

## Research Article

# Application of Principal Component Analysis-Assisted Neural Networks for the Rotor Blade Load Prediction

Jiahong Zheng<sup>1</sup>, Shuaike Jiao<sup>1</sup> and Ding Cui<sup>2</sup>

<sup>1</sup>Institute of Flight Test Technology, Chinese Flight Test Establishment, Xi'an 710089, China

<sup>2</sup>Institute of Applied Electronics, China Academy of Engineering Physics, Mianyang 621900, China

Correspondence should be addressed to Jiahong Zheng; 040715316@163.com

Received 16 January 2021; Revised 5 April 2021; Accepted 13 May 2021; Published 24 May 2021

Academic Editor: Zhiguang Song

Copyright © 2021 Jiahong Zheng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a novel approach of principal component analysis- (PCA-) assisted back propagation (BP) neural networks for the problem of rotor blade load prediction. 86.5 hours of real flight data were collected from many steady-state and transient flight maneuvers at different altitudes and airspeeds. Prediction of the blade loads was determined by the PCA-BP model from 16 flight parameters measured and monitored by the flight control computer already present in the helicopter. PCA was applied to reduce the dimension of the flight parameters influencing the component load and eliminate the correlation among flight parameters. Thus, obtained principal components were used as input vectors of the BP neural network. The combined PCA-BP neural network model was trained and tested by real flight data. Comparison of this model and to a BP neural network model as well as to a multiple linear regression (MLR) model was also done. The results of comparison demonstrate that the PCA-BP model has higher prediction precision with an average error of 2.46%, while 4.49% for BP and 10.20% for MLR. The results also reveal that the PCA-BP model has a shorter convergence path than the BP model. This method not only is useful in establishing the load spectra of helicopter rotor in-service where installation of strain gauges is impractical but also can reduce the cost of installation and maintenance measured by strain gauges.

## 1. Introduction

The fatigue life of helicopter structures plays an essential role in helicopter-maintenance work, as components of a helicopter could be fatigue-damaged under various working conditions with loads. Reliable measurement and prediction of fatigue damage then would be the key for timely replacement of necessary components and comply to the Strength Regulations for Military Helicopters. The common practice in determining the fatigue life applies the Palmgren-Miner cumulative damage theory, where one takes the loads in different maneuvers as input data [1–3].

A common method of obtaining the component loads is to directly measure by installing strain gauges. Although this method is able to get a high accuracy for component loads, this way will bring some disadvantages. Firstly, this can be expensive considering installation and maintenance costs [2]. Secondly, it is often impractical because strain gauges normally cannot be mounted on an in-service helicopter

[4]. Besides, the installation of strain gauges would lead to the occupation of helicopter space [5]. In addition, it often takes more than one year to collect data and needs a long time analysis as the load spectrum involves a series of flight maneuvers [6].

Some researchers try to predict component loads by a mathematical model and an intelligent model actively. A mathematical model is an important method for the prediction of rotor blade loads. However, the variables and the component loads are intertwined in such a complex way that currently, a precise mathematical model is still not an option [7]. Therefore, many studies have proposed to predict component loads by using an intelligence model that can model the complex relationship for improving the prediction precision. For example, prediction of the component loads is determined by a linear regression or neural network model from the roll, pitch, airspeed, and other flight parameters measured by the flight control computer on the helicopter [8].

Cabell et al. predicted oscillatory loads using a neural network for on-line health monitoring of flight essential components in an AH-64A helicopter. The linear model was also used to predict the load, and its accuracy was obviously lower than that of the neural network, especially at higher load values that cause fatigue damage [9]. Haas et al. used a more complex method to model the load model derivation of the helicopter rotor system. Both linear regression and neural networks were used to create load models. Nonlinear effects were found in the data and explained using a linear model with derived parameters created by a nonlinear mapping function or using a neural network model [10]. Allen and Dibley applied neural networks to model the wing loads of aircrafts. The linear model was established first as the starting point of the network training, which improved the accuracy of the neural network model [11]. Gómez-Escalonilla et al. developed a parametric full-scale fatigue monitoring system for an Airbus A330 using an artificial neural network with several strain gauges installed on some areas of the wings and fuselage [4]. Cooper and DiMaio predicted the static load on a wing rib of aircraft using an artificial neural network. This was achieved by using strain values obtained from the static test as an input parameter [7]. Elshafey et al. researched on the use of neural networks to predict structural response on structures [12]. De Paula et al. predicted nonlinear unsteady aerodynamic loads for NACA0012 airfoil using neural networks [13].

Despite insufficient accuracy in the fatigue damage estimation, in these papers, encouraging progress has been made. However, the methods developed in these papers are rather inflexible for generalization when one attempts to feed the neural network with the influencing factors directly. One way to solve this problem is to preprocess the data with the principal component analysis (PCA) method. The original data of the influencing factors is very likely intercorrelated and possibly redundant. PCA is designed to reduce the data to the most compact and independent sets while retaining the data's information. Compared to the original data, data processed with PCA mitigates the demand of the complexity in the neural network. In this paper, a combination of PCA and the back propagation (BP) neural network method was experimented and compared to the pure BP method, as well as to multilinear regression (MLR). The results show evident enhancement in the accuracy of fatigue-damage estimation by the combined method of BP with PCA.

## 2. PCA and BP Neural Network Structure

*2.1. Principal Component Analysis.* The principal component analysis is a data-dimension-reduction technique. Constructing an appropriate set of linear combinations of the original indicators, a series of unrelated comprehensive indicators are produced, from which a few comprehensive indicators are selected such that they contain as much information contained in the original indicators as possible. Thus, it is possible to use fewer indicators to explain most of the variations in the original data [14].

Suppose a data set has  $n$  samples, each expressed by an  $m$ -dimensional transposed vector,  $(x_1, x_2, \dots, x_m)$ , so the original data matrix can be represented by an  $n \times m$  matrix  $\mathbf{X} = (x_{ij})_{n \times m}$ , with each row representing a sample. The PCA consists of the following steps, which are shown in Ref. [15–18].

*2.1.1. Normalization of Raw Data.* The original variables need to be standardized to eliminate the influence of different dimensions, and it can be done according to the following two equations:

$$\bar{x}_{ij} = \frac{x_{ij} - \min(x_{*j})}{\max(x_{*j}) - \min(x_{*j})}, \quad (i = 1, 2, \dots, n, j = 1, 2, \dots, m), \quad (1)$$

where  $\min(x_{*j})$  stands for the smallest number of the  $n$  entries of the  $j$ th column in the matrix  $\mathbf{X}$ . The logic behind this choice is that the numbers in the same column quantify the same characteristic feature. Equivalently, we can write the above equation in a more compact form,

$$\bar{x}_{*j} = \frac{x_{*j} - (\min(x_{*j}))_n}{\max(x_{*j}) - \min(x_{*j})}, \quad (j = 1, 2, \dots, m), \quad (2)$$

where  $(\min(x_{*j}))_n$  represents a vector with  $n$  entries which are all equal to  $\min(x_{*j})$ .

