Active Disturbance Rejection Control Based on Deep Reinforcement Learning of PMSM for More Electric Aircraft

Yicheng Wang, Shuhua Fang, *Senior Member, IEEE*, and Jianxiong Hu, *Student Member, IEEE*

***Abstract*—In this article, an active disturbance rejection controller (ADRC) based on deep reinforcement learning (DRL) algorithm is proposed to be used in the flux weakening control (FWC) system of motors for more electric aircraft. Artificial intelligence algorithm is introduced into ADRC motor control system for the first time, and DRL is designed as the automatic tuning for the parameters optimization of ADRC. The interface module scheme is proposed to realize the conversion between the relevant quantities of the control system and the DRL Agent according to the characteristics of ADRC. The parameters are optimized in the form of parameter modification, and a new DRL-ADRC control framework is proposed which can avoid being trapped into local optimum. The ADRC model designed for the speed loop of FWC system are first introduced. An interface module is subsequently built to enable DRL to interact with the FWC system automatically. DRL agent is trained to optimize the internal parameters of ADRC, which have the characteristics of large quantities, weak sensitivity and strong coupling. Deep deterministic policy gradient is used as the strategy of DRL, which can quickly determine the descent gradient and converge the multiobjective optimization problem. Simulation and comparison with classical heuristic algorithms and disturbance rejection methods are carried out to show the superiority of DRL. The feasibility and effectiveness of the proposed control method are verified by experiments on an aerospace motor for MEA.**

***Index Terms*—Active disturbance rejection control (ADRC), deep deterministic policy gradient (DDPG), deep reinforcement learning (DRL), flux weakening, more electric aircraft (MEA), parameter optimization.**

I. Introduction

MORE electric aircraft (MEA) is becoming popular due to its energy saving feature and more reliable operation characteristics [1], and permanent magnet synchronous motor (PMSM) is now widely used in MEA as it can provide higher power density and higher efficiency [2]. Most of the PMSMs for MEA use flux weakening control directly to expand speed range. Various flux weakening control schemes have been proposed [3], [4], [5], [6], and a feedforward flux weakening method proposed in [7] and [8] can turn the research of the electromagnetic working point of the flux weakening control into calculating the intersection of different curves, such as torque curves, voltage curves and so on, which makes the control result more accurate.

The motor for MEA needs fast response speed and strong disturbance rejection ability, especially the disturbance rejection ability, which is related to the safety of MEA. The advantage of the flux weakening control system using the feedforward method is its good accuracy and transient response compared with other flux weakening strategies. However, the method is more dependent on the accuracy of the parameters of the motor, which is easily affected by time and the material of the motor, etc. When the motor is disturbed either by the parameter changes of the motor itself or external disturbance, the torque T_{ref} will change at a certain extent, which will affect the stability of control. Hence, the speed fluctuates easily and the safety of MEA is affected [9], [10]. In order to overcome those disadvantages, some methods are proposed to reject the disturbance of flux weakening control such as combining flux weakening control with sliding mode control (SMC) [11] or model predictive control (MPC) [12]. SMC usually requires the large control gain and may exhibit chattering behavior [13]. MPC has provided a faster dynamic performance and has good ability for disturbance rejection, but it is sensitive to parameter variations [14], [15].

Because ADRC not only has simple structure, strong robustness and has better performance in high-noise environments, but also has a good suppression effect on both internal disturbance and external disturbance [16], this article chooses to introduce ADRC into FWC system as the speed loop to obtain better control effect and intuitively make up for this shortcoming of feedforward flux weakening. Han proposes an active disturbance rejection controller (ADRC), which has strong error attenuation ability and disturbance rejection ability [16]. ADRC relies on the observation of interference and does not depend on the parameters of the controlled objective model. In [17], an ADRC composed of the linear extended state observer (ESO) is proposed that can make the calculation process simpler and the calculation time shorter. In [18], a scheme is proposed to improve the ADRC by designing an infinite-dimensional disturbance estimator to carry on the online disturbance estimation. An adaptive linear ADRC is designed by integrating a linear full-order ESO with

Manuscript received 12 April 2022; revised 29 July 2022; accepted 8 September 2022. Date of publication 19 September 2022; date of current version 10 October 2022. This work was supported in part by the Aeronautical Science Foundation of China under Grant 20160769002 and in part by the Six Talent Peaks Project of Jiangsu Province under Grant GDZB-103. Recommended for publication by Associate Editor K. Akatsu. *(Corresponding author: Shuhua Fang.)*

The authors are with the School of Electrical Engineering, Southeast University, Nanjing 210096, China (e-mail: 230218324@seu.edu.cn; shfang@seu.du.cn; 220182640@seu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TPEL.2022.3206089.

Digital Object Identifier 10.1109/TPEL.2022.3206089

a parallel controller [19]. The above two methods enhance the disturbance rejection ability of ADRC by improving the observer. ADRC has also been used in many different fields, such as exoskeleton joints [20], the power plant process with considerable time-delay [21], and wind energy conversion system [22], etc., which all achieve good control effects. However, the disadvantage of ADRC is that it has many internal parameters, the sensitivity of the parameters is not strong, and there is a coupling relationship between parameters, which means that it is difficult to select and debug the optimal parameters. The method of scaling and bandwidth-parameterization is a classical parameters tuning method [23], which cannot guarantee the best solution. At the same time, the formula of nonlinear ADRC contains nonlinear functions, which means its transfer function cannot be written, its bandwidth cannot be calculated, and the method has its limitations. Trial-and-error (TAE) tune method with experience is used to design ADRC [24]. These processes are time-consuming and cannot achieve the best control effect. Therefore, it is necessary to study the parameters to make ADRC obtain satisfactory results.

The heuristic algorithm represented by ant colony optimization (ACO) algorithm [25], genetic algorithms (GAs) [26] and particle swarm optimization (PSO) algorithm [27] is a common method to solve the parameter optimization problem. In [28], an ADRC method which employs ACO is proposed and has a better robustness than the conventional ADRC. A hybrid algorithm tuning method which combines radial basis function neural network with PSO and bacterial foraging optimization algorithm is proposed to improve the control effect [29]. However, the traditional heuristic algorithms are often trapped into local optimum while dealing with complex and practical design problems because of the local precise search and the fast convergence.

With the development of computer science, recognition methods based on machine learning (ML) are gradually being applied. ML can construct direct relationships between observations and parameters to make the effect better, and avoid the disadvantages of the heuristic algorithms. Deep reinforcement learning (DRL) as one of ML has been gradually used in some simple controller theory, such as parameter optimization of proportional integration differential controller [30] and MPC [31]. In [32], reinforcement learning is directly used in the control system of PMSM by compensate the output of PI controller after learning and training. The above applications have achieved good control results in the aspect of simulation under ideal conditions or at the level of theories, but have not been applied into the complex and strongly coupled motor control application environment to optimize the controller parameters. Therefore, DRL can be applied to the parameter optimization of ADRC of flux-weakening control system for MEA motor to benefit the control effect of this multiparameters and strong-coupling system.

