Journal of Theoretical and Applied Information Technology
10th February 2013. Vol. 48 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE IMPROVEMENT OF RESEARCH AND APPLICATION OF BP NEURAL NETWORK IN MEASUREMENT ROBOT ATMOSPHERIC REFRACTION COEFFICIENT CORRECTION

ZHOU, JING[1] ZHANG, XIQIAN[2] TANG, QINGRONG[3]
School of Civil Engineering, Chongqing University
No.83, Shabei Street, Shapingba District, Chongqing, 400045
E-mail: [1]1000zzz@sina.com.cn , [2]wfzhwfzh@yahoo.com.cn

ABSTRACT

Measurement Robot is a kind of full-automatic and high-precision measuring instrument, but its height measurement has a biggish error because of the effect of atmospheric refraction. Using the artificial neural network to correct the atmospheric refraction is the relatively advanced solving method at present. However, the traditional BP neural network has some disadvantages. This paper proposes out an improved BP neural network, by improving its weight, the neural network has the self-adapting ability and quicker learning speed. According to the experiments, the predictive effect is good and the height measurement problem of the Measurement Robot has been solved well.

Keywords: *Measurement Robot, Atmospheric Refraction, BP Neural Network*

1. INTRODUCTION

Measurement Robot(also called as Georobot) is a video imaging system integrated with step motor and CCD image sensor based on the total station , and it is improved to what it is today with the intelligentized control and the application soft wares. Measurement Robot directly measures the three-dimensional coordinate of a certain point, which is convenient and quickly, and its two-dimensional location precision usually can meet all the application demands. However, the height measurement of the robot probably has a biggish error, and the main reason for the error of the total station trigonometric leveling precision is the atmospheric refraction error, as it is difficult to set the precise refraction coefficient ---K value in time. Currently the common methods for calculating K value include using the apposition observation dispersion and using the known precise dispersion, both have certain disadvantages; now the method of introducing neural network to correct the atmospheric refraction in time is a relatively advanced method. This kind of correction, in fact, is a prediction. It uses the existing data which include some relatively variables to do some training. When the training demand is satisfied, the prediction can be made.

Currently, the mainly used neural network is BP network, which has the following advantages: the highly nonlinear mapping ability, generalization ability, fault-tolerant capability, and easy implementation and so on, so it can used in the multi-parameter nonlinear prediction. But its disadvantages are also very obvious. Its network convergence is so slow that it will easily goes into the local minimum value, and resulting in the extension of the neural network training period.

Lots of works have been carried out to improve the BP network mainly based on mathematics optimization theory. Literature [5] suggests a new method based on LM pre-feedback network learning algorithm and this method has the same speed of LM algorithm but it is of less complexity than LM. Literature[6] proposes out a second-order learning algorithm based on Block Hessian matrix and it avoids the disadvantage of BP network such as lower convergence speed. Literature[7] figures out the best learning parameter BP network and its result shows it can accelerate the convergence speed.

So this paper will discuss how to improve the BP network, to speed up its network convergence and reduce the training period. For this purpose, we add the momentum items of BP network and make the learning rate adapt by itself.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2. CORRECTION ON THE NEURAL NETWORK OF THE ATMOSPHERIC REFRACTION

2.1 Construction and Improvement of BP Network

BP network includes the input layer, the hidden layer and the output layer. The basic idea of BP calculation is that the learning of the whole network includes two processes: the forward-propagating of the input signal and the backward-propagating of the error. The forward-propagating process means that the sample signal is input in the input layer, processed by the network weight, threshold and the transition function of neurons, and output from the output layer. If the error between the output value and the expected value is beyond the limitation, then it will be corrected and goes into the error back-forward process. The process in which the weight is being corrected all the time is the training process of the network. This circulation will not stop until the output error is reduced to the allowable value or the set training time is due. The specific calculation will not be described here. The following is just the correction function of the weight:

$$\left.\begin{aligned} w_{kl}^{"}(n_0+1) &= w_{kl}^{"}(n_0)+\eta\sum_{pl=1}^{p}\delta_{kl}^{pl}x_k^{"pl} \\ w_{jk}^{"}(n_0+1) &= w_{jk}^{"}(n_0)+\eta\sum_{pl=1}^{p}\delta_{jk}^{pl}x_j^{"pl} \\ w_{ij}^{"}(n_0+1) &= w_{ij}^{"}(n_0)+\eta\sum_{pl=1}^{p}\delta_{ij}^{pl}x_k^{"pl} \end{aligned}\right\} \quad (1)$$

In the above function, δ_{sq}^{pl} （sq为ij、jk、kl）is the error of each layer, pl is the No. pl sample; η is the step length, also called the learning rate; n0 is the No.n0 training. In the standard BP arithmetic, the leaning step length η is invariable. As the gradient change of the approaching error curve of BP network is uneven, if we use the fixed step length, when η is small, the convergence is slow in the relatively flat area of the error curve; and when η is big, a oscillation will be easily caused in the gorge area of the error curve. The self-adapting BP arithmetic proposed in this paper is an improved arithmetic aiming at the mentioned disadvantage.