*2.1.2. Calculate the Correlation Coefficient Matrix  $\mathbf{R}$ .* The correlation coefficient matrix  $\mathbf{R}$  can be calculated according to the following specification:

$$\mathbf{R} = (r_{sq})_{m \times m}. \quad (3)$$

With the standard definition for correlation between two sets of data

$$r_{pq} = \frac{1}{\sigma(\bar{x}_{*s})\sigma(\bar{x}_{*q})} \sum_{k=1}^n (\bar{x}_{*s} - \text{mean}(\bar{x}_{*s})) \cdot (\bar{x}_{*q} - \text{mean}(\bar{x}_{*q})), \quad (s, q = 1, 2, \dots, m), \quad (4)$$

where  $\sigma(\bar{x}_{*s})$  and  $\text{mean}(\bar{x}_{*s})$  stand for the variance and average of the data, respectively. It is noted that  $r_{ss} = 1$  and  $r_{sq} = r_{qs}$ .

*2.1.3. Calculate the Eigenvalues and Feature Vector.* Calculate the eigenvalues  $\lambda_i (i = 1, 2, \dots, m)$  of matrix  $\mathbf{R}$  and arrange them by size, namely,  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m \geq 0$ . Then, the feature vectors  $\mathbf{u}_i$  are just the eigenvectors of the eigenvalues  $\lambda_i$ . Specifically,

$$\mathbf{R} \cdot \mathbf{u}_i = \lambda_i \cdot \mathbf{u}_i. \quad (5)$$

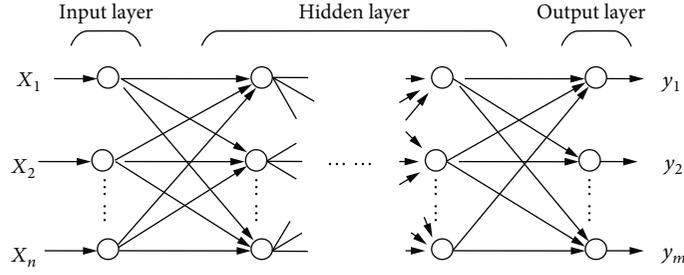


FIGURE 1: Structure of BP neural network.

In case of degeneracy, we assume that all real eigenvectors can still be found. Principal components can then be expressed in terms of

$$\begin{cases} F_1 = u_{11}\bar{x}_1 + u_{21}\bar{x}_2 + \dots + u_{m1}\bar{x}_m, \\ F_2 = u_{12}\bar{x}_1 + u_{22}\bar{x}_2 + \dots + u_{m2}\bar{x}_m, \\ \dots \\ F_m = u_{1m}\bar{x}_1 + u_{2m}\bar{x}_2 + \dots + u_{mm}\bar{x}_m, \end{cases} \quad (6)$$

where  $F_1$  denotes the first principal components,  $F_2$  the second principal components, and  $F_m$  the  $m$ th principal components.

**2.1.4. Extract  $p$  Principal Components.** First, the principal component contribution rate and cumulative contribution rate are computed. Then,  $p$  principal components based on the cumulative contribution rate are extracted. Calculate, respectively, the contribution rate  $b_j$  of the  $j$ th principal component and the cumulative contribution rate  $a_p$  by

$$b_j = \frac{\lambda_j}{\sum_{k=1}^m \lambda_k}, \quad (j = 1, 2, \dots, m), \quad (7)$$

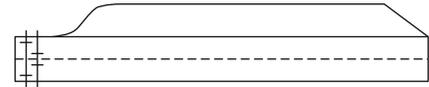
$$a_p = \frac{\sum_{k=1}^p \lambda_k}{\sum_{k=1}^m \lambda_k}. \quad (8)$$

The 1st, 2nd, ...,  $p$ th ( $p \leq m$ ) principal components are taken, when the contribution rates of the characteristic value  $a_p$  reach up to 90%. Finally, the  $p$  principal components of the original samples can be expressed as in

$$y_k = u_{1k}\bar{x}_1 + u_{2k}\bar{x}_2 + \dots + u_{mk}\bar{x}_m \quad (k = 1, 2, \dots, p). \quad (9)$$

**2.2. BP Neural Network.** The BP is a popular technique of neural networks for supervised learning. Its general structure is depicted in Figure 1.

As shown in Figure 1, there are three layers in a BP model, which are the input layer, hidden layer, and output layer. Two nodes of each adjacent layer are connected directly, and it is called a link. Each link has a weighted value representing the degree of relationship between the two nodes. Suppose there are  $n$  input neurons,  $m$  hidden neurons, and one output neuron. A training process can be divided into two steps.



Location of strain gauges

FIGURE 2: Location of blade measured loads.

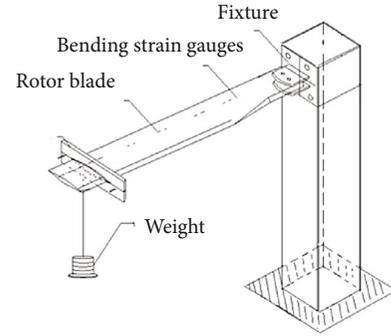


FIGURE 3: Calibration test of strain gauges.

**2.2.1. Hidden Layer Stage.** Calculate the outputs of all neurons in the hidden layer according to the following steps:

$$\text{net}_j = \sum_{i=0}^n v_{ij}x_i \quad j = 1, 2, \dots, m, \quad (10)$$

$$y_j = f_H(\text{net}_j) \quad j = 1, 2, \dots, m, \quad (11)$$

where  $v_{ij}$  is the weighted value from the  $i$ th neuron node in the input layer to the  $j$ th node in the hidden layer,  $\text{net}_j$  is the activation value of the  $j$ th node,  $y_j$  is the output of the hidden layer, and  $f_H$  is the activation function of a node. A sigmoid function is usually expressed by formula (11):

$$f_H(x) = \frac{1}{1 + \exp(-x)}. \quad (12)$$

**2.2.2. Output Stage.** The outputs of all neurons in the output layer is calculated in

$$o = f_o \left( \sum_{j=0}^m \omega_{jk}y_j \right), \quad (13)$$

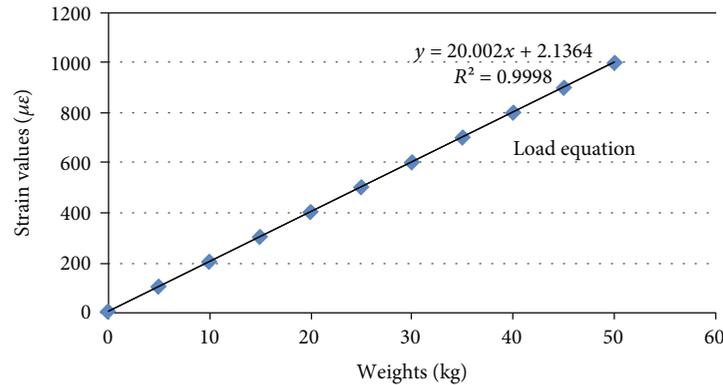


FIGURE 4: Weight against strain results with their corresponding line of best fit.

where  $\omega_{jk}$  is the weighted value from the  $j$ th neuron node in the hidden layer to the  $p$ th node in the output layer and  $f_o$  is the activation function, and it is a linear function generally. All weights are assigned with random values initially. They are modified according to the learning samples traditionally by the delta rule [19–22].

