The Circular Self-Configuring Algorithm and its Application to Equipment Fault Diagnosis

Hong Rao, Mingfu Fu, Lian Chen, Mingxiang Xie

1. INTRODUCTION

BP neural network is a common method for the fault diagnosis [1]. But there is no uniform theory for constructing the network frame by far, especially for determining the number of hidden neurons, which is the most crucial part for representing the knowledge in BP network. Recently, the researches focus on how to get the least number of hidden neurons, but any method can’t reduce the number to the least due to the restriction of the network’s initial values [2].

The self-configuring algorithm is a correlatively pruning method of multi-layer forward neural network based on BP network. In order to delete the redundant hidden neurons, the correlation coefficient and dispersion degree between hidden neurons are evaluated, which based on mathematical statistical theory [3]. But the simulation shows that the network is always reduced to different number of hidden neurons, i.e. the network can’t be converged uniformly and reduced to the simplest structure by the self-configuring algorithm.

In order to acquire the uniform network structure and the least hidden neurons, randomness and divide algorithm were introduced to the self-configuring algorithm, which results in the circular self-configuring algorithm. Simulation of the algorithm in MATLAB shows it can reduce the neural network to the simplest structure. At last, the application of the algorithm to the fault diagnosis of centrifugal fan underlines the theoretical results.

2. THE SELF-CONFIGURING ALGORITHM

2.1 The correlation coefficient and dispersion degree

In the same hidden layer, for the hidden node i, the sample output sequence is \( \{V_{ip}\} \), and for the j, it is \( \{V_{jp}\} \), their average value is \( \bar{V}_i \) and \( \bar{V}_j \) [4-5].

Definition 1 The correlation coefficient \( r_{ij} \) between neuron i and j in the same hidden layer is:

\[
r_{ij} = \frac{\sum_p V_{ip} - \bar{V}_i \sum_p V_{jp} - \bar{V}_j \bar{V}_j}{\sqrt{\sum_p (V_{ip} - \bar{V}_i)^2} \sqrt{\sum_p (V_{jp} - \bar{V}_j)^2}}
\]

Definition 2 The output dispersion degree of the hidden node i is calculated as following:

\[
s_i^2 = \sum_p (V_{ip} - \bar{V}_i)^2
\]

2.2 The train condition of self-configuring

The neural network will merge or delete the hidden nodes if the output error square sum of all the samples is smaller than the error threshold of self-configuring.

\[
\frac{1}{2} \sum_p \sum_k \left( O_{kp} - T_{kp} \right)^2 \leq \varepsilon_1
\]

Where \( T_{kp} \) and \( O_{kp} \) is the expected and real value of the p output in the neural network, \( \varepsilon_1 \) is the error threshold of self-configuring, which is little greater than the aim error value.

In a same hidden layer, if the correlation coefficient \( r_{ij} \) and dispersion degree \( s_i^2 \) satisfied the conditions as following, the hidden node i and j can be merged into one node.

\[
|r_{ij}| > \theta_1 \quad \text{and} \quad s_i^2 > \theta_2, \quad s_j^2 > \theta_2
\]

Where \( \theta_1 \) and \( \theta_2 \) is the threshold of the correlation coefficient and dispersion degree.

If the dispersion degree \( s_i^2 \) of the hidden node i satisfied the following condition, the hidden node i can be deleted.

\[
s_i^2 < \theta_2
\]

The \( \theta_1 \) is usually between 0.8~0.9 and \( \theta_2 \) is between 0.01~0.001.
2.3 The self-configuring algorithm

If the correlation coefficient \( |r_{ij}| \) of the hidden node \( i \) and \( j \) in the same layer is a large value, there is a linear regression equation as following:

\[
V_j = aV_i + b \tag{6}
\]

where

\[
a = \frac{\frac{1}{n} \sum_{p=1}^{n} V_{wp}V_{wp} - \overline{V}_j\overline{V}_j}{\frac{1}{n} \sum_{p=1}^{n} V_{wp}^2 - \overline{V}_j^2} \tag{7}
\]

\[
b = \overline{V}_j - a\overline{V}_i \tag{8}
\]

And then the input for any node \( k \) in next layer is:

\[
w_{ik}V_i + w_{ik}\overline{V}_j + w_{kb} \ast 1 + \sum_{i \in l\_j} w_{ki}V_i = w_{ik}V_i + w_{ik}(aV_i + b) + w_{kb} \ast 1 + \sum_{i \in l\_j} w_{ki}V_i = (w_{ki} + aw_{ik})V_i + (w_{kb} + bw_{ik}) \ast 1 + \sum_{i \in l\_j} w_{ki}V_i \tag{9}
\]

Where \( w_{kb} \) is the weight of node \( k \).

So merging hidden nodes method is to delete the hidden node \( j \), and to revise the weight of any node \( k \) in the next layer.

Let

\[
w_{ki}^{new} = w_{ki}^{old} + aw_{kj}^{old} \tag{10}
\]

\[
w_{kb}^{new} = w_{kb}^{old} + bw_{kj}^{old} \tag{11}
\]

When the dispersion degree \( s_i^2 \) is a smaller value, the output value can be replaced by the average:

\[
V_i = \overline{V}_i, \text{then the input of any node } k \text{ in next layer is:}
\]

\[
w_{ik}V_i + w_{ik} \overline{V}_j + w_{kb} \ast 1 + \sum_{i \in l\_j} w_{ki}V_i = (w_{ik} + \overline{V}_j)w_{ik} \ast 1 + \sum_{i \in l\_j} w_{ki}V_i = (w_{ki} + w_{ik}\overline{V}_j) + \sum_{i \in l\_j} w_{ki}V_i \tag{12}
\]

So deleting hidden nodes method is to delete the hidden node \( i \), and to revise the bias of any node \( k \) in the next layer.

