Pose Error Compensation for the Actuator of an Automatic Docking Connector

Cheng Zhou, Cungui Yu, Huaying Liu, Chun Zhao
Nanjing University of Science and Technology, Nanjing, Jiangsu, 210094, China
Corresponding author: Cungui Yu; e-mail: yu_cungui@sina.com

Abstract. To improve the automatic docking connector (ADC) of a launch vehicle docking accuracy and reduce the pose error of the actuator, an error compensation prediction model based on BP neural network is established. An error compensation method based on pseudo-object pose is developed to ensure the real-time performance of the control system. A prototype ADC is constructed, and a control system was set up to verify the effectiveness of the proposed error compensation method. A comparison of the measured positions before and after the application of error compensation and the theoretic position of the reference point revealed that the output error could be significantly reduced after application of the proposed error compensation method.

1. Introduction
The automatic docking technology of a launch vehicle has received intense attention by launch vehicle experts seeking to shorten the time needed for preparing fuel filling, improving launch efficiency, and eliminating the safety risk of manual docking\cite{1,2}. The ADC of a certain launch vehicle consists of the actuator, pose compensation mechanism, locking mechanism, etc.\cite{3}. The 3-PSS parallel actuator is chosen to serve as the actuator of this launch vehicle for two reasons: a) compared with serial actuators, the parallel actuators have advantages like high rigidity, good stability, large load bearing capacity, small motion load, and no error accumulation; b) the launch vehicle has small swing angle\cite{4}. The precision of parallel actuators can be downgraded by the error that is created during the processes of component machining and assembly, the deformation of rods caused by external force or heating during the movement, the dynamic error of the hinge gap, and so on\cite{5}. To further improve the pose precision of the actuator, it is necessary to develop an effective terminal pose error compensation method. At present, the commonly used error compensation methods can be divided into hardware error compensation methods and software error compensation methods\cite{6}. The hardware error compensation methods can guarantee real-time compensation with high compensation accuracy, but the high cost prohibits its wide usage. The software compensation method is low in cost and easy to implement. The software error compensation method of the parallel actuator usually uses kinematic calibration to compensate for the pose error of the actuator caused by the mechanism error. The conventional kinematic calibration method is difficult to accurately model the error of multi-factor coupling\cite{7}. It is feasible to remedy the deficiency of the parameter calibration method based on the kinematic model by using the following method: establish a mapping model between the pose error and the driving rod length error at first, then obtain the optimal driving rod parameter using an intelligent algorithm (e.g., particle swarm optimization algorithm, ant colony algorithm\cite{8}, or BP neural network algorithm\cite{9}), and finally compensate the output pose error by adjusting the length of the driving rod. Among all calibration methods, the one based on a neural network stands out as an
easy-to-implement and high-accuracy method because it considers a variety of error factors and does not need to establish a complex error model.

In this study, an error compensation method based on the BP neural network prediction model is developed in order to enable high-precision docking of ADC under the influence of multiple coupled error sources. In addition, the effectiveness of the proposed method is verified through an experiment.

2. Description of the ADC actuator

The structure of the ADC of a certain launch vehicle is illustrated in Figure 1. The ADC consists of an actuator (3-PSS actuator), a pose compensation mechanism, a ground connector panel, an onboard connector panel, a center guiding structure, a locking mechanism, and so on. The 3-PSS actuator is composed of a fixed platform, three sets of identical rail-slider-rod mechanism, and a movable platform. The three guide rails are distributed equiangular on the circumference of the fixed platform, and the movement of the slider mounted on the guide rail is driven by the hydraulic cylinder. The slider and the movable platform are connected to the connecting rod through the ball joints at both ends of the connecting rod. The actuator controls the displacement of the slider by adjusting the length of the piston rod of the hydraulic cylinder and drives the movable platform to perform three-freedom-degree translational movement.

![Figure 1 Schematic diagram of ADC](image)

The forward kinematic solution of the actuator is to obtain the spatial position and pose of the actuator at the end of the actuator when the structural parameters and joint variables of the actuator are known. In contrast, the reverse kinematic solution is to obtain the joint variable when the spatial position and pose of the end actuator and the structural parameters of the actuator for accomplishing a certain job are known. For the 3-PSS parallel actuator, the forward kinematic solution is to obtain the spatial pose \((X, Y, Z)\) of the reference point of the movable platform when the driving rod lengths of the three hydraulic cylinders are known \((S_1, S_2, S_3)\); the reverse kinematic solution is to obtain the rod lengths of the three hydraulic cylinders when the pose \((X, Y, Z)\) of the reference point of the movable platform is known\(^{[10]}\).