### 3. Math Experimental Data

The flight load data and flight parameter data were collected by a specially instrumented helicopter. Rotor blade load data with the root position were collected through the use of strain gauges. The location of the strain gauges in the blade is shown in Figure 2. These strain gauges are temperature compensated to consider the expected temperature changes with altitude, and there is a special coating on the strain gauge to provide mechanical and environmental protection.

The ground test was conducted to calibrate the strain gauges' output, and it is shown in Figure 3. The strain gauges were mounted on the root area of the blade which was prone to cracking, and the strain gauges' outputs were measured. Static load data has been taken from a blade supported in the frame. Applying some weights to the blade in specifically designed ways, one can measure the blade's response to the weights, i.e., strain values. To avoid the destructive limit being exceeded, the strain on the blade has constantly been measured with strain gauges at a certain frequency. In the meantime, the data has also been recorded for later utilization. The measured gauge outputs and load values were used to develop equations using linear regression to calculate blade loads from the strain gauge output. In Figure 4, the correlation coefficient of this load equation in this ground test reached up to 0.9998. This equation was later used to determine the blade loads during flight.

The blade is a rotating component. In order to collect the strain gauges' signal, the strain test equipment was designed, and it was installed at the top of the rotor hub and rotated synchronously with the rotor hub. The strain signal can be transmitted wirelessly to the recording equipment in the cabin. The strain test scheme for the rotating blade is shown in Figure 5. The airspeed, roll, pitch, altitude, acceleration, overload, and other flight parameters were measured by the flight control computer already present in the helicopter.



FIGURE 5: Strain test scheme for rotating blade.

The experimental data were synchronized in time using measured time tags from the helicopter.

The data were obtained in-flight from a series of maneuvers. Sensor readings were recorded for roll, left turn, hovering, sideslip-left, sideslip-right, push-pull, and other maneuvers. The maneuvers, the speeds, and the altitudes at which they were performed were chosen to cover the full range of helicopter motions and a considered portion of its flight envelope. 86.5 hours of data on the helicopter were collected. The percentage of the flight envelope that was covered was considered to be sufficient for establishing a strain prediction method.

The rotor angular velocity of the helicopter is 212 revolutions per minute. Therefore, the sampling rate of 256 Hz for strain data was considered enough to capture each possible associated strain data. However, flight parameters were recorded at 32 Hz. Naturally, the output data from the chosen model is paired up with the input data for each single moment describing the state of the blade and its load. Therefore, the measured data which is to be compared with the model output (or to train the model) must be correspondent to the input data for the same moment. Put in another way, all the data, whether directly measured or generated by the model must be matched for the same time points.

An example of the raw data from the helicopter is shown in Figures 6 and 7, which is the time history curve of blade loads during an asymmetric push-pull maneuver. According

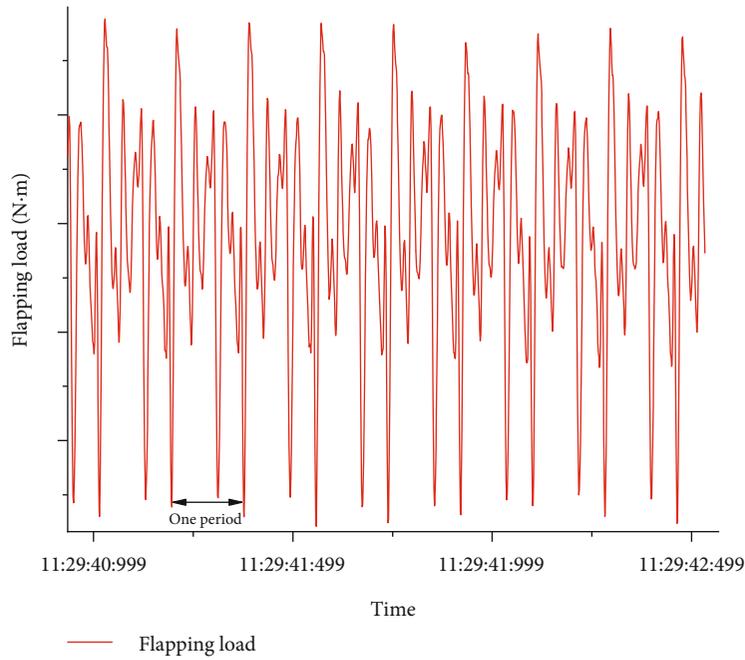


FIGURE 6: Load-time curve during a hovering maneuver.

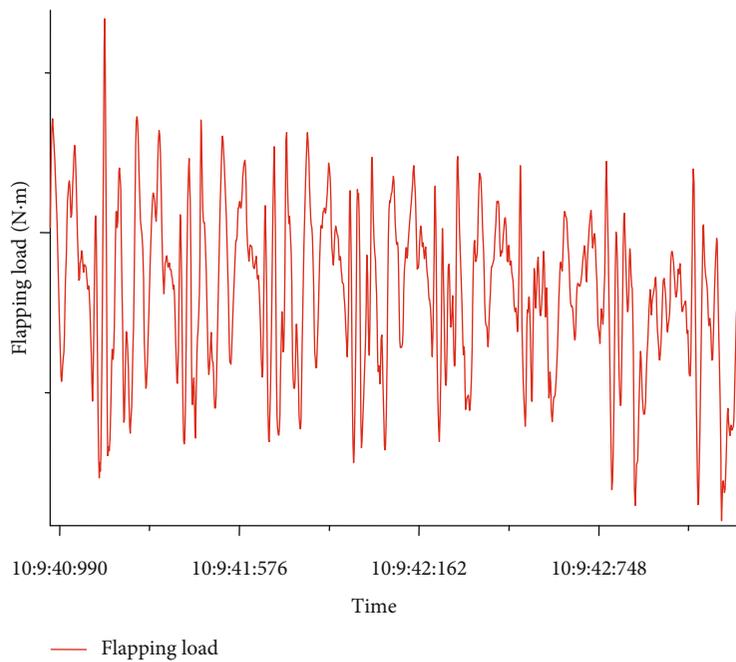


FIGURE 7: Load-time curve during a symmetric push-pull maneuver.

to the Figure 6, we can see that the waveform is smooth, periodic, and without data spikes. When a helicopter performs a steady flight maneuver, the aerodynamic load of the blade changes periodically with the azimuth of the rotor blade. Therefore, the load-time curve of the blade shows a periodic trend. The rotor angular velocity is 212 revolutions per minute, so the rotor rotates 3.53 laps in one second. We can see that blade load data in a hovering maneuver were periodical waves corresponding to the rotor shaft's rotation rate, which

is consistent with the theoretical analysis, while the load waveform during a symmetric push-pull maneuver in Figure 7 changed with the flight parameters. The time history curves of some flight parameter data were shown in Figures 7 and 8. Those waveforms were smooth and show no data spikes. Therefore, it satisfies the model's requirements for variables because data spikes, noisy data, or invalid data may lead to abnormal training of neural networks, thus hindering the generalizability of models. They are only a small