Let

\[
w_{kb}^{new} = w_{kb}^{old} + \overline{V}_jw_{ki}^{old} \tag{13}
\]

3. THE CIRCULAR SELF-CONFIGURING ALGORITHM

3.1 The conception of randomness

**Definition 3** The randomness

Randomness is a positive integer, which denotes the allowed random selection times for the initial value and bias value when the self-configuring algorithm can’t prune the nodes successfully.

When the self-configuring algorithm can’t prune effectively, the initial value and bias value are rechoosed, which represent the new initial space. The bigger randomness, the more possibility of the initial space it keeps. The larger initial space, the less of the network depends on. In theory, the self-configuring algorithm can converge as long as the randomness is big enough.

**Definition 4** The circular self-configuring algorithm

The circular self-configuring algorithm is the self-configuring algorithm with randomness. When the randomness is zero, it belongs to the general self-configuring algorithm.

3.2 The algorithm thinking

The circular self-configuring algorithm thought is shown in Figure 1, which originated from three parts:

1. There is possibility for the network to converge to the consistent frame by self-configuring algorithm itself.

2. To convert the disadvantage factors of the randomness for the initial value and the bias value to the network’s advantage, by which the self-configuring algorithm can converge uniformly. The essential is to make use of the possibility statistic theory, and comes to the conception of randomness.

3. The algorithm which stems from the divide thinking is to establish a convergent uniformly network. Once a step hardly finishes the constringency, the task can be divided into several subtasks. That is, the network is pruned into a simplified structure firstly, and then takes this network as a beginning, rechooses the initial values, continues to prune until all the redundant nodes are merged or deleted. Then the simplest structure BP network is formed.

In Fig 1, \( h_n \), \( h_k \) are the given hidden nodes which contain redundancy, under the ideal condition, the network should be converged and pruned to the simplest structure with different hidden nodes, which is shown by the bold line.

This simplified network contains \( h_{\text{min}} \) hidden nodes. But the simulation in MATLAB shows that the network pruned by the self-configuring algorithm just reduce the number of hidden nodes from \( h_n \) to \( h_{n-i} \), rather than the least number \( h_{\text{min}} \) . That is, the networks with different number of hidden nodes cannot always converge to the same number \( h_{\text{min}} \). But the network trained by the circular self-configuring algorithm with a random initial value and bias value, which is shown by broken line, can prune the number of hidden layer nodes from \( h_n \) to \( h_{n-i} \), \( h_{n-i} \) and \( h_{\text{min}} \), prune the hidden nodes from \( h_k \) to \( h_{k-j} \), \( h_{k-j} \) and \( h_{\text{min}} \). At last, the networks with different initial hidden nodes are all reduced to the simplest structure with \( h_{\text{min}} \) hidden nodes. The flow chart of circular self-configuring algorithm is shown in Fig 2.
4 THE SIMULATION OF ALGORITHMS

4.1 The BP network model for simulation in MATLAB

Take the centrifugal fan fault diagnosis as an example, set the eight kinds of fault diagnosis including unbalanced, misalignment and so on as the output of the network, the vector which unified disposed by different frequencies spectrum peak energy in 9 frequency bands of vibrating signal as a feature to form the training sample, take the corresponding mode of fault diagnosis as the goal output, which shown in Table 1 and Table 2. We construct the BP neural network with two layers. There are 20 nodes in hidden layer. The initial value and bias value are initialized randomly between [-0.5, 0.5]. The activation function of hidden layer and output layer are all the sigmoid function\[6-8\].

The network was trained respectively in the mode of traditional self-configuring algorithm and circular self-configuring algorithm, the randomness value was 3 for the circular self-configuring algorithm.

4.2 The network training comparison

The results of the network trained respectively by traditional self-configuring algorithm and circular self-configuring algorithm are shown in Table 3. It shows that the network frame 9-15-8 is reduced to 9-13-8 in Experiment T1_1, the network frame 9-15-8 is reduced to 9-12-8 in Experiment T1_2, the network frame 9-20-8 is simplified to 9-13-8 in Experiment T1_3, the network frame 9-20-8 simplified 9-13-8 in Experiment T1_4.

The fault data in Table 4 was trained in the established BP network, and the result was shown in Table 5.

The data in table 5 shows that the result was closer to the unbalanced fault mode, so we concluded that the fault is unbalanced. Examined by the expert of the factory, the fault was rotor unbalanced exactly. In a word, the BP network simplified by the circular self-configuring algorithm is successfully applied to centrifugal fan’s fault diagnosis.

5 CONCLUSIONS

According to the above experiments, the circular self-configuring algorithm based on the randomness and divide thinking can converge the network uniformly and
reduce the neural network to the simplest structure. The simplified network reduced the complication of the knowledge representation greatly. The application of the algorithm to the centrifugal fan fault diagnosis underlines the theoretical results.

References