There are many improved BP arithmetic. One is to change the step length and one is to add momentum items. The self-adapting arithmetic proposed out in this paper is based on the mentioned two improved methods. While we add the momentum items, we make the learning rate adapt by itself. The specific method is as follows:

Add a momentum item to function (l) as follows:

$$\left.\begin{aligned} w_{kl}^{"}(n_0+1) &= w_{kl}^{"}(n_0)+\eta(n_0)\sum_{pl=1}^{p}\delta_{kl}^{pl}x_k^{"pl}+\alpha\Delta w_{kl}^{"}(n_0) \\ w_{jk}^{"}(n_0+1) &= w_{jk}^{'}(n_0)+\eta(n_0)\sum_{pl=1}^{p}\delta_{jk}^{pl}x_j^{"pl}+\alpha\Delta w_{jk}^{"}(n_0) \\ w_{ij}^{"}(n_0+1) &= w_{jk}(n_0)+\eta(n_0)\sum_{pl=1}^{p}\delta_{ij}^{pl}x_i^{"pl}+\alpha\Delta w_{ij}^{"}(n_0) \end{aligned}\right\} \quad (2)$$

In the function, α is the momentum factor, the added $\Delta w_{sp}^{"}(n_0)$ is to memorize the correction direction of the weight in last moment, the step length—the initial value has a very small effect on the following iteration calculation. This paper set η（0）as 1, the other values are calculated as the following method according to the change of the iteration error:

$$\eta(n_0)=\varphi\eta(n_0-1) \quad \varphi>1 \text{ when } \Delta E<0 \qquad (3)$$

$$\eta(n_0)=\beta\eta(n_0-1) \quad \beta<1 \text{ when } \Delta E>0 \qquad (4)$$

Here φ、β are constant, and:

$$\Delta E=\Delta E_{total}(n_0)-\Delta E_{total}(n_0-1) \qquad (5)$$

$$\Delta E_{total}=\frac{1}{2}\sum_{pl=1}^{p}\sum_{l=1}^{m}(t_l^{pl}-y_l^{pl})^2 \qquad (6)$$

In the function, p is the total number of samples; m is the number of the neurons in the output layer. After we made the mentioned improvement, when the total error is reduced, i.e. ΔE<0, the iteration goes into the flat area of the error curve, if we multiply the learning rate with a number larger than 1, the step length is increased and the iteration number will be reduced; while the total error is increased, i.e. ΔE>0, the iteration goes into the area of fierce changes in the error curve. At this time, if we multiply the learning rate with a number smaller than 1, it is helpful to reduce the step length and the error. Thus the learning rate will change according to the change of the total error.

2.2 Content of Modeling

Firstly, we ensure the parameters in relative with the refraction coefficient correction are the terrain, time interval, temperature and air pressure, the

Journal of Theoretical and Applied Information Technology
10th February 2013. Vol. 48 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

output result is the refraction coefficient. Using the neural network to simulate the relative between the terrain, time interval, temperature, air pressure and the refraction coefficient; the model uses the three-layer BP network, the number of neurons in the input layer is 4, respectively meaning the terrain, time interval, temperature and air pressure. At present, there is no standard method to make sure the number of layers in the hidden layer and the number of neurons. In the practical processing, we usually make sure of those by trail. After many trails, the finally chosen neural network in this paper is the three-layer forward neural network 4:3:1; the number of neurons in the output layer is 1, i.e. the refraction coefficient K.

2.3 Model Training

As to the trained model, we need to do the prediction train. When the train is passed, it will be able to be used as a prediction model to predict the want-dealing problems. Otherwise, we need to reset the training parameters of the network model and retrain them. Using the relative error in the proving sample subset to prove whether the gotten model meets the prediction demand, and if the proving samples whose relative error is smaller than or equal to 10% occupy more than 80% of the proving sample subset, the training is over.

3. CALCULATION EXAMPLES

3.1 Training Data

We take three points in the construction plant, respectively name them as A0, A1 and A2. We set a Measurement Robot in A0, observe A1 and A2 from 7:00am to 7:00pm with an interval of 2 hours, and calculate the refraction coefficient. Here, the terrain factor for A0—A1 is 0.8, for A0—A2 is 0.2, the calculated refraction coefficient of each time is shown in Table 1.

