is used to assess the prediction accuracy. Small MAPE values indicate small deviations between the predicted and actual values. In this study, the results generated by the ER-based model and GA-SVR model are superior to those by the other models. Furthermore, the difference in terms of prediction precision between the ER-based model and GA-SVR model is small. However, in the experiment, the training process of GA-SVR model is time-consuming and takes about 67.4 s for model training in this study. On the other hand, our modeling process is more efficient and the training

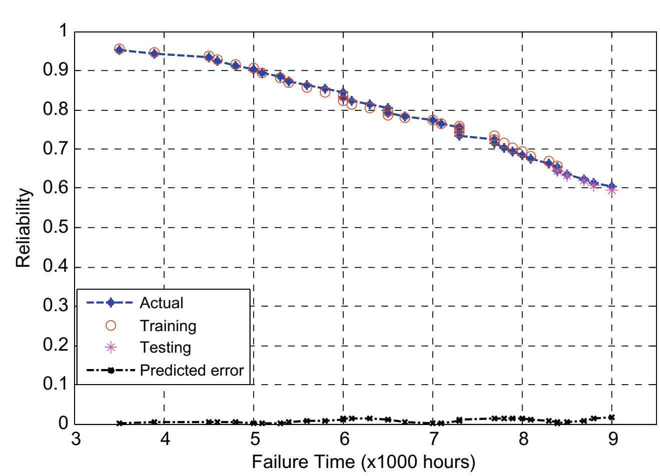

Fig. 2. Reliability prediction results with the ER-based prediction model.

Table 3
Comparison of the prediction results.

No.	Actual	ER-based	GA-SVR[a]	SVM[b]	NF[c]	GRNN[d]	RBF[e]	ARIMA[f](1,0,0)
36	0.6444	0.6445	0.6446	0.6459	0.6845	0.6533	0.6466	0.6460
37	0.6345	0.6348	0.6346	0.6370	0.6344	0.6389	0.6369	0.6392
38	0.6245	0.6239	0.6248	0.6281	0.6236	0.6270	0.6270	0.6338
39	0.6145	0.6131	0.6148	0.6195	0.6163	0.6049	0.6170	0.6294
40	0.6046	0.6025	0.6049	0.6111	0.6070	0.5989	0.6072	0.6256
MAPE (%)		0.1500	0.0387	0.6200	0.2972	0.9960	0.3914	1.6753

[a] GA-SVR: support vector regression with genetic algorithms.
[b] SVM: support vector machine with Gaussian activation.
[c] NF: neural fuzzy network.
[d] GRNN: generalized regression neural network.
[e] RBF: radial basis function neural network with Gaussian activation.
[f] ARIMA: autoregressive integrated moving average.

time is about 2.2 s. This is very important when there is high real-time requirement for system reliability forecasting.

Based on the above analysis and comparison, it shows that the prediction performance of the ER-based prediction model outperforms some existing prediction methods in terms of prediction accuracy and speed. The experimental results demonstrate the potential of the ER-based model in the reliability forecasting field.

6.6. Simulation results based on belief distribution output

For illustration purpose, the first 31 sets of data are used as the training data for parameter estimation. In this study, the output value $\hat{y}(t)$ in (52) is a belief distribution structure through the rule based transformation similar to Section 4.3. After training, all the 36 sample data are used for testing the trained ER-based prediction model. The initial parameter vector $\mathbf{Q}$ is given by experts. Suppose that the initial attribute weights are all set to be equivalent. For the belief distribution output, it is unnecessary to initialize the utilities of the reliability assessment grades$u(F_i), i = 1, 2, 3$, so the initial parameter vector $\mathbf{Q}$ is set to

$$\mathbf{Q} = [w_1, w_2, w_3, w_4] = [1/4, 1/4, 1/4, 1/4]$$

For the belief distribution output, there is a need to solve the multiple objective nonlinear optimization problems that are defined by (42). To solve the multiple objective optimization problems, FMINIMAX function in MATLAB is used. In this example, the error tolerance is set to 0.000001, and the maximum iteration is set to 60 to avoid dead loop in the optimal learning process. Applying the method presented in Section 5.3, the ER-based prediction model can be trained step by step with sequential input–output data. After training, the trained parameter vector $\mathbf{Q}$ can be obtained as follows:

$$\mathbf{Q} = [0.8021, 0.1030, 0.0655, 0.0294]$$

where we can see that y_{t-1} has the most influence on the prediction result $O(\hat{y}(t))$, and the larger the attribute weights, the greater influence they have on the prediction results.

