Active Disturbance Rejection Control of Bearingless Permanent Magnet Slice Motor Based on RPROP Neural Network Optimized by Improved Differential Evolution Algorithm

Chuang Sun and Huangqiu Zhu, *Member, IEEE*

***Abstract*—In order to realize the strong antidisturbance capability and the precise control of suspension forces in a bearingless permanent magnet slice motor (BPMSM), an active disturbance rejection control (ADRC) strategy based on the combination of resilient backpropagation neural network (RPROPNN) and improved differential evolution (IDE) algorithm is proposed. Based on the mathematical model of the BPMSM, the first-order ADRC controller of the rotating part and the second-order ADRC controller of the suspension part are designed, respectively, for the BPMSM according to the different orders of the BPMSM system. Then, the elite group bootstrap mechanism and parameter adaptive techniques are introduced to improve the DE algorithm to speed up its convergence and enhance the global search capability, and the initial weights of RPROPNN are optimized using the IDE algorithm to obtain the best network model. According to the feedback information of the BPMSM, the optimization mechanism and self-learning capability of RPROPNN are used to adjust the parameters of ADRC to reduce the dependence of ADRC on parameters. The comparative experimental results indicate that the proposed ADRC optimization design method has stronger robustness when disturbances occur, and the method is feasible and effective.**

***Index Terms*—Active disturbance rejection control (ADRC), bearingless permanent magnet slice motor (BPMSM), improved differential evolution (IDE), parameter optimization, resilient backpropagation neural network (RPROPNN).**

I. Introduction

DUE to the small axial length of the permanent magnet rotor in relation to the radius of the rotor and its flake structure, a bearingless permanent magnet slice motor (BPMSM) can achieve passive suspension of the rotor in three degrees of freedom by relying on magnetic resistance force. Only two radial degrees of freedom need to be actively controlled to achieve reliable and stable motor operation, simplifying the mechanical structure and greatly reducing the difficulty of controlling the bearingless motor. At the same time, the BPMSM not only has the advantages of simple structure, high power density, and high efficiency but also has excellent characteristics, such as no contact, no loss, no lubrication, no pollution, and long life. Therefore, centrifugal pumps driven by the BPMSM, as shown in Fig. 1(a), can realize contactless and pollution-free transmission of liquids and have broad application prospects in special transmission fields with high requirements for precision and cleanliness, such as biomedicine, semiconductor industry, aerospace, and other areas [1], [2], [3].

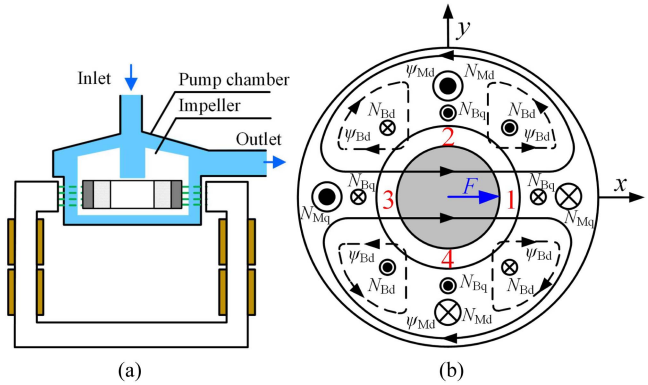

Fig. 1. BPMSM pump system. (a) Structural diagram of the BPMSM pump system. (b) Generation principle of active suspension force.

The BPMSM is a multivariable, nonlinear, strongly coupled complex system, and the traditional proportional integral (PI) vector control system is susceptible to load changes and motor parameter changes, and the controller robustness is poor. In recent years, with the rapid development of control theory, some advanced algorithms have been proposed to improve the robustness of the controller to external disturbances and system parameter changes, such as sliding-mode control [4], [5], [6], model reference adaptive control [7], [8], model predictive control [9], active disturbance rejection control (ADRC), and model-free control [10], [11]. In [4], a sliding-mode variable structure control method based on inverse system decoupling is proposed, which reduces the influence of rotor parameters and load disturbances on the control performance. In [7], a fuzzy

Manuscript received 16 July 2023; revised 20 October 2023; accepted 3 December 2023. Date of publication 7 December 2023; date of current version 26 January 2024. This work was supported in part by the National Natural Science Foundation of China under Grants 62273168 and 61973144. Recommended for publication by Associate Editor R. Kennel. *(Corresponding author: Huangqiu Zhu.)*

The authors are with the School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, China (e-mail: sunchuang122425@163.com; zhuhuangqiu@ujs.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPEL.2023.3340265.

Digital Object Identifier 10.1109/TPEL.2023.3340265

model reference adaptive identification method of rotor resistance is proposed. The rotor resistance identification method is combined with the inverse system dynamic decoupling control method to improve the robustness of the bearingless induction motor system to the variation in rotor resistance parameters. The above model-based control methods usually provide the accurate control, but their control accuracy is highly dependent on the accuracy of the system model, and the controller design is more complex and computationally intensive. In [11], a model-free adaptive control (MFAC) based on the full-form dynamic linearization technique is proposed for current control. MFAC only depends on the real-time measurement data of the controlled system and does not rely on any mathematical model information of the controlled system, so the computational burden is small and the robustness is strong.

ADRC is a nonlinear control for uncertain systems, which can be estimated and compensated by attributing all the system's uncertainties to the system's total disturbance, utterly independent of the specific mathematical model of the controlled object. Therefore, it is considered an effective method for solving nonlinear system control problems in the control field. In [12], the ADRC controller is applied to the decoupled linear subsystem, which effectively improves the robustness of the control system to changes in motor parameters and load. In [13], a displacement ADRC controller is designed to improve the radial suspension performance of the BPMSM.

In [14], an ADRC strategy based on the hyperbolic tangent tracking differentiator (TD) is proposed, simplifying TDs structure and parameter rectification process. The above work based on the ADRC controller effectively improves the system's control performance, but the disadvantages of the ADRC controller are a lot of parameters, the poor sensitivity of the parameters, and the coupling relationship between the parameters, which means that the selection and debugging of the optimal parameters are difficult [15]. Usually, the ADRC parameters are determined by empirical methods, but this method is time-consuming and makes it difficult to obtain satisfactory control performance [16]. Therefore, some artificial intelligence tuning methods have emerged. In [17], an ADRC controller parameter optimization strategy based on the ant colony algorithm is proposed, which has the advantage of adjusting many parameters simultaneously, but the computational efficiency is low. In [18], a method for optimizing the parameters of the ADRC based on an improved particle swarm optimization (PSO) genetic algorithm (GA) is proposed, which improves the situation that the PSO algorithm easily falls into local optimum and improves the stability of the algorithm. In [19], backpropagation neural networks (BPNNs) are introduced to the ADRC controller to solve the controller parameter tuning problem. However, BPNN is a global approximation network with slow learning speed, which cannot meet the real-time requirement, and high sensitivity to initial weights. A method for ADRC parameter tuning based on radial basis function neural network (RBFNN) is proposed in [20]. RBFNN has strong nonlinear approximation ability and fast training speed, but each input sample needs distance calculation with each radial basis function, which increases the computational complexity of the network, and the performance of the network is too dependent on the selection of initial parameters.

In this article, a novel ADRC strategy based on a resilient backpropagation neural network (RPROPNN) optimized by an improved differential evolution (IDE) algorithm is proposed to achieve a strong antidisturbance capability for system parameter variations and uncertain disturbances, hereafter referred to as improved ADRC (IADRC). First, the elite group bootstrap mechanism and parameter adaptive technique are introduced to improve the DE algorithm to accelerate its convergence speed and enhance its global search capability, and the performance of the IDE algorithm is verified by the performance test of the test function. Second, the IDE algorithm is used to optimize the initial weights of RPROPNN to solve the problem that RPROPNN is easy to converge locally and ensure that it approaches the global optimum [21]. Finally, it can achieve automatic optimization of the key parameters of the ADRC, thus effectively reducing the impact of ADRC parameters on motor control performance. The control strategy has fast convergence speed and high reliability. The simulation and experimental results show that the proposed method in this article has better control accuracy, response characteristics, and antidisturbance capability.

