applied sciences

MDPI

Article

Research on Positioning Error Compensation of Rock Drilling Manipulator Based on ISBOA-BP Neural Network

Qiaoyu Xu *, Wenhao Ju, Yansong Lin and Tianle Zhang

College of Mechanical and Electrical Engineering, Henan University of Science and Technology, Luoyang 471000, China; juwenhao@stu.haust.edu.cn (W.J.); lys@stu.haust.edu.cn (Y.L.); ztl@stu.haust.edu.cn (T.Z.)

* Correspondence: xiaoyu0622@163.com; Tel.: +86-137-8311-5261

Abstract: In order to solve the problem of the low end positioning accuracy of large hydraulic rock drilling robotic arms due to machining error and the working environment, this paper proposes an end positioning error compensation method based on an Improved Secretary Bird Optimization Algorithm (ISBOA) optimized Back Propagation (BP) neural network. Firstly, the good point set strategy is used to initialize the secretary bird population position to make the initial population distribution more uniform and accelerate the convergence speed of the algorithm. Then, the ISBOA is used to optimize the initial weights and biases of the BP neural network, which effectively overcomes the defect of the BP neural network falling into a local optimum. Finally, by establishing the mapping relationship between the joint value of the robot arm and the end positioning error, the error compensation is realized to improve the positioning accuracy of the rock drilling robot arm. The experimental results show that the average positioning error of the rock drilling robotic arm is reduced from 187.972 mm to 28.317 mm, and the positioning accuracy is improved by 84.94%, which meets the engineering requirements.

Keywords: rock drilling robotic arm; secretary bird optimization algorithm; good point set; BP neural network; error compensation

check for updates

Citation: Xu, Q.; Ju, W.; Lin, Y.; Zhang, T. Research on Positioning Error Compensation of Rock Drilling Manipulator Based on ISBOA-BP Neural Network. *Appl. Sci.* **2024**, *14*, 8480. https://doi.org/10.3390/app14188480

Academic Editors: Pavol Božek, Tibor Krenicky and Yury Nikitin

Received: 23 August 2024
Revised: 14 September 2024
Accepted: 18 September 2024
Published: 20 September 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the rapid advancement of modern engineering technology, large hydraulic rock drilling robots have found widespread applications in various fields, including mining, tunnel boring, and underground engineering. Drilling and blasting are the most commonly used excavation methods in mining and tunneling [1]. In traditional rock drilling methods, the requirement for positioning accuracy is relatively low, as it primarily serves blasting and rock removal purposes. However, in fully automated rock drilling robots, the drilling task is carried out automatically according to a predefined hole-laying plan, and precise positioning is critical to ensuring the accuracy of the drilling location. This accuracy, in turn, guarantees the successful execution of the drilling and blasting operations. Therefore, high positioning accuracy is essential for fully automated rock drilling robots. The positioning accuracy of the rock drilling robot arm also directly influences both construction efficiency and operational safety. However, due to the influence of mechanical geometry errors, drive system errors, load variations, control system errors, and environmental factors, the actual position of the rock drilling arm's end often deviates from its theoretical position. Therefore, exploring methods to enhance the positioning accuracy of rock drilling robotic arms, particularly in fully automated rock drilling robots, is of substantial practical significance.

Common error compensation methods primarily fall into two categories: model-based and non-model-based methods. Model-based methods [2–10] correct the robot's kinematic model by accounting for kinematic parameter errors, which can yield better results when the system is well-defined and stable. However, these methods involve complex modeling

and computational processes [11] and are less effective in compensating for non-geometric errors. Non-model-based methods [12–17], on the other hand, identify error patterns and trends using real-time feedback or historical data to achieve error compensation without the need for detailed system modeling. This method contributes to enhancing the system's stability and accuracy. However, in the practical application of error compensation for rock drilling robotic arms, traditional non-model-based approaches exhibit certain limitations.

Among the non-model-based approaches, neural network-based error compensation algorithms [18,19] offer advantages such as high efficiency, adaptability, and robustness, enabling them to handle complex nonlinear relationships and improve error correction efficiency. Compared to other traditional non-model-based methods, neural networks possess the ability to adaptively learn error patterns and optimize compensation strategies, thereby addressing the limitations of traditional methods regarding data processing and model complexity. Zhou et al. [20] proposed a position error compensation method for rock drilling robots based on LightGBM and RBFN, while ref. [21] employed a genetic algorithm (GA)-optimized deep neural network (DNN) to enhance robot localization accuracy. BP neural networks have been widely applied in the field of robot localization due to their powerful nonlinear mapping capabilities and self-learning properties [22–24]. However, the BP neural network's training process is prone to falling into local optima, and the training speed is slow, which significantly hinders its application in practical engineering scenarios.To address these challenges, various optimization algorithms have been proposed. Zhang et al. [25] optimized the BP neural network using the whale optimization algorithm, significantly enhancing the absolute positioning accuracy of the robotic arm. Jiang et al. [26] utilized a particle swarm optimization (PSO) algorithm to enhance the BP neural network's error compensation performance. Chen et al. [27] applied a quantum genetic algorithm to improve the BP neural network for robot localization compensation. Li et al. [28] combined the genetic algorithm with particle swarm optimization to enhance neural network performance, resulting in improved robot localization accuracy. Zhang et al. [29] used the sparrow search algorithm to optimize the BP neural network, effectively reducing robot localization errors. Although these optimization algorithms have improved robot localization accuracy to some extent, issues such as low search efficiency and suboptimal optimization effects persist and require further improvement.

To address the aforementioned challenges, this paper proposes an error compensation method based on an ISBOA optimized BP neural network.

The main contributions of this paper are as follows:

1. Through improvements to the SBOA, the ISBOA is proposed. By employing the good point set strategy to initialize the population, the diversity of the initial population is increased, and the convergence speed of the algorithm is improved.
2. The ISBOA is utilized to optimize the initial weights and biases of the neural network, effectively addressing the issue of the BP neural network's tendency to fall into local optima.
3. A neural network mapping relationship is established between the joint values of the robotic arm and the end-positioning error. This enables effective error compensation and significantly improves the positioning accuracy of the rock-drilling robotic arm.

The remainder of this paper is organized as follows: Section 2 presents the forward kinematic modeling and error calculation for the robotic arm. Section 3 describes the method for obtaining the BP neural network's weights and biases using the ISBOA and introduces the ISBOA-BP neural network-based error compensation method for rock drilling robotic arm positioning. Section 4 validates the effectiveness of the proposed method through experimental results.

2. Rock-Drilling Robot Arm Modeling and Error Calculation

To enhance the positioning accuracy of the rock-drilling robot arm, it is essential to minimize the distance between the actual position of the end-effector and its theoretical position. By modeling the robot arm, the theoretical position of the end-effector can be

determined, enabling the analysis and calculation of the error between the actual and theoretical positions.

2.1. Rock-Drilling Robot Arm Modeling

The rock-drilling arm possesses seven degrees of freedom, which include five rotary joints and two telescopic joints. In its unextended state, the total length of the arm and the beam is 7835 mm. The arm can be extended by an additional 1600 mm, and the beam can be extended by 1800 mm. The structure of the rock-drilling arm is depicted in Figure 1.