The test results are illustrated in Fig. 3–5, where the comparisons are shown between the actual output and the predicted output that is generated using the trained ER-based prediction model. As shown in Fig. 3–5, it is obvious that the predicted outcomes generated by the trained ER-based prediction model can trace the changes of the system reliability. Adopting Theorem 1, we can give a clear interpretation of the predicted outputs and assessed results. From Fig. 1, it can be see that the belief degree assigned to grade "*High*" decrease from 0.8166 to 0 monotonously. According to Theorem 1, we can say that the probability of system reliability assigned to grade "*High*" decreases from 0.8166 to 0. At the same time, the probability of system reliability assigned to grade "*Average*" firstly increases monotonously from 0.1834 to 0.9833, then decreases to 0.3832. The probability of system reliability assigned to grade "*Low*" increases monotonously from 0 to 0.6168. The above analysis shows that system reliability is experiencing the process of degradation over time and the proposed predicted method can give complete information about the state of system reliability. The detailed prediction results under this case can be found in Appendix A. As marked in gray in Appendix A, the belief

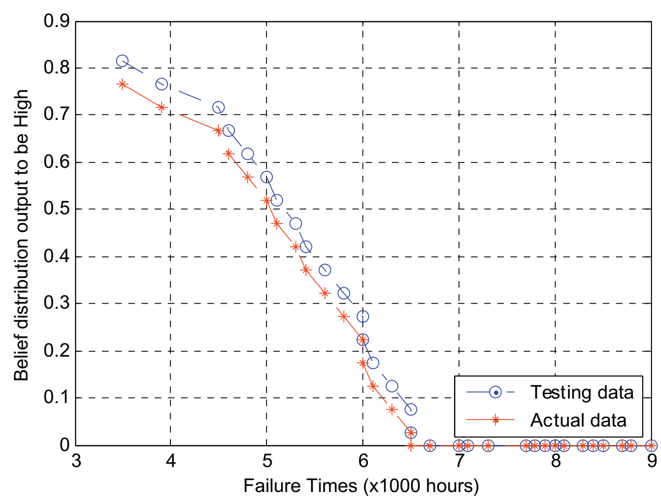

Fig. 3. Belief distribution output to be "High".

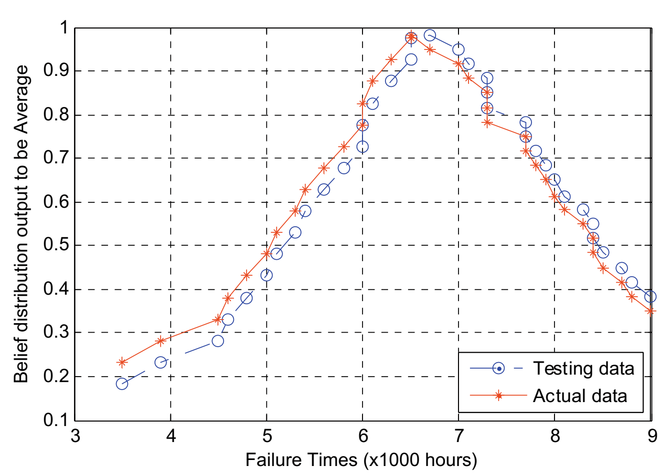

Fig. 4. Belief distribution output to be "*Average*".

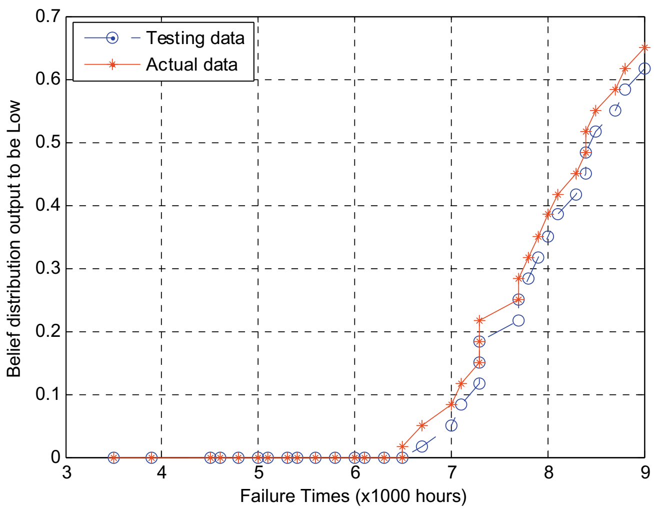

Fig. 5. Belief distribution output to be "*Low*".

degree assigned to grade "*Low*" in the first seventeen prediction results is zero, which is caused by $\beta_{3,i}(y_{t-i}) = 0$ in the input information of the ER-based prediction model, according to the independent synthesis theorem proposed by Yang and Xu (2002a). Similarly, the belief degree assigned to grade "*High*" in the remainder is zero, which is caused by $\beta_{1,i}(y_{t-i}) = 0$.

