

# Prediction of Spindle Thermal Deformation and Displacement Using Back Propagation Neural Network

Bo-Lin Jian, Yu-Syong Guo, Chi-Hsien Hu, Li-Wei Wu, and Her-Terng Yau\*

Department of Electrical Engineering, National Chin-Yi University of Technology, Taichung, Taiwan

(Received February 4, 2019; accepted December 13, 2019)

**Keywords:** back propagation neural network, partial least squares regression, spindle thermal deformation, thermal error, thermal effect

Over the years, machine tool manufacturers have moved steadily towards the enhancement of machining accuracy to improve the quality of finished products. In this study, the thermal deformation of a machine spindle, which has a profound effect on machining accuracy, was investigated. The temperatures of the front and rear spindle bearings, and of the environment as well as the Z-axis displacement on a model MC4200BL CNC lathe (Hybrid Sphere) were measured under long-term operating conditions. Measurements were carried out at spindle speeds of 1000, 1500, 2000, 2500, and 3000 rpm, and the data were used to establish a model for the prediction of spindle displacement. A back propagation neural network (BPNN) was used to establish the model and explore adjustments of the training function, the data training ratio, and the number of neurons in the hidden layer. Results of the experiments showed that the coefficient of determination ( $R^2$ ) of the prediction model derived from the best parameters can be up to 0.9948. This was much better than the 0.8273 achieved by the partial least squares regression method.

## 1. Introduction

Over the years, machine tool manufacturers have moved steadily towards the enhancement of machining accuracy to improve the quality of finished products. Errors in accuracy, which can have a large impact on production, include those caused by geometry, by changes in temperature, and by the cutting force itself.<sup>(1,2)</sup> To enhance production capacity, machines must work faster, and when working time increases, so does the temperature. However, when processing time is increased, thermal error becomes more important and can seriously affect machining accuracy. According to Bryan, thermal error accounts for between 40 and 70% of the total machining error.<sup>(3,4)</sup> Thermal errors can arise from either an internal or external heat source.<sup>(5)</sup> Internal heat is mainly generated by moving parts of the machine, such as the motor and spindle bearings, as well as the cutting process itself. External heat comes from the operating environment. In a conventional spindle, the main heat source is the bearings.<sup>(6)</sup> To enhance machining accuracy, errors should be reduced and compensation for displacement applied.<sup>(7)</sup> Many studies related to this problem have been carried out. Zhu *et al.*<sup>(8)</sup>

---

\*Corresponding author: e-mail: htyau@ncut.edu.tw  
<https://doi.org/10.18494/SAM.2020.2606>

used multiple regression to construct a thermal error prediction model and succeeded in reducing the thermal displacement error of 20  $\mu\text{m}$  down to 4  $\mu\text{m}$ , a reduction of about 75%. Xiaohong *et al.*<sup>(9)</sup> used the rough set theory to analyze the relationship between temperature and error, and were able to reduce thermal errors to less than 5  $\mu\text{m}$  using a back propagation neural network (BPNN) training model. Pahk and Lee<sup>(10)</sup> proposed a spindle temperature thermal error measurement system that used multiple linear regression, a neural network (NN), and the system identification method to establish a spindle-thermal error model. Lin and Chang<sup>(11)</sup> used complex multivariate regression analysis to predict spindle thermal displacement at any particular time. Li *et al.*<sup>(12)</sup> used the bat algorithm (BA) to optimize a traditional BPNN. It is always difficult to confirm the number of hidden layers in a traditional BPNN, and using the BA-back propagation (BP) solved this modeling problem. Ma *et al.*<sup>(13)</sup> used particle swarm optimization (PSO) and the genetic algorithm (GA) to improve the shortcomings of BPNN, and were able to improve processing accuracy from 67 to 78 and 89%, respectively. There are many types of artificial neural networks (ANNs), and the BPNN has the additional feature of nonlinear input–output mapping.<sup>(14,15)</sup>

To solve the problem of inaccuracy caused by the thermal displacement of the spindle, it was necessary to establish a spindle thermal displacement model that could be used to predict these changes. Traditionally, regression analysis and the generalized least squares are used for this. Although these methods are relatively simple and easy to use, the prediction results are less than ideal. Thermal error is dynamic and nonlinear, and in this study, an appropriate ANN was therefore selected to suit the features of the data and establish a model for prediction. However, Li *et al.*<sup>(12)</sup> observed that traditional BPNN often has trouble confirming the number of neurons in the hidden layer. To attain the objective of this study, it was necessary to find the best BPNN parameter training function and ratio of data training, as well as overcome the problem of the number of hidden layer neurons. This allowed the prediction ability of the model to be enhanced, which reduced deviations in prediction, improved machining accuracy, and established an excellent model that was superior to partial least squares regression (PLSR).

## 2. Method

### 2.1 Experimental equipment and architecture

In this study, we adopted the ball-type CNC lathe (model MC4200BL) from Mike Machine Industry Co., Ltd., together with the SYNTEC21-TA-type controller of Syntec Technology Co., Ltd., and the Posa spindle (model TAC-10-CY), which is a belt-type spindle. Temperature sensors are mainly divided into contact and noncontact types. The noncontact type requires much equipment to go with, such as the assist optical system, and the equipment cost is much higher than that of the contact type. In addition, for the machine tool, the noncontact type is more difficult to install. Therefore, contact temperature sensors are buried in the front and rear bearings of the spindle. In this study, we adopted the DS18B20 temperature sensor; it can measure temperature in the range from  $-55$  to  $125$   $^{\circ}\text{C}$  and has a resolution of  $0.5$   $^{\circ}\text{C}$ , and the sampling frequency is 5 Hz. Its special feature is to convert the measured temperature

data into digital format, and the accuracy will not be affected during the receiving process. Moreover, the microcontrol board Arduino Uno equipped with a USB serial port was used as the data capture module. As for the laser displacement meter, we used KEYENCE's IL-S025 and IL-1000 to measure the displacement through noncontact measurement. The IL-S025 has a measurement range of 20 to 30 mm and a resolution of 1  $\mu\text{m}$ , and the sampling frequency is 5 Hz. Finally, the NI-6341 capture card was used to capture signals.