This paper presents an error compensation strategy for the 3-PSS parallel actuator with the aim of enabling high-precision docking of ADC under the influence of multiple coupled error sources. First, the relationship between the pose of the movable platform and the length and error of the driving rod is established using a method based on a neural network, and this function relationship is used as the model for predicting the length error of the driving rod in the process of actuator error compensation. Then, using the predictive model to calculate the driving rod length correction of the mechanism in a certain position, the control system is used to control the modified driving rod length control so that the actual position of the end of the actuator approaches the theoretical pose, and finally, the actuator is realized.
3. **Error compensation prediction model based on a neural network**

3.1 **Design of the BP neural network**

The BP neural network is characterized by a simple network structure and strong self-learning ability. The BP neural network is used to learn the relationship between the pose coordinates \((x, y, z)\) of the movable platform and the length error \((\Delta S_1, \Delta S_2, \Delta S_3)\) of the driving rod in this study. The BP neural network is a typical multi-layer forward network with one input layer, several hidden layers, and one output layer\(^{11}\). The input layer is the pose coordinates \((x, y, z)\) of the movable platform, and the output layer is the driving rod length error \((\Delta S_1, \Delta S_2, \Delta S_3)\). The structure is illustrated in Figure 2. The BP neural network uses the error back-propagation algorithm to train the mapping function between the network input data and the network output data.

![Figure 2 Structure of BP neural network](image)

The complexity of the neural network has a major influence on the performance of the network, and therefore the neural network structure is optimized using the particle swarm optimization\(^{12}\). In a neural network, each neuron receives the output of the immediate upper layer as the input of this layer. The activation function controls the output of each neuron to change the linear relationship between the input and output of the neuron. The commonly used activation functions in neural networks include Sigmoid functions, threshold functions, linear functions, and piecewise functions. In this paper, the hidden layers of the neural network use the S-type activation function \(\text{tansig}\), and the output layer uses a linear function.

3.2 **Formation and standardization of sample data**

Obtaining the training data of the movable platform coordinates \((x, y, z)\) and the driving rod length error \((\Delta S_1, \Delta S_2, \Delta S_3)\) is the first part of the work of establishing the prediction model. This process is shown in Figure 3. The first step is to determine the object pose for measurement. To ensure the accuracy of the neural network training, it is necessary to select a sufficient number of object poses and ensure that the poses are evenly distributed in the operation space. First, the theoretical rod length \(S'\) corresponding to the object pose can be calculated through inverse kinematics solution. Then, the 3-PSS actuator is controlled to track the theoretical rod length \(S'\). Finally, the actual pose \((x, y, z)\) of the movable platform corresponding to the actual rod length \(S\) of the driving rod is measured. The actual pose coordinate set is used as the output samples in the training of the neural network.

![Figure 3 Flow chart of acquiring sample data](image)
3.3 Establishment of the predictive model

After the actual pose coordinate set \((x, y, z)\) and the driving rod length error set \((\Delta S_1, \Delta S_2, \Delta S_3)\) are normalized, they can be used to train the neural network structure and establish a predictive model. The procedure is as follows:

1. Define the loss function of the neural network and use it as an index for evaluating the neural network. The mean square error (MSE) is used as the loss function.
2. Configure the BP neural network algorithm and set related parameters of the particle swarm optimization.
3. Write a neural network training program in MATLAB and start training the network. Upon the completion of the training, the weight and threshold values of the network are extracted, and the expression of the neural network is formulated. The expression of the neural network serves as the expression of the function relationship between the pose of the movable platform and the driving rod length error, and the function is the predictive model of actuator error compensation. The mapping function of the neural network is as follows:

\[
(\Delta S_1, \Delta S_2, \Delta S_3)^T = \tan \text{sgn} \left[w_{ij} \cdot \left[\tan \text{sgn}(w_{(i-1)(j-1)} \cdot \left[\ldots \tan \text{sgn}(w_{10} \cdot (x, y, z)^T) + b_1 + \ldots\right] + b_i\right]\right] \tag{1}
\]

where \((x, y, z)^T\) is the input of the neural network; \((\Delta S_1, \Delta S_2, \Delta S_3)^T\) is the output of the neural network; \(w_{ij}(i \geq 1, j \geq 0)\) represents the weight of the connection between the \(i\)-th layer and the \(j\)-th layer of the neural network (the weight of the connection between the input layer and the first hidden layer of the network is \(w_{10}\)); and \(b_i\) represents the threshold of each hidden layer of the neural network.

