Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 292197, 8 pages
http://dx.doi.org/10.1155/2015/292197

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

Academic Editor: Marco Mussetta

CNC machining centers, as the key device in modern manufacturing industry, are complicated electrohydraulic products. The reliability is the most important index of CNC machining centers. However, simple life distributions hardly reflect the true law of complex system reliability with many kinds of failure mechanisms. Due to Weibull model's versatility and relative simplicity and artificial neural networks' (ANNs) high capability of approximating, they are widely used in reliability engineering and elsewhere. Considering the advantages of these two models, this paper defined a novel model: Weibull neural network (WNN). WNN inherits the hierarchical structure from ANNs which include three layers, namely, input layer, hidden layer, and output layer. Based on more than 3000 h field test data of CNC machining centers, WNN has been successfully applied in comprehensive operation data analysis. The results show that WNN has good approximation ability and generalization performance in reliability assessment of CNC machining centers.

1. Introduction

Common life distributions, like normal distribution, lognormal 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 neutron 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.

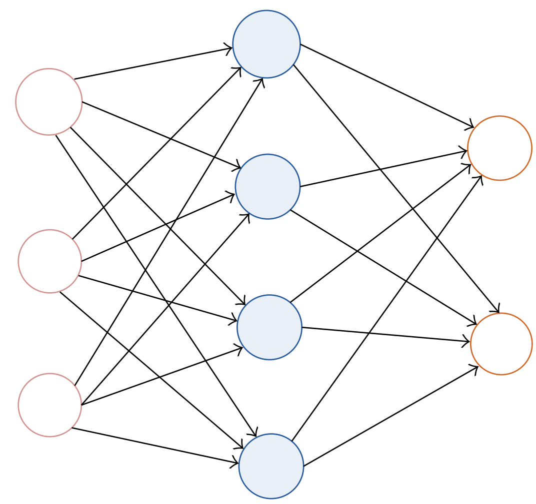

Figure 1: Hierarchical structure of artificial neural networks (ANNs).

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:

$$F(t) = \alpha_1 F_1(t) + \alpha_2 F_2(t) + \cdots + \alpha_n F_n(t). \quad (1)$$

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

$$f(t) = \alpha_1 f_1(t) + \alpha_2 f_2(t) + \cdots + \alpha_n f_n(t). \quad (2)$$

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 β_i and scale parameter η_i or three-parameter Weibull cumulative distribution with shape parameter β_i, scale parameter η_i, and positional parameter ζ_i, two-parameter Weibull cumulative distribution is shown below:

$$F_i(t) = 1 - \exp\left[-\left(\frac{t}{\eta_i}\right)^{\beta_i}\right]. \quad (3)$$

