Tsinghua Science and Technology

Volume 20 | Issue 2 Article 9

2015

Design of Fault Detection Observer Based on Hyper Basis Function

Xin Wen
both Faculty of Aerospace Engineering, Shenyang Aerospace University, Shenyang 110136, China and College Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

Xingwang Zhang
Faculty of Aerospace Engineering, Shenyang Aerospace University, Shenyang 110136, China.

Yaping Zhu
College Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

Recommended Citation
Xin Wen, Xingwang Zhang, Yaping Zhu. Design of Fault Detection Observer Based on Hyper Basis Function. Tsinghua Science and Technology 2015, 20(2): 200-204.

TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 09/11 pp200-204
Volume 20, Number 2, April 2015

Design of Fault Detection Observer Based on Hyper Basis Function

Xin Wen*, Xingwang Zhang, and Yaping Zhu

Abstract: In this paper, we propose the Hyper Basis Function (HBF) neural network on the basis of Radial Basis Function (RBF) neural network. Compared with RBF, HBF neural networks have a more generalized ability with different activation functions. A decision tree algorithm is used to determine the network center. Subsequently, we design an adaptive observer based on HBF neural networks and propose a fault detection and diagnosis method based on the observer for the nonlinear modeling ability of the neural network. Finally, we apply this method to nonlinear systems. The sensitivity and stability of the observer for the failure of the nonlinear systems are proved by simulation, which is beneficial for real-time online fault detection and diagnosis.

Key words: observer; fault detection; hyper basis function; neural networks

1 Introduction

While fault detection and diagnosis for dynamic systems is a research hotspot, fault detection for nonlinear systems is the problem area in the control field[1, 2]. In recent years, the state observer for nonlinear systems has been widely used in process monitoring, fault detection, and fault diagnosis. Gao et al.[3] used the known effects of redundancy on kinetics to produce residuals, constructing a fault diagnosis observer for an underwater vehicle. Zarei and Shokri[4] designed a nonlinear unknown input observer, which decoupled disturbances and uncertainties from estimated states, and used the cubature rule to overcome nonlinear calculations in the presence of external disturbances for sensor fault detection. Wang et al.[5] proposed an observer with unknown inputs and used it to estimate the state and detect faults in satellite attitude control systems. However, the observer design for complex nonlinear dynamic systems suffers from large computation and tedious variable decoupling problems. Because of their self-learning, adaptive capabilities, and other features, neural networks are paid more attention to by scholars in various fields. In recent years, the observer design based on neural networks has developed at a rapid rate. Song et al.[6] proposed a fault diagnosis and detection method based on the Radial Basis Function (RBF) neural network observer to detect faults in the actuator in an aircraft control system. Du et al.[7] presented a dual neural networks combined strategy to detect faults in sensors. Vanini et al.[8] utilized dynamic neural networks to obtain residuals and reliable criteria, thereby accomplishing fault detection and isolation for an aircraft jet engine. However, the slow learning rate in the neural network training process limits its real-time online applications. Consequently, the study of observer design for nonlinear systems based on neural networks needs to be improved and optimized further[9–11].

The Hyper Basis Function (HBF) neural network, which has a stronger generalization ability, is proposed on the basis of RBF neural networks. This network uses a Mahalanobis-like distance to calculate the distance

• Xin Wen is with both Faculty of Aerospace Engineering, Shenyang Aerospace University, Shenyang 110136, China and College Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China. E-mail: wen_xin2004@126.com.
• Xingwang Zhang is with Faculty of Aerospace Engineering, Shenyang Aerospace University, Shenyang 110136, China. E-mail: aaron_zxw@163.com.
• Yaping Zhu is with College Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China.
∗ To whom correspondence should be addressed.
Manuscript received: 2014-09-30; revised: 2014-11-17; accepted: 2015-01-05

between neurons. A new matrix is introduced to present the similarity between neurons, which enables the network to achieve higher accuracy approximation for complex nonlinear functions. This paper introduces HBF neural network theory in Section 1. Section 2 proposes a design method of an adaptive state observer based on HBF neural networks, and Section 3 gives a fault diagnosis method based on the observer for the nonlinear system. Finally, the effectiveness of the fault diagnosis method is proved using an example by simulation.