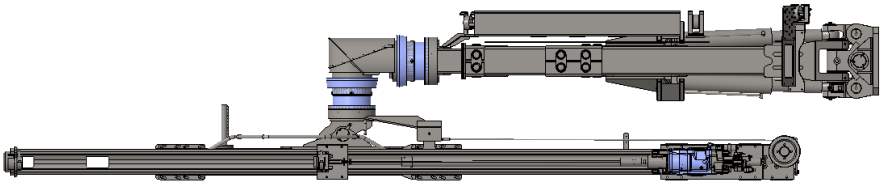


Figure 1. Structure of the robotic arm.

In this study, based on the mechanical structure of the rock-drilling robotic arm depicted in Figure 1, the DH method was utilized to establish the corresponding link coordinate system, as illustrated in Figure 2. The DH parameters of the robotic arm are presented in Table 1.

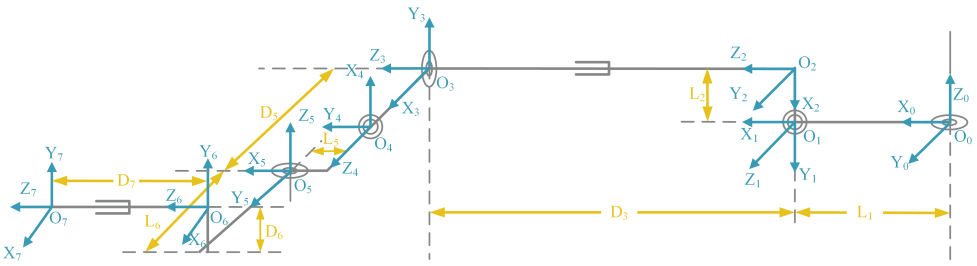


Figure 2. Coordinate system of each joint linkage in the robotic arm.

Table 1. DH parameters of the robotic arm.

Joint	θ_i/°	D_i/mm	L_i/mm	α_i/°
1	θ_1 (0)	0	200	−90
2	θ_2 (90)	0	−35	90
3	90	D_3 (4297)	0	0
4	θ_4 (90)	0	0	90
5	θ_5 (90)	650	80	90
6	θ_6 (90)	679.5	241	90
7	0	D_7 (3058)	0	0

The coordinates between coordinate system O_{i-1} and coordinate system O_i are transformed through rotation or translation to obtain the transformation matrix between the neighboring coordinate systems. The corresponding transformation matrix is presented in Equation (1).

$$ {}^{i-1}_{i}T = Rot(z,\theta_i)Trans(0,0,D_i)Trans(L_i,0,0)Rot(x,\alpha_i) \quad (1) $$

where ${}^{i-1}_{i}T$ is the positional transformation matrix of coordinate system O_{i-1} relative to coordinate system O_i, *Rot* represents the rotational transformation, and *Trans* represents the translational transformation.

Expanding (1) yields the positional matrix between the neighboring coordinate systems.

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & L_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & L_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & D_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

The end position of the robotic arm is obtained by sequentially multiplying the coordinate systems using the corresponding parameters within the transformation matrices.

$$ {}^{0}_{7}T = {}^{0}_{1}T{}^{1}_{2}T{}^{2}_{3}T{}^{3}_{4}T{}^{4}_{5}T{}^{5}_{6}T{}^{6}_{7}T = \begin{bmatrix} R & P \\ 0 & 1 \end{bmatrix} \tag{3} $$

where $R = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix}$ and $P = [p_x, p_y, p_z]^T$. R denotes the end orientation of the robotic arm, and P represents the end position.

2.2. Error Calculation

The DH parameter method is employed to calculate the theoretical position coordinates of the robotic arm's end joints. The actual position coordinates of the corresponding joints' ends are then measured and recorded using a total station. The difference between the theoretical position coordinates and the actual position coordinates is calculated along the three XYZ coordinate axes, yielding the positioning errors X_{error}, Y_{error} and Z_{error} in each respective direction. The total positioning error is represented as $\sqrt{{X_{error}}^2 + {Y_{error}}^2 + {Z_{error}}^2}$.

In this study, a Leica TS16 total station is used to collect and record the actual position coordinates of the robotic arm's end. The TS16 offers an angular accuracy of 1 arcsecond and a ranging accuracy of 1 mm + 1.5 ppm, ensuring that the accuracy requirements are met.

The principle and process of data collection using a total station are as follows:

1. The vehicle coordinate system was established based on the mechanical structure of the rock-drilling robot. Two prisms were installed on the robot, and their coordinates were measured within the vehicle coordinate system.
2. The total station was set up using these two prisms with known coordinates.
3. The total station was used to measure the coordinates of a reflective sticker placed at the end of the robot arm, thereby collecting accurate positional information of the robot arm's endpoint.

3. Research on Positioning Error Compensation for Rock-Drilling Robot Arms Based on ISBOA-BP Neural Network

To address the issues of the BP neural network, specifically its tendency to fall into local minima and its slow convergence speed, this paper utilizes the ISBOA to optimize the initial weights and biases of the BP neural network. This optimization enhances both the convergence speed and the output accuracy of the BP neural network.

3.1. SBOA

In this paper, the newly proposed optimization algorithm SBOA (introduced in 2024) is analyzed and studied. The search accuracy, convergence speed, and stability of SBOA are evaluated using the CEC2017 and CEC2022 benchmark test suites [30]. Inspired by the predation strategy of secretary birds searching for prey in harsh environments and their escape tactics to avoid predators, SBOA was designed to address complex optimization problems. In SBOA, each secretary bird's position represents a candidate solution to the optimization problem, and the algorithm not only demonstrates a high search efficiency but also identifies high-quality solutions.

In the secretary bird's predation strategy, the bird searches across a wide area; a behavior abstracted to simulate the global optimization process, where the algorithm performs an extensive search within the solution space. This global search strategy allows the algorithm to explore potential high-quality solutions throughout a broad region of the solution space. The predation strategy of the secretary bird is divided into three phases: finding prey, attacking prey, and consuming prey. These correspond to the phases $t < \frac{1}{3}T$, $\frac{1}{3}T < t < \frac{2}{3}T$, and $\frac{2}{3}T < t < T$ within the algorithm, where t represents the current iteration number and T denotes the maximum number of iterations, as shown in Figure 3.

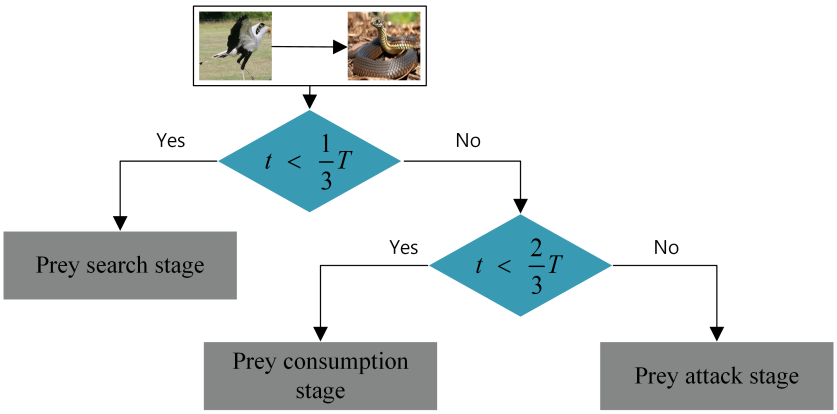


Figure 3. Secretary bird predation strategy.