The experimental flow chart of the study is shown in Fig. 1. The machine spindle speeds used in the experiments were 1000, 1500, 2000, 2500, and 3000 rpm. The spindle was run at each temperature for a measured time. The sensors used to measure the temperature were embedded in the front and rear bearing housings on the spindle assembly of a model MC4200BL CNC lathe (Hybrid Sphere). Measurements were made at 5 s intervals and the displacement of the Z-axis was measured at the same time using a laser displacement meter. To establish the BPNN model, the training function and the proportion of training data were adjusted, and the coefficient of determination ( $R^2$ ) based on the number of hidden layer neurons was found. The root mean square error (RMSE), ratio of performance to deviation (RPD), mean square error (MSE), and mean absolute error (MAE) were all used to determine the degree of prediction and to explore the parameters needed to establish the best model compared with PLSR.

## 2.2 BPNN

The BPNN, a classic ANN, is a combination of multilayer perceptron (MLP) and error back propagation (EBP). The BPNN is a supervised learning network; this means that it requires a set of training materials that includes input features and target results. The main feature of the BPNN is the updating of the weight between points by calculating the error to enhance the

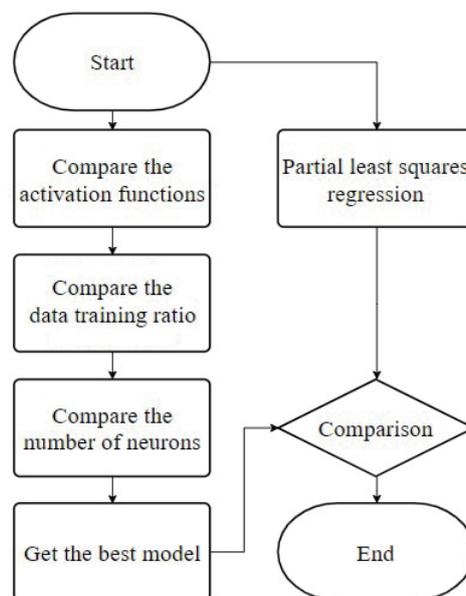


Fig. 1. Experimental flowchart.

prediction ability of the model. The advantages of the BPNN are that it can perform nonlinear input and output mapping and also use the gradient descent method<sup>(16)</sup> to reduce errors and give better results.

### 2.2.1 BPNN theory

Figure 2 is the basic architecture of the BPNN. The input layer is orange, the hidden layer is blue, and the output is yellow. The blue arrow represents the weight of the input layer connecting the hidden layer, the yellow arrow represents the weight of the hidden layer connecting the output layer, and the BPNN flow chart is shown in Fig. 3.

The sigmoid function was used to calculate the equation of the hidden layer output value, as shown in Eqs. (1) and (2).<sup>(17)</sup>

$$net_k = \sum_i w_{ik} x_i \quad (1)$$

$$h_k = \frac{1}{1 + \exp(-net_k)} \quad (2)$$

In these equations,  $h_k$  is the input value of the  $k$ th hidden unit,  $net_k$  is the weighted product sum of the input values,  $x_i$  is the input value of the  $i$ th input unit, and  $w_{ik}$  is the connected weight value between the  $i$ th input unit and the  $k$ th hidden unit.

By using the sum of the squared errors of all the output nodes of the network as the objective function, the error calculation equation is as shown in Eq. (3).<sup>(18)</sup>

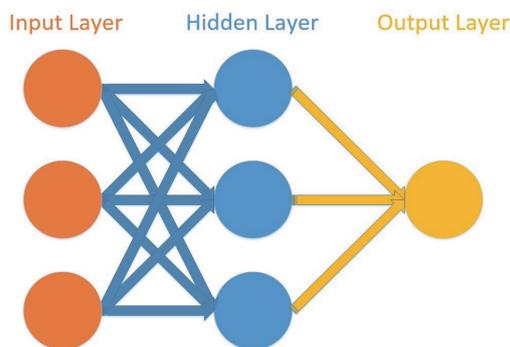


Fig. 2. (Color online) BPNN architecture.

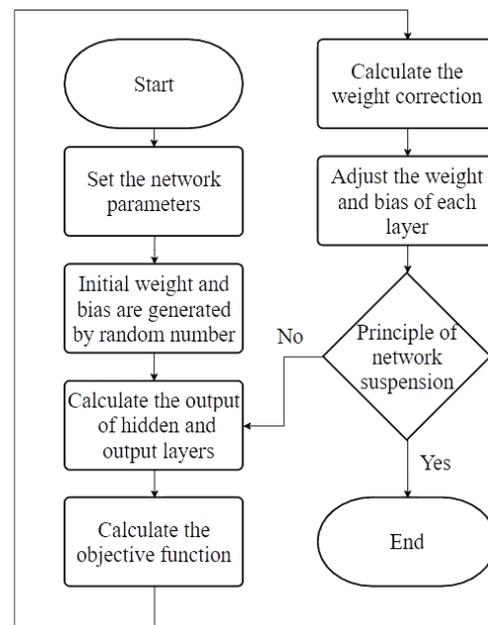


Fig. 3. BPNN flow chart.

$$E = \frac{1}{2} \sum_{i \in \text{outputs}} (y_{(i)} - \hat{y}_{(i)})^2 \quad (3)$$

In this equation,  $E$  is the error,  $y_{(i)}$  is the actual value, and  $\hat{y}_{(i)}$  is the predicted value.

### 2.2.2 Training function

In this study, we used three training functions, as follows:

1. Trainlm: The updated weight and bias values were optimized according to Levenberg–Marquardt.<sup>(19)</sup> The BPNN model achieves the fastest convergence, but it requires a lot of random access memory (RAM).

2. Trainbr: This minimizes the linear combination of square error and weight according to the Bayesian Regulation,<sup>(20)</sup> and confers the generated network with good generalized quality. It takes slightly longer than the other methods, but is suitable for more complicated data.

3. Trainscg: According to the scaled conjugate gradient method,<sup>(21)</sup> it reduces the time taken to search the network when adjusting the direction; it consumes fewer resources and is suitable for use with less RAM.

### 2.3 PLSR

PLSR is a regression modeling method suitable for multi-dependent and multi-independent variables. In the modeling process, the principal component analysis (PCA)<sup>(22)</sup> is extracted as much as possible from the independent and dependent variables, and the correlation between the extracted principal components is then maximized. Simply put, PLSR is a combination of three methods: PCA, canonical correlation analysis (CCA), and multiple linear regression.<sup>(23)</sup>

The main components are extracted from the input and output actual values; the equations are as follows.<sup>(14)</sup>

$$\mathbf{X} = \mathbf{U}\mathbf{a}^T + \mathbf{X}_0 \quad (4)$$

$$\mathbf{Y} = \mathbf{K}\mathbf{y}^T + \mathbf{Y}_0 \quad (5)$$

In these equations,  $\mathbf{X}$  is the independent variable matrix, and  $\mathbf{Y}$  is the dependent variable matrix. Moreover,  $\mathbf{U}$  and  $\mathbf{K}$  are the principal component matrices of  $\mathbf{X}$  and  $\mathbf{Y}$ ,  $\mathbf{a}^T$  and  $\mathbf{y}^T$  are the load vector matrixes of  $\mathbf{X}$  and  $\mathbf{Y}$ , and  $\mathbf{X}_0$  and  $\mathbf{Y}_0$  are the residual matrices of  $\mathbf{X}$  and  $\mathbf{Y}$ , respectively.