2 HBF Neural Networks

HBF networks are similar to the generalized RBF networks. The output function of the networks is as follows.

$$y_i = f(\boldsymbol{x}_i) = \sum_{j=1}^{J} w_{ij} h_j(\boldsymbol{x}_i, \boldsymbol{c}_j, \sigma_j) + b_i \qquad (1)$$

where $\boldsymbol{x}_i \in \mathbf{R}^{n_x}$ is the i-th input vector of the networks (n_x stands for the number of input dimensions), w_{ij} is the connecting weight of the j-th basis function, $h_j(*)$ represents the basis function of the j-th neuron, $\boldsymbol{c}_j$ is the center of the j-th basis function, $\sigma_j \in \mathbf{R}^1$ stands for the similarity degree between $\boldsymbol{x}_i$ and $\boldsymbol{c}_j$, and b_i is a constant.

Let $h_0 = 1, b_i = w_{i0}$, then Eq. (1) can be simplified as follows.

$$y_i = f(\boldsymbol{x}_i) = \sum_{j=0}^{J} w_{ij} h_j(\boldsymbol{x}_i, \boldsymbol{c}_j, \sigma_j) \qquad (2)$$

Using Eq. (2), we see that the radial function of HBF has weighting coefficients, which makes the networks have the ability of interpolation decision.

2.1 Hyper basis function

For measuring the proximity between the input neurons and the central neurons, we use a Mahalanobis-like distance instead of a Euclidean-like distance[12–14]. The Mahalanobis-like distance in hyper basis function is given in the following form,

$$h_j(\boldsymbol{x}_i, \boldsymbol{c}_j, \boldsymbol{\Sigma}_j) = \mathrm{e}^{-0.5(x_i - c_j)^{\mathrm{T}} \Sigma_j (x_i - c_j)} \qquad (3)$$

where $\boldsymbol{\Sigma}_j$ is a positive definite square matrix. It represents the similarity between the i-th input vector $\boldsymbol{x}_i$ and the j-th center vector $\boldsymbol{c}_j$ invariant to scaling and local orientation of the data. Here we choose $\boldsymbol{\Sigma}_j = \mathrm{diag}(1/\sigma_1^2,\ 1/\sigma_2^2,\ \cdots,\ 1/\sigma_{n_x}^2)$, $j = 1, 2, \cdots, J$. It indicates that every neuron has an elliptical shape with a varying size, but with a restricted orientation that is aligned with the original input coordinates.

This form takes into account scaling of dimensions of the data and provides better flexibility and more optimization parameters. At the same time, the degree of freedom model cannot lead to under-fitting or serious over-fitting.

2.2 Algorithm

There are many algorithms for HBF neural networks, such as unsupervised competitive learning, LVQ learning, K-means clustering, and decision trees[9]. In this paper, decision trees are used to determine the network center.

The essence of the decision trees (or hierarchical trees) classification is to choose the attribute characteristics that can produce the maximum information gain in the learning process to divide the feature space. These attributes and their corresponding values constitute a variety of decision boundaries; these boundaries use feature values to divide the space into mutually exclusive decision regions. Subsequently, all kinds of decision regions can be integrated through conjunction and disjunction[15].

(1) Calculating the centers and widths

Calculate the centers $\boldsymbol{c}_j = (c_{1j}, \cdots, c_{n_x j})$ as follows.

$$c_{ij} = (\min(x_{ij}) + \max(x_{ij}))/2, i = 1, \cdots, n_x \qquad (4)$$

Calculate the kernel widths as follows.

$$\sigma_{ij} = (\max(x_{ij}) - \min(x_{ij}))/2, i = 1, \cdots, n_x \qquad (5)$$

(2) Calculating the weighting values

Assume that there are K neurons in the hidden layer, $\boldsymbol{x}^{\mu}$ and $\boldsymbol{y}^{\mu}$, $\mu = 1, \cdots, M$ are the feature vector and the target vector of the training sample, respectively. The error function of the networks is as follows.

$$E(\boldsymbol{W}) = \|\boldsymbol{HW} - \boldsymbol{Y}\|^2 \qquad (6)$$

where $\boldsymbol{W}$ is the matrix of output layer weights, $\boldsymbol{H} = (H_{\mu j}) = (h_j(\boldsymbol{x}^{\mu}, \boldsymbol{c}_j, \sigma_j))$, $H_{\mu j}$ stands for the output of the j-th basis function with the μ-th input vector. $\boldsymbol{Y} = (Y_{\mu j})$, $Y_{\mu j}$ is the j-th component of the μ-th target vector $\boldsymbol{y}^{\mu}$.