In order to study the performance of the prediction model, we choose the mean squared error (MSE) which is defined as $\|O(y(t_m)) - O(\hat{y}(t_m))\|^2$ to measure the prediction accuracy, where $\|\cdot\|$ is Euclidean norm. As shown in Fig. 6, if the appropriate initial parameters are chosen, the estimates generated by the proposed algorithm converge to the true values very quickly.

In order to further demonstrate the proposed ER-based prediction method under belief distribution outputs, the MAPE and RMSE between the observed values and the predicted belief degrees to turbochargers reliability state "*High*", "*Average*" and "*Low*" respectively are used to evaluate the prediction accuracy. The calculated results are listed in Table 4 as follows.

Obviously, the predicted outcomes of the ER-based prediction method match and trace the observed ones by and large. However, the prediction precision under belief distribution outputs is a bit lower than that under numerical outputs. The main cause is that, in determining model parameters (attributes weight or grade utilities), the optimization model is single-objective under numerical outputs while multi-objective optimization under belief distribution outputs. In reality, to solve multi-objective optimization problem is much more difficult than to solve single-objective optimization problem. Practically, it is very difficult to obtain the general optimal solution for multi-objective optimization problems, that is, the obtained solution is a local solution. As a result, the imprecise parameters result in the low prediction precision. As such, in order to improve the prediction performance, it could be the further research applying more efficient optimization algorithms to select the parameters of the ER-based prediction model.

The above results show that the ER-based prediction method can be used not only to effectively solve numerical output problems such as that represented in Section 6.5, but also to deal with prediction problems involving output values in the form of belief distribution structure through solving multiple objective optimization problems.

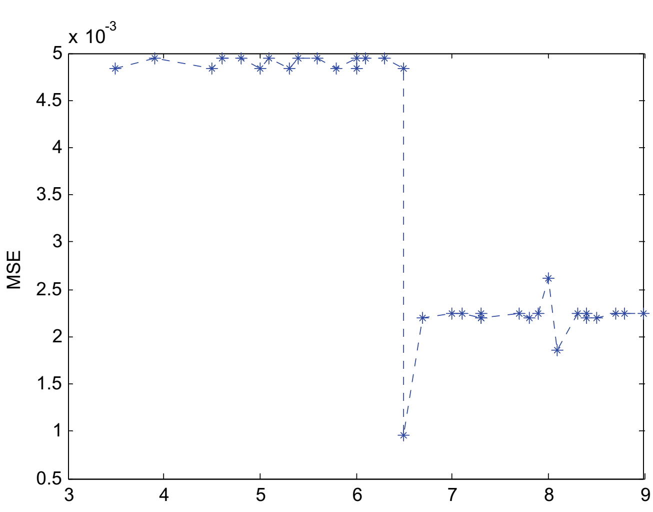

Fig. 6. MSE between the true outputs and the predicted outputs.

Table 4
The prediction errors.

	High	Average	Low
MAPE	0.3589	0.0777	1.0514
RMSE	0.0332	0.0410	0.0244

7. Conclusion

An accurate and effective system reliability prediction model can not only learn and track a system or product's reliability and operational performance, but also offer useful information for managers to take follow-up actions timely to improve the performance of the system. In this paper, a novel reliability prediction technique is developed and applied to forecast reliability in turbochargers engine systems. From this research, the following conclusions can be drawn.

1. A novel reliability prediction technique that can be used to predict the reliability of turbochargers engine systems has been developed, tested, and validated. The new method that is discussed is based on the ER algorithm.
2. Compared with several existing methods, the proposed method is capable of generating better results in predicting future reliability of repairable systems in terms of prediction accuracy and speed.
3. The simulation results show that the ER-based prediction method can be used not only to effectively solve numerical output problems, but also to deal with prediction problems involving output values in the form of belief distribution structure through solving multiple objective optimization problems.

