Research Article

Reliability Assessment of CNC Machining Center Based on Weibull Neural Network

Zhaojun Yang, Chao Chen, Jili Wang, and Guofa Li

School of Mechanical Science and Engineering, Jilin University, Changchun 130025, China

Correspondence should be addressed to Jili Wang; wangjili1100@126.com

Received 23 July 2015; Revised 24 October 2015; Accepted 28 October 2015

1. Introduction

Common life distributions, like normal distribution, log-normal distribution, and Weibull distribution, usually are simple for system reliability modeling [1, 2]. However, CNC machining centers are complex repairable systems in which reliability distribution could not be responded precisely by these simple life distribution models. Mixture distribution has been used popularly during the development process of modern statistics. The application of mixture distribution could trace back to the late 19th century, while Weibull mixture distribution started in 1950s [3–5]. At present, the most common Weibull mixture distribution is twofold Weibull distribution [6, 7]. Multifold Weibull mixture distribution has been seldom used so far. There are two reasons for this: (1) it is hard to estimate large number of parameters, and (2) its bad generalization performance makes it difficult to avoid overfitting.

With the rapid development of computer technology, artificial neural networks (ANNs), as machine learning model with powerful nonlinear approximation ability, have been developed and get wide applications [8–10]. It is often used to deal with the nonlinear relationship between input and output of complex system [11]. However, ANNs easily bring overfitting phenomenon which is a hot topic and attracts many researchers [12, 13]. Improving the generalization performance of artificial neural networks is a key point to solve overfitting problem.

In this paper, Weibull neural network (WNN) is defined based on some advantages of Weibull mixture distribution and artificial neural networks. In this network, hierarchical structure of radical-basis function network (RBF), which has simple structure and powerful nonlinear approximation performance [14, 15], is adopted. RBF was proposed by Moody and Darken [16, 17] with three layers, namely, input layer, hidden layer, and output layer. The input layer is a series of source nodes that connects the networks to reliability data of CNC machining centers. The hidden layer applies a finite Weibull mixture distribution model connecting the input layer and the output layer. The output layer is the probability density of the data. Finite Weibull mixture distribution [7] is applied as hidden layer nodes function (HLNF). Wide application and multiple distribution curve shape are the main characteristics of finite Weibull mixture distribution which suits not only the life distributions of electronic products, but also the life distributions of mechanical parts.
This paper will be focused on WNN’s two key issues: to develop an efficient learning method and to improve the generalization performance of WNN. And the rest of the paper is organized as follows: a definition of Weibull neural network (WNN) is given in Section 2 with the introduction of basic characteristics of artificial neural networks and Weibull mixture distribution. Section 3 presents the learning process of Weibull neural network (WNN). Section 4 offers field test data of CNC machining centers and applies them into comprehensive simulations and reliability assessment by WNN. For comparison, authors also analyze the data by two-parameter Weibull distribution (TPWD). Finally, conclusions are given in Section 5.

2. Weibull Neural Network

2.1. Artificial Neural Networks. Artificial neural networks (ANNs) [18] are the abstraction and simulation of certain basic characteristics of biological neural networks. As complex nonlinear approximation mathematical models, ANNs rely on the complexity of network structure by adjusting the internal connections between nodes and then achieve the purpose of training and learn any complex nonlinear relationship with strong robustness and fault tolerance [19].

Hierarchical structure is the most common structure of ANNs as BP neural network [20], RBF neural network [14, 15], and so on. Hierarchical structure of ANNs can be divided into several layers by function, such as the input layer, an intermediate layer (also called hidden layer), and the output layer. Each layer is connected in order, as shown in Figure 1. The input layer is responsible for receiving input information from the outside and transfers the information to the neurons of hidden layer. A neuron is an information processing unit which is the fundamental of neural networks. Neuron model constituted different transformation functions with various information processing abilities. The hidden layer is internal information processing layer of neural network, responsible for information conversion. According to required information processing capacity, the hidden layer may be designed as one or more layers. The final one, output layer, supplies the response of neural network to the activation pattern (signal) which is applied to the input layer.