During the prey-finding phase $t < \frac{1}{3}T$, for optimization algorithms, the initial phase requires stronger exploration capabilities to collect adequate information about the entire search space. Therefore, a differential evolution strategy is applied to strengthen exploration during this stage. Differential evolution leverages differences between individuals to generate new solutions, enhancing both the diversity of the algorithm and its global search capability. The position of the secretary bird in this phase is updated using Equation (4):

$$x_{i,j}^{newP1} = x_{i,j} + (x_{random_1} - x_{random_2}) \times R_1 \tag{4}$$

where $x_{i,j}^{newP1}$ denotes the value of its j-th dimension, x_{random_1} and x_{random_2} are random candidate solutions for the first-stage iteration, and R_1 represents a randomly generated array of dimension $1 \times Dim$ in the interval [0, 1], where Dim is the dimension of the solution space.

Once the secretary birds spot potential prey, they initially hover around it, exhausting its patience and causing it to lower its guard. Similarly, by treating the best current individual as the prey, the secretary birds update their positions to approach it. Brownian motion RB is introduced to model the stochastic motion of the secretary bird during the prey consumption phase $\frac{1}{3}T < t < \frac{2}{3}T$. Equation (5) is used to update the position of the secretary bird during this phase:

$$x_{i,j}^{newP1} = x_{best} + \exp((\frac{t}{T})^{\wedge 4}) \times (RB - 0.5) \times (x_{best} - x_{i,j}) \tag{5}$$

where x_{best} denotes the current best value, $RB = randn(1, Dim)$ represents the Brownian motion formula, and $randn(1, Dim)$ is a randomly generated array of dimension $1 \times Dim$ from a standard normal distribution: mean = 0, standard deviation = 1.

After continuous exertion, the secretary bird initiates its attack once the prey is exhausted. During the prey attack phase $\frac{2}{3}T < t < T$, the secretary bird uses the Levy flight strategy to simulate various attack methods, thereby enhancing the algorithm's global search capability and improving its convergence accuracy. The rapid approach of the bird

to its prey signifies that the candidate solution is nearing the current optimal solution. The position of the secretary bird during this phase is updated using Equation (6):

$$x_{i,j}^{newP1} = x_{best} + (1 - \frac{t}{T})^{\wedge(2\times\frac{t}{T})} \times x_{i,j} \times RL \quad (6)$$

where $RL = 0.5 \times Levy(Dim)$, and $Levy(Dim)$ denotes the Levy flight distribution function.

In the wild, secretary birds also face the risk of predation while hunting other prey. When confronted with a predator and forced to flee, secretary birds engage in rapid, confined movements within a small area. This behavior can be likened to an algorithm performing a fine-grained local search in a specific region of the solution space. This local search strategy enables the algorithm to explore potentially better solutions in the vicinity of already known high-quality solutions. Secretary birds employ two main escape strategies: environmental camouflage and fleeing.

When confronted by a predator, secretary birds initially attempt to camouflage themselves to avoid danger. They adjust their position relative to their prey (the best individual), mimicking the behavior of avoiding local optima in algorithms. The mathematical model for this strategy is represented in Equation (7).

$$x_{i,j}^{\text{new}P2} = x_{\text{best}} + (2 \times RB - 1) \times \left(1 - \frac{t}{T}\right)^2 \times x_{i,j} \quad (7)$$

If they are unable to avoid the predator, they resort to fleeing to ensure their safety. This behavior helps avoid confinement to a local optimum. The secretary birds then update their positions using Equation (8):

$$x_{i,j}^{\text{new}P2} = x_{i,j} + R_2 \times \left(x_{\text{random}} - K \times x_{i,j}\right) \quad (8)$$

where R_2 denotes a randomly generated array of dimension $1 \times Dim$ from a normal distribution, x_{random} denotes a random candidate solution for the current iteration, and K denotes a random choice of the integers 1 or 2.

In the secretary bird's escape strategy, environmental camouflage and fleeing are executed with equal probability, highlighting the randomness in searching the feasible solution space.

3.2. Improvement of SBOA

Population initialization is crucial for enhancing the search ability of stochastic algorithms. Due to its simplicity and inherent randomness, the SBOA also utilizes a random initialization method to determine the initial positions of the secretary birds within the search space:

$$X_{i,j} = lb_j + r \times \left(ub_j - lb_j\right), i = 1, 2, \ldots, N, j = 1, 2, \ldots, Dim \quad (9)$$

where X_i denotes the position of the i-th secretary bird, lb_j and ub_j represent the lower and upper bounds, respectively, and r denotes a random number between 0 and 1.

However, the distribution of randomly generated populations in the solution space often tends to be uneven, leading to over-density in some regions and excessive sparsity in others. This uneven distribution reduces the algorithm's efficiency in utilizing the entire search space and affects population diversity. The good point set strategy [31], an optimization method aimed at improving search efficiency, has been successfully applied to enhance various optimization algorithms [32–35]. To address these issues, this paper proposes an improved SBOA that initializes the secretary bird population using the good point set strategy. This approach results in a more uniform distribution of the initial population within the solution space and improves the algorithm's convergence speed. The good point set strategy enables more uniform coverage of the entire search space with fewer sample points.

Suppose H_s is the unit cube in an s-dimensional Euclidean space. If $r \in H_s$, it takes the following form:

$$P_n(k) = \left\{ \left(r_1^{(n)} \cdot k \right), \left(r_2^{(n)} \cdot k \right), \cdots, \left(r_s^{(n)} \cdot k \right), 1 \le k \le n \right\} \tag{10}$$

Its deviation $\varphi(n)$ satisfies $\varphi(n) = c(r,\varepsilon)n^{-1+\varepsilon}$, where $c(r,\varepsilon)n^{-1+\varepsilon}$ is a constant related only to r and ε, with ε being an arbitrary positive number. Then, $P_n(k)$ is referred to as the set of good points, and r is the good point, where $\left(r_s^{(n)} \cdot k \right)$ represents the fractional part and n denotes the number of points, with r $= \{2\cos(2\pi k/p), 1 \le k \le s\}$, where p is the smallest prime number satisfying $(p-3)/2 \ge s$.

The mapping to the search space is as follows:

$$x_i(j) = (ub_j - lb_j) \cdot \{ r_j^{(i)} \cdot k \} + lb_j \tag{11}$$

where lb_j and ub_j represent the lower and upper bounds, respectively. To verify the uniformity of the point sets generated by the good point set strategy compared to randomly generated point sets, 100 initial points were generated using both the good point set strategy and the random strategy in 2D and 3D spaces, respectively. Their distributions are shown in Figure 4.