Table A
Reliability of turbocharger (Xu et al., 2003).

i	*R(Ti)*	*i*	*R(Ti)*	*i*	*R(Ti)*	*i*	*R(Ti)*
1	0.9930	11	0.8934	21	0.7938	31	0.6942
2	0.9831	12	0.8835	22	0.7839	32	0.6834
3	0.9731	13	0.8735	23	0.7739	33	0.6743
4	0.9631	14	0.8635	24	0.7639	34	0.6643
5	0.9532	15	0.8536	25	0.7540	35	0.6544
6	0.9432	16	0.8436	26	0.7440	36	0.6444
7	0.9333	17	0.8337	27	0.7341	37	0.6345
8	0.9233	18	0.8237	28	0.7241	38	0.6245
9	0.9133	19	0.8137	29	0.7141	39	0.6145
10	0.9034	20	0.8038	30	0.7042	40	0.6045

Table B
The prediction results with the ER-based prediction model.

Grade/number	1	2	3	4	5	6	7	8	9	10
High	0.8166	0.7674	0.7177	0.6685	0.6188	0.5691	0.5191	0.4702	0.4210	0.3713
Average	0.1834	0.2326	0.3315	0.3315	0.3812	0.4309	0.4309	0.5298	0.5790	0.6287
Low	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Grade/number	11	12	13	14	15	16	17	18	19	20
High	0.3216	0.2724	0.2227	0.1735	0.1238	0.0741	0.0249	0.0000	0.0000	0.0000
Average	0.6784	0.7276	0.7773	0.8265	0.8762	0.9259	0.9751	0.9833	0.9501	0.9167
Low	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0167	0.0499	0.0833
Grade/number	21	22	23	24	25	26	27	28	29	30
High	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Average	0.8832	0.8501	0.8166	0.7835	0.75	0.7165	0.6834	0.6499	0.6138	0.5833
Low	0.1168	0.1499	0.1834	0.2165	0.25	0.2835	0.3166	0.3501	0.3862	0.4167
Grade/number	31	32	33	34	35	36				
High	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000				
Average	0.5499	0.5167	0.4833	0.4501	0.4167	0.3832				
Low	0.4501	0.4833	0.5167	0.5499	0.5833	0.6168				

4. Although the reliability prediction model tested in this paper is focused on turbochargers engine systems, the proposed method can be applied to different problem domains.
5. The methodology presented in this paper can be improved. The exact characteristics of the model and the details of its implementation have not been fully explored. For example, the influence of the number of input nodes is not analyzed and the number of input nodes is entirely determined using expert knowledge, and recursive training models can be developed for online reliability prediction. Another direction for the further research could be to apply more efficient optimization algorithms to select the parameters of the ER-based prediction model to improve prediction performance, such as genetic algorithm or simulated annealing.

Acknowledgements

The work was supported by the National Science Foundation of China (NSFC) under Grant No. 60736026 and National High Technology Research Development Program (863 program) of China under Grant No. 2008AAJ112.

Appendix A

Tables A and B.