The regression equation of PLSR is<sup>(24)</sup>

$$y = \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_i x_i + \varepsilon. \quad (6)$$

In this equation,  $\beta$  is the regression coefficient,  $\varepsilon$  is the error,  $x_i$  is the  $i$ th independent variable, and  $y$  is the dependent variable.

## 2.4 Model verification coefficient

To verify the prediction ability of the model, the  $R^2$ ,  $RMSE$ ,  $RPD$ ,  $MSE$ , and  $MAE$  were used. The  $R^2$  values lies between  $0 < R^2 < 1$ , but the model has a better prediction ability when the  $R^2$  value is close to 1. The formula is as shown in Eq. (7) below.<sup>(25)</sup>

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} = 1 - \frac{\sum_i (f_i - y_i)^2}{\sum_i (y_i - \bar{y})^2} \quad (7)$$

In Eq. (7),  $SS_{res}$  is the residual sum of squares,  $SS_{tot}$  is the total sum of squares,  $f_i$  is the  $i$ th predictive value,  $y_i$  is the  $i$ th actual value, and  $\bar{y}$  is the average value.

$RMSE$ ,  $MSE$ , and  $MAE$  were used to calculate the errors; the smaller the error, the better the prediction ability. The following three equations [Eqs. (8)–(10)] can be used to calculate the three types of error.<sup>(26)</sup>

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (f_i - y_i)^2} \quad (8)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (f_i - y_i)^2 \quad (9)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| \quad (10)$$

In these equations,  $f_i$  is the  $i$ th predictive value and  $y_i$  is the  $i$ th actual value.

When  $RPD < 1.4$ , the prediction ability of the model will be poor, when  $1.4 < RPD < 2$ , the prediction ability will be fair, and when  $RPD > 2$ , the prediction ability will be excellent. The  $RPD$  equations are as shown in Eqs. (11) and (12).<sup>(27)</sup>

$$RPD = \frac{SD}{RMSE} \quad (11)$$

$$SD = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (12)$$

In the equations,  $SD$  is the standard deviation,  $y_i$  is the  $i$ th actual value, and  $\bar{y}$  is the average value.

The several verification coefficients shown above can be used to compare the prediction ability of the model with respect to the number of neurons in the hidden layer, the training function, and the data training ratio to find the best parameters.

### 3. Experimental Results and Discussion

#### 3.1 BPNN parameter adjustment and PLSR model

In this study, the training temperature and displacement data after adjustments of the training function, the data training ratio, the number of hidden layer neuron interval, and the number of hidden layer neurons were used for comparison with five indicators, namely,  $R^2$ ,  $RMSE$ ,  $RPD$ ,  $MSE$ , and  $MAE$ , and as references for the enhancement of the prediction ability of the model.

##### 3.1.1 Comparison of training function

As mentioned in the first chapter, the BPNN has a disadvantage in that it is difficult to confirm the number of neurons in the hidden layer. Therefore, in this study, the BPNN parameters were adjusted to find the best parameters. With the other parameters fixed, the training results were taken as the average of 10 training exercises with three types of training function: Trainlm, Trainbr, and Trainscg, and the results of each spindle speed under different training functions,  $R^2$ ,  $RMSE$ ,  $RPD$ ,  $MSE$ , and  $MAE$ , were compared [Figs. 4(a)–4(e)].

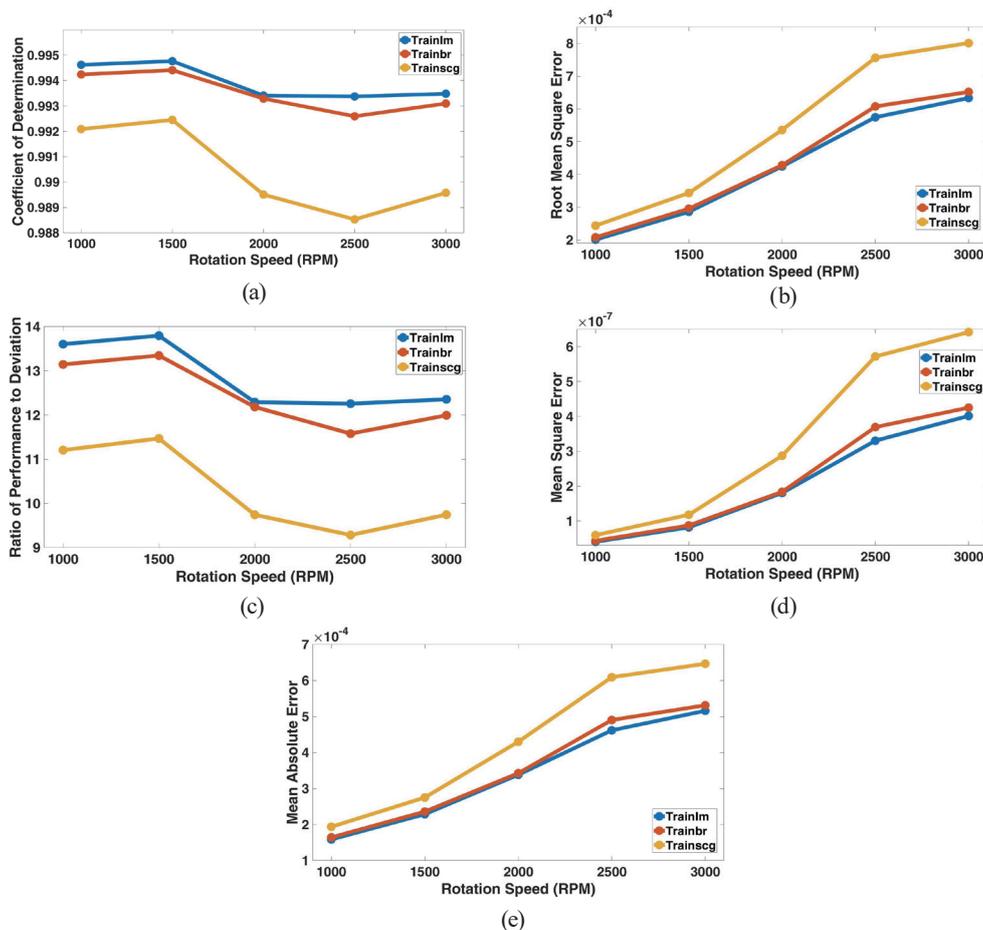


Fig. 4. (Color online) BPNN training function comparison. (a)  $R^2$ , (b)  $RMSE$ , (c)  $RPD$ , (d)  $MSE$ , and (e)  $MAE$ .

