# Temperature Compensation of Wind Tunnel Balance Signal Detection System Based on IGWO-ELM

Xiang Dong [1], Hu Xu [1], Huibin Cao [2,3,*], Tao Cui [4] and Yuxiang Sun [2,3,*]

1 School of Electrical Engineering and Automation, Anhui University, Hefei 230601, China; xdong@ahu.edu.cn (X.D.); z21301065@stu.ahu.edu.cn (H.X.)
2 Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China
3 Department of Science Island, University of Science and Technology of China, Hefei 230026, China
4 Chengdu Science and Technology Development Center of CAEP, Chengdu 610200, China; cuit0013@yinhe596.cn
* Correspondence: hbcao@iim.ac.cn (H.C.); yxsun@iim.ac.cn (Y.S.); Tel.: +86-0551-65592409 (H.C.)

**Abstract:** The wind tunnel balance signal detection system is widely employed in aerospace applications for the accurate and automated measurement of aerodynamic forces and moments. However, measurement errors arise under different environmental temperature. This paper addresses the issue of measurement accuracy under different temperature conditions by proposing a temperature compensation method based on an improved gray wolf optimization (IGWO) algorithm and optimized extreme learning machine (ELM). The IGWO algorithm is enhanced by improving the initial population position, convergence factor, and iteration weights of the gray wolf optimization algorithm. Subsequently, the IGWO algorithm is employed to determine the optimal network parameters for the ELM. The calibration decoupling experiment and high-low temperature experiment are designed and carried out. On this basis, ELM, GWO-ELM, PSO-ELM, GWO-RBFNN and IGWO-ELM are used for temperature compensation experiments. The experimental results show that IGWO-ELM has a good temperature compensation effect, reducing the measurement error from $20\%_{FS}$ to within $0.04\%_{FS}$. Consequently, the accuracy and stability of the wind tunnel balance signal detection system under different temperature environments are enhanced.

**Keywords:** wind tunnel balance; signal detection system; temperature compensation; gray wolf optimization algorithm; extreme learning machine

## 1. Introduction

The wind tunnel balance is an experimental device widely utilized in the study of aerodynamics [1]. It comprises a multi-component force measurement sensor designed to measure the aerodynamic forces and moments acting on various models, including spacecraft and automobiles. The balance incorporates multiple full-bridge circuits composed of strain gauges, enabling the sensing of changes in aerodynamic forces and moments in all dimensions [2]. During testing, aerodynamic loads can be determined by supplying excitation to the six-axis balance circuit using a high-precision power supply and capturing the response signal of the balance through the acquisition of a signal detection system [3]. In practical applications of the six-axis balance signal detection system, different working environments can have an impact on the detection system, particularly the influence of ambient temperature. When operating under extreme conditions, the wide range of ambient temperature can induce significant drift in the system's characteristics. The characteristic drifts will cause bias between measured values and real values, thereby affecting measurement accuracy [4]. Therefore, compensating for the temperature effect on the balance signal detection system is crucial, as the characteristic drift caused by temperature profoundly affects the accuracy of balance testing.

The effects of temperature on the balance signal detection system can be mainly attributed to two components: the elastic bodies and the signal conditioning circuit. The elastic bodies component comprises the elastic body and the strain gauges affixed to its surface. The temperature generated error is primarily caused by the thermal effects on the strain gauges and the disparate expansion coefficients between the strain gauges and elastic bodies. These factors play a crucial role in inducing temperature generated error in the elastic bodies. The signal conditioning circuit consists of various modules, including the excitation source, zero adjustment, anti-electromagnetic filter, amplifier circuit, low-pass filter, and reference source circuit. The entire signal conditioning circuit is susceptible to temperature effects, resulting in changes in the temperature coefficient and voltage drift of the amplifier. These temperature-induced variations, in turn, cause changes in the output value of the circuit. Additionally, apart from resistors and amplifiers, the excitation source, reference source, and capacitors within the conditioning circuit also exhibit distinct characteristics with temperature changes. Collectively, these factors contribute to the characteristic drift phenomenon observed in the balance signal detection system [5].

The compensation of the balance signal detection system for temperature can be achieved through hardware compensation or software compensation methods. Hardware compensation involves designing the circuit to eliminate temperature coefficients [6]. For instance, Chu et al. [7] adopted double thermocouples to measure the airflow temperature and proposed a time domain compensation method to compensate for the airflow temperature measurement results. Wu et al. [8] presented a method that utilizes pulsed light received by a Faraday rotating mirror to differentiate between the temperature and current. Lee et al. [9] involved a fluorescence quenching-based gas sensor and a resistive temperature detector (RTD) with an integrated thermometer for in situ temperature monitoring and compensation. However, hardware compensation is more costly, subject to material-processing constraints, and introduces new uncertainties to the system, limiting its compensation accuracy. On the other hand, software compensation involves constructing a compensation model to obtain the compensated output voltage value using the measured voltage and real-time temperature values. Software compensation is more cost-effective, flexible, and accurate than hardware compensation, making it more suitable for practical applications. Numerous software compensation methods have been proposed by researchers, such as least squares method (LSM), wavelet transform (WT) [10], and various artificial neural networks (ANNs) [11–14]. The LSM is more suitable for linear problems, and the error caused by temperature is usually non-linear. And WT requires a large number of parameters for model building, making it complex and challenging to implement. In recent years, artificial neural networks, such as backpropagation (BP) and least squares support vector machine (LSSVM) [15,16] models, have gained popularity for temperature compensation. However, these models tend to be more complex, prone to overfitting, and require a significant number of parameters. In contrast, the ELM model [17] is simple yet possesses strong generalization abilities, making it more suitable for addressing overfitting issues. For example, Li et al. [18] proposed an ELM-based temperature compensation method for piezoresistive differential pressure sensors. Another study by Long et al. introduced a temperature-compensated ELM model for low coupling and temperature compensation [19].

Although ELM demonstrates efficient and accurate performance in regression prediction, its random generation of input layer weights and implicit layer thresholds can lead to network instability. To address this issue, optimization algorithms are utilized to optimize the weights and thresholds of ELM, thereby reducing prediction error values, improving network performance, and enhancing the ELM prediction accuracy. In recent years, the field of metaheuristic algorithms has seen rapid development, with examples including particle swarm optimization (PSO) [20], the sparrow search algorithm (SSA) [21], the whale optimization algorithm (WOA) [22], the fireworks algorithm (FWA) [23], and the gray wolf optimization (GWO) [24,25], among others. GWO offers a faster optimization process compared to other algorithms, as it first derives the solutions and then ranks and

compares them to obtain the best solution. However, GWO has drawbacks, such as difficulty in handling a large number of variables and escaping local optima when solving large-scale problems [26]. In a previous study, Xie et al. proposed a three-stage update function for inertia weights and a dynamic update method for learning rate to enhance GWO performance by avoiding local optima [27]. Jin et al. proposed optimal Latin hypercube sampling initialization, nonlinear convergence factor, and dynamic weights for IGWO to enhance its accuracy in optimizing support vector regression (SVR) parameters [28]. Therefore, this paper aims to improve the gray wolf algorithm by enhancing three aspects: initial population location, convergence factor, and iteration weights.

Based on the aforementioned discussions, this paper employs the IGWO to optimize the ELM neural network, aiming to obtain the optimal weights and thresholds for the network. Building upon this, we propose the IGWO-ELM for the temperature compensation method. In this study, we focus on the calibration decoupling experiments, high- and low-temperature experiments, and temperature compensation experiments using a six-axis force/torque sensor and a balance signal detection system board applied to the space robotic arm as our research object. The experimental results demonstrate that IGWO-ELM exhibits a remarkable temperature compensation effect and demonstrates excellent stability.