This article proposes a novel ADRC control method based on DRL which is used in the flux weakening control system of an aerospace motor for MEA. The model of ADRC controller is built according to the theoretical analysis and is used as the speed loop of the flux weakening control system in Section II. A DRL-based ADRC model parameter optimization framework is proposed in Section III-A, which can be applied to calculate the best parameters for complex models without labels. The deep deterministic policy gradient (DDPG) strategy is used to converge the multiobjective optimization problem in Section III-B. The model of DRL-ADRC control system is simulated, and experiment setup is built to verify the simulation results in Sections IV and V.

II. Design of the ADRC

ADRC uses tracking differentiator (TD) to track the reference input of the system and arranges the transition process. ESO is used to observe the generalized output and state. Nonlinear state error feedback (NLSEF) is used to control the error of the system.

ADRC can be expressed as

$$\begin{cases} x^{(n)} = f(x, \dot{x}, \cdots, x^{(n-1)}, t) + \omega(t) + bu \\ y = x(t) \end{cases} \tag{1}$$

where $f(x, \ldots, x^{(n-1)}, t)$ is the control system, $\omega(t)$ is the disturbance, $x(t)$ is the output that can be measured, u is the control effect of the system, and b is the control effect gain. The following sets the three main components of ADRC in combination with (2) [16].

A. Tracking Differentiator

$$\begin{cases} \dot{z}_1 = z_2 \\ \cdots\cdots \\ \dot{z}_{n+1} = z_n \\ \dot{z}_n = f(z_1, z_2, \cdots, z_n) \end{cases}. \tag{2}$$

If the solution of system (2) satisfies $Z_i(t) \to 0 (t \to \infty)$, $i = 12, \ldots n$, then for any bounded function $\nu(t)$ and any constant $T>0$, the solution $x_1(t)$ of the new system (3) satisfies

$$\begin{cases} \dot{x}_1 = x_2 \\ \cdots\cdots \\ \dot{x}_{n-1} = x_2 \\ \dot{x}_n = R^n f(x_1 - \nu, \frac{x_2}{R}, \cdots, \frac{x_n}{R^{n-1}}) \end{cases} \tag{3}$$

$$\lim_{R\to\infty} \int_0^T |x_1(t) - \nu(t)|\, dt = 0 \tag{4}$$

$x_1{}'(t) = x_2(t)$ and the derivative of $x_1(t)$ can be regarded as the derivative of the input signal. Hence, TD can be used to replace actual system differentiation.

B. Extended State Observer

ESO is used to observe the output and its derivatives. The disturbance of ADRC is divided into two aspects. One is the uncertainty of the system itself, which is the internal disturbance of ADRC. The other is external disturbances. Both are collectively referred to as total disturbances, which is observed and estimated by ESO. For the uncertain system in (1), $x^{(n)}(t)$ is extended into the state variables to obtain new state variables as $x(t), \ldots, x^{(n-1)}(t), x^{(n)}(t)$. The system can be written as

$$\begin{cases} \dot{x}_1(t) = x_2(t) \\ \cdots\cdots \\ \dot{x}_n(t) = x_{n+1}(t) \\ \dot{x}_{n+1}(t) = f(x_1, x_2, \ldots x_{n+1}, t) + bu \\ y = x_1 \end{cases}. \tag{5}$$

Let the observer error $\varepsilon = z_1\text{-}x(t)$, select appropriate coefficient β, and the observer system can be written as

$$\begin{cases} \dot{z}_1 = z_2 - \beta\varepsilon \\ \cdots\cdots \\ \dot{z}_n = f(z_1, z_2, \ldots, t) - \beta_n\varepsilon + bu \end{cases}. \tag{6}$$

Equation (5) needs to use $f(x_1, x_2, \ldots, x_{n+1}, t)$ in calculation, but the function is often unknown in many cases. At this time, it is necessary to treat $f(x_1, x_2, \ldots, x_{n+1}, t)$ as a disturbance and use a nonlinear function to suppress the disturbance as follows:

$$\mathrm{fal}(x, a, \delta) = \begin{cases} \frac{x}{\delta^{1-a}}, |x| \le \delta \\ \mathrm{sign}(x), |x| > \delta \end{cases} \tag{7}$$

where fal() is a nonlinear function, sign(x) is a symbolic function, when $x \geq 0$, sign $= 1$, and when $x<0$, sign $= 0$, a is the nonlinear factor, and δ is the filter factor.

Let $x_{n+2} = f(x_1, x_2, \ldots, x_{n+1}, t)$, and $\dot{x}_{n+2} = g(x_1, x_2, \ldots, x_{n+1}, t)$, it can be seen that $g(x_1, x_2, \ldots, x_{n+1}, t)$ is also unknown. Equations (5) and (6) can be rewritten as

$$\begin{cases} \dot{x}_1(t) = x_2(t) \\ \cdots\cdots \\ \dot{x}_{n+1}(t) = x_{n+1}(t) + bu \\ \dot{x}_{n+2}(t) = g(x_1, x_2, \ldots, x_{n+1}, t) \\ y = x_1 \end{cases} \tag{8}$$

$$\begin{cases} \varepsilon_1 = z_1 - y \\ \dot{z}_1 = z_2 - \beta_1\varepsilon_1 \\ \cdots\cdots \\ \dot{z}_{n+1} = z_{n+2} - \beta_{n+1}\mathrm{fal}(\varepsilon_1, \alpha_n, \delta) + bu \\ \dot{x}_{n+2}(t) = -\beta_{n+2}\mathrm{fal}(\varepsilon_1, \alpha_{n+1}, \delta) \\ y = x_1 \end{cases}. \tag{9}$$

The extended state z_{n+2} makes a good estimation of the "real-time action" $a(t) = x_{n+2} = f(x_1, x_2, \ldots, x_{n+1}, t)$ of the unknown disturbance. At this time, (9) is the ESO of (5).

If $f(x_1, x_2, \ldots, x_{n+1}, t) = f_0(x_1, x_2, \ldots, x_{n+1}) + f_1(x_1, x_2, \ldots, x_{n+1}, t)$ and f_0 is known, x_{n+2} can be set as $f_1(x_1, x_2, \ldots, x_{n+1}, t)$, and the observer can be written as

$$\begin{cases} \varepsilon_1 = z_1 - y \\ \dot{z}_1 = z_2 - \beta_1\varepsilon_1 \\ \cdots\cdots \\ \dot{z}_{n+1} = z_{n+2} - \beta_{n+1}\mathrm{fal}(\varepsilon_1, \alpha_n, \delta) \\ \quad + f_0(x_1, x_2, \ldots, x_{n+1}) + bu \\ \dot{x}_{n+2}(t) = -\beta_{n+2}\mathrm{fal}(\varepsilon_1, \alpha_{n+1}, \delta) \\ y = x_1 \end{cases}. \tag{10}$$

In this case, $f(x_1, x_2, \ldots, x_{n+1}, t)$ is the total disturbance of internal and external disturbances, and the extended state z_{n+2} is the estimation of the $a(t) = f_1(x_1, x_2, \ldots, x_{n+1}, t)$,which is the unknown part of the total disturbance.