According to the results shown in Figs. 4(a)–4(e), the values of  $R^2$  and  $RPD$  using Trainlm at each speed were higher than those using Trainbr and Trainscg, and the values of  $RMSE$ ,  $MSE$ , and  $MAE$  were lower than those using Trainbr and Trainscg. It is clear that Trainlm had the highest prediction effect. The analysis shows that the prediction abilities of these training functions, for spindle thermal displacement prediction, ranked from high to low are Trainlm, Trainbr, and Trainscg.

### 3.1.2 Data training ratio

The results shown in Sect. 3.1.1 indicate that Trainlm had the best prediction ability at data ratios of 70 and 80%. Each training data ratio was the average of 10 training sessions. The comparison results are shown in Figs. 5(a)–5(e).

Figures 5(a)–5(e) show that when the data training ratio of the BPNN is 80%, the values of  $R^2$  and  $RPD$  at each rotation speed are higher than 70%, but the values of  $RMSE$ ,  $MSE$ ,

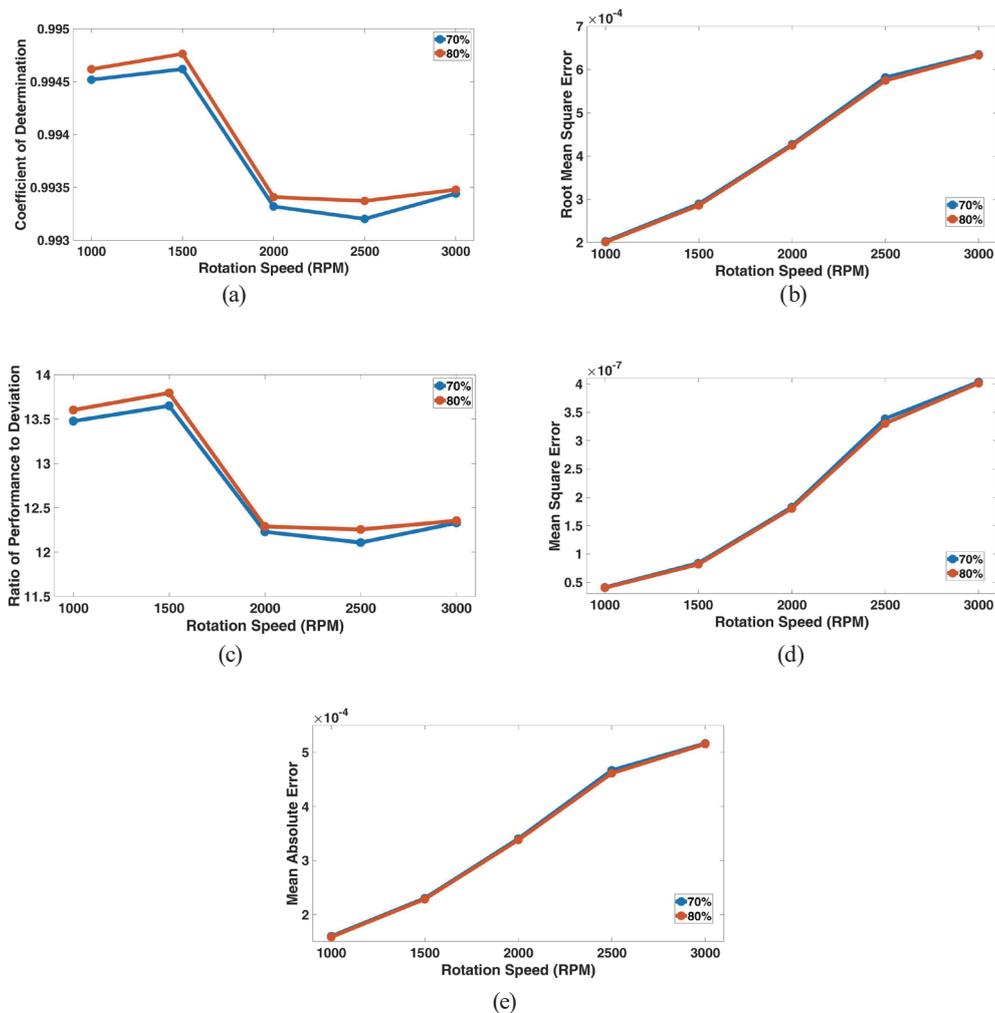


Fig. 5. (Color online) Model data training ratio of BPNN. (a)  $R^2$ , (b)  $RMSE$ , (c)  $RPD$ , (d)  $MSE$ , and (e)  $MAE$ .

and *MAE* are lower than 70%. The verification coefficients show that 80% has the highest prediction effect. The analysis showed that the data training ratio of the BPNN was predictive of the spindle thermal displacement.

### 3.1.3 Hidden layer neuron number intervals

The results described in Sect. 3.1.1 showed that Trainlm had the best prediction ability, and in Sect. 3.1.2, it can be seen that the best BPNN prediction was at a training data ratio of 80%. Under these circumstances, the number of hidden layer neurons increased by 5 each time, training for each hidden layer was performed 10 times, and the average was used. The results are shown in Figs. 6(a)–6(e).

