

## Research Article

# Hybrid Optimal Kinematic Parameter Identification for an Industrial Robot Based on BPNN-PSO

Guanbin Gao <sup>1</sup>, Fei Liu,<sup>1</sup> Hongjun San <sup>1</sup>, Xing Wu,<sup>1</sup> and Wen Wang <sup>2</sup>

<sup>1</sup>Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming, China

<sup>2</sup>School of Mechanical Engineering, Hangzhou Dianzi University, Hangzhou 310018, China

Correspondence should be addressed to Hongjun San; sanhjun@163.com

Received 25 March 2018; Accepted 16 May 2018; Published 8 July 2018

Academic Editor: Wenbo Wang

Copyright © 2018 Guanbin Gao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A novel hybrid algorithm that employs BP neural network (BPNN) and particle swarm optimization (PSO) algorithm is proposed for the kinematic parameter identification of industrial robots with an enhanced convergence response. The error model of the industrial robot is established based on a modified Denavit-Hartenberg method and Jacobian matrix. Then, the kinematic parameter identification of the industrial robot is transformed to a nonlinear optimization in which the unknown kinematic parameters are taken as optimal variables. A hybrid algorithm based on a BPNN and the PSO is applied to search for the optimal variables which are used to compensate for the error of the kinematic parameters and improve the positioning accuracy of the industrial robot. Simulations and experiments based on a realistic industrial robot are all provided to validate the efficacy of the proposed hybrid identification algorithm. The results show that the proposed parameter-identification method based on the BPNN and PSO has fewer iterations and faster convergence speed than the standard PSO algorithm.

## 1. Introduction

The nominal parameters of industrial robots for the mechanical design are usually not accurate due to manufacturing and assembly errors, limited precision of components, flexible deformation of linkages and joints and so on, which will lead to the decrease of the positional accuracy of industrial robots in practical applications. Kinematic calibration is an effective way to improve the accuracy of industrial robots, and parameter identification is a key step of calibration [1]. Hence, many research works have been focusing on this area. Parameter identifications are usually realized through minimizing the residuals of the end-effectors' poses of industrial robots. It is a nonlinear or standard linear least-squares optimization process. As a commonly used algorithm, the least-squares method [2, 3] does not need to consider any prior information of the system, but its low computability and the noise sensitivity limit its application [4, 5]. The extended Kalman filter [6, 7] is a useful method for dealing with nonlinear problems, which is possible to realize the state estimation under some mild conditions on the measuring

error. However, the actual distribution of the positioning errors is not taken into account in the above work, resulting in a situation where the state estimate is not accurate enough and the filter is divergent. The Levenberg–Marquardt algorithm [8, 9] is used to solve nonlinear least-squares problems; however, it generally can only find the local minimum, which is not necessarily for the global minimum. Daney et al. [10] proposed an algorithm based on a constrained optimization method to select a set of measurement configurations in the calibration of robots. Jiang et al. [1] proposed a hybrid kinematic calibration method based on the extended Kalman filter and particle filter algorithm that can significantly improve the positioning accuracy of the robot. Xiong et al. [11] presented a systematic and practical calibration method based on the global product-of-exponential formula considering some practical constraints for an industrial robot to improve its absolute accuracy, in which all the kinematic parameters are identified via the linear least-squared iteration.

In recent years, many intelligent bionic algorithms have been used in parameter identification. Gong et al. [12]

proposed a new hybrid optimization algorithm based on the bee swarm particle swarm optimization algorithm to obtain the optimum structural parameters of a manipulator. Fan et al. [13] conducted the parameter identification of a parallel mechanism based on genetic algorithm. Wang et al. [14] proposed a universal index and an improved PSO algorithm for optimal pose selection. Shi et al. [15] proposed a quantum particle swarm optimization (QPSO) algorithm based on the path-planning method, so that the base position and the end position can simultaneously reach the desired state. Fang and Dang [16] proposed a method based on the QPSO algorithm, which is suitable for the kinematic calibration of both serial and parallel robots.

As an evolutionary algorithm, PSO starts from a random solution and searches the optimal solution through iteration. However, it needs much iteration in dealing with parameter identification of industrial robots since there are more than 20 dimensions in the optimization model. BPNN can improve the convergence ability of PSO [17, 18]. Inspired by this fact, this paper proposes a kinematic parameter-identification method based on BPNN-PSO, which can greatly improve the convergence speed of the PSO algorithm. To maintain the positioning accuracy and repeatability, industrial robots are required to be calibrated regularly, especially after collisions and overload operations. Thus, the proposed method can improve the efficiency of identification greatly in the follow-up calibration of industrial robots.

This paper is organized as follows. Section 2 presents the kinematic modelling of the industrial robots with MDH model. In Section 3, the kinematic identification of the structural parameter is formulated as a nonlinear optimization problem. Simulations and experiments are conducted to verify the identification model and search for the optimal settings of PSO and BPNN in Section 4 and Section 5, respectively. Section 6 provides the conclusion.

## 2. Kinematic Modeling of the Industrial Robot

ER20-C10 is a kind of universal industrial robot, which is a six-degree-of-freedom (6-DOF) joint-type industrial robot, as shown Figure 1. According to the D-H modified method [19], the coordinate systems for each joint of the robot are built, as shown in Figure 2. There are a base coordinate system ( $F_0$ ) and six joint coordinate systems ( $F_1, F_2, \dots, F_6$ ) in the coordinate systems, where  $F_6$  is the end-flange coordinate system. A laser tracker shown in Figure 1 will be used to acquire the end position data of the robot in experimental verification, which is a portable, highly accurate coordinate measuring system with an ADM (absolute distance measurement) accuracy of  $\pm 10 \mu\text{m}$ . An active target (AT) is installed at the end of the robot as the end effector, which can assure that the tracker will not lose the laser in the process of measurement. In addition, the tool coordinate system  $F_{\text{tool}}$  and the world coordinate system  $F_{\text{World}}$  must also be considered.  $F_{\text{World}}$  is set at the measurement coordinate system of the laser tracker.  $F_{\text{tool}}$  is set at the center of the active target (AT) mounted on the flange, and its direction is the same as the end-flange coordinate system  $F_6$ .



FIGURE 1: Experimental setup with ER20-C10 robot and Radian™ laser tracker.