II. Operation Principle and Mathematical of BPMSM

A. Operation Principle of BPMSM

The stator of the BPMSM studied in this article consists of six L-shaped core columns, each wound with a set of suspension force windings and a set of torque windings, the pole pairs of which are recorded as P_B and $P_{M,}$ respectively. According to the conditions of radial suspension force, the pole pairs of the torque windings and the pole pairs of the suspension force windings must satisfy $P_M = P_B \pm 1$ and the electrical angular frequencies of the two sets of windings meet equal magnitude and rotate in the same direction. The winding pole pairs are set to $P_B = 2$ and $P_M = 1$. When current is applied to the suspension force windings N_{Bd} and the torque windings N_{Md}, a four-pole suspension flux and a two-pole torque flux are generated, respectively. As shown in Fig. 1(b), when the magnetic fields generated by the two sets of windings are superimposed, the magnetic field is strengthened at air gap 1 and weakened at air gap 3, resulting in a suspension force along the positive direction in the x-axis. Similarly, the suspension force in the y-axis can be generated by controlling the current in the suspension force windings N_{Bq}.

Fig. 2 illustrates the principle of passive suspension forces generation. The magnetic field generated by the rotor's permanent magnets, combined with the principle of minimum reluctance, allows the rotor to be passively suspended in three degrees of freedom: left-right tilt, front-back tilt, and axial translation.

B. Mathematical Model of BPMSM

1) Mathematical Model of the Suspension Part: According to the Maxwell tensor method, the radial suspension force mathematical model of the BPMSM is established, and the control current is expressed by the d- and q-axis components, when the

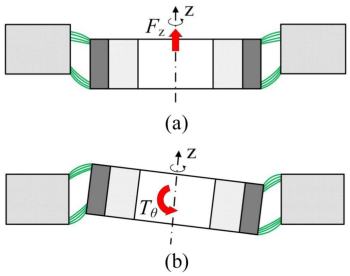

Fig. 2. Generation principle of passive suspension force.

radial x-axis and radial y-axis offsets are x and y, respectively, and the radial x-axis suspension force F_x and the radial y-axis suspension force F_y can be expressed as follows:

$$\begin{cases} F_x = (k_M + k_L)\left(i_{Bd}\psi_{Md} + i_{Bq}\psi_{Mq}\right) + k_S \cdot x \\ F_y = (k_M + k_L)\left(i_{Bq}\psi_{Md} - i_{Bd}\psi_{Mq}\right) + k_S \cdot y \end{cases} \tag{1}$$

$$\begin{cases} k_M = \pi P_B P_M L_B/(8lr\mu_0 N_1 N_2) \\ k_L = mP_M N_B/(4rP_B N_1) \end{cases} \tag{2}$$

where i_{Bd} and i_{Bq} are the current components of the suspension force windings in the d- and q-axis, respectively, ψ_{Md} and ψ_{Mq} are the torque windings air-gap flux-linkage components in the d- and q-axis synthesized by the torque windings and the rotor permanent magnet, k_M is the Maxwell force constant, k_L is the Lorentz force constant, k_S is the displacement constants, P_M and P_B are the pole pairs of the torque windings and the suspension force windings, respectively, L_B is the inductance of the suspension force windings, l and r are the rotor length and radius, respectively, μ_0 is the vacuum permeability, and N_1 and N_2 are the turns of the torque windings and the suspension force windings, respectively.

2) Mathematical Model of the Rotating Part: According to the magnetic field orientation control strategy, the voltage equation for the torque windings in the rotating coordinate system is expressed as follows:

$$\begin{cases} u_{1d} = 0 \\ u_{1q} = P_M \psi_f \omega_r = c_u \omega_r \end{cases} \tag{3}$$

where u_{1d} and u_{1q} are the voltage components of the torque windings in the d- and q-axis, ψ_f is the flux linkage generated by the permanent magnet slice rotor, ω_r is the angular speed of rotation of the permanent magnet slice rotor, and c_u is the voltage–speed coefficient.

The equation for the electromagnetic torque of the BPMSM is expressed as follows:

$$T_e = (m_1 P_M/2)\psi_f i_{Mq} = c_m i_{Mq} \tag{4}$$

where m_1 is the phase number of the torque windings and the suspension windings, i_{Mq} is the current component of the torque windings in the q-axis, and c_m is the torque–current coefficient.

According to the rotor dynamics theory, the rotor's motion equation can be expressed as follows:

$$\begin{cases} F_x + F_{sx} + F_{dx} = m\ddot{x} \\ F_y + F_{sy} + F_{dy} = m\ddot{y} \\ T_e - T_L = J\dot{\omega}_r/PM \end{cases} \tag{5}$$

where m is the mass of the rotor, F_{sx} and F_{sy} are the eccentric magnetic pull in the x- and y-axis, F_{dx} and F_{dy} are the external disturbance force components in the x- and y-axis, J is the rotor moment of inertia, ω_r is the rotor angular velocity, and T_L is the load torque.

III. ADRC OF BPMSM

The ADRC controller is an advanced control strategy based on the disturbance observation and dynamic compensation techniques, capable of estimating and compensating for various disturbances inside and outside the system in real time, optimizing the steady-state and dynamic performance of the system, and having strong robustness and adaptability to different control scenarios and complex engineering systems. It consists of three main components: a TD, an extended state observer (ESO), and a nonlinear state error feedback (NLSEF). The BPMSM has an additional set of suspension force windings compared with a conventional motor, so the ADRC for the rotating and suspension force parts of the BPMSM must be designed separately.

The ADRC design for the rotating part is given as follows:

$$\text{TD}\ \dot{v}_{11} = \mathit{fhan}(v_{11} - \omega_r^*, r, h) \tag{6}$$

$$\text{ESO}\ \begin{cases} e_1 = \omega_{r1} - \omega_r \\ \dot{\omega}_{r1} = \omega_{r2} - \beta_{11}\mathit{fal}(e_1, \alpha 1, \delta_1) + \frac{P_M^2 \psi_M}{J} i_{Mq} \\ \dot{\omega}_{r2} = -\beta_{12}\mathit{fal}(e_1, \alpha 2, \delta_1) \end{cases} \tag{7}$$

$$\text{NLSEF}\ \begin{cases} e_2 = v_{11} - \omega_{r1} \\ i_{Mq0} = k\mathit{fal}(e_2, \alpha 3, \delta 2) \\ i_{Mq} = i_{Mq0} - J\,\omega_{r2}/P_M^2 \psi_M \end{cases} \tag{8}$$

where v_{11} is the tracking value for a given speed, ω_r^* is the given speed of the system, r is the tracking speed factor, and h is the filter factor; the error between the tracking value and the given reference value is processed by the fhan function to obtain an approximate differential signal of ω_r^*, and the fhan function is given in (13); ω_{r1} is the observed value of the actual velocity ω_r, ω_{r2} is an estimate of the total system disturbance, β_{11}, β_{12}, α_1, α_2, and δ_1 are the control parameters of the ESO; k, α_3, and δ_2 are the control parameters of the NLSEF, and i_{Mq0} is the control variable to be compensated. The expression for the nonlinear function fal is defined as follows:

$$\mathit{fal}(e, \alpha, \delta) = \begin{cases} |e|^\alpha \text{sign}(e), |e| > \delta \\ e/\delta^{(1-\alpha)}, |e| \le \delta. \end{cases} \tag{9}$$

The ADRC design for the suspension part is given as follows:

$$\text{TD}\ \begin{cases} \dot{v}_{21} = v_{22} \\ \dot{v}_{22} = \mathit{fhan}(v_{21} - s^*, v_{22}, r, h) \end{cases} \tag{10}$$

$$\text{ESO}\ \begin{cases} \varepsilon_1 = s_1 - s \\ \dot{s}_1 = s_2 - \beta_{21}\varepsilon_1 \\ \dot{s}_2 = s_3 - \beta_{22}\mathit{fal}(\varepsilon_1, \alpha'_1, \delta'_1) + F/m \\ \dot{s}_3 = -\beta_{23}\mathit{fal}(\varepsilon_1, \alpha'_2, \delta'_1) \end{cases} \tag{11}$$

$$\text{NLSEF}\ \begin{cases} e_{21} = v_{21} - s_1 \\ e_{22} = v_{22} - s_2 \\ F_0 = k_1\mathit{fal}(e_{21}, \alpha'_3, \delta'_2) + k_2\mathit{fal}(e_{22}, \alpha'_4, \delta'_2) \\ F = F_0 - ms_3 \end{cases} \tag{12}$$