The main contributions are as follows:

- This paper proposes three improvements to enhance the performance of the gray wolf algorithm:
    1. Improving the initialized population by employing circle chaotic mapping to enhance the diversity of the initial population, thus promoting exploration in the search space;
    2. Enhancing the convergence factor by utilizing a nonlinear function, which enhances the global search capability in the early stages of the algorithm and improves the convergence speed in the later stages;
    3. Accelerating the convergence speed of the algorithm towards the optimal solution by enhancing the dynamic weighting strategy.
- Presents a temperature compensation method for the wind tunnel balance signal detection system based on the IGWO algorithm and the ELM model. The proposed method aims to accurately predict and compensate for the errors induced by temperature in the system. By utilizing the IGWO algorithm and the ELM model, the temperature-related errors can be effectively mitigated, leading to improved measurement accuracy and reliability of the wind tunnel balance signal detection system.
- The calibration decoupling experiment and high–low temperature experiment are designed and carried out. On this basis, ELM, GWO-ELM, PSO-ELM, GWO-RBFNN and IGWO-ELM are used for temperature compensation experiments. The experimental results show that IGWO-ELM has a good temperature compensation effect, and the measurement error is reduced from $20\%_{FS}$ to less than $0.04\%_{FS}$.

## 2. Basic Algorithm

### 2.1. Extreme Learning Machine (ELM)

ELM is a feed-forward neural network model for machine learning that offers several advantages over other models, such as BPNN and RBFNN. ELM models [29] do not require the adjustment of structural parameters during training, making them simple and efficient. In ELM, the connection weights between the input layer and the hidden layer, as well as the thresholds of the hidden layer neurons, are randomly generated and do not require adjustments during training. The only requirement is to set the number of neurons in the hidden layer to obtain the optimal solution. The network structure is illustrated in Figure 1.
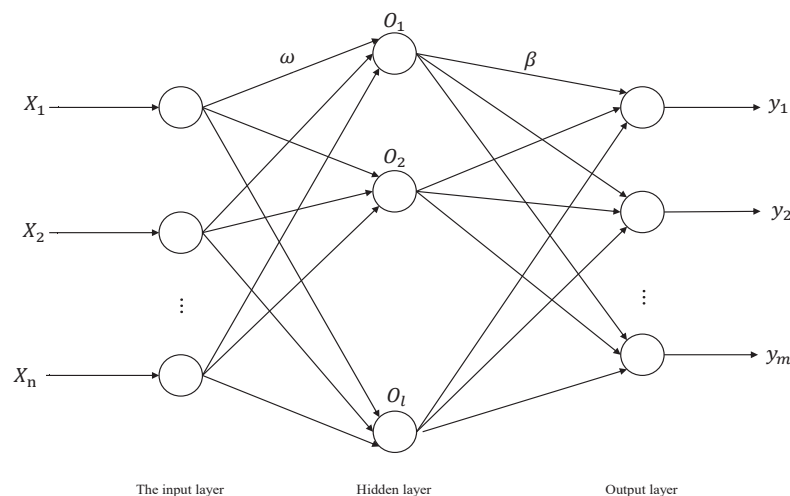
**Figure 1.** ELM network structure.

Suppose there are N training samples $(X_i, Y_i)$, where $1 \leq i \leq N$, $X_i = [X_{i1}, X_{i2}, \cdots, X_{in}]^T \in R^n$ is the input vector of the *i*th sample, and $Y_i = [y_{i1}, y_{i2}, \cdots, y_{im}]^T \in R^m$ is the output vector of the *i*th sample, then the ELM network can be represented as

$$\sum_l^f \beta_f g\left(W_f X_i + b_f\right) = T_i \tag{1}$$

where $g(x)$ is the activation function; $W_f = \left[w_{f1}, w_{f2}, \cdots, w_{fn}\right]^T$ is the input weight; $b_f$ is the threshold of the *f*-th hidden layer neuron; $\beta_f = \left[\beta_{f1}, \beta_{f2}, \cdots, \beta_{fm}\right]$ is the output weight; and $T_i = [t_{i1}, t_{i2}, \cdots, t_{im}]^T$ is the output vector of the *i*th sample.

The learning goal of ELM is to minimize the output error of the network as much as possible, expressed as

$$\sum_{i=1}^N ||T_i - Y_i|| = 0 \tag{2}$$

That is, there exists a suitable set of *W*, *b* and *β* such that

$$H\beta = Y \tag{3}$$

Among them, $H = \begin{bmatrix} g(W_1 X_1 + b_1) & \cdots & g(W_l X_1 + b_l) \\ \vdots & \ddots & \vdots \\ g(W_1 X_N + b_1) & \cdots & g(W_l X_N + b_l) \end{bmatrix}$ represents the implied layer output matrix of the network; $\beta = \left[\beta_1^T, \beta_2^T, \cdots, \beta_l^T\right]_{l \times m}^T$; and $Y = \left[Y_1^T, Y_2^T, \cdots, T_N^T\right]_{N \times m}^T$. The output weights *β* can be obtained by solving the least squares solution of the following system of equations:

$$\min_\beta = ||H\beta - Y|| \tag{4}$$

The solution is

$$\hat{\beta} = H^+ Y \tag{5}$$

where $H^+$ is the generalized inverse matrix of the output matrix *H* of the hidden layer.

### 2.2. Gray Wolf Optimization Algorithm (GWO)

The gray wolf optimization algorithm is a novel meta-heuristic algorithm that draws inspiration from the internal system and hunting patterns of gray wolves in nature. It aims to find optimal solutions by emulating the hunting behavior of gray wolf packs. In the natural environment, gray wolves occupy a higher position in the food chain and exhibit a

preference for group living. Within the gray wolf community, a strict hierarchical structure is maintained, as depicted in Figure 2.
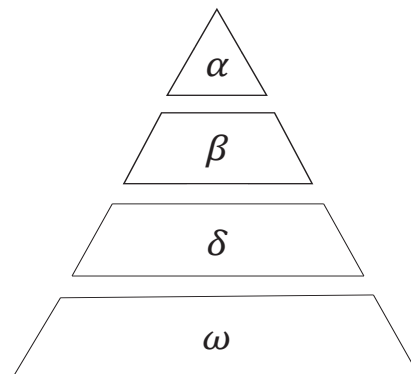


**Figure 2.** Gray wolf hierarchy.

The GWO classifies wolf packs into four categories. The first category consists of $\alpha$ wolves, who hold the dominant position and are primarily responsible for decision making regarding hunting and food distribution. The second category comprises $\beta$ wolves, who strictly follow the leadership of $\alpha$ wolves and serve as mediators for transmitting information. In the absence of an $\alpha$ wolf, $\beta$ wolves can assume control over all other wolves, except those in the first category. The third category encompasses $\delta$ wolves, who only dominate $\omega$ wolves and are mainly responsible for fundamental tasks within the pack, such as standing guard and scouting. Lastly, the fourth category comprises $\omega$ wolves, who occupy the lowest position in the pack's hierarchy and are tasked with executing orders from higher-ranking wolves and maintaining pack harmony. Gray wolves show collective predation, characterized by encirclement, hunting, and attack strategies. In the GWO algorithm, the first three optimal solutions correspond to the positions of $\alpha$, $\beta$, and $\delta$ wolves, while the remaining solutions represent positions of $\omega$ wolves. In each iteration, the optimal position is estimated using the first three optimal solutions. The gray wolves are then guided to randomly update their positions around these optimal solutions, gradually converging towards the global optima. Figure 3 illustrates the schematic diagram of the optimal search process.
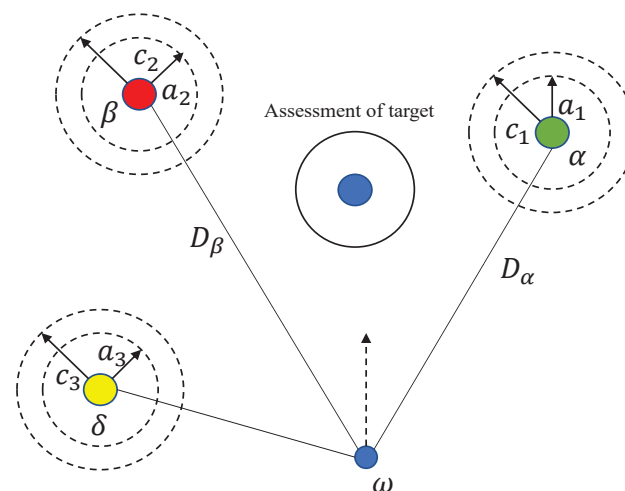