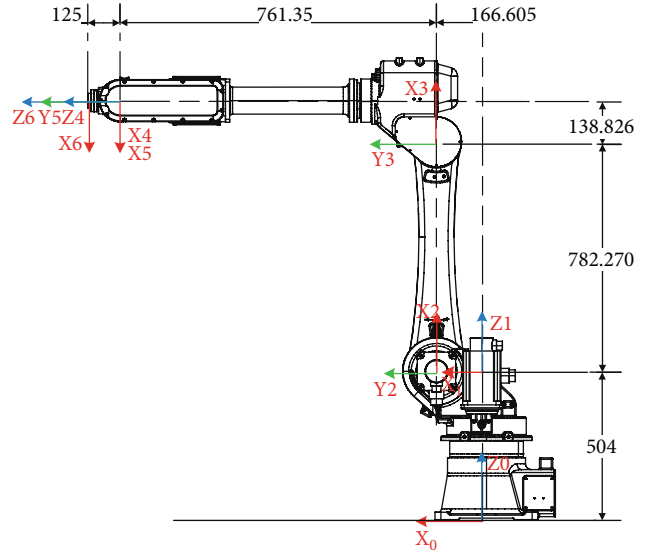


FIGURE 2: The coordinate systems of the robot.

**2.1. Kinematics and Error Identification Modelling.** Given the joint variable vector  $q = [\theta_1, \theta_2, \dots, \theta_6]$ , the end-effector's pose is represented as follows:

$$H(q) = T_{\text{world}}^0 T_0^6 T_6^{\text{tool}}, \quad (1)$$

where  $T_i^j$  is the homogeneous matrix representing the pose of frame  $F_j$  with respect to  $F_i$ . These three homogeneous matrices are calculated as follows:

$$T_0^{\text{world}} = \begin{bmatrix} R_0 & t_0 \\ 0 & 1 \end{bmatrix}, \quad (2)$$

$$T_6^{\text{tool}} = \begin{bmatrix} 1 & 0 & 0 & x_t \\ 0 & 1 & 0 & y_t \\ 0 & 0 & 1 & z_t \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$T_0^6 = T_0^1 T_1^2 T_2^3 T_3^4 T_4^5 T_5^6,$$

where  $R_0$  is the rotation matrix representing the orientation of the base frame with respect to the world frame and  $t_0$  is

TABLE 1: Nominal values of DH kinematic parameters of the industrial robot.

Number of linkage	$a_{i-1}$ (mm)	$d_i$ (mm)	$\theta_i$ (°)	$\alpha_{i-1}$ (°)
1	0	504	$\theta_1$	0
2	166.605	0	$\theta_2$	-90
3	782.270	0	$\theta_3$	0
4	138.826	761.35	$\theta_4$	-90
5	0	0	$\theta_5$	90
6	0	125	$\theta_6$	-90

the position of the origin of the base frame with respect to the world frame.  $x_t$ ,  $y_t$ , and  $z_t$  are the position of the tool frame with respect to the end-flange frame. For  $i = 1, 2, \dots, 6$ , the homogeneous matrix of each successive pair of frames is obtained using the MD-H parameters as follows:

$$T_{i-1}^i = R_X(\alpha_{i-1})D_X(a_{i-1})R_Z(\theta_i)D_Z(d_i)$$

$$= \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -d_i s\alpha_{i-1} \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & d_i c\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

where  $\alpha_{i-1}$ ,  $a_{i-1}$ , and  $d_i$  are the MD-H parameters,  $s\theta_i = \sin \theta_i$ ,  $c\theta_i = \cos \theta_i$ ,  $s\alpha_i = \sin \alpha_i$ , and  $c\alpha_i = \cos \alpha_i$ . The nominal values of the industrial robot are shown in Table 1.

## 2.2. Preliminary Identification of the Base Frame and the Tool Frame

**2.2.1. Preliminary Identification of the Base Frame.** As we know, the location of the base frame of a robot can be measured directly if the robot is properly mounted. But unfortunately, the location of the base frame in the robot is difficult to measure with instruments directly; hence, we will get the location through preliminary identification. Firstly, we define the base frame  $F_0$  with respect to frame  $F_1$  [20]. The steps of defining the base frame  $F_0$  can be described as follows.

- (1) One keeps the robot at zero position (the value of each joint angle  $\theta_i$  is 0).
- (2) One keeps joints 2–6 unchanged and joint 1 rotating; the position of the AT is measured at a certain angle interval using a laser tracker. Based on these positions measured, we can get circle 1.
- (3) Similar to step 1, one rotates joint 2 and we can get circle 2.
- (4) According to the normals of circle 1 and circle 2, their common vertical line can be obtained.
- (5) The intersection of the common vertical line and the normal of circle 1 is set as the origin of the frame  $F_1$ .

TABLE 2: The complete kinematic parameters of the industrial robot.

Number of linkage	$a_{i-1}$ (mm)	$d_i$ (mm)	$\theta_i$ (°)	$\alpha_{i-1}$ (°)
1	$\Delta a_0$	$504 + \Delta d_1$	$\theta_{1,c} + \Delta\theta_{1,0}$	$0 + \Delta\alpha_0$
2	$166.605 + \Delta a_1$	$\Delta d_2$	$\theta_{2,c} + \Delta\theta_{2,0}$	$-90 + \Delta\alpha_1$
3	$782.270 + \Delta a_2$	$\Delta d_3$	$\theta_{3,c} + \Delta\theta_{3,0}$	$\Delta\alpha_2$
4	$138.826 + \Delta a_3$	$761.35 + \Delta d_4$	$\theta_{4,c} + \Delta\theta_{4,0}$	$-90 + \Delta\alpha_3$
5	$\Delta a_4$	$\Delta d_5$	$\theta_{5,c} + \Delta\theta_{5,0}$	$90 + \Delta\alpha_4$
6	$\Delta a_5$	$125 + \Delta d_6$	$\theta_{6,0} + \Delta\theta_{6,0}$	$-90 + \Delta\alpha_5$

The normal of circle 1 is taken as the  $z$ -axis and the vertical is the  $x$ -axis.

- (6) One translates  $d_1$  along the negative direction of the  $z$ -axis of  $F_1$ ; the base frame  $F_0$  can be obtained.

After establishing the base frame of the robot, we could transfer the measurement coordinate system to the base frame. Then the world frame  $F_{\text{World}}$  is unified with  $F_0$ .

**2.2.2. Preliminary Identification of the Tool Frame.** The tool frame also needs to be preliminarily identified, where the values of  $x_t$  and  $y_t$  depend primarily on the concentricity of the AT and the end flange. With high machining accuracy of the connecting flange, we can set the value of  $x_t$  and  $y_t$  to be 0. And the value of  $z_t$  could be obtained by the method as follows. Rotating joint 5 with the others being blocked, the position of the AT is measured at a certain angle with the laser tracker. Based on these positions, a circle is fitted. Then, the radius of this circle is the sum of the length of  $z_t$  and  $d_6$ .

Once the sum of the length of  $z_t$  and  $d_6$  is obtained, we can compensate the length of  $z_t$  to  $d_6$ , and  $z_t$  will not participate in the following identification calculation.