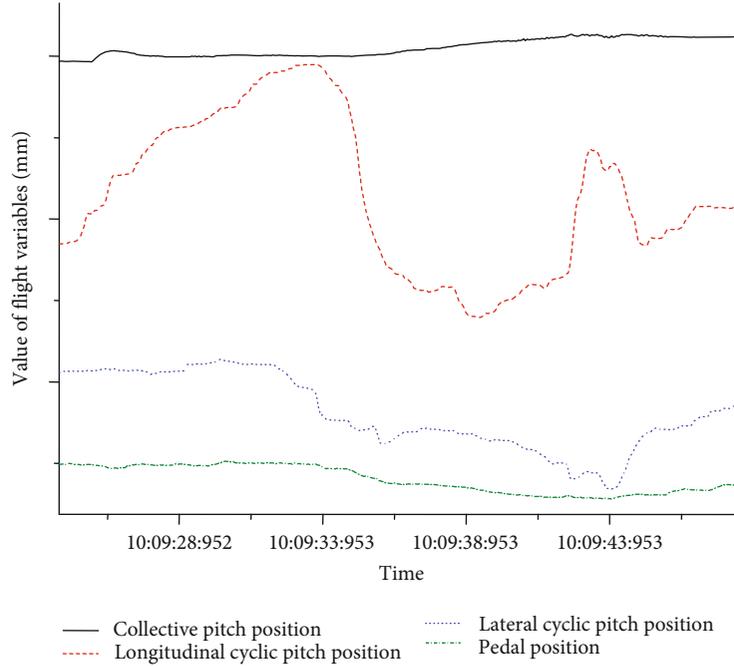


FIGURE 8: Manipulation parameter-time curves during a push-pull maneuver.

TABLE 1: Description of input variables.

Variables	Minimum	Maximum	Units	Symbol
Height	0	5500	m	$X_1$
Indicated airspeed	0	240	km/h	$X_2$
Collective pitch position	0	100	%	$X_3$
Longitudinal cyclic pitch position	-180	180	mm	$X_4$
Lateral cyclic pitch position	-180	180	mm	$X_5$
Pedal position	-180	180	mm	$X_6$
Pitch angle	-30	30	( $^\circ$ )	$X_7$
Roll angle	-45	45	( $^\circ$ )	$X_8$
Angle of yaw	0	360	( $^\circ$ )	$X_9$
Pitch rate	-30	30	$^\circ/s$	$X_{10}$
Roll rate	-30	30	$^\circ/s$	$X_{11}$
Yaw rate	-30	30	$^\circ/s$	$X_{12}$
Normal acceleration	0.5	2.5	g	$X_{13}$
First engine torque	0	100	%	$X_{14}$
Second engine torque	0	100	%	$X_{15}$
Third engine torque	0	100	%	$X_{16}$
Flapping loads	—	—	N·m	$Z$

part of flight test data of the helicopter, which were used to create the prediction load models.

86.5 hours of flight data were arbitrarily divided into training data and test data, accounting for up to about 98% and 2%, respectively. It should be noted that training data and test data were recorded on two different days to provide differing flight conditions, which can determine how general-

izable the models are and whether it can work in significantly different operating circumstances.

## 4. Establishment of Model

4.1. Selection of Input and Output Variables. The selection of the input and output variables of the model is crucial as it

TABLE 2: Correlation analysis of the process variables.

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$	$x_{13}$	$x_{14}$	$x_{15}$	$x_{16}$
$x_1$	1.00	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$x_2$	-0.46	1.00	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$x_3$	0.42	-0.40	1.00	—	—	—	—	—	—	—	—	—	—	—	—	—
$x_4$	-0.77	0.54	-0.77	1.00	—	—	—	—	—	—	—	—	—	—	—	—
$x_5$	-0.71	0.55	0.90	0.93	1.00	—	—	—	—	—	—	—	—	—	—	—
$x_6$	-0.09	-0.23	0.65	0.34	-0.49	1.00	—	—	—	—	—	—	—	—	—	—
$x_7$	-0.50	0.12	0.25	-0.13	0.08	0.69	1.00	—	—	—	—	—	—	—	—	—
$x_8$	-0.43	0.23	0.91	0.79	0.91	-0.65	-0.31	1.00	—	—	—	—	—	—	—	—
$x_9$	0.65	-0.47	0.35	-0.48	-0.54	0.02	-0.32	0.34	1.00	—	—	—	—	—	—	—
$x_{10}$	0.41	-0.85	-0.22	-0.55	-0.43	0.20	0.00	-0.34	0.37	1.00	—	—	—	—	—	—
$x_{11}$	-0.25	-0.33	-0.33	-0.66	0.13	0.34	0.49	0.21	0.00	-0.05	1.00	—	—	—	—	—
$x_{12}$	0.49	-0.53	0.49	-0.52	0.67	-0.45	-0.19	0.44	0.79	-0.50	0.01	1.00	—	—	—	—
$x_{13}$	-0.50	0.35	-0.32	0.71	0.54	-0.21	0.38	-0.29	0.60	0.55	0.32	0.89	1.00	—	—	—
$x_{14}$	-0.61	-0.57	-0.99	0.70	-0.62	-0.21	-0.34	0.45	-0.78	0.49	-0.21	-0.90	0.93	1.00	—	—
$x_{15}$	-0.59	0.42	-0.34	-0.65	0.56	0.18	-0.37	-0.42	-0.88	0.60	0.33	-0.90	0.99	0.99	1.00	—
$x_{16}$	0.53	0.59	0.37	-0.34	0.49	0.20	0.13	0.28	0.60	0.44	-0.04	-0.64	0.46	-0.58	-0.45	1.00
Z	-0.51	0.48	-0.50	-0.30	-0.38	-0.45	0.25	0.21	0.29	-0.58	0.30	-0.04	0.64	0.54	-0.53	0.53

TABLE 3: Specific value and variance of each factor and cumulative variance.

Component	Initial eigenvalues			Total	Variance %	Cumulative %
	Total	Variance %	Cumulative %			
1	8.812	55.076	55.076	8.812	55.076	55.076
2	3.147	19.668	74.744	3.147	19.668	74.744
3	1.382	8.636	83.380	1.382	8.636	83.380
4	1.188	7.427	90.807	1.188	7.427	90.807
5	0.731	4.572	95.379			
6	0.462	2.886	98.265			
7	0.118	0.734	98.999			
8	0.086	0.538	99.537			
9	0.031	0.192	99.729			
10	0.022	0.140	99.869			
11	0.012	0.073	99.942			
12	0.004	0.026	99.968			
13	0.002	0.013	99.981			
14	0.002	0.010	99.991			
15	0.001	0.007	99.998			
16	0.000	0.002	100.000			

directly affects the prediction accuracy of the rotor blade load model. For the load model of the rotor blade, the output variable is the loads of the rotor blade, and the input variables are determined by flight parameters. 16 flight parameters were chosen as inputs to the model. These 16 flight parameters were selected because they can describe the status of the helicopter during flight. In addition, they are monitored and recorded by the flight control computer on the helicopter. Therefore, a flight load prediction system using these vari-

ables is convenient to be implemented by tapping into existing data based on the helicopter. There is height, indicated airspeed, collective pitch position, longitudinal cyclic pitch position, lateral cyclic pitch position, pedal position, pitch angle, roll angle, angle of yaw, pitch rate, roll rate, yaw rate, normal acceleration, first engine torque, second engine torque, and third engine torque. These 16 variables, as well as their minimum, maximum, units, and symbols are shown in Table 1.