According to the results shown in Figs. 6(a)–6(e), when the number of neuron intervals in the hidden layer of the BPNN is 40, the values of  $R^2$  and *RPD* at 1000, 1500, 2000, and 2500 rpm are higher than those in the other hidden layers, and the value of *RMSE*, *MSE*, and *MAE* are

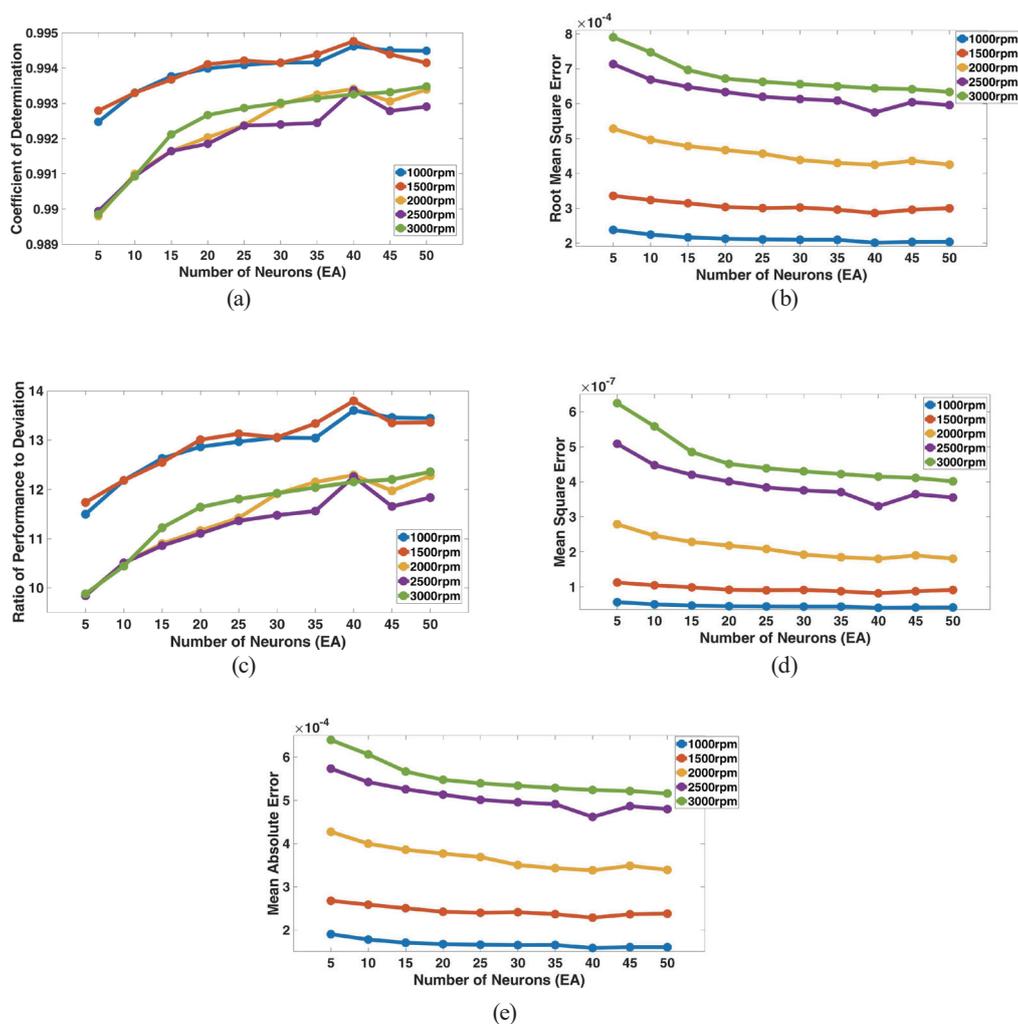


Fig. 6. (Color online) BPNN hidden layer neuron interval. (a)  $R^2$ , (b) *RMSE*, (c) *RPD*, (d) *MSE*, and (e) *MAE*.

lower than those in the other hidden layers. The verification coefficients show that the highest prediction rate is coincident with the number of hidden neuron intervals of around 40. Further analysis confirmed that for this displacement model, the hidden layer interval of 40 gives the best prediction rate.

### 3.1.4 Selection of the number of hidden layer neurons

All the results in Sects. 3.1.1 to 3.1.3 show that for Trainlm, the data training ratio of BPNN is 80%, and the number of hidden layer neurons that gives the best prediction is 40. The number of neurons in the hidden layers from 35 to 50 is increased by one each time. Training was performed 10 times and an average was taken; the comparison results are shown in Figs. 7(a)–7(e).

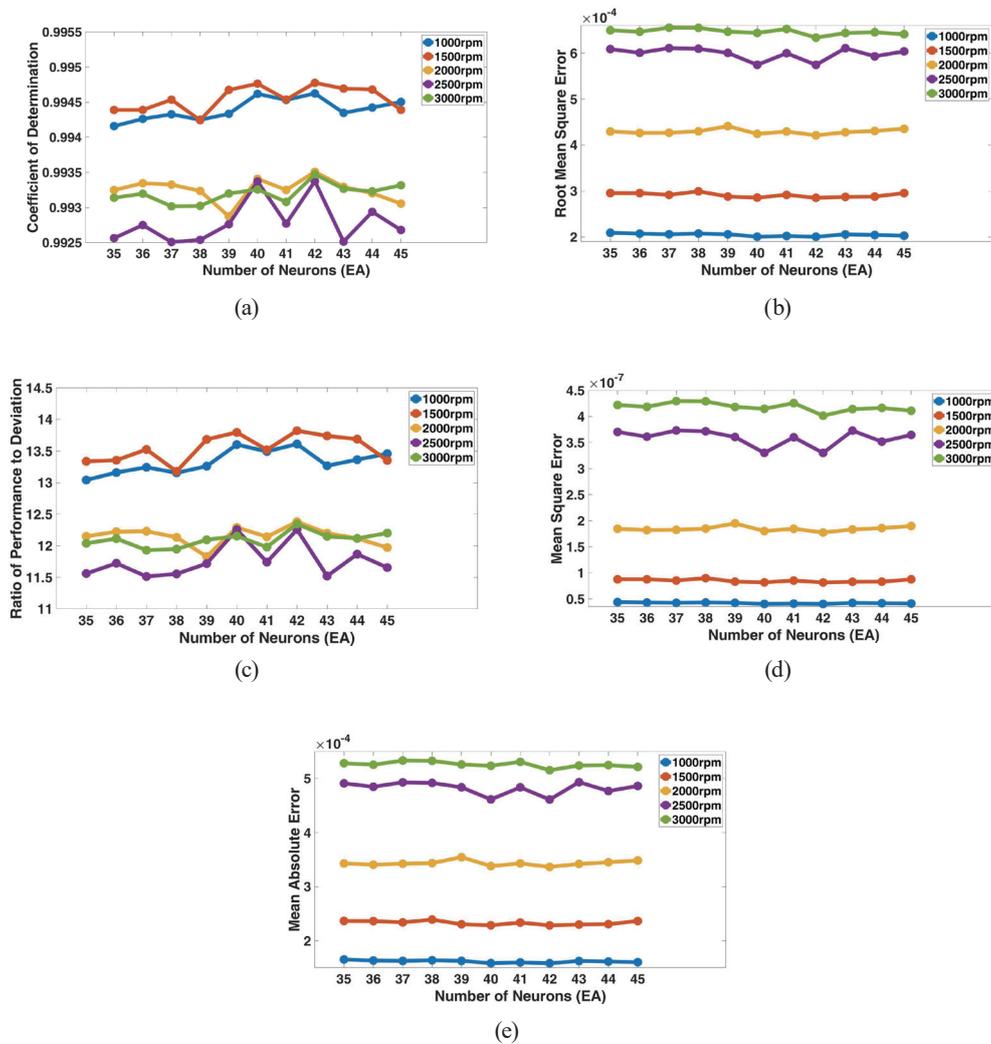


Fig. 7. (Color online) Number of hidden layer neurons of BPNN. (a)  $R^2$ , (b)  $RMSE$ , (c)  $RPD$ , (d)  $MSE$ , and (e)  $MAE$ .