**2.3. Design of Fitness Function.** After preliminary identification and unification of the base frame and measurement coordinate system, we can describe the position of the tool frame with respect to the base frame as follows:

$$T_0^{\text{tool}} = T_0^1 T_1^2 T_2^3 T_3^4 T_4^5 T_5^6 T_6^{\text{tool}}. \quad (4)$$

Therefore, the complete kinematic model consists of 24 kinematic parameter errors as shown in Table 2. The kinematic model of the robot can be expressed as

$$p = f(k, q), \quad (5)$$

where  $p$  is the position vector of the end-effector calculated by the model,  $k$  is the vector of 24 kinematic parameters, and  $q$  is the vector of six joint variables. The vector of kinematic parameters  $k$  can also be written as

$$k = k_0 + \delta k, \quad (6)$$

where  $k_0$  is the nominal set of kinematic parameters and  $\delta k$  is the kinematic parameter error. Then, the residuals for such a model are

$$e(\delta k) = p^R - f(k, q), \quad (7)$$

where  $p^R$  is the position vector measured by the laser tracker; the identification problem is to select  $\delta k$  to minimize a cost function:

$$L = e^T(\delta k)e(\delta k) = \|p^R - f(k, q)\|^2. \quad (8)$$

Then, the identification problem will be referred to as the nonlinear least-squares method, and the optimization of  $L$  could be accomplished through evolutionary algorithms.

### 3. Kinematic Parameter Identification

The identification is transformed into an optimization problem of nonlinear systems, and a novel hybrid algorithm of BPNN-PSO is applied to solve the problem.

**3.1. PSO.** Particle swarm optimization (PSO) algorithm is a parallel global optimization algorithm based on swarm intelligence. In PSO, the potential solutions, called particles, search through the problem space by following their own experiences and the current best particles. Due to its simple structure, fast convergence speed, and advantages in dealing with high-dimensional problems, it has been widely used in science and engineering in recent years [21]. The specific steps are as follows.

Assuming the number of particles as  $N$ , the number of iterations is denoted by  $t$  and the maximum number of iterations is denoted by  $t_{\max}$ .

- (1) One selects  $N$  groups of poses in the robot's workspace and gets the actual position error  $e_i$ , where  $i = 1, 2, \dots, N$ .
- (2) Initializing the positions of particle swarms, the position vector and the velocity vector of the  $i$ th particle are set as  $X_i(t)$  and  $V_i(t)$ , respectively. In this step, the initial positions of particles are used as the input of BPNN.
- (3) One calculates the fitness of the  $i$ th particle according to the fitness function.  $p_{id}$  is set as the current position of the particle  $i$ , and  $p_{gd}$  is set as the position of the best particle in the initial population.
- (4) One determines whether iteration termination conditions are met. If the termination conditions are met, the algorithm will stop running and output the optimal result; otherwise, the algorithm will go to step 5.
- (5) One calculates velocity and position of each particle with the following two equations.

$$V_i(t+1) = \omega V_i(t) + c_1 r_1(t)[p_{id} - X_i(t)] + c_2 r_2(t)[p_{gd} - X_i(t)], \quad (9)$$

$$X_i(t+1) = X_i(t) + V_i(t+1), \quad (10)$$

where  $c_1$  and  $c_2$  are accelerating constants,  $r_1$  and  $r_2$  are two independent random functions, and  $\omega$  is a nonnegative number which is called the weight.

- (6) One calculates the fitness of each particle, updates the new local optimal position  $p_{id}$  of each particle, and updates the global optimal position  $p_{gd}$  of the particle swarm.
- (7) One determines whether the iteration termination conditions are met. If met, the algorithm will stop running and output the optimal result; otherwise, the algorithm will go to step 5.

**3.1.1. Design of Objective Function.** According to (1), there are various errors in the robot error model which can affect the positioning accuracy of the robot. In the PSO identification model, each particle represents a set of solutions for optimizing the robot parameters, through which one optimization variable  $\Delta k$  will be reached for the optimization problem.

In Section 2.3, we constructed the objective functions of a set of data for parameter identification. However, in the calibration process, we collected multiple sets of robot data for parameter identification. Therefore, the objective function is set as the sum of fitness functions for multiple sets of robot data. Then the objective function can be expressed as follows:

$$\min F(\Delta k) = \sum_{i=1}^N L_i = \sum_{i=1}^N \|p_i^R - f_i(k, q)\|^2, \quad (11)$$

where  $N$  indicates the number of positions acquired by measurement instruments and  $L_i$  represents the fitness function (8) which means the sum of squared errors of point  $i$ .

**3.1.2. Parameter Setting of PSO.** The parameters that need to be determined in the particle swarm optimization algorithm [22] are the search space dimension  $D$ , the number of particles in the population  $n$ , the acceleration factors  $c_1$  and  $c_2$ , and the inertia weight  $\omega$ . The search space dimension  $D$  is the same as the number of robot kinematic parameters that need to be identified, which is 24 in this paper. For such a high-dimensional search space, the number of particles  $n$  is selected as 200. According to [23], we set acceleration factor  $c_1$  equal to 0.5,  $c_2$  equal to 1.25, and inertia weight  $\omega$  equal to 0.9.

**3.2. BPNN-PSO.** The BP neural network (BPNN) is a representative neural network, which is widely used in many practical systems [24]. The training of neural network is a process that makes the neural network interactive to the external environment in a new way. In the process of training, the free parameters of the neural network will be adapted by the stimulating effect of the environment. The BP neural network is usually composed of an input layer, hidden layer, and output layer. The adjacent layers are fully interconnected, but the nodes of the same layer are not connected. In this paper, a BPNN with a single hidden layer is chosen to combine with particle swarm optimization (PSO) algorithm to solve the parameter error vector  $\Delta k$ .

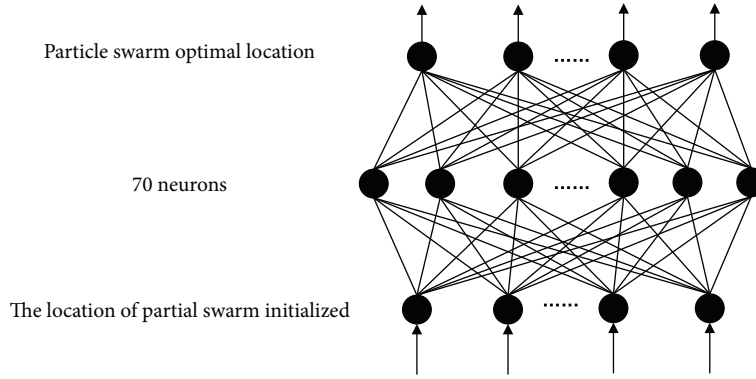


FIGURE 3: The structure of BPNN.