It can be seen that even if the system function $f(x, \ldots, x_n, t)$ is not clear, the total disturbance can still be estimated. The disturbance of the unclear objective whose parameters may change can be observed.

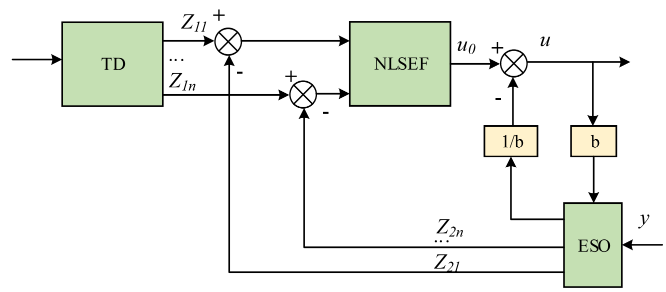

Fig. 1. Structure of ADRC.

C. Nonlinear State Error Feedback

NLSEF controls the combination of the reference input and the error signal of the extended state obtained by TD and ESO, and can compensate the total disturbance by nonlinear function (7)

$$NC\begin{cases} u_0 = k_1\mathrm{fal}(e_{31}, a_{31}, \delta_{31}) + \cdots k_n\mathrm{fal}(e_{3n}, a_{3n}, \delta_{3n}) \\ u_3 = u_0 - \frac{1}{b}z_{2(n+1)} \end{cases} \tag{11}$$

where $k_1 \ldots, k_n$ are constants, $e_{31} \ldots e_{3n}$,which means $e_{3x} = e_{1x}\text{-}e_{2x}$,$(x = 1 \ldots, n)$, represent the state error of each order.

D. Overall Construction of ADRC

Fig. 1 shows the overall construction of ADRC, where v is the reference input, TD arranges the transition process and gives the reference input and its derivative signal $Z_{11}, \ldots, Z_n$, ESO gives the extended state $Z_{21} \ldots Z_{2(n+1)}$ of the control objective after analyzing the output y of the whole system. The total disturbance $Z_{2(n+1)}$ (or defined as $a(t)$) is used to calculate current compensation, and the compensation coefficient is $1/b$.

The order of ADRC is affected by the order of the control system. Because of the input of flux weakening control is T_e, whose jitter will cause the jitter of output current, ADRC is used to replace the speed loop to provide a stable input to the flux weakening system. The ADRC of speed loop is one-order and the model is expressed as

$$\mathrm{TD}\begin{cases} e_1 = z_{11} - \omega^* \\ \dot{z}_{11} = -r\mathrm{fal}(e_1, a_1, \delta_1) \end{cases} \tag{12}$$

$$\mathrm{ESO}\begin{cases} e_2 = z_{21} - \omega \\ \dot{z}_{11} = z_{22} - \beta_{21}\mathrm{fal}(e_2, a_{21}, \delta_{21}) + bu \\ \dot{z}_{22} = -\beta_{22}\mathrm{fal}(e_2, a_{22}, \delta_{22}) \end{cases} \tag{13}$$

$$\mathrm{NLSEF}\begin{cases} e = z_{21} - z_{22} \\ u_0 = \beta_3\mathrm{fal}(e, a_3, \delta_3) \\ u = u_0 - z_{22}/b \end{cases} \tag{14}$$

where r, β_{21}, β_{22}, and β_3 are constants.

Since the traditional ADRC is directly connected with the current loop, i_q is the output u. The value of b can be calculated by the motion equation

$$\frac{d\omega}{dt} = \frac{1}{J}T_e - \frac{1}{J}T_L - \frac{1}{J}B\omega_m$$
$$= \frac{1.5p(\psi_f + (L_d - L_q)i_d)}{J}i_q - \frac{1}{J}T_L - \frac{1}{J}B\omega_m$$

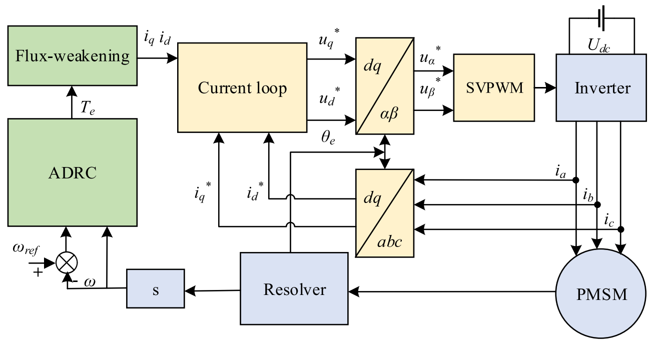

Fig. 2. Structure of flux weakening control system based on ADRC.

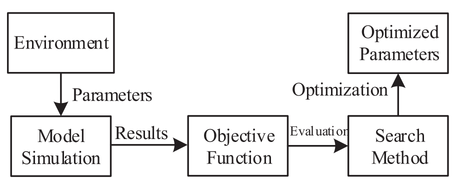

Fig. 3. Traditional parameters optimization method.

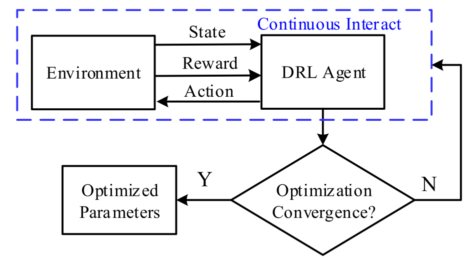

Fig. 4. DRL parameters optimization method.

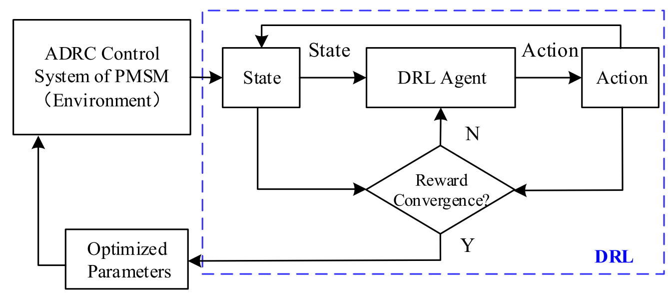

Fig. 5. DRL parameters optimization of ADRC for PMSM control system.

$$= bi_q - \frac{1}{J}T_L - \frac{1}{J}B\omega_m. \tag{15}$$

The value of b in this article is different from that of traditional ADRC. When the combination of ADRC and flux weakening control is used, the output of ADRC is torque T_e and b is different according to the rewritten formula (15)

$$\frac{d\omega}{dt} = bT_e - \frac{1}{J}T_L - \frac{1}{J}B\omega_m. \tag{16}$$

Subtracting (16) from the actual observations, one can have

$$\dot{e}_{\omega-\omega_o} = d\omega/dt - d\omega_o/dt = b(T_e - T_{eo}) - \frac{B}{J}e_{\omega-\omega_o} \tag{17}$$

where ω_o and i_{qo} are the observed value of ω and i_q, respectively. Since the output u is bT_e, b can be calculated as $1/J$. Fig. 2 shows the structure of flux weakening control system based on ADRC.

