Hindawi
Complexity
Volume 2020, Article ID 5896813, 13 pages
https://doi.org/10.1155/2020/5896813

Research Article

Improvement of Heavy Load Robot Positioning Accuracy by Combining a Model-Based Identification for Geometric Parameters and an Optimized Neural Network for the Compensation of Nongeometric Errors

Yuxiang Wang,[1] **Zhangwei Chen**,[1] **Hongfei Zu**,[2] **Xiang Zhang**,[3] **Chentao Mao**,[1] **and Zhirong Wang**[1]

[1]*School of Mechanical Engineering, Zhejiang University, Hangzhou 310000, China*
[2]*Faculty of Mechanical Engineering&Automation, Zhejiang Sci-Tech University, Hangzhou 310000, China*
[3]*Institute of Computer Science, Hangzhou Dianzi University, Hangzhou 310000, China*

Correspondence should be addressed to Zhangwei Chen; chenzw@zju.edu.cn

Received 20 May 2019; Revised 23 September 2019; Accepted 22 October 2019; Published 16 January 2020

Academic Editor: Xiaoping Liu

The positioning accuracy of a robot is of great significance in advanced robotic manufacturing systems. This paper proposes a novel calibration method for improving robot positioning accuracy. First of all, geometric parameters are identified on the basis of the product of exponentials (POE) formula. The errors of the reduction ratio and the coupling ratio are identified at the same time. Then, joint stiffness identification is carried out by adding a load to the end-effector. Finally, residual errors caused by nongeometric parameters are compensated by a multilayer perceptron neural network (MLPNN) based on beetle swarm optimization algorithm. The calibration is implemented on a SIASUN SR210D robot manipulator. Results show that the proposed method possesses better performance in terms of faster convergence and higher precision.

1. Introduction

The positioning accuracy of robots is of great significance in advanced robotic manufacturing systems. In general, the robot shows good repeatability, but poor positioning accuracy. Many factors would result in errors in positioning, including set-up errors, manufacturing tolerance, wear and tear, transmission errors, and compliance [1]. Therefore, researchers have focused on how to improve the positioning accuracy of a robot by errors compensation according to the factors mentioned above.

The calibration technique can improve the robot's positioning accuracy through software algorithm without changing its mechanical structure or design. Generally, calibration can be classified into two types, namely geometric and nongeometric calibration. Almost 80% of robot positioning errors are caused by errors of geometric parameters, including link length errors, link twist angle errors, link offset, and joint angle offsets [2]. Therefore, most researches have concentrated on kinematic-based calibration [3–9]. Most of the researchers employed the D-H model which employs a minimum set of parameters to describe the relationship of joint coordinates for calibration. However, D-H model is not continuous when two consecutive joint axes are parallel or nearly parallel. In order to overcome these drawbacks, Hayati proposed a new model namely MDH model by adding a redundant parameter to the D-H model [10]. However, both the two methods are complicated in the process of modeling a manipulator. Brockett firstly used the POE formula to describe an open-chain robot [11]. The POE formula possesses a simpler expression to describe the relationship between the joint angles and the end-effector's position and posture. Additionally, previous studies mostly focused on the calibration of the link lengths and the joint offsets, and little attention has been paid to the errors of the reduction ratio and the coupling ratio. The reduction ratio and coupling ratio errors mainly caused by the deviation during manufacturing and assembly process will lead to joint angle errors, further resulting in a decline in the positioning accuracy. Moreover, owing to the nonlinear trigonometric relationship between joint angles and the pose of endpoint, it is complicated to identify the transmission errors using the DH model [12]. The POE formula, however, possesses a simpler expression.

However, the kinematic-based calibration has the following limitations: (1) the model only takes the geometric factors into account, but the nongeometric factors such as backlash and joint compliance, which may have significant influence on positioning accuracy, are ignored. (2) The identification process is a complex numerical procedure which may suffer from the numerical problem of ill-conditioning. (3) The implementation of the identified model is problematic due to the difficulty of modifying the controller parameters [13].

In fact, the nongeometric errors are an important component of the parameters identification, especially in high-precision application scenarios. The positioning errors caused by joint stiffness account for 6%~8% as well as other nongeometric errors such as gear backlash and temperature drift [14]. Some researchers investigated both the kinematic parameters and joint compliance [15, 16]. However, other nongeometric errors still affect the positioning accuracy significantly. Especially for a heavy load robot, not only the joint compliance but also the link compliance affects the accuracy [17, 18].

Recently, for nonparametric calibration, some intelligent algorithms, such as genetic algorithm [7, 19, 20], maximum likelihood estimation [21, 22], neural network [23–29], and various hybrid algorithms [30], have emerged. Among them, neural network has been widely employed to build the relationship, especially nonlinear relationship between inputs and outputs. It can approximate any nonlinear function with arbitrary precision [31]. The neural network is utilized to build the relationship between the end-effector position and the error of the position in [25]. However, the positioning error depends on the configurations. In [26], a multilayer perceptron neural network is utilized to describe the relationship between the joint angles and the corresponding joint errors. Nguyen proposed a technique for the calibration of industrial robots by combining the geometric model-based calibration method and the ANN to identify the kinematic errors, joint compliance errors, and the nongeometric errors [29]. But, at the same time, the BP neural network has the problem that the performance relies too much on the input data and initial values of weights and biases and easily falls into the local optimum. To overcome these drawbacks, researchers have proposed many hybrid algorithms such as GA-BPNN [32, 33] and PSO-BPNN [34]. But, convergence rate of these algorithms is relatively slow, and implementation of code is complex.

In this paper, an enhanced POE-based method is proposed for the identification of geometric parameters, reduction ratio, and coupling ratio. Joint stiffness identification is carried out by adding a load to the end-effector. For the nongeometric errors compensation, an optimized MLPNN based on a hybrid optimization method of beetle antennae search algorithm and particle swarm optimization is proposed. Experiments are carried out on a SIASUN SR210D robot manipulator to validate the proposed method by using a laser tracker to measure the position of the Spherical Mounted Retroreflector (SMR) located at the end flange of the manipulator.

The rest of the paper is organized as follows. In Section 2, the kinematic model of the SIASUN SR210D robot manipulator is presented. In Section 3, the POE-based error model including the transmission errors is established. Section 4 proposes the joint stiffness identification method. In Section 5, a hybrid beetle swarm optimization and MLPNN algorithm are presented for the nongeometric errors compensation. In Section 6, Compensation verification and analysis are presented. In Section 7, conclusions and future works are presented.

2. Kinematic Model of the SIASUN SR210D Robot Manipulator

In this section, a kinematic model of the SIASUN SR210D robot manipulator is described in detail. The structure of the 6R manipulator is in Figure 1. $L_1, L_2, L_3, L_4,$ and L_5 are the link lengths and $\omega_1, \omega_2, \omega_3, \omega_4, \omega_5,$ and ω_6 are the joint angular velocities, respectively. The nominal value of robot structure parameters mentioned above is listed in Table 1.

Then, the screw of each joint is

$$\xi_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \xi_2 = \begin{bmatrix} 0 \\ 0 \\ -L_1 \\ 0 \\ -1 \\ 0 \end{bmatrix} \xi_3 = \begin{bmatrix} L_2 \\ 0 \\ -L_1 \\ 0 \\ -1 \\ 0 \end{bmatrix} \xi_4 = \begin{bmatrix} 0 \\ L_2 + L_3 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \xi_5 = \begin{bmatrix} L_2 + L_3 \\ 0 \\ -L_1 - L_4 \\ 0 \\ -1 \\ 0 \end{bmatrix} \xi_6 = \begin{bmatrix} 0 \\ L_2 + L_4 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}. \quad (1)$$

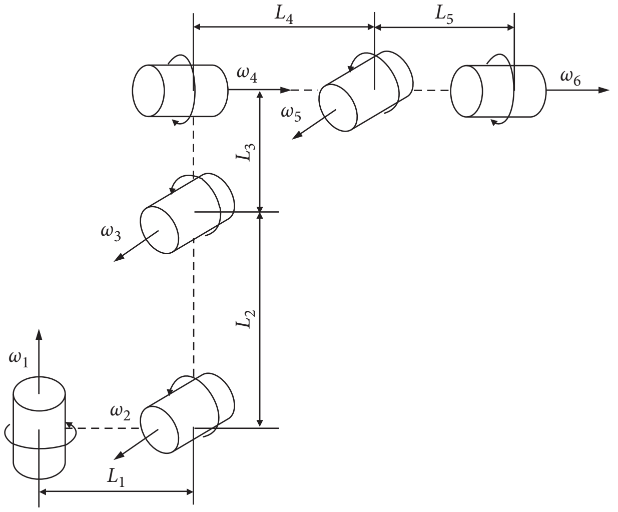


Figure 1: Structure of 6R SIASUN SR210D manipulator.

