Fault Diagnosis of Gas Turbine Based on Improved BP Neural Network with the Combination of N-W and L-M Algorithm

Zhang Yun¹, Qian Yu-liang¹, Qiu Zheng¹ and Zhang Xiao¹

¹ Department of automation, Shanghai University of Electric Power, Shanghai, China.
E-mail:ZhangYun0911@126.com

Abstract. In order to solve the problem that the training speed of traditional BP neural network is slow in the process of gas turbine fault diagnosis, a new fault diagnosis method based on a combination of Nguyen-Widrow method and L-M optimized BP algorithm was proposed. The Nguyen-Widrow method is used to initialize the weights and thresholds of neurons in the BP neural network, and the L-M algorithm is used to improve the search space of the BP neural network, which reduces the times of network training and accelerates the learning speed of the network. The gradient descent method, the conjugate gradient method and the N-W and L-M combination optimization methods are respectively applied to the fault diagnosis of gas turbine. The results show that the BP neural network model optimized by combining N-W and L-M has faster learning speed and higher diagnostic efficiency for gas turbine fault diagnosis.

1. Introduction
In recent years, under the guidance of the national energy conservation and emission reduction policies, gas-fired power generation technologies with green pollution-free, small size, rapid start-stop, and other outstanding advantages have been vigorously promoted. The installed capacity of domestic gas turbine unit has been continuously improved. As the core component of the gas turbine unit, the gas turbine is operated under high temperature and high pressure conditions for a long time. It is prone to failure, resulting in production accidents. Therefore, the condition monitoring and fault diagnosis of the gas turbine is of great significance for ensuring safe and reliable operation of the gas turbine unit [1]-[3].

At present, various intelligent algorithms such as neural networks, expert systems, and knowledge systems are widely used for gas turbine fault diagnosis. The neural network is widely used in fault diagnosis due to its simple structure, strong self-learning ability, and the ability to approximate any nonlinear function. For example, in 2000, Huang et al. [4] studied the application of the standard BP algorithm in fault diagnosis of gas turbines in power stations. However, there is a slow convergence rate and it is easy to fall into the local minimum. In 2006, Xie and Shi [5] proposed a dynamic gas turbine fault diagnosis based on RBF neural network. It can effectively identify fault types. In 2012, Jiang and Zhu [6] proposed to apply PNN neural network to gas turbine fault diagnosis system, which has advantages of fast training speed and high diagnostic efficiency. In 2017, Zhu et al. [7] proposed a fault diagnosis method combining fuzzy logic and neural networks. Firstly, the continuous variable membership function is used to represent the fuzzy subset of symptoms. Then the BP neural network is used for training after fuzzification. This method has better training results.

Aiming at the deficiency of BP neural network, a BP neural network method based on combination optimization of N-W and L-M is proposed in this paper. The N-W algorithm is mainly used to
initialize the weights and thresholds of neurons in the hidden layer, so as to reduce the adjustment of the weights and thresholds during the network training process [8]. The L-W algorithm combines the locality of the Gauss-Newton method with the globality of the gradient method to quickly search optimal solutions of the weights and threshold of the BP neural network, thereby accelerating the network convergence speed. This paper establishes a fault diagnosis model based on this method to diagnose gas turbine faults. The diagnosis results show that the BP neural network constructed by this method has fast learning speed and good diagnosis result. The method proposed in this paper provides a new approach for gas turbine fault diagnosis.