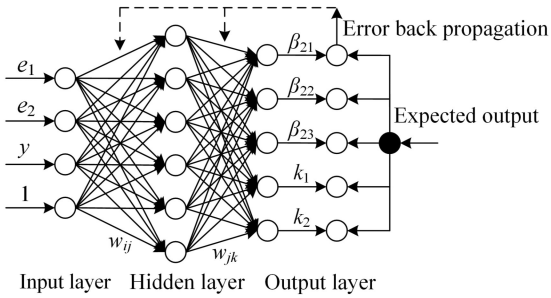

Fig. 3. Structure of the RPROPNN.

$$\begin{cases} d = rh, d_0 = hd, y = v_1 + hv_2 \\ a_0 = \left(d^2 + 8r|y|\right)^{1/2} \\ a = \begin{cases} v_2 + (a_0 - d)/2, |y| > d_0 \\ v_2 + y/h, |y| \le d_0 \end{cases} \\ fhan = -\begin{cases} r\text{sign}(a), |a| > d \\ ra/d, |a| \le d \end{cases} \end{cases} \tag{13}$$

where v_{21} is the tracking signal for a given displacement signal $s*$ and v_{22} is the differential signal of v_{21}; s_1 and s_2 are the observed and estimated values of the actual displacement s, respectively, and s_3 is the estimated value of the total disturbance. β_{21}, β_{22}, β_{23}, α'_1, α'_2, and δ'_1 are the ESO control parameters; and k_1, k_2, α'_3, α'_4, and δ'_2 are the NLSEF control parameters. The NLSEF uses a nonlinear function to nonlinearly combine the errors e_{21} and e_{22} to calculate the error feedback control quantity F_0 and then uses the compensation of the disturbance estimate s_3 to determine the final control quantity F. The nonlinear fhan function is shown in (13).

IV. ADRC Strategy for BPMSM Based on IDE-RPROPNN

A. Establishment of RPROPNN

The RPROPNN has the excellent nonlinear approximation capability of the BPNN and is well suited for training neural networks with mean square error as the performance metric. The RPROPNN consists of an input layer, a hidden layer, and an output layer, the structure shown in Fig. 3. The input nodes are the signal error e_1, the differential signal error e_2, the system output y, and the constant 1. The number of nodes in the hidden layer is determined after several attempts to be six, and the output nodes correspond to the five adjustable parameters $\beta_{21}, \beta_{22}, \beta_{23}$, k_1, and k_2 of ESO and NLSEF.

Each layer of the RPROPNN is explained as follows.

The input layer: The function of this layer is to transfer the input vector to the hidden layer. The output for each node can be written as follows:

$$O_i^1 = x_i, i = 1, 2, 3, 4 \tag{14}$$

where x_i is the input variable, and the superscripts 1, 2, and 3 represent the input layer, the hidden layer, and the output layer, respectively. *The hidden layer*: The input and output for each node can be written as follows:

$$\begin{cases} h_j^2 = \sum_{j=1}^{6} w_{ij}^2 O_i^1 \\ O_j^2 = g(h_j^2) \end{cases} \quad j = 1, 2, \ldots, 6 \tag{15}$$

$$g(x) = (e^x - e^{-x})/(e^x + e^{-x}) \tag{16}$$

where w_{ij}^2 is the hidden layer weight. The transfer function of the hidden layer neurons is chosen as a hyperbolic tangent function with positive and negative symmetry, and the expression is shown in (16).

Output layer: The input and output of a node can be written as follows:

$$\begin{cases} y_k^3 = \sum_{j=1}^{6} w_{jk}^3 O_j^2 \\ O_k^3 = f(y_k^3) \end{cases} \quad k = 1, 2, \ldots, 5 \tag{17}$$

$$f(x) = e^x/(e^x + e^{-x}) \tag{18}$$

where w_{jk}^3 is the output layer weight. Since the optimization parameters of the network output should be nonnegative, the following sigmoid function is used for the transformation function of the neurons in the output layer, and the expression is shown in (18).

The training objective of RPROPNN is to minimize the mean square error of the difference between the network output and the desired output, and the error indicator function is given by

$$E(w) = \frac{1}{2}\sum \left(y(k) - \hat{y}(k)\right)^2 = \frac{1}{2}\sum e_{\text{error}}^2 \tag{19}$$

where $y(k)$ is the actual network output, $\hat{y}(k)$ is the expected network output, and e_{error} is the error.

The RPROP algorithm is a first-order algorithm that uses only the gradient sign of the iterations to correct the weights with the following weight update formula:

$$\begin{cases} w_{ij}^{(k+1)} = w_{ij}^{(k)} - \Delta_{ij}^{(k)} \cdot \text{sign}(\nabla E^{(k)}) \\ \nabla E^{(k)} = \partial E^{(k)} \Big/ \partial w_{ij}^{(k)} \end{cases} \tag{20}$$

where $\nabla E^{(k)}$ is the gradient of the kth iteration and $\Delta_{ij}^{(k)}$ is the adaptive correction value, whose learning rule is determined by the following equation:

$$\Delta_{ij}^{(k)} = \begin{cases} \eta^+ \cdot \Delta_{ij}^{(k-1)}, & \text{if} \nabla E^{(k-1)} \cdot \nabla E^{(k)} > 0 \\ \eta^- \cdot \Delta_{ij}^{(k-1)}, & \text{if} \nabla E^{(k-1)} \cdot \nabla E^{(k)} < 0 \\ \Delta_{ij}^{(k-1)}, & \text{if} \nabla E^{(k-1)} \cdot \nabla E^{(k)} = 0 \end{cases} \tag{21}$$

where η^+ and η^- are the incremental and decremental factors, respectively, and $0 < \eta^- < 1 < \eta^+$.

B. IDE Algorithm

Since the selection of initial weights largely affects the convergence and training time of RPROPNN, if the initial values of weights are not chosen properly, the weighted inputs are likely to fall into the saturation region of the activation function, thus stopping the adjustment of weights. The IDE algorithm is used to optimize it so that the iterative training of the parameters is less likely to fall into local extremes, thus accelerating the convergence speed and finally achieving the global optimization of the ADRC parameters. In this article, the performance of

DE algorithm is improved mainly by using an elite population bootstrap mechanism and adaptive adjustment technique of control parameters. The following four steps are described in turn: initialization, mutation, crossover, and selection.

The DE algorithm uses real number encoding to represent each weight as a real value. A complete set of RPROPNN weights and the *i*th individual in the current population can be represented as follows:

$$\begin{cases} W = \{W_1, W_2, \ldots, W_{NP}\} \\ W_i = (w_1, w_2, \ldots, w_D), i = 1, 2, \ldots, NP \end{cases} \tag{22}$$

where *NP* is the size of the population, and *D* is the dimension of the object parameters, which is set as 20.

Then, the population fitness is calculated, and the fitness function is defined as finding the minimum value of the objective function

$$f^* = f^*(w'_i) = \min\left(E(w)\right) \tag{23}$$

where f^* is the fitness function, and w'_i is the current optimal solution after *G* generation.

Determine whether the individuals in the initial population meet the termination condition based on the value of the fitness function and set the termination condition to the maximum number of evolutionary generations; for individuals with poor fitness, mutation, crossover, selection, etc., are performed.