Table 1: Nominal value of robot structure parameters.

L_i, $i = 1$–5 (mm)	312	1075	235	1282	260	
$\mathrm{d}\theta_i$, $i = 1$–6 (°)	0	0	0	0	0	0

The forward kinematics can be expressed as mentioned below:

$$P_{cT} = e^{\hat{\xi}_1\theta_1} \cdots e^{\hat{\xi}_n\theta_n} P_{cT0}, \tag{2}$$

where θ_i is the angle of the *ith* joint and P_{cT0} is the end-effector position in the base frame corresponding to $\theta_i = 0$.

3. Geometric Parameters and Transmission Errors Identification

3.1. Model of Geometric Parameters Errors. The position of the tool center point (TCP) in the measure frame is shown below:

$$P = M \cdot P_{cT} = M \cdot g(\theta) \cdot P_{cT0}, \tag{3}$$

where P is the position of the TCP, M is a homogenous matrix that represents the transformation from the measuring device frame to the base frame, and P_{CT} represents the position of TCP expressed in the base frame corresponding to $\theta = [\theta_1, \theta_2, \ldots, \theta_n]$, $g(\theta) = e^{\hat{\xi}_1\vartheta_1} e^{\hat{\xi}_2\vartheta_2} \cdots e^{\hat{\xi}_n\vartheta_n}$.

Then, we have the deviation of the TCP position by taking the differential on both sides of equation (3):

$$\begin{aligned}\mathrm{d}P &= \mathrm{d}M \cdot g \cdot P_{cr0} + M \cdot \mathrm{d}g \cdot P_{cr0} + M \cdot g \cdot \mathrm{d}P_{cr0},\\ &= \mathrm{d}M \cdot M^{-1} \cdot P + M \cdot \mathrm{d}g \cdot g^{-1} \cdot P_{cr} + M \cdot g \cdot \mathrm{d}P_{cr0},\end{aligned} \tag{4}$$

where

$$g \cdot \mathrm{d}P_{cT0} = \begin{bmatrix} R_g & b_g \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathrm{d}P_{cT0} \\ 0 \end{bmatrix} = \begin{bmatrix} R_g \mathrm{d}P_{cT0} \\ 0 \end{bmatrix}. \tag{5}$$

Taking a close observation of (4), we notice that $\mathrm{d}M \times M^{-1} \in se(3)$, $\mathrm{d}g \times g^{-1} \in se(3)$, and the element s belonging to lie algebra satisfies

$$sq^* = \begin{bmatrix} \hat{\omega} & V \\ o & 0 \end{bmatrix} \begin{bmatrix} q \\ 1 \end{bmatrix} = \begin{bmatrix} \hat{\omega} q + v \\ 0 \end{bmatrix} = \begin{bmatrix} I_3 & -\hat{q} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \tag{6}$$

$\mathrm{d}P$ can be expressed as

$$\begin{aligned}\mathrm{d}P = \begin{bmatrix} I & -\hat{P} \\ 0 & 0 \end{bmatrix} \left(\mathrm{d}M \cdot M^{-1}\right)^V + M \begin{bmatrix} I & -\hat{P}_{cT} \\ 0 & 0 \end{bmatrix} \left(\mathrm{d}g \cdot g^{-1}\right)^V \\ + M \begin{bmatrix} R_g \mathrm{d}P_{cT0} \\ 0 \end{bmatrix}.\end{aligned} \tag{7}$$

Since rigid body motion can be realized by a rotation and a translation about an axis, M can be expressed as

$$M = e^{\hat{\xi}}\vartheta_0 \in SE(3). \tag{8}$$

Take the differential on both sides:

$$\mathrm{d}M \cdot M^{-1} = \mathrm{d}e^{\hat{\xi}_0\vartheta_0} \cdot e^{-\hat{\xi}_0\vartheta_0 = \theta_0} \int_0^1 e^{\hat{\xi}_0\vartheta_0 s} \mathrm{d}\hat{\xi}_0 e^{-\hat{\xi}_0\vartheta_0 s} \mathrm{d}s \,. \tag{9}$$

The matrix g is

$$g = \begin{bmatrix} R_g & b_g \\ 0 & 1 \end{bmatrix} \in SE(3). \tag{10}$$

Take the differential on both sides:

$$\begin{aligned}\mathrm{d}g \cdot g^{-1} &= de^{\hat{\xi}_1\theta_1} e^{-\hat{\xi}_1\theta_1} + e^{\hat{\xi}_1\theta_1} de^{\hat{\xi}_2\theta_2} e^{-\hat{\xi}_2\theta_2} e^{-\hat{\xi}_1\theta_1} + \\ &\quad \cdots + e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2}\theta_2 \ldots e^{\hat{\xi}_{n-1}\theta_{n-1}} de^{\hat{\xi}_n\theta_n} e^{-\hat{\xi}_n}\theta_n \ldots e^{-\hat{\xi}_2\theta_2} e^{-\hat{\xi}_1\theta_1},\\ &= de^{\hat{\xi}_1\theta_1} e^{-\hat{\xi}_1\theta_1} + Ad_{e^{\hat{\xi}_1\theta_1}} de^{\hat{\xi}_2\theta_2} e^{-\hat{\xi}_2\theta_2} + \cdots + \\ &\quad Ad_{e^{\xi_1\theta_1} e^{\hat{\xi}_2\theta_2} \ldots e^{\hat{\xi}_{n-1}\theta_{n-1}}} de^{\hat{\xi}_n\theta_n} e^{-\hat{\xi}_n\theta_n},\end{aligned} \tag{11}$$

where $d(e^{\hat{\xi}_i\theta_i})e^{-\hat{\xi}_i\theta_i}$ is

$$\begin{aligned}d\left(e^{\hat{\xi}_i\theta_i}\right)e^{-\hat{\xi}_i\theta_i} &= \int_0^1 e^{\hat{\xi}_i\theta_i s}\left(\mathrm{d}\hat{\xi}_i\right)\theta_i e^{-\hat{\xi}_i\theta_i s} \mathrm{d}S + \int_0^1 e^{\hat{\xi}_i\theta_i s}\xi,\\ &= \theta_i \int_0^1 Ad_{\widehat{e^{\xi_i\theta_i s}}} \mathrm{d}\hat{\xi}_i \mathrm{d}s + \hat{\xi}_i \mathrm{d}\theta_i.\end{aligned} \tag{12}$$

3.2. Model of Transmission Errors. Transmission errors mainly include the reduction ratio error and coupling ratio error. The errors are caused by the process of manufacturing and assembly. Both the errors lead to positioning inaccuracy.

Considering the transmission errors, the nominal value of the *ith* joint angle can be given by

$$\begin{aligned} \theta_i^n &= \theta_i^{n'} + \sum_{j=1}^{n} c_{ji}^n \theta_j^{n'}, \\ k_i &= \frac{r_i^n}{r_i^a} = \frac{\theta_i^m/\theta_i^{n'}}{\theta_i^m/\theta_i^{a'}}, \\ h_{ji} &= c_{ji}^a - c_{ji}^n, \end{aligned} \tag{13}$$

where r_i^n and r_i^a are the nominal and identified reduction ratios of the *ith* joint. c_{ji}^n and c_{ji}^a are the nominal and identified coupling ratios. $\theta_i^{n'}$ and $\theta_i^{a'}$ are the nominal and actual output angles of the *ith* joint reducer. θ_i^m is the *ith* joint motor rotation angle. k_i and h_{ji} are the errors of the reduction ratio and the coupling ratio, namely the transmission errors.