**4.2. Principal Component Analysis for the Influence Factor.** Correlation analysis was performed with SPSS statistical software between the 16 factors and the blade loads. The result is shown in Table 2. Only data in the lower triangle are shown, as the matrix is symmetric. The absolute value of the correlation coefficient between a given factor and the blade loads is a good measure of the latter's dependence on the former. From Table 2, it can be seen that the coefficients have the following order:  $|X_{13}| > |X_{10}| > |X_{14}| > |X_{16}| > |X_{15}| > |X_1| > |X_3| > |X_2| > |X_6| > |X_5| > |X_{11}| > |X_4| > |X_9| > |X_7| > |X_8| > |X_{12}|$ . This indicates that the blade loads have a strong dependence on the normal acceleration and pitch rate, while the reliance on the yaw rate is negligible.

Data in Table 2 illustrates that the 16 factors of the blade load are strongly correlated. Directly inputting the 16 factors into a neural network is possible but may not produce the optimal predictive power. Because of the large volume of data as well as the correlation among them, the structure of the neural network will be complicated, the training intensity of the neural network will be high, and it is easy to fall into local minimum points, which will eventually lead to low generalizability. Therefore, the PCA was employed to extract the most relevant and independent factors influencing the loads before feeding data into a BP neural network (this turned out to be of great value). As a result, fewer but more effective factors were obtained, and the size (thus, the computation load, too) of our neural network was significantly reduced.

The result of PCA for the 16 blade-load factors is presented in Table 3, where one finds the specific values and variances of each factor and the cumulative variance. The criterions for the factors to be kept are (1) they form the most economic combination and (2) they predict more than 90 percent of the total variance of independent observations. According to these criteria, four principal components were eventually selected and specified in formulas (14)–(17). The corresponding percentage data are shown in Table 4. The four principal components bear a cumulative contribution of 90.807 percent. They are sufficient to cover the characteristics of the original variables. Further, the correlation coefficients of the four selected principal components are vanishing. Thus, the purpose of eliminating the correlation between the original variables was achieved by PCA.

$$\begin{aligned}
 F_1 = & 0.3259\bar{X}_1 + 0.2670\bar{X}_2 + 0.2299\bar{X}_3 + 0.2411\bar{X}_4 \\
 & + 0.2925\bar{X}_5 + 0.3018\bar{X}_6 + 0.1273\bar{X}_7 + 0.0516\bar{X}_8 \\
 & + 0.2505\bar{X}_9 + 0.2545\bar{X}_{10} + 0.2114\bar{X}_{11} + 0.0156\bar{X}_{12} \\
 & + 0.2953\bar{X}_{13} + 0.2848\bar{X}_{14} + 0.2992\bar{X}_{15} + 0.2927\bar{X}_{16},
 \end{aligned} \tag{14}$$

$$\begin{aligned}
 F_2 = & 0.0597\bar{X}_1 + 0.0973\bar{X}_2 + 0.0225\bar{X}_3 + 0.1911\bar{X}_4 \\
 & + 0.0253\bar{X}_5 + 0.1107\bar{X}_6 + 0.2394\bar{X}_7 + 0.2616\bar{X}_8 \\
 & + 0.1911\bar{X}_9 + 0.0904\bar{X}_{10} + 0.0397\bar{X}_{11} + 0.2005\bar{X}_{12} \\
 & + 0.0491\bar{X}_{13} + 0.1261\bar{X}_{14} + 0.1027\bar{X}_{15} + 0.1187\bar{X}_{16},
 \end{aligned} \tag{15}$$

TABLE 4: Percentage accounted in each factor of the 4-factor model.

	Component			
	1	2	3	4
$X_1$	0.110	-0.060	0.068	0.065
$X_2$	-0.090	-0.097	-0.319	-0.169
$X_3$	0.077	-0.023	0.287	-0.416
$X_4$	-0.081	0.191	-0.095	-0.173
$X_5$	0.099	-0.025	0.136	0.328
$X_6$	0.102	-0.111	0.149	0.132
$X_7$	-0.043	0.239	0.344	0.089
$X_8$	0.017	0.262	0.233	0.146
$X_9$	0.084	-0.191	0.032	0.222
$X_{10}$	-0.086	-0.090	-0.162	0.426
$X_{11}$	-0.071	-0.040	0.279	-0.002
$X_{12}$	0.005	0.201	-0.185	0.376
$X_{13}$	-0.100	-0.049	0.124	0.272
$X_{14}$	0.096	0.126	-0.242	-0.023
$X_{15}$	0.101	0.103	-0.222	-0.031
$X_{16}$	0.099	0.119	-0.222	-0.006

$$\begin{aligned}
 F_3 = & 0.0530\bar{X}_1 + 0.2486\bar{X}_2 + 0.2242\bar{X}_3 + 0.0743\bar{X}_4 \\
 & + 0.1063\bar{X}_5 + 0.1166\bar{X}_6 + 0.2682\bar{X}_7 + 0.1816\bar{X}_8 \\
 & + 0.0253\bar{X}_9 + 0.1263\bar{X}_{10} + 0.2176\bar{X}_{11} + 0.1445\bar{X}_{12} \\
 & + 0.0967\bar{X}_{13} + 0.1890\bar{X}_{14} + 0.1728\bar{X}_{15} + 0.1732\bar{X}_{16},
 \end{aligned} \tag{16}$$

$$\begin{aligned}
 F_4 = & 0.0707\bar{X}_1 + 0.1848\bar{X}_2 + 0.4543\bar{X}_3 + 0.1892\bar{X}_4 \\
 & + 0.3580\bar{X}_5 + 0.1441\bar{X}_6 + 0.0976\bar{X}_7 + 0.1599\bar{X}_8 \\
 & + 0.2428\bar{X}_9 + 0.4648\bar{X}_{10} + 0.0025\bar{X}_{11} + 0.4106\bar{X}_{12} \\
 & + 0.2971\bar{X}_{13} + 0.0254\bar{X}_{14} + 0.0342\bar{X}_{15} + 0.0064\bar{X}_{16}.
 \end{aligned} \tag{17}$$

**4.3. Establishment of the MLR Model.** The MLR is a multivariate statistical technique that has been used early in model evaluation [23, 24]. In this study, a MLR model was first established to predict the blade loads based on flight data. Flight data was normalized with formulas (1) and (2). The optimization algorithm was the Levenberg-Marquardt (LM) method. The MLR model was obtained by regression analysis of blade loads and its influencing factors (16 flight parameters). It was done by SPSS software.

