Power System Load Frequency Active Disturbance Rejection Control via Reinforcement Learning-Based Memetic Particle Swarm Optimization

YUEMIN ZHENG¹, ZHAOYANG HUANG¹, JIN TAO¹,², (Member, IEEE), HAO SUN¹, QINGLIN SUN¹, MATTHIAS DEHMER³, MINGWEI SUN¹, AND ZENGQIANG CHEN¹,⁴, (Member, IEEE)

¹College of Artificial Intelligence, Nankai University, Tianjin 300350, China
²Department of Electrical Engineering and Automation, Aalto University, 02500 Espoo, Finland
³Department of Computer Science, Swiss Distance University of Applied Sciences, 3900 Brig, Switzerland
⁴Key Laboratory of Intelligent Robotics of Tianjin, Nankai University, Tianjin 300350, China

Corresponding author: Jin Tao (jin.tao@aalto.fi)

This work was supported in part by the National Natural Science Foundation of China under Grant 61973172, Grant 61973175, Grant 62003175, and Grant 62003177; in part by the National Key Research and Development Project under Grant 2019YFC1510900; in part by the Key Technologies Research and Development Program of Tianjin under Grant 19JCZZD1C32800; in part by the China Postdoctoral Science Foundation under Grant 2020M670633; and in part by the Academy of Finland under Grant 315660.

ABSTRACT Load frequency control (LFC) is necessary to guarantee the safe operation of power systems. Aiming at the frequency and power stability problems caused by load disturbances in interconnected power systems, active disturbance rejection control (ADRC) was designed. There are eight parameters that need to be adjusted for an ADRC, which are challenging to adjust manually, thus limiting the development of this approach in industrial applications. Regardless of the theory or application, there is still no unified and efficient parameter optimization method. The traditional particle swarm optimization (PSO) algorithm suffers from premature convergence and a high computational cost. Therefore, in this paper, we utilize an improved PSO algorithm, a reinforcement-learning-based memetic particle swarm optimization (RLMPSO), for the parameter tuning of ADRC to obtain better control performance for the controlled system. Finally, to highlight the advantages of the proposed RLMPSO-ADRC method and to prove its superiority, the results were compared with other control algorithms in both a traditional non-reheat two-area thermal power system and a non-linear power system with a governor dead band (GDB) and a generation rate constraint (GRC). Moreover, the robustness of the proposed method was tested by simulations with parameter perturbations and different working conditions. The simulation results showed that the proposed method can meet the demand for the frequency deviation to stabilize to 0 in LFC with higher performance, and it is worthy of popularization and application.

INDEX TERMS Interconnected power system, load frequency control, active disturbance rejection control, parameter optimization, reinforcement-learning-based memetic particle swarm optimization.

I. INTRODUCTION Maintaining the relative stability of the frequency and voltage is a prerequisite for ensuring the safe operation of a power system. As a well-known issue in power systems research, how to implement load frequency control (LFC) to ensure a constant frequency of a power system and improve the power quality and economic benefits have been widely studied. Furthermore, as the demand for power quality has increased, interconnected power systems have emerged. However, the exchange power of tie-lines in an interconnected power system is susceptible to disturbances,
which may cause unnecessary economic losses. Therefore, determining how to design a controller to ensure the safe, stable, and efficient operation of the power system is of great significance.

The essential requirement of a control strategy is to have the ability to process parameter uncertainty and achieve a good anti-disturbance performance while obtaining the desired dynamic performance to the greatest possible extent. Moreover, the design of the controller should not be too complicated to provide practical solutions for engineering debugging. At present, as an important means of active frequency modulation in power systems, LFC control strategies currently cover sliding mode control (SMC) [1], linear operator inequality [2], robust control [3], and predictive control [4].

As the most traditional control method, proportional–integral (PI) [5] and proportional–integral–derivative (PID) [6] controllers have been the most widely used in the LFC field due to their clear principles, simple implementation, and certain degrees of robustness. They can restore the stability of a power system, but their performances are poor. Other control strategies, such as adaptive control, can make corresponding real-time adjustments to controller parameters or rules, but the design is complex, and they are not easy to apply in industry [7]. A variable-structure controller can respond to system disturbances and parameter changes at faster speeds, thereby significantly improving the dynamic performance of the controlled system, but the practical applications of such a controller are limited [8]. For LFC, a robust control strategy can theoretically cope with the problems caused by system disturbances and parameter modeling errors. However, the order is high, and the design algorithm is relatively complex, relying on engineering experience [9]. Model predictive control (MPC) exhibits strong robustness and adaptability, and it does not require high model accuracy. However, the stability analysis and robust performance detection of multivariable MPC need further study [10].

In practical applications, PID still occupies a dominant position in engineering. Han [11], [12] first proposed an active disturbance rejection control (ADRC) technology, which was based on the PID control idea of eliminating the effects of disturbances based on errors. It attributes all uncertain factors and external disturbances of the system to the total disturbance of the system based on the system inputs and outputs. The disturbance can be eliminated by designing a control law. ADRC has been widely used in many industrial fields, such as aircraft control [13], motor control [14], ship control [15], and vehicle control [16]. ADRC has also been applied for LFC. For example, Rahman and Chowdhury [17] compared the control effects of ADRC and PID for an LFC system, and the simulation results showed that ADRC is a powerful substitute for PID and has significant performance advantages for LFC. Zheng et al. [18] applied ADRC to a three-area interconnected power system both in regulated and deregulated environments. It is worth mentioning that most of the controllers in ADRC currently designed for LFC use linear ADRC (LADRC), which was proposed by Gao [19], where the internal structures of the ADRC method have been greatly simplified. To a certain extent, this was caused by the difficulty of ADRC parameter tuning. Therefore, determining how to tune the parameters for ADRC is of great significance to promote its application.

Intelligent optimization algorithms are derived from the observation and simulation of biological systems in the natural world, and they are a new class of strategies suitable for optimization problems. In terms of parameter tuning, intelligent optimization algorithms, such as particle swarm optimization (PSO) [20], the genetic algorithm (GA) [21], simulated annealing (SA) [22], and ant colony optimization (ACO) [23], have shown good optimization capabilities. However, some limitations and shortcomings of intelligent optimization algorithms cannot be ignored. As mentioned in the ‘No Free Lunch (NFL)’ theorem, no single algorithm can be designated as the best algorithm that is applicable to all optimization problems. Therefore, various optimization and improved intelligent algorithms are constantly being proposed. For example, for the standard PSO, on the one hand, it can quickly fall into local optima at the beginning of the search process; on the other hand, the computational cost will increase with the increase in the sample population size [24]. Therefore, improved PSO algorithms such as the differential evolution particle swarm optimization (DEPSO) [25] and reinforcement-learning-based memetic particle swarm optimization (RLMPSO) [26] came into being. RLMPSO is an improved algorithm from a memetic algorithm (MA) perspective, where the MA is a hybrid algorithm that consist of a local search method, reinforcement learning (RL), and a globally optimal PSO algorithm.