Improvement strategies:

1) Elite Group Bootstrap Variation Vector: In the basic DE algorithm, the variation vector is calculated from three randomly selected vectors in the parent population, which fails to fully use the excellent experience in the evolutionary process to guide the variation direction of new individuals. In this article, we introduce the concept of the elite population, rank the current population from good to bad fitness values, and define the top EP individuals in the population with better fitness values as the elite population. Individuals in the elite group are used as the base vector to guide the variation vector to optimize in the direction of better fitness value. The expression is

$$\mathrm{EP} = \mathrm{ceil}\left\{\frac{NP}{4}\left(\cos\left(\frac{G\pi}{G_{\max}}\right)+1\right)\right\} \tag{24}$$

where EP is the number of elite individuals, ceil$\{i\}$ is the smallest integer greater than *i*, *G* is the current number of iterations, and $G_{\max}$ is the maximum number of iterations set by the algorithm. The number of elite individuals is larger at the early stage of evolution but decreases nonlinearly with the number of iterations so that it can take into account the diversity of the population at the early stage of iteration and converge to the global optimal solution at the later stage, which accelerates the convergence speed of the algorithm.

2) Adaptive Adjustment Technology for Control Parameters: The DE algorithm is very sensitive to the choice of parameters; the variation factor *F* affects the search range and the crossover factor CR determines the search direction. Therefore, the settings of the variation and crossover factors will have a profound impact on the convergence performance and the search efficiency of the algorithm. In this article, the parameter adaptive adjustment technique is used to adjust the variation and crossover factors to balance the global exploration and local opening capability of the algorithm. The adaptive adjustment method is given as follows:

$$\begin{cases} F = F_{\min} + (F_{\max} - F_{\min}) \cdot \exp\left(0.2\pi \frac{G}{G_{\max} - G}\right) \\ CR = CR_{\min} + (CR_{\max} - CR_{\min}) \cdot \exp\left(0.2\pi \frac{G}{G_{\max} - G}\right) \end{cases} \tag{25}$$

where $F_{\min}$ and $F_{\max}$ are the minimum and maximum values of the variation factor, respectively, and $\mathrm{CR}_{\min}$ and $\mathrm{CR}_{\max}$ are the minimum and maximum values of the crossover factor, respectively.

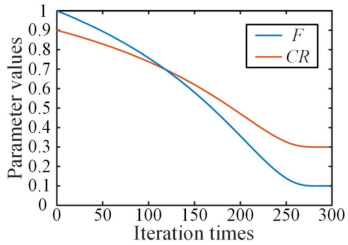

Fig. 4. Relationship between control parameters with generation number.

TABLE I
COMPARISON TABLE OF SPECIFIC PARAMETERS OF FOUR ALGORITHMS

Algorithm	Basic parameter setting
GA	P_c=0.8, P_m=0.1
PSO	c_1=c_2=1.5, w_1=0.8, w_2=0.4
DE	F=0.5, CR=0.7
IDE	$F_{\min}$=0.1, $F_{\max}$=1, $CR_{\min}$=0.3, $CR_{\max}$=0.9

The relationship between control parameters *F* and CR and evolutionary generations is obtained by plotting the function image of (25) using MATLAB, as shown in Fig. 4, where the maximum and minimum values of *F* and CR are shown in Table I. Fig. 4 shows the relationship between parameters *F* and CR and evolutionary generations. It can be seen from the figure that, in the initial stage of evolution, the IDE algorithm has large *F* and CR, which can make individuals change more, thus ensuring the diversity of the population and preventing the algorithm from falling into local optimal solutions. With the increasing number of evolutionary generations, the *F* and CR converge to a stable minimum, which speeds up the algorithm's convergence to the optimal solution set and improves the search efficiency of the algorithm.

In the improved variation strategy, two different individuals are selected from the parent population to generate a difference vector that is weighted and linearly combined with elite individuals. The improved variation strategy using the elite population bootstrap mechanism and the adaptive variation factor is given as follows:

$$V_{i,G+1} = W_{\mathrm{EP},G} + F\left(W_{r1,G} - W_{r2,G}\right). \tag{26}$$

where $W_{\mathrm{EP},G}$ is the individual in the elite population iterated to the difference vector, satisfying $F \in [0,1]$, $Wr_{1,G}$, $Wr_{2,G}$ are

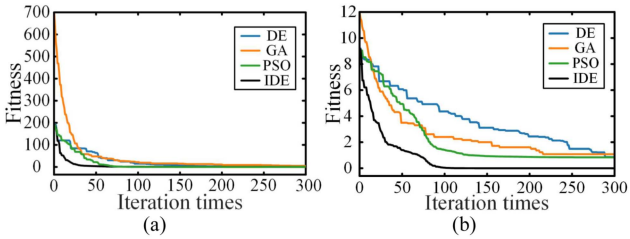

Fig. 5. Convergence curves of four algorithms under two test functions. (a) Sphere function. (b) Ackley function.

often called difference terms, and $r1$ and $r2$ are the random integers between [1,NP], satisfying $r1 \neq r2 \neq i$.

In the improved crossover strategy, the crossover operation increases the population's diversity; its main effect is to cross the offspring individuals produced by the mutation with the parent to produce new individuals. The crossover operation is given as follows:

$$U_{i,G+1}^{j} = \begin{cases} V_{i,G+1}^{j}, & \text{if rand}\,(0,1) < CR \text{ or } j = j_{\text{rand}} \\ X_{i}^{j}, & \text{otherwise.} \end{cases} \tag{27}$$

where rand(0,1) denotes a random number between [0,1], CR is a crossover factor satisfying CR $\in$ [0,1], and j_{rand} is a random number between [1,D], which guarantees that at least one-dimensional element in the mutant individual evolves into the next generation of individuals.

Finally, in the selection operation, the DE algorithm adopts the greedy strategy to select individuals with good fitness values for the next evolution from the parent individuals $W_{i,G}$ and the individuals $U_{i,G+1}$ generated by the mutation crossover. The selection operation is shown as follows:

$$W_{i,G+1} = \begin{cases} U_{i,G+1}, & \text{if } f^{*}\left(U_{i,G+1}\right) > f^{*}\left(W_{i,G}\right) \\ W_{i,G}, & \text{otherwise.} \end{cases} \tag{28}$$

C. Algorithm Performance Testing and Analysis

In order to verify the effectiveness of the proposed algorithm, the IDE algorithm is written based on the C language, and simulation tests are performed in Python. Meanwhile, two different types of benchmark functions, Sphere and Ackley, are selected for performance testing, and the basic PSO algorithm, GA, and DE algorithm are compared with the IDE algorithm. The specific parameters of the algorithm are shown in Table I. Considering the randomness of the algorithm, each optimization algorithm is run 30 times, and the fitness function value obtained in the average case is compared and analyzed. The population size is set to 175, the number of iterations is set to 300, and the search range is set to [−55]. The comparison of the results of the four algorithm runs is shown in Fig. 5. Table II presents the average value of each algorithm under the two benchmark functions. As can be seen from the figure, for the Sphere test function, the IDE algorithm finds the optimal value 0 after the 80th iteration, while the DE, GA, and PSO algorithms take 294, 277, and 98 iterations to stably converge, respectively. For Ackley test functions, the IDE algorithm found the optimal value of 0 after 113 iterations, while the DE, GA, and PSO algorithms need 289, 216, and 171 iterations for stable convergence, respectively. The IDE algorithm proposed in this article has obvious advantages in terms of final solution accuracy and convergence speed. This is because the elite group in the IDE algorithm guides the mutation vector to evolve in a better direction, which can effectively improve the convergence speed of the algorithm. At the same time, the adaptively adjusted variation and crossover factors can effectively coordinate the global exploration and local exploitation capabilities of the algorithm when processing the objective function, and better adapt to the characteristics of particle optimization. In summary, the IDE algorithm has better performance through the performance test of the test function.

TABLE II
COMPARISON OF AVERAGE VALUES OF ALGORITHMS UNDER DIFFERENT BENCHMARK FUNCTIONS

Function	GA	DE	PSO	IDE
Sphere	1.0739	0.9122	0.8606	0
Ackley	1.3609	0.4016	0	0

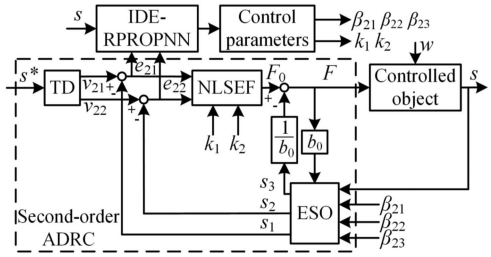

Fig. 6. Structure diagram of ADRC optimized by IDE-RPROPNN.