2. BP neural network [9]-[10]
BP neural network is a kind of error backpropagation neural network. The BP network is trained by minimizing the value of the objective function through error back propagation. The objective function is Eq. (1):

$$E = \frac{1}{2} \sum_{k=1}^{n} (y_k - o_k)^2$$

(1)

Where,

- $E$: Systematic average error
- $n$: Number of system learning samples
- $y_k$: The target output of the k-th sample
- $o_k$: The actual number of the k-th samples

Traditional BP algorithm uses gradient descent method to adjust weights and thresholds of individual neurons during training of learning samples:

$$\Delta \omega_{ji} = -\alpha \frac{\partial E}{\partial \omega_{ji}}$$

(2)

Where,

- $\alpha$: Learning rate

Specific algorithm derivation refer to literature[9]. The traditional BP neural network mainly has the following two defects. The first is that the learning speed is slow, especially when the learning rate and initialization weights are not selected properly. Second, it is easy to fall into the local minimum during the training process and cannot guarantee convergence to the global minimum point. To solve the above problems, the L-M algorithm can be used to improve the traditional BP algorithm.

3. L-M algorithm [11]-[13]
The L-M algorithm is a combination of the advantages of the Gauss-Newton method and the gradient descent method. It has the global characteristics of gradient descent method, that is, the descent of initial several steps is faster. It also has the local characteristics of Gaussian Newton method, that is, it can produce an ideal search direction near the optimal value. Therefore, the L-M algorithm can effectively overcome the problems that the BP algorithm has a slow convergence rate and is easy to fall into local minimum. It is assumed that the vector composed of each neuron weight and threshold is $x$. $x^{(k)}$ denotes the vector $x$ of the k-th iteration, then the iterative expression of the L-M algorithm is as follows:

$$x^{(k+1)} = x^{(k)} + \Delta x$$

(3)

Where, we have

$$\Delta x = -\left(J^T(x)J(x) + uI\right)^{-1}J(x)e(x),$$
ons use the sigmod function as the activation function. Let \( \omega_k \) be the weight and \( \epsilon_k \) the \( k \)-th network node. It is the identity matrix. The Jacobian matrix \( J(x) \) is a Jacobian.

\[
J(x) = \begin{bmatrix}
\frac{\partial e_1(x)}{\partial x_1} & \frac{\partial e_1(x)}{\partial x_2} & \cdots & \frac{\partial e_1(x)}{\partial x_j} \\
\frac{\partial e_2(x)}{\partial x_1} & \frac{\partial e_2(x)}{\partial x_2} & \cdots & \frac{\partial e_2(x)}{\partial x_j} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\partial e_n(x)}{\partial x_1} & \frac{\partial e_n(x)}{\partial x_2} & \cdots & \frac{\partial e_n(x)}{\partial x_j}
\end{bmatrix}
\]

In the formula, \( u \) is the scale coefficient and \( u > 0 \). \( e_i(x) \) is the error of the \( i \)-th network node. \( I \) is the identity matrix. \( J(x) \) is a Jacobian matrix.

Set the error evaluation function for Eq.\((4)\):

\[
E(x) = \frac{1}{2} \sum_{i=1}^{N} e_i(x) \tag{4}
\]

When the scale coefficient \( u \rightarrow 0 \), then \( uI \rightarrow 0 \), \( \Delta x = -(J^T(x)J(x))^{-1} J(x)e(x) \). This formula is Gauss-Newton method.

When the scale coefficient \( u \rightarrow 1 \), then \( uI \rightarrow I \), \( \Delta x = -(J^T(x)J(x)+I)^{-1} J(x)e(x) \). The L-M algorithm is transformed into a gradient descent method. As the number of successful iterations increases, the value of \( u \) gradually decreases. When the approximation error is at a minimum, the L-M algorithm gradually evolves to a Gauss-Newton method.

3.1. L-M algorithm implementation steps

(1) Set training error allowable values \( \epsilon \), coefficients \( \beta \), and initialize the vector of weights and thresholds \( x^{(0)} \). Let \( k = 0 \), \( u = u_0 \).

(2) Calculate network output and systematic average error \( E(x^{(k)}) \).

(3) If \( E(x^{(k)}) < \epsilon \), go to step (7), otherwise continue.

(4) Calculate the Jacobian matrix \( J(x) \).