III. DRL-Based Parameters Optimization of ADRC

There are eleven parameters of the ADRC need to be optimized which are as shown in

$$\begin{cases} \text{TD} : r, a_1, \delta_1 \\ \text{ESO} : \beta_{21}, a_{21}, \delta_{21}, \beta_{22}, a_{22}, \delta_{22} \\ \text{NLSEF} : \beta_3, a_3, \delta_3 \end{cases}. \tag{18}$$

A. Optimization Framework Based on DRL of ADRC

Fig. 3 shows the traditional optimization method. Different outputs are obtained by entering different parameters and the results are returned to the optimized search algorithm. The results will be evaluated and the optimization algorithm will select the best group of parameters as the result.

As shown in Fig. 4, compared with the traditional method, the DRL algorithm has no intermediate links. DRL makes a decision called action A based on the state S in the current environment. The reward mechanism will calculate the reward R according to the effect of action A in the environment. The above behaviors will be fed back to a DRL agent in the algorithm, and the agent will continuously interact with the behaviors to learn and adjust to form an autonomous decision-making algorithm finally.

It can be seen that the structure of DRL is similar to the traditional parameter optimization by comparing Figs. 3 and 4. Both of them are used to evaluate the state when using different parameters, and optimize the final parameters according to the evaluation results. However, they have some fundamental differences as follows.

1) The purpose of traditional methods is to use optimization algorithms to find the optimal parameters by given data. The DRL-based parameter optimization method is to explore and learn the parameter setting strategy through its own interaction with the model.
2) The traditional method finally gives specific parameters for one case, while DRL gives a specific plan, which can quickly calculate optimal parameters of ADRC even if the optimization objective or situation is changed.
3) The traditional method is equivalent to optimize each parameter without coupling, and the independent parameters only affect themselves. But each "action" taken by DRL will affect the subsequent overall decision-making, which is more suitable for coupling problems.

Fig. 5 shows the structural block diagram of DRL-ADRC control system. DRL-ADRC takes the entire motor control system as the "environment" and the motor control effect as the "reward" evaluation standard. Based on the current state, the DRL agent makes a judgment and selects the action to be performed next, and state will update itself after taking the action. Reward evaluates the optimization effect according to the state and influence after the Action. When the evaluation

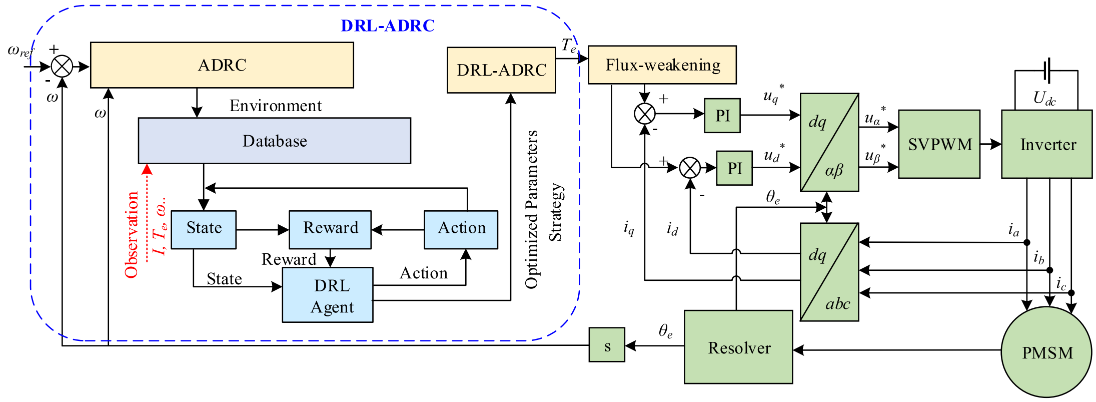

Fig. 6. ADRC control system based on DRL of PMSM for MEA.

value converges, the optimization parameters will be obtained. Otherwise, DRL agent will continue to interact until reward converges. The agent is trained to make the decisions about how to optimize the parameters by itself. The whole flux weakening system based on DRL-ADRC as shown in Fig. 6 is used as the scenarios to train the DRL agent and optimize the ADRC parameters. In this article, the control system of the motor is set as environment, and the speed curve of the motor is set as state. After the initial parameters are randomly given, each correction of the parameters values is set as action. The agent interacts with Environment after taking actions, then collects states, calculates evaluations reward, completes this round of learning and repeats the process until convergence.

B. DDPG-Based DRL

Parameters are usually distributed in the continuous space. Therefore, this article selects but do not specifies the DDPG, a classical actor-critic structure algorithm which is always applied to solve continuous action space problems, as the DRL algorithm to implement the parameters optimization of ADRC. DDPG uses three parts to achieve the goal of convergence.

1) Actor, which can learn and build a policy network and select different action in different state according to the network.
2) Critic, which can evaluate the value of actions to optimize the policy network.
3) Noise, which can simulate the disturbance of the system to make the result more accurate.

The deterministic policy gradient method is used to make the critic converge and update the parameters of the network. Fig. 7 shows the structure of DDPG.

The actor policy network is expressed as $\mu(\mathrm{s}|\theta^{\mu})$ in DDPG, and θ^{μ} is the internal parameters of the policy network μ. The current state and action are expressed as s_t and a_t, respectively. Agent takes a_t according to $\mu(\mathrm{s}|\theta^{\mu})$ based on s_t, and some noise will also be considered as shown in

$$a_t = \mu\left(s_t|\theta^{\mu}\right) + \mathrm{Noise}. \tag{19}$$

The reward r_t and the next state s_{t+1} are fed back when an action has been done, and these data (s_t, a_t, r_t, s_{t+1}) will be

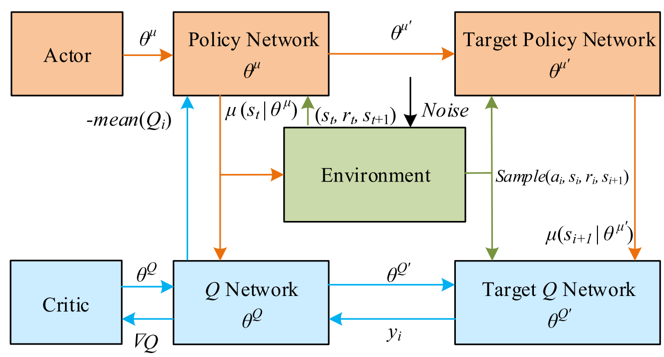

Fig. 7. Structure of DDPG.

stored in the database. $\{(s_i, a_i, r_i, s_{i+1}) \,|\, i = 12, \ldots, N\}$, which includes some data sets selected randomly from the database, is used to train the DDPG.