Then, the *ith* joint angle θ_i can be expressed as

$$\begin{aligned} \theta_i &= \theta_{i0} + \theta_i^{a'} + \sum_{j=1}^{n} c_{ji}^a \theta_j^{a^*} = \theta_{i0} + k_i\theta_i^n \\ &\quad - k_i \sum_{j=1}^{n} c_{ji}^n \theta_j^{n'} + \sum_{j=1}^{n} c_{ji}^n \theta_j^{a^*} + \sum_{j=1}^{n} h_{ji}\theta_j^{a^*}, \\ &= \theta_{i0} + k_i\theta_i^n + \sum_{j=1}^{n}\left(k_j - k_i\right)c_{ij}^n\theta_j^n + \sum_{j=1}^{n} h_{ij}k_j\theta_j^n, \end{aligned} \tag{14}$$

where θ_{i0} is the zero positioning error of the joint angle θ_i, namely the *ith* joint offset and θ_j is the *jth* joint angle which is coupled with θ_i.

Furthermore, the deviation of the *ith* joint angle θ_i can be deduced as

$$\mathrm{d}\theta_i = \mathrm{d}\theta_{i0} + \left(\theta_i^n - \sum_{j=1}^{n} c_{ji}^n\theta_j^n\right)\mathrm{d}k_i + \sum_{j=1}^{n} k_j\theta_j^n \mathrm{d}h_{jt}. \tag{15}$$

3.3. Linearization of the Error Model. Linearization of the error model is a premise for parameters identification. The error model can be obtained by combining the two error models mentioned above as

$$\begin{aligned} \mathrm{d}P &= \theta_0 \begin{bmatrix} I & -\widehat{P} \\ O & 0 \end{bmatrix} \int_0^1 Ad_{e^{\widehat{\xi_0}\theta_0 s}} \mathrm{d}s\mathrm{d}\xi_0 \\ &\quad + M\begin{bmatrix} I & -\widehat{P}_{cT} \\ O & 0 \end{bmatrix}\left(\mathrm{d}g\cdot g^{-1}\right)^V + M\begin{bmatrix} R_g \mathrm{d}P_{cr0} \\ 0 \end{bmatrix}, \end{aligned} \tag{16}$$

where

$$\begin{aligned} \left(\mathrm{d}g\cdot g^{-1}\right)^V &= \theta_1 \int_0^1 Ad_{e^{\widehat{\xi_1}\theta_1 s}} \mathrm{d}s\mathrm{d}\xi_1 + \xi_1\mathrm{d}\theta_1 + \\ &\quad Ad_{e^{\widehat{\xi_1}\theta_1}}\left(\theta_2 \int_0^1 Ad_{e^{\widehat{\xi_2}\theta_{2S}}} \mathrm{d}s\mathrm{d}\xi_2 + \xi_2\mathrm{d}\theta_2\right) + \cdots + \\ &\quad Ad_{e^{\widehat{\xi_1}\theta_1} e^{\widehat{\xi_2}\theta_2}\ldots e^{\widehat{\xi_{n-1}}\theta_{n-1}}}\left(\theta_n \int_0^1 Ad_{e^{\xi_n\theta_{nS}}} \mathrm{d}s\mathrm{d}\xi_n + \xi_n\mathrm{d}\theta_n\right), \end{aligned} \tag{17}$$

where $e^{\widehat{\xi_i}\theta_i}$ can be expressed as a homogeneous matrix through Taylor expansion and the adjoint transformation associated with $e^{\widehat{\xi_i}\theta_i}$ can be expressed as a 6×6 matrix. Equation (16) can be deduced as

$$\mathrm{d}P = Ax, \tag{18}$$

where

$$\mathrm{d}P = P^a - P^n, \tag{19}$$

where P^a is the actual TCP position measured by the laser tracker and P^n is the nominal TCP position obtained by the forward kinematics. The parameters to be identified can be expressed as

$$x = \left[\mathrm{d}\xi_0^T, \mathrm{d}\xi_1^T \ldots, \mathrm{d}\xi_n^T \ldots, \mathrm{d}\theta_{n0} \ldots, \mathrm{d}k_n, \mathrm{d}h_{21} \ldots, \mathrm{d}h_{ji}, \mathrm{d}P_{cT}^T\right]^T. \tag{20}$$

A is the corresponding coefficient matrix which can be expressed as

$$A = \left[Q_0 \mid Q_1 \ldots \mid Q_n \mid T_1 \ldots T_n \mid K_1 \cdots K_n \mid H_{21} \ldots H_{ji} \mid MR_g\right]. \tag{21}$$

With the measurement of the TCP position at different configurations, we can get the following equation set:

$$\mathrm{d}\widetilde{P} = \begin{bmatrix} \mathrm{d}P_1 \\ \cdots \\ \mathrm{d}P_n \end{bmatrix} = \begin{bmatrix} A_1 \\ \cdots \\ A_n \end{bmatrix} x = \widetilde{A}x, \tag{22}$$

where n is the number of measurements.

The least-squares solution for x is

$$x = \left(\widetilde{A}^T \cdot \widetilde{A}\right)^{-1} \cdot \widetilde{A}^T \cdot \mathrm{d}\widetilde{P}. \tag{23}$$

4. Joint Stiffness Identification

The stiffness of the manipulator arm is much greater than that of the joint. For this reason, the joint stiffness identification is investigated in this section.

For a joint, the dynamic model can be described as

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta}) + \tau_g(\theta) + \tau_f(\dot{\theta}) = \tau_m, \tag{24}$$

where $M(\theta)$ is the inertial matrix, $C(\theta,\dot{\theta})$ is the Coriolis and centrifugal terms, $\tau_g(\theta)$ relates to the gravitational torque, and $\tau_f(\dot{\theta})$ is the friction torque in the joint. τ_m is the input torque from the joint motor.

On the assumption that the angular deformation of the reducer is proportional to the input torque, the relationship between the input torque and the deformation of the reducer is as follows.

$$\tau_{mi} = k_i^*\,\delta\theta_{ki}, \tag{25}$$

where k_i is the stiffness coefficient of the *ith* joint and $\delta\theta_{ki}$ is the angular deformation of the *ith* joint. The joint torque τ_m is obtained as

$$\tau_m = C_e\phi I, \tag{26}$$

where I is the motor current that can be read directly from the controller of the robot. C_e is the motor potential constant, and ϕ is the magnetic flux.

For a manipulator, the stiffness matrix of the robot joint can be obtained as follows by ignoring the interaction of each joint.

$$K = \begin{bmatrix} k_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & k_n \end{bmatrix}, \tag{27}$$

where n is the number of degrees of freedom.

Substitute equation (25) into equation (24), and the relationship between joint angle deformation and motor control current can be given by

$$\delta\theta_{ki} = \left(\frac{C_e\phi}{k_i}\right) * I_i. \tag{28}$$

The flexibility matrix of the robot joint is as follows:

$$C = \begin{bmatrix} \dfrac{C_{e1}\phi_1}{k_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \dfrac{C_m\phi_n}{k_n} \end{bmatrix}. \tag{29}$$

According to the differential error relationship, the joint angle deviation vector $x = [\delta\theta_{k1} \cdots, \delta\theta_{kn}]^T$ can be obtained.

Since only the gravity load is loaded during the experiment, the direction of gravity is consistent with the direction of the first axis of the robot and the moment of gravity decomposition to the first axis is 0, and the first axis is not identified. Similarly, the flexibility coefficient of the sixth axis is not identified.

Substitute the flexibility matrix into the equation below to improve the positioning accuracy by compensating the joint angles.

$$\theta^* = \theta^n + IC, \tag{30}$$

where θ^* is the modified joint angle value of the robot. θ^n is the nominal value of the joint angle, and the vector I is the current value of each joint motor.