Under the external stimulus of input samples, neural network continuously changes connection weights of network as well as topology structure, so that the output of network is close to desired output. The above process is called learning progress of neural network. In this progress, adjustments and changes of connection weights need to follow certain rules called learning rule.

2.2. Weibull Mixture Distribution. Weibull distribution, including two-parameter type and three-parameter type, is the most common life distribution, which is originally proposed by Swedish physicist Waloddi Weibull, for studying the life of components [21]. In practical application, Weibull distribution is often used as the basic model of more complex distributions, such as Weibull mixture distribution [7], Weibull competing risk distribution [22], Weibull parallel distribution, and Weibull segmentation distribution [23]. Among them, Weibull mixture distribution is the most widely used.

In many cases, a sample population may be composed of two or more subsamples. Because of the difference of design methods, raw materials, manufacturing processes, and other aspects of reasons, products may follow different life distributions in different conditions. If the sample population is composed of n subsamples, corresponding subsample cumulative distribution function is expressed as \( F_1(t), F_2(t), \ldots, F_n(t) \), respectively, probability density is \( f_1(t), f_2(t), \ldots, f_n(t) \), and corresponding mixture weight of subsample is indicated as \( \alpha_1, \alpha_2, \ldots, \alpha_n \), respectively; then the cumulative mixture distribution function of sample population is shown below:

\[
F(t) = \alpha_1 F_1(t) + \alpha_2 F_2(t) + \cdots + \alpha_n F_n(t) .
\]

Corresponding probability density function for the mixture distribution is shown below:

\[
f(t) = \alpha_1 f_1(t) + \alpha_2 f_2(t) + \cdots + \alpha_n f_n(t) .
\]

The general form of Weibull mixture distribution is (2), which is called n-fold Weibull mixture distribution. n-fold refers to any n distinct subsamples. Mixture weights should satisfy the equation: \( \sum_{i=1}^{n} \alpha_i = 1 \) and \( 0 < \alpha_i < 1 \).

If \( F_i(t) \) is two-parameter Weibull cumulative distribution with shape parameter \( \beta_i \) and scale parameter \( \eta_i \), or three-parameter Weibull cumulative distribution with shape parameter \( \beta_i \), scale parameter \( \eta_i \), and positional parameter \( \xi_i \), two-parameter Weibull cumulative distribution is shown below:

\[
F_i(t) = 1 - \exp \left[ -\left( \frac{t}{\eta_i} \right)^{\beta_i} \right] .
\]
Three-parameter Weibull cumulative distribution is shown below:

\[ F_i(t) = 1 - \exp \left[ - \left( \frac{(t - \xi_i)}{\eta_i} \right)^{\beta_i} \right]. \] (4)

The most important feature of mixture Weibull distributions is the diversity in shape. Taking twofold mixture Weibull distribution with two parameters, for example, there are four basic types of density functions shape, as shown in Figure 2. Therefore, it is significant to research on the nonlinear approximation performance of mixture Weibull distribution.

2.3. Weibull Neural Network Model. In practical application, the mixture Weibull distributions have two significant limitations: difficult to precisely estimate parameters and hard to choose a suitable folds number. However, the folds affect generalization performance of mixture Weibull distributions seriously. Low folds lead to underfitting; however, high folds result in overfitting. To improve the generalization performance of mixture Weibull distributions, Weibull neural network is proposed in this paper. Weibull neural network is a kind of mixture distribution but is different from traditional mixture Weibull distributions in structure and learning process.

The hierarchical structure of Weibull neural network is identical to radical-basis function network (RBF), including the three layers shown in Figure 3. The input layer is made up of source nodes that connect the network to reliability data. The hidden layer connects the input layer and the hidden layer and finite Weibull mixture distribution is used as hidden layer function in the network. The output layer is the probability density of the data. The connection between input layer and hidden layer is the probability, namely, that each input data and the hidden layer node are connected in a certain probability. There is a linear weighted connection between the hidden layer and output layer.