(5) Calculate \( \Delta x \). Calculate the error function \( E(x^{(k+1)}) \) with \( x^{(k+1)} = x^{(k)} + \Delta x \) as the weight and the threshold.

(6) If \( E(x^{(k+1)}) < E(x^{(k)}) \), then \( k = k + 1 \), \( u = u/\beta \), and go to step (2). Otherwise, do not update \( x \) (weights and thresholds) this time. Let \( \omega^{(k+1)} = \omega^{(k)} \), \( \mu = \mu \beta \) and go to step (5).

(7) End.

4. N-W variable parameter initialization method [14]

L-M algorithm has the advantage of fast convergence rate. It is also not easy to fall into local minimum. So it is an algorithm suitable for training neural networks. However, when the algorithm is determined, the training process of the network will have a great relationship with the initial weights and thresholds of the network. After Nguyen-Widrow’s analysis of the neurons in the hidden layer, an initialization method for N-W variable parameters was obtained. For the three-layer BP neural network whose structure is n-m-k, the hidden layer neurons use the sigmod function as the activation function, the output layer neurons use the linear function. The BP neural network handles the input data as follows.

\[
\text{Hidden layer: } y_j = f\left[\sum_{i=1}^{n} \omega_{ij} x_i + b_j\right], \quad (j = 1, 2, \ldots, m) \tag{5}
\]
Output layer: \( o_i = \sum_{j=1}^{n} \omega_{ji} y_j + b_i \), \( (l = 1,2,...,k) \) \hspace{1cm} (6)

Where, \( y_j \) is the output of the j-th neuron of the hidden layer. \( \omega_{ji} \) is the weight from the i-th neuron in the input layer to the j-th neuron in the hidden layer. \( f(x) \) is the sigmoid function. \( b_j \) is the threshold of the j-th neuron of the hidden layer; \( o_i \) is the output of the i-th neuron in the output layer; \( \omega_{lj} \) is the weight from the j-th neuron in the hidden layer to the l-th neuron in the output layer. \( b_i \) is the threshold of the i-th neuron in the output layer.

Since the sigmoid function is close to a linear function within a range \([-0.7, 0.7]\] centered on 0. It can be seen from Eq.(5) and Eq.(6) that the output of neurons in the hidden layer is close to a linear function in a certain interval. The length of the interval is related to \( \omega_{ji} \), and the center position is related to \( b_j \). Hidden layer neuron linear interval can average sample input interval by adjusting the initial values of the weights \( \omega_{ji} \) and the threshold \( b_j \) with the N-W method, so as to reduce the adjustment times of the weights and thresholds in the network training process. The adjustment formula is:

\[ \omega_{ji} = \omega_{ji} / |r_i - l_i| \]
\[ |\omega_{ji}| = 1.4H^{1/n} \]
\[ b_j = \text{rand} - |\omega_{ji}||\omega_{ji}| - \omega_{ji}M \]

Where, \( \omega_{ji} \) is the weight from the j-th neuron in the hidden layer to the i-th neuron in the input layer. \( r_i \) is the maximum value of the i-th dimension in the input sample matrix. \( l_i \) is the minimum value of the i-th dimension in the input sample matrix. \( H \) is the number of neurons in the hidden layer. \( n \) is the number of input layer neurons. \( \omega_{ji} \) is the weight vector from the input layer neuron to j-th neuron in the hidden layer, and \( M = [v_1, v_2, ..., v_m]^T \), where \( v_i = (l_i + r_i) / 2 \).

The output layer neurons use a linear function as an activation function, which has nothing to do with the linear interval of the hidden layer neurons. Therefore, the initial value of the weight and threshold which is from the hidden layer to the output layer can be any value within the range of \([-1, 1]\).

When the multi-layer neural network adopts random number to initialize the weights and thresholds of neurons, if the input range of the network is large, the activation function of neurons in the hidden layer is easily saturated, which affects the learning speed of the network. Through the Nguyen-Widrow method, initializing the weights and thresholds of the hidden layer can effectively accelerate the learning speed of the BP network.