5. MLPNN Based on a Hybrid Optimization Method of Beetle Antennae Search Algorithm and Particle Swarm Optimization for Nongeometric Errors Compensation

The positioning error is still large after the geometric parameters, transmission errors, and joint stiffness compensation. The residual positioning errors are caused by nongeometric errors which is difficult to model. In order to overcome the drawbacks, in this section, a MLPNN based on a new hybrid optimization method is proposed. This new model will be presented in details.

5.1. Beetle Antennae Search Algorithm (BAS)

5.1. Beetle Antennae Search Algorithm (BAS). The beetle antennae search algorithm (BAS) is a metaheuristic algorithm that is inspired by the searching behavior of longhorn beetles [35, 36]. A vector x^t at *tth* time instant $(t = 1, 2, \ldots)$ denotes the position of the beetle. At position x, $f(x)$ is defined as the fitness function to represent the concentration of odour. Normally, we can use two rules to simplify the algorithm, including search and detection behavior. In an unknown environment, the beetle searches randomly. A normalized random unit vector $\overrightarrow{b}$ is utilized to model the searching behavior as follows:

$$\overrightarrow{b} = \frac{\text{rands}(m, 1)}{\|\text{rands}(m, 1)\|}, \tag{31}$$

where rands(.) represents a random function and m is the number of dimensions of the position. x_r and x_1 are defined to imitate the searching behavior of both the right and left antennae:

$$\begin{aligned} x_r &= x^t + d^t\overrightarrow{b}, \\ x_l &= x^t + d^t\overrightarrow{b}, \end{aligned} \tag{32}$$

where d is the sensing length of antennae.

An iterative form is proposed to model the detecting behavior:

$$x^{t+1} = x^t + \delta^t\overrightarrow{b}\,\text{sign}\left(f(x_r) - f(x_1)\right), \tag{33}$$

where δ is the step size and sign(.) represents a sign function.

The antennae length d and step size δ are updated according to the following formula.

$$\begin{aligned} d^t &= 0.95d^{t-1} + 0.01, \\ \delta^t &= 0.95\delta^{t-1}. \end{aligned} \tag{34}$$

Compared with other heuristic algorithms such as PSO, GA, and ABC, the BAS algorithm possesses better performance in terms of faster speed and simpler implementation. A second order Michalewicz function shown below was used to validate the algorithm.

$$f(x) = \sin(x_1)\left[\sin\left(\frac{x_1^2}{\pi}\right)\right]^{20} + \sin(x_2)\left[\sin\left(\frac{2x_2^2}{\pi}\right)\right]^{20}, \tag{35}$$

where the minimized value satisfies $f(x^*) = -1.8013$, locating in $(x^*) = (2.20, 1.57)$. The initial values of d and δ are 2 and 0.5. Taking the absolute error less than 0.0001 as the end evolution, the time these algorithms mentioned above takes is shown in Table 2.

The results showed that the search speed of BAS algorithm was tremendous. The BAS algorithm can fit well for the real-time task.

Table 2: Time comparisons of different algorithms in a two-dimensional situation.

Algorithms	BAS	PSO	GA	ABC
T (s)	0.087843	0.215134	0.262317	0.132563

5.2. Beetle Swarm Optimization (BSO). The performance of beetle antennae search algorithm relies heavily on the initial position, and it will easily fall into the local optimum when dealing with high-dimensional problems [37–39]. In order to overcome these drawbacks, further improvements are made by combining the antennae search algorithm and particle swarm optimization. That is the beetle swarm optimization algorithm.

In beetle swarm optimization, each particle in the standard PSO is defined as a beetle. Each beetle represents a potential solution to the optimization problem. In the iterative process, the update of the position relies not only on the individual and global historical best solution but also on the way of beetle antennae search. The formula can be expressed as follows [34]:

$$X_{is}^{k+1} = X_{is}^{k} + \lambda V_{is}^{k} + (1-\lambda)\xi_{is}^{k}, \tag{36}$$

where i is the *ith* beetle, s is the *sth* dimension, k is the current number of iterations, V_{is} is the speed of the beetle, ξ_{is} is the increase in movement, and λ is a positive constant. V_{is} is expressed as:

$$V_{is}^{k+1} = \omega V_{is}^{k} + c_1 \text{rand}_1\left(p_{is}^{k} - X_{is}^{k}\right) + c_2 \text{rand}_2\left(p_{gs}^{k}\right), \tag{37}$$

where w is the inertia weight, c_1 and c_2 are constants, and rand_1 and rand_2 are two random functions in (0, 1). P_i and P_g are the individual and group extremities.

The inertia weight is decreasing in the process as follows:

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{K} * k, \tag{38}$$

where $\omega_{\max}$ and $\omega_{\min}$ represent the maximum and minimum values of ω and k and K are the current number of iterations and the maximum number of iterations.

ξ defines the increase in movement and is expressed as

$$\xi_{is}^{k+1} = \delta^{k*} V_{is}^{k*} \operatorname{sign}\left(f\left(X_{rs}^{k}\right) - f\left(X_{rs}^{k}\right)\right), \tag{39}$$

where δ is the step size. The position of the left and right antenna can be expressed as

$$\begin{aligned} X_{rs}^{k+1} &= X_{rs}^{k} + V_{iS}^{k} * \frac{d}{2}, \\ X_{ls}^{k+1} &= X_{ls}^{k} + V_{is}^{k} * \frac{d}{2}. \end{aligned} \tag{40}$$

5.3. MLPNN. Artificial neural network and, in particular, the multilayer perceptron neural networks are widely used in many application areas over the years. Typically, the multilayer perceptron neural network contains three kinds of layers, which are the input layer, hidden layer, and output layer. It is claimed that a three-layered feed-forward neural network can approximate any nonlinear function with arbitrary accuracy. The number of neurons in input and output layers is normally selected according to the actual needs. However, the best number of neurons in the hidden layer can just be determined by trial-and-error methods.

Generally, each neuron can be denoted by the following equation:

$$y_j = f\left(\sum_{i=1}^{n} \omega_{ij} x_i + b_j\right), \tag{41}$$

where x_i and y_j are the input and output values of the *jth* neuron, respectively; ω_{ij} denotes the connection weight from the *ith* neuron in the previous layer to the *jth* neuron in the latter layer; b_j represents bias value in the *jth* neuron; and f is normally a sigmoid function as follows:

$$f(x) = \frac{1}{1+e^{-x}}. \tag{42}$$

Its derivation is

$$f'(x) = f(x)(1 - f(x)). \tag{43}$$

The error of the *kth* neuron in the output layer is

$$e_k = Y_k - 0_k, \tag{44}$$

where Y_k and O_K are the desired and actual value of the *ith* neuron in the output layer. The total error of the output layer is

$$E = \frac{1}{2}\sum_{k=1}^{m} e_k^2, \tag{45}$$

where m is the number of neurons in the output layer.

For nongeometric errors compensation, the joint angles are selected as the inputs and the positioning errors in each axis are selected as the outputs. By using the Jacobian matrix, the positioning errors are mapped to the errors of joint angles. The compensation process is shown in Figure 2.

5.4. The MLPNN Optimized by BSO. The validity and accuracy of MLFNN may be reduced if the weights and biases are improperly selected. The BSO algorithm is used to optimize the network parameters of MLFNN in this process. In this section, the BSO-MLFNN method is proposed to overcome the shortcomings of low accuracy. The process is shown in Figure 3.