In the hierarchical structure of Weibull neural network (WNN), the input of network is expressed as \( T = [t_1, t_2, \ldots, t_N] \), where \( N \) is the number of input nodes.
this paper, the hidden layer nodes function (HLNF) is finite Weibull mixture distribution expressed as $\varphi_k(t)$. The output of Weibull neutral network (WNN) is expressed as $Y = [y_1, y_2, \ldots, y_Q]$, where $Q$ is the number of output data. Connection probability between input layer and hidden layer is expressed as $p_{jk}$ which means the probability of the $j$th data sampled by the $k$th node. The process is random sampling with replacement, so the value of connection probability $p_{jk}$ is $1/N$. The connection weight between hidden layer and output layer is expressed as $\omega_{jk}$ which means the weight of the $k$th node to the $l$th output $y$. According to the characteristic of mixture distribution, $\omega_{jk}$ needs to satisfy

$$\sum_{k=1}^{M} \omega_{jk} = 1,$$  \hspace{1cm} (5)

where $M$ is the number of the hidden layer nodes. As every node has the same weight, $\omega_{jk} = 1/M$.

As shown in Figure 3, the input layer achieves nonlinear mapping from input data to hidden layer nodes function (HLNF) $\varphi_k(t)$, while the output layer achieves linear mapping from hidden layer nodes function (HLNF) $\varphi_k(t)$ to output data $Y$. The mathematical model is shown below:

$$y_l = \sum_{k=1}^{M} \omega_{jk} \varphi_k(t) \quad k = 1, 2, \ldots, M. \hspace{1cm} (6)$$

According to the finite Weibull mixture distribution $f_l(t)$ shown in (7), the hidden layer nodes function (HLNF) is calculated by (8). Consider

$$f_l(t) = \left( \frac{\beta_l}{\eta_l} \right) \left( \frac{t}{\eta_l} \right)^{\beta_l-1} \exp \left[ -\left( \frac{t}{\eta_l} \right)^{\beta_l} \right],$$

$$\varphi_k(t) = \alpha_1 f_1(t) + \cdots + \alpha_n f_n(t),$$ \hspace{1cm} (7)

where $n$ is the folds number of mixture Weibull distributions, $i = 1, 2, \ldots n$.

3. Parameter Estimation of Weibull Neural Network (WNN)

As shown above, in order to define the hidden layer, we need to find the hidden layer nodes function (HLNF) $\varphi(t)$ and make sure of the number of hidden layer nodes. Then Weibull neural network (WNN) can be finally proposed. This process is called parameter estimation of WNN which could be divided into three steps. They are parameter estimation of HLNF, selection of HLNF, and determining the number of hidden layer nodes. The following is the detailed process.

### 3.1. Parameter Estimation of HLNF

Expectation maximization (EM) algorithm is widely used in mixture distribution to estimate parameters [24]. Because of the independence of learning process of neuron function, EM algorithm can be used to estimate parameters of neuron function separately [25]. EM algorithm is an iterative algorithm based on maximum likelihood estimation, each iteration is divided into two steps, namely, expectation step (E-step) and maximization step (M-step). E-step is to calculate expectations of likelihood function and estimate parameters; M-step is to maximize the expectations. According to input data, the core idea of EM algorithm is to estimate parameters by the iterations of expectations. The whole EM algorithm steps are as follows.

**Step 1.** Initialize parameters: means $\bar{\mu}_1, \bar{\mu}_2, \ldots, \bar{\mu}_n$, variances $\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2$, and mixture weights $\omega_1, \omega_2, \ldots, \omega_n$.