Subsequently, the evaluation network of critic, which is represented by $Q(s,a|\theta^{Q})$, will give the evaluation according to s and a of the previous step. The self-optimization of the algorithm is achieved by updating θ^{μ} and θ^{Q}. As shown in Fig. 7, in order to make the algorithm have enough time to learn, $\mu(\mathrm{s}|\theta^{\mu})$ and $Q(s,a|\theta^{Q})$ are not used directly, and the process of parameters update is split and enlarged by using $\mu'(\mathrm{s}|\theta^{\mu'})$ and $Q'(s, \mu'(\mathrm{s}|\theta^{\mu'})|\theta^{Q'})$, which are called the of target network of policy and critic respectively. The target networks have the same structure as the policy network and Q network, and their parameters are slowly tracked by the learned network with a parameter update rate $\tau \leq 1$

$$\begin{cases} \theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\,\theta^{Q'} \\ \theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\,\theta^{\mu'} \end{cases}. \tag{20}$$

The evaluation value y_i of the target Q network and the loss L between the target network and the original one can be calculated according to (21). The negative average value J of the Critic network in (22) is used as the optimization target to update θ^{μ} to maximize the evaluation value of Q. The loss L is the difference between the values of the target network and the actual network. The smallest L is used as the optimization target so Agent can find the gradient that makes L reduce continuously to optimize

Q network parameters θ^Q

$$\begin{cases} y_i = r_i + \gamma Q'\left(s_{i+1}, \mu'\left(s_{i+1}|\theta^{\mu'}\right)|\theta^{Q'}\right) \\ L = \frac{1}{N}\sum_i^N \left(\left(y_i - Q\left(s_i, a_i|\theta^Q\right)\right)^2\right) \end{cases} \tag{21}$$

$$J = -\frac{1}{N}\sum_{i=1}^{N}\left(-Q\left(s_i, a_i|\,\theta^Q\right)\right). \tag{22}$$

In this article, there are 11 parameters of the ADRC system which need to be defined. The action setting allows the algorithm to correct the parameters of ADRC which need to be optimized, and uses (23) to realize the normalization, restoration and correction of the parameters. Equation (24) is used to make the optimized parameters feasible

$$\begin{cases} \tilde{\theta}^i = (\theta^i - \theta^i_{\min})/\theta^i_{\max} \\ \theta^i = \theta^i_{\max}\tilde{\theta}^i + \theta^i_{\min} \\ \tilde{\theta}^i_{k+1} = \tilde{\theta}^i_k + a^i \end{cases} \tag{23}$$

$$\tilde{\theta}^i = \begin{cases} 0.05\left(\tilde{\theta}^i + 1\right)\Big/1.05 & \tilde{\theta} < 0.05 \\ \tilde{\theta}^i & 0.05 \leq \tilde{\theta} \leq 0.95 \\ 1 - 0.05\left(2 - \tilde{\theta}^i\right)\Big/1.05 & 0.95 < \tilde{\theta} \end{cases} \tag{24}$$

where $\theta_{\max}$, $\theta_{\min}$, and θ^i,$\tilde{\theta}^i$ are the upper limit, lower limit, original and normalized value of the *i*th parameter, respectively.

Equation (25) is used to evaluate and process the error between the real value and the given value after setting the optimization objective. Equation (26) is used to penalize and reward the correction of parameters, and (27) is used as the final evaluation of the optimization

$$R_{\text{Obs}} = \text{error evaluation} \tag{25}$$

$$\begin{cases} R^i_\theta = \begin{cases} 1 - \left(\tilde{\theta}^i + 1\right)\Big/1.05 & \tilde{\theta} < 0.05 \\ 0 & 0.05 \leq \tilde{\theta} \leq 0.95 \\ 1 - \left(2 - \tilde{\theta}^i\right)\Big/1.05 & 0.95 < \tilde{\theta} \end{cases} \\ R_\theta = \frac{1}{n}\sum_{i=1}^{n} R^i_\theta \end{cases} \tag{26}$$

$$R = -\left(\alpha_{\text{Obs}}R_{\text{Obs}} + \alpha_\theta R_\theta\right) \tag{27}$$

where α_{Obs} and α_θ are the corresponding weights of observation reward and parameter reward, respectively.

IV. Simulation of Control System

A. Multiobjective Optimization Setting

A simulation model of the control system is built according to the parameters of the motor in Table I. A complex simulated operation condition is used to verify the correctness of the proposed method in most possible environments. In order to simulate the disturbance environment that the MEA motor may be faced with [33], [34] and further strengthen the disturbance rejection ability of the motor, the initial speed of the motor is set to 1500 r/min and will start with a starting periodic torque of $|\sin(\pi t)|$ N·m. The torque will suddenly change to a constant load of 2 N·m at 3s, and the speed will rise to 3500 r/min at 6 s. In order to improve the disturbance rejection ability and response speed of the motor, a multiobjective optimization function is proposed

TABLE I
Main Parameters of PMSM

Symbol	Item	Value
R_s	phase resistance	0.0713Ω
L_q	*q*-axis inductance	605uH
L_d	*d*-axis inductance	519.5uH
Ψ_f	rotor flux	0.0201Wb
J	rotational inertia	$4.07\text{x}10^{-4}$kg.m^2
p_n	pole pairs	5

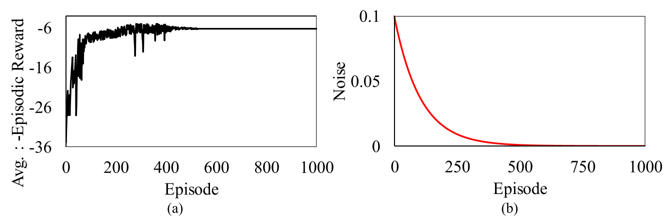

Fig. 8. Convergence curve of DRL-ADRC. (a) Convergence curve under noise. (b) Noise curve.

$$R = r_1 e_{os}/s_1 + r_2 t_{rs}/s_2 + r_3 t_{rl}/s_3 + r_4|e_l|/s_4 \tag{28}$$

where e_{os}, e_l, t_{rs}, and t_{rl} are the optimized objectives. e_{os} and e_l are the speed error at start-up and sudden load transition, respectively. t_{rs} and t_{rl} are the starting time and the time when the speed returns to normal after sudden load, respectively. s_1, s_2, s_3, s_4 are standardization coefficients because of different dimensions between optimized objectives. r_1, r_2, r_3, r_4 are weight coefficients of the four optimized objectives, which can be changed according to different demands of the applied environment. Considering the safety performance requirements of aero motors, $[r_1\ r_2\ r_3\ r_4]$ are set to [11 1.52]. The best optimization results will be obtained when the final evaluation value *R* is the smallest.