6. Compensation Verification and Analysis

As shown in Figure 4, a spherical mounted retroreflector (SMR) was mounted at the end flange of the manipulator, and the position of the SMR, namely, the tool center point (TCP) position, was measured by a laser tracker (FARO

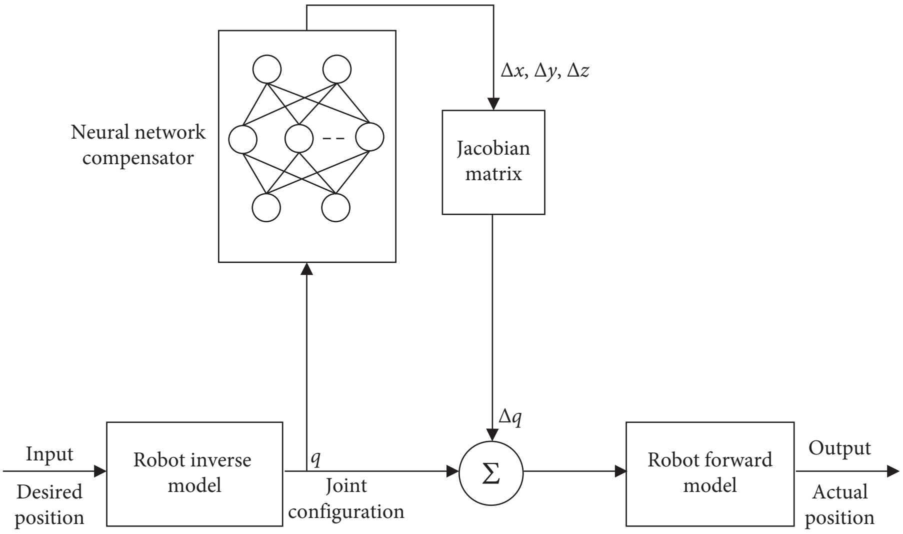


Figure 2: NN-based nongeometric errors compensation.

Vantage) with the positioning accuracy of $10\,\mu\text{m} + 2.5\,\mu\text{m/m}$. Experiments were performed on a heavy load SIASUN SR210D robot.

The main steps of the experiment were as follows: (1) a set of 50 robot joint angle values in the controller and the corresponding positions without any load were obtained in the measuring coordinate system. Geometric parameters and transmission errors were identified using the improved POE model. (2) Identify the joint stiffness by measuring 50 points with a 210 kg load after geometric parameters and transmission errors compensation. (3) Compensate the nongeometric errors based on the proposed BSO-MLPNN by measuring 200 points after the two steps mentioned above.

The measurement points should be selected evenly in the workspace, so that the measuring configurations should go through all controllable DoFs and close to the boundary of the workspace, where the maximum errors are most prominent.

6.1. Geometric Parameters and Transmission Errors Identification. According to equation (22), the actual geometric parameters, transmission errors and coupling ratio can be identified. The nominal and identified value of geometric parameters and reduction ratio are shown in Tables 3 and 4.

The nominal coupling ratio error is 0 and the identified coupling ratio error is -2.5342×10^{-4}. The positioning accuracy of the 50 points before and after calibration is shown in Figure 5. The average positioning accuracy of measurement points is enhanced from 1.1288 mm to 0.2898 mm. For validation, another set of 50 points is selected. The validation results show that the average positioning accuracy is 0.3008 mm which is 0.011 mm larger than 0.2898 mm. Details are presented in Table 5.

6.2. Stiffness Identification and Compensation. According to equation (28), the joint stiffness is identified. The stiffness results are shown in Table 6. The positioning accuracy of the 50 points before and after calibration is shown in Figure 6. The average positioning accuracy is enhanced from 7.0261 mm to 0.9847 mm. Details are presented in Table 7.

6.3. Nongeometric Errors Compensation Based on BSO-MLPNN. Figure 6 shows that the positioning errors are still large after the geometric parameters, transmission errors, and the joint stiffness compensation. These errors caused by nongeometric sources are compensated by the BSO-MLPNN method. For this purpose, 200 evenly distributed points in the workspace and corresponding joint angles are collected for training the BSO-MLPNN. In the training process, joint angles are selected as the inputs and the residual positioning errors are selected as the outputs. The residual positioning errors in each axis are shown in Figure 7.

The positioning accuracy before and after nongeometric calibration at the measurement points is shown in Figure 8. The results show that, after the nongeometric errors compensation at the measurement points, the average positioning accuracy is enhanced from 0.9262 (mm) to 0.5907 (mm). Details are shown in Table 8.

Normally, the positioning accuracy of the measurement points is higher than that of the validation points. For validation, another set of 200 points are selected. The positioning errors are shown in Figure 9. The validation results

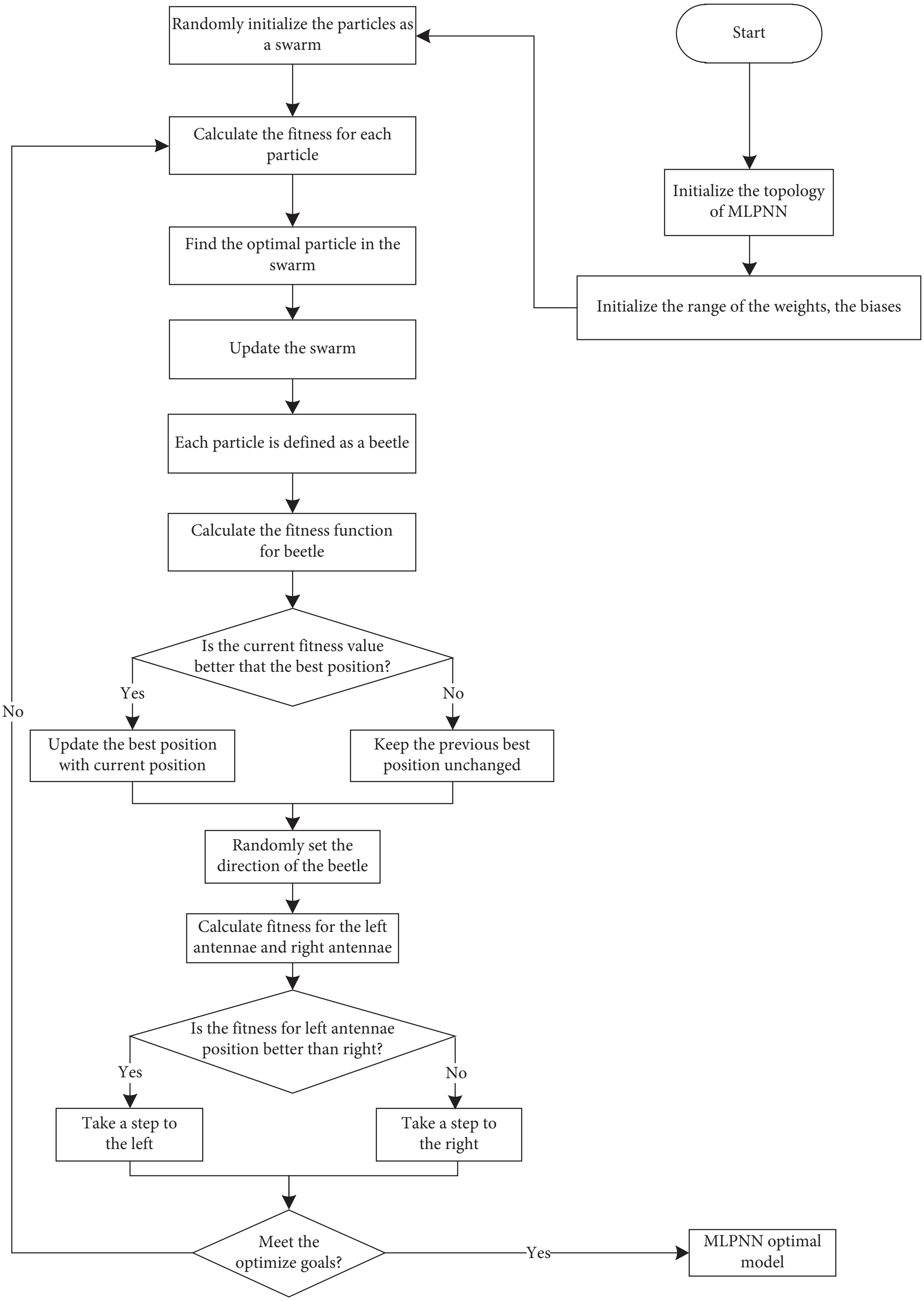

Randomly initialize the particles as a swarm
Start
Calculate the fitness for each particle
Initialize the topology of MLPNN
Find the optimal particle in the swarm
Initialize the range of the weights, the biases
Update the swarm
Each particle is defined as a beetle
Calculate the fitness function for beetle
Is the current fitness value better that the best position?
Yes
No
No
Update the best position with current position
Keep the previous best position unchanged
Randomly set the direction of the beetle
Calculate fitness for the left antennae and right antennae
Is the fitness for left antennae position better than right?
Yes
No
Take a step to the left
Take a step to the right
Meet the optimize goals?
Yes
MLPNN optimal model

Figure 3: Algorithm flow chart of BSO-MLPNN.