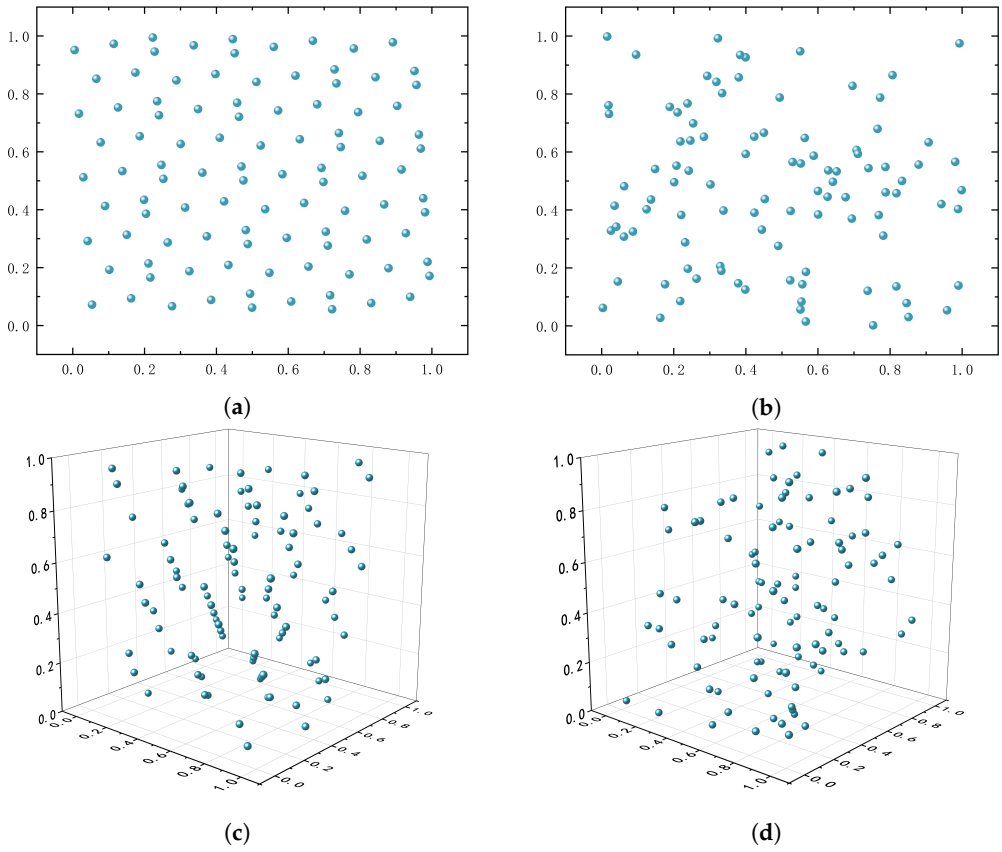


Figure 4. Distribution of points generated by the two strategies. (**a**) Two-dimensional point distribution generated by the good point set strategy. (**b**) Two-dimensional point distribution generated by the randomized strategy. (**c**) Three-dimensional point distribution generated by the good point set strategy. (**d**) Three-dimensional point distribution generated by the randomized strategy.

It can be observed that the points generated by the good point set strategy, as illustrated in Figure 4a,c, are more uniformly distributed compared to those generated by the random strategy, as shown in Figure 4b,d. Consequently, the incorporation of the good point set

strategy effectively enhances population diversity, which facilitates global search in the early stages of the algorithm and accelerates convergence.

3.3. ISBOA-BP-Based Localization Error Compensation Method for Rock-Drilling Robotic Arms

ISBOA demonstrates a strong global search capability by simulating the secretary bird's behavior over a wide search space. This assists the BP neural network in efficiently optimizing its weights and biases, preventing it from becoming trapped in local optima and ultimately enhancing the accuracy of the final training outcome. In optimizing the BP neural network using the ISBOA, the neural network topology is first determined by processing the dataset. The positions generated by the good point set represent the search agents, i.e., the secretary bird individuals, in the ISBOA. The initial position of each agent represents a potential solution to the weights and biases in the neural network. According to the ISBOA strategy, the search agents explore and update their positions in the solution space, with fitness calculations performed at each population iteration. This iterative process evolves the search agents towards better solutions, eventually identifying the optimal search agent. The optimal weights and biases obtained from the ISBOA are then used as the initial weights and biases for training the BP neural network. The flow of the algorithm is illustrated in Figure 5.

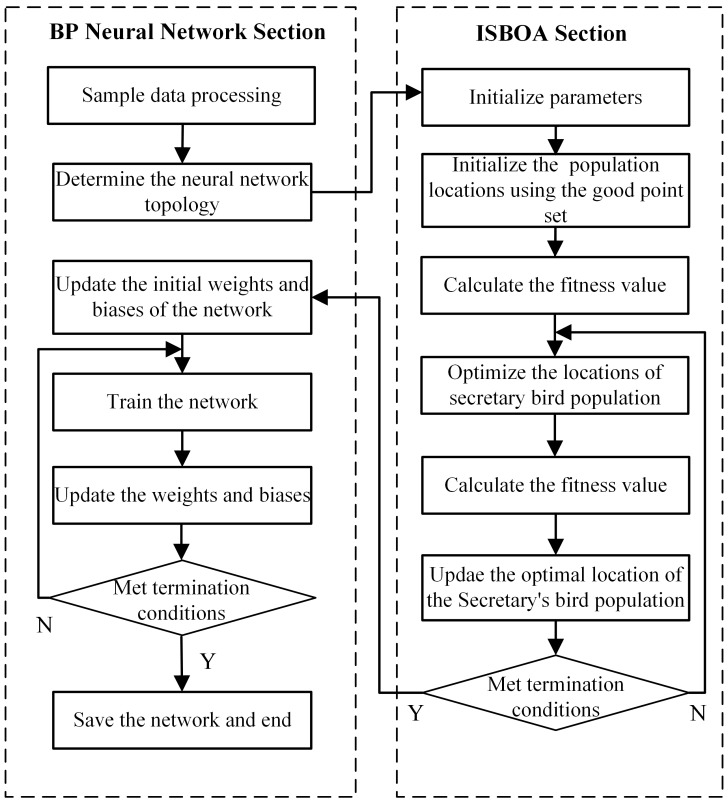


Figure 5. Flowchart of the BP neural network algorithm optimized by ISBOA.

The BP neural network consists of an input layer, a hidden layer, and an output layer, with neurons in each layer connected by weights. In this study, the BP neural network takes seven joint values as input and outputs the error in the XYZ directions corresponding to these joint values. Therefore, the BP neural network has seven input nodes and three output nodes. After multiple experiments and analyses, it was found that the network achieves the best error prediction when the hidden layer consists of a single layer with 15 neurons and the tansig function is used as the activation function. The best results were obtained under these conditions. The network is trained with 1500 iterations, and the learning rate is set at 0.01. Given that the BP neural network has a total of 168 weights and biases, the individual dimension of the population in ISBOA is set to 168, and the

population size is set to 50. The topology of the BP neural network used in this study is illustrated in Figure 6.