Solving for the output weight matrix, we get the following:

$$\boldsymbol{W} = \boldsymbol{H}^{+}\boldsymbol{Y} \qquad (7)$$

where $\boldsymbol{H}^{+}$ is the pseudo inverse matrix of $\boldsymbol{H}$, which can be obtained by Singular Value Decomposition (SVD).

3 Results and Discussion

Given the nonlinear system[16]

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}), \\ y(t) = \boldsymbol{C}\boldsymbol{x}(t) \end{cases} \tag{8}$$

where $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u})$ is the vector of the nonlinear function, $\boldsymbol{C} \in \mathbf{R}^{m \times n}$ is a constant matrix, and $\boldsymbol{A} \in \mathbf{R}^{n \times n}$, $(\boldsymbol{A}, \boldsymbol{C})$ needs to be observed.

For a nonlinear system, the observer based on HBF neural networks can be constructed as shown in Fig. 1.

As shown in Fig. 1, the state observer can be described as follows.

$$\begin{cases} \dot{\hat{\boldsymbol{x}}}(t) = \boldsymbol{A}\hat{\boldsymbol{x}} + \hat{\boldsymbol{g}}(\hat{\boldsymbol{x}}, \boldsymbol{u}) + \boldsymbol{L}(\boldsymbol{y} - \hat{\boldsymbol{y}}), \\ \hat{\boldsymbol{y}}(t) = \boldsymbol{C}\hat{\boldsymbol{x}}(t) \end{cases} \tag{9}$$

where $\boldsymbol{L}$ is the observer gain. This makes $(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})$ be the asymptotically stable Hurwitz matrix.

We define the state error $\boldsymbol{e}(t)$ and residual $\boldsymbol{e}_y(t)$ as follows.

$$\begin{cases} \boldsymbol{e}(t) = \boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t), \\ \boldsymbol{e}_y(t) = \boldsymbol{y} - \hat{\boldsymbol{y}} = \boldsymbol{C}\boldsymbol{e}(t) \end{cases} \tag{10}$$

From Eqs. (9) and (10), we obtain the following.

$$\dot{\boldsymbol{e}}(t) = \dot{\boldsymbol{x}}(t) - \dot{\hat{\boldsymbol{x}}}(t) = (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C})\boldsymbol{e}(t) + \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) - \hat{\boldsymbol{g}}(\hat{\boldsymbol{x}}, \boldsymbol{u}) \tag{11}$$

According to the approaching performance of neural networks, in the case of the given approximation error $\boldsymbol{\varepsilon}(\boldsymbol{x}) > 0$, the nonlinear function $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u})$ can be expressed as

$$\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{W}^{\mathrm{T}} f(\boldsymbol{x}, \boldsymbol{u}) + \boldsymbol{\varepsilon}(\boldsymbol{x}) \tag{12}$$

where $\|\boldsymbol{W}\|_{\mathrm{F}} \leqslant W_M$, which ensures $\boldsymbol{W}$ is bounded.

Subsequently, according to the networks estimation, we obtain the following.

$$\hat{\boldsymbol{g}}(\boldsymbol{x}, \boldsymbol{u}) = \hat{\boldsymbol{W}}^{\mathrm{T}} f(\hat{\boldsymbol{x}}, \boldsymbol{u}) \tag{13}$$

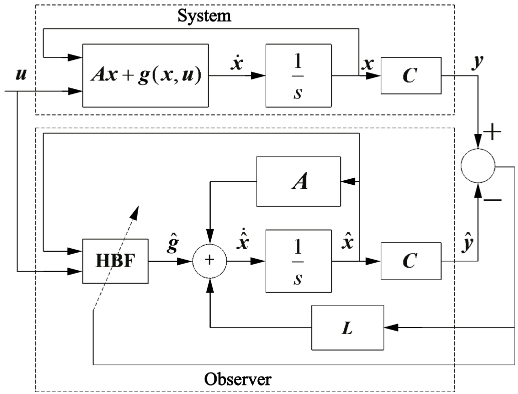

Fig. 1 Observer model for nonlinear systems based on neural networks.