B. Verification of the DRL-ADRC Control

In the simulation, all the parameters of the conventional ADRC controller as shown in (18) are tuned by the TAE method and experience while the parameters of DRL-ADRC are optimized automatically. The DRL calculation is implemented in TensorFlow v2.10 and carried out by a computer configured with Inter(R) Core(TM) Inter(R) Core i7-10700KF,32GB. The noise interference as shown in Fig. 8(b) is set to be large first and then small to prevent falling into local optimum and obtain best optimization results. The convergence process of DRL algorithm is the convergence process of ADRC parameters optimization. The convergence curves of DRL-ADRC and each parameter are as shown in the Figs. 8 and 9. The parameters of DRL-ADRC and ADRC are given in Tables II and III, respectively for a fair comparison.

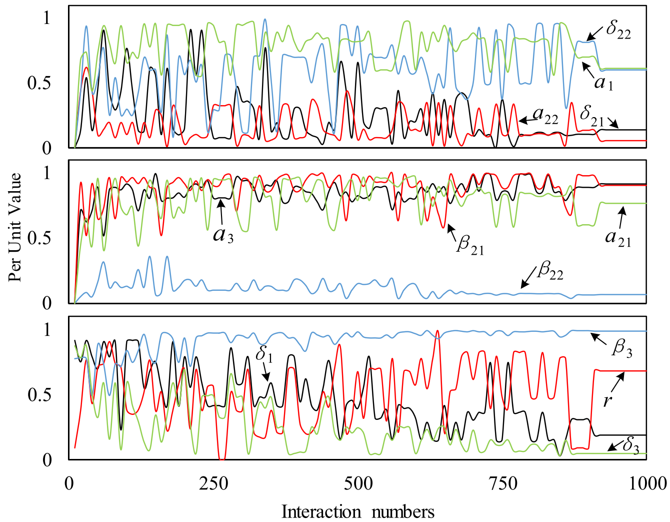

Fig. 9. Convergence curves of parameters of DRL-ADRC.

TABLE II
PARAMETERS OF DRL-ADRC

r	a_1	δ_1	β_{21}	a_{21}	δ_{21}
5.19467	0.61158	0.20654	59725.9	0.633108	0.10807
β_{22}	a_{22}	δ_{22}	β_3	a_3	δ_3
4116.49	0.664865	1.86799	2.99607	0.677487	0.198022

TABLE III
PARAMETERS OF ADRC

r	a_1	δ_1	β_{21}	a_{21}	δ_{21}
0.9	0.5	1.1	10000	0.5	0.01
β_{22}	a_{22}	δ_{22}	β_3	a_3	δ_3
10000	0.25	0.01	0.9	0.5	1.1

In order to show the change processes of parameters on the same platform more clearly, Fig. 9 shows the convergence change trend after unifying the parameters, in which the abscissa is the number of interactions between agent and environment, and the ordinate is the per unit value of each ADRC parameter. It can be seen that although each parameter is changing irregularly, the parameters have same time change point, which means that the DRL is learning the optimization without ignoring the coupling relationships between parameters. It proves that compared with the independent parameter convergence of the traditional optimization method, the convergence of DRL-ADRC is systematic and coupled after exploration, which can better meet the requirements of multiobjective strong coupling system. In the process of DRL parameters optimization, the setting noise changes from large to zero, which can better prevent local optimization, and also makes the process of convergence jitter at first. The parameters in Tables II and III are used for simulation respectively, and the results are as shown in Fig. 10.

It can be seen from Fig. 10 that the parameters obtained by trial-and-error method and experience have been applied to the ADRC controller and achieved good control effect. The speed curve has almost no overshoot, and the response speed is less than 0.15s. However, when the parameters optimized by DRL are applied to ADRC, the response time of the motor is only one third of that of traditional ADRC. When sudden load transitions are applied to the simulation model, DRL-ADRC also has smoother curve, faster recovery speed and less jitter. Thus, the effectiveness of DRL-ADRC can be verified.

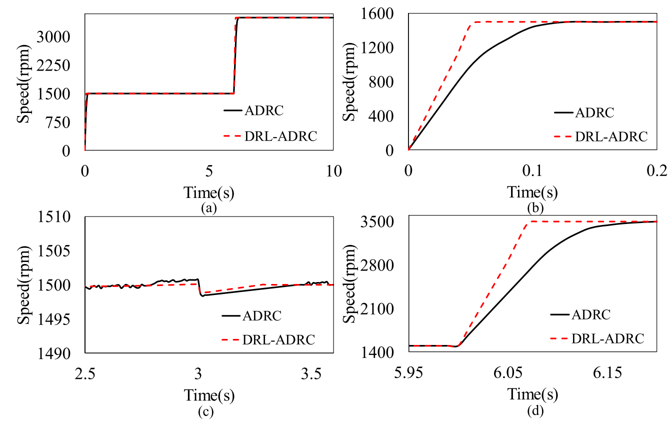

Fig. 10. Comparison of simulation results with normal ADRC. (a) Overview. (b) Start-up state. (c) Load transition state. (d) Speed transition state.

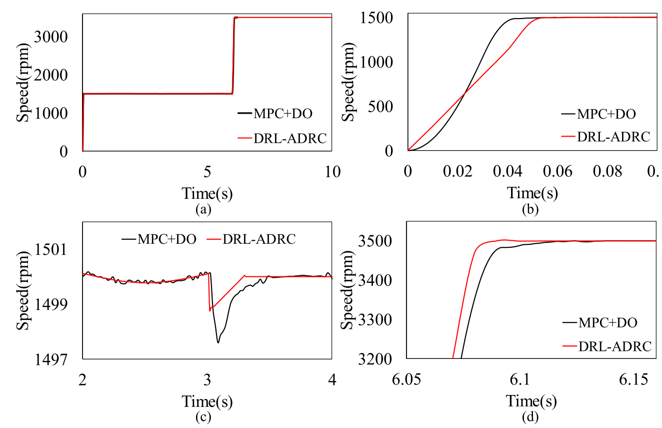

Fig. 11. Comparison of simulation results with MPC+DO. (a) Overview. (b) Start-up state. (c) Load transition state. (d) Speed transition state.

C. Comparison With Disturbance Rejection Algorithm

In order to verify the superiority of DRL-ADRC, a disturbance rejection algorithms combining MPC with linear disturbance observer (MPC+DO) based on ESO [35], are used to compare with DRL-ADRC. Fig. 11 shows the simulation results.