5. The process of optimizing BP algorithm based on combination of N-W and L-M

The algorithm proposed in this paper mainly consists of two parts:

1. Firstly, the N-W method is used to initialize the weights and thresholds of the hidden layers in the determined BP neural network.
2. L-M algorithm is used to train learning samples in BP neural network. Then we can test the samples with the trained BP network.

The specific flow of optimizing BP algorithm based on combination of N-W and L-M is shown in Figure 1:
Begin

Determine the BP network structure model

Use N-W method to initialize the weights and thresholds of neurons

Use L-M algorithm to train the BP network

Test the sample with a trained BP network

End

**Figure 1.** Flow of optimizing BP algorithm based on combination of N-W and L-M.

6. **Experimental simulation and result analysis**

In the paper, it adopts the BP neural network structure in the form of 5-11-10. (that is, the number of input, hidden and output neurons is 5, 11, and 10 respectively.) The network error is in the form of a square sum, and the target error is 10-3. The maximum number of training is 10000. The gas turbine fault samples were trained respectively by gradient descent method, conjugate gradient method and combination algorithm of N-W Method and L-M algorithm. The five neurons in the input layer of the BP network are expressed as: the speed change of low pressure compressor ($\delta n_1$), the speed change of high pressure compressor ($\delta n_2$), compression ratio variation of low pressure compressor ($\delta \pi_{LC}$), compression ratio variation of high pressure compressor ($\delta \pi_{HC}$) and the change in fuel flow ($\delta w_f$). The input of the sample data needs to be normalized and the normalized formula is as follows:

$$x^* = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}$$

(10)

The number of input layer neurons is m and the number of hidden layer neurons is n. They have the following approximate relationship [15]:

$$n = 2m + 1$$

(11)

Since the number of input layer neurons of the BP neural network is 5, the number of hidden layers is 11. Experiments have shown that the training results using 11 hidden layer neurons are more effective than those of other hidden layers. The combined vectors of the 10 neurons in the output layer represent 10 types of gas path faults common in gas turbines, respectively. Learning sample data based on an improved BP neural network which is constructed in this paper is shown in Table 1. It can be seen from the table that there are 10 gas fault types (9 fault types and 1 fault-free type) for gas turbine, and there are totally 30 groups of training samples. Among them, data from the first group to the third group represents the data of fault-free system, and the output of the sample target is (0000000000).
Data from the fourth group to the sixth group represents the data of compressor blade fouling fault system. The three groups of data are respectively the fault data of running samples under the condition of 70%, 80%, and 90% of the gas turbine system. The output of the sample target is (0100000000). Similarly, the data from the seventh group to the ninth group indicates the sample data in the event of compressor tip clearance failure. The data of the tenth group to the twelfth group indicates the sample data of the compressor blade corrosion failure in the system, etc. Each type of failure has three groups of learning sample data which corresponds to the data when the gas turbine system is running at 70%, 80%, and 90% operating conditions[16], [17].

Table 1. BP neural network training samples.