Substituting Eqs. (12) and (13) into Eq. (11) gives the following.

$$\dot{\boldsymbol{e}}(t) = A_C \boldsymbol{e}(t) + \boldsymbol{e}_W^{\mathrm{T}} f(\hat{\boldsymbol{x}}, \boldsymbol{u}) + \boldsymbol{W}^{\mathrm{T}}[f(\boldsymbol{x}, \boldsymbol{u}) - f(\hat{\boldsymbol{x}}, \boldsymbol{u})] + \boldsymbol{\varepsilon}(\boldsymbol{x}) \tag{14}$$

where $\boldsymbol{e}_W = \boldsymbol{W} - \hat{\boldsymbol{W}}$ and $\boldsymbol{A}_C = \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$.

4 Fault Detection for Nonlinear Systems

For a nonlinear system with the formation of Eq. (9), HBF neural networks can be used to obtain the state observer. With the output values of the state observer, we can carry on the system output forecast in the next, and then achieve fault detection of the system[17, 18]. The output residual of the system prediction can be expressed as follows.

$$\boldsymbol{e}_y(t) = \boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t).$$

According to the design features of the state observer, $\boldsymbol{e}_y(t)$ quickly decays to zero, in order to achieve the forecast of the system output. However, when a system fault occurs, such as a sensor fault, system output changes. Since the self-learning of neural networks needs some time, at that time, the tracking ability to the state declines, which causes the output prediction residual of the system to change suddenly. This abrupt change can be used to detect the fault. We define the evaluation function of the residual as follows.

$$\Upsilon(t) = \boldsymbol{e}_y^{\mathrm{T}}(t)\boldsymbol{V}\boldsymbol{e}_y(t) \tag{15}$$

where $\boldsymbol{V}$ is a diagonal weighting matrix. Its form can be determined based on the specific features of practical problems.

The basic idea of residual evaluation is designing an effective decision logic and threshold. Once the fault residual exceeds the threshold, the warning is given immediately. Generally, the evaluation function of the residual and the selected threshold are compared to determine whether the fault occurs.

Thus, the rule of fault detection is as follows.

$$\Upsilon(t) \begin{cases} \leqslant T, & \text{normal;} \\ > T, & \text{failure} \end{cases} \tag{16}$$

where T is the threshold of fault detection. The detection principle is shown in Fig. 2.

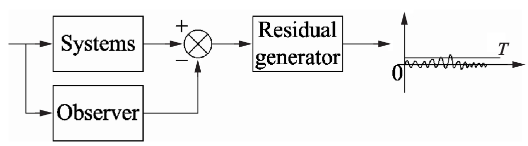

Fig. 2 Detection principle.

Generally, the threshold is a small positive constant. However, there are modeling errors, noise, interference, and other uncertain factors in real systems, which make the residual a stochastic process[2]. In this case, the conditional probability density curve of the residual without fault overlaps with the curve where the fault occurs. If the threshold is chosen inappropriately, a false alarm and missed detection may occur, which seriously affects the accuracy of fault detection.

5 Simulation

Considering the nonlinear system, its state equation can be described as Eq. (9). The corresponding parameter values are as follows.

$$\boldsymbol{A}=\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad \boldsymbol{C}=(1 \;\; 0),$$

$$\boldsymbol{g}(\boldsymbol{x},t)=\begin{pmatrix} 0.1\cos t \\ 2\cos t - 9.8\sin x_1 \end{pmatrix}.$$

The system input is white noise. Online training, forecasting, and tracking are performed using the method mentioned above. Figures 3–5 show the simulation results. The simulation results show that the HBF neural network observer has a better ability to track the state variable of the nonlinear system. However, due to the differences in the initial value chosen in experience, there is tracking error at the beginning phase. The changing rate of the state variable causes the tracking error at the inflection point of the curve. The average tracking error is less than 0.08.

We use this method to detect fault. Assume the sensor failure occurs at t = 15 s. At this point, the residual curve of the output forecasts shows serious deviation oscillation, as shown in Figs. 6 and 7.