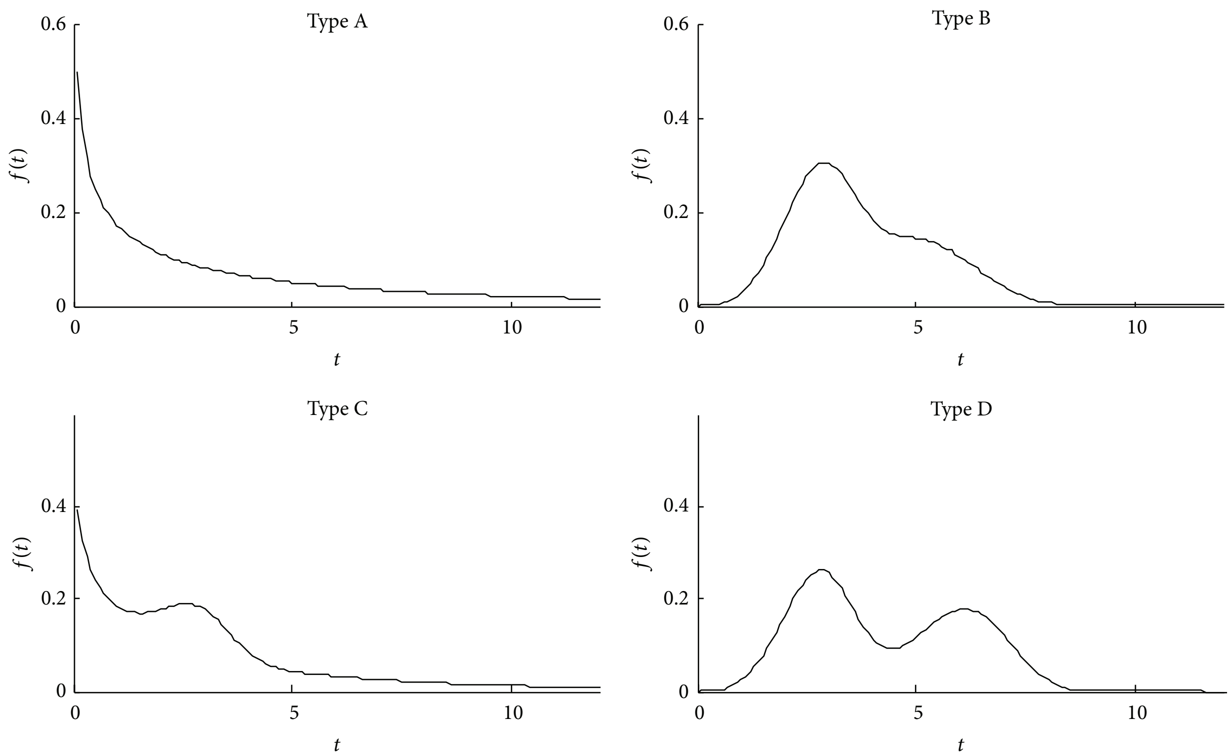

FIGURE 2: Basic types of density functions of twofold mixture Weibull distribution with two parameters.

Three-parameter Weibull cumulative distribution is shown below:

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

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.

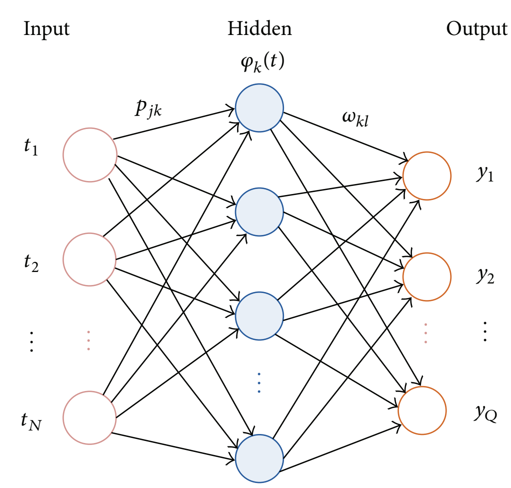

FIGURE 3: Hierarchical structure of Weibull neural network (WNN).

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. In 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 jth data sampled by the kth 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 ω_{kl} which means the weight of the kth node to the lth output y. According to the characteristic of mixture distribution, ω_{kl} needs to satisfy

$$\sum_{k=1}^{M} \omega_{kl} = 1, \tag{5}$$

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

As shown in Figure 3, the input layer achieves nonlinear mapping from input data T 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:

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

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

$$f_i(t) = \left(\frac{\beta_i}{\eta_i}\right)\left(\frac{t}{\eta_i}\right)^{\beta_i - 1} \exp\left[-\left(\frac{t}{\eta_i}\right)^{\beta_i}\right], \tag{7}$$

$$\varphi_k(t) = \alpha_1 f_1(t) + \cdots + \alpha_n f_n(t), \tag{8}$$

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 $\hat{\mu}_1^{(0)}, \hat{\mu}_2^{(0)}, \ldots, \hat{\mu}_n^{(0)}$, variances $\hat{\sigma}_1^{2(0)}, \hat{\sigma}_2^{2(0)}, \ldots, \hat{\sigma}_n^{2(0)}$, and mixture weights $\hat{\omega}_1^{(0)}, \hat{\omega}_2^{(0)}, \ldots, \hat{\omega}_n^{(0)}$.