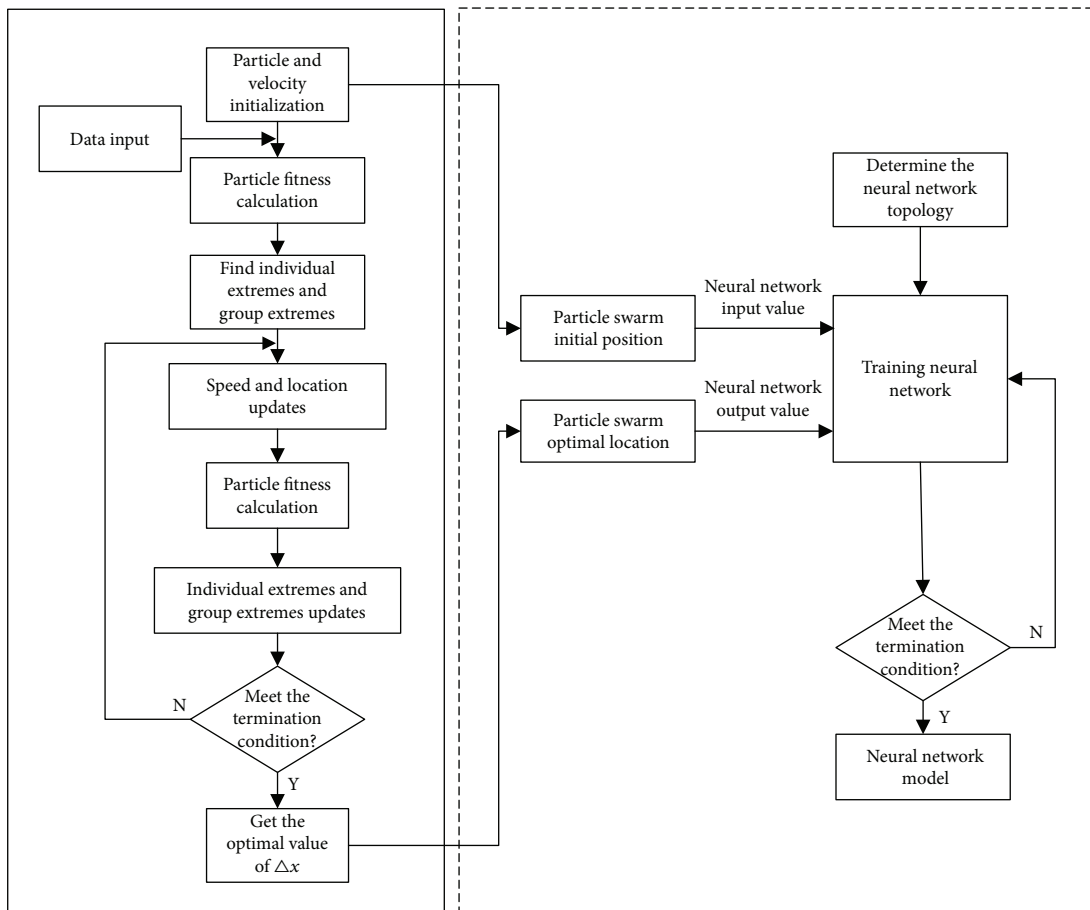


FIGURE 4: The flow chart of BPNN-PSO.

3.2.1. *Training of BPNN.* A three-layer BP neural network with a hidden layer (enough of hidden nodes) can approximate any nonlinear continuous function with arbitrary precision in the closed set [25]. In this paper, we use a three-layer BP neural network with a hidden layer as our network architecture. The model of a neural network based on BPNN is established as shown in (12), where  $w_{ij}$  is the neural network's weight from neuron  $j$  to  $i$ ,  $x_j$  is the input (the positions of particle swarms),  $T_j$  is the threshold of the neuron  $j$ ,  $f$  is the

activation function of neurons, and  $P$  is the position of the best particle.

$$P = f \left( \sum_{j=1}^n w_{ij} x_j - T_j \right). \quad (12)$$

The structure of the BPNN used in this paper is shown in Figure 3. There are  $n$  neurons in the input layer which are the locations of particle swarms initialized in the second step of