| Fault type                      | \(\delta n_1\) % | \(\delta n_2\) % | \(\delta n_{lc}\) % | \(\delta n_{hc}\) % | \(\delta w_f\) % | Target output of the sample |
|---------------------------------|------------------|------------------|---------------------|---------------------|------------------|---------------------------|
| Fault-free                      | -0.050           | -0.050           | -0.050              | -0.050              | -0.050           | 0000000000                |
|                                 | 0.000            | 0.000            | 0.000               | 0.000               | 0.000            | 0000000000                |
|                                 | 0.050            | 0.050            | 0.050               | 0.050               | 0.050            | 0000000000                |
| Compressor blade fouling        | 3.781            | 1.732            | -2.083              | -0.592              | 1.438            | 0100000000                |
|                                 | 4.312            | 1.824            | -2.127              | -0.732              | 1.773            | 0100000000                |
|                                 | 4.823            | 1.738            | -2.462              | -0.672              | 1.896            | 0100000000                |
| Compressor tip clearance        | 1.354            | 0.942            | -1.731              | -0.100              | 0.286            | 0010000000                |
|                                 | 1.654            | 1.042            | -1.891              | -0.121              | 0.364            | 0010000000                |
|                                 | 1.983            | 1.254            | -2.280              | -0.139              | 0.438            | 0010000000                |
| Compressor blade wear           | 1.132            | -0.013           | 0.961               | -0.421              | 0.915            | 0001000000                |
|                                 | 1.412            | -0.011           | 1.182               | -0.531              | 1.143            | 0001000000                |
|                                 | 1.693            | -0.014           | 1.422               | -0.643              | 1.371            | 0001000000                |
| Compressor blade mechanical     | 2.971            | -0.023           | 2.491               | -1.117              | 2.412            | 0000100000                |
| damage                         | 3.593            | -0.026           | 2.957               | -1.328              | 2.861            | 0000100000                |
|                                 | 4.088            | -0.031           | 3.432               | -1.539              | 3.315            | 0000100000                |
| Turbine blade hot corrosion     | 1.012            | -0.319           | 0.872               | -0.037              | 0.123            | 0000010000                |
|                                 | 1.149            | -0.359           | 1.008               | -0.038              | 0.137            | 0000010000                |
|                                 | 1.311            | -0.402           | 1.147               | -0.041              | 0.163            | 0000010000                |
| Turbine blade fouling           | 0.073            | -0.191           | -0.282              | -0.663              | 0.051            | 0000001000                |
|                                 | 0.161            | -0.252           | -0.271              | -0.787              | 0.063            | 0000001000                |
|                                 | 0.251            | -0.303           | -0.262              | -0.928              | 0.082            | 0000001000                |
| Turbine blade wear              | 1.963            | -0.729           | 1.451               | -0.573              | 0.262            | 0000000100                |
|                                 | 2.351            | -0.882           | 1.735               | -0.710              | 0.319            | 0000000100                |
|                                 | 2.740            | -1.039           | 2.011               | -0.842              | 0.373            | 0000000100                |
| Turbine blade mechanical        | 2.521            | -1.113           | 1.521               | -1.403              | 0.382            | 0000000010                |
| damage                         | 3.001            | -1.324           | 1.813               | -1.672              | 0.451            | 0000000010                |
|                                 | 3.482            | -1.531           | 2.101               | -1.933              | 0.519            | 0000000010                |
| Combustion chamber failure      | -2.121           | -0.242           | -0.783              | 1.373               | 1.872            | 0000000001                |
|                                 | -2.523           | -0.292           | -0.934              | 1.632               | 2.781            | 0000000001                |
|                                 | -3.212           | -0.349           | -1.113              | 1.962               | 2.661            | 0000000001                |
6.1 Result analysis
The experimental results are shown in Figure 2, Figure 3, and Figure 4 respectively. They represent the error curve simulation diagram of gas turbine fault learning samples training which respectively use gradient descent method, conjugate gradient method, and L-M optimization algorithm combined with N-W method. From these three graphs, it can be seen that the training times of BP network using the N-W and L-M combined optimization algorithm are much smaller than those of the BP network using the gradient descent algorithm and the conjugate gradient algorithm. It takes only 13 steps to complete the training of BP neural network. Error accuracy reaches 10^-7, and no training trapped in local minima. The conjugate gradient method requires about 170 iterations to make the error accuracy reach 10^-2. Gradient descent method will fall into the local minimum during training of BP neural network, and the error accuracy has not yet reached 10^-1. Therefore, compared with the gradient descent method and the conjugate gradient method to optimize BP algorithm, the BP algorithm based on the combination of N-W and L-M has the advantage of fast training, and it is not easy to fall into the local minimum. It effectively improves the traditional BP neural network.