3.4 Error compensation method based on the prediction model

To ensure the real-time performance of the control system, it is necessary to write an analytic expression program of the reverse kinematic solution to realize the control of the actuator. Although the prediction model based on neural network algorithm achieves high precision, the analytic expression for the neural network training result represented by Formula (1) is very complex. On the one hand, the \(\text{tansig}\) function in the expression is difficult to program, especially the several nested functions in the expression. On the other hand, due to the high computational complexity of the prediction model, it is difficult to guarantee high real-time performance in the control process. This paper proposes a compensation control flow based on the pseudo-object pose, as shown in Figure 4. First, the theoretical length of the driving rod corresponding to the object pose is obtained through the reverse kinematic solution. At the same time, the rod length error is obtained using the error prediction model, and the corrected rod length is obtained by adding the theoretical rod length and the rod length error. Then, the pseudo-object pose is obtained according to the corrected rod length through a forward kinematic solution, and the actual rod length is obtained through the reverse kinematic solution. Finally, the 3-PSS actuator is controlled to track the actual rod length, which reduces the discrepancy between the actual pose and the object pose. Despite the ease of implementation, this method is not universally applicable in practice because each object pose requires the calculation of pseudo-object pose in advance. To further simplify the process of solving the pseudo-object pose, one effective method is to establish the function relationship between the object pose and the pseudo-object pose using multivariate nonlinear fitting equation, and thus the calculation process can be simplified using such function relationship. The simplified error compensation control flow is shown in Figure 5.
Figure 4 Error compensation control flow based on the pseudo-object pose

Figure 5 Error compensation control process

Multivariate nonlinear fitting is a process of solving the function relationship between multiple dependent and independent variables using a certain fitting algorithm based on a nonlinear model involving these dependent variables and independent variables\cite{13}. In this paper, the problem of solving the relationship between the object pose \((x_n, y_n, z_n)\) and pseudo-object pose \((x_m, y_m, z_m)\) involving multi-variable input and multi-variable output is simplified into the problem of multi-variable input and single-variable output, that is, \((x_m, y_m, z_m) = f(x_n, y_n, z_n)\) is converted to the function relationship problem indicated by Equation (2).

\[
\begin{align*}
  x_m &= f_1(x_n, y_n, z_n) \\
  y_m &= f_2(x_n, y_n, z_n) \\
  z_m &= f_3(x_n, y_n, z_n)
\end{align*}
\]  

For the problem of solving the single function relationship in Equation (2), the cftool (Curve Fitting Tool) toolkit in MATLAB is used for nonlinear fitting.

4. Experimental verification

4.1 Experimental verification system

The driving rod length \((S_1, S_2, S_3)\), which is the actual displacement of the hydraulic cylinder piston rod, was acquired in real time by a magnetostrictive displacement sensor, and the pose of the movable platform \((x, y, z)\) was measured by a coordinate measuring device based on pull-wire linear displacement sensor. The experimental verification system is shown in Figure 6.
The sample data distribution of the neural network should be controlled to ensure that as many as possible points fall within the operation space of the ADC. The theoretical displacement equation of the movable platform is as follows:

\[
\begin{align*}
x_d &= \sqrt{2}/2 \times 0.065 \sin (0.4 \times 2\pi t) (1 - \exp(-0.001t^3)) \\
y_d &= 0.02, 0.04, ... 0.12 \\
z_d &= \sqrt{2}/2 \times 0.065 \sin (0.4 \times 2\pi t) (1 - \exp(-0.001t^3))
\end{align*}
\] (3)

4.2 Results of the movable platform position measurement

During the experiment, the control system used the theoretical displacement \((x_d, y_d, z_d)\) of the movable platform to obtain the driving rod length \((S_1', S_2', S_3')\) through the reverse kinematic solution. After that, the control system drove the actuator to move according to the driving rod length \((S_1', S_2', S_3')\), and the coordinate measuring device measured the actual position of the movable platform \((x, y, z)\). The test data was processed by MATLAB, and the motion trajectories of the movable platform in the three axial directions were plotted to give a visualized presentation of the experimental result. Figure 7 shows the motion trajectories of the reference point of the movable platform in the three axial directions when \(y_d = 0\).

4.3 Neural network training

The set of collected data, including the position of the movable platform \((x, y, z)\) and the driving rod length \((S_1, S_2, S_3)\), provided samples to train the neural network. The process is as follows:

1. Feed the actual position of the movable platform \((x, y, z)\) to the neural network as the input sample.
2. The theoretical driving rod length \((S_1', S_2', S_3')\) is obtained using the actual position of the movable platform \((x, y, z)\) through the reverse kinematic solution.
3. Calculate the difference between the theoretical length \((S_1', S_2', S_3')\) of the driving rod and the measured rod length \((S_1, S_2, S_3)\), and the result is the driving rod length error \((\Delta S_1, \Delta S_2, \Delta S_3)\), which is the output sample of the neural network. This sample will undergo a min-max normalization process.
4. Import the sample data into the MATLAB toolkit for neural network training. The number of hidden layers of the neural network was set to 5, and the range of the number of nodes in each layer...
was [3, 7]. The key parameters of the neural network are shown in Table 1. Three conditions were used to terminate network training: 1) The error of the training set was less than the minimum error of the training set error (1e-06); 2) The error of the verification set did not decrease for 100 accumulated number of training rounds; 3) When the number of training rounds reached the maximum number of training rounds (2,000). The network training was terminated once any of the above three conditions were met.