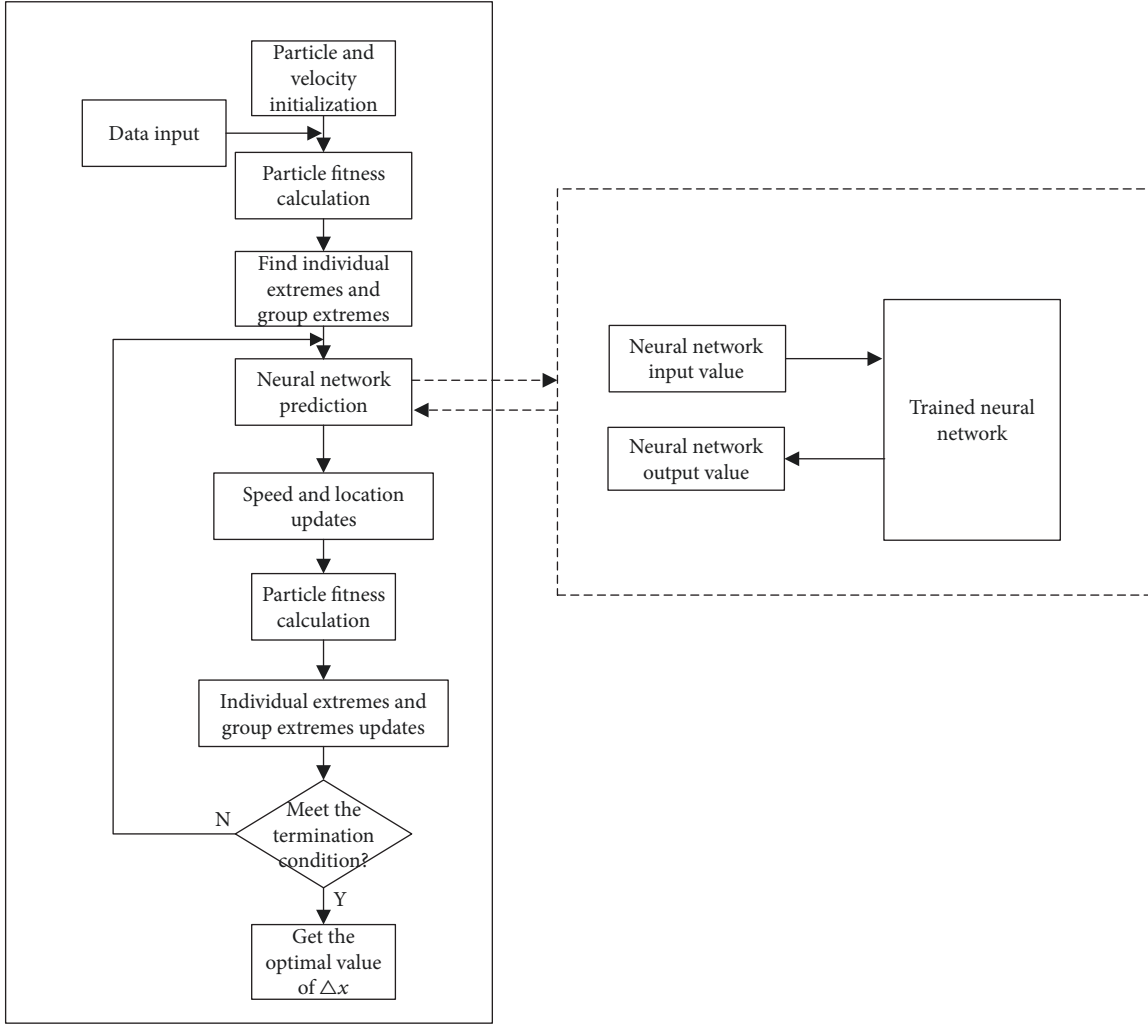


FIGURE 5: The training flow chart of BPNN-PSO.

TABLE 3: The settings of kinematic parameter errors.

Number of linkage	$a_{i-1}$ (mm)	$d_i$ (mm)	$\theta_i$ (°)	$\alpha_{i-1}$ (°)
1	0 + 0.1	504 + 0.05	$\theta_{1,c} + 0.01$	0 + 0.01
2	166.605 + 0.05	0 + 0.1	$\theta_{2,c} + 0.015$	-90 + 0.005
3	782.270 + 0.065	0 + 0.05	$\theta_{3,c} + 0.003$	0 + 0.015
4	138.826 + 0.06	761.35 + 0.008	$\theta_{4,c} + 0.012$	-90 + 0.006
5	0 + 0.03	0 + 0.36	$\theta_{5,c} + 0.008$	90 + 0.004
6	0 + 0.25	125 + 0.01	$\theta_{6,c} + 0.025$	-90 + 0.125

PSO. When the iteration of PSO stops, we will get the optimal value of  $\Delta k$  which is used as the output of BPNN. The number of hidden nodes is a key parameter of the neural network. Determination of the optimal number of hidden nodes has always been a problem that is raised in neural network applications [26]. In this paper, the reference value of the number of nodes in the hidden layer is calculated by the empirical formula  $n_1 = \sqrt{m + na}$ , where  $m$ ,  $n$ , and  $n_1$  represent the number of neurons in the output layers, input layers, and hidden layers, respectively.  $a$  is a constant, and its value is usually between 1 and 10. Then the number of nodes in the

hidden layer is determined through a step-by-step experiment method. As described in Figure 4, the number of particle swarm particles can be regarded as the size of the training data of the BP neural network. When the particle swarm algorithm is used to identify parameters, the number of particles in the swarm is set to 200, so the training data of the BP neural network are 200 groups.

**3.2.2. Parameter Identification Based on BPNN-PSO.** After training, the BPNN model will be added to the PSO algorithm on the next identification procedure of the same



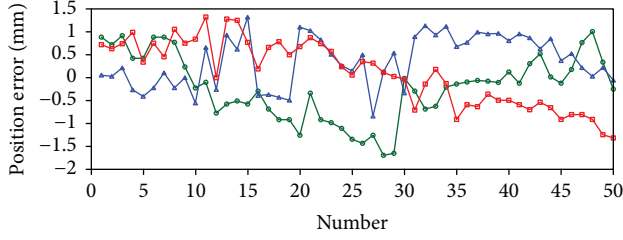


FIGURE 6: The position error before compensation in simulations.

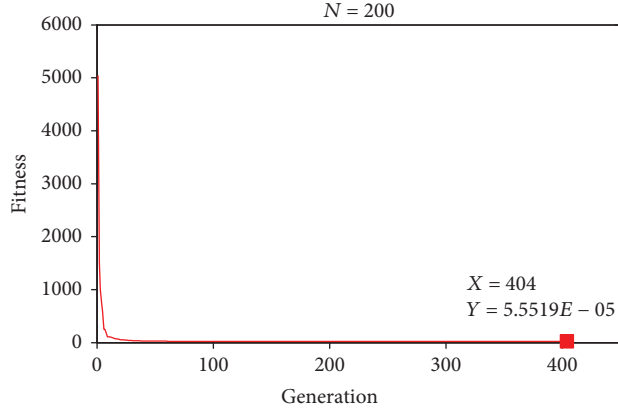


FIGURE 7: Iteration calculation results of PSO in simulations.

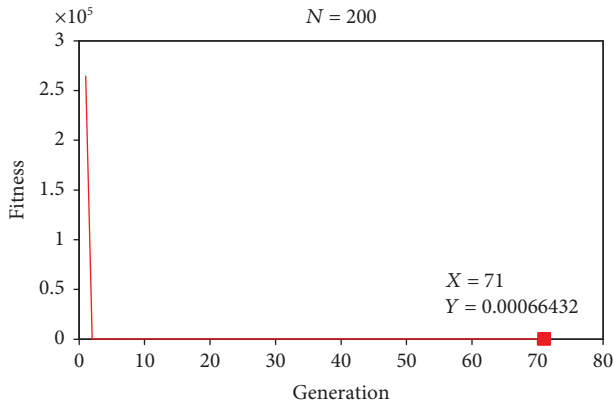


FIGURE 8: Iteration calculation results of BPNN-PSO in simulations.

industrial robot. Comparing with PSO, the objective function is the same, but the proposed BPNN-PSO uses the trained neural networks to predict the location of particles before updating the particles, which can increase the convergence ability and global search capabilities. The training flow chart of BPNN is shown as Figure 5.

## 4. Simulations

**4.1. Error Settings.** To verify the identification model of industrial robots, we set errors with the kinematic parameters and the kinematic parameter-identification simulations are carried out based on PSO and BPNN-PSO. First, we acquired joint angle data, and set errors with kinematic parameters for each joint, as shown in Table 3. The theoretical position can

TABLE 4: The identified error with PSO in simulations.

Number of linkage	$\Delta a_{i-1}$ (mm)	$\Delta d_i$ (mm)	$\Delta \theta_{i,0}$ ( $^\circ$ )	$\Delta \alpha_{i-1}$ ( $^\circ$ )
1	0.0994	0.0508	0.01	0.01
2	0.0577	0.5017	0.0146	0.005
3	0.065	-0.3564	0.0022	0.0149
4	0.0396	0.0115	0.012	0.0065
5	0.035	0.9515	0.0039	0.0043
6	0.229	0.0098	0.9910	-0.0260

TABLE 5: The identified error with BPNN-PSO in simulations.