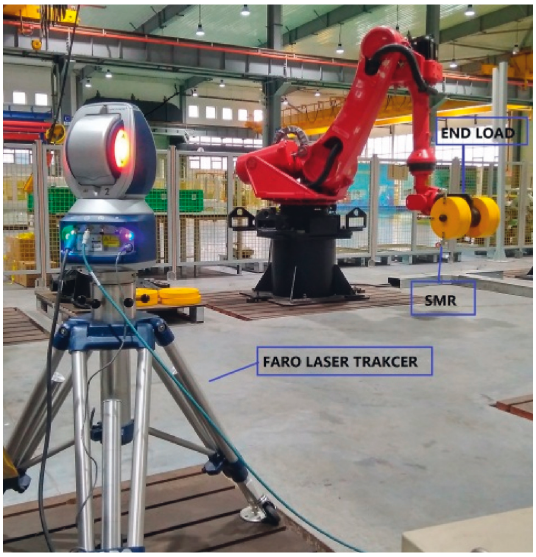


Figure 4: Experimental scene.

Table 3: Nominal value and identified value of robot geometric parameters.

			Nominal parameters			
L_i (mm), $i = 1–5$	312	1075	235	1282	260	0
θ_{i0} (°), $i = 1–6$	0	0	0	0	0	0
			Identified parameters			
L_i' (mm), $i = 1–5$	313.1011	1075.1209	234.2737	1281.3124	260	0
θ_{i0}' (°), $i = 1–6$	0	0.0044	0.0479	0.4524	−0.1892	0

Table 4: Nominal and identified value of reduction ratio errors.

			Nominal reduction ratio errors			
k_i, $i = 1–6$	1	1	1	1	1	1
			Identified reduction ratio errors			
k_i', $i = 1–6$	1.0002	1.0013	1.0004	0.9998	0.9996	1.0003

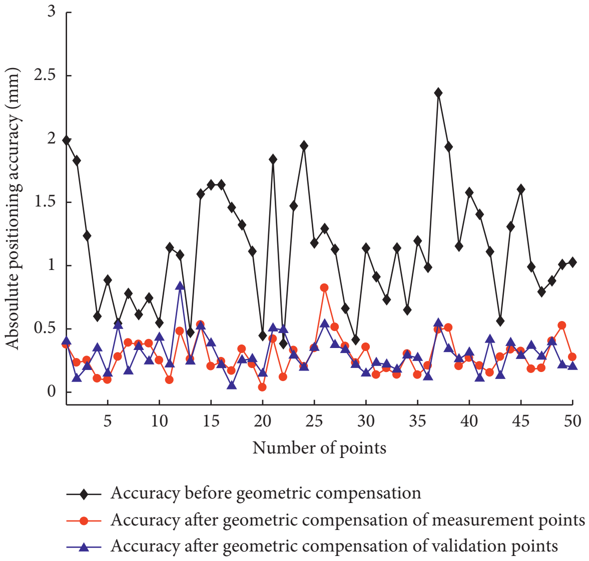


Figure 5: Positioning accuracy before and after geometric calibration.

Table 5: Absolute positioning accuracy of the SIASUN SR210D robot.

	Mean (mm)	Standard deviation (mm)	Maximum (mm)
Before calibration	1.1288	0.4709	2.3640
After robot link geometry and transmission errors compensation of measurement points	0.2898	0.1464	0.8226
After robot link geometry and transmission errors compensation of validation points	0.3008	0.1532	0.8436

Table 6: Flexibility factors for stiffness calibration identification.

c_{ii}, $i = 1\text{–}6$	0	$1.89e-5$	$-2.49e-5$	$-5.35e-5$	$-8.26e-5$	0

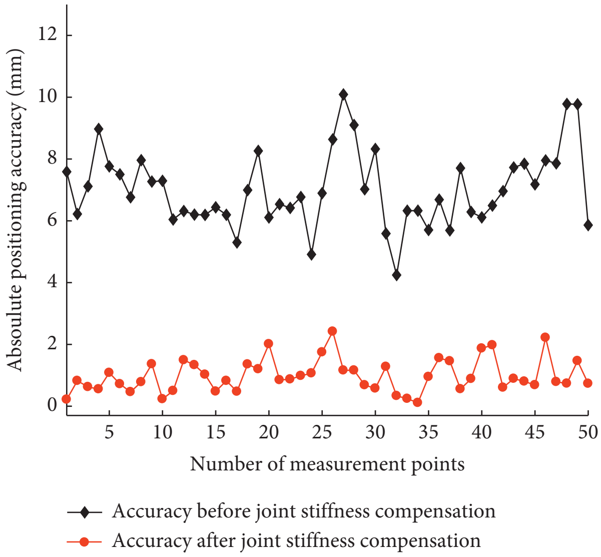


Figure 6: Positioning accuracy after the calibration of stiffness parameters under the full load condition.

Table 7: Absolute positioning accuracy of the SIASUN SR210D manipulator.

	Mean (mm)	Standard deviation (mm)	Maximum (mm)
Before joint stiffness compensation	7.0261	1.2351	10.0898
After joint stiffness compensation	0.9847	0.5340	2.4176

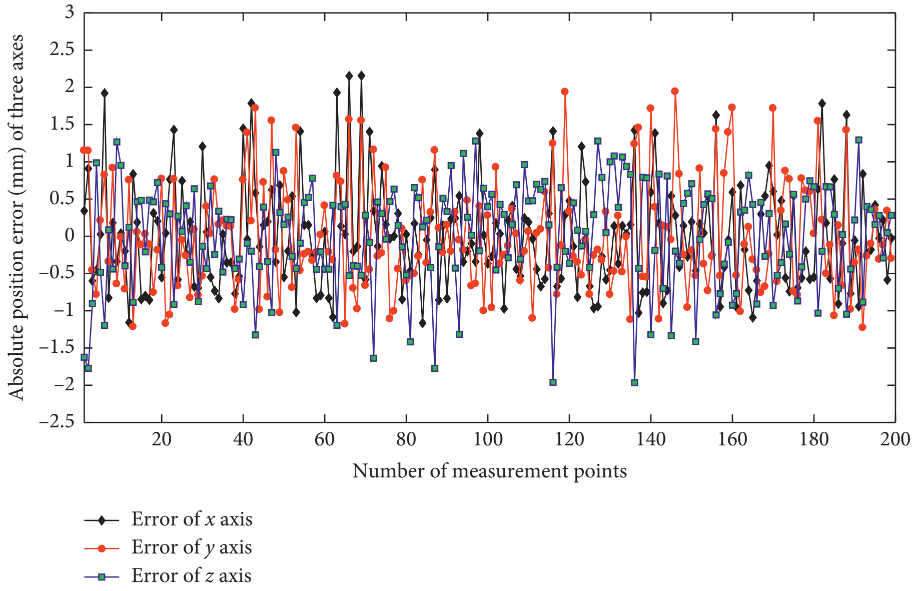


Figure 7: Residual positioning errors in each axis.

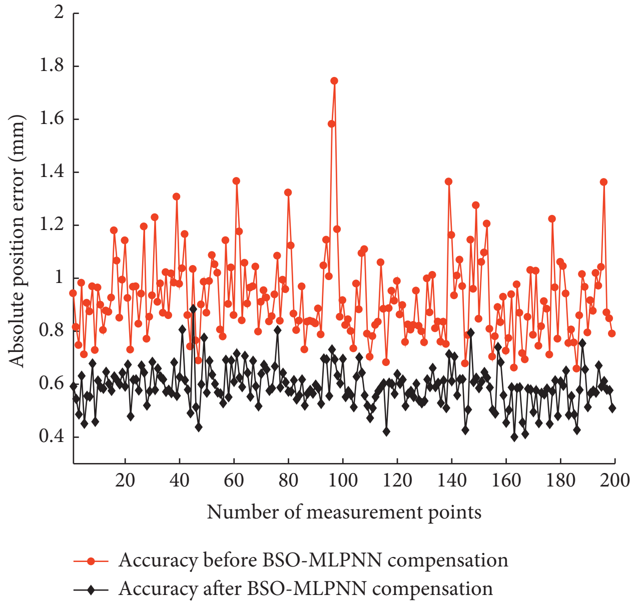


Figure 8: Absolute positioning accuracy before and after nongeometric compensation of measurement points.