Step 2. E-step: calculate responsivity γ_j according to

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

Step 3. M-step: calculate weight ω_k, mean $\hat{\mu}_k$, and variance $\hat{\sigma}_k^2$ according to

$$\begin{aligned} \hat{\mu}_k^{(l+1)} &= \frac{\sum_{j=1}^{N} \gamma_{jk}^{(l)} t_j}{\sum_{j=1}^{N} \gamma_{jk}^{(l)}}, \\ \hat{\sigma}_k^{2(l+1)} &= \frac{\sum_{j=1}^{N} \gamma_{jk}^{(l)} (t_j - \hat{\mu}_k)^2}{\sum_{k=1}^{M} \gamma_{jk}^{(k)}}, \\ \omega_k^{(l+1)} &= \frac{\sum_{j=1}^{N} \gamma_{jk}^{(l)}}{N}, \\ &\qquad j = 1, 2, \ldots, N; \; k = 1, 2, \ldots, M. \end{aligned} \tag{10}$$

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

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

$$\begin{aligned} \mu &= \gamma + \eta\Gamma\left(1 + \frac{1}{\beta}\right), \\ \sigma^2 &= \eta^2\left[\Gamma\left(1 + \frac{2}{\beta}\right) - \Gamma^2\left(1 + \frac{1}{\beta}\right)\right]. \end{aligned} \tag{11}$$

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

$$\lambda = \frac{\Gamma(1 + 1/\beta)}{\left[\Gamma(1 + 2/\beta) - \Gamma^2(1 + 1/\beta)\right]^{0.5}}. \tag{12}$$

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

$$\eta = \frac{\mu}{\Gamma(1 + 1/\beta)}. \tag{13}$$

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 ω_{jko}, shape parameter β_{jko}, and scale parameter η_{jko}, $j = 1, 2, \ldots, N$, $k = 1, 2, \ldots, M$, $o = 1, 2, \ldots, k$.

Step 3. According to the parameters of training samples in Step 2, maximum likelihood function values l_{jko} 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 $N \times N \times M$:

$$l = \ln\left(\prod_{i=1}^{N} Y_i\right). \quad (14)$$

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:

$$P_o = \sum_{j=1}^{N} \sum_{k=1}^{M} l_{jko}. \quad (15)$$

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.

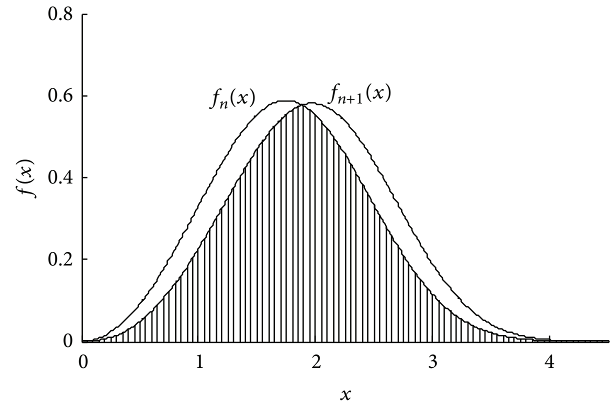

Figure 4: Schematic diagram of similarity coefficient.

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

$$\text{SC} = \frac{\int_0^{+\infty} \min\{f_k(x), f_{k+1}(x)\}\,dx}{\int_0^{+\infty} \max\{f_k(x), f_{k+1}(x)\}\,dx}. \quad (16)$$

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 θ_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 θ_{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:

$$f_{k+1}(x) = \frac{1}{k+1} \sum_{q=1}^{k+1} p(x; \theta_q), \quad (17)$$

where $p(x;\theta_q)$ is the probability density function of qth 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: $\text{SC}_k > \text{SC}_c$. If SC_k does not satisfy the condition, go to Step 3; if SC_k satisfies the condition, end the training.

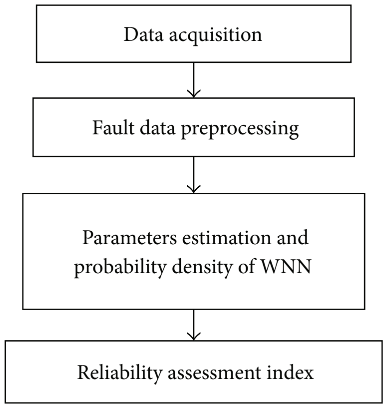

Figure 5: Flow chart of reliability assessment based on WNN.

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 $SC_c = 0.99$ is set to be the stopping condition in learning process of Weibull neural network. According to $SC_k > SC_c$, 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) $\varphi(t)$, and get mixture weights α, shape parameter β, and scale parameter η 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.

Table 1: Time between failures of 23 CNC machining centers.