D. ADRC Parameter Tuning Based on IDE-RPROPNN

Since the BPMSM control system includes a rotating part and a suspension part, the ADRC parameters of these two parts need to be adjusted separately, increasing the parameter correction's difficulty. ESO is the core of the ADRC, whose primary function is to estimate and compensate the total disturbance of the system in real time. Meanwhile, the ADRC uses a nonlinear combination of error feedback to input the controlled variable, which greatly impacts the controller performance. Therefore, in this article, using the characteristics of neural network (NN) infinitely approximating nonlinear functions, we use IDE-RPROPNN to optimize the key parameters of ADRC in the suspension section, including β_{21}, β_{22}, and β_{23} in ESO and k_1 and k_2 in NLSEF, to achieve the online adjustment of its key parameters. Then, the initial values are set for the above parameters, and the appropriate range of parameter variation is selected based on the empirical method.

The range of parameters selected in this article is as follows: $\beta_{21} \in$ [100,400], $\beta_{22} \in$ [100,1000], $\beta_{23} \in$ [50,400], $k_1 \in$ [1,100], and $k_2 \in$ [1,50]. The structure diagram of the second-order ADRC optimized by IDE-RPROPNN is shown in Fig. 6. The specific flowchart of the algorithm is shown in Fig. 7.

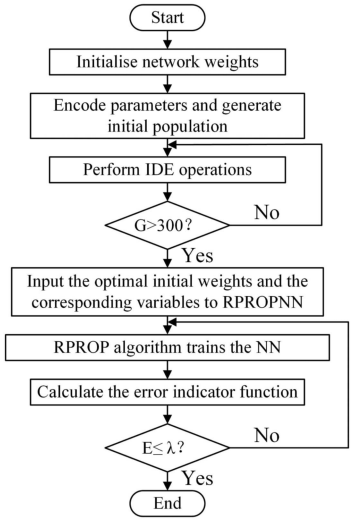

Fig. 7. Flowchart of model establishing.

TABLE III
MAIN PARAMETERS OF BPMSM

Symbol (unit)	Value	Symbol (unit)	value
U_n (V)	220	δ_0 (mm)	2
P_n (kW)	4	Ψ_f (Wb)	0.165
n_n (r/min)	6000	L_{md} / L_{mq} (mH)	13.42
P_M /P_B	1/2	J (kg·m^2)	0.00059
m (kg)	1	R_s (Ω)	1.65

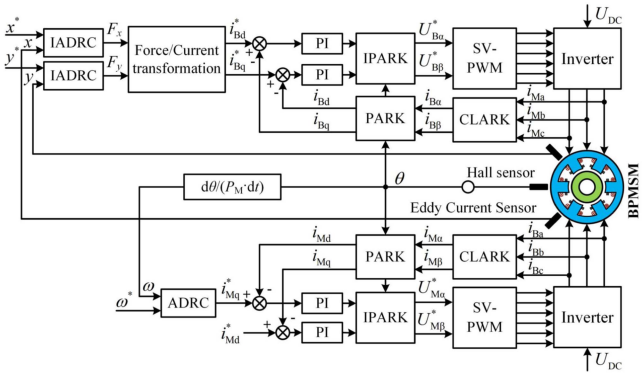

Fig. 8. BPMSM control block diagram of ADRC based on IDE-RPROPNN.

V. SIMULATION TEST

A. Simulation Parameters and System Composition

In order to verify the effectiveness of the ADRC optimized by IDE-RPROPNN in the BPMSM control system, a simulation model of the control system is built in MATLAB/Simulink, and simulation studies are carried out. The BPMSM parameters used in the simulation are shown in Table III. Combined with Fig. 6, the control block diagram of the BPMSM ADRC optimized by IDE-RPROPNN is shown in Fig. 8. The ADRC controller is written in S function, and IDE-RPROPNN is written in M-file. In the simulation, the key parameters of the ADRC of the suspension part are automatically optimized by IDE-RPROPNN,

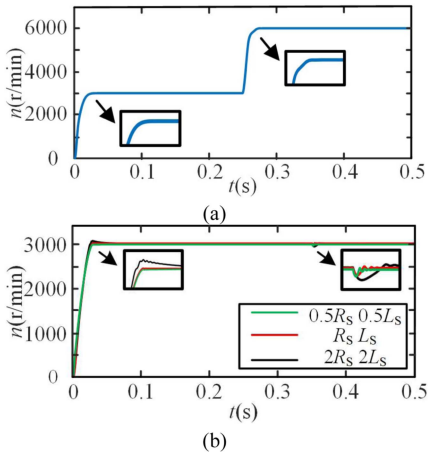

Fig. 9. Speed response curves. (a) Speed step response curves. (b) Speed response curves for system parameters mismatch.

and the remaining parameters and the parameters of the ADRC of the rotating part are adjusted by the trial-and-error method and the empirical method. The ADRC parameters in the simulation results are as follows.

1) The rotating part:

TD: $r = 3000$ and $h = 0.01$;

ESO: $\alpha_1 = 0.5$, $\alpha_2 = 0.25$, $\delta_1 = 0.01$, $\beta_{11} = 160\,000$, and $\beta_{12} = 5000$;

NLSEF: $\alpha_3 = 0.78$, $\delta_2 = 0.01$, and $k = 0.065$.

2) The suspension part:

TD: $r = 400$ and $h = 0.01$;

ESO: $\alpha'_1 = 0.5$, $\alpha'_2 = 0.25$, $\delta'_1 = 0.003$, $\beta_{21} = 210.3$, $\beta_{22} = 652.5$, and $\beta_{23} = 149.7$;

NLSEF: $\alpha'_3 = 0.5$, $\alpha'_4 = 0.25$, $\delta'_2 = 0.01$, $k_1 = 54.2$, and $k_2 = 5.5$.

B. Analysis of Simulation Results

Since this article adopts IDE-RPROPNN to optimize the ADRC parameters for the suspension part and the rotation part is still conventional ADRC, the speed simulation results are basically the same under both methods. Therefore, only the speed response based on ADRC is given when the speed is accelerated from 0 to 6000 r/min. As can be seen from Fig. 9, when the motor speed is accelerated from 0 to 3000 r/min, the ADRC reaches the given speed at 0.03 s. At 0.25 s, the motor speed is accelerated to 6000 r/min and the ADRC enters the steady state at 0.278 s. Under ADRC control, there is almost no overshoot and the steady-state error in reaching steady state is less than 3 r/min. The speed response curves when the system parameters are varied are shown in Fig. 9(b). The speed characteristic curves at 0.5Rs and 0.5Ls and when the system parameters are normal show basically no overshoot, whereas an overshoot of 53 r/min occurs at 2Rs and 2Ls. A load of 7 N·m applied to the system at 0.35 s reduces the speed by 36 r/min at 0.5Rs and 0.5Ls and normal system parameters, whereas the speed is reduced by 62 r/min at 2Rs and 2Ls. The above results show that the ADRC is insensitive to changes in system parameters and has good dynamic response and speed characteristics.

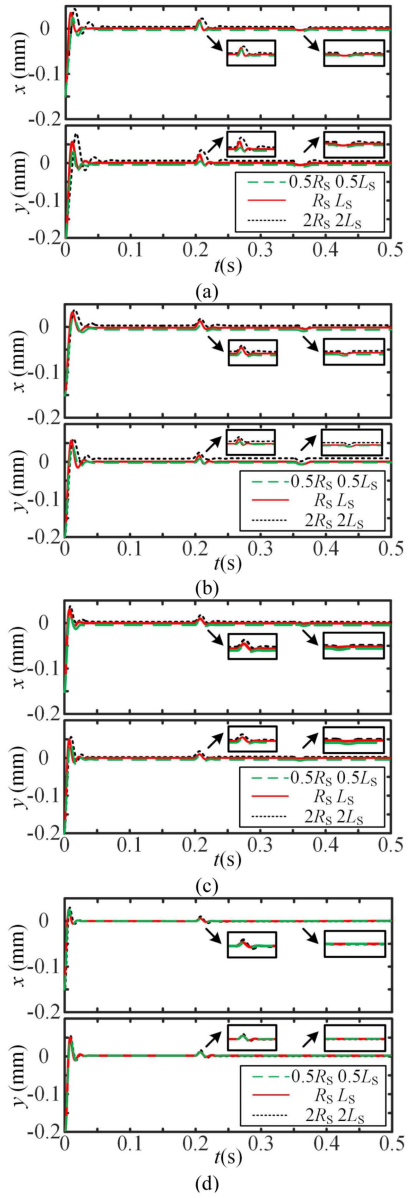

Fig. 10. Rotor radial displacements curves in the x and y directions. (a) SMC. (b) ADRC. (c) BPNN-ADRC. (d) IADRC.