According to the results shown in Figs. 7(a)–7(e), when the number of neurons in the hidden layer of the BPNN is 42, the values of  $R^2$  and  $RPD$  at each rotation speed are higher than those in the other hidden layers. On the other hand, the values of  $RMSE$ ,  $MSE$ , and  $MAE$  are lower than those in the other hidden layers. The verification coefficients show the highest prediction effect with 42 neurons in the hidden layer and further analysis confirms this. From the results described in Sects. 3.1.1 to 3.1.4, the number of hidden layer neurons of 42 gives the best results and is the best parameter to use for the prediction of the thermal deformation of the spindle.

### 3.2 Prediction of displacement

Training was carried out using the experimental results described in Sects. 3.1.1 to 3.1.4, at the spindle rotation speeds of 1000, 1500, 2000, 2500, and 3000 rpm, and the results are shown in Figs. 8–12.

Figure 8 shows the thermal displacement model at a spindle rotation speed of 1000 rpm, where  $R^2$  was 0.9947,  $RMSE$  was 0.000199,  $RPD$  was 13.74,  $MSE$  was  $3.95 \times 10^{-8}$ , and  $MAE$  was 0.000157. The orange line is the actual displacement value, whereas the blue line is the predicted displacement value.

Figure 9 shows the thermal displacement model at a spindle speed of 1500 rpm, where  $R^2$  was 0.9948,  $RMSE$  was 0.000284,  $RPD$  was 13.84,  $MSE$  was  $8.09 \times 10^{-8}$ , and  $MAE$  was 0.000227.

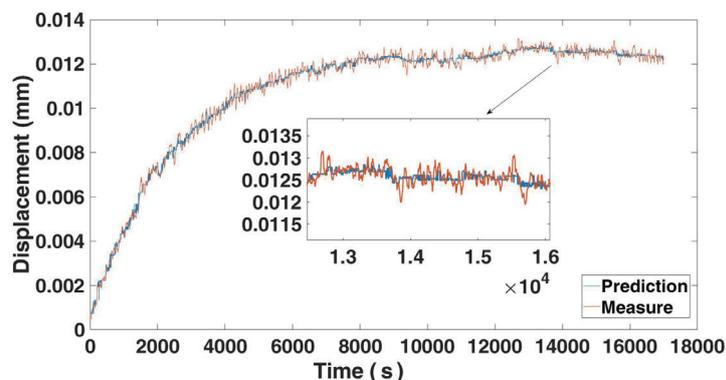


Fig. 8. (Color online) Result of thermal displacement predicted by BPNN at a spindle rotation speed of 1000 rpm.

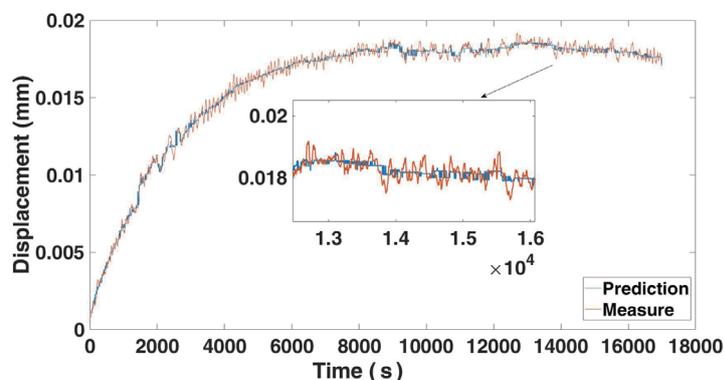


Fig. 9. (Color online) Result of thermal displacement predicted by BPNN at a spindle speed of 1500 rpm.

Figure 10 shows the thermal displacement model at a spindle rotation speed of 2000 rpm, where  $R^2$  was 0.9936,  $RMSE$  was 0.000418,  $RPD$  was 12.45,  $MSE$  was  $1.75 \times 10^{-7}$ , and  $MAE$  was 0.00033.

Figure 11 shows the thermal displacement model at a spindle speed of 2500 rpm, where  $R^2$  was 0.9935,  $RMSE$  was 0.000556,  $RPD$  was 12.42,  $MSE$  was  $3.21 \times 10^{-7}$ , and  $MAE$  was 0.000452.

Figure 12 shows the thermal displacement model at a spindle speed of 3000 rpm, where  $R^2$  was 0.9936,  $RMSE$  was 0.000562,  $RPD$  was 12.53,  $MSE$  was  $3.15 \times 10^{-7}$ , and  $MAE$  was 0.000446.

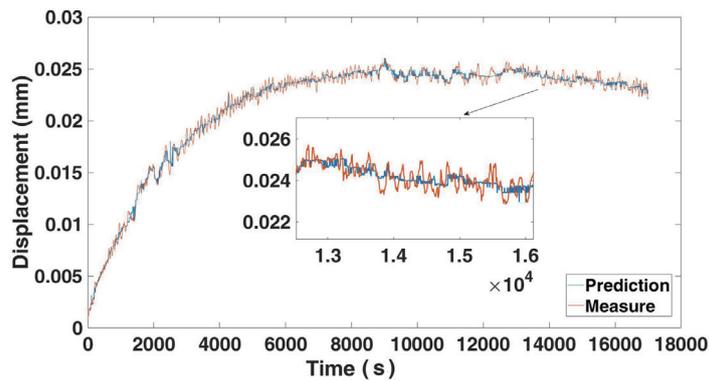


Fig. 10. (Color online) Result of thermal displacement predicted by BPNN at a spindle speed of 2000 rpm.