Table 1. Important parameters of BP neural network algorithm and particle swarm optimization

| BP neural network | Particle swarm optimization |
|-------------------|----------------------------|
| Number of samples in the training set | 2000 | Population size | 20 |
| Number of samples in the verification set | 1000 | Maximum number of iterations | 100 |
| Number of samples in the test set | 1000 | Speed update parameter | 1.49445 |
| Maximum number of training rounds | 2000 | Speed range | [-0.5, 0.5] |
| Minimum error of the training set | 1e-06 | Position range | -2,2 |
| Accumulated number of training rounds without decrease of verification set error | 100 |

Figure 8 shows the loss function convergence curves of the training set, validation set, and test set. As shown from the figure, as the number of training rounds increased, the errors of the training set, the verification set, and the test set all decreased gradually. The error of the verification set did not decrease from the 138th to the 238th training round, and thus the network training was terminated at the 238th training round.

4.4 Nonlinear fitting

Given the object position, the corresponding pseudo-object position can be calculated using the neural network training result. The object position and the pseudo-object position were as shown in Figure 9.
Totally 100 sets of data were imported into the MATLAB *cftool* toolkit for nonlinear fitting. The equation obtained through multiple times of fitting is:

\[
\begin{align*}
    x_m &= -3.516 + 1.011x_n - 0.000246y_n + 0.000106x_n^2 + (7.622e^{-5})x_ny_n \\
    y_m &= -0.4613 + 0.004258x_n + 0.9988y_n + (1.115e^{-5})x_n^2 + (4.118e^{-5})x_ny_n \\
    z_m &= -5.444 + 0.008218x_n + 1.001z_n + (4.625e^{-5})x_n^2 + (1.421e^{-5})x_nz_n
\end{align*}
\]

4.5 **Result**

The fitting Equation (4) was written into the program of the control system, and the motion trajectory of the movable platform could be obtained using Equation (3). The theoretical and measured motion trajectories of the reference point of the movable platform in the three axial directions are shown in Figure 10.

5. **Conclusion**

In this study, an error compensation method based on the BP neural network prediction model was developed to improve the docking accuracy of the 3-PSS actuator of the ADC of a certain launch vehicle. An ADC experimental prototype was constructed to serve as the experiment platform, and a motion control platform was constructed based on an xPC object. The experimental results showed that the output error could be reduced dramatically through error compensation, which proves the effectiveness of the position error compensation method proposed in this paper.
References

[1] I.H. Jung and D.K. Hwang 2011 Control of Mechanical Ground Support Equipment for Korean Launch Complex AIAC14 Fourteenth Australian International Aerospace Congress.

[2] Nasser Houshangi 2000 Autonomous robotic positioning umbilical system IEEE Conference Publications 4 pp 2981-29.

[3] Li Baoping 2017 Research on the design of pose compensation mechanism for automatic docking connector Nanjing University of Science and Technology.

[4] Wouter D.van Dorsser and Rogier Barents 2007 Gravity-Balanced Arm Support With Energy-Free Adjustment Journal of Medical Devices 2007(1) pp 151-6.

[5] Wang Zezheng 2012 Error Analysis and Compensation of 6-PSS Parallel Robot Chongqing: Chongqing University.

[6] Liu Zhijie 2010 Research on pose error analysis and compensation method of parallel robot Yanshan University.

[7] Wang Dongshu and Chi Jiannan 2007 Overview of robot kinematics calibration Journal of Computer Applications 24(9) pp 8-11.

[8] Li Bingbing 2014 Research on Modeling and Compensation of Pose Error of Parallel Robot Based on Particle Swarm Optimization Journal of Northeastern University.

[9] Wang Zhanzhong, Wang Wenlong and Yan Chaosong 2015 Research and simulation of inverse neural calibration method for robot neural network Manufacturing Automation 37(06) pp 12-15.

[10] Huang Chao 2015 Implementation of automatic docking connector, suspension system design and research Nanjing University of Science and Technology.

[11] Yu Lei, Shi Feng and Wang Hui 2015 30 Case Analysis of MATLAB Intelligent Algorithm Beijing: Beijing Aerospace University Press pp 237-9.

[12] Shi Feng, Wang Xiaochuan and Yu Lei 2010 30 Case Analysis of MATLAB Neural Network Beijing: Beijing Aerospace University Press pp 236-42.

[13] Dong Daxiao 2009 Multivariate Nonlinear Regression Model Based on MATLAB Journal of Yunnan Normal University (Natural Sciences) 29(02) pp 45-8.