**4.4. Establishment of the BP Neural Network Model.** The MLR model can predict blade loads, and the BP neural network can also be designed to do the same. To achieve optimal performance, the weight and bias of the neurons are iterated until the error in the test data calculated from equation (18) reaches a predetermined level. In the

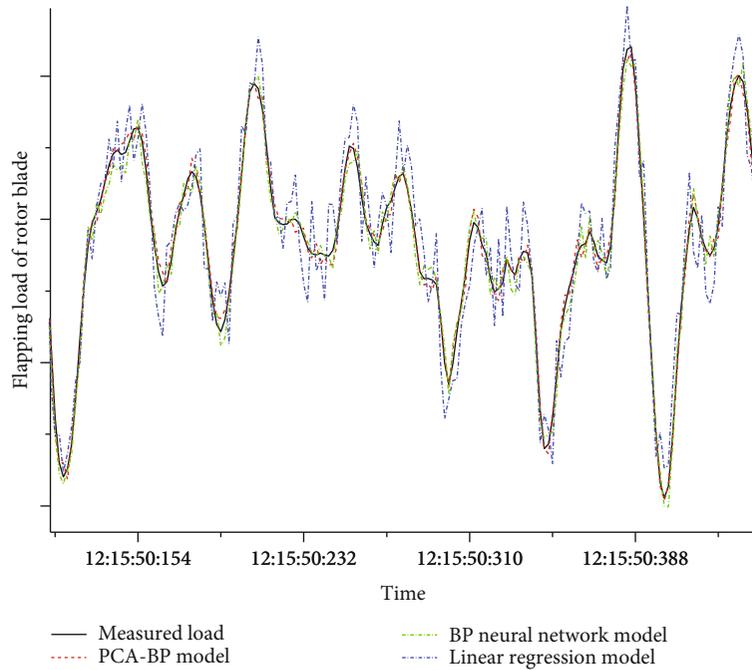


FIGURE 9: Comparison of three models during hovering maneuver.

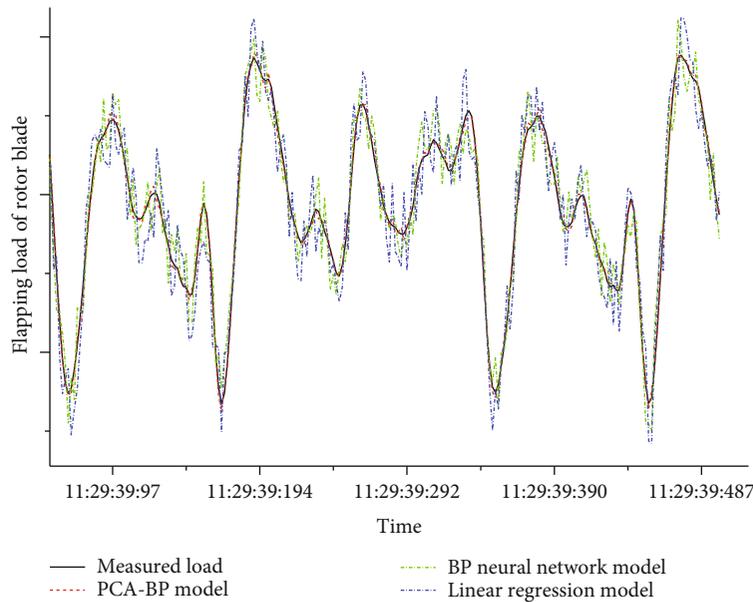


FIGURE 10: Comparison of three models during flight forward maneuver.

meantime, the size of the network is also analyzed following parametric studies.

$$\text{percentage error} = \frac{|\text{measured load} - \text{estimated load}|}{\text{measured load}} \times 100\% \tag{18}$$

It is shown in parametric studies that utilizing single hidden-layer networks is a good tradeoff between the complexity and performance of a network, mainly because

increasing the number of layers and expanding the layer size have a limited effect on the performance of the network. For the hidden layer and the output layer activation functions, we have assigned a sigmoid function and a linear function, respectively. This choice is made based on the analysis of suitable networks for data with internal coherence [25]. To train the network efficiently, we have adopted the LM algorithm, which is shown to converge faster than other algorithms [26].

Since we have not adopted PCA to preprocess the data, 16 input variables are corresponding to 16 neurons,

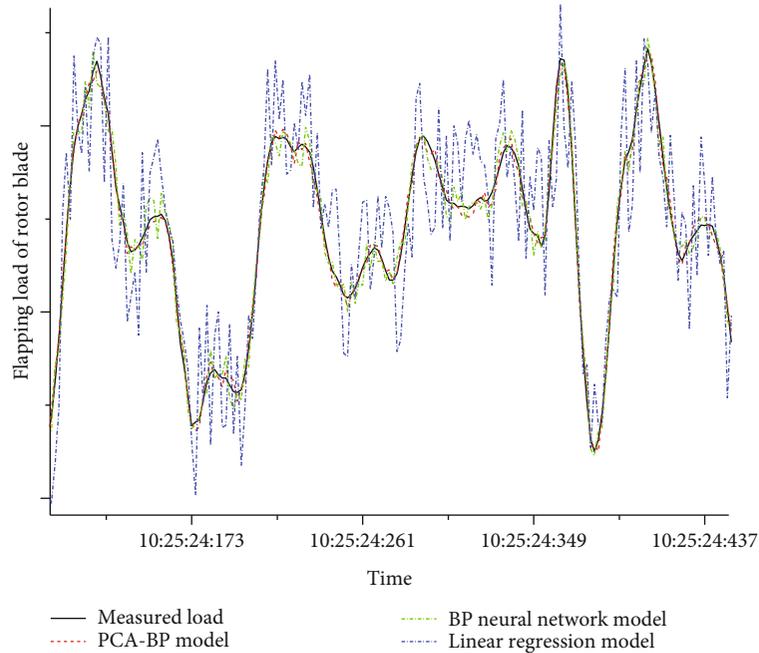


FIGURE 11: Comparison of three models during horizontal turn maneuvers.

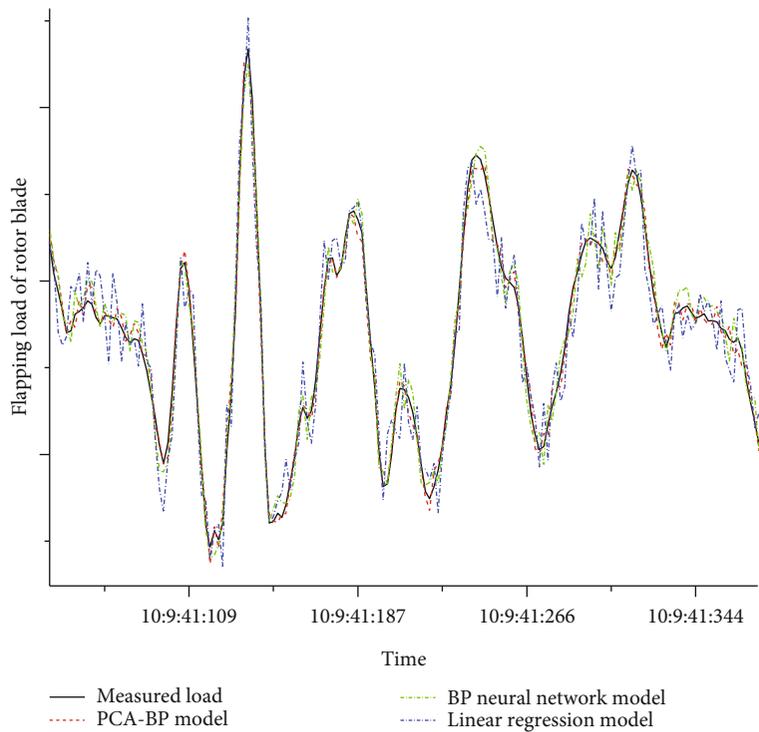


FIGURE 12: Comparison of three models during symmetric pull-rush maneuver.

and one output variable is corresponding to one neuron. The training data and test data are the same as those used in the model of MLR, and they were normalized before being fed into the network. The error goal of the BP model was set to 0.0001, and the learning efficiency was set to 0.01.