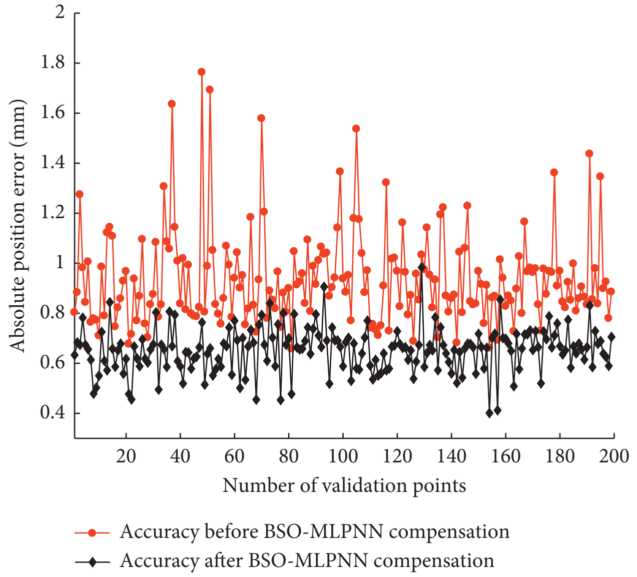


Figure 9: Absolute positioning accuracy before and after nongeometric errors compensation of validation points.

Table 8: Absolute positioning accuracy of the SIASUN SR210D manipulator of measurement points.

	Mean (mm)	Standard deviation (mm)	Maximum (mm)
Before BSO-MLPNN calibration	0.9262	0.1632	1.7435
After BSO-MLPNN calibration	0.5907	0.0760	0.8834

show that the average positioning accuracy is enhanced from 0.9370 mm to 0.6522 mm. Details are shown in Table 9.

Therefore, the proposed three stages calibration method including geometric parameters, transmission errors, joint stiffness, and nongeometric errors compensation is effective.

7. Conclusion and Future Works

This paper proposed a three-stage calibration method for enhancing robot positioning accuracy. The geometric parameters and transmission errors are identified in the first stage. Then, joint stiffness compensation is carried out. For

Table 9: Absolute positioning accuracy of the SIASUN SR210D manipulator of validation points.

	Mean (mm)	Standard deviation (mm)	Maximum (mm)
Before BSO-MLPNN calibration	0.9370	0.1871	1.7633
After BSO-MLPNN calibration	0.6522	0.0887	0.9834

the nongeometric errors compensation, a hybrid algorithm, namely, BSO-MLPNN, is introduced.

Experiments were performed on the SIASUN SR210D robot manipulator. The average positioning accuracy was enhanced from 7.0261 mm to 0.5907 mm. Results demonstrated the effectiveness and correctness of the proposed method. In addition, the validation results showed that the manipulator after calibration has the same level of positioning accuracy in the entire workspace.

In future, nongeometric factors will be modeled in order to obtain more accurate knowledge of error sources and accelerate the convergence rate.

Data Availability

The authors declare that all relevant data within the paper are fully available without restriction.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was supported by the 2017 National Key R&D Program of China (no. 2017YFB1301400).

References

[1] G. Chen, T. Li, M. Chu et al., "Review on kinematics calibration technology of serial robots," *International Journal of Precision Engineering and Manufacturing*, vol. 15, pp. 1759–1774, 2014.

[2] R. P. Judd and A. B. Knasinski, "A technique to calibrate industrial robots with experimental verification," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 1, pp. 20–30, 1990.

[3] T. Li, K. Sun, Y. Jin, and H. Liu, "A novel optimal calibration algorithm on a dexterous 6 DOF serial robot-with the optimization of measurement poses number," *IEEE International Conference on Robotics and Automation*, vol. 10, pp. 975–981, 2011.

[4] G. Chen, Q. Jia, T. Li, and H. Sun, "Calibration method and experiments of robot kinematics parameters based on error model," *Robot*, vol. 34, no. 6, pp. 680–688, 2012.

[5] I.-W. Park, B.-J. Lee, S.-H. Cho, Y.-D. Hong, and J.-H. Kim, "Laser-based kinematic calibration of robot manipulator using differential kinematics," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 6, pp. 1059–1067, 2012.

[6] T. Sun, P. Wang, B. Lian, S. Liu, and Y. Zhai, "Geometric accuracy design and error compensation of a one-translational and three-rotational parallel mechanism with articulated traveling plate," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 232, no. 12, pp. 2083–2097, 2018.

[7] Y. Song, J. Zhang, B. Lian, and T. Sun, "Kinematic calibration of a 5-DoF parallel kinematic machine," *Precision Engineering*, vol. 45, pp. 242–261, 2016.

[8] T. Sun, Y. Zhai, Y. Song, and J. Zhang, "Kinematic calibration of a 3-DoF rotational parallel manipulator using laser tracker," *Robotics and Computer-Integrated Manufacturing*, vol. 41, pp. 78–91, 2016.

[9] T. Sun, B. Lian, J. Zhang, and Y. Song, "Kinematic calibration of a 2-DoF over-constrained parallel mechanism using real inverse kinematics," *IEEE Access*, vol. 6, pp. 67752–67761, 2018.

[10] S. A. Hayati, "Robot arm geometric link parameter estimation," in *Proceedings of the 22th IEEE Conference on Decision Control*, pp. 1477–1483, San Antonio, TX, USA, December 1983.

[11] R. W. Brockett, "Robotic manipulators and the product of exponentials formula," *Mathematical Theory of Networks and Systems*, vol. 58, pp. 120–129, 1984.

[12] X. Yang, "Kinematics calibration research based on the positioning error of the 6-DOF industrial robot," in *Proceedings of the 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems*, pp. 913–917, Shenyang, China, June 2015.

[13] X. Zhong, J. Lewis, and F. L. N-Nagy, "Inverse robot calibration using artificial neural networks," *Engineering Applications of Artificial Intelligence*, vol. 9, no. 1, pp. 83–93, 1996.

[14] D. Wang and J. Chi, "Survey on robot kinematics calibration," *Application Research of Computers*, vol. 24, no. 9, pp. 8–11, 2007, in Chinese.

[15] C. Gong, J. Yuan, and J. Ni, "Nongeometric error identification and compensation for robotic system by inverse calibration," *International Journal of Machine Tools and Manufacture*, vol. 40, no. 14, pp. 2119–2137, 2000.

[16] Z. Jian, H. N. Nguyen, and H. J. Kang, "Simultaneous identification of joint compliance and kinematic parameters of industrial robots," *International Journal of Precision Engineering and Manufacturing*, vol. 15, pp. 2257–2264, 2014.

[17] G. Duelen and K. Schröer, "Robot calibration-Method and results," *Robotics and Computer-Integrated Manufacturing*, vol. 8, no. 4, pp. 223–231, 1991.

[18] J. C. Hudgens, E. Hernandez, and D. Tesar, "A compliance parameter estimation method for serial manipulator DSC," *Applications of modeling and identification to improve machine performance*, vol. 29, pp. 15–23, 1991.

[19] K. Wang, "Application of genetic algorithms to robot kinematics calibration," *International Journal of Systems Science*, vol. 40, no. 2, pp. 147–153, 2009.

[20] Y. Liu, B. Liang, C. Li, L. Xue, S. Hu, and Y. Jiang, "Calibration of a stewart parallel robot using genetic algorithm," in *Proceedings of the International Conference on Mechatronics and Automation*, pp. 2495–2500, Harbin, China, August 2007.

[21] J.-M. Renders, E. Rossignol, M. Becquet, and R. Hanus, "Kinematic calibration and geometrical parameter identification for robots," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 6, pp. 721–732, 1991.