**Figure 3.** Schematic diagram of the gray wolf algorithm for finding advantages.

1.  Surrounding behavior
    Throughout the encirclement process, the separation distance, denoted as $D$, between the wolf pack and the prey, is captured by the formulation outlined in

Equation (6). To adapt their positions dynamically, the wolf pack undergoes position updates contingent upon the aforementioned distance as delineated in Equation (7). By manipulating the coefficient vectors $A$ and $C$, it becomes possible to guide the wolves towards the prey from varied vantage points. The determination of these coefficients is facilitated through the deployment of Equations (8) and (9):

$$D = \left| CX_p(t) - X(t) \right| \tag{6}$$

$$X(t+1) = X_p(t) - AD \tag{7}$$

$$A = 2ar_1 - a \tag{8}$$

$$C = 2r_2 \tag{9}$$

where $t$ is number of current iterations; $X_p$ is the position vector of the prey; and $D$ is the distance. $a$ is the convergence factor, slowly decreasing from 2 to 0, and $r_1, r_2$ are random numbers in $[0,1]$.

2. Hunting behavior

Once the wolf pack has successfully encircled the prey, the $\alpha$ wolf, $\beta$ wolf, and $\delta$ wolf position themselves closest to the prey. Under the leadership of these wolves, the entire pack advances towards the prey. The positional hunting behavior of the $\omega$ wolf can be described by the following mathematical model:

$$D_\alpha = C_1 X_\alpha(t) - X(t) \tag{10}$$

$$D_\beta = C_2 X_\beta(t) - X(t) \tag{11}$$

$$X_1(t+1) = X_\alpha(t) - A_1 D_\alpha \tag{12}$$

$$X_2(t+1) = X_\beta(t) - A_2 D_\beta \tag{13}$$

$$X_3(t+1) = X_\delta(t) - A_3 D_\delta \tag{14}$$

$$X(t+1) = \frac{X_1(t+1) + X_2(t+1) + X_3(t+1)}{3} \tag{15}$$

3. Attacking the prey

The objective of this phase is to capture the prey, which corresponds to obtaining the optimal solution. In the GWO algorithm, the process of approximating the prey is simulated by gradually decreasing the value of parameter $a$. As $a$ decreases, the elements of vector $A$ are constrained to the interval $[-a, a]$. When $|A| < 1$, the wolves can attack the prey; conversely, when $|A| > 1$, the wolves disperse in various directions, leading to a loss of the optimal position. This behavior highlights the tendency of the GWO algorithm to become trapped in local optima.

*2.3. Improved Gray Wolf Optimization Algorithm (IGWO)*

2.3.1. Optimize Initial Population Location

Chaotic motion possesses significant advantages, such as regularity, randomness, and ergodicity. To increase population diversity and prevent the algorithm from getting trapped in local optima during the optimization process, the concept of chaotic mapping is incorporated into the GWO algorithm. Various methods can be used for chaotic mapping,

including logistic mapping, tent mapping, and circle mapping [30]. However, it has been shown that the distribution of logistic mapping is nonuniform and biased towards the extremes, which adversely affects the speed and accuracy of finding the optimal solution [31]. Tent mapping [32] exhibits a more uniform distribution, but it is prone to convergence towards stationary points. In comparison, circle mapping demonstrates a more stable distribution with uniform values across the range as depicted in Figure 4. Therefore, in this paper, circle chaotic mapping is employed to initialize the population of the gray wolf algorithm.The expression for circle chaotic mapping is as follows:

$$X_{i+1} = mod(X_i + 0.2 - (0.5/2\pi)\sin(2\pi X_i), 1) \tag{16}$$

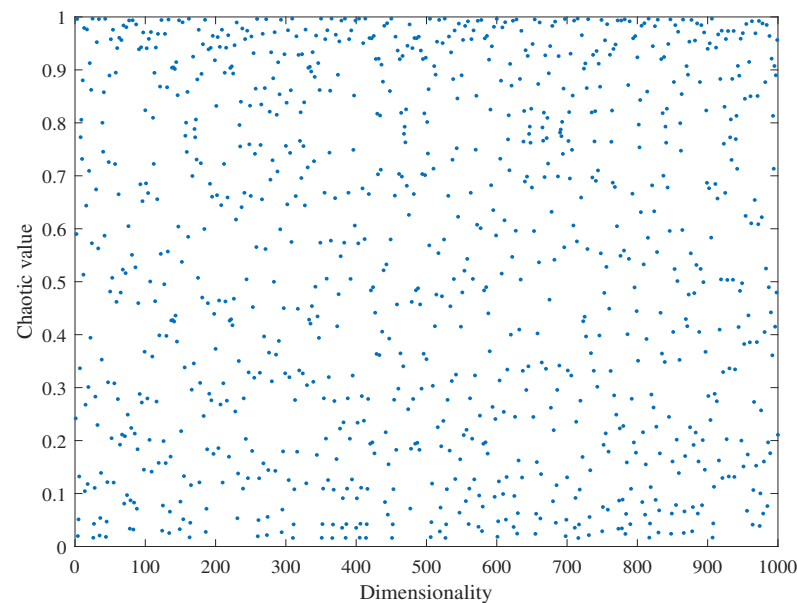where $X_i$ is the $i$-th dimensional random value.



**Figure 4.** Circle chaotic mapping scatter plot.

### 2.3.2. Optimized Convergence Factor

The GWO algorithm's ability to balance exploration and exploitation is influenced by the parameter $A$, and the convergence factor $a$ determines the value of $A$. In each iteration, the convergence factor $a$ is linearly reduced from 2 to 0. However, the GWO algorithm does not solve problems in a linear optimization process. Therefore, the linear decrease in $a$ does not fully adapt to the actual convergence search process and does not effectively coordinate global and local search. To address this issue, many researchers have proposed nonlinear improvements to the mathematical expression of $a$ to enhance the algorithm's convergence accuracy [33]. Nonlinearization of the convergence factor $a$ is a common approach used to solve this equilibrium problem. The improved expression for the convergence factor $a$ is as follows:

$$a = 2 - 2\left[\frac{1}{e-1} \times \left(e^{\frac{t}{T}} - 1\right)\right] \tag{17}$$

where $a$ is the convergence factor of the gray wolf algorithm; $e$ is the natural constant; $t$ is the number of iterations; and $T$ is the maximum number of iterations.

The improved convergence method involves non-linearly decreasing the convergence factor $a$ from 2 to 0 during the iteration process. This approach aligns more closely with the hunting behavior of gray wolves, where the iteration speed starts slow in the early stage and becomes faster in the later stage. This strategy enhances both the global and local search capabilities of the algorithm. By accelerating the convergence of the algorithm, it ensures better coordination between local and global search efforts.