Radial displacement waveforms of the rotor in the x- and y-direction are shown in Fig. 10. The initial position of the slice rotor is (-0.15 mm, -0.2 mm). When the motor begins to accelerate, the slice rotor position also begins to be adjusted. In the x-direction, the maximum radial displacement value of the sliding-mode control (SMC), ADRC, and BPNN-ADRC is 0.042 mm, 0.033 mm, and 0.031 mm, respectively, and the minimum values of stabilization time are 0.061 s, 0.056 s, and 0.045 s, respectively. However, the maximum radial displacement value of the IADRC is 0.029 mm, and the minimum value of the stabilization time is 0.039 s. In the y-direction, the maximum radial displacement value of the SMC, ADRC, and BPNN-ADRC is 0.062 mm, 0.056 mm, and 0.049 mm, respectively, and the minimum values of stabilization time are 0.069 s, 0.063 s, and 0.047 s, respectively. However, the maximum radial displacement value of the IADRC is 0.047 mm, and the minimum value of the stabilization time is 0.042 s. At 0.2 s, a 10 N disturbance force is added in both the x- and y-direction, and a load of 7 N·m is applied to the system at 0.35 s. It can be seen from the figure that the proposed method has stronger antidisturbance ability and stability. In addition, when the system parameters change, the proposed method is little affected by the parameter disturbance, which further proves the reliability and robustness of the IADRC.

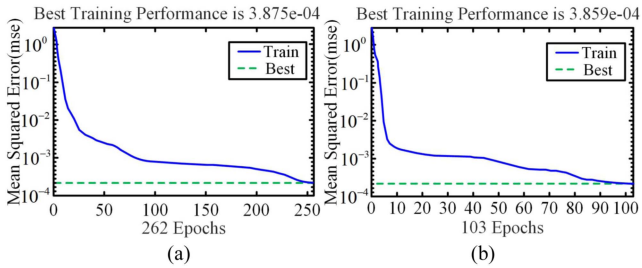

Fig. 11. Training error curve. (a)Training error of RPROPNN. (b) Training error of IDE-RPROPNN.

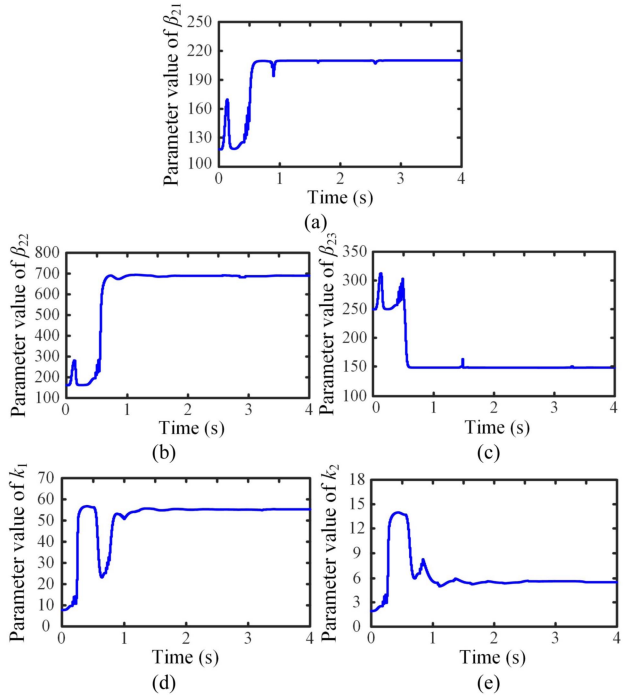

Fig. 12. Curve of parameter variation optimized by IDE-RPROPNN. (a) Optimization curve of β_{21}. (b) Optimization curve of β_{22}. (c) Optimization curve of β_{23}. (d) Optimization curve of k_1. (e) Optimization curve of k_2.

C. Convergence Analysis of IDE-RPROPNN

The convergence of IDE-RPROPNN is verified in this part. The mean square error between the expected and measured values is used as the evaluation criterion to visually compare the models built by RPROPNN and IDE-RPROPNN, and the training error is shown in Fig.11. The convergence curve of parameters β_{21}, β_{22}, β_{23}, k_1, and k_2 under the IADRC is shown in Fig. 12. After 262 iterations, the training error value

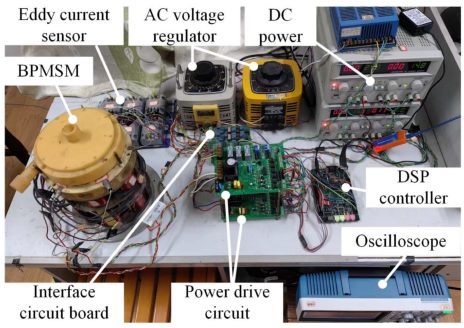

Fig. 13. Schematic diagram of the experimental platform.

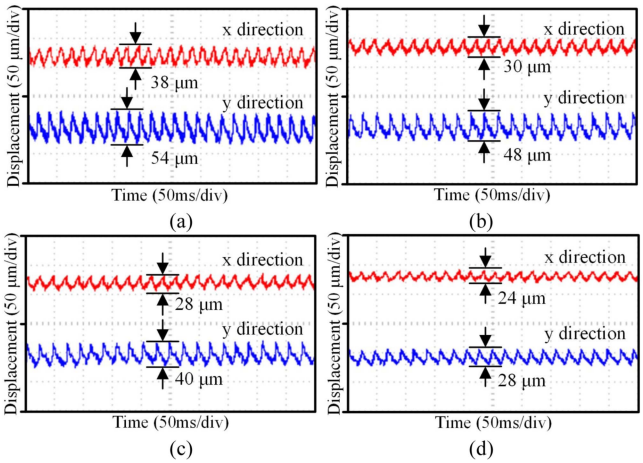

Fig. 14. Experimental waveforms of radial displacement in the x- and y-direction. (a) SMC. (b) ADRC. (c) BPNN-ADRC. (d) IADRC.

of RPROPNN is less than the set value of 0.0004 and the performance of the algorithm reaches convergence accuracy. And the training error value of IDE-RPROPNN is less than 0.0004 after only 103 iterations. Compared with RPROPNN, the number of iterations of IDE-RPROPNN is significantly reduced under the condition of achieving the same training accuracy. This can indicate that the IDE algorithm can indeed play a role in the optimization of RPROPNN with significantly fewer iterations and correspondingly faster convergence, thus verifying the convergence of IDE-RPROPNN.

VI. Experimental Research

In order to further verify the effectiveness of the proposed strategy, two control methods are applied to a 4-kW BPMSM prototype for experimental verification. The prototype experimental platform in this article is shown in Fig. 13. In the experiment, the parameters of the prototype are the same as those of the simulation, and the sampling frequency is set to 20 kHz. TMS320F28335 is used as the controller to build a digital control system. The experimental results are shown in Figs. 14–17.