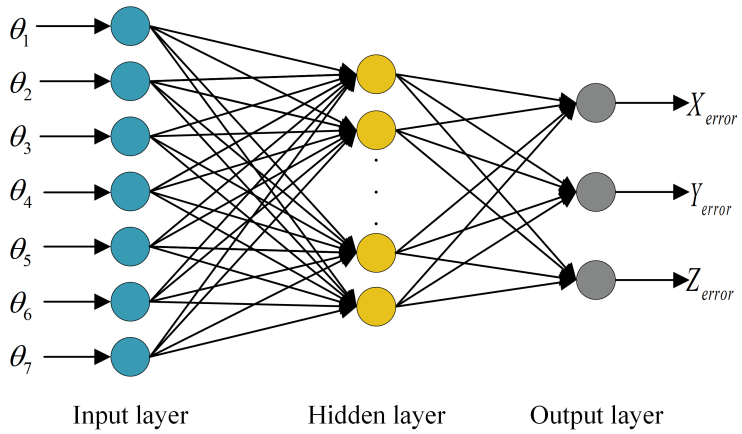


Figure 6. Structure of the BP neural network.

To find the global optimal solution, the mean square error (MSE) between the predicted and actual values is used as the fitness function.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(e_i - \hat{e}_i)^2 \tag{12}$$

A smaller MSE value indicates a smaller gap between the predicted and true values, thereby reflecting a more accurate and optimal solution.

In the sample data selected for this study, each group contains the values of seven joints of the robotic arm $\theta_i (i = 1 \text{ to } 7)$ and their corresponding end position errors E, where $E = (X_{error}, Y_{error}, Z_{error})$. The mapping model between the robotic arm's joint values and end positioning errors is constructed through network training.

The process of positioning error compensation for the rock drilling robotic arm is as follows: first, determine the joint values of each joint of the robotic arm to be compensated, $\theta_i^{aim} (i = 1 \text{ to } 7)$, and use the mapping relationship established by the neural network to calculate the corresponding end positioning error E^{aim}, i.e., the required compensation amount. Next, the theoretical position of the end, P_t^{aim}, is calculated using the forward kinematics of the robotic arm. Finally, this theoretical position is added to the compensation amount to obtain the actual position of the end, $P_a^{aim} = P_t^{aim} + E^{aim}$. The flow of the error compensation process is shown in Figure 7.

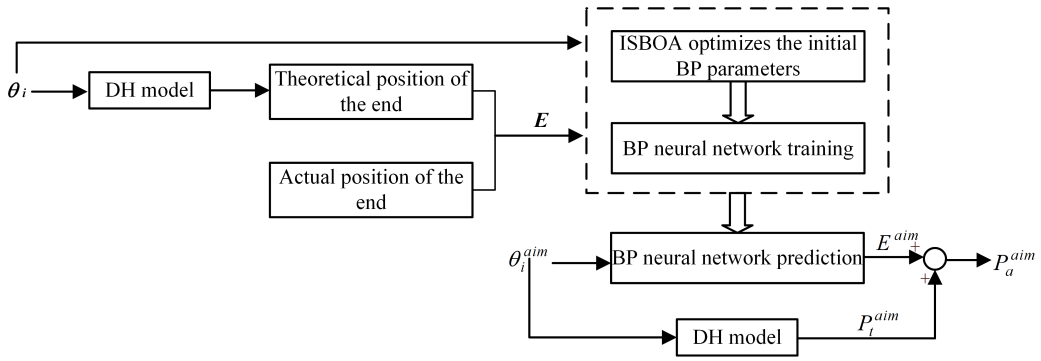


Figure 7. Compensation process for positioning error at the end of the robotic arm.

4. Experimental Verification

In this study, a G3Zi-type large hydraulic rock-drilling robot with dimensions of 16.5 × 2.9 × 3.4 m and an effective working area of 16 × 13 m was selected as the experimen-

tal object, as shown in Figure 8. This setup was used to validate the effectiveness of the method for compensating for the positioning error of the rock-drilling robotic arm based on the ISBOA-BP neural network.



Figure 8. G3Zi rock-drilling robot.

4.1. Sample Data Acquisition and Parameter Setting

The construction of the training sample dataset is critical, as it directly impacts the accuracy and generalization ability of the prediction model. To build a comprehensive training dataset within the working range of the robotic arm, the Monte Carlo pseudo-random number method is first employed to determine the full spatial range accessible by the end of the rock-drilling robotic arm based on the joint limits of the arm. During the dataset construction process, the straight-line distance between the rock-drilling robot and the palm face is continuously adjusted. Position points are planned within the coverage of each palm face, ensuring an even distribution across the entire surface. This guarantees both the completeness and diversity of the training sample dataset.

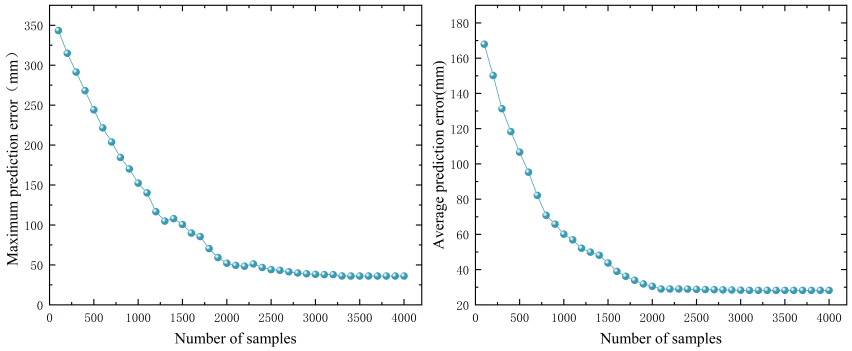
In the rock-drilling robot, each joint of each arm is equipped with an encoder that outputs joint information in real time during the arm's operation, as shown in Figure 9a,b. The encoder used is the AAM58A10-1213-CDCOR3P absolute encoder produced by AiYan Mechatronics, featuring a single-turn resolution of $\leq$13 bits and excellent repeatability. Its error range is controlled within ±1 bit, which meets the experimental accuracy requirements. The control unit is responsible for collecting and storing the values of each robotic arm joint.The actual position data at the end of the robotic arm are then collected and recorded by the total station. The use of a high-precision encoder and total station ensures the accuracy of the training sample dataset.



(**a**) (**b**)

Figure 9. Encoders applied to rock-drilling robots. (**a**) Encoders at rotary joints. (**b**) Encoders at telescopic joints.

The number of training samples is a key factor for the neural network. When the training sample size is too small, the model tends to overfit. Conversely, an excessively large sample size increases the cost of data processing and training time, with only limited improvements in model performance. To determine the optimal number of training samples in this study, 85% of the sample data were used as the training set, and 15% were used as the test set. The maximum prediction bias relative to the number of samples is shown in the curve in Figure 10a, and the average prediction bias relative to the number of samples is illustrated in Figure 10b.



(**a**) (**b**)

Figure 10. Prediction error vs. sample size. (**a**) Maximum prediction error. (**b**) Average prediction error.

From Figure 10a, it can be seen that when the number of samples is less than 3500, the maximum prediction deviation gradually decreases as the sample size increases. However, once the number of samples exceeds 3500, the maximum prediction deviation stabilizes, eventually converging to 36.078 mm. Figure 10b shows a similar trend for the average prediction deviation; it decreases initially and then stabilizes as the sample size increases. When the number of samples exceeds 3500, the average prediction deviation stabilizes at approximately 28.208 mm. Based on this analysis, when the number of samples reaches 3500, the prediction error of the model begins to stabilize at a lower level, and further increasing the sample size does not significantly improve model performance.