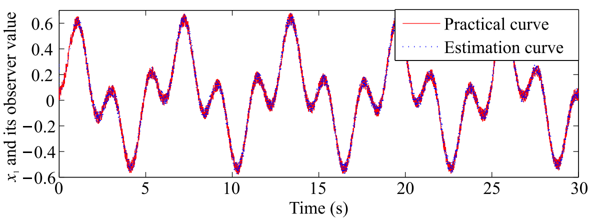

Fig. 3 State estimation curves of x_1 and $\hat{x}_1$.

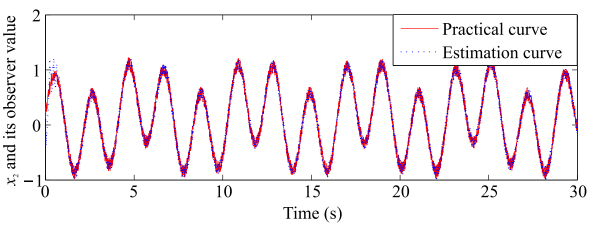

Fig. 4 State estimation curves of x_2 and $\hat{x}_2$.

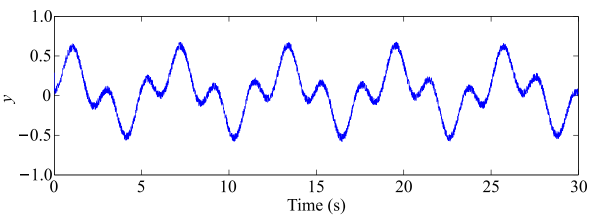

Fig. 5 Normal output y.

6 Conclusions

This study proposes an observer design method of fault detection based on HBF neural networks. The use of HBF networks requires a smaller number of neurons, reduces the complexity of the traditional observer design method, and has a higher sensitivity to the fault of nonlinear dynamic neural systems. This method does not significantly increase complexity even if there are multi-variable inputs. Therefore, it is easily extended to multi-input and multi-output systems for use in real-time online detection.

References

[1] H. Song, H. Y. Zhang, and X. R. Wang, Fault detection approach based on fuzzy observer for uncertain nonlinear systems, (in Chinese), *Aerospace Control*, vol. 23, no. 3, pp. 74–78, 2005.

[2] X. H. Zhu, Y. H. Li, N. Li, and J. D. Han, Novel observer-based robust fault detection method for nonlinear uncertain systems, (in Chinese), *Control Theory & Applications*, vol. 30, no. 5, pp. 644–648, 2013.

[3] L. E. Gao, W. D. Liu, and Y. Lu, Failure diagnose observer design and simulation for X-type rudder plane of underwater vehicle, (in Chinese), *Journal of Projectiles, Rockets, Missiles and Guidance*, vol. 28, no. 4, pp. 222–224, 2008.

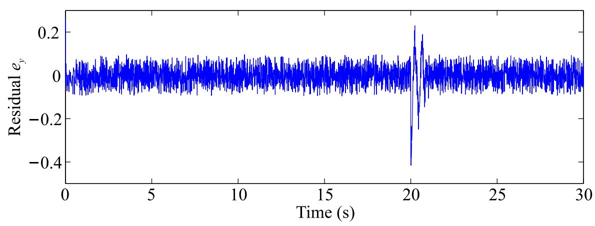

Fig. 6 Residual curve at the time of failure.

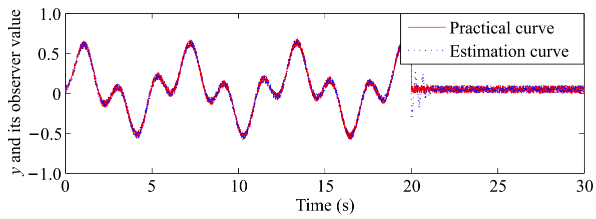

Fig. 7 Output y and its observer value at the time of failure.

[4] J. Zarei and E. Shokri, Robust sensor fault detection based on nonlinear unknown input observer, *Measurement*, vol. 48, no. 2, pp. 355–367, 2014.

[5] Z. H. Wang, M. Zhang, and Y. Shen, Actuator fault detection and isolation for the attitude control system of satellite, (in Chinese), *Journal of Harbin Institute of Technology*, vol. 45, no. 2, pp. 72–76, 2013.