In this study, the proposed ADRC is evaluated on a two-area thermal power system, which mainly contains a non-reheat turbine and nonlinear links with a generation rate constraint (GRC) and a governor dead band (GDB). For the first time, RLMPSO is used to adjust the controller’s parameters. To verify the effectiveness of the proposed method, simulation analysis on both linear and non-linear power systems with a GRC and a GDB was conducted, and the results were compared with those from other methods. Moreover, robustness tests were also carried out for uncertain parameter values and disturbances. The main contributions of this study are summarized as follows:

1. Aiming at the two-area interconnected power system with non-reheat turbines, two third-order ADRC controllers were designed.

2. The RLMPSO algorithm was used to optimize the eight parameters in the ADRC, and the effectiveness of the designed method was verified by comparison with other methods.

The rest of the paper is arranged as follows. Section 2 describes the mathematical model of the LFC system. In Section 3, the ADRC is designed for LFC. Section 4 introduces the parameter optimization process based on RLMPSO. Section 5 shows the simulation results, and the
corresponding analysis is presented. Section 6 concludes this paper.

II. POWER SYSTEM MODEL DESCRIPTION

LFC is a popular research subject for power systems, where the stability of the frequency is a prerequisite for the safe and reliable operation of the power grid. Fig. 1 shows a schematic diagram of the linear structure of the i-th area in the interconnected power system, which mainly includes three links: the governor $G_{gi}$, the turbine $G_{ti}$, and the generator $G_{pi}$. The governor mainly controls the guide vane or intake valve through the feedback of the speed deviation to control the speed and load of the turbine. The turbine acts as a prime mover to generate mechanical power to drive the generator to convert mechanical energy into electrical energy. Generally, due to the strong coupling between interconnected power systems, when the load disturbance occurs in one area, other areas will also be affected by this area, resulting in instability of the entire grid. Therefore, the main goal of LFC is to control the frequency deviation $1f_{i}$ within a safe range by overcoming the influence of load disturbances.

Furthermore, in the interconnected power system, the tie-line that exchanges power $1P_{tiei}$ between two adjacent areas also needs to be stabilized at the planned value. This study does not consider the load frequency control problem under deregulated environments involving economic benefits, so the planned value of the exchange power of the tie-line is 0. The mathematical expressions of the above three links are introduced as

$$
\begin{align*}
G_{gi} &= \frac{1}{T_{gi} s + 1}, \\
G_{ti} &= \frac{1}{T_{ti} s + 1}, \\
G_{pi} &= \frac{K_{pi}}{T_{pi} s + 1},
\end{align*}
$$

where $T_{gi}$, $T_{ti}$, and $T_{pi}$ are the time constant of the governor, the non-reheat turbine, and the generator. $K_{pi}$ represents the gain of the generator.

According to Fig. 1, $\Delta f_{i}$ can be derived as

$$
\Delta f_{i} = \frac{G_{gi} G_{ti} G_{pi}}{1 + G_{gi} G_{ti} G_{pi} R} u_{i} - \frac{G_{pi}}{1 + G_{gi} G_{ti} G_{pi} R} \Delta P_{Li} - \frac{G_{pi}}{1 + G_{gi} G_{ti} G_{pi} R} \Delta P_{tiei}.
$$

III. DESIGN OF ACTIVE DISTURBANCE REJECTION CONTROL (ADRC)

ADRC is derived from the combination of the classic PID and modern control theory, which has the advantages of not relying on model information and eliminating unknown disturbances. It only needs to know the order of the system. Therefore, this section mainly focuses on the design of the ADRC for the above model. For the multi-area interconnected power system, to stabilize the frequency deviation and the exchange power of the tie-line to 0, tie-line bias control (TBC) mode is adopted, which uses $ACE_{i}$ as the controller input.
In addition, in view of the coupling problem between areas, Tan [27] proposed a decentralized control method and carried out a theoretical proof, in which the controller in each area can be designed separately on the premise of ignoring the tie-line exchanged power.

A. TRANSFER FUNCTION PROCESSING
According to the decentralized control mentioned above, ignoring \( \Delta P_{tie} \), Eq. 3 can be written as
\[
Y_i(s) = G_i(s)U_i(s) + G_{di}(s)D_i(s),
\]
where \( Y(s), U(s), \) and \( D(s) \) are the Laplace transforms of \( ACE_i, u_i, \) and \( \Delta P_{tie} \). Combining this with Eq. 1, the expression of \( G_i(s) \) and \( G_{di}(s) \) can be presented as
\[
G_i(s) = \frac{K_{pi}B_R}{T_i(s) + 1} \frac{R_i + K_{pi}}{(T_i(s) + 1) R_i + K_{pi}},
\]
\[
G_{di}(s) = \frac{K_{pi}B_R}{T_i(s) + 1} \frac{(T_i(s) + 1) R_i + K_{pi}}{(T_i(s) + 1) R_i + K_{pi}}.
\]
Converting the above equation into a differential equation form, we obtain
\[
T_iT_iT_i\hat{y}_i(t) + (T_i + T_i + T_i + T_i + T_i + T_i + T_i + T_i) \hat{y}_i(t) + y_i(t)
+ (T_i + T_i + T_i + T_i + T_i + T_i + T_i + T_i) \hat{y}_i(t) + y_i(t)
= K_{pi}B_Ru(t) + T_iT_iB_R \hat{d}(t) + T_iT_iB_R \hat{d}(t) + T_iT_iB_R \hat{d}(t).
\]
Therefore, Eq. 3 can be organized into the following third-order system as
\[
y(t) = f_0(\hat{y}, \hat{y}, \hat{d}, \hat{d}) + bu
= f_0 + (b - b_0)u + b_0u
= f + b_0u,
\]
where \( b = \frac{K_{pi}B_R}{T_iT_iT_i} \). \( f \) is the total disturbance containing the uncertainties caused by parameter perturbations of the system model and external disturbances caused by load disturbances. Since the actual value of \( b \) cannot be known in practice, the adjustable parameter \( b_0 \) is used to substitute \( b \).

B. DESIGN OF ADRC FOR LOAD FREQUENCY CONTROL (LFC)
ADRC is composed of a tracking differentiator (TD), extended state observer (ESO), and nonlinear state error feedback (NLSEF), where the TD can suppress the noise amplification effect in the input signal, the ESO can estimate the total unknown disturbance \( f \), and the NLSEF can eliminate the estimated disturbance. Moreover, the ADRC structure diagram corresponding to the third-order system is shown in Fig. 2.