Therefore, 3500 groups of sample data are selected for this experiment. Each group contains the values of seven joints of the robotic arm and their corresponding end position errors, covering the entire working space of the robotic arm. A total of 85% of the sample data are used for training, and 15%, i.e., 525 groups, is used for network prediction.

4.2. Comparison Experiments

4.2.1. Optimization Algorithm Performance Comparison

To verify the performance of the SBOA, comparison experiments are conducted using three algorithms: GA, PSO, and SBOA. To ensure the fairness of the experiment, the individual dimension dim, the number of populations N, the maximum number of iterations $T_{\max}$, and the fitness value function are unified across all algorithms. The optimal parameters for each optimization algorithm are determined through multiple sets of experiments.

Figure 11 illustrates the best fitness value curves for the four algorithms, GA, PSO, SBOA, and ISBOA, during the iterative process. The GA, PSO, SBOA, and ISBOA algorithms reach a steady state after 631, 557, 660, and 484 iterations, respectively, with ISBOA demonstrating a faster convergence speed and achieving the optimal fitness value.

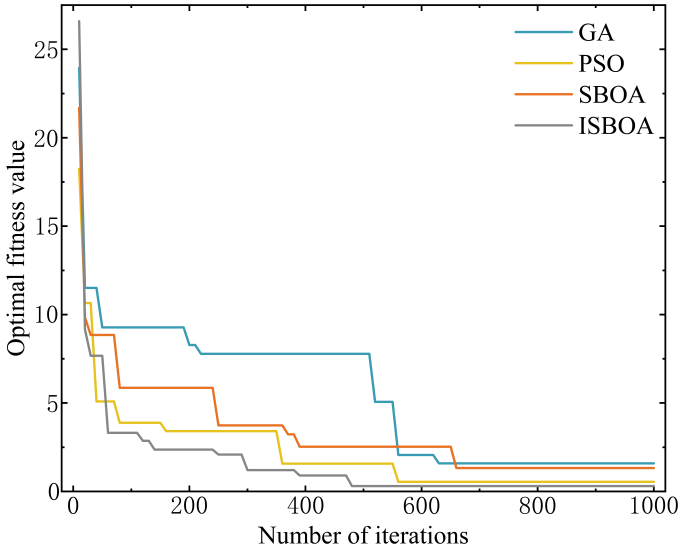


Figure 11. Adaptation curves of the four algorithms.

4.2.2. Comparison of Compensation Effects

The optimal solutions obtained by the four algorithms were used as the initial weights and biases for training the BP neural network. To ensure the fairness of the comparison, each model was trained with the number of samples required for its maximum prediction deviation to begin converging. After training, the test dataset was fed into the four models as well as the pure BP model to obtain the total error range, average absolute error, and standard deviation, as shown in Table 2.

Table 2. Analysis of compensation results for each algorithm.

	Total Tolerance (mm)	Average Absolute Error (mm)	Standard Deviation (mm)
Uncompensated	[52.381, 354.211]	190.792	89.525
BP	[1.697, 85.965]	33.791	13.608
GA-BP	[1.228, 54.572]	30.085	10.724
PSO-BP	[0.954, 48.331]	29.943	10.471
SBOA-BP	[1.132, 37.742]	29.139	9.864
ISBOA-BP	[0.803, 36.427]	28.831	9.423

As shown in Table 2, after error compensation, the total error for each method is significantly lower than before compensation, demonstrating the effectiveness and feasibility of these methods in improving the positioning accuracy of the rock-drilling robot arm. After compensation using the BP, GA-BP, PSO-BP, SBOA-BP, and ISBOA-BP algorithms, the maximum positioning errors at the end of the drill arm are reduced to 85.965 mm, 54.572 mm, 48.331 mm, 37.742 mm, and 36.427 mm, respectively. In addition, the standard deviations of the positioning errors at the end of the drill arm are reduced to 13.608 mm, 10.724 mm, 10.471 mm, 9.864 mm, and 9.423 mm. These results indicate that the method proposed in this paper is more effective in compensating for errors in the rock-drilling robot arm.

4.3. Experimental Validation

To verify the actual error compensation effect of the ISBOA-BP neural network model on the rock-drilling robotic arm, 50 sets of data were randomly selected from the robotic arm's motion space and compensated using the trained model. The experimental results are presented in Figure 12. Figure 12a–c show the positioning error compensation results of the robotic arm's end in the XYZ directions, respectively, while Figure 12d illustrates

the total positioning error of the robotic arm. The statistical results before and after error compensation are summarized in Table 3.