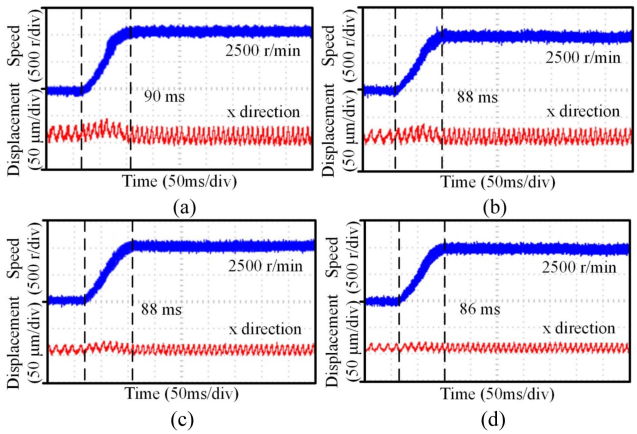

Fig. 15. Experimental waveforms of rotational speed and radial displacement in the x-direction. (a) SMC. (b) ADRC. (c) BPNN-ADRC. (d) IADRC.

A. Static Suspension Experiment

The experimental waveforms of radial displacement in the x- and y-direction when the rotor is running at 1000 r/min and stably suspended are shown in Fig. 14. The displacement amplitudes of the SMC are about 38 μm and 54 μm in the x- and y-direction, respectively. The displacement amplitudes of the ADRC are about 30 μm and 48 μm in the x- and y-direction, respectively. The displacement amplitudes of the BPNN-ADRC are about 28 μm and 40 μm in the x- and y-direction, respectively. The displacement amplitudes of the IADRC are about 24 μm and 28 μm in the x- and y-direction, respectively. It can be seen that the static suspension performance of the system is effectively improved under the IADRC.

B. Speed Variation Experiment

The experimental waveforms of rotational speed and the x-direction radial displacement when the speed accelerates from 1500 to 2500 r/min are shown in Fig. 15. There is almost no overshoot under the four control methods, and the difference in adjustment time is very small. During the speed rise, the displacement vibration amplitude in the x-direction of the control strategy used in this article is significantly smaller than that of the other methods, and it is almost unaffected when the speed changes, while the rotor vibration amplitude is also significantly reduced after the speed stabilization (the displacement change in the y-direction is similar to the x-direction, so it is not shown again). The experimental results show that the IADRC method proposed in this article has better dynamic performance and can achieve stable suspension of the rotor at different rotational speeds, and the suspension force is significantly reduced by the influence of torque and the decoupling performance is improved.

C. Antidisturbance Experiment

When the speed is stable at 1500 r/min, a radial disturbance force of the same direction and size 10 N is applied to the rotor using a tensiometer, and the radial displacement curves under the four methods are shown in Fig. 16. From the figure, it can be seen

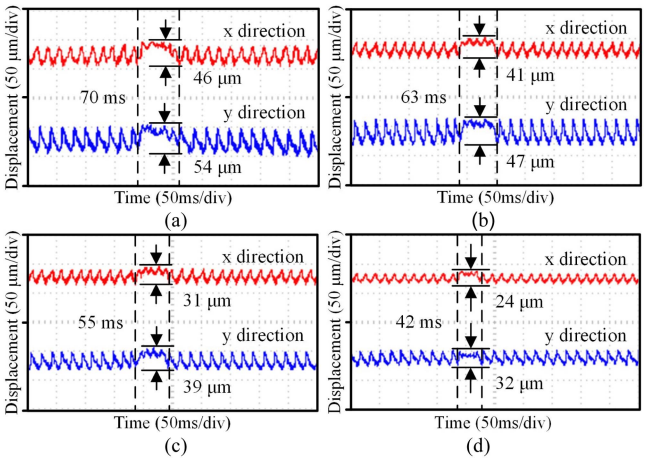

Fig. 16. Experimental waveforms of radial displacement when the rotor is disturbed. (a) SMC. (b) ADRC. (c) BPNN-ADRC. (d) IADRC.

that the disturbance amplitudes of SMC are 46 μm and 54 μm in the x- and y-direction, respectively, and the adjustment time is 70 ms. The disturbance amplitudes of ADRC are 41 μm and 47 μm in the x- and y-direction, respectively, and the adjustment time is 63 ms. The disturbance amplitudes of BPNN-ADRC are 31 μm and 39 μm in the x- and y-direction, respectively, and the adjustment time is 55 ms. The disturbance amplitudes of IADRC are 24 μm and 32 μm in the x- and y-direction, respectively, and the adjustment time is 42 ms. From the analysis of the above data, it can be seen that the proposed method has good antidisturbance ability and higher suspension force control accuracy, which verifies that the controller designed in this article has stronger robustness.

D. Load Experiment

When the speed is stable at 1500 r/min, a water load of 7 N·m is applied to the system through the pump chamber. The speed response curve under ADRC control shows no steady-state error and the transition is relatively smooth, and the experimental waveforms are shown in Fig. 17. The disturbance amplitudes of SMC are 42 μm and 50 μm in the x- and y-direction, respectively, and the adjustment time is 72 ms. The disturbance amplitudes of ADRC are 38 μm and 44 μm in the x- and y-direction, respectively, and the adjustment time is 65 ms. The disturbance amplitudes of BPNN-ADRC are 30 μm and 35 μm in the x- and y-direction, respectively, and the adjustment time is 57 ms. The disturbance amplitudes of IADRC are 23 μm and 30 μm in the x- and y-direction, respectively, and the adjustment time is 43 ms.

The above experiment is repeated when the motor is running stably at 2500 r/min. The experimental waveforms are shown in Fig. 17. The disturbance amplitudes of SMC are 50 μm and 54 μm in the x- and y-direction, respectively, and the adjustment time is 80 ms. The disturbance amplitudes of ADRC are 42 μm and 49 μm in the x- and y-direction, respectively, and the adjustment time is 72 ms. The disturbance amplitudes of BPNN-ADRC are 40 μm and 44 μm in the x- and y-direction, respectively, and the adjustment time is 68 ms. The disturbance

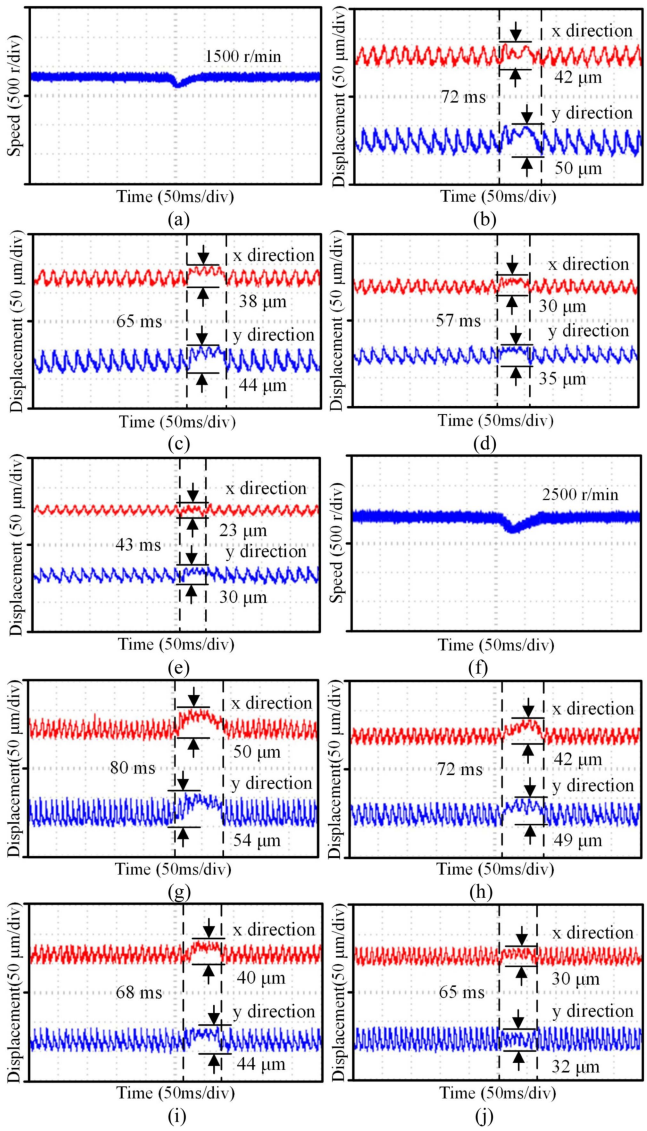

Fig. 17. Experimental waveforms of radial displacement for sudden load. When the rotor speed is 1500 r/min. (a) Rotational speed. (b) SMC. (c) ADRC. (d) BPNN-ADRC. (e) IADRC. When the rotor speed is 2500 r/min. (f) Rotational speed. (g) SMC. (h) ADRC. (i) BPNN-ADRC. (j) IADRC.

amplitudes of IADRC are 30 μm and 32 μm in the x- and y-direction, respectively, and the adjustment time is 65 ms. The above results show that the effect of load on rotor suspension control is reduced after using the proposed method, which verifies that the proposed control method has better antiload disturbance ability.