Because the control goal in this study seeks to make \( ACE_i(t) = 0 \) at steady state, the tracking reference input \( u = 0 \). Moreover, since the load disturbance of the power system does not contain white noise, there is no need for a filter. Thus, the design of the TD can be ignored.

According to Eq. 8, the state can be defined as
\[
x_1 = y, x_2 = \dot{y}, x_3 = \ddot{y}, x_4 = f.
\]
The corresponding state equation is
\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= x_3 \\
\dot{x}_3 &= f + b_0u \\
\dot{x}_4 &= y = x_1.
\end{align*}
\]
The ESO can be expressed as
\[
fal(e, a, \delta) = \begin{cases} 
\frac{e}{\delta a^{-1}}, & |e| \leq \delta \\
\frac{\delta}{\delta a^{-1}}, & |e| > \delta,
\end{cases}
\]
where \( a \) and \( \delta \) are the adjustable parameters.

After the disturbance state is estimated, it needs to be eliminated. We define \( e_i = v_i - z_i, i = 1, 2, 3, \) and then the control law can be designed as
\[
u_0 = \sum_{i=1}^{3} \beta_i fal(e_i, a_i, \delta_0),
\]
where \( \beta_i, i = 1, 2, 3 \) are the feedback control gains, and \( fal(\cdot) \) has the same form as Eq. 12.
\( u_i \) in Eq. 8 is then expressed as

\[
u = \frac{u_0 - z_4}{b_0}.
\]

By substituting Eq. 13 and 14 into Eq. 8, under the premise of \( f \approx z_4 \), we can obtain

\[
y^{(3)} \approx u_0.
\]

Therefore, appropriate feedback control gains can ensure that \( \text{ACE}_{i} \) converges to 0. In this study, \( \delta = 0.05, \delta_0 = 0.005, a_1 = 0.3, a_2 = 0.8, \) and \( a_3 = 1.1 \).

The above design process shows that for a third-order controlled system, the parameters that need to be adjusted in the ADRC are \( \beta_01, \beta_02, \beta_03, \beta_04, \beta_1, \beta_2, \beta_3, \) and \( b_0 \). It is very difficult to manually adjust eight parameters at the same time, and an effective optimization method is urgently needed.

**IV. REINFORCEMENT-LEARNING-BASED MEMETIC PARTICLE SWARM OPTIMIZATION (RLMPSO)-OPTIMIZED ADRC**

As a control technology for estimating and compensating for uncertain disturbances, ADRC has attracted widespread attention in industry and academia since its proposal. However, there has not been a set of efficient and unified tuning rules for parameter adjustment. In this study, the RLMPSO algorithm is applied for the parameter tuning problem of the nonlinear ADRC to test the superiority of the combination of RL and the intelligent optimization algorithm in a new field.

**A. BASICS OF PARTICLE SWARM OPTIMIZATION (PSO) AND REINFORCEMENT LEARNING (RL)**

1) **PRINCIPLE OF PSO**

The PSO originated from the simulation of bird predation behavior. The main concepts of PSO include a population, potential solutions (called particles), and iterative search space, where each particle is composed of a position, speed, and fitness value. It moves at an adaptive speed in the search space, where each particle is composed of a position, speed, potential solutions (called particles), and iterative search behavior. The main concepts of PSO include a population, the group. The position and velocity of each particle in the population are updated according to Eq. 16.

\[
\begin{align*}
V_{k+1} &= c_1 r_1 (p_{\text{best}} - X_k) + c_2 r_2 (g_{\text{best}} - X_k) \\
X_{k+1} &= V_{k+1} + X_k,
\end{align*}
\]

where \( k \) is the current iteration number, \( X_k \) is the current position of the particle, and \( V_k \) is the current velocity of the particle. \( c_1 \) and \( c_2 \) represent the learning factors.\( \omega_k \) denotes the inertia weight. \( r_1 \) and \( r_2 \) are random numbers between [0, 1].

The initial population is generally randomly generated by

\[
\begin{align*}
X_i &= R_0 (U - L) + L \\
V_i &= R'_0 (V_{\text{max}} - V_{\text{min}}) + V_{\text{min}},
\end{align*}
\]

where \( X_i \) and \( V_i \) represent the position and velocity of the particle, respectively, \( R_0 \) and \( R'_0 \) are the random vector with the same dimension as \( X_i \), and each component is between [0, 1]. \( U \) and \( L \) represent the upper and lower bounds of the particle position (that is, the parameter solution), respectively. \( V_{\text{max}} \) and \( V_{\text{min}} \) denote the upper and lower bounds of the particle velocity, respectively.

2) **PRINCIPLE OF RL**

RL is a method to find the optimal strategy through the continuous interaction between an agent and an uncertain environment. When the intelligent agent “communicates” with the environment through actions, the environment will return the current reward to the agent, through which the action can be evaluated [28]. The basic framework is shown in Fig. 3.

**FIGURE 3.** Basic framework of reinforcement learning (RL).

We assume that the environment produces state \( s_t \) at time \( t \), and the reward value \( r_t \) can be obtained based on the reward function. The agent can obtain the optimal action \( a_t \) through the state-action value function based on the cumulative reward \( R_c \), where the state-action value function can be regarded as the evaluation value of the action.

Q-learning is the most common RL algorithm, first proposed by Watkins in 1989 [29]. As an RL method based on a time difference, the selection of the current state and action in Q-learning can be regarded as an event, and any event corresponds to a state-action value function \( Q(s_t, a_t) \), which is stored in the \( Q \) table. Through the updated iteration of the \( Q \) table, the intelligent agent in Q-learning will gradually approach the optimal strategy for sequential decision-making problems in the continuous in-depth interaction between the agent and the environment. A typical learning process of Q-learning can be described as follows:

Step 1: Model initialization. For all discrete states \( s \) and actions \( a \), initialize their corresponding value functions \( Q(s, a) \in Q \).
Step 2: Initialize state $s$. On the basis of a Q table, select the initial action value $a$ by an $\epsilon$-greedy policy, shown as

$$
\pi(a, s) \left\{ \begin{array}{ll}
1 - \epsilon + \frac{\epsilon}{|A(s)|} & \text{if } a = \arg \max_{a \in A} Q(s, a) \\
\frac{\epsilon}{|A(s)|} & \text{otherwise,}
\end{array} \right.
$$

(18)

where $\epsilon$ represents the probability of exploration.

From Eq. 18, we observe that the probability of selecting an action corresponding to the largest Q value is $\epsilon$. Otherwise, an action value is randomly selected from the action space.