[6] Y. Q. Song, W. G. Zhang, and X. X. Liu, Fault diagnosis based on RBF neural network observer in flight control system, (in Chinese), *Computer Simulation*, vol. 27, no. 3, pp. 85–88, 2010.

[7] Z. M. Du, B. Fan, J. L. Chi, and X. Q. Jin, Sensor fault detection and its efficiency analysis in air handling unit using the combined neural networks, *Energy and Buildings*, no. 72, pp. 157–166, 2014.

[8] Z. N. S. Vanini, K. Khorasani, and N. Meskin, Fault detection and isolation of a dual spool gas turbine engine using dynamic neural networks and multiple model approach, *Information Sciences*, no. 259, pp. 234–251, 2014.

[9] F. Schwenker, H. A. Kestler, and G. Palm, Three learning phases for radial-basis-function networks, *Neural Networks*, vol. 14, nos. 4&5, pp. 439–458, 2001.

[10] D. M. Adhyaru, State observer design for nonlinear systems using neural network, *Applied Soft Computing*, vol. 12, no. 8, pp. 2530–2537, 2012.

[11] H. W. Wu, Q. H. Dai, P. Wang, and Y. D. Li, Neural-network-based observers for nonlinear systems, (in Chinese), *Journal of Tsinghua University (Science and Technology)*, vol. 40, no. 3, pp. 44–47, 2000.

[12] J. M. Valls, R. Aler, and O. Fernandez, Using a Mahalanobis-like distance to train radial basis neural networks, *Computational Intelligence and Bioinspired Systems*, vol. 12, no. 8, pp. 2530–2537, 2012.

[13] S. A. Vorobyov and A. Cichocki, Hyper radial basis function neural networks for interference cancellation with nonlinear processing of reference signal, *Digital Signal Processing*, vol. 11, no. 3, pp. 204–221, 2001.

[14] N. Vukovic and Z. Miljkovic, A growing and pruning sequential learning algorithm of hyper basis function neural network for function approximation, *Neural Networks*, vol. 46, no. 10, pp. 220–226, 2013.

[15] A. J. Li, S. W. Luo, H. Huang, and Y. H. Liu, Decision tree based neural network design, (in Chinese), *Journal of Computer Research and Development*, vol. 2, no. 8, pp. 1312–1317, 2005.

[16] Z. C. Xu, Y. G. Zhou, and F. M. Yu, Design of observers for nonlinear systems based on improved neural-network, (in Chinese), *Microcomputer & Its Applications*, vol. 30, no. 8, pp. 76–78, 2011.

[17] F. A. Shaik, S. Purwar, and B. Pratap, Real-time implementation of Chebyshev neural network observer for twin rotor control system, *Expert Systems with Applications*, vol. 38, no. 10, pp. 13 043–13 049, 2011.

[18] X. Wen, Q. Wang, F. Qian, and H. Y. Zhang, Method of state observer design and fault detection, (in Chinese), *Journal of Beijing University of Aeronautics and Astronautics*, vol. 24, no. 6, pp. 676–679, 1998.

Xin Wen is a professor of Nanjing University of Aeronautics and Astronautics and of Shenyang Aerospace University. His main research areas include spacecraft design, fault detection and diagnosis, intelligent control, and formation control of satellites. Dr. Wen received his MS degree and PhD degree from Harbin Institute of Technology in 1993 and 1995, respectively. He became the young academic leader of China Academy of Space Technology in 1999 and was selected to be the first batch of “511 Talents Project” of National Defense in 2001. He has presided over the National 921 Project to build up “Shenzhou spacecraft fault modeling and simulation lab”, organized and completed a number of national “863” projects and assembly support pre-research projects, and has presided over “high-precision satellite navigation products industry” of the National Development and Reform Commission and other projects.

Xingwang Zhang is a master degree candidate in Shenyang Aerospace University. He received his BS degree from Ningbo University in 2012. His main research areas include spacecraft design, fault detection and diagnosis of spacecraft, and artificial intelligence.

Yaping Zhu is a master degree candidate in Nanjing University of Aeronautics and Astronautics. She received her BS degree from Nanjing University of Aeronautics and Astronautics in 2013. Her main research areas include artificial intelligence and fault detection and diagnosis of spacecraft.