Number of linkage	$\Delta a_{i-1}$ (mm)	$\Delta d_i$ (mm)	$\Delta \theta_{i,0}$ ( $^\circ$ )	$\Delta \alpha_{i-1}$ ( $^\circ$ )
1	0.1000	0.0503	0.0101	0.01
2	0.0508	-0.1087	0.0149	0.0049
3	0.0643	0.2521	0.0028	0.0148
4	0.0561	0.0086	0.0129	0.0064
5	0.0306	-0.3314	0.0352	0.0046
6	0.3562	0.0122	0.2028	0.3001

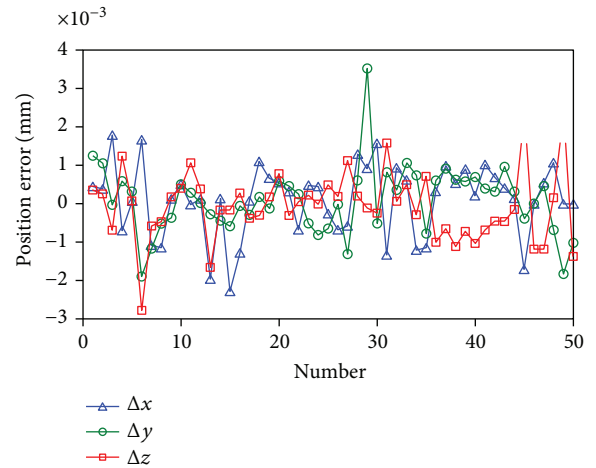


FIGURE 9: The error after simulation compensation with PSO.

be calculated by (1) with the nominal kinematic parameters (shown in Table 1). Similarly, we can obtain the reference position (which is considered as the real one) with the kinematic parameter. Then the position error before identification and compensation is calculated, as shown in Figure 6.

The simulation results of the PSO algorithm and the BPNN-PSO algorithm are used to solve (10), and the convergence speed of the fitness is shown in Figures 7 and 8, respectively. The PSO algorithm achieves the minimum fitness of  $5.5519E-5$  with 404 iterations. The BPNN-PSO achieves the minimum fitness of  $6.6432E-5$  with 71 iterations. The parameter error  $\Delta k$  identified by the two methods is shown in Tables 4 and 5, respectively. Compared with the setting error of the kinematic parameters, not all parameters are accurately identified, which is due to the existence of a coupling relationship among them, but it does not affect the accuracy of compensation [27, 28]. The position error of the two methods after compensation is shown in Figures 9

and 10. Compared with the compensation effect based on PSO algorithm, the identification based on BPNN-PSO can achieve nearly the same precision, but BPNN-PSO has much fewer iteration times and faster convergence speed. To compare the time spent in parameter identification of the two algorithms, we performed parameter identification on the same computer, whose configurations are 64-bit Windows 10 operating system, 8 GB RAM, and an Intel i7 processor. The simulation results show that the time for the identification of the two algorithm parameters is 1067 s and 183 s, respectively.

## 5. Experimental Verification

**5.1. Experimental Data Acquisition.** The data-acquisition process consists of moving the end effector to some positions in the workspace of the robot and recording the joint displacements. In this paper, we obtained the location of the end effector with the laser tracker. The data-acquisition experiment platform is shown in Figure 1. We measured the position data of the robot in 100 different poses and recorded the joint angle data in the corresponding pose. During the measuring process, the position data of the robot should be distributed evenly in the working space as much as possible. Of the 100 groups of acquired data, 50 groups of data are used to identify the kinematic parameters of the robot, and the other 50 groups of data are used for independent verification.

**5.2. Experimental Results.** Following data acquisition, the identification process is performed by PSO algorithm and BPNN-PSO algorithm. The parameters identified by the two algorithms are given by Tables 6 and 7. Figure 11 shows the position errors of the 50 groups of the robot before identification and compensation. The identification results of the two algorithms are used to compensate the kinematic parameters' error, and the position error after compensation is shown in Figures 12 and 13. Figures 14 and 15 show the iteration results and convergence rates, respectively, of the two algorithms. From the two figures, we can know that the PSO algorithm has a slower convergence speed in the late stage of the search, which is due to the fact that the particle swarm tends to be homogenous and the global search capability becomes worse. Compared with PSO, the BPNN-PSO has a faster convergence speed. In Table 8, the effects of the two algorithms are compared. The experiment results show that, compared with the PSO algorithm, the position error after compensation based on the BPNN-PSO algorithm is nearly the same but the convergence rate is improved by 89%. Particularly, the time consumed by the identification-based PSO is 1235 seconds, while the time consumed by BPNN-PSO is only 162 seconds, which means the identification efficiency increased by 86%.

## 6. Conclusions

A novel hybrid parameter-identification method based on BPNN-PSO was proposed for industrial robots to solve the convergence efficiency of standard PSO. The kinematic

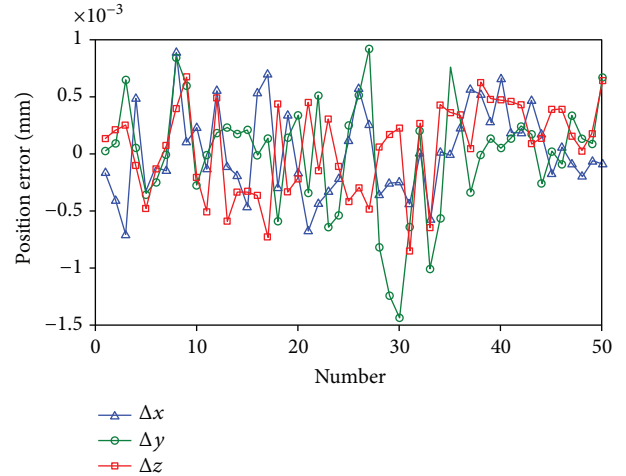


FIGURE 10: The error after simulation compensation with BPNN-PSO.

TABLE 6: The identified error with PSO in experiments.

Number of linkage	$\Delta a_{i-1}$ (mm)	$\Delta d_i$ (mm)	$\Delta \theta_{i,0}$ ( $^\circ$ )	$\Delta \alpha_{i-1}$ ( $^\circ$ )
1	-0.303	1.055	0.292	0.006
2	0.323	-1.227	0.039	0.006
3	0.389	-1.380	0.063	-0.074
4	-1.487	-1.400	0.069	0.229
5	0.204	-1.385	0.441	-0.304
6	0.053	0.572	0.195	0.338

TABLE 7: The identified error with BPNN-PSO in experiments.