Step 3: Perform action $a_t$, thus obtaining the corresponding reward value $r_t$ and the state $s_{t+1}$ at the next moment.

Step 4: Update $Q(s_t, a_t)$ according to

$$
Q_{t+1}(s_t, a_t) = Q(s_t, a_t) + \alpha \left[ r_{t+1} + \gamma \max_{a} Q(s_{t+1}, a) - Q(s_t, a_t) \right],
$$

(19)

where $\alpha$ is the learning rate. $\gamma$ represents the discount factor, which reflects the importance of rewards for future moments.

Step 5: Determine whether to end the process. If yes, output the optimal strategy; otherwise, return to the third step.

**B. RLMPSO**

The particle swarm algorithm suffers from premature convergence and a high computational cost. Improving the particle swarm algorithm from the perspective of a memetic algorithm is a significant research direction.

1) **PRINCIPLE OF RLMPSO**

The main idea of RLMPSO is to embed RL into the operation of each particle search stage in the particle swarm algorithm. Under RL control, each particle performs one of five possible operations: exploration, convergence, high-jump, low-jump, and fine-tuning. Moreover, each action will be rewarded or punished based on the performance. In addition, the population size in the RLMPSO algorithm is small, and each particle evolves independently. For example, one particle performs exploration, while other particles perform their own operations. The schematic diagram of the RLMPSO structure is shown in Fig. 4, where Q-learning is adopted.

![Fig. 4. Structure diagram of reinforcement-learning-based memetic particle swarm optimization (RLMPSO).](image)

**2) DEFINITION OF ACTION**

As mentioned earlier, there are five operations that each particle can perform. The five operations are introduced in this section.

Exploration and convergence are two operations defined by the difference in the power of the particle’s global search during the search process and the preference of whether to track the current global optimal position $g_{best}$, which is realized by the differences of the values of $\omega$, $c_1$, and $c_2$. For the exploration, the particle probes the solution space with a larger $\omega$, and to maximize the global search, the particle will be far from the current global optimal position $g_{best}$, where $c_1 > c_2$. The convergence operation is the opposite. The search power of the particles in the solution space will be attenuated. Thus, particles will converge in the direction of $g_{best}$. At this time, $c_1 < c_2$, and $\omega$ is small.

The main idea of the jump operation is to avoid premature convergence of particles by changing the individual optimal particle $p_{best,i}$ ($i$ is the particle subscript) to escape the possible local optima. Specifically, a random value is added to each dimension (that is, each parameter value to be optimized) of $p_{best,i}$, shown as

$$
X_i = p_{best,i} + r_n(U - L),
$$

(20)

where $r_n$ is a normally distributed random number in the range $[0, 1]$, that is, $r_n \sim N(0, \sigma^2)$. For the high-jump operation, this means that the particle will change $p_{best,i}$ with a larger step length, and the standard deviation $\sigma$ is closer to 1 at this time, while for the low-jump operation, the particle step size is smaller, and the standard deviation $\sigma$ is close to 0.

Similar to the jump operation, the fine-tuning operation also adds a random value to each dimension of the individual optimal particle $p_{best,i}$, so that the entire population performs a local search within the current global optimal solution neighborhood. The difference is that the fine-tuning...
of each dimension of the particles that needs to be manipulated is performed independently, and the fine-tuning operation will perform a certain number of fitness evaluations in cycles when each dimension changes. During the evaluation process, \( p_{\text{best},i} \) and the corresponding speed variable \( V_{i,d} \) (\( i \) represents the current particle subscript, and \( d \) is the dimension subscript) will change as the fitness value changes.

For example, supposing that particle \( i \) performs fine-tuning operations, the maximum dimension (the number of parameters to be optimized) is \( D \), and the maximum number of fitness evaluations is \( E_m \). For the current dimension \( d \in [1, D] \), the current fitness evaluation number of this dimension is \( e \in [1, E_m] \), and the minimum objective function is \( f(x) \). The realization process (*) is described as follows:

Step 1: Update the speed using the following equation, where \( L_{i,d} \) is the step size, \( a \) is the acceleration factor, \( p \) is the parameter that controls the speed attenuation, and \( r \) is a uniformly distributed random number in \([-0.5, 0.5]\):

\[
V_{i,d} = \frac{a}{ep^r} + L_{i,d}. \tag{21}
\]

Step 2: Record the original best fitness value \( f_{\text{best}} \) and calculate \( f(p_{\text{best},i} + V_{i,d}) \).

Step 3: Update \( p_{\text{best},i} \) as follows:

\[
p_{\text{best},i} = \begin{cases} p_{\text{best},i} + V_{i,d}, & \text{if } f_{\text{best}} > f(p_{\text{best},i} + V_{i,d}) \\ p_{\text{best},i}, & \text{otherwise}. \end{cases} \tag{22}
\]

Step 4: Update \( L_{i,d} \) with the following equation and let \( e = e + 1 \):

\[
L_{i,d} = \begin{cases} 2V_{i,d}, & \text{if } f_{\text{best}} > f(p_{\text{best},i} + V_{i,d}) \\ \frac{1}{2}L_{i,d}, & \text{otherwise}. \end{cases} \tag{23}
\]

Step 5: If \( e \leq E_m \), return \( a \); otherwise, let \( d = d + 1 \).

3) Q TABLE

Unlike the standard PSO, RLMPSO can perform any operation at any stage of the search process, namely exploration, convergence, high-jump, low-jump, and fine-tuning. RL is responsible for tracking the best performance of each particle.

Each particle has a Q table that only belongs to itself and not to the population during the search process. The dimension of the Q table is \( 5 \times 5 \):

\[
\begin{pmatrix}
E & C & H & L & F \\
\hline
E & (a_{11}, a_{12}, a_{13}, a_{14}, a_{15}) \\
C & (a_{21}, a_{22}, a_{23}, a_{24}, a_{25}) \\
H & (a_{31}, a_{32}, a_{33}, a_{34}, a_{35}) \\
L & (a_{41}, a_{42}, a_{43}, a_{44}, a_{45}) \\
F & (a_{51}, a_{52}, a_{53}, a_{54}, a_{55})
\end{pmatrix}. \tag{24}
\]

In Eq. 24, rows represent states, and columns represent actions. \( E, C, H, L, \) and \( F \) represent the five operations, exploration, convergence, high-jump, low-jump, and fine-tuning, respectively. The state of \( E \) indicates that the current particle execution operation is exploration, and the action of \( E \) indicates that the next operation performed by the particle is exploration.