No.	Time between failures							
1	1008	251.5	369.3	402.5	119.3	402.5	193.3	
2	637.3	723.2	185.8	274	183.4	246.7	334.3	
3	613.5	178.8	826.3	125.9	391.9	1088.2		
4	1044							
5	916	383.2	724.7	208.7	509.8			
6	1039.5	162.5	426.5	598.3	150	807		
7	720.5	154	112.7	573.8	841.3	693.8		
8	1283.5	446.5	512.8					
9	721.5	1232.7						
10	662.2	373.5	516.5	115.7	462.8	135.5	475.8	316.5
11	938.2	171.7	343.5	884.2	208.8	209.5	149	175.2
12	1238.5							
13	2243	132.7	694.7					
14	1537.2							
15	912	1707						
16	1359.8	303.7	582.2					
17	998.7	448.7	765.2					
18	2954.2							
19	1597.3	356.8	1249.7					
20	2377	983.3						
21	2725	612						
22	2523							
23	2784							

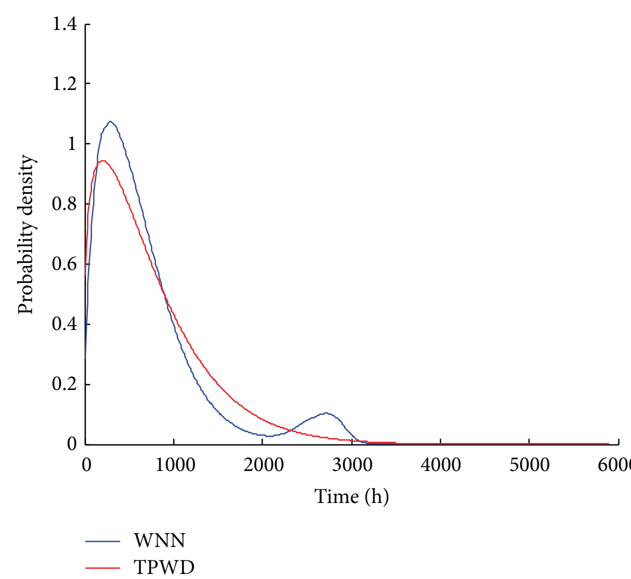

Figure 6: The probability density curves on WNN and TPWD.

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

Table 2: Parameters estimation results of WNN.

Nodes	Parameters α		β		η	
1	0.044068	0.955932	9.033967	1.551175	2637.021	693.5101
2	0.959583	0.040417	1.455573	24.94987	667.7464	2845.028
3	0.082716	0.917284	10.44422	1.39166	2747.245	653.5588
4	0.068718	0.931282	14.46845	1.367156	2747.849	537.2695
5	0.94974	0.05026	1.306634	24.94987	723.8985	2466.681
6	0.056699	0.943301	10.93411	1.494548	2760.401	663.9032
7	0.025545	0.974455	13.8031	1.403081	2557.15	624.5496
8	0.085692	0.914308	13.55691	1.369662	2819.825	579.1172
9	0.107066	0.892934	19.14584	1.43281	2736.786	654.4722
10	0.068432	0.931568	13.75564	1.550829	2724.443	634.1239
11	0.936825	0.063175	1.331295	13.47746	604.2012	2860.957
12	0.055034	0.944966	13.10611	1.355398	2797.873	606.5653
13	0.896614	0.103386	1.347853	16.53756	671.6885	2674.685
14	0.041398	0.958602	16.30011	1.426415	2814.403	684.1883
15	0.086068	0.913932	11.68258	1.546744	2662.409	713.4144
16	0.053959	0.946041	12.07066	1.454367	2669.093	679.6944
17	0.066304	0.933696	17.7975	1.531203	2817.255	722.0759
18	0.932007	0.067993	1.664349	15.259	694.9488	2750.57
19	0.119909	0.880091	16.52835	1.397195	2816.441	706.3912
20	0.08975	0.91025	16.62012	1.656576	2582.263	597.3034
21	0.941526	0.058474	1.561535	24.89791	716.9525	2864.898
22	0.960506	0.039494	1.735449	24.94987	674.279	2433.25
23	0.106106	0.893894	13.30659	1.350209	2830.795	590.5806
24	0.119571	0.880429	13.3864	1.459669	2582.189	577.6149
25	0.922893	0.077107	1.451593	13.29719	620.3416	2750.541
26	0.96321	0.03679	1.448205	12.71502	649.7289	2833.593
27	0.947078	0.052922	1.34193	24.94987	698.0294	2464.589
28	0.945748	0.054252	1.35939	13.40081	617.7688	2778.687
29	0.079954	0.920046	11.68818	1.854535	2766.994	667.8148
30	0.960552	0.039448	1.693323	17.84738	672.5292	2629.48
31	0.627252	0.372748	1.576105	2.480788	923.7622	283.3751
32	0.053352	0.946648	17.31282	1.544846	2786.179	719.6491
33	0.961311	0.038689	1.683207	12.15247	686.9369	2510.049
34	0.881935	0.118065	1.508832	12.78487	651.8541	2706.579
35	0.039565	0.960435	18.62358	1.588752	2760.565	645.8836
36	0.921309	0.078691	1.503808	22.29027	678.9718	2808.833
37	0.958295	0.041705	1.535188	16.46958	743.197	2781.517
38	0.931281	0.068719	1.631859	13.85231	652.3391	2772.066
39	0.912698	0.087302	1.593501	14.62023	696.2038	2602.976
40	0.350292	0.649708	2.550637	1.42347	284.9288	958.2812
41	0.787888	0.212112	1.515084	1.854923	554.2234	1567.009
42	0.980145	0.019855	1.586655	24.94987	678.2233	2932.822
43	0.063178	0.936822	13.20558	1.394374	2851.574	645.8096
44	0.028453	0.971547	11.44262	1.525982	2573.366	627.3912
45	0.87827	0.12173	1.284063	5.906063	753.368	177.2281
46	0.058714	0.941286	10.43322	1.472725	2659.111	612.2541
47	0.922711	0.077289	1.62935	15.13635	690.0196	2786.844
48	0.95051	0.04949	1.543645	24.94987	650.0357	2715.602
49	0.96071	0.03929	1.499167	22.50363	630.3924	2831.961
50	0.922835	0.077165	1.420721	11.26682	618.792	2585.095
51	0.072285	0.927715	13.29679	1.309585	2727.193	631.4372