The number of neurons  $M$  in the hidden layer has a crucial influence on the performance of the model. Commonly, it can reduce the training error of the network as the increase of hidden layer neurons. However, too many hidden layer neurons not only can increase the complexity and training time of the network but also may lead to overfitting.

TABLE 5: Comparison of three kinds of RMSE models from four maneuvers for blade loads.

Maneuver	MLR error	BP error	PCA-BP error	PCA-BP to MLR reduced error	PCA-BP to BP reduced error
Hovering	9.76%	4.53%	3.15%	6.61%	1.38%
Flight forward	8.35%	4.39%	1.56%	6.79%	2.83%
Left and right turns	12.14%	5.22%	2.80%	9.34%	2.42%
Symmetric push-pull	10.53%	3.83%	2.31%	8.22%	1.52%
Average	10.20%	4.49%	2.46%	7.74%	2.03%

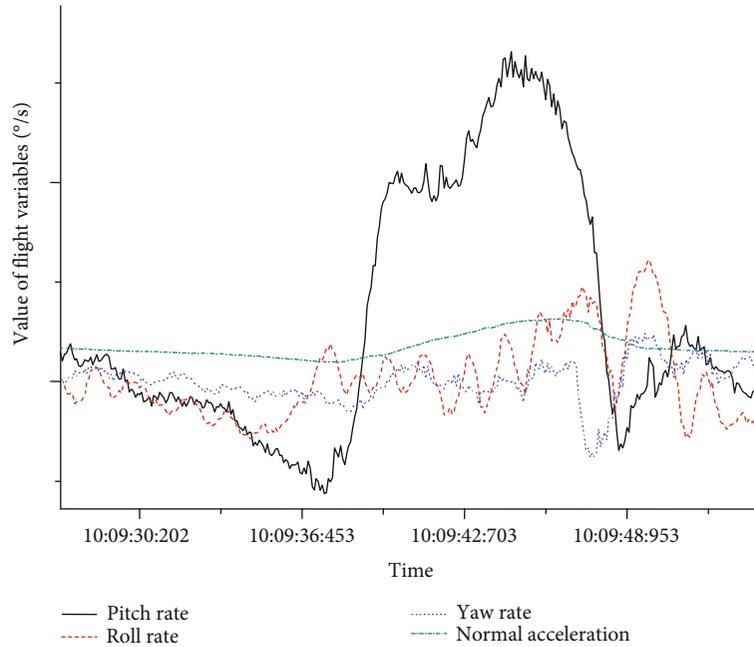


FIGURE 13: Attitude parameter-time curves during a push-pull maneuver.

In this article,  $M$  was obtained from the following empirical formula:

$$M = \sqrt{A + B} + C, \quad (19)$$

where  $A$  is the number of input layer neurons,  $B$  is the number of output layer neurons, and  $C$  is a constant from 1 to 10 [27]. According to the empirical formula,  $M$  is an integer. If the square root of  $A + B$  is not an integer, the number will be rounded to the nearest integer. In our paper,  $A$  is 16 and  $B$  is 1, so we round the square root of 17 to 4. By using the trial-and-error method, the network model of an increasing number of hidden layer neuron from 5 to 14 was trained. As a result, it was found that the output error was the smallest when  $M$  was 10. So this study used 5 hidden layer neurons.

In this way, the network was eventually designed with the architecture of  $16 \times 10 \times 1$ . In this study, to reach the accuracy level 0.0001, we found that 40342 epochs are needed for this particular model.

**4.5. Establishment of PCA-BP Neural Network Model.** As in the cases of the MLR and the BP neural network models, this study used the same data for training and testing the com-

binated PCA-BP neural network model. The difference between the combined model and the pure BP model is distinct: now there are only 4 input variables corresponding to the four principal components, instead of 16 in the pure BP model. Similar with the BP model, the PCA-BP model's error goal was set to 0.0001 and the learning efficiency was set to 0.01, and the LM optimization algorithm was used in the process of training the network. Using the trial-and-error method, networks of different numbers of hidden-layer neurons were trained, and it was found that the output error was the smallest when  $M$  was 7. Therefore, the selected PCA-BP model has the neural network architecture of  $4 \times 7 \times 1$  in the PCA-BP model. Once the optimal neural network structure of the PCA-BP model was determined, the weights and thresholds of the network were saved. In this study, to reach the accuracy level 0.0001, we found that 13862 epochs were needed for this particular model.

## 5. Comparison of Models

The test data, accounting up to 2% of 86.5 hours of flight data, were used to determine the accuracy and generalizability of models to predict the blade loads. The performance can

be determined by comparing the percentage error from equation (14). Figures 9–12 show the comparison of the linear model prediction, BP neural network model prediction, and PCA-BP model prediction to the measured load during hovering, flight forward, horizontal turns, and symmetric push-pull maneuver. Table 5 shows the average percentage of errors from the three models and the error percentage reduced by the PCA-BP model compared to the MLR and BP, for loads in four maneuvers.

According to Figures 9–11 and 13, it can be seen that the predicted loads of the MLR model were not very consistent with the measured loads, and the evaluation errors are relatively large, which mainly occurred in breakpoints. This is because there are complex nonlinear relationships between the loads and other influencing factors, while the MLR model simplifies it into a linear relationship. Therefore, the MLR model can only provide a limited guidance for actual flight, and it is necessary to develop other more accurate prediction methods.

The BP neural network model has a better performance in the prediction accuracy than the MLR model. Highly oscillatory behavior can be seen in the linear regression model while not in the BP model. However, at several points, the errors in the prediction were a bit large. As pointed out in the earlier text, the 16 factors are correlated, and this fact causes a discount in the performance of the neural network. Intuitively, we can say that correlated input variables are not clean compared to mutually independent input variables.

It is clear from Table 5 that the combined use of PCA and BP has higher precision in prediction with an average 2.46% percentage error compared to that of MLR (10.20% error) and BP neural network (4.49% error). In addition to that, the PCA-BP model has consistently lower errors when compared to the MLR model and the BP model. This summary of the model performance shows the capacity of the PCA-BP model to predict loads on a rotor blade in different flight maneuvers.