References

Ascher, H., & Hung, S. (1987). *Repairable system reliability*. New York: Marcel Dekker Inc..
Amjady, N., & Ehsan, M. (1999). Evaluation of power systems reliability by artificial neural network. *IEEE Transactions on Power Systems, 14*, 287–292.
Aljahdali, S., Sheta, A., & Rine, D. (2001). Prediction of software reliability: A comparison between regression and neural network non-parametric models. In *Proceedings of the ACS/IEEE international conference on computer systems and applications*.
Box, G. E., & Jenkins, G. M. (1970). *Time series analysis forecasting and control*. San Francisco: Holden-Day.
Coleman, T., Branch, M. A., & Grace, A. (1999). *Optimization toolbox-for use with Matlab*. The Mathworks Inc.
Cai, K., Cai, L., Wang, W., Yu, Z., & Zhang, D. (2001). On the neural network approach in software reliability modeling. *The Journal of Systems and Software, 58*, 47–62.
Chang, P. T., Lin, K. P., & Pai, P. F. (2004). Hybrid learning fuzzy neural models in forecasting engine systems reliability. In *Proceeding of the fifth Asia Pacific industrial engineering and management systems conference* (pp. 2361–2366). Australia: Gold Coast.
Costa, E., Vergilio, S., Pozo, A., & Souza, G. (2005). Modeling software reliability growth with genetic programming. In *Proceedings of the 16th IEEE international symposium on software reliability engineering (ISSRE'05)*.
Chen, K. Y. (2007). Forecasting systems reliability based on support vector regression with genetic algorithms. *Reliability Engineering and System Safety, 92*, 423–432.
Dempster, A. P. (1967). Upper and lower probabilities induced by a multi-valued mapping. *Annals Mathematical Statistics, 38*, 325–339.
El-Aroui, M., & Soler, J. (1996). A Bayes nonparametric framework for software reliability analysis. *IEEE Transactions on Reliability, 45*, 652–660.
El-Sebakhy, E. A. (2008). Software reliability identification using functional networks: A comparative study. *Expert Systems with Applications, 36*, 4013–4020.
Gran, B. A., & Helminen, A. (2001). A Bayesian belief network for reliability assessment. *Safecomp, 2187*, 35–45.
Halpern, J. Y., & Fagin, R. (1992). Two views of belief: Belief as generalized probability and belief as evidence. *Artificial Intelligence, 54*(3), 275–317.
Ho, S. L., & Xie, M. (1998). The use of ARIMA models for reliability forecasting and analysis. *Computer & Industrial Engineering, 35*(1/2), 213–216.
Ho, S. L., Xie, M., & Goh, T. N. (2002). A comparative study of neural network and Box–Jenkins ARIMA modeling in time series prediction. *Computer & Industrial Engineering, 42*, 371–375.
Hong, W. C., & Pai, P. F. (2006). Predicting engine reliability by support vector machine. *International Journal of Manufacture Technology, 28*, 154–161.
Kimura, M., Yamada, S., & Osaki, S. (1995). Statistical software reliability prediction and its applicability based on mean time between failures. *Mathematical and Computer Modeling, 22*, 149–155.
Liu, M. C., Kuo, W., & Sastri, T. (1995). An exploratory study of neural network approach for reliability data analysis. *Quality Reliability Engineering Intelligence, 11*, 107–112.
Lyu, M. (1996). *Handbook of software reliability engineering*. New York: McGraw-Hill.
Lu, S., Lu, H., & Kolarik, W. J. (2001). Multivariate performance reliability prediction in real-time. *Reliability Engineering and Systems Safety, 72*, 39–45.
Liang, Y. H., & Tong, L. I. (2001). Analyzing and forecasting the field failure data for repairable systems. *Journal of Quality, 8*(2), 75–96.
Liu, J., Yang, J. B., Wang, J., Sii, H. S., & Wang, Y. M. (2004). Fuzzy rule-based evidential reasoning approach for safety analysis. *International Journal of General Systems, 33*(2/3), 183–204.
Liang, T., & Afzel, N. (2005). Evolutionary neural network modeling for software cumulative failure time prediction. *Reliability Engineering and System Safety, 87*, 45–51.
Liang, T., & Afzel, N. (2006). Reply to "comment on 'evolutionary neural network modeling for software cumulative failure time prediction' by Liang Tian and Afzel Noore", by Bev Littlewood. *Reliability Engineering and System Safety, 91*, 487–491.
Yi-Hui, L. (2007). Evolutionary neural network modeling for forecasting the field failure data for repairable systems. *Expert Systems with Applications, 33*, 1090–1096.
Liu, J., Yang, J. B., Ruan, D., Martinez, L., & Wang, J. (2008). Self-tuning of fuzzy belief rule bases for engineering system safety analysis. *Annals Operational Research, 163*(2), 143–168.
Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization, 26*, 369–395.
MIL-HDBK-189 (1981). *Reliability growth management*. Department of Defense, Washington, USA.
Nelson, W. (1988). *Applied life data analysis*. New York: Wiley.
Pham, L., & Pham, H. (2000). Software reliability models with time dependent hazard function based on Bayesian approach. *IEEE Transactions on Systems, Man, and Cybernetics Part A, 30*, 25–35.
Pham, L., & Pham, H. (2001). A Bayesian predictive software reliability model with pseudo-failures. *IEEE Transactions on Systems, Man, and Cybernetics Part A, 31*, 233–238.
Pai, P. F., & Hong, W. C. (2006). Software reliability forecasting by support vector machines with simulated annealing algorithms. *The Journal of Systems and Software, 79*, 747–755.
Pai, P. F. (2006). Systems reliability forecasting based on support vector machines with genetic algorithms. *Mathematical and Computer Modelling, 43*, 247–262.