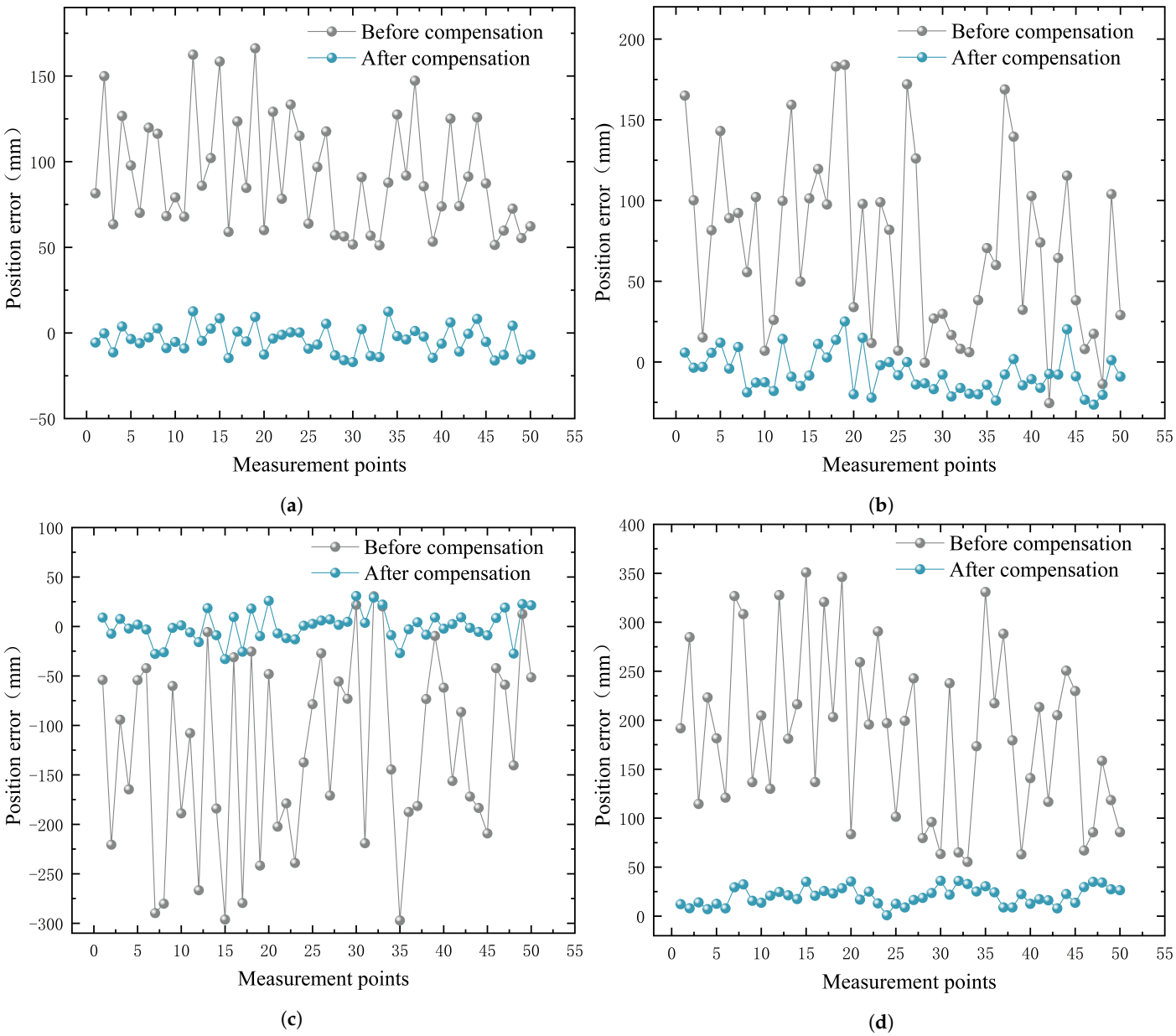


Figure 12. Position error before and after compensation. (**a**) Errors in x-direction. (**b**) Errors in y-direction. (**c**) Errors in z-direction. (**d**) Total position errors.

As shown in Figure 12, the robotic arm exhibits significant positioning errors in all directions before compensation, failing to meet the required accuracy standards. Moreover, the positioning errors along different axes show varying distribution characteristics: the x- and y-direction errors tend to shift positively, while the z-direction errors shift negatively. This variation arises from the differing impact of error sources on the positioning accuracy in the x-, y-, and z-directions. Notably, the positioning errors along the z-axis are considerably larger than those along the other axes. This is primarily because the z-axis is perpendicular to the ground, and gravity's effect on the connecting rods and joints becomes more pronounced, especially when the end-effector is extended farther, exacerbating this error. After compensation, both the magnitude of the error and its fluctuation range in each direction are significantly reduced, resulting in a more consistent and reasonable error distribution. As shown in Table 3, the maximum positioning errors of the robotic arm in

the x-, y-, and z-directions are reduced from 166.243 mm, 184.063 mm, and 297.178 mm to 16.974 mm, 26.443 mm, and 32.986 mm, respectively. Similarly, the average absolute errors decrease from 92.282 mm, 73.824 mm, and 129.112 mm to 7.321 mm, 12.305 mm, and 11.755 mm, respectively. The maximum total positioning error of the arm is reduced from 350.901 mm to 36.095 mm, and the average absolute positioning error decreases from 187.972 mm to 28.317 mm, indicating an improvement in the positioning accuracy of the rock-drilling arm by 84.94%. These results collectively demonstrate the effectiveness of the neural network-based error compensation method proposed in this paper.

Table 3. Position error before and after ISBOA-BP neural network compensation.

		Error Range (mm)	Average Absolute Error (mm)	Standard Deviation (mm)
X-error	Before	[51.175, 166.243]	92.282	32.644
	After	[−16.974, 12.633]	7.321	7.901
Y-error	Before	[−25.489, 184.063]	73.824	55.757
	After	[−26.443, 25.110]	12.305	12.431
Z-error	Before	[−297.178, 30.573]	129.112	94.319
	After	[−32.986, 30.905]	11.755	15.131
Total error	Before	[55.387, 350.901]	187.972	85.280
	After	[0.876, 36.095]	28.317	9.091

5. Conclusions

In this paper, an ISBOA-optimized BP neural network-based error compensation method for rock-drilling robotic arms is proposed to address the end positioning accuracy problem. The good point set strategy is used to initialize the ISBOA population position, facilitating a faster search for better solutions. ISBOA optimizes the initial weights and biases of the BP neural network, effectively preventing the BP neural network model from falling into local extrema and improving training speed and accuracy. The ISBOA-BP optimal model is then established based on the joint values of the rock-drilling robotic arm and the corresponding end position errors. Compared to the BP, GA-BP, PSO-BP, and SBOA-BP algorithms, the ISBOA-BP algorithm enhances both the convergence accuracy and the generalization ability of the neural network. The experimental results indicate that the average total positioning error is reduced from 187.972 mm to 28.317 mm, representing an 84.94% improvement in positioning accuracy. This significantly enhances the positioning accuracy of the rock-drilling robotic arm, meeting the requirements of practical engineering applications. However, differences in hardware conditions, error characteristics, and application environments exist among various robots. No in-depth research has been conducted on how to directly apply the neural network trained on the G3Zi rock-drilling robot to other types of robots through transfer learning for error compensation. Therefore, this will be the focus of future research.