![Figure 2. Error curve trained by gradient descent method.](image1)

![Figure 3. Error curve trained by conjugate gradient method.](image2)
The training samples are tested by the BP network based on N-W and L-M combinatorial optimization algorithms. The test results are shown in Table 2. It can be seen from Table 2 that the output of the training sample is very close to the target output of the actual sample, which shows that the trained BP algorithm model has a good learning ability.

Table 2. Verification results of training samples.

| Training sample number | Network output |
|------------------------|----------------|
|                        | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  |
| 1                      | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0001 | 0.0000 | 0.0002 | 0.0000 |
| 2                      | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0001 | 0.0000 |
| 3                      | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 4                      | 0.0000 | 1.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 5                      | 0.0000 | 1.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 6                      | 0.0000 | 1.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 7                      | 0.0000 | 0.0000 | 0.9999 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 8                      | 0.0000 | 0.0000 | 0.9999 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 9                      | 0.0000 | 0.0000 | 0.9999 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 10                     | 0.0000 | 0.0000 | 0.0000 | 0.9998 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 11                     | 0.0000 | 0.0000 | 0.0000 | 0.9999 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 12                     | 0.0000 | 0.0000 | 0.0000 | 0.9998 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 13                     | 0.0000 | 0.0000 | 0.0000 | 0.0001 | 0.9995 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |

Figure 4. Error curve trained by the combination of N-W and L-M.
In the test of the test samples, there are totally 150 test samples, and the number of samples for each gas circuit fault type is 15. The test results are shown in Table 3. It can be seen from the table that for the 150 groups of test samples, only three sets of sample data were diagnosed incorrectly, and 147 sets of sample data were correctly diagnosed. The accuracy rate reached 98%, which can meet the requirements of the actual gas turbine fault diagnosis. Therefore, this algorithm has a good practical application value for gas turbine fault diagnosis.

Table 3. The results of test samples.

| Fault types of test samples | The number of test samples | The number of correct diagnoses | The number of incorrect diagnoses |
|-----------------------------|----------------------------|-------------------------------|----------------------------------|
| Fault-free                  | 15                         | 15                            | 0                                |
| Compressor blade fouling    | 15                         | 15                            | 0                                |
| Compressor tip clearance    | 15                         | 14                            | 1                                |
| Compressor blade wear       | 15                         | 15                            | 0                                |
| Compressor blade            | 15                         | 15                            | 0                                |
It can be seen from the above experimental results that the optimized BP algorithm based on the combination of N-W and L-M algorithm has good effects on the training results, the training speed, the test of the training samples, and the results of the test samples. It can make accurate and efficient intelligent fault diagnosis for gas turbine.

6.2. **Comparison with a PNN fault diagnosis method**

For the complex fault diagnosis problem of gas turbine, Jiang and Zhu[6] proposed a probabilistic neural network (PNN) fault diagnosis method. Probabilistic Neural Networks (PNN) is a radial basis network suitable for classification problems. It is a parallel algorithm developed from Bayesian classification rules and Parzen window probability density function. It has the advantages of being fast, accurate, and easy to modify. Here, we use the two methods which are PNN algorithm and the method we proposed in this paper to diagnose the original fault samples respectively. The contrast result is shown in Table 4.

**Table 4.** Comparison of N/W - L/M Algorithm and PNN network diagnosis results.

| Diagnostic method | Number of iterations | Diagnostic time /s | Accuracy rate |
|-------------------|----------------------|--------------------|---------------|
| PNN network       | 20                   | 6.162482           | 95%           |
| N/W - L/M Algorithm | 13                   | 4.352974           | 98%           |

It can be seen from Table 4 that the fault diagnosis of gas turbine based on N/W - L/M Algorithm is superior to PNN network in diagnostic speed and diagnostic accuracy. The combination of N-W and L-M Algorithm has faster network convergence rate than PNN network, and smaller network can fall into the local minima. This algorithm can effectively train the fault diagnosis model of gas turbine. Therefore, the algorithm proposed in this paper has higher diagnostic accuracy than PNN network, It can meet the requirements of gas turbine fault diagnosis.