methods for MTBF. To deeply estimate WNN and TPWD, these two MTBF are calculated based on the above two methods. The two MTBF of WNN are calculated by the method in [28–30]. The comparison results are shown in Table 3.

Table 3: MTBF of CNC machining centers.

Model	MTBF based on point estimation (h)	MTBF based on interval estimation (h) Lower limit	Upper limit
WNN	751.42	628.4559	874.3859
TPWD	737.1336	635.7854	858.1939

The results show that MTBF based on point estimation under WNN is larger than that under TPWD by 14.2864 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 close to the actual distribution law of time between failures.

5. Conclusions

The basic idea discussed in this paper is the study on applications of Weibull neural 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 neural 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 better approximation 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

[1] M. Rausand and A. Hoyland, *System Reliability Theory: Models, Statistical Methods, and Applications*, John Wiley & Sons, 2003.

[2] F. Downton, "Bivariate exponential distributions in reliability theory," *Journal of the Royal Statistical Society Series B: Methodological*, vol. 32, pp. 408–417, 1970.

[3] J. H. K. Kao, "A graphical estimation of mixed Weibull parameters in life-testing of Electron tubes," *Technometrics*, vol. 1, no. 4, pp. 389–407, 1959.

[4] D. M. Titterington, A. F. M. Smith, and U. E. Makov, *Statistical Analysis of Finite Mixture Distributions*, John Wiley & Sons, New York, NY, USA, 1985.

[5] W. Mendenhall and R. J. Hader, "Estimation of parameters of mixed exponentially distributed failure time distributions from censored life test data," *Biometrika*, vol. 45, pp. 504–520, 1958.

[6] E. E. Elmahdy and A. W. Aboutahoun, "A new approach for parameter estimation of finite Weibull mixture distributions for reliability modeling," *Applied Mathematical Modelling*, vol. 37, no. 4, pp. 1800–1810, 2013.

[7] T. Bučar, M. Nagode, and M. Fajdiga, "Reliability approximation using finite Weibull mixture distributions," *Reliability Engineering and System Safety*, vol. 84, no. 3, pp. 241–251, 2004.

[8] C.-K. Goh, E.-J. Teoh, and K. C. Tan, "Hybrid multiobjective evolutionary design for artificial neural networks," *IEEE Transactions on Neural Networks*, vol. 19, no. 9, pp. 1531–1548, 2008.