**Step 2.** E-step: calculate responsivity $y_j$ according to

$$y_j^{(l+1)} = \frac{\omega_j^{(l)} f(t_j)}{\sum_{k=1}^{n} \omega_k^{(l)} f(t_j)} \quad j = 1, 2, \ldots, N. \hspace{1cm} (9)$$

**Step 3.** M-step: calculate weight $\omega_k$, mean $\bar{\mu}_k$, and variance $\sigma_k^2$ according to

$$\bar{\mu}_k^{(l+1)} = \frac{\sum_{j=1}^{N} y_{jk}^{(l)} f(t_j)}{\sum_{j=1}^{N} y_{jk}^{(l)}},$$

$$\sigma_k^2^{(l+1)} = \frac{\sum_{j=1}^{N} (y_{jk}^{(l)} - \bar{\mu}_k^{(l+1)})^2}{\sum_{k=1}^{M} y_{jk}^{(l)}},$$

$$\omega_k^{(l+1)} = \frac{\sum_{j=1}^{N} y_{jk}^{(l)}}{N}, \hspace{1cm} j = 1, 2, \ldots, N; \quad k = 1, 2, \ldots, M. \hspace{1cm} (10)$$

**Step 4.** Repeat Steps 2 and 3 until the convergence of maximum likelihood function value is realized.

In Step 3, $\bar{\mu}_k, \sigma_k^2, \bar{\sigma}_k, \eta_k$ meet (11) in EM algorithm. However the following is a transcendental equation which is hard to calculate analytical solutions:

$$\mu = \gamma + \eta \Gamma \left( 1 + \frac{1}{\beta} \right),$$

$$\sigma^2 = \eta^2 \left[ \Gamma \left( 1 + \frac{2}{\beta} \right) - \Gamma^2 \left( 1 + \frac{1}{\beta} \right) \right].$$ \hspace{1cm} (11)

In each iteration of EM algorithm, let $\lambda = \mu/\sigma$. When $\gamma = 0$, then the following exists:

$$\lambda = \frac{\Gamma \left( 1 + 1/\beta \right)}{\left[ \Gamma \left( 1 + 2/\beta \right) - \Gamma^2 \left( 1 + 1/\beta \right) \right]^{\frac{1}{2}}}.$$

According to (12), there is a monotonic relationship between $\lambda$ and $\beta$. Therefore, RBF interpolation can be used to establish the mapping relationship between $\lambda$ and $\beta$. Then $\beta$ can be calculated based on $\lambda$. Then, according to the following, $\eta$ can be easily gotten:

$$\eta = \frac{\mu}{\Gamma \left( 1 + 1/\beta \right)}.$$

$$\eta = \frac{\mu}{\Gamma \left( 1 + 1/\beta \right)}.$$
\hspace{1cm} (13)
3.2. Selection of HLNF. Selection of HLNF has a great impact on the generalization performance of neutral network and the efficiency of learning process. Single Weibull distribution as neuron function may lead to underfitting problem. What is more, multifolds Weibull mixture distribution as neuron function may lead to overfitting phenomenon. It is necessary to design an algorithm to select the most appropriate HLNF. Based on the idea of random sampling, this research chooses the value of maximum likelihood function as the index to select the most appropriate finite Weibull mixture distribution as hidden layer nodes function (HLNF) by multiple sampling. This selection algorithm is as follows.

**Step 1.** Based on the original sample data \( S \), use the bootstrap methodology [26] to generate \( N \) groups of training samples \( X = [X_1, X_2, \ldots, X_N] \) and \( N \) groups of testing samples \( X' = [X'_1, X'_2, \ldots, X'_N] \).

**Step 2.** For the \( p \) groups of training samples \( X \), use the EM algorithm to estimate parameters of finite Weibull mixture distributions which are from 1 to \( M \) folds, separately getting mixture weight \( \omega_{k,o} \), shape parameter \( \beta_{k,o} \), and scale parameter \( \eta_{k,o} \), \( j = 1, 2, \ldots, N \), \( k = 1, 2, \ldots, M \), \( o = 1, 2, \ldots, k \).