Pham, H., & Zhang, X. (2003). NHPP software reliability and cost models with testing coverage. *European Journal of Operational Research, 145*, 443–454.
Shafer, G. (1976). *A mathematical theory of evidence*. New Jersey: Princeton University Press.
Singpurwala, N. D. (1978). Estimating reliability growth using time series analysis. *Naval Research Logistics Quarterly, 25*, 1–14.
Singh, N., Chrys, D. A., & Frishwick, P. A. (1994). Forecasting time-dependent failure rates of systems operating in series and/or in parallel. *Microelectronic Reliability, 34*(3), 391–400.
Saranga, H., & Knezevic, J. (2001). Reliability prediction for condition-based maintained systems. *Reliability Engineering and Systems Safety, 71*, 219–224.
Tong, L. I., & Liang, Y. H. (2005). Forecasting field failure data for repairable systems using neural networks and SARIMA model. *International Journal of Quality and Reliability Management, 22*(4), 410–420.
Wang, J., Yang, J. B., & Sen, P. (1995). Safety analysis and synthesis using fuzzy sets and evidential reasoning. *Reliability Engineering and Systems Safety, 47*, 103–118.
Wang, J., Yang, J. B., & Sen, P. (1996). Multi-person and multi-attribute design evaluations using evidential reasoning based on subjective safety and cost analysis. *Reliability Engineering and Systems Safety, 52*, 113–128.
Wang, Y. M., Yang, J. B., & Xu, D. L. (2006). Environmental impact assessment using the evidential reasoning approach. *European Journal of Operational Research, 174*(3), 1885–1913.
Wang, Y. M., & Elhag, T. M. S. (2007). A comparison of neural network, evidential reasoning and multiple regression analysis in modeling bridge risks. *Expert Systems and Applications, 32*, 336–348.
Wang, W. (2007). An adaptive predictor for dynamic systems forecasting. *Mechanical Systems and Signal Processing, 21*, 809–823.
Wang, Y. M., & Elhag, T. M. S. (2008). Evidential reasoning approach for bridge condition assessment. *Expert Systems with Applications, 34*, 689–699.
Xie, M. (1991). *Software reliability modeling*. Singapore: World Scientific.
Xu, K., Xie, M., Tang, L. C., & Ho, S. L. (2003). Application of neural networks in forecasting engine systems reliability. *Applied Soft Computing, 2*(4), 255–268.
Xu, D. L., Liu, J., Yang, J. B., Liu, G. P., Wang, J., Jemkinson, I., et al. (2007). Inference and learning methodology of belief-rule-based expert system for pipeline leak detection. *Expert Systems and Applications, 32*(1), 103–113.
Yang, J. B., & Singh, M. G. (1994). An evidential reasoning approach for multiple-attribute decision making with uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics, 24*(1), 1–18.
Yang, J. B. (2001). Rule and utility based evidential reasoning approach for multi-attribute decision analysis under uncertainties. *European Journal of Operational Research, 131*(1), 31–61.
Yang, J. B., & Xu, D. L. (2002a). On the evidential reasoning algorithm for multiple attribute decision analysis under uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans, 32*(2), 289–304.
Yang, J. B., & Xu, D. L. (2002b). Nonlinear information aggregation via evidential reasoning in multiattribute decision analysis under uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, 32*(3), 376–393.
Yang, J. B., Wang, Y. M., Xu, D. L., & Chin, K. S. (2006a). The evidential reasoning approach for MADA under both probabilistic and fuzzy uncertainties. *European Journal of Operational Research, 171*(1), 309–343.
Yang, J. B., Liu, J., Wang, J., Sii, H. S., & Wang, H. W. (2006b). Belief rule-base inference methodology using the evidential reasoning approach – RIMER. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, 30*(2), 266–285.
Yang, J. B., Liu, J., Xu, D. L., Wang, J., & Wang, H. W. (2007). Optimization models for training belief-rule-based systems. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, 37*(4), 569–585.
Zhou, Z. J., Hu, C. H., Yang, J. B., Xu, D. L., & Zhou, D. H. (2008). New recursive algorithms for updating belief-rule-base systems. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*. Under review.
Zheng, J. (2009). Predicting software reliability with neural network ensembles. *Expert Systems with Applications, 36*, 2116–2122.
Zio, E. (2009). Reliability engineering: Old problems and new challenges. *Reliability Engineering and System Safety, 94*, 125–141.