[22] H. Naruse, A. Nobiki, T. Yabuta, and M. Tateda, "High-accuracy multiviewpoint stereo measurement using the maximum-likelihood method," *IEEE Transactions on Industrial Electronics*, vol. 44, no. 4, pp. 571–578, 1997.

[23] Z. Wang, “A robot calibration method based on joint angle division and an artificial neural network,” *Mathematical Problems in Engineering*, vol. 2019, p. 12, 2019.

[24] G. Gao, “Hybrid optimal kinematic parameter identification for an industrial robot based on BPNN-PSO,” *Complexity*, vol. 2018, p. 11, 2018.

[25] D. Wang, Y. Bai, and J. Zhao, “Robot manipulator calibration using neural network and a camera-based measurement system,” *Transactions of the Institute of Measurement and Control*, vol. 34, no. 1, pp. 105–121, 2012.

[26] N. Takanashi, “6 DOF manipulators absolute positioning accuracy improvement using a neural-network,,” in *Proceedings of IEEE International Workshop on Intelligent Robots and Systems, towards a New Frontier of Applications*, pp. 635–640, Ibaraki, Japan, July 1990.

[27] S. Aoyagi, A. Kohama, Y. Nakata et al., “Improvement of robot accuracy by calibrating kinematic model using a laser tracking system-compensation of non-geometric errors using neural networks and selection of optimal measuring points using genetic algorithm,” in *Proceedings of 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5660–5665, Taipei, Taiwan, November 2010.

[28] J. H. Jang, S. H. Kim, and Y. K. Kwak, “Calibration of geometric and non-geometric errors of an industrial robot,” *Robotica*, vol. 19, no. 3, pp. 311–321, 2001.

[29] H.-N. Nguyen, “A new calibration method for enhancing robot positioning accuracy by combining a robot model–based identification approach and an artificial neural network–based error compensation technique,” *Advances in Mechanical Engineering*, vol. 11, no. 1, 2018.

[30] X. Xie, Z. Li, and G. Wang, “Manipulator calibration based on PSO-RBF neural network error model,” *AIP Conference Proceedings*, vol. 2073, no. 1, 2019.

[31] J. Li, J.-H. Cheng, J.-Y. Shi, and F. Huang, “Brief introduction of back propagation (BP) neural network algorithm and its improvement,” *Advances in Intelligent and Soft Computing*, vol. 169, pp. 553–558, 2012.

[32] D. S. Wang and X. H. Xu, “Genetic neural network and application in welding robot error compensation,” *Proceedings of International Conference on Machine Learning and Cybernetics*, vol. 10, pp. 4070–4075, 2005.

[33] H. Yao and Z. Zhang, “Research of camera calibration based on genetic algorithm bp neural network,” in *Proceedings of the 2016 IEEE International Conference on Information and Automation (ICIA)*, pp. 350–355, Ningbo, China, August 2016.

[34] G. Jiang, M. Luo, K. Bai, and S. Chen, “A precise positioning method for a puncture robot based on a PSO-optimized BP neural network algorithm,” *Applied Sciences*, vol. 7, no. 10, 2017.

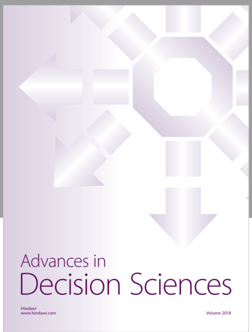
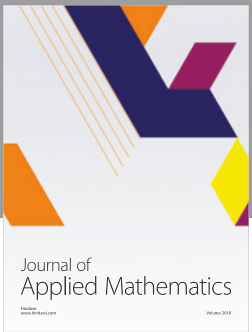
[35] X. Jiang and S. Li, “BAS: beetle antennae search algorithm for optimization problems,” 2017, http://arxiv.org/abs/1710.10724.

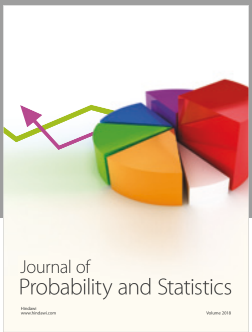
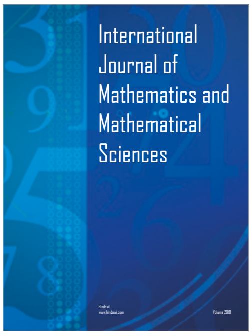
[36] X. Jiang and S. Li, “Beetle antennae search without parameter tuning (bas-wpt) for multi-objective optimization,” 2017, http://arxiv.org/abs/1711.02395.

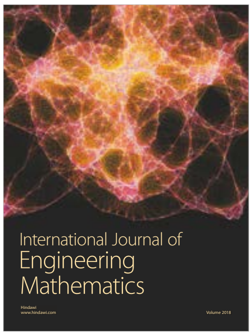
[37] M. Lin and Q. Li, “A hybrid optimization method of beetle antennae search algorithm and particle swarm optimization,” *International Conference on Robotics, Control and Automation*, vol. 10, pp. 396–401, 2018.

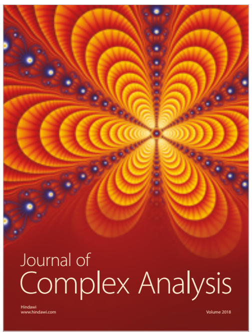
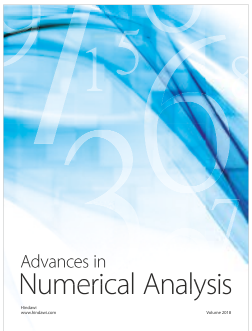
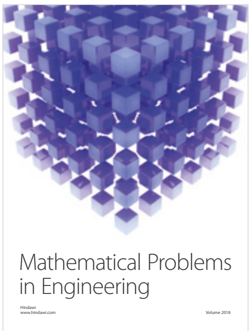
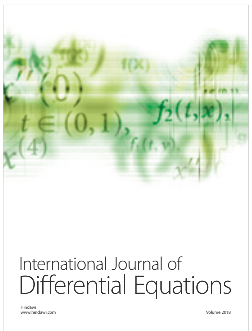
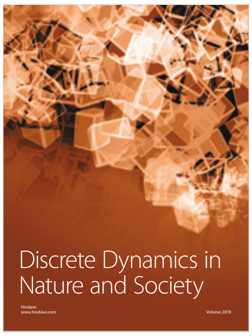
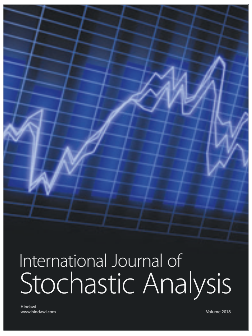
[38] T. Wang, L. Yang, and Q. Liu, “Beetle swarm optimization algorithm: theory and application,” 2018, http://arxiv.org/abs/1808.00206.

[39] T. Chen, Y. Zhu, and J. Teng, “Beetle swarm optimisation for solving investment portfolio problems,” *The Journal of Engineering*, vol. 2018, no. 16, pp. 1600–1605, 2018.

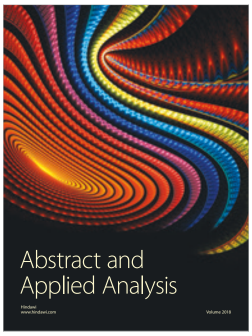
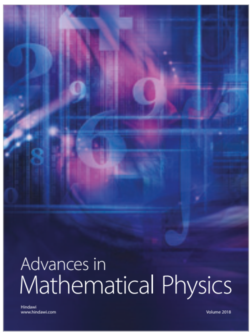
Advances in
Operations Research

Advances in
Decision Sciences

Journal of
Applied Mathematics

The Scientific
World Journal

Journal of
Probability and Statistics

International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Optimization

International Journal of
Engineering
Mathematics

International Journal of
Analysis

Journal of
Complex Analysis

Advances in
Numerical Analysis

Mathematical Problems
in Engineering

International Journal of
Differential Equations

Discrete Dynamics in
Nature and Society

International Journal of
Stochastic Analysis

Journal of
Mathematics

Journal of
Function Spaces

Abstract and
Applied Analysis

Advances in
Mathematical Physics