**Step 3.** According to the parameters of training samples in Step 2, maximum likelihood function values \( I_{k,o} \) of \( N \) groups testing samples can be calculated by (14), where \( Y_i \) is obtained by formulas (6) (7), and (8). Maximum likelihood function values form an evaluation matrix \( L \) whose size is \( N \times N \times M \):

\[
L = \ln \left( \prod_{i=1}^{N} Y_i \right).
\]

**Step 4.** According to (15), mean evaluation value \( P_o \) of \( N \) groups testing samples corresponding to \( o \)-folds Weibull mixture distribution can be gotten, and select the maximum evaluation value corresponding to \( o \)-folds Weibull mixture distribution. Then \( o \)-folds Weibull mixture distribution would be the most appropriate finite Weibull mixture distribution which is the hidden layer nodes function (HLNF) for the original sample data \( S \):

\[
P_o = \sum_{o=1}^{N} \sum_{k=1}^{M} I_{k,o}.
\]

3.3. Determining the Number of Hidden Layer Nodes. The number of nodes in the hidden layer is associated with not only the function between input and output, but also sample size, random noise, and so forth. Generally small amount of nodes causes poor recognition performance and fitting performance. However large numbers of nodes easily result in random noise and poor recognition performance. Therefore, choosing an appropriate nodes in hidden layer is critical for improving generalization performance of network.

Sampling study is helpful to weaken the influence of random noise. Therefore, as the number of nodes in hidden layer dynamically increases, the shape of density function of sample data tends to be stable in the learning process of WNN. For the learning process, similarity coefficient is defined to determine the stopping condition.

When the number of nodes in hidden layer is \( k \) and \( k + 1 \), define the density function as \( f_k(x) \) and \( f_{k+1}(x) \), respectively. Then the similarity coefficient (SC) between \( f_k(x) \) and \( f_{k+1}(x) \) is defined as

\[
SC = \int_{0}^{+\infty} \min \{ f_k(x), f_{k+1}(x) \} \, dx - \int_{0}^{+\infty} \max \{ f_k(x), f_{k+1}(x) \} \, dx.
\]

As is shown in Figure 4, the ratio between intersection and union of coverage area by \( f_k(x) \) and \( f_{k+1}(x) \) is the similarity coefficient (SC). The interval of theoretical value of similarity coefficient is \([0, 1]\). The larger SC means the higher similarity between \( f_k(x) \) and \( f_{k+1}(x) \). When SC = 1, it means that two density functions are identical.

By combining HLNF, bootstrap algorithm, and EM algorithm, the specific algorithm to determine the hidden layer nodes is given as follows.

**Step 1.** Initialize SC\(_c\) which is the stopping condition of the algorithm.

**Step 2.** Using Bootstrap algorithm to generate a group of initial training sample data \( X_1 \), estimate parameters \( \theta_1 \) of HLNF corresponding to sample data \( X_1 \) and \( \theta_1 = (\omega_1, \eta_1, \beta_1) \); set \( k = 1 \).

**Step 3.** Using Bootstrap algorithm to generate \( k + 1 \)th group of training sample data \( X_{k+1} \), estimate parameters \( \theta_{k+1} \) of HLNF corresponding to sample data \( X_{k+1} \), and \( \theta_{k+1} = (\omega_{k+1}, \eta_{k+1}, \beta_{k+1}) \). Then the WNN model can be gotten:

\[
f_{k+1}(x) = \frac{1}{k+1} \sum_{q=1}^{k+1} p \left( x; \theta_q \right),
\]

where \( p(x; \theta_q) \) is the probability density function of \( q \)-th hidden layer node.

**Step 4.** According to (16), similarity coefficient \( SC_k \) between \( k \) and \( k + 1 \) layers WNN can be gotten to judge whether it meets the stopping condition: \( SC_k > SC_c \). If \( SC_k \) does not satisfy the condition, go to Step 3; if \( SC_k \) satisfies the condition, end the training.