Number of linkage	$\Delta a_{i-1}$ (mm)	$\Delta d_i$ (mm)	$\Delta \theta_{i,0}$ ( $^\circ$ )	$\Delta \alpha_{i-1}$ ( $^\circ$ )
1	-0.3025	1.055	0.292	0.006
2	0.352	-1.515	0.040	0.009
3	0.381	-1.545	0.063	-0.037
4	-1.487	-1.527	0.069	0.286
5	0.204	-1.374	0.172	-0.281
6	0.033	0.592	0.447	0.332

model was established based on the MDH model. To unify the position data of the industrial robot and measurement instrument, a preliminary identification of the base frame and the tool frame were presented. The modeling and training method of BPNN-PSO were conducted to identify the kinematic parameters. Simulations and experiments were carried out to verify the efficiency of the proposed method. The results showed that the identification method based on BPNN-PSO can identify the error kinematic parameters and improve the position accuracy of industrial robots. Compared with standard PSO, the identification accuracy of BPNN-PSO is nearly the same, but its convergence efficiency can be significantly improved.

This work mainly aimed at improving the convergence efficiency of parameter identification for industrial



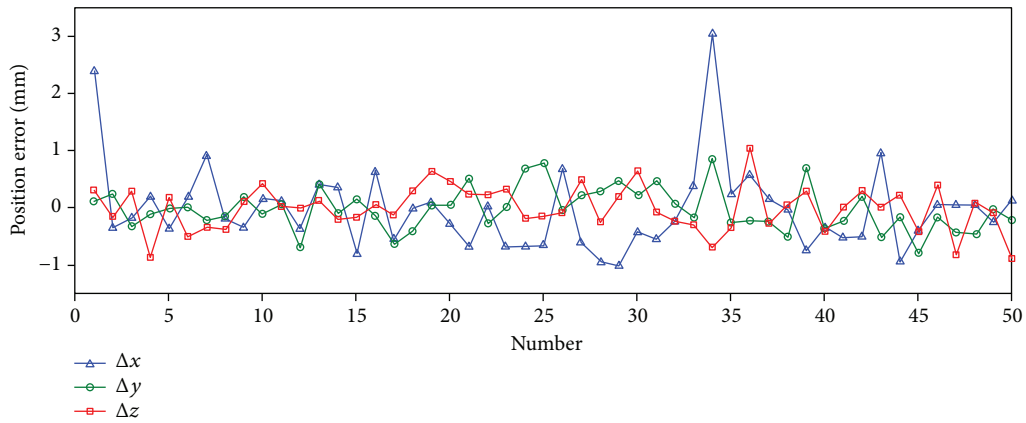


FIGURE 11: The position errors of the industrial robot before compensation.

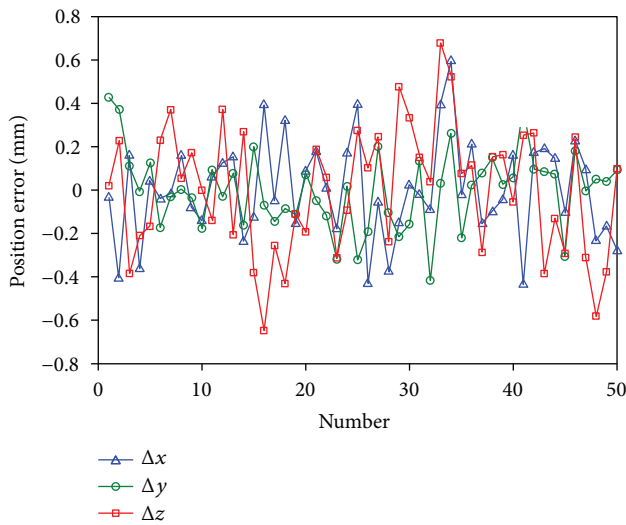


FIGURE 12: The errors after compensation of PSO in experiments.

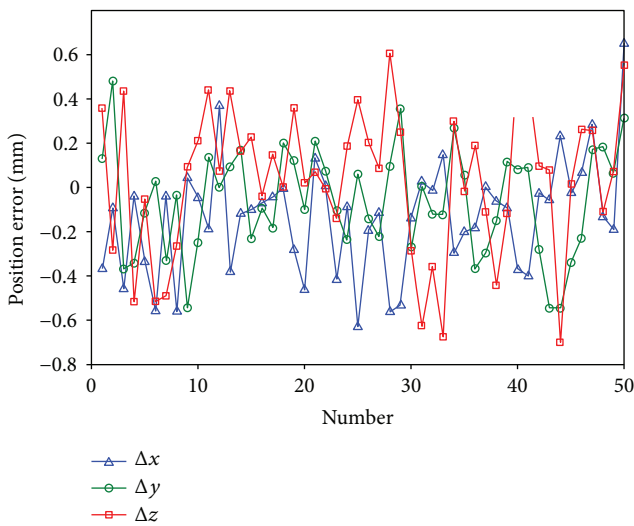


FIGURE 13: The errors after compensation of BPNN-PSO in experiments.

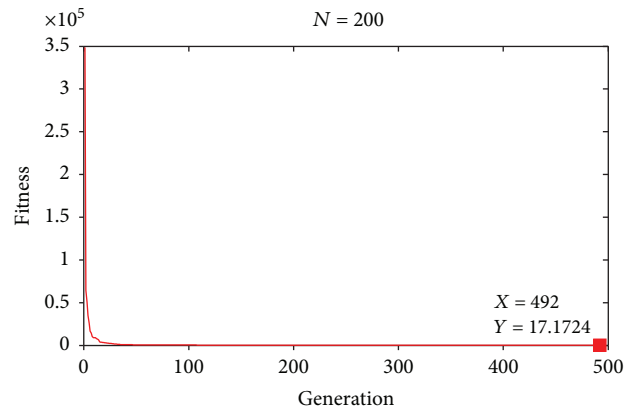


FIGURE 14: Iteration calculation results of PSO in experiments.

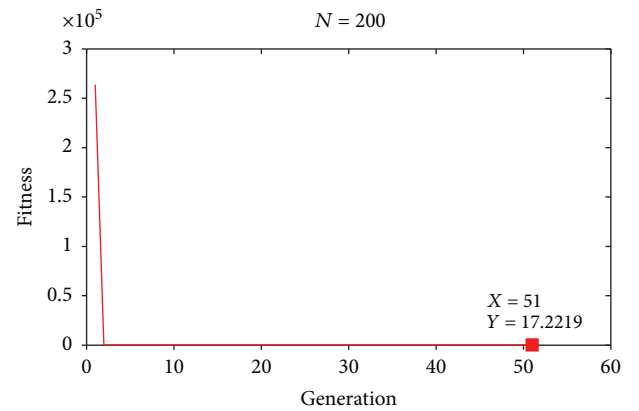


FIGURE 15: Iteration calculation results of BPNN-PSO in experiments.