### 2.3.3. Improvement of Gray Wolf Iteration Weights

In the original GWO algorithm, the $\alpha$, $\beta$, and $\delta$ wolves, selected based on their fitness function values, represent the current three optimal solutions. However, it is not reasonable to have these three wolves guide the other groups with the same strength during the optimization process. Such an approach would slow down the convergence of the algorithm, make it more prone to reaching local optima, and ultimately fail to produce the desired optimal solution. To address this limitation, an improved approach based on the concept of dynamic proportional weights is proposed [34]. In each iteration, the positions of the remaining wolves are dynamically guided by the distances between the three types of leading wolves and the prey. This dynamic guidance significantly enhances the algorithm's adaptability to the environment [35]. The $\omega$ wolves update their positions as follows, under the guidance of the $\alpha$, $\beta$, and $\delta$ wolves, respectively:

$$X_1 = X_\alpha - A_1 D_\alpha \tag{18}$$

$$X_2 = X_\beta - A_2 D_\beta \tag{19}$$

$$X_3 = X_\delta - A_3 D_\delta \tag{20}$$

Based on $X_1$, $X_2$, and $X_3$ generated by the algorithm in each iteration, the specific formula for updating the location of $\omega$ wolf based on dynamic weights is as follows:

$$\omega_1 = \frac{|X_1|}{|X_1| + |X_2| + |X_3|} \tag{21}$$

$$\omega_2 = \frac{|X_1|}{|X_1| + |X_2| + |X_3|} \tag{22}$$

$$\omega_3 = \frac{|X_1|}{|X_1| + |X_2| + |X_3|} \tag{23}$$

In Equations (21)–(23), $\omega_1$, $\omega_2$, and $\omega_3$ correspond to the learning weights of $\omega$ wolves to $\alpha$ wolves, $\beta$ wolves, and $\delta$ wolves, respectively. Therefore, the $\omega$ wolves finally update their positions in the following way:

$$X(t+1) = \omega_1 X_1 + \omega_2 X_2 + \omega_3 X_3 \tag{24}$$

From the weight Equations (21)–(23), we can see that $\omega_1$, $\omega_2$, and $\omega_3$ are constantly changing during each iteration, which means "dynamic weights". Therefore, the dynamic weights in this paper can be adjusted according to the actual convergence in each iteration so that they can be better adapted to the environment.

### 2.4. Improve GWO Algorithm to Optimize ELM

The improved GWO algorithm is employed to optimize the weights and thresholds of ELM, aiming to find the optimal network parameters. The weights of the input layer and the thresholds of the hidden layer serve as the location parameters for the gray wolf individuals. By applying the improved GWO algorithm, the optimal initial weights and thresholds are obtained, which are then used for training the network. This optimization process effectively mitigates the influence of randomly generated weights and thresholds on the network performance. Furthermore, GWO evaluates the fitness value based on the network error during each iteration, thereby enhancing the accuracy and stability of the model's predictions. The fitness function is defined as follows:

$$fit = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y_l}| \tag{25}$$

where $y_i$ is the true value, $\hat{y}_l$ is the predicted value, and $n$ is the number of samples. The IGWO-ELM algorithm flow is shown in Figure 5 and is as follows:

1.  Determining the network model structure and encoding the network weights $w$ and threshold $B$ to generate the gray wolf's position vector.
2.  Defining the dimensions of the weight variables $dim$, the population size $N$, the maximum number of iterations, and the upper and lower bounds of the search space $u_b$ and $l_b$.
3.  Initialization of gray wolf populations using circle mapping to increase the population diversity of the gray wolf algorithm.
4.  By optimizing the convergence factor and dynamically adjusting the gray wolf iteration weights, the optimal solution of fitness is searched in the solution space, and the location of the individual gray wolf with the optimal fitness value is used as the network initialization parameter.
5.  By determining whether the iteration termination condition holds, if it does, the iteration is terminated; otherwise, the execution continues.
6.  The parameters corresponding to the optimal gray wolf individuals are used as the optimal initial weights and thresholds of ELM to construct the IGWO-ELM network model.
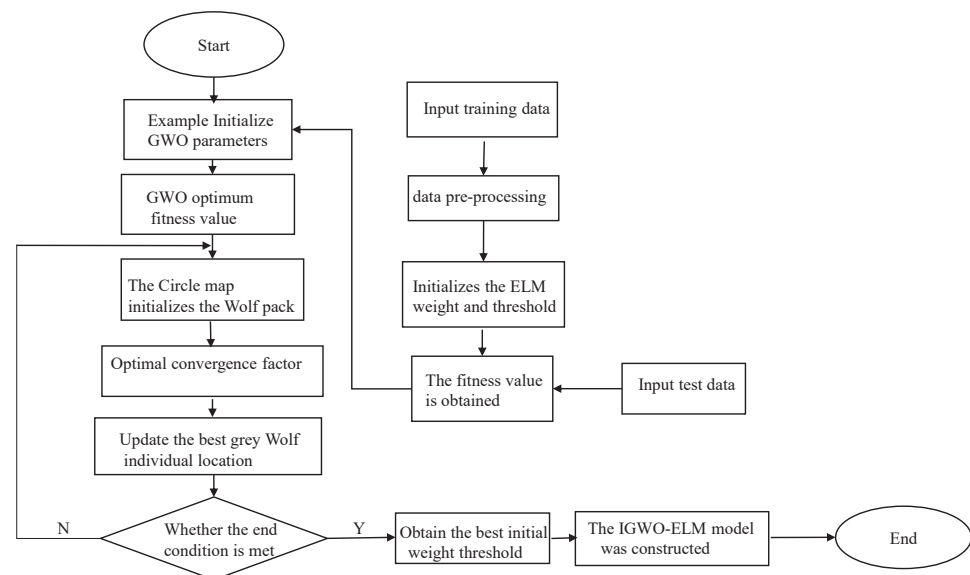


**Figure 5.** IGWO-ELM algorithm flow chart.

## 3. Experiment

The wind tunnel balance signal detection system comprises a six-axis sensor and a balance signal detection system board. In this study, we utilized the six-axis force/torque sensor developed by the Institute of Intelligent Mechanics (IIM) of the Chinese Academy of Sciences specifically designed for space robotic arm applications, along with a custom-designed signal detection system board. The sensor parameters are presented in Table 1 below. The balance signal detection system board consists of a signal conditioning module, excitation source module, AD acquisition module, and signal transmission module, with Ethernet transmission employed for signal transmission. The overall structure of the wind tunnel balance signal detection system is depicted in Figure 6 below.

**Table 1.** Six-axis force/torque sensor parameters.

| | |
|---|---|
| Measurement Range ($\pm$) | $F_x/F_y/F_z$: 1000 N, $M_x/M_y/M_z$: 40 Nm |
| Dimensional Parameters | Height: 53 mm, Diameter: 93 mm |
| Overload capacity | $\leq 300\%_{FS}$ |
| Zero output | $\leq \pm 0.5\%_{FS}$ |
| Supply voltage/power | DC 5V/2W |
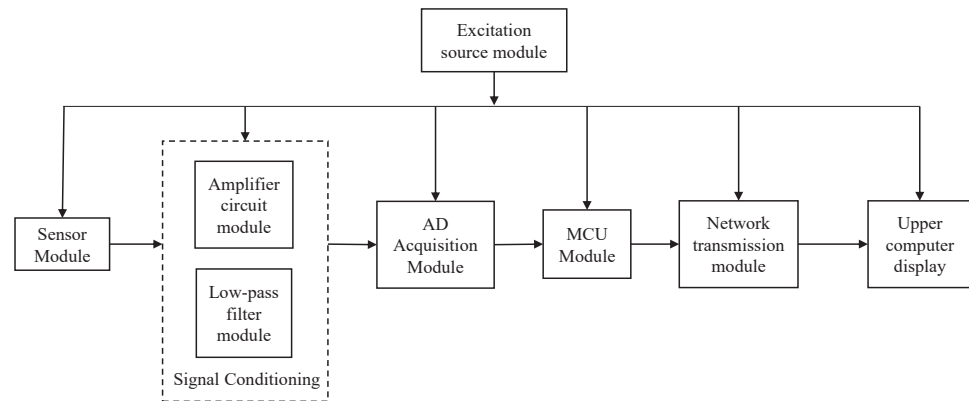


**Figure 6.** Wind tunnel balance signal detection system structure diagram.

### 3.1. Calibration and Decoupling Experiments

Calibration involves the adjustment of the equipment under test using standard metrological equipment to ensure accurate and reliable measurements. In our study, data are collected using a six-axis force/torque transducer, and precise calibration of this transducer is crucial for achieving accurate measurement performance. By calibrating the six-axis force/torque transducer, we establish the relationship between the applied force/torque and the corresponding output voltage for different force states of the transducer. This relationship can be expressed mathematically as follows:

$$F = W \cdot U + B \tag{26}$$

where the $W$ is the weight matrix, also known as the decoupling matrix, and the $B$ is the deviation vector.