![Figure 4: Schematic diagram of similarity coefficient.](image-url)
Finally the WNN is defined. And the reliability assessment process is shown in Figure 5. In the process, at the 3rd step, when parameters of WNN are gotten, the probability density on WNN is also gotten.

4. Data Collection and Analysis

In order to validate the WNN model, we collected 23 CNC machining centers’ field test data, almost more than 3000 h running time for each. After data preprocessing, the time between failures obtained within the operation time is listed in Table 1.

For the failure data in Table 1, according to Section 3.2, twofold Weibull mixture distribution is selected as neuron function. Similarity coefficient $SC_\gamma = 0.99$ is set to be the stopping condition in learning process of Weibull neural network. According to $SC_k > SC_\gamma$, the number of hidden layer nodes is calculated as 51. Follow the steps in Section 3.1, estimate the parameters for hidden layer nodes function (HLNF) $\phi(t)$, and get mixture weights $\alpha$, shape parameter $\beta$, and scale parameter $\eta$ of 51 nodes that are shown in Table 2. Substitute those three parameters into (6), (7), and (8) to calculate probability density $Y$ which is the blue line in Figure 6.

Distribution law of time between failures of 23 CNC machining centers is modeled by WNN. For comparison, general two-parameter Weibull distribution (TPWD) is used to analysis the same data. And the probability density curves on two methods are shown in Figure 6. The blue curve is the probability density on WNN and the red one is on TPWD. The probability density function curves of the data are continuous and derivable in this case, presenting good generalization performance which inherits Weibull method.

Different from the only one peak of the probability density curve on TPWD, the blue one has another peak around 1500 h which means the CNC machining center has more probability to be a failure after running about 2500 h. The probability density curve on WNN reveals more accurate information of the distribution law of time between failures than that on TPWD. Two-peak curve by WNN approximates actual condition. In other words, WNN has better approximation ability than TPWD in distribution modeling of life data.

Mean time between failures (MTBF) describes the expected time between two failures for a repairable system [27]. MTBF is the major index of the reliability of CNC machining centers; MTBF based on point estimation and MTBF based on interval estimation are the most famous
Weibull neural network (WNN) has better approximation on interval estimation. The result of the case indicates that WNN is wider than the result of TPBD by over 23.5 h. And MTBF based on interval estimation, under the same confidence level, are much different: the lower limit of WNN is lower than that of TPBD, and the upper limit of WNN is higher than that of TPBD by 16.192 h. What is more, the confidence interval of MTBF based on interval estimation of WNN is wider than the result of TPBD by over 23.5 h equivalent to 10.58% of TPBD. The big differences of MTBF based on interval estimation and point estimation further show that WNN and TPBD reveal the different distribution laws of time between failures of CNC machining centers. And there is a big error for TPBD in reliability assessment of CNC machining centers. Because WNN has better approximation ability than TPWD in distribution modeling of life data of CNC machining centers, combined with the above comparison, WNN is more suitable to estimate the time between failures.