[9] G. Z. Li, T. Y. Liu, and G. F. Wu, "Improving generalization ability of neural networks ensemble with multi-task learning," *Journal of Computational Information Systems*, vol. 2, no. 4, pp. 1235–1240, 2006.

[10] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: the RPROP algorithm," in *Proceedings of the IEEE International Conference on Neural Networks*, H. Ruspini, Ed., pp. 586–591, IEEE, San Francisco, Calif, USA, 1993.

[11] S. Haykin, *Neural Networks and Learning Machines*, Prentice Hall, 2008.

[12] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[13] P. M. Bentler and D. G. Bonett, "Significance tests and goodness of fit in the analysis of covariance structures," *Psychological Bulletin*, vol. 88, no. 3, pp. 588–606, 1980.

[14] N. Mai-Duy and R. I. Tanner, "A collocation method based on one-dimensional RBF interpolation scheme for solving PDEs," *International Journal of Numerical Methods for Heat and Fluid Flow*, vol. 17, no. 2, pp. 165–186, 2007.

[15] S. Z. Li, X. Y. Li, and T. M. Jiang, "A prediction method of life and reliability for CSALT using Grey RBF neural networks," in *Proceedings of the IEEE 16th International Conference on Industrial Engineering & Engineering Management*, pp. 699–703, IEEE, Beijing, China, October 2009.

[16] J. Moody, "Fast learning in multi-resolution hierarchives," in *Advances in Neural Information Processing Systems I*, vol. 1, pp. 29–39, Morgan Kaufmann Publishers, 1989.

[17] J. Moody and C. J. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Computation*, vol. 1, no. 2, pp. 281–294, 1989.

[18] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice-Hall, 1999.

[19] C. Jutten and J. Herault, "Blind separation of sources, part I: an adaptive algorithm based on neuromimetic architecture," *Signal Processing*, vol. 24, no. 1, pp. 1–10, 1991.

[20] Z. C. Xue and H. J. Wang, "The application of artificial BP neural networks and Monte-Carlo method for the reliability analysis on frame structure," *Applied Mechanics & Materials*, vol. 204–208, pp. 3256–3259, 2012.

[21] W. Weibull, "A statistical distribution function of wide applicability," *Journal of Applied Mechanics*, vol. 18, pp. 293–297, 1951.

[22] J. I. McCool, "Competing risk and multiple comparison analysis for bearing fatigue tests," *Tribology Transactions*, vol. 21, no. 4, pp. 271–284, 1978.

[23] R. C. Elandt-Johnson and N. L. Johnson, *Survival Models and Data Analysis*, John Wiley & Sons, New York, NY, USA, 1980.

[24] T. Denœux, "Maximum likelihood estimation from fuzzy data using the EM algorithm," *Fuzzy Sets and Systems*, vol. 183, pp. 72–91, 2011.

[25] S. Nandi and I. Dewan, "An EM algorithm for estimating the parameters of bivariate Weibull distribution under random censoring," *Computational Statistics and Data Analysis*, vol. 54, no. 6, pp. 1559–1569, 2010.

[26] V. A. Jochen and J. P. Spivey, "Probabilistic reserves estimation using decline curve analysis with the bootstrap method," in *Proceedings of the SPE Annual Technical Conference and Exhibition*, pp. 589–598, Society of Petroleum Engineers, Denver, Colo, USA, October 1996.

[27] M. J. Mondro, "Approximation of mean time between failure when a system has periodic maintenance," *IEEE Transactions on Reliability*, vol. 51, no. 2, pp. 166–167, 2002.

[28] R. Billinton and P. Wang, "Teaching distribution system reliability evaluation using Monte Carlo simulation," *IEEE Transactions on Power Systems*, vol. 14, no. 2, pp. 397–403, 1999.

[29] J. G. Yang, Z. M. Wang, G. Wang, and G. Zhang, "Likelihood ratio test interval estimation of reliability indices for numerical control machine tools," *Journal of Mechanical Engineering*, vol. 48, no. 2, pp. 9–15, 2012.

[30] A. Mustafa, "Reliability equivalence factors for some systems with mixture weibull failure rates," *African Journal of Mathematics and Computer Science Research*, vol. 2, no. 1, pp. 6–13, 2009.