A known standard force source is employed to apply a standard force/torque to the sensor in various directions, with the applied force/torque continuously varying from the minimum to the maximum measurement range. The output voltage of the sensor is simultaneously measured and recorded at each loading point. This process is repeated three times to obtain comprehensive calibration data. The calibration ranges for the platform are as follows: $F_x/F_y$—1000 N; $F_z$—2000 N; and $M_x/M_y/M_z$—50 Nm. There is an accuracy of $0.1\%_{FS}$.

After the calibration experiments are completed, the decoupling matrix $W$ deviation vector $B$ is calculated using a linear neural network as follows:

$$W = \begin{Bmatrix} -12.833 & 0.115 & -0.113 & 0.122 & -17.068 & -0.130 \\ 0.093 & -12.90 & -0.515 & 16.741 & 0.208 & 0.034 \\ -0.076 & -0.501 & -9.964 & 0.122 & 0.023 & 0.001 \\ 0.002 & -0.028 & -0.017 & 0.128 & 0.004 & -0.001 \\ 0.026 & 0.002 & -0.001 & -0.003 & 0.132 & 0.001 \\ 0.037 & -0.001 & -0.003 & 0.001 & 0.002 & -0.123 \end{Bmatrix} \tag{27}$$

$$B = [38203.083, -4828.718, 13017.101, -112.087, -201.395, -162.169] \tag{28}$$

Finally, the transfer expression (26) is obtained.

*3.2. High-Low Temperature Experiment*

To investigate the temperature characteristics of the wind tunnel balance signal detection system, the drift resulting from system measurement errors is analyzed through high-low temperature experiment. The obtained data are then used to train a model. In order to accurately study the temperature-induced drift characteristics, no force/torque is applied to the six-axis force/torque sensor, allowing the system output to vary solely due to temperature changes.

The wind tunnel balance signal detection system is placed inside a sealed high-low temperature experimental chamber, with the system powered by a 24 V DC power supply. The temperature is varied in increments of 20 °C, ranging from −40 to 80 °C. To ensure accurate temperature measurements, each sampling point is maintained for 2 h. The sampling points include −40, −20, 0, 20, 25, 40, 60, and 80 °C. The acquisition module transmits the voltage output values at each temperature to the computer via the network for storage and processing. The high–low temperature experimental setup is depicted in Figure 7.
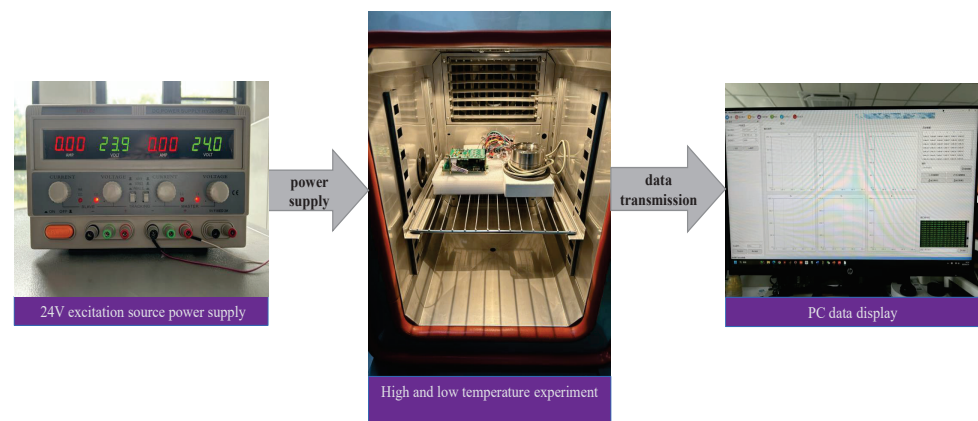


**Figure 7.** High–low temperature experiment demonstration.

In this paper, the sensor measurement error $E_m$ is employed as the evaluation index for quantifying the degree of temperature-induced drift in each bridge of the wind tunnel balance signal detection system. The defined equation is as follows:

$$E_m = \frac{U_T - U_{ref}}{U(FS)} \times 100\% \tag{29}$$

where $U_T$ is the measured value of the wind tunnel balance signal detection system at T°C; $U_{ref}$ indicates the measured value at the standard temperature, i.e., 25 °C; and $U(FS)$ indicates the range of the wind tunnel balance signal detection system.

The temperature-induced drift caused by measurement errors in the six dimensions can be observed using Equation (27) as depicted in Figure 8. From the figure, it is evident that the temperature-related errors exhibit predominantly nonlinear behavior. With increasing the temperature, the errors of $F_x$ and $M_z$ gradually transition from negative to zero and then increase; the errors of $F_z$ and $M_y$ follow a similar trend, but with opposite correlation; the errors of $F_y$ and $M_X$ remain relatively smaller as the temperature increases. Among these, $F_z$ exhibits the largest error, reaching up to $20\%_{FS}$, while $F_x$ and $M_y$ also have maximum errors of $10\%_{FS}$. In short, the temperature-induced error can reach a minimum of $0.6\%_{FS}$. Furthermore, temperature introduces significant nonlinearity-induced errors, significantly impacting the accuracy of the wind tunnel balance signal detection system.
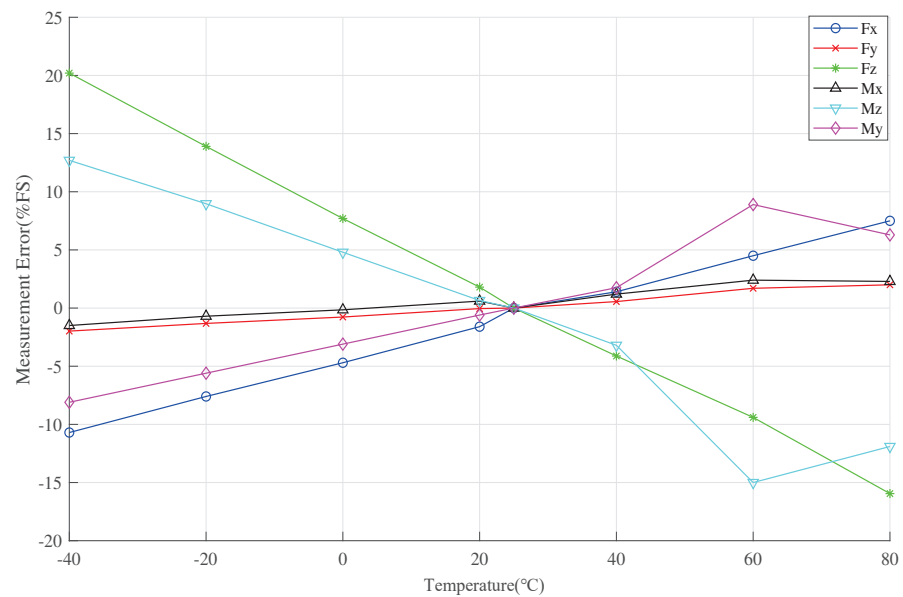
**Figure 8.** Uncompensated measurement error.