### Table 2: Parameters estimation results of WNN.

<table>
<thead>
<tr>
<th>Nodes</th>
<th>Parameters</th>
<th>$\alpha$</th>
<th>$\beta$</th>
<th>$\eta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>0.044068</td>
<td>0.955932</td>
<td>9.033967</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>0.959583</td>
<td>0.040417</td>
<td>1.455573</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>0.082716</td>
<td>0.917284</td>
<td>10.44422</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>0.068718</td>
<td>0.931282</td>
<td>14.46845</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>0.94974</td>
<td>0.05026</td>
<td>1.306634</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>0.056699</td>
<td>0.943301</td>
<td>10.93411</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>0.025545</td>
<td>0.974455</td>
<td>13.8031</td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>0.085892</td>
<td>0.943088</td>
<td>13.55691</td>
</tr>
<tr>
<td>9</td>
<td></td>
<td>0.107466</td>
<td>0.892934</td>
<td>19.14584</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>0.068432</td>
<td>0.931568</td>
<td>13.75564</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>0.938625</td>
<td>0.063175</td>
<td>1.331925</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>0.055034</td>
<td>0.949466</td>
<td>13.1061</td>
</tr>
<tr>
<td>13</td>
<td></td>
<td>0.896614</td>
<td>0.103386</td>
<td>1.347853</td>
</tr>
<tr>
<td>14</td>
<td></td>
<td>0.041398</td>
<td>0.958602</td>
<td>16.30011</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>0.086608</td>
<td>0.913932</td>
<td>11.68258</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td>0.053959</td>
<td>0.946041</td>
<td>12.07066</td>
</tr>
<tr>
<td>17</td>
<td></td>
<td>0.066304</td>
<td>0.933696</td>
<td>1.77975</td>
</tr>
<tr>
<td>18</td>
<td></td>
<td>0.932007</td>
<td>0.067993</td>
<td>1.664349</td>
</tr>
<tr>
<td>19</td>
<td></td>
<td>0.119909</td>
<td>0.880091</td>
<td>16.52835</td>
</tr>
<tr>
<td>20</td>
<td></td>
<td>0.08975</td>
<td>0.91025</td>
<td>16.62012</td>
</tr>
<tr>
<td>21</td>
<td></td>
<td>0.941526</td>
<td>0.058474</td>
<td>1.56535</td>
</tr>
<tr>
<td>22</td>
<td></td>
<td>0.960506</td>
<td>0.039449</td>
<td>1.355989</td>
</tr>
<tr>
<td>23</td>
<td></td>
<td>0.106106</td>
<td>0.893894</td>
<td>1.350209</td>
</tr>
<tr>
<td>24</td>
<td></td>
<td>0.119571</td>
<td>0.880429</td>
<td>13.3846</td>
</tr>
<tr>
<td>25</td>
<td></td>
<td>0.922893</td>
<td>0.077107</td>
<td>1.451953</td>
</tr>
<tr>
<td>26</td>
<td></td>
<td>0.96321</td>
<td>0.03679</td>
<td>1.448205</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td>0.947078</td>
<td>0.052922</td>
<td>1.34913</td>
</tr>
<tr>
<td>28</td>
<td></td>
<td>0.945748</td>
<td>0.054252</td>
<td>1.35939</td>
</tr>
<tr>
<td>29</td>
<td></td>
<td>0.079954</td>
<td>0.920046</td>
<td>1.66881</td>
</tr>
<tr>
<td>30</td>
<td></td>
<td>0.960552</td>
<td>0.039448</td>
<td>1.693323</td>
</tr>
<tr>
<td>31</td>
<td></td>
<td>0.627252</td>
<td>0.372748</td>
<td>1.576105</td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>0.053532</td>
<td>0.946648</td>
<td>17.31282</td>
</tr>
<tr>
<td>33</td>
<td></td>
<td>0.963131</td>
<td>0.038689</td>
<td>1.683207</td>
</tr>
<tr>
<td>34</td>
<td></td>
<td>0.881935</td>
<td>0.118065</td>
<td>1.508832</td>
</tr>
<tr>
<td>35</td>
<td></td>
<td>0.39565</td>
<td>0.960435</td>
<td>18.62358</td>
</tr>
<tr>
<td>36</td>
<td></td>
<td>0.912109</td>
<td>0.078691</td>
<td>1.53088</td>
</tr>
<tr>
<td>37</td>
<td></td>
<td>0.958295</td>
<td>0.041705</td>
<td>1.355888</td>
</tr>
<tr>
<td>38</td>
<td></td>
<td>0.93281</td>
<td>0.068719</td>
<td>1.63859</td>
</tr>
<tr>
<td>39</td>
<td></td>
<td>0.912698</td>
<td>0.087302</td>
<td>1.593501</td>
</tr>
<tr>
<td>40</td>
<td></td>
<td>0.350292</td>
<td>0.649708</td>
<td>2.350637</td>
</tr>
<tr>
<td>41</td>
<td></td>
<td>0.787888</td>
<td>0.212122</td>
<td>1.510548</td>
</tr>
<tr>
<td>42</td>
<td></td>
<td>0.980145</td>
<td>0.019853</td>
<td>1.586655</td>
</tr>
<tr>
<td>43</td>
<td></td>
<td>0.603718</td>
<td>0.936822</td>
<td>13.20558</td>
</tr>
<tr>
<td>44</td>
<td></td>
<td>0.028453</td>
<td>0.971547</td>
<td>11.44262</td>
</tr>
<tr>
<td>45</td>
<td></td>
<td>0.87827</td>
<td>0.12173</td>
<td>1.284036</td>
</tr>
<tr>
<td>46</td>
<td></td>
<td>0.058714</td>
<td>0.941286</td>
<td>10.43322</td>
</tr>
<tr>
<td>47</td>
<td></td>
<td>0.922711</td>
<td>0.077289</td>
<td>1.62935</td>
</tr>
<tr>
<td>48</td>
<td></td>
<td>0.95051</td>
<td>0.04949</td>
<td>1.543645</td>
</tr>
<tr>
<td>49</td>
<td></td>
<td>0.96071</td>
<td>0.03929</td>
<td>1.499167</td>
</tr>
<tr>
<td>50</td>
<td></td>
<td>0.922835</td>
<td>0.077165</td>
<td>1.420721</td>
</tr>
</tbody>
</table>