3.2 Fitting Results

We construct, compare and select models by taking experimental data in Table 1 as training sample subset. The parameters in relative with the neural network training are: the training process shows the frequency is 10, the maximum training step number is 1 000, the system error is 1E-4. If using Matlab to calculate, the iteration calculation is done after 35 calculations, the speed is so fast. The fitting result of this network as to the training and the proving sample subset is shown in Table 2.

Table.1 Refraction Coefficients In Different Areas And Times

Point Number	Terrain	ime	Temperature	Air Pressure	Refraction Coefficient
1	0.8	0.2	21	957	0.16
2	0.8	0.8	22	955	0.15
3	0.8	0.8	23	954	0.14
4	0.8	0.8	25	952	-0.13
5	0.8	0.8	28	949	-0.14
6	0.8	0.5	30	946	-0.18
7	0.8	0.5	33	942	-0.19
8	0.8	0.2	31	943	-0.20
9	0.8	0.2	28	948	-0.24
10	0.8	0.2	26	950	-0.14
11	0.8	0.8	22	954	0.15
12	0.2	0.5	21	956	0.16
13	0.2	0.5	23	954	0.17
14	0.2	0.8	25	953	0.14
15	0.2	0.8	27	949	0.17
16	0.2	0.5	29	947	0.15
17	0.2	0.8	32	943	0.14
18	0.2	0.2	28	948	0.12
19	0.2	0.2	25	952	0.16
20	0.2	0.8	22	955	0.17

Table 2 Fitting Result

No.	Practical Measuring Value	Fitting Value	Error	Relative Error（%）
1	0.16	0.163	-0.003	1.9
2	0.15	0.152	-0.002	1.3
3	0.14	0.138	0.002	1.4
4	-0.13	-0.133	0.003	2.3
5	-0.14	-0.14	0.00	0
6	-0.18	-0.177	-0.003	1.7
7	-0.19	-0.192	0.002	1.1
8	-0.20	-0.195	-0.005	2.5
9	-0.24	-0.246	0.006	2.5
10	-0.14	-0.141	0.001	0.7
11	0.15	0.153	-0.003	2
12	0.16	0.159	0.001	0.6
13	0.17	0.168	0.002	1.2
14	0.14	0.141	-0.001	0.7
15	0.17	0.172	-0.002	1.2
16	0.15	0.148	0.002	1.3
17	0.14	0.14	0.00	0
18	0.12	0.119	0.001	0.8
19	0.16	0.163	-0.003	1.9
20	0.17	0.173	-0.003	1.8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. CONCLUSIONS

From Table 2 we can see that the maximum relative error of the constructed model is only 2.3%, proving that this neural network well fits the nonlinear relation between different terrains, times, temperatures, air pressures and the refraction coefficient.

Using neural network to correct the refraction coefficient in time, taking terrain, time, temperature and air pressure as the network input, the refraction coefficient as network output, and using the trained neural network model and parameters, we can correct the refraction coefficient under a certain condition of constructions in time.

The method using BP network to correct the refraction coefficient also has some limitations because that to correct the refraction coefficient we need to collect the information in the scene and make experiments. The experimental data can only reflect the local refraction situation in a certain period. If we can use neural network to construct models, to build up a more elaborate refraction coefficient model data base, finally to provide the building construction monitoring and controlling with accurate and in-time precise positioning data is really feasible.

REFRENCES:

[1] Harris D, Cun YL. In Proving generalization using double back-Pro-Pagation, IEEE Trans Neural netW0rks, Vol. 3, 1992, pp. 991 一 997.

[2] Andreas S. Bernardo A. Huberman, “Predicting Sunspots and exchange rates with connectionist networks. Nonlinear Modeling and Forecasting, Vol. 5.2002. pp.395 一 431.

[3] Yao X, Evolving artificial neural networks, Proc. IEE, Vol. 9, 1999, pp. 1423 一 1447.

[4] Stefan Luká, Milan Zák and Ján Hardo. Deformation Measurements of the Most Important Structures in Slovak Nuclear Power Plants. FIG Working Week．Paris,2003

[5] Yang Bin，Lie Zaiping，Xia Yaoxian.A quick learning algorithm based on improved conjugation gradient method. electronic xuebao，Vol. 2, 2012，pp.351 一 355.

[6] Yi-Jen Mang， ChinngLin. A second-order leaning algorithm for mutilayer network based on Block Hessian matrix,Neural Networks, Vol.8, 1998, pp,1617 一 1622.

[7] Xiao-Hu Yu, Guo-An Chen. Efficient Backpropagation Leaning Using Optimal Leaning Rate and Moment. Neural Network, Vol.10, 1997,pp.10(3):517 一 527.