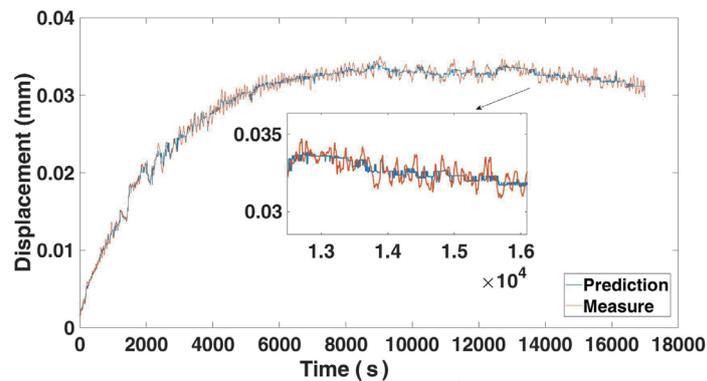


Fig. 11. (Color online) Result of thermal displacement predicted by BPNN at a spindle speed of 2500 rpm.

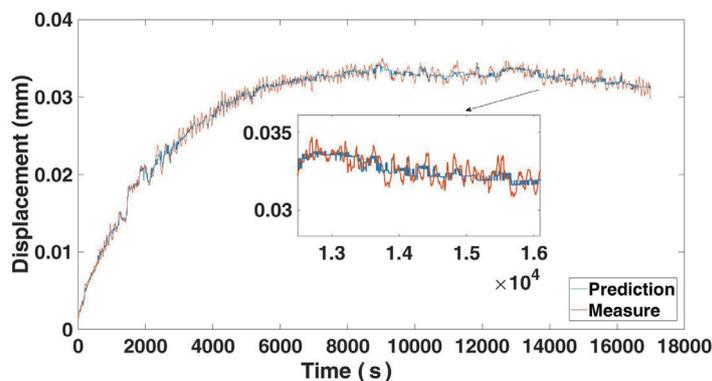


Fig. 12. (Color online) Result of thermal displacement predicted by BPNN at a spindle speed of 3000 rpm.

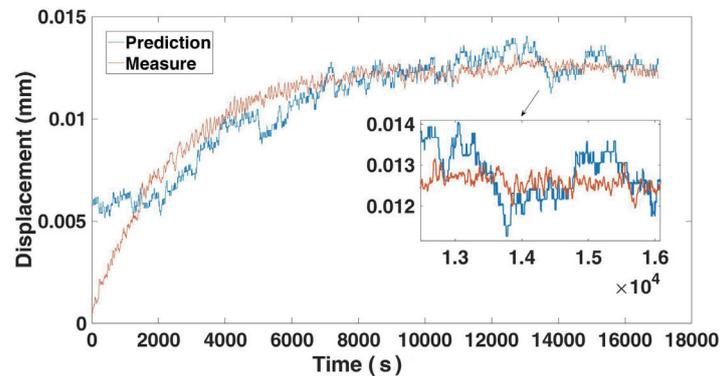


Fig. 13. (Color online) Result of thermal displacement predicted by PLSR at a spindle speed of 1000 rpm.

To summarize the results shown in Figs. 8–12, in the model trained with the best parameter, the predicted value is very close to the actual value, the lowest  $R^2$  value is 0.9935, and the highest is 0.9948, which is a very high prediction rate. The experiments clearly show that the BPNN established by the best parameter can effectively enhance the prediction ability of the model.

Figure 13 shows the thermal displacement model predicted by PLSR at a spindle speed of 1000 rpm, where  $R^2$  was 0.8273,  $RMSE$  was 0.0011,  $RPD$  was 2.27,  $MSE$  was  $1.29 \times 10^{-6}$ , and  $MAE$  was 0.000807. It can be seen that the predicted displacement value was significantly different from the actual displacement value. At other speeds, PLSR showed even greater differences between the predicted and measured displacement values. The BPNN model  $R^2$  established by the best parameters can be up to 0.9948, compared with the partial least squares regression where  $R^2 = 0.8273$ .

#### 4. Conclusion

In this study, temperature sensors were used to measure thermal changes in front and rear spindle bearings of a model MC4200BL CNC lathe, as well as the temperature of the environment. A laser displacement meter was used to measure dimensional changes in the Z-axis at the same time. A BPNN was used and parameters including the training function, the data training ratio, and the number of neurons in hidden layers were adjusted to determine the accuracy of the prediction of judgment models with five different indicators, namely,  $R^2$ ,  $RMSE$ ,  $RPD$ ,  $MSE$ , and  $MAE$ . Comparisons revealed that the model showing the best prediction was generated using the Trainlm parameter Training Function. The data training ratio was 80% and the number of neurons in the hidden layer was 42. This parameter, generated from the experiment, was used to establish the model, and the results showed very good prediction, where  $R^2$  was 0.9948,  $RMSE$  was 0.000284,  $RPD$  was 13.84,  $MSE$  was 0.0000000809, and  $MAE$  was 0.000227; the predicted displacement values were very close to the actual values. A comparison of BPNNs established with the best parameters with the partial least squares regression model showed that the BPNN using the best parameters has a better prediction ability than PLSR regardless of  $R^2$  or the maximum error. Moreover, the verification index of  $R^2$  shows

that of the BPNN to be 0.9948, which is much higher than the 0.8273 of the PLSR. These best BRNN parameters effectively enhance the robustness and predictive accuracy of the model.

### Acknowledgments

This work was supported in part by the Ministry of Science and Technology of the ROC, Taiwan, under Contract Nos. MOST 107-2218-E-167-004 and 108-2218-E-167-001.