TABLE 8: The movement errors after identification and compensation in experiments.

Method	Mean (mm)	Std (mm)	Max (mm)	Generation	Fitness	Time cost (s)
PSO	0.4614	0.1843	0.650	492	17.1724	1235
BPNN-PSO	0.4600	0.1836	0.637	51	17.2219	162

robots. Our future work will focus on improving the identification accuracy.

### Data Availability

The data used to support the findings of this study are included within the article.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 51465027).

### References

- [1] Z. Jiang, W. Zhou, H. Li, Y. Mo, W. Ni, and Q. Huang, "A new kind of accurate calibration method for robotic kinematic parameters based on the extended Kalman and particle filter algorithm," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3337–3345, 2018.
- [2] A. Joubair, L. F. Zhao, P. Bigras, and I. Bonev, "Absolute accuracy analysis and improvement of a hybrid 6-DOF medical robot," *Industrial Robot: An International Journal*, vol. 42, no. 1, pp. 44–53, 2015.
- [3] Y. Qiao, Y. Chen, B. Chen, and J. Xie, "A novel calibration method for multi-robots system utilizing calibration model without nominal kinematic parameters," *Precision Engineering*, vol. 50, pp. 211–221, 2017.
- [4] R. Miranda-Colorado and J. Moreno-Valenzuela, "Experimental parameter identification of flexible joint robot manipulators," *Robotica*, vol. 36, no. 3, pp. 313–332, 2018.
- [5] J. Jin and N. Gans, "Parameter identification for industrial robots with a fast and robust trajectory design approach," *Robotics and Computer-Integrated Manufacturing*, vol. 31, pp. 21–29, 2015.
- [6] H. N. Nguyen, J. Zhou, and H. J. Kang, "A calibration method for enhancing robot accuracy through integration of an extended Kalman filter algorithm and an artificial neural network," *Neurocomputing*, vol. 151, pp. 996–1005, 2015.
- [7] 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.
- [8] J. Santolaria, J. Conte, and M. Ginés, "Laser tracker-based kinematic parameter calibration of industrial robots by improved CPA method and active retroreflector," *The International Journal of Advanced Manufacturing Technology*, vol. 66, no. 9–12, pp. 2087–2106, 2013.
- [9] L. S. Ginani and J. M. S. T. Motta, "Theoretical and practical aspects of robot calibration with experimental verification," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 33, no. 1, pp. 15–21, 2011.
- [10] D. Daney, Y. Papegay, and B. Madeline, "Choosing measurement poses for robot calibration with the local convergence method and Tabu Search," *The International Journal of Robotics Research*, vol. 24, no. 6, pp. 501–518, 2005.
- [11] G. Xiong, Y. Ding, L. M. Zhu, and C. Y. Su, "A product-of-exponential-based robot calibration method with optimal measurement configurations," *International Journal of Advanced Robotic Systems*, vol. 14, no. 6, 2017.
- [12] C. Gong, Y. Sun, and L. Yuan, "Study on application of hybrid optimization algorithm in parameters optimization of manipulator," *Journal of System Simulation*, vol. 30, pp. 105–113, 2018.
- [13] C. Fan, G. Zhao, J. Zhao, L. Zhang, and L. Sun, "Calibration of a parallel mechanism in a serial-parallel polishing machine tool based on genetic algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 81, no. 1–4, pp. 27–37, 2015.
- [14] W. Wang, H. Song, Z. Yan, L. Sun, and Z. Du, "A universal index and an improved PSO algorithm for optimal pose selection in kinematic calibration of a novel surgical robot," *Robotics and Computer-Integrated Manufacturing*, vol. 50, pp. 90–101, 2018.
- [15] Y. Shi, B. Liang, X. Wang, and W. Xu, "Cartesian non-holonomic path planning of space robot based on quantum-behaved particle swarm optimization algorithm," *Journal of Mechanical Engineering*, vol. 47, no. 23, p. 65, 2011.
- [16] L. Fang and P. Dang, "Kinematic calibration method of robots based on quantum-behaved particle swarm optimization," *Journal of Mechanical Engineering*, vol. 52, no. 7, p. 23, 2016.
- [17] G. Gao, H. Zhang, H. San, X. Wu, and W. Wang, "Modeling and error compensation of robotic articulated arm coordinate measuring machines using BP neural network," *Complexity*, vol. 2017, Article ID 5156264, 8 pages, 2017.
- [18] W. Yang, X. Liu, W. Lu, and X. Guo, "Influence of probe dynamic characteristics on the scanning speed for white light interference based AFM," *Precision Engineering*, vol. 51, pp. 348–352, 2018.
- [19] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, Addison-Wesley Pub. Co, 3rd edition, 2005.
- [20] A. Nubiola and I. A. Bonev, "Absolute calibration of an ABB IRB 1600 robot using a laser tracker," *Robotics and Computer-Integrated Manufacturing*, vol. 29, no. 1, pp. 236–245, 2013.
- [21] J. Wei and G. B. Liu, "An improved particle swarm optimization algorithm with immunity," in *2009 Second International Conference on Intelligent Computation Technology and Automation*, pp. 241–244, Changsha, Hunan, China, October 2009.
- [22] L. Liu, Y. Wang, F. Xie, and J. Gao, "Legendre cooperative PSO strategies for trajectory optimization," *Complexity*, vol. 2018, Article ID 5036791, 13 pages, 2018.
- [23] N. Zeng, H. Zhang, W. Liu, J. Liang, and F. E. Alsaadi, "A switching delayed PSO optimized extreme learning machine for short-term load forecasting," *Neurocomputing*, vol. 240, pp. 175–182, 2017.
- [24] 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.
- [25] Z. R. Tsai, "Robust digital design of continuous-time nonlinear control systems using adaptive prediction and random-local-optimal NARMAX model," *Applied Mathematics and Computation*, vol. 221, pp. 152–163, 2013.
- [26] C. Cheng, X. Cheng, N. Dai, X. Jiang, Y. Sun, and W. Li, "Prediction of facial deformation after complete denture prosthesis

using BP neural network,” *Computers in Biology and Medicine*, vol. 66, pp. 103–112, 2015.

- [27] G. Gao, J. Na, X. Wu, and Y. Guo, “A self-calibration method for articulated arm coordinate measuring machines,” *Transactions of the Canadian Society for Mechanical Engineering*, vol. 40, no. 4, pp. 645–655, 2016.
- [28] W. Lu, X. Liu, L. Zhou, and H. Heiyang, “Non-linear dynamic compensation and re-sampling for tactile scanning measurement of curved surface topography based on GPS standards,” *Measurement*, vol. 45, no. 6, pp. 1633–1638, 2012.



**Hindawi**

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