Because the number of fitness evaluations for each dimension in the fine-tuning operation will be calculated based on the number of iterations, the fine-tuning operation requires a large number of global iterations, while other operations only require one. At the same time, since the fine-tuning operation is an operation on the optimal position of the individual, the execution of the fine-tuning must be postponed to the global operation, that is, after the exploration, convergence, and jump operations. Therefore, to delay the execution of the fine-tuning at the beginning of the search process and give higher priority to the other operations to be performed, the initial Q table entry of action \( F \) (corresponding to the last column in the Q table) is set to an infinitesimal negative value. Before activating the fine-tuning operation, RLMPSO must perform \( M \) iterations and consider the minimum value (including negative numbers) of the other four columns in the Q table as the column item’s initial value of the action \( F \) is located. During the RLMPSO execution, the best action can be retrieved in the current state from the Q table, presented as

\[
a_t + 1 = \arg \max_{a_i} \{Q(s_t, a_i)\}, \tag{25}
\]

where \( a_t + 1 \) is the action to be performed next, that is, the best action for the current state, \( s_t \) represents the current state, \( a_i \in A \) denotes an action, and \( A \) is the action space collection.

In summary, the principle diagram of the movement adjustment of each particle in the RLMPSO is shown in Fig. 5.

C. DESIGN OF RLMPSO-OPTIMIZED ADRC

As reported previously [26], RLMPSO performs significantly better than many other PSO variants in optimizing multiple unimodal functions, multimodal functions, composite functions, and two practical optimization problems involving train gear and pressure vessel design. This section uses the RLMPSO algorithm to optimize the ADRC parameters to better control the control system’s dynamic performance. The following time-weighted integral of absolute error (ITAE) is selected as the objective function:

\[
\text{ITAE} = \int_0^T \left( \sum_{i=1}^{n} |\Delta f_i| + \sum_{i=1}^{n} \sum_{j=1}^{i-1} |\Delta P_{\text{tie},ij}| \right) dt, \tag{26}
\]
where \( n \) is the total number of controlled system areas, and \( T \) is the simulation time. When the ITAE decreases, the controller receives an instant reward 1; otherwise, it receives a \(-1\). The schematic diagram of the ADRC based on the RLMPSO algorithm is shown in Fig. 6.

The parameters of the RLMPSO are shown in Table 2. In addition, the ranges \([L, U]\) of the controller parameters are given in Table 3. The corresponding speed range is 0.2\([L, U]\).

The parameters of the RLMPSO are shown in Table 2. In addition, the ranges \([L, U]\) of the controller parameters are given in Table 3. The description of the optimization process is shown in Fig. 7.

V. NUMERICAL SIMULATION RESULTS AND ANALYSIS

A. TRADITIONAL TWO-AREA NON-REHEAT THERMAL POWER SYSTEM

In this section, the two-area non-reheat thermal power system [30] is used as the simulation model, where the two areas of the interconnected power system have the same structure, and each area contains a non-reheat turbine. The model parameters are listed as

\[
\begin{align*}
B_1 &= B_2 = 0.425 \text{p.u. MW/Hz}; \\
R_1 &= R_2 = 2.4 \text{Hz/p.u.}; T_{s1} &= T_{s2} = 0.03 \text{s}; \\
T_{11} &= T_{12} = 0.3 \text{s}; K_{p1} = K_{p2} = 120 \text{Hz/p.u.}; \\
T_{p1} &= T_{p2} = 20 \text{s}; T_{12} = 0.545 \text{p.u.}; a_{12} = -1.
\end{align*}
\]  

FIGURE 6. Schematic diagram of ADRC based on RLMPSO.

TABLE 2. Parameters of RLMPSO.

| Parameter                  | Values                     |
|----------------------------|----------------------------|
| Population size \( n \)    | 5                          |
| Cognitive factor \( c_2 \) | Exploration: 2.5, Convergence: 0.5 |
| Parameters of fine-tuning  | \( B_m = 30, p = 20, a = 150 \) |
| Inertia weight \( \omega \) | Exploration: 0.9, Convergence: 0.4 |
| Social factor \( c_2 \)     | Exploration: 0.5, Convergence: 2.5 |
| Maximum number of iterations \( m_{max} \) | \( 10^5 \) |

TABLE 3. Ranges of ADRC parameters.

| Parameter | Range                      |
|-----------|----------------------------|
| \( \beta_{10} \) | \([0,400]\) |
| \( \beta_{20} \) | \([0,8000]\) |
| \( \beta_{30} \) | \([0,80000]\) |
| \( \beta_{40} \) | \([0,80000]\) |
| TLSEF      | \([0,1000]\) |
| \( \beta_1 \) | \([0,1000]\) |
| \( \beta_2 \) | \([0,1000]\) |
| \( \beta_3 \) | \([0,2500]\) |
| \( \beta_4 \) | \([0,5000]\) |

TABLE 4. ADRC optimization results of two-area non-reheat thermal power system based on RLMPSO.

| Control method | \( \Delta f_1 \) | \( \Delta f_2 \) | \( \Delta f_s \) | ITAE   |
|----------------|-----------------|-----------------|-----------------|--------|
| RLMPSO-ADRC    | 0.22            | 0.0             | 0.0             | 0.0015 |
| DSA-FOPI-FOPD  | 0.72            | 1.48            | 0.86            | 0.018  |
| DSA-FOPID      | 1.13            | 2.73            | 2.23            | 0.0778 |
| ISFS-PID       | 2.15            | 3.66            | 3.01            | 0.1516 |
| hPSO-PS-FUZZY-PID | 2.26          | 3.74            | 2.94            | 0.1438 |
| TLBO-ZDOPF-PID | 2.41            | 2.13            | 2.56            | 0.0269 |
| TLBO-PID       | 4.84            | 5.13            | 3.03            | 0.2452 |
| HFOO-AFO-PI    | 5.17            | 6.81            | 4.59            | 1.457  |
| BFOA-PI        | 5.52            | 7.09            | 6.35            | 1.827  |

TABLE 5. Comparison of performance indices of a two-area non-reheat thermal power system with different controllers.

1) SIMULATION ANALYSIS OF NOMINAL PARAMETERS

Supposing that when \( t = 0 \), the first area is disturbed by a step-load disturbance (SLP) \( \Delta P_{L1} = 0.1 \text{ p.u.} \). According to the discussion above, each area adopted a third-order ADRC. The optimized parameters are shown in Table 4. Fig. 8 shows the response results of the frequency deviation and tie-line exchanged power, where the optimization results of various algorithms, including BFOA-PID [31], HFOA-FOPI-PID [32], HPSO-PS-FUZZY-PID [33], TLBO-PID [34], ISFS-PID [35], DSA-FOPID [36], and DSA-FOPI-FOPD [36], are also given.

Table 5 shows the adjustment time and ITAE performance index results of the frequency deviation in each area and the system tie-line exchanged power.