## 4. Temperature Compensation

### 4.1. Wind Tunnel Balance Signal Detection System Compensation Principle

To mitigate the impact of temperature-induced drift, temperature compensation is necessary for the wind tunnel balance signal detection system. The temperature compensation process is illustrated in Figure 9. Initially, the force and torque measured by the system are converted to a voltage output $U_0$. Subsequently, the voltage output and temperature are compensated using a model to obtain the thermal output $U_T$. Finally, the measured output is subtracted from the thermal output to obtain the actual output. The compensated output is determined using Equation (26).
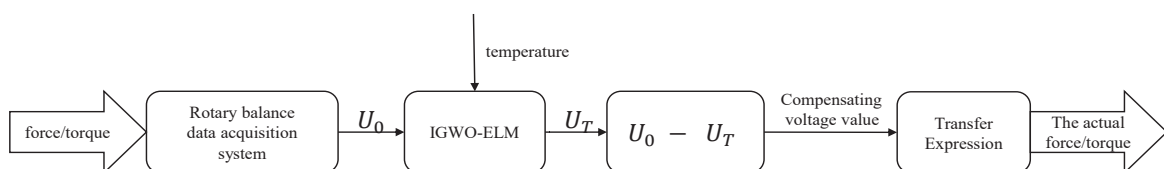


**Figure 9.** IGWO-ELM temperature compensation system diagram.

### 4.2. Compensation Model

Firstly, the acquired data are divided into training and test sets using the leave-out method. In this process, 80% of the acquired data is utilized to train the IGWO-ELM algorithm, while the remaining 20% is reserved to assess the accuracy of the compensation. Both the training and test sets consist of measured output and temperature values for each direction. Furthermore, to validate the prediction performance of the IGWO-ELM algorithm, BPNN, GWO-ELM, PSO-ELM, and GWO-RBFNN are trained using the same data samples, respectively.

The parameters of the GWO algorithm are set as follows: the number of populations is 30, the upper limit is 1, the lower limit is 0, and the number of iterations is 200. The parameters of the PSO algorithm are set as follows: $c_1 = c_2 = 1.49445$, the number of populations is 30, the maximum value of individuals $popmax = 1$, the minimum value of individuals $popmin = 0$, the maximum value of individuals velocity $V_{max} = 100$, the minimum value $V_{min} = -100$, and the maximum number of iterations $maxgen = 200$. The implicit layer

neurons of ELM are set to 8, the implicit layer activation function is sigmoid, and the RBFNN centroids are $-40$, $-20$, 0, 20, 25, 40, 60, and 80. All parameters are kept the same as much as possible.

In addition, the performance of the algorithm is evaluated by means of the mean square error (*MSE*) as a fitness function, which is calculated as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_l)^2 \tag{30}$$

where $y_i$ is the true value of the $i$-th sample, $\hat{y}_l$ is the predicted value of the $i$-th sample, and $n$ is the number of samples.

### 4.3. Compensation Results and Analysis

The compensation results for each dimension in the test sets are presented in Table 2 below. Among them, the test results of IGWO-ELM demonstrate the best performance, while ELM and GWO-RBFNN perform relatively worse. From the test results, it can be observed that the errors of IGWO-ELM in the four dimensions of $F_x$, $F_y$, $M_y$, and $M_z$ can be limited to within $0.01\%_{FS}$. Furthermore, the test results of IGWO-ELM outperform those of single GWO-ELM, showcasing effective compensation for temperature-induced errors. It is evident that under the same network structure conditions, the performance of GWO-ELM is not as satisfactory, while the performance of GWO-ELM is superior to that of GWO-RBFNN.

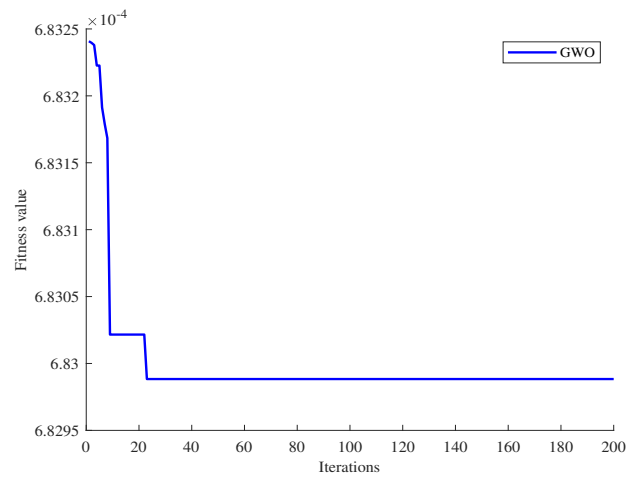**Table 2.** Comparison of prediction errors of different algorithms.

| Parameters | ELM | GWO-ELM | IGWO-ELM | PSO-ELM | GWO-RBFNN |
|---|---|---|---|---|---|
| $F_x$ | $3.7567 \times 10^{-5}$ | $3.3718 \times 10^{-5}$ | $3.0089 \times 10^{-5}$ | $3.5404 \times 10^{-5}$ | $3.7606 \times 10^{-5}$ |
| $F_y$ | $1.8891 \times 10^{-5}$ | $1.7274 \times 10^{-5}$ | $1.3948 \times 10^{-5}$ | $1.7350 \times 10^{-5}$ | $1.6507 \times 10^{-5}$ |
| $F_z$ | $3.5412 \times 10^{-4}$ | $3.4942 \times 10^{-4}$ | $2.8792 \times 10^{-4}$ | $3.2449 \times 10^{-4}$ | $3.2074 \times 10^{-4}$ |
| $M_x$ | $2.5310 \times 10^{-4}$ | $2.5227 \times 10^{-4}$ | $1.9278 \times 10^{-4}$ | $2.4042 \times 10^{-4}$ | $2.2995 \times 10^{-4}$ |
| $M_y$ | $2.2462 \times 10^{-5}$ | $1.9885 \times 10^{-5}$ | $1.7164 \times 10^{-5}$ | $1.9339 \times 10^{-5}$ | $1.9990 \times 10^{-5}$ |
| $M_z$ | $7.2498 \times 10^{-5}$ | $6.8145 \times 10^{-5}$ | $6.2343 \times 10^{-5}$ | $6.7122 \times 10^{-5}$ | $7.2946 \times 10^{-5}$ |

By examining the data presented in Table 2, it becomes evident that the proposed IGWO-ELM algorithm in this study exhibits superior performance in compensating for temperature variations as reflected by the smaller error indicators compared to other algorithms. Of particular importance is the significant reduction in compensation errors for $F_x$, $F_y$, $M_y$, and $M_z$, which are decreased by 20% when compared to the original standard ELM algorithm. This signifies the effectiveness of the proposed IGWO-ELM algorithm in optimizing temperature compensation for the wind tunnel balance data acquisition system. As a result, the algorithm utilizing IGWO-ELM outperforms GWO-ELM, PSO-ELM, and GWO-RBFNN in achieving accurate temperature compensation.

From the data presented in the table above, it is evident that the iterations in the $F_y$ dimension yield the best results. Therefore, $F_y$ is chosen as an example to evaluate the convergence and ability to escape local optima of the IGWO algorithm after compensation. The iterative convergence curve of the algorithm is depicted in Figure 10, where the horizontal axis represents the number of iterations, and the vertical axis represents the fitness value. From the figure, it can be observed that the IGWO algorithm shows faster convergence and a smooth curve in the temperature compensation model. A lower fitness value indicates higher optimization search accuracy. The appearance of the curve's inflection point signifies faster convergence speed. The smooth curve suggests that the algorithm excels in escaping local optima. After conducting several experiments, a comparison between the predicted and actual values of the test samples can be obtained as shown in Figure 11.
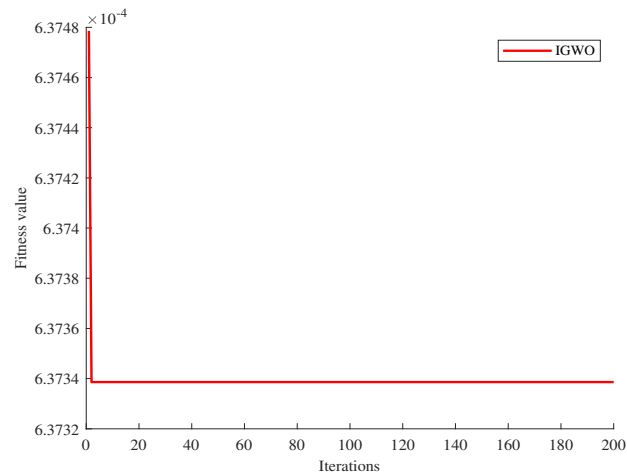
(**a**)



(**b**)