VII. Conclusion

In order to improve the antidisturbance performance and suspension performance of the BPMSM, an ADRC control strategy based on IDE-RPROPNN is proposed. Through theoretical analysis, simulation, and experimental study, the following conclusions are drawn.

1) The proposed IDE-RPROPNN improves the situation that the DE algorithm is easy to fall into the local optimum,

enhances the global search ability, and improves the optimization accuracy and stability of the algorithm while speeding up the convergence speed.

2) The strategy uses self-learning ability to optimize ADRC parameters, which reduces the difficulty of parameter tuning.
3) The comparative experimental results show that under the IADRC control, the rotor suspension steady-state error, and the radial displacement disturbance caused by the velocity step response, the radial interference force and load are significantly improved, and the IADRC has better dynamic, static, and antidisturbance performance.

REFERENCES

[1] B. S. Weinreb, M. Noh, D. C. Fyler, and D. L. Trumper, "Design and implementation of a novel interior permanent magnet bearingless slice motor," *IEEE Trans. Ind. Appl.*, vol. 57, no. 6, pp. 6774–6782, Nov./Dec. 2021.

[2] T. Pei, D. Li, J. Liu, and W. Kong, "A novel flux-reversal bearingless slice motor controlled by rotor angle independent suspension current," *IEEE Trans. Ind. Electron.*, vol. 70, no. 6, pp. 5615–5625, Jun. 2023.

[3] S. Szoke and M. Noh, "Homopolar bearingless slice motor with quadruple three-phase windings," in *Proc. IEEE Energy Convers. Congr. Expo.*, 2022, pp. 1–6.

[4] W. Bu, X. Zhang, and F. He, "Sliding mode variable structure control strategy of bearingless induction motor based on inverse system decoupling," *IEEJ Trans. Elect. Electron. Eng.*, vol. 13, no. 7, pp. 1052–1059, Jul. 2018.

[5] P. N. Rao, R. Devarapalli, F. P. García Márquez, and H. Malik, "Global sliding-mode suspension control of bearingless switched reluctance motor under eccentric faults to increase reliability of motor," *Energies*, vol. 13, no. 20, Oct. 2020, Art. no. 5485.

[6] P. N. Rao, N. Manoj Kumar, S. Padmanaban, M. S. P. Subathra, and A. A. Chand, "A novel sensorless approach for speed and displacement control of bearingless switched reluctance motor," *Appl. Sci.*, vol. 10, no. 12, May 2020, Art. no. 4070.

[7] W. Bu, F. Zhang, Y. Chen, and X. Zhang, "Fuzzy MRAS rotor resistance identification of BL-IM based on reactive power," *Int. J. Appl. Electromagn. Mech.*, vol. 65, no. 4, pp. 661–680, May 2021.

[8] A. Omari, B. I. Khalil, A. Hazzab, B. Bouchiba, and F. E. Benmohamed, "Real-time implementation of MRAS rotor time constant estimation for induction motor vector control based on a new adaptation signal," *Compel-Int. J. Comput. Math. Elect. Electron. Eng.*, vol. 38, no. 1, pp. 287–303, Jan. 2019.

[9] X. Jiang et al., "An improved implicit model predictive current control with continuous control set for PMSM drives," *IEEE Trans. Transp. Electrific.*, vol. 8, no. 2, pp. 2444–2455, Jun. 2022.

[10] Y. Zhang, J. Jin, and L. Huang, "Model-free predictive current control of PMSM drives based on extended state observer using ultralocal model," *IEEE Trans. Ind. Electron.*, vol. 68, no. 2, pp. 993–1003, Feb. 2021.

[11] S. A. Hashjin, S. Pang, E.-H. Miliani, K. Ait-Abderrahim, and B. Nahid-Mobarakeh, "Data-driven model-free adaptive current control of a wound rotor synchronous machine drive system," *IEEE Trans. Transp. Electrific.*, vol. 6, no. 3, pp. 1146–1156, Sep. 2020.

[12] B. Zhong, L. Ma, H. Dong, and Z. Ren, "Research on the hoisting motor drive system's active disturbance rejection control and energy consumption for a crane," *IEEE Access*, vol. 9, pp. 94338–94351, 2021.

[13] W. Zhang, H. Zhu, Y. Xu, and M. Wu, "Direct control of bearingless permanent magnet slice motor based on active disturbance rejection control," *IEEE Trans. Appl. Supercond.*, vol. 30, no. 4, Jun. 2020, Art. no. 5202205.

[14] Z. Yang, J. Ji, X. Sun, H. Zhu, and Q. Zhao, "Active disturbance rejection control for bearingless induction motor based on hyperbolic tangent tracking differentiator," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 8, no. 3, pp. 2623–2633, Sep. 2020.

[15] M. Tian, B. Wang, Y. Yu, Q. Dong, and D. Xu, "Discrete-time repetitive control-based ADRC for current loop disturbances suppression of PMSM drives," *IEEE Trans. Ind. Inform.*, vol. 18, no. 5, pp. 3138–3149, May 2022.

[16] C. Du, Z. Yin, Y. Zhang, J. Liu, X. Sun, and Y. Zhong, "Research on active disturbance rejection control with parameter autotune mechanism for induction motors based on adaptive particle swarm optimization algorithm with dynamic inertia weight," *IEEE Trans. Power Electron.*, vol. 34, no. 3, pp. 2841–2855, Mar. 2019.

[17] Z. Yin, C. Du, J. Liu, X. Sun, and Y. Zhong, "Research on autodisturbance-rejection control of induction motors based on an ant colony optimization algorithm," *IEEE Trans. Ind. Electron.*, vol. 65, no. 4, pp. 3077–3094, Apr. 2018.

[18] Z. Yang, C. Lu, X. Sun, J. Ji, and Q. Ding, "Study on active disturbance rejection control of a bearingless induction motor based on an improved particle swarm optimization–genetic algorithm," *IEEE Trans. Transp. Electrific.*, vol. 7, no. 2, pp. 694–705, Jun. 2021.

[19] S. Wang, H. Zhu, M. Wu, and W. Zhang, "Active disturbance rejection decoupling control for three-degree-of-freedom six-pole active magnetic bearing based on BP neural network," *IEEE Trans. Appl. Supercond.*, vol. 30, no. 4, Jun. 2020, Art. no. 3603505.

[20] K. Zhao, L. Yang, H. Hu, and Z. Shuang, "Ship PMSM nonlinear ADRC parameter self-tuning based on neural network," in *Proc. 25th Int. Conf. Elect. Mach. Syst.*, 2022, pp. 1–5.

[21] X. Wu, Q. Yuan, and L. Wang, "Multiobjective differential evolution algorithm for solving robotic cell scheduling problem with batch-processing machines," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 2, pp. 757–775, Apr. 2021.

Chuang Sun received the B.S. degree in electrical engineering and automation in 2020 from Jiangsu Normal University, Xuzhou, China, where he is currently working toward the M.S. degree.

His research interests include design and control technology for bearingless motor.

Huangqiu Zhu (Member, IEEE) received the B.S. degree in automation from the School of Electrical and Information Engineering, Jiangsu University, Zhenjiang, China, in 1987, the M.S. degree in management from Jiangsu University, Zhenjiang, China, in 1993, and the Ph.D. degree from the Nanjing University and Aeronautics and Astronautics, Nanjing, China, in 2000.

He was a Visiting Scholar with the Swiss Federal Institute of Technology, Zurich, Switzerland, from 2002 to 2003. He is currently a Professor with the Department of Electrical Engineering, Jiangsu University. His research interests include magnetic bearings, magnetic suspension (bearingless) motors, and motors and movement control.