### References

- 1 R. Ramesh, M. A. Mannan, and A. N. Poo: *Int. J. Mach. Tools Manuf.* **40** (2000) 1235. [https://doi.org/10.1016/S0890-6955\(00\)00009-2](https://doi.org/10.1016/S0890-6955(00)00009-2)
- 2 S. R. Postlethwaite, J. P. Allen, and D. G. Ford: *Proc. Institution of Mechanical Engineers, Part B: J. Eng. Manuf.* **213** (2016) 1. <https://doi.org/10.1177/095440549921300101>
- 3 J. Bryan: *CIRP Ann.* **39** (1990) 645. [https://doi.org/10.1016/S0007-8506\(07\)63001-7](https://doi.org/10.1016/S0007-8506(07)63001-7)
- 4 D. A. Krulwich: *Mechatronics* **8** (1998) 395. [https://doi.org/10.1016/S0957-4158\(97\)00059-7](https://doi.org/10.1016/S0957-4158(97)00059-7)
- 5 R. Ramesh, M. A. Mannan, and A. N. Poo: *Int. J. Mach. Tools Manuf.* **40** (2000) 1257. [https://doi.org/10.1016/S0890-6955\(00\)00010-9](https://doi.org/10.1016/S0890-6955(00)00010-9)
- 6 I. Mancisidor, M. Zatarain, J. Munoa, and Z. Dombovari: *Adv. Mater. Res.* **223** (2011) 622. <https://doi.org/10.4028/www.scientific.net/AMR.223.622>
- 7 M. Weck, P. McKeown, R. Bonse, and U. Herbst: *CIRP Ann.* **44** (1995) 589. [https://doi.org/10.1016/S0007-8506\(07\)60506-x](https://doi.org/10.1016/S0007-8506(07)60506-x)
- 8 R. Zhu, S. Dai, Y. Zhu, X. Wu, and Y. Guo: 2008 IEEE Int. Conf. Industrial Engineering and Engineering Management (2008) 2056. <https://doi.org/10.1109/IEEM.2008.4738233>
- 9 R. Xiaohong, S. Yong, Z. Tianpeng, X. Weidong, and Y. Yinggao: 2011 Int. Conf. Electric Information and Control Engineering (2011) 630. <https://doi.org/10.1109/ICEICE.2011.5778165>
- 10 H. J. Pahk and S. W. Lee: *Int. J. Adv. Manuf. Tech.* **20** (2002) 487. [https://doi.org/DOI 10.1007/s001700200182](https://doi.org/DOI%2010.1007/s001700200182)
- 11 Z. C. Lin and J. S. Chang: *Int. J. Adv. Manuf. Tech.* **34** (2007) 556. <https://doi.org/10.1007/s00170-006-0613-4>
- 12 Y. Li, J. Zhao, and S. J. Ji: *Int. J. Adv. Manuf. Tech.* **97** (2018) 2575. <https://doi.org/10.1007/s00170-018-1978-x>
- 13 C. Ma, L. Zhao, X. S. Mei, H. Shi, and J. Yang: *Int. J. Adv. Manuf. Tech.* **89** (2017) 3071. <https://doi.org/10.1007/s00170-016-9254-4>
- 14 J. Qiao, G. Wang, W. Li, and X. Li: *Neural Netw.* **104** (2018) 68. <https://doi.org/10.1016/j.neunet.2017.10.006>
- 15 Q. Cheng, Z. Qi, G. J. Zhang, Y. S. Zhao, B. W. Sun, and P. H. Gu: *Int. J. Adv. Manuf. Tech.* **83** (2016) 753. <https://doi.org/10.1007/s00170-015-7556-6>
- 16 C. Jin, S. W. Jin, and L. N. Qin: *Appl. Soft Comput.* **12** (2012) 2147. <https://doi.org/10.1016/j.asoc.2012.03.015>
- 17 Y. Luo, B. Lin, C.-B. Wen, and M. Luo: *Digital Chin. Medi.* **1** (2018) 84. [https://doi.org/10.1016/S2589-3777\(19\)30010-2](https://doi.org/10.1016/S2589-3777(19)30010-2)
- 18 X. Yang, Q. Zhou, J. Wang, L. Han, R. Zhou, Y. He, and K.-C. Li: *Annal. Nucl. Energy* **132** (2019) 576. <https://doi.org/10.1016/j.anucene.2019.06.034>
- 19 S. Mammadli: *Procedia Comput. Sci.* **120** (2017) 602. <https://doi.org/10.1016/j.procs.2017.11.285>
- 20 M. Aucejo and O. De Smet: *Mech. Syst. Sig. Process.* **126** (2019) 98. <https://doi.org/10.1016/j.ymsp.2019.02.021>
- 21 R. R. F. Amaral, M. V. Ribeiro, and E. R. de Aguiar: *Neurocomputing* **355** (2019) 57. <https://doi.org/10.1016/j.neucom.2019.05.002>
- 22 J. C. Boulet, D. Bertrand, G. Mazerolles, R. Sabatier, and J. M. Roger: *Chemom. Intell. Lab. Syst.* **120** (2013) 116. <https://doi.org/10.1016/j.chemolab.2012.11.002>
- 23 M. Zhang, H. L. Mu, G. Li, and Y. D. Ning: *Energy* **34** (2009) 1396. <https://doi.org/10.1016/j.energy.2009.06.032>
- 24 S. Chen and Y. Tsai: 2016 Int. Conf. Advanced Materials for Science and Engineering (ICAMSE) (2016) 673. <https://doi.org/10.1109/ICAMSE.2016.7840292>
- 25 Y. Dai, J. Guo, L. Yang, and W. You: *Procedia Comput. Sci.* **147** (2019) 519. <https://doi.org/10.1016/j.procs.2019.01.235>
- 26 H. Yang, X. Li, and B. Wu: *Transp. Res. Procedia* **25** (2017) 3840. <https://doi.org/10.1016/j.trpro.2017.05.284>
- 27 J. Tan, R. Li, Z. T. Jiang, S. H. Tang, and Y. Wang: *J. Food Compos. Anal.* **77** (2019) 20. <https://doi.org/10.1016/j.jfca.2019.01.001>

## About the Authors



**Bo-Lin Jian** received his B.S. degree from the Department of Electrical Engineering, National Formosa University in 2009 and his M.S. degree in Materials science and engineering from the National Taiwan University of Science and Technology in 2011. His Ph.D. degree was awarded by the Department of Aeronautics and Astronautics, National Cheng Kung University in 2017. He is an assistant professor in the Department of Electrical Engineering, National Chin-Yi University of Technology, Taichung, Taiwan. His current research interests include signal and image processing, machine learning, and control systems.



**Yu-Syong Guo** received his B.S. degree from National Chin-Yi University of Technology, Taichung, Taiwan, in 2017, where he has been working toward his M.S. degree in electrical engineering since 2017. His research interests include integrated systems, Internet of Things, and chaotic systems.



**Chi-Hsien Hu** received his B.S. degree from the Department of Electrical Engineering, National Chin-Yi University of Technology in 2018.



**Li-Wei Wu** received his B.S. and M.S. degrees from the Department of Electrical Engineering, National Chin-Yi University of Technology in 2017 and 2019, respectively.



**Her-Terng Yau** received his B.S. degree from the National Chung Hsing University, Taichung, Taiwan, in 1994, and his M.S. and Ph.D. degrees from the National Cheng Kung University, Tainan, Taiwan, in 1996 and 2000, respectively, all in mechanical engineering. He is currently a professor in the Department of Electrical Engineering, National Chin-Yi University of Technology, Taichung, Taiwan. His research interests include energy converter control, system control of mechatronics, nonlinear system analysis, and control. He is the author of more than 150 research articles on a wide variety of topics in mechanical and electrical engineering.