7. **Conclusion**

In order to solve the problem that BP algorithm has slow convergence rate and is easy to fall into local minimum problem, this paper presents the combination of N-W algorithm and L-M algorithm to optimize BP algorithm and builds a fault diagnosis model to diagnose gas turbine faults based on this method. Comparing this algorithm with BP algorithm based on gradient descent method and BP algorithm based on conjugate gradient method, the experimental results show that the optimized BP algorithm model based on combination of N-W and L-M has better training speed and can effectively avoid falling into local minimum. It meets the real-time requirements for fault diagnosis of gas turbine. When the trained model tests the training samples and the test samples, it can be found that the output value of the training sample is almost consistent with the expected result, and the the diagnosis of the test samples has a good accuracy, which can meet the needs of gas turbine fault diagnosis. Therefore,
this method provides a new approach for fault diagnosis of gas turbines. It is especially suitable for application in diagnostic environments with high real-time requirements and accurate diagnostic results. The method has broad application prospects.

8. References

[1] Xie C L and Dai J M 2010 Review and prospect of gas turbine fault diagnosis technology J. Steam Turbine Technol. 52 1-3
[2] Sina T B 2014 Dynamic neural network based fault diagnosis of gas turbine engines J. Neuro Computing 125 153-165
[3] Jiang K 2016 Research on gas turbine fault diagnosis technology J. Petrochemical Technol. 23 105
[4] Huang X G, Wang Y H and Weng S L 2000 Gas turbine fault diagnosis based on BP algorithm J. Chinese Journal of Electrical Engineer 20 72-74
[5] Xie C L and Shi X C 2006 Gas turbine fault diagnosis based on neural network J. Gas Turbine Technol. 19 58-60
[6] Jiang R, Zhu W 2012 A PNN fault diagnosis method for gas turbine C. World Automation Congress 1-4
[7] Zhu Y J, Ding C and Fan Z J 2017 Research on fault diagnosis of thermal component of gas turbine based on fuzzy neural network J. Electric Automation 110-112
[8] Xu Z N and Lv F C 2006 Nguyen-Widrow method for fault diagnosis of transformer oil and gas analysis J. High Voltage Technol. 32 46-48
[9] Yu J L and Bian S 2014 Transformer fault diagnosis model based on BP neural network J. Journal of System Simulation 26 1343-49
[10] Wu L Y and Wang H 2003 Application of BP neural network expert system in fault diagnosis. Info. Technol. 2 66-68
[11] Xiang W Q, Zhang H and Wang H 2011 J. Power System Protection and Control 100-103
[12] Liu J H and Ren X H 2015 Fault diagnosis of spindle system of CNC machine tool based on BP neural network J. Machine Tools & Hydraulics 43 193-196
[13] Zhu B H, Zhou D, and Chen S X 2011 Using BP neural network improved by L-M method to evaluate slope stability J. Western Exploration Project 23 21-24
[14] Xu Z N, Lv F C and Liu Y P 2015 J. Journal of North China Electric Power University 32 1-4
[15] Yang S, Yao S G, Liu H W and Jiang L 2007 Research and application of BP neural network in expert system development J. Sci. Tech.Engng. 7
[16] Yao J, Li H W and Ma X 2012 Fault diagnosis of hot and cold gas turbine components based on BP neural network J. Gas Turbine Technol. 25 39-43
[17] Peng T 2007 An improved neural network expert system for mechanical fault diagnosis J. Computer Engineering and Appl. 43 232-234

Acknowledgments
This work was sponsored by Shanghai Sailing Program(16YF1404700).