Author Contributions: Q.X.: conceptualization, writing—original draft, data curation, formal analysis, validation, visualization. W.J.: writing—review and editing, conceptualization, data curation, formal analysis, validation. Y.L.: writing—review and editing. T.Z.: writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request. The data are not publicly available due to privacy.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ocak, I.; Bilgin, N. Comparative studies on the performance of a roadheader, impact hammer and drilling and blasting method in the excavation of metro station tunnels in Istanbul. *Tunn. Undergr. Space Technol.* **2010**, *25*, 181–187. [CrossRef]
2. Wu, L.; Yang, X.; Chen, K.; Ren, H. A Minimal POE-Based Model for Robotic Kinematic Calibration With Only Position Measurements. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 758–763. [CrossRef]
3. Lattanzi, L.; Cristalli, C.; Massa, D.; Boria, S.; Lepine, P.D.; Pellicciari, M. Geometrical calibration of a 6-axis robotic arm for high accuracy manufacturing task. *Int. J. Adv. Manuf. Technol.* **2020**, *111*, 1813 – 1829. [CrossRef]
4. Alam, M.M.; Ibaraki, S.; Fukuda, K.; Morita, S.; Usuki, H.; Otsuki, N.; Yoshioka, H. Inclusion of Bidirectional Angular Positioning Deviations in the Kinematic Model of a Six-DOF Articulated Robot for Static Volumetric Error Compensation. *IEEE/ASME Trans. Mechatron.* **2022**, *27*, 4339–4349. [CrossRef]
5. Zhao, N.; Ibaraki, S. Novel kinematic model of a SCARA-type robot with bi-directional angular positioning deviation of rotary axes. *Int. J. Adv. Manuf. Technol.* **2022**, *120*, 4901 – 4915. [CrossRef]
6. Jiang, W.; Wu, D.; Dong, W.; Ding, J.; Ye, Z.; Zeng, P.; Gao, Y. Design and Validation of a Nonparasitic 2R1T Parallel Hand-Held Prostate Biopsy Robot With Remote Center of Motion. *J. Mech. Robot.* **2023**, *16*, 051009. [CrossRef]
7. Kwon, J.; Choi, K.; Park, F.C. Kinodynamic Model Identification: A Unified Geometric Approach. *IEEE Trans. Robot.* **2021**, *37*, 1100–1114. [CrossRef]
8. Dmytriv, V.; Dmytriv, I.; Horodetskyy, I.; Hutsol, T.; Kukharets, S.; Cesna, J.; Bleizgys, R.; Pietruszynska, M.; Parafiniuk, S.; Kubon, M.; et al. A Method for Simulating the Positioning Errors of a Robot Gripper. *Appl. Sci.* **2024**, *14*, 6159. [CrossRef]
9. Lao, D.; Quan, Y.; Wang, F.; Liu, Y. Error Modeling and Parameter Calibration Method for Industrial Robots Based on 6-DOF Position and Orientation. *Appl. Sci.* **2023**, *13*, 10901. [CrossRef]
10. Lu, H.; Yang, Z.; Zhu, D.; Deng, F.; Guo, S. Dynamics Modeling and Parameter Identification for a Coupled-Drive Dual-Arm Nursing Robot. *Chin. J. Mech. Eng.* **2024**, *37*, 1–15. [CrossRef]
11. Yuan, P.; Chen, D.; Wang, T.; Cao, S.; Cai, Y.; Xue, L. A compensation method based on extreme learning machine to enhance absolute position accuracy for aviation drilling robot. *Adv. Mech. Eng.* **2018**, *10*, 1687814018763411. [CrossRef]
12. Qin, G.; Ji, A.; Cheng, Y.; Zhao, W.; Hongtao, P.; Shi, S.; Song, Y. Position error compensation of the multi-purpose overload robot in nuclear power plants. *Nucl. Eng. Technol.* **2021**, *53*, 2708–2715. [CrossRef]
13. Bayro-Corrochano, E.; Lechuga-Gutiérrez, L.; Garza-Burgos, M. Geometric techniques for robotics and HMI: Interpolation and haptics in conformal geometric algebra and control using quaternion spike neural networks. *Robot. Auton. Syst.* **2018**, *104*, 72–84. [CrossRef]
14. Zhou, W. Theory and Experiment of Industrial Robot Accuracy Compensation Method Based on Spatial Interpolation. *J. Mech. Eng.* **2013**, *49*, 42. [CrossRef]
15. Zhong, X.L.; Lewis, J.M.; N.-Nagy, F.L. Inverse robot calibration using artificial neural networks. *Eng. Appl. Artif. Intell.* **1996**, *9*, 83–93. [CrossRef]
16. Zeng, Y.; Tian, W.; Liao, W. Positional error similarity analysis for error compensation of industrial robots. *Robot. Comput.-Integr. Manuf.* **2016**, *42*, 113–120. [CrossRef]
17. Ma, S.; Deng, K.; Lu, Y.; Xu, X. Error compensation method of industrial robots considering non-kinematic and weak rigid base errors. *Precis. Eng.* **2023**, *82*, 304–315. [CrossRef]
18. Nguyen, H.N.; Zhou, J.; Kang, H.J. A calibration method for enhancing robot accuracy through integration of an extended Kalman filter algorithm and an artificial neural network. *Neurocomputing* **2015**, *151*, 996–1005. [CrossRef]
19. Wang, D.; Bai, Y.; Zhao, J. Robot manipulator calibration using neural network and a camera-based measurement system. *Trans. Inst. Meas. Control* **2012**, *34*, 105 – 121. [CrossRef]
20. Zhou, X.; Bai, W.; He, J.; Dai, J.; Liu, P.; Zhao, Y.; Bao, G. An Enhanced Positional Error Compensation Method for Rock Drilling Robots Based on LightGBM and RBFN. *Front. Neurorobotics* **2022**, *16*, 883816. [CrossRef]
21. Chen, X.; Zhang, Q.; Sun, Y. Evolutionary Robot Calibration and Nonlinear Compensation Methodology Based on GA-DNN and an Extra Compliance Error Model. *Math. Probl. Eng.* **2020**, *2020*, 3981081. [CrossRef]
22. Li-ning, S. Positioning error compensation for a parallel robot based on BP neural networks. *Opt. Precis. Eng.* **2008**, *16*, 878.
23. Gao, G.; Zhang, H.; San, H.; Wu, X.; Wang, W. Modeling and Error Compensation of Robotic Articulated Arm Coordinate Measuring Machines Using BP Neural Network. *Complex.* **2017**, *2017*, 5156264:1–5156264:8. [CrossRef]
24. Liu, H.; Yan, Z.; Xiao, J. Pose error prediction and real-time compensation of a 5-DOF hybrid robot. *Mech. Mach. Theory* **2022**, *170*, 104737. [CrossRef]
25. Zhang, X.; Hu, X.; Li, H.; Zhang, Z.; Chen, H. Research on six-joint industrial robotic arm positioning error compensation algorithm based on motion decomposition and improved CIWOA-BP neural network. *Proc. Inst. Mech. Eng. Part J. Mech. Eng. Sci.* **2024**, 09544062241264706. [CrossRef]
26. Jiang, G.; Luo, M.; Bai, K.; Chen, S. A Precise Positioning Method for a Puncture Robot Based on a PSO-Optimized BP Neural Network Algorithm. *Appl. Sci.* **2017**, *7*, 969. [CrossRef]
27. Chen, B.; Zhang, T.; Cong, L.; Ma, J.; Hu, W. Forward kinematics of body posture perception using an improved BP neural network based on a quantum genetic algorithm. *Laser Phys. Lett.* **2022**, *19*, 095201. [CrossRef]
28. Li, B.; Tian, W.; Zhang, C.; Hua, F.; Cui, G.; Li, Y. Positioning error compensation of an industrial robot using neural networks and experimental study. *Chin. J. Aeronaut.* **2021**, *35*, 346–360. [CrossRef]

29. Zhang, Y.; Fang, Q.; Xie, Y.; Zhang, W.; Yu, R. Enhancing Robot Calibration Through Reliable High-Order Hermite Polynomials Model and SSA-BP Optimization. *J. Comput. Inf. Sci. Eng.* **2023**, *24*, 021007. [CrossRef]
30. Fu, Y.; Liu, D.; Chen, J.; He, L. Secretary bird optimization algorithm: A new metaheuristic for solving global optimization problems. *Artif. Intell. Rev.* **2024**, *57*, 123. [CrossRef]
31. Keng, H.L.; Yuan, W. *Applications of Number Theory to Numerical Analysis*; Springer: Berlin/Heidelberg, Germany , 1981.
32. Li, Y.; Ni, Z.; Jin, F.; Li, J.; Li, F. Research on Clustering Method of Improved Glowworm Algorithm Based on Good-Point Set. *Math. Probl. Eng.* **2018**, *2018*, 8724084. [CrossRef]
33. Wang, C.; Liu, K.; Shen, P. A Novel Genetic Algorithm for Global Optimization. *Acta Math. Appl. Sin. Engl. Ser.* **2020**, *36*, 482–491. [CrossRef]
34. Zhao, P.; Liu, S. An improved symbiotic organisms search algorithm with good point set and memory mechanism. *J. Supercomput.* **2023**, *79*, 11170–11197. [CrossRef]
35. Zhu, F.; Li, G.; Tang, H.; Li, Y.; Lv, X.; Wang, X. Dung beetle optimization algorithm based on quantum computing and multi-strategy fusion for solving engineering problems. *Expert Syst. Appl.* **2023**, *236*, 121219. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.