### Table 3: MTBF of CNC machining centers.

<table>
<thead>
<tr>
<th>Model</th>
<th>MTBF based on point estimation (h)</th>
<th>MTBF based on interval estimation (h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>WNN</td>
<td>751.42</td>
<td>628.4599</td>
</tr>
<tr>
<td>TPWD</td>
<td>737.1336</td>
<td>635.7854</td>
</tr>
</tbody>
</table>

5. Conclusions

The basic idea discussed in this paper is the study on applications of Weibull network for complex system reliability assessment. General reliability model easily results in overfitting or underfitting in reliability modeling process. The poor generalization performance of general reliability model could not reflect the actual life distribution law of reliability data. For the above problem, through analyzing the characteristics of artificial neural networks and Weibull mixture distribution, the authors propose Weibull neural network (WNN) for system reliability modeling. A common structure of neural network, hierarchical structure, is adopted in Weibull neural network. And three-step learning process of Weibull network is proposed in this paper. In the learning process, using interpolation method to solve transcendental equation significantly improves the computational efficiency of EM algorithm. Finally, a practical application case is presented. WNN is used for analyzing distribution law of time between failures of certain type of CNC machining centers. The probability density function curve of the data is continuous and derivable in the case, presenting good generalization performance. For further comparison, authors introduce two-parameter Weibull distribution (TPWD) to calculate MTBF based on point estimation and MTBF based on interval estimation. The result of the case indicates that Weibull neural network (WNN) has actual distribution approximation methods. The two MTBF of WNN are calculated based on the above two.

methods for MTBF. To deeply estimate WNN and TPWD, these two MTBF are calculated based on the above two.
ability than TPWD in distribution modeling of life data. And Weibull neural network (WNN) could be popularized and applied on reliability assessment for complex systems.

### Conflict of Interests

The authors declare that there is no conflict of interests regarding the publishing of this paper.

### Acknowledgments

Research in this paper was supported by National Science and Technology Major Project of China: reliability promotion of thousands Chinese CNC machining centers (Grant no. 2013ZX04011-012); and Jilin province science and technology development plan: reliability system design for key functional components of CNC machine tools (Grant no. 20130302009GX).

### References