It can be seen that MPC performs better in the start-up phase, and the curves show its fast response ability and good disturbance rejection ability, but there is a slight jitter in the speed when facing continuously changing disturbances. DRL-ADRC has a better performance in dealing with sudden changes both in speed and torque.

D. Comparison With Other Classical Optimization Algorithm

In order to further verify the superiority of DRL-ADRC, classical heuristic algorithms, GA, PSO, and DE, are applied to

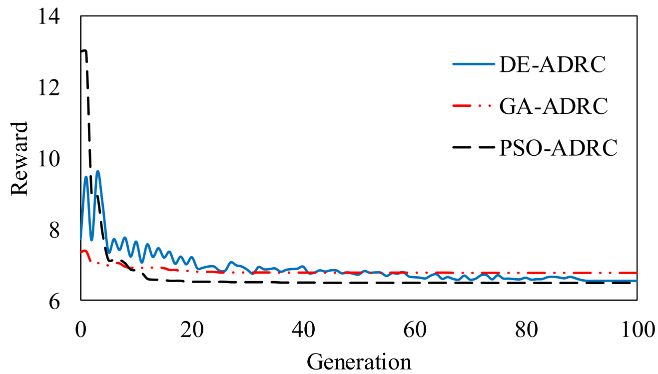

Fig. 12. Convergence curves of DE, GA, and PSO.

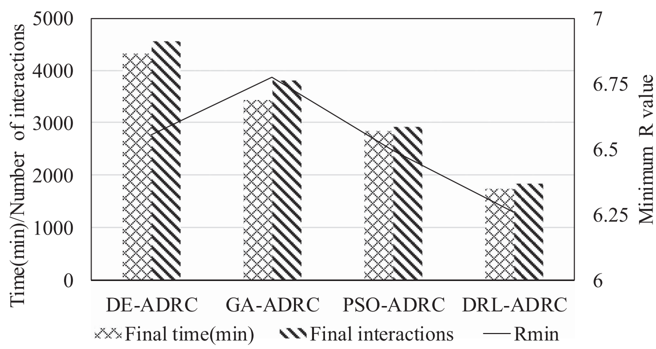

Fig. 13. Comparison between optimization algorithms.

the parameter optimization of ADRC, and the simulation results are compared as shown in Figs. 12 and 13.

It can be seen from Fig. 12 that the three algorithms have converged, and GA has fastest speed of convergence but the R value is the largest while PSO performs worst at first but obtains the best results finally. In Fig. 13, Final time is the convergence time, final interaction is the number of interactions with the model, and $R_{\min}$ is the minimum R in all times. From an algorithm perspective, Fig. 13 shows the comparison of the four algorithms that DRL uses the least time and interaction times but obtain the best effect. Optimized parameters of GA, DE, and PSO are applied to ADRC respectively, and the simulation results are compared with DRL-ADRC.

It can be seen from Fig. 14 that the four methods have all achieved good optimization results, and the histogram is used to make the comparison of simulation results more obvious as shown in Fig. 15. The results of the four optimization methods are relatively close in the case of start-up and speed transition, but DRL-ADRC is more prominent in terms of disturbance rejection performance. The noise disturbance is added to the speed observation value to test the disturbance rejection ability of the algorithms. The mean value of the white noise of the added normal distribution is 0, the variance is 1, and the sampling time is 0.01s. It can be seen from Fig. 16 that when the noise is added to the ideal environment, the four all have good noise suppression ability and the suppression effect is close to each other. In conclusion, DRL has smaller overshoot, shorter time to return to normal, and smaller error after sudden torque transition,

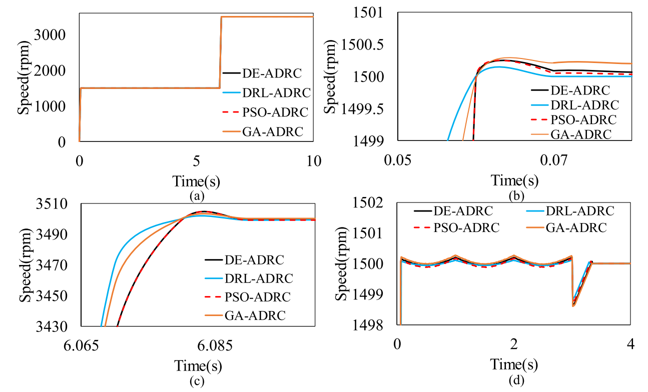

Fig. 14. Comparison of simulation results with optimization methods. (a) Overview. (b) Start-up state. (c) Speed transition state. (d) Load transition state.

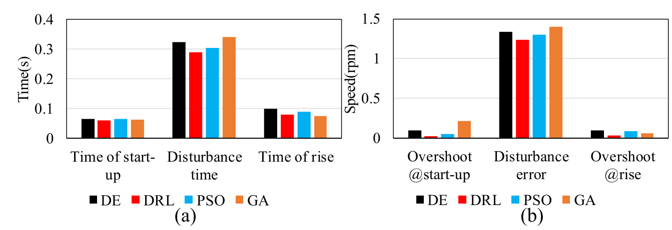

Fig. 15. Histogram of simulation results with optimization methods. (a) Comparison in time. (b) Comparison in error.

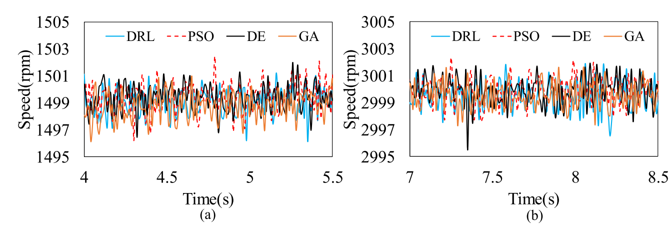

Fig. 16. Performances of adding white noise disturbance. (a) @1500 r/min. (b) @3000 r/min.

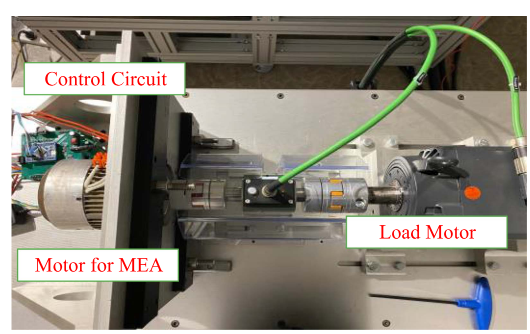

Fig. 17. Experimental platform.

which means the proposed method is a superior optimization method.

V. Experimental Verification

In order to verify the effectiveness of the proposed control method, an experimental platform is built as shown in Fig. 17.