**Figure 10.** Adaptability curve. (**a**) Improved GWO adaptation curve. (**b**) GWO adaptation curve.
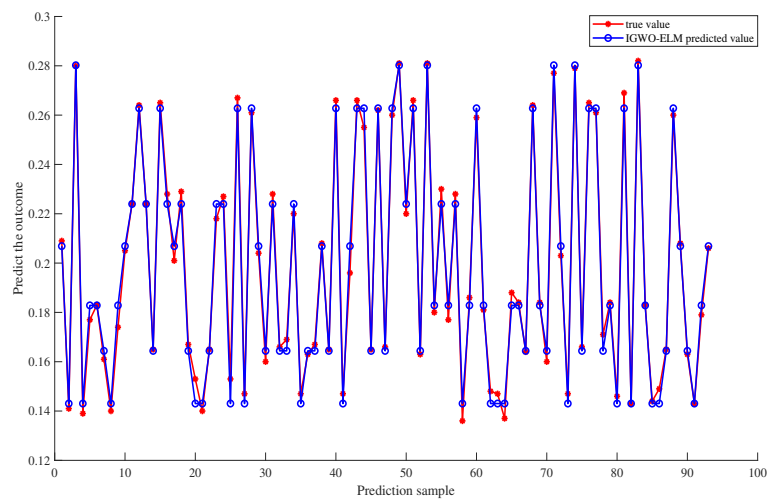


**Figure 11.** Comparison of predicted values and actual graphs.

The measurement errors of the wind tunnel balance signal detection system after compensation using IGWO-ELM are illustrated in Figure 12. Overall, the measurement errors in

all dimensions are significantly reduced after the application of IGWO-ELM compensation. The best compensation effect is achieved in the $F_y$ dimension, with a value of $2.5 \times 10^{-3}\%_{FS}$. The errors in all other dimensions are within $0.04\%_{FS}$. As a result, the measurement data obtained after the implementation of the IGWO-ELM compensation algorithm are largely independent of the temperature, reducing the measurement error from $20\%_{FS}$ to within $0.04\%_{FS}$, effectively controlling the measurement error caused by temperature.
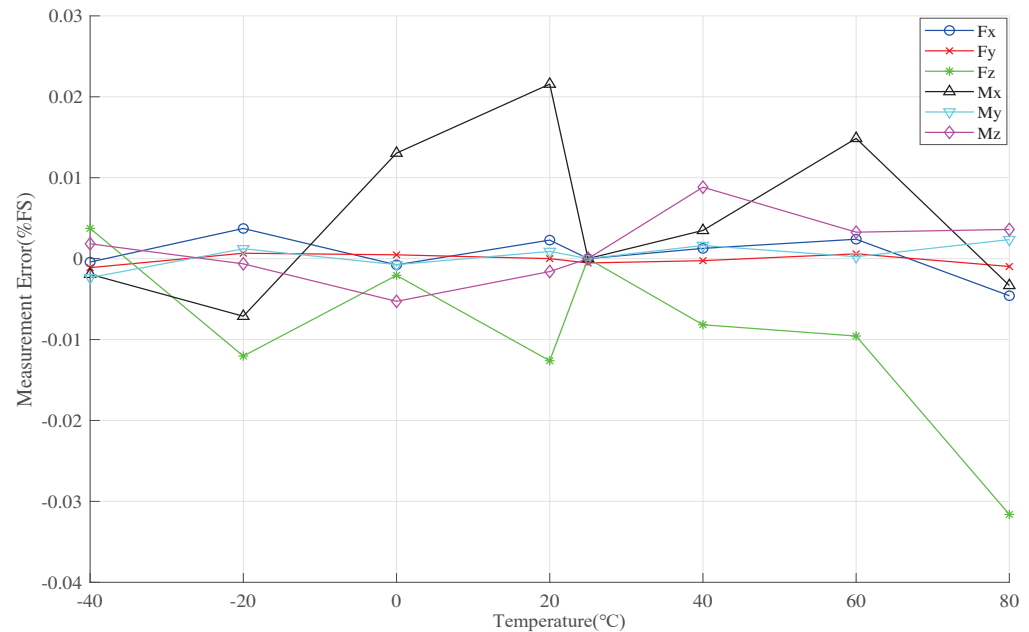


**Figure 12.** Measurement error after compensation by IGWO-ELM algorithm.

## 5. Conclusions

In this paper, aiming at the temperature error problem of the wind tunnel balance signal detection system, a temperature compensation method is proposed. The method is based on an optimized extreme learning machine using an IGWO algorithm. The research focuses on a wind tunnel balance signal detection system that consists of a six-axis force/torque sensor applied to a space robotic arm and a signal detection system board. High–low temperature experiments are conducted to collect experimental data. For temperature compensation, various algorithms, including ELM, GWO-ELM, PSO-ELM, GWO-RBFNN, and IGWO-ELM, are employed and compared. The experimental results demonstrate that IGWO-ELM outperforms other algorithms in terms of prediction ability and reduces the temperature-induced error to within $0.04\%_{FS}$. By improving the initialization of the gray wolf population, optimizing the convergence factor, and adapting iteration weights, the IGWO-ELM algorithm exhibits improved stability, is less likely to converge to local optimal solutions, and determines the optimal network parameters for the ELM. In short, the experimental results confirm that the proposed IGWO-ELM temperature compensation algorithm effectively addresses the temperature error issue and provides better temperature compensation performance with enhanced stability.

Although the temperature compensation achieved by the IGWO-ELM algorithm is superior, its controllable performance is relatively weak, and its ability to adapt to different characteristics of the dataset is not highly effective. Future research can focus on enhancing the controllability of the IGWO-ELM algorithm, particularly in relation to its adjustment based on the specific characteristics of the dataset. This investigation will contribute to further improving the performance and versatility of the IGWO-ELM algorithm in temperature compensation applications.