Fig. 8 shows that the frequency deviation and tie-line power deviation in each area were stable at zero in the steady state, which means that the steady-state performances of the two controllers were the same. As shown in Table 5, the proposed LFC controller with RLMPSO-ADRC achieved a significantly smaller ITAE value than the LFC controllers from previous studies. The ITAE value under this method was 0.00015, which was \( \frac{1}{120} \) of the optimal ITAE value of the other methods (0.018 of DSA-FOPI-FOPD). This means that the proposed method has a better control performance.

2) ROBUSTNESS ANALYSIS

In modern complex power systems, the uncertainty of the system parameters is a crucial issue. Therefore, it is crucial for the LFC controller to be robust to the uncertainty of the parameters in the system. To examine the robustness of the proposed control strategy, Fig. 9 and 10 show time-domain
response curves when the system model parameters, $B$, $R$, $T_p$, $K_p$, $T_g$, and $T_r$, were changed by $+30\%$ and $-30\%$, respectively. The response performance of the proposed method was better than those of the other methods, i.e., the minimum overshoot and undershoot and the shortest stabilization time, thus verifying the robustness of the proposed method.
In addition, to test the suppression ability of the proposed method to different load disturbances, Fig. 11 shows the output response of the system controlled by the proposed method with different load disturbances, where $P_{L1} = 0$ p.u. and $P_{L2} = 0.1$ p.u. at $t = 0–10$ s, and $P_{L1} = 0.1$ p.u. and $P_{L2} = 0.1$ p.u. at $t = 10–20$ s.

As shown in Fig. 11, the RLMPSO-ADRC control system could always suppress the system frequency deviation and the fluctuations of the tie-line power deviation under different load disturbances and restore the system to a stable state in a very short time, thus demonstrating that the proposed LFC controller had good disturbance rejection capabilities and robustness.

### B. TWO-AREA POWER SYSTEM WITH NONLINEARITY

In practice, the system will inevitably be subject to the internal constraints of the physical system dynamics. The nonlinearity caused by both the GRC and the GDB are considered, which are shown in Fig. 12 and 13, respectively. The GRC is a saturated nonlinear phenomenon that occurs when the generator has difficulty responding on-demand when a large load disturbance occurs and cannot provide a sufficient rate of change. The GDB can avoid the loss of control due to the excessive response of the governor valve. The model parameters are shown in Eq. 28, where $l_d$ and $u_d$ are the bounds of the GDB, and $l_g$, $u_g$ are the bounds of the GRC.

$$
\begin{align*}
B_1 &= 0.3483p.u.MW/Hz; T_{g1} = 0.08s; \\
T_1 &= 0.40s; T_{p1} = 11.1333s; \\
K_p &= 66.67Hz/p.u.; T_{12} = 0.20p.u.; R_1 = 3Hz/p.u.; \\
l_1 &= -0.05; u_1 = 0.05; l_{g1} = -0.167; u_{g1} = 0.167; \\
B_2 &= 0.3823p.u.MW/Hz; T_{g2} = 0.06s; \\
T_2 &= 0.44s; T_{p2} = 13.5625s; \\
K_p &= 62.50Hz/p.u.; T_{12} = 0.20p.u.; R_2 = 2.73Hz/p.u.; \\
l_2 &= -0.0005; u_2 = 0.0005; \\
l_{g2} &= -0.00167pu/sec; u_{g2} &= 0.00167pu/sec.
\end{align*}
$$

(28)
TABLE 7. Comparison of performance indices of two-area power systems with GDB and GRC under different controllers.

| Control method      | $\Delta f_1$ | $U_{sh}$ | $O_{sh}$ | $T_s$ | $\Delta f_2$ | $U_{sh}$ | $O_{sh}$ | $T_s$ | $\Delta P_{tie}$ | $U_{sh}$ | $O_{sh}$ | $T_s$ | IAE  |
|---------------------|--------------|----------|----------|-------|--------------|----------|----------|-------|----------------|----------|----------|-------|------|
| RLMPSO-ADRC         | -0.002       | 0        | 2.39     | -0.004| 0            | 2.46     | -0.00002 | 0.005 | 9.5           | -0.0029  | 0.015    | 8.6   | 3.716|
| PEO-EDMPC [37]      | -0.039       | 0.007    | 9.66     | -0.034| 0.005        | 9.5      | -0.00004 | 0.017 | 9.3           | 0.0043   | 0.017    | 9.42  | 4.479|
| DMPC [38]           | -0.055       | 0.00005  | 11.1     | -0.036| 0.002        | 11.6     | -0.0004  | 0.017 | 9.3           | 13.24    | 0.017    | 9.42  | 4.479|
| PEO-DPI [39]        | -0.044       | 0.015    | 13.89    | -0.036| 0.01        | 13.24    | -0.0043  | 0.017 | 9.42          | 13.24    | 0.017    | 9.42  | 4.479|
| DI [38]             | -0.049       | 0.0013   | 27.47    | -0.036| 0.0011       | 26.73    | -0.001   | 0.015 | 22.41         | 27.47    | 0.015    | 22.41 | 7.427|

Similarly, using the third-order ADRC, where the controller parameter ranges are still as shown in Table 3, the optimized parameter results are shown in Table 6. For a step load disturbance of $\Delta P_{L2} = 0.02$ p.u. added to the second area at $t = 0$, the simulation results are shown in Fig. 14. To facilitate the performance comparison with other algorithms, this section uses the following performance indicators for evaluation: integral absolute error (IAE), overshoot $O_{sh}$, undershoot $U_{sh}$, and settling time $T_s$, as shown in Table 7, where $IAE = \int_0^\infty |e(t)|dt$.

As shown by the response curve of the time-domain system in Fig. 14, the RLMPSO-ADRC had the smallest fluctuations of $\Delta f_1$, $\Delta f_2$, and $\Delta P_{tie}$ compared with the other four algorithms and the fastest recovery speed. Table 7 also numerically shows the effectiveness of the proposed method. The transient performance and response speed were significantly better than those of other algorithms, thus verifying the effectiveness of the proposed method for the LFC control system.

VI. CONCLUSION

In this study, for the problem of frequency instability caused by load disturbances in a power system, a load-frequency active disturbance rejection controller was designed. In view of the difficulty of determining the controller parameters, we introduced the RLMPSO algorithm, with a better convergence speed, which combines RL with PSO. To verify the effectiveness of the proposed RLMPSO-PSO, we conducted many simulations and comparison experiments. We first applied the traditional two-area non-reheat thermal power system method, and the performance is compared with those of previously reported methods. Compared with other strategies, the proposed method showed a smaller response deviation and settling time, thus highlighting the advantages of the proposed method. The robustness of the method in the power system was studied. The system response under critical parameter states and different disturbance conditions maintained good dynamics, proving that the proposed method had good robustness to the uncertainties of the system. Finally, the proposed method was applied to a nonlinear power system containing a GDB and a GRC, and the dynamic response performance of the system was compared with control strategies, such as predictive control and conventional control. The results showed that the method had a stronger ability to suppress system load disturbances, which indicated that the proposed method is an effective solution to LFC problems. Therefore, the proposed method has theoretical significance and research value for the application of ADRC in actual LFC systems.
REFERENCES

[1] J. Guo, “Application of full order sliding mode control based on different areas power system with load frequency control,” ISA Trans., vol. 92, pp. 23–34, Sep. 2019.
[2] C. Hua and Y. Wang, “Delay-dependent stability for load frequency control system via linear operator inequality,” IEEE Trans. Cybern., early access, Dec. 14, 2021, doi: 10.1109/TCYB.2020.3037113.
[3] H. Zhang, J. Liu, and S. Xu, “H-infinity load frequency control of networked power systems via an event-triggered scheme,” IEEE Trans. Ind. Electron., vol. 67, no. 8, pp. 7104–7113, Aug. 2020.
[4] Y. Jia, Z. Y. Dong, C. Sun, and K. Meng, “Cooperation-based distributed economic MPC for economic load dispatch and load frequency control of interconnected power systems,” IEEE Trans. Power Syst., vol. 34, no. 5, pp. 3964–3968, Sep. 2019.
[5] H. Shayeghi, H. A. Shayanfar, and A. Jalili, “Load frequency control strategies: A state-of-the-art survey for the researcher,” Energy Convers. Manage., vol. 50, no. 2, pp. 344–353, Feb. 2009.
[6] W. Tan, “Unified tuning of PID load frequency controller for power systems via IMC,” IEEE Trans. Power Syst., vol. 25, no. 1, pp. 341–350, Feb. 2010.
[7] R. Ramachandran, B. Madasamy, V. Veerasamy, and L. Saravanan, “Load frequency control of a dynamic interconnected power system using generalised Hopfield neural network based self-adaptive PID controller,” IET Gener., Transmiss. Distrib., vol. 12, no. 21, pp. 5713–5722, Nov. 2018.
[8] P. Daihya, V. Sharma, and R. Naresh, “Optimal sliding mode control for frequency regulation in deregulated power systems with DFIG-based wind turbine and TCSC–SMES,” Neural Comput. Appl., vol. 31, no. 7, pp. 3039–3056, Jul. 2019.
[9] C. Komboigo, U. Naomitsu, S. Tomonobu, E. L. Mohammed, and L. Liu, “Robust load frequency control schemes in power system using optimized PID and model predictive controllers,” Energies, vol. 11, no. 11, pp. 1–18, Nov. 2018.
[10] J. Liu, Q. Yao, and Y. Hu, “Model predictive control for load frequency of hybrid power system with wind power and thermal power,” Energy, vol. 172, pp. 555–565, Apr. 2019.
[11] J. Han, “Auto-disturbance-rejection controller and its applications,” Control Decis., vol. 13, no. 1, pp. 19–23, 1998.
[12] J. Han, “From PID to active disturbance rejection control,” IEEE Trans. Ind. Electron., vol. 56, no. 3, pp. 900–906, Mar. 2009.
[13] Z. Wang, R. Zu, D. Duan, and J. Li, “Tuning of ADRC for QTR in transition process based on NBPO hybrid algorithm,” IEEE Access, vol. 7, pp. 177219–177240, 2019.
[14] W. Lu, Q. Li, K. Lu, Y. Lu, L. Guo, W. Yan, and F. Xu, “Load adaptive PMSM drive system based on improved ADRC for manipulator joint,” IEEE Access, vol. 9, pp. 33369–33384, 2021.
[15] J. Tao, L. Du, M. Dehmer, Y. Wen, G. Xie, and Q. Zhou, “Path following control for towing system of cylindrical drilling platform in presence of disturbances and uncertainties,” ISA Trans., vol. 95, pp. 185–193, Dec. 2019.
[16] Y. Xia, F. Pu, S. Li, and Y. Gao, “Lateral path tracking control of autonomous land vehicle based on ADRC and differential flatness,” IEEE Trans. Ind. Electron., vol. 63, no. 5, pp. 3091–3099, May 2016.
[17] M. M. Rahman and A. H. Chowdhury, “Comparative study of ADRC and PID based load frequency control,” in Proc. Int. Conf. Electr. Eng. Inf. Commun. Technol. (ICEEI'T), Savar, Bangladesh, May 2015, pp. 21–23.
[18] Y. Zheng, Z. Chen, Z. Huang, M. Sun, and Q. Sun, “Active disturbance rejection controller for multi-area interconnected power system based on reinforcement learning,” Neurocomputing, vol. 425, pp. 149–159, Feb. 2021.
[19] Z. Gao, “On the centrality of disturbance rejection in automatic control,” ISA Trans., vol. 53, no. 4, pp. 850–857, Feb. 2014.
[20] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in Proc. IEEE Int. Conf. Neural Netw., Denver, CO, USA, 2002.
[21] Y. Zhang, P. Li, and X. Wang, “Intrusion detection for IoT based on improved genetic algorithm and deep belief network,” IEEE Access, vol. 7, pp. 31711–31722, 2019.
[22] W. L. Jorgensen, “Perspective on equation of state calculations by fast computing machines,” Theor. Chem. Acc., vol. 103, nos. 3–4, pp. 225–227, 2000.
[23] W. Deng, J. Xu, and H. Zhao, “An improved ant colony optimization algorithm based on hybrid strategies for scheduling problem,” IEEE Access, vol. 7, pp. 20281–20292, 2019.
[24] Y. Zhang, S. Wang, P. Phillips, and G. Ji, “Binary PSO with mutation operator for feature selection using decision tree applied to scan detection,” Knowl.-Based Syst., vol. 64, pp. 22–31, Jul. 2014.
[25] W.-J. Zhang and X.-F. Xie, “DEPSO: Hybrid particle swarm with differential evolution operator,” in Proc. Conf. IEEE Int. Conf. Syst., Man Cybern. Conf. Theme Syst. Secure. Assurance (SMC), Washington, DC, USA, Oct. 2003, pp. 3816–3821.
[26] H. Samma, C. P. Lim, and J. M. Saleh, “A new reinforcement learning-based memetic particle swarm optimizer,” Appl. Soft Comput., vol. 43, pp. 276–297, Jun. 2016.
[27] W. Tan, “Tuning of PID load frequency controller for power systems,” Energy Convers. Manage., vol. 50, no. 6, pp. 1465–1472, Jun. 2009.
[28] D. Dewey, “Reinforcement learning and the reward engineering principle,” in Proc. AAAI Spring Symp. Ser., Jun. 2014, vol. 1, no. 13, pp. 13–16.
[29] C. Watkins, Learning From Delayed Rewards. Cambridge, U.K.: Cambridge Univ. Press, 1989.
[30] M. R. Sathyaa and M. M. T. Ansari, “Load frequency control using bat inspired algorithm based dual mode gain scheduling of PI controllers for interconnected power system,” Int. J. Elect. Power Energy Syst., vol. 64, pp. 365–374, Jan. 2015.
[31] E. S. Ali and S. M. Abd-Elazim, “Bacteria foraging optimization algorithm based load frequency controller for interconnected power system,” Int. J. Elect. Power Energy Syst., vol. 33, no. 3, pp. 633–638, Mar. 2011.
[32] S. Panda, B. Mohanty, and P. K. Hota, “Hybrid BFOA–PSO algorithm for automatic generation control of linear and nonlinear interconnected power systems,” Appl. Soft Comput., vol. 13, no. 12, pp. 4718–4730, Dec. 2013.
[33] R. K. Sahu, S. Panda, and G. T. C. Sekhar, “A novel hybrid PSO-PSO optimized fuzzy PI controller for AGC in multi area interconnected power systems,” Int. J. Electr. Power Energy Syst., vol. 64, pp. 880–893, Jan. 2015.
[34] R. K. Sahu, S. Panda, U. K. Rout, and D. K. Sahoo, “Teaching learning based optimization algorithm for automatic generation control of power system using 2-DOF PID controller,” Int. J. Electr. Power Energy Syst., vol. 77, pp. 287–301, May 2016.
[35] E. Celik, “Improved stochastic fractal search algorithm and modified cost function for automatic generation control of interconnected electric power systems,” Eng. Appl. Artif. Intell., vol. 88, no. 2, pp. 1–20, Feb. 2020.
[36] E. Celik, “Design of new fractional order PI–fractional order PD cascade controller through dragonfly search algorithm for advanced load frequency control of power systems,” Soft Comput., vol. 25, no. 2, pp. 1193–1217, Jan. 2021.
[37] G. Q. Zeng, X.-Q. Xie, M.-R. Chen, and J. Weng, “Adaptive population extremal optimization-based PID neural network for multivariable nonlinear control systems,” Swarm Evol. Comput., vol. 44, pp. 320–334, Feb. 2019.
[38] T. H. Mohamed, H. Bevrani, A. A. Hassan, and T. Hijama, “Decentralized model predictive based load frequency control in an interconnected power system,” Energy Convers. Manage., vol. 52, no. 2, pp. 1208–1214, 2011.
[39] G.-Q. Zeng, J. Chen, M.-R. Chen, Y.-X. Dai, L.-M. Li, K.-D. Lu, and C.-W. Zheng, “Design of multivariable PID controllers using real-coded population-based extremal optimization,” Neurocomputing, vol. 151, pp. 1343–1353, Mar. 2015.

YUEMIN ZHENG was born in 1996. She received the B.E. degree from Shijiazhuang Tiedao University, Shijiazhuang, China, in 2018. She is currently a Graduate Student with Nankai University, Tianjin, China. Her current research interests include active disturbance rejection control and reinforcement learning.
ZHAOYANG HUANG was born in 1995. He received the B.Sc. degree in automation and the M.E. degree in control science and engineering from Nankai University, Tianjin, China, in 2018 and 2021, respectively. His research interests include intelligent algorithms and intelligent control.

JIN TAO (Member, IEEE) received the B.Sc. degree in automation from Qingdao University of Science and Technology, Qingdao, China, in 2008, the M.Sc. degree in control theory and control engineering from Guangxi University of Science and Technology, Liuzhou, China, in 2011, and the Ph.D. degree in control science and engineering from Nankai University, Tianjin, China, in 2017. He is currently an Associate Professor with the College of Artificial Intelligence, Nankai University. He is also with the Department of Electrical Engineering and Automation, Aalto University. He has published more than 50 peer-reviewed articles in international journals and conferences. His research interests include intelligent control, evolutionary optimization, and multi-agent systems.

HAO SUN received the B.Sc. degree in information security from Yunnan University, Yunnan, China, in 2013, the M.Sc. degree in mechatronic engineering from Tianjin University of Technology, Tianjin, China, in 2016, and the Ph.D. degree in control science and engineering from the College of Artificial Intelligence, Nankai University, Tianjin, in 2019. He is currently a Research Assistant with Nankai University. His current research interests include intelligent control, evolutionary optimization, dynamic modeling, and their application in paraglider systems.

QINGLIN SUN received the B.E. and M.E. degrees in control theory and control engineering from Tianjin University, Tianjin, China, in 1985 and 1990, respectively, and the Ph.D. degree in control science and engineering from Nankai University, Tianjin, in 2003. He is currently a Professor with the Intelligence Predictive Adaptive Control Laboratory, Nankai University, and the Associate Dean of the College of Artificial Intelligence. His research interests include self-adaptive control and the modeling and control of flexible spacecraft, and embedded control systems.

MATTHIAS DEHMER received the B.Sc. degree in mathematics from the University of Siegen, Germany, in 1997, and the Ph.D. degree in computer science from Darmstadt University of Technology, Germany, in 2005. He is currently a Professor with the Department of Biomedical Computer Science and Mechatronics, UMIT-The Health and Life Sciences University, Tirol, Austria. He has authored and coauthored more than 200 journal articles. His H-index is 28 and his i10-index is 78. His research interests include data science, cybersecurity, disaster management, complex networks, risk analysis, information systems, machine learning, information theory, bioinformatics, visual analytics, and computational statistics.

MINGWEI SUN received the Ph.D. degree in control theory and control engineering from the Department of Computer and Systems Science, Nankai University, Tianjin, China, in 2000. From 2000 to 2008, he was a Flight Control Engineer with Beijing Electro-Mechanical Engineering Research Institute, Beijing, China. Since 2009, he has been with Nankai University, where he is currently a Professor. His research interests include flight control, guidance, model predictive control, active disturbance rejection control, and nonlinear optimization.

ZENGQIANG CHEN (Member, IEEE) was born in 1964. He received the B.S., M.E., and Ph.D. degrees from Nankai University, in 1987, 1990, and 1997, respectively. He is currently a Professor of control theory and engineering with Nankai University and a Deputy Director of the Institute of Robotics and Information Automation. His current research interests include intelligent predictive control, chaotic systems and complex dynamic networks, and multi-agent system control.