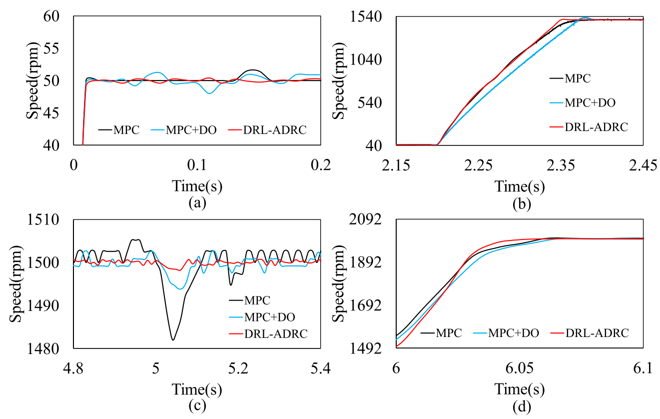

Fig. 18. Comparison of experimental results with disturbance rejection methods. (a) Overview. (b) Start-up state. (c) Speed transition state. (d) Load transition state.

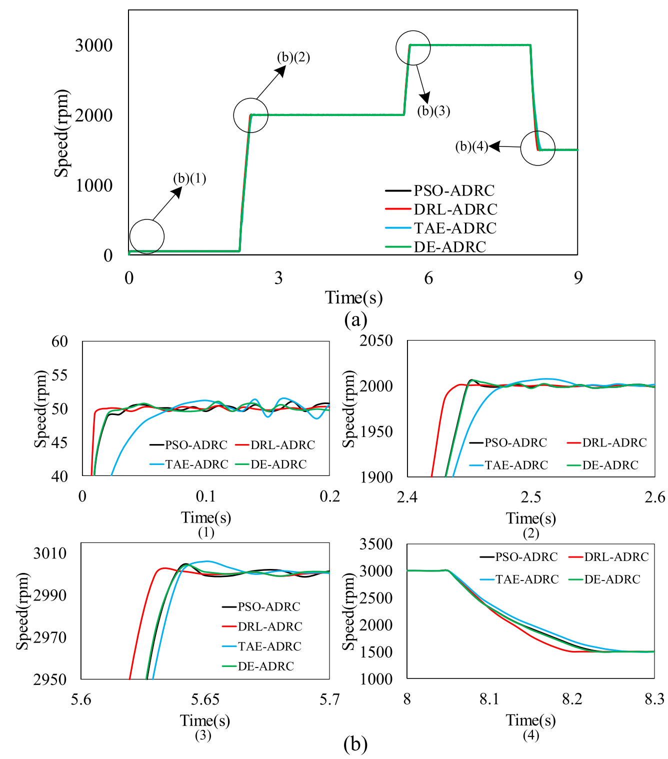

Fig. 19. Speed transition experiment. (a) Overview. (b) Comparison of details.

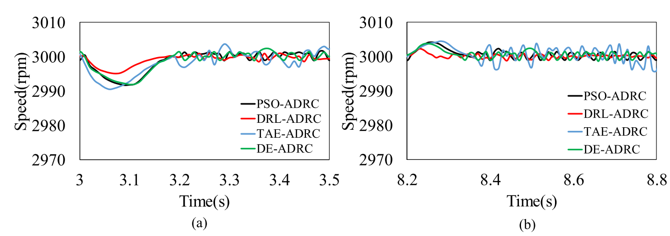

Fig. 20. Speed of Load transition experiment. (a) Load increase. (b) Load decrease.

The experimental platform used in this article is MEA RDS loading experimental platform. All the control methods are implemented in microcontroller unit of STM32F407 with a clock frequency of 168 MHz, and the sampling period of this article is 50 μs. The specific parameters of the motor are given in Table I and a 50V/9A current source is used for power supply. Different experiments are carried out to test the performance of the proposed method.

A. Comparison Experiment With Disturbance Rejection Methods

MPC and MPC+DO are used as disturbance rejection methods to compare with DRL-ADRC. The speed is set to 50 r/min without load to show the ability of start-up under low speed. The speed increases to 1500 r/min, the torque suddenly changes to a constant load of 1 N·m, and the speed rises to 2000 r/min at last.

It can be seen from Fig. 18 that MPC which depends on the accuracy of parameters has fast response speed, but poor disturbance rejection ability, and there are obvious jitters in the steady state. After combining with a disturbance observer base on ESO to MPC, the disturbance rejection performance has been better improved, and the jitter has been reduced correspondingly, but the response speed has been decreased. DRL-ADRC has the best control effect due to the suppression of both internal and external disturbances.

B. Comparison Experiment of Speed Transition With Optimization Methods

Since PSO and DE are the two with the smallest R value in the heuristic algorithms, they are used in the experiment to compare with the optimized result of DRL-ADRC. The traditional TAE method is also applied to comparison. The speed is set to 50 r/min without load to show the ability of start-up under low speed. The speed is increased to 2000 and 3000 r/min, respectively, and then decreased to 1500 r/min. For fair comparison, the change time of the four methods is set at the same time. Fig. 19 shows the experimental results. It can be seen that the response speed and overshoot of TAE, which need a lot of time to debug, are good separately, but they are the worst when compared with the other three. TAE has a small gap with the other methods in overshoot, but the response speed is significantly slower. PSO and DE have similar speed control effects in the experiment. It can be seen from Fig. 19 that the DRL-ADRC has faster response speed and smaller overshoot no matter under accelerating or decelerating.

C. Comparison Experiment of Load Transition With Optimization Method

Sudden load transition experiment is carried out to test the disturbance rejection ability of the four methods. When the speed is maintained at 3000 r/min under no load, a load of 1 N·m is given to the motor, and suddenly subtracted after a period of time. The experimental results are as shown in Figs. 20 and 21.

It can be seen from Fig. 20 that the speed errors of DRL-ADRC are both smaller than PSO-ADRC no matter the load is increased or decreased suddenly, and the speed waveform is more stable. The control effect of DE and TAE is similar to that of PSO, but TAE is slightly worse. Fig. 21 shows that the

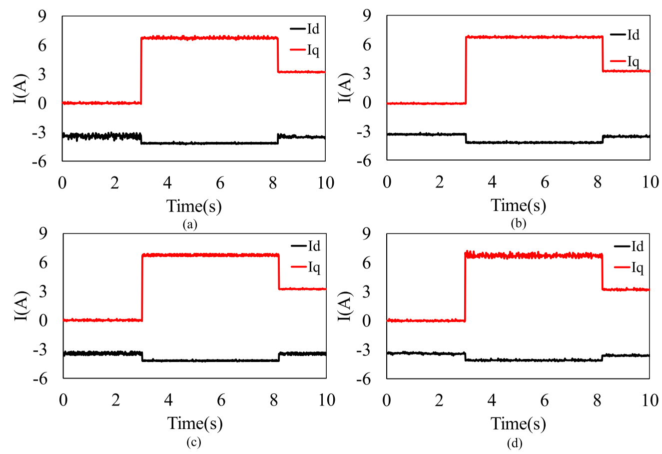

Fig. 21. Current of Load transition experiment. (a) PSO-ADRC. (b) DRL-ADRC. (c) DE-