## 6. Conclusion

A model combining the PCA and BP neural network has been built and tested for predicting blade loads. In the PCA stage of the model, four independent factors were extracted as principal components from the sixteen original factors influencing the blade loads. Then, a BP neural network model was built with the 4 principal components as input variables. The ultimately decided architecture of the neural network is  $4 \times 7 \times 1$  with 7 the number of neurons in the hidden layer of the network.

To assess the quality of prediction of this combined model, we compared the results with an MLR model and a BP neural network model with the same data. The following are the main conclusions of this study.

- (1) A MLR model and a BP neural network model for prediction of blade loads were established. The results show that the error percentage in the prediction is relatively high, and some data points in the

two models are large. Therefore, they can only provide limited guidance for actual flight

- (2) The PCA-BP model successfully demonstrates that it is possible to predict the blade load with high accuracy from the flight parameter data collected in different maneuvers. It is beneficial to predicting the blade loads of helicopter in-service where installation of strain gauges is impractical
- (3) The PCA not only simplifies the structure of the BP network but also improves the network estimation accuracy and convergence speed. The results show that an average error percentage of 2.46% was achieved by the PCA-BP model, compared to 10.20% by MLR and 4.49% by a pure BP neural network model. In order to reach the accuracy of 0.01, the PCA-BP model requires much fewer epochs than the pure BP model (in the specific cases described in this study, it is two-thirds less).

## Data Availability

The flight data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

We declare that we have no conflicts of interest.

## Acknowledgments

This work was supported by the General Armament Department of China for the development of a type of navy helicopter (Grant No. [2013] 416).

## References

- [1] M. Zhitao and Z. Benyin, *Helicopter Structure Fatigue*, National Defence Industry Press, Beijing, CHN, 2009.
- [2] D. Kim and M. Marciniak, "Prediction of vertical tail maneuver loads using backpropagation neural networks," *Journal of Aircraft*, vol. 37, no. 3, pp. 526–530, 2000.
- [3] M. Chierichetti, C. McColl, and M. Ruzzene, "Prediction of UH-60A blade loads: insight on load confluence algorithm," *AIAA Journal*, vol. 52, no. 9, pp. 2007–2018, 2014.
- [4] J. Gómez-Escalonilla, J. García, M. M. Andrés, and J. I. Armijo, "Strain predictions using artificial neural networks for a full-scale fatigue monitoring system," in *Thirteenth Australian International Aerospace congress*, Springer, 2015.
- [5] D. Kim and M. Marciniak, "Horizontal tail maneuver load predictions using backpropagation neural networks," *Journal of Aircraft*, vol. 39, no. 2, pp. 365–370, 2002.
- [6] Z. Yan, S. Gao, and Z. Duoyuan, "An improved genetic algorithm for flight load measurements," *Mechanical Science and Technology for Aerospace Engineering*, vol. 31, no. 8, pp. 243–247, 2012.
- [7] S. B. Cooper and D. DiMaio, "Static load estimation using artificial neural network: application on a wing rib," *Advances in Engineering Software*, vol. 125, pp. 113–125, 2018.

- [8] F. G. Polanco, "Estimation of structural component loads in helicopters: a review of current methodologies," DSTO Aeronautical and Maritime Research Laboratory, Australia, 1999, D.o.D.S.T. Organisation, Editor.
- [9] R. H. Cabell, C. R. Fuller, and W. F. O'Brien, "Neural network modelling of oscillatory loads and fatigue damage estimation of helicopter components," *Journal of Sound and Vibration*, vol. 209, no. 2, pp. 329–342, 1998.
- [10] D. J. Haas, L. Flitter, and J. Milano, "Helicopter flight data feature extraction or component load monitoring," *Journal of Aircraft*, vol. 33, no. 1, pp. 37–45, 1996.
- [11] M. J. Allen and R. P. Dibley, "Modeling aircraft wing loads from flight data using neural networks," *SAE Transactions*, vol. 112, no. 1, p. 521, 2003.
- [12] A. A. Elshafey, M. R. Haddara, and H. Marzouk, "Damage detection in offshore structures using neural networks," *Marine Structures*, vol. 23, no. 1, pp. 131–145, 2010.
- [13] N. C. G. De Paula, F. D. Marques, and W. A. Silva, "Volterra kernels assessment via time-delay neural networks for nonlinear unsteady aerodynamic loading identification," *AIAA Journal*, vol. 57, no. 4, pp. 1725–1735, 2019.
- [14] F. He and L. Zhang, "Prediction model of end-point phosphorus content in BOF steelmaking process based on PCA and BP neural network," *Journal of Process Control*, vol. 66, pp. 51–58, 2018.
- [15] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer Science+ Business Media, London, UK, 2006.
- [16] Y. Dong, X. Wei, L. Tian, F. Liu, and G. Xu, "A novel double cluster and principal component analysis-based optimization method for the orbit design of earth observation satellites," *International Journal of Aerospace Engineering*, vol. 2017, 15 pages, 2017.
- [17] L. Sivanandam, S. Periyasamy, and U. M. Oorkavalan, "Power transition X filling based selective Huffman encoding technique for test-data compression and scan power reduction for SOCs," *Microprocessors and Microsystems*, vol. 72, article 102937, 2020.
- [18] S. H. Berguin and D. N. Mavris, "Dimensionality reduction using principal component analysis applied to the gradient," *AIAA Journal*, vol. 53, no. 4, pp. 1078–1090, 2015.
- [19] K. Krzykowska and M. Krzykowski, "Forecasting parameters of satellite navigation signal through artificial neural networks for the purpose of civil aviation," *International Journal of Aerospace Engineering*, vol. 2019, 11 pages, 2019.
- [20] Y. Zhang and L. Wu, "Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network," *Expert Systems with Applications*, vol. 36, no. 5, pp. 8849–8854, 2009.
- [21] Z. Li, C. Chen, H. Pei, and B. Kong, "Structural optimization of the aircraft NACA inlet based on BP neural networks and genetic algorithms," *International Journal of Aerospace Engineering*, vol. 2020, 9 pages, 2020.
- [22] A. S. Tenney, M. N. Glauser, C. J. Ruscher, and Z. P. Berger, "Application of artificial neural networks to stochastic estimation and jet noise modeling," *AIAA Journal*, vol. 58, no. 2, pp. 647–658, 2020.
- [23] F. P. tei and Z. Guan, "Prediction on geometrical characteristics of cermet laser cladding based on linear regression and neural network," *Surface Technology*, vol. 48, no. 12, pp. 353–354, 2019.
- [24] F. Kaitai, Q. Hui, and C. Qingyun, *The Analysis of Regression*, Science Press, Beijing, CHN, 1988.
- [25] M. N. S. Hadi, "Neural networks applications in concrete structures," *Computers & Structures*, vol. 81, no. 6, pp. 373–381, 2003.
- [26] A. R. Rao and B. Kumar, "Neural modeling of square surface aerators," *Journal of Environmental Engineering*, vol. 133, no. 4, pp. 411–418, 2007.
- [27] C. O. Hulse, S. M. Copley, and J. A. Pask, "Effect of crystal orientation on plastic deformation of magnesium oxide," *Journal of the American Ceramic Society*, vol. 46, no. 7, pp. 317–323, 1963.