# References

1. Zhu, H.; Yang, Z. Numerical study on scale of high speed train model for wind tunnel testing. In Proceedings of the 2011 International Conference on Electrical and Control Engineering (ICECE), Yichang, China, 16–18 September 2011; pp. 725–728. [CrossRef]
2. Lin, K.; Xiao, S.; Liu, H.; Zhou, A. Investigation on dynamic performance of wind turbines using different scaling methods in wind tunnel tests. *Eng. Struct.* **2023**, *284*, 115961. [CrossRef]
3. Campagnolo, F.; Schreibe, J.; Bottasso, C.L. Wake Deflection Control with Wind Direction Changes: Wind Tunnel Comparison of Different Wind Farm Flow Models. In Proceedings of the 2020 American Control Conference (ACC), Denver, CO, USA, 1–3 July 2020; pp. 4817–4823. [CrossRef]
4. Li, X.; Gao, L.; Cao, H.; Sun, Y.; Jiang, M.; Zhang, Y. A Temperature Compensation Method for aSix-Axis Force/Torque Sensor Utilizing Ensemble hWOA-LSSVM Based on Improved Trimmed Bagging. *Sensors* **2022**, *22*, 4809. [CrossRef] [PubMed]
5. Zhang, J.; Liu, W.; Gao, L.; Zhang, Y.; Tang, W. Design, Analysis and Experiment of a Tactile Force Sensor for Underwater Dexterous Hand Intelligent Grasping. *Sensors* **2018**, *18*, 2427. [CrossRef] [PubMed]
6. Do, C.; SeshiaActive, A.A. SeshiaActive Temperature Compensation for MEMS Capacitive Sensor. *IEEE Sens. J.* **2021**, *21*, 18588–18592. [CrossRef]
7. Chu, J.; Wang, S.; Yang, G.; Wang, W. Dynamic Compensation Method for Temperature Measurement Error of High Temperature and High-speed Airflow. In Proceedings of the 2022 4th International Conference on Intelligent Control, Measurement and Signal Processing (ICMSP), Hangzhou, China, 8–10 July 2022; pp. 94–98. [CrossRef]
8. Wu, J.; Zhang, X.; Chen, L.; Wu, B.; Peng, C. Research on All-Fiber Dual-Modulation Optic Current Sensor Based on Real-Time Temperature Compensation. *IEEE Photonics J.* **2022**, *14*, 6822609. [CrossRef]
9. Lee, Y.-C.; Cheng, S.-W.; Fang, W. Monolithic Integrated CMOS-MEMS Fluorescence Quenching Gas Sensor and Resistive Temperature Detector (RTD) for Temperature Compensation. In Proceedings of the 2019 20th International Conference on Solid-State Sensors, Actuators and Microsystems & Eurosensors XXXIII (TRANSDUCERS & EUROSENSORS XXXIII), Berlin/Heidelberg, Germany, 23–27 June 2019; pp. 1293–1296. [CrossRef]
10. Ni, Z. Error Compensation Algorithm for UAV Telemetry Temperature Data. In Proceedings of the 2021 14th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Shanghai, China, 23–25 October 2021; pp. 1–5. [CrossRef]
11. Zhao, B.; Shi, W.; Sun, D.; Li, J.; Li, F.; Tan, J. Extending Velocity Sensor Bandwidth by Compensating Temperature Dependency Based on BP Neural Network. *IEEE Access* **2019**, *7*, 154889–154898. [CrossRef]
12. Wang, H.; Zeng, Q.; Zhang, Z.; Wang, H. Research on Temperature Compensation of Multi-Channel Pressure Scanner Based on an Improved Cuckoo Search Optimizing a BP Neural Network. *Micromachines* **2022**, *13*, 1351. [CrossRef]
13. Liu, C.; Wang, D.; Li, C.; Su, K.; Li, D.; Yu, D.; Wang, L.; Shi, L.; Jin, J. Fiber optic current sensor temperature compensation through RBF neural network.In Proceedings of the Conference on Optical Information and Network, Beijing, China, 5 November 2020. [CrossRef]
14. Sun, Y.; Liu, Y.; Liu, H. Temperature compensation for six-dimension force/torque sensor based on Radial Basis Function Neural Network. In Proceedings of the 2014 IEEE International Conference on Information and Automation (ICIA), Hailar, China, 28–30 July 2014; pp. 789–794. [CrossRef]
15. Sun, Y.; Liu, Y.; Liu, H. Temperature Compensation for a Six-Axis Force/Torque Sensor Based on the Particle Swarm Optimization Least Square Support Vector Machine for Space Manipulator. *IEEE Sens. J.* **2016**, *16*, 798–805. [CrossRef]
16. Li, J.; Hu, G.; Zhou, Y.; Zou, C.; Peng, W.; Alam SM, J. A Temperature Compensation Method for Piezo-Resistive Pressure Sensor Utilizing Chaotic Ions Motion Algorithm Optimized Hybrid Kernel LSSVM. *Sensors* **2016**, *16*, 1707. [CrossRef]

17. Zheng, B.-D.; Liu, W; Lv, M.; Wang, R.; Dai, H.-D. Segmental compensation of FOG temperature error based on ELM prediction model.In Proceedings of the 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 22–24 May 2021; pp. 6286–6290. [CrossRef]
18. Li, J.; Hu, G.; Zhou, Y.; Zou, C.; Peng, W.; Alam SM, J. Study on Temperature and Synthetic Compensation of Piezo-Resistive Differential Pressure Sensors by Coupled Simulated Annealing and Simplex Optimized Kernel Extreme Learning Machine. *Sensors* **2017**, *17*, 894. [CrossRef]
19. Long, J.; Liang, Q.; Sun, W.; Wang, Y.; Zhang, D. Ultrathin Three-Axis FBG Wrist Force Sensor for Collaborative Robots. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 3519115. [CrossRef]
20. Cai, W.; Yang, J.; Yu, Y.; Song, Y.; Zhou, T.; Qin, J. PSO-ELM: A Hybrid Learning Model for Short-Term Traffic Flow Forecasting. *IEEE Access* **2020**, *8*, 6505–6514. [CrossRef]
21. Lou, A.; Liu, B. Temperature prediction of roller kiln based on mechanism and SSA-ELM data-driven integrated model. In Proceedings of the 2022 China Automation Congress (CAC), Xiamen, China, 25 November 2022; pp. 5598–5603. [CrossRef]
22. Lian, Z.; Duan, L.; Qiao, Y.; Chen, J.; Miao, J.; Li,M. The Improved ELM Algorithms Optimized by Bionic WOA for EEG Classification of Brain Computer Interface. *IEEE Access* **2021**, *9*, 67405–67416. [CrossRef]
23. Gou, Y.; Kong, W.; Wang, L. Temperature Error Compensation Method for FBG Sensor in Electric Power System Based on Fireworks Algorithm. In Proceedings of the 2022 2nd International Conference on Electrical Engineering and Control Science (IC2ECS), Nanjing, China, 16–18 December 2022; pp. 60–63. [CrossRef]
24. Lu, J.; Gan, X.; Wei, Y.; Li, S. Research on A Route Planning Method for Military Transport Aircraft Based on GWO Algorithm. In Proceedings of the 2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), Changchun, China, 24–26 February 2023; pp. 1359–1364. [CrossRef]
25. Anbumani, K.; Ranihemamalini, R.; Pechinathan, G. GWO based tuning of PID controller for a heat exchanger process. In Proceedings of the 2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS), Chennai, India, 4–5 May 2017; pp. 417–421. [CrossRef]
26. Dong, L.; Yuan, X.; Yan, B.; Song, Y.; Xu, Q.; Yang, X. An Improved Grey Wolf Optimization with Multi-Strategy Ensemble for Robot Path Planning. *Sensors* **2022**, *22*, 6843. [CrossRef]
27. Xie, F.; Liang, G.; Chien, Y.-R. Highly Robust Adaptive Sliding Mode Trajectory Tracking Control of Autonomous Vehicles. *Sensors* **2023**, *23*, 3454. [CrossRef]
28. Jin, S.; Cao, M.; Qian, Q.; Zhang, G.; Wang, Y. Study on an Assembly Prediction Method of RV Reducer Based on IGWO Algorithm and SVR Model. *Sensors* **2023**, *23*, 366. [CrossRef]
29. Tang, J.; Deng, C.; Huang, G.-B. Extreme Learning Machine for Multilayer Perceptron. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 909–821. [CrossRef] [PubMed]
30. Li, M.; Sun, X.; Li, W.; Wang, Y. Improved Chaotic Particle Swarm Optimization using circle map for training SVM. In Proceedings of the 2009 Fourth International on Conference on Bio-Inspired Computing, Beijing, China, 16–19 October 2009; pp. 1–7. [CrossRef]
31. Sinhal, R.; Ansari, I.A. Tunable Q-Factor Wavelet Transform-Based Robust Image Watermarking Scheme Using Logistic Mapping and Antlion Optimization. *Circuits Syst. Signal Process.* **2022**, *41*, 6370–6410. [CrossRef]
32. Gao, Q.; Zheng, W.; Tao, X.;Teng, J. Research on Substation Video Surveillance Data Encryption Based on Improved Tent Chaos Map. In Proceedings of the 2022 4th International Conference on Smart Power & Internet Energy Systems (SPIES), Beijing, China, 9–12 December 2022; pp. 2256–2259. [CrossRef]
33. Luo, Y.; Qin, Q.; Hu, Z.; Zhang, Y. Path Planning for Unmanned Delivery Robots Based on EWB-GWO Algorithm. *Sensors* **2023**, *23*, 1867. [CrossRef]
34. Hou, Y.; Gao, H.; Wang, Z.; Du, C. Improved Grey Wolf Optimization Algorithm and Application. *Sensors* **2022**, *22*, 3810. [CrossRef]
35. Tang, M.; Yi, J.; Wu, H.; Wang, Z. Fault Detection of Wind Turbine Electric Pitch System Based on IGWO-ERF. *Sensors* **2021**, *21*, 6218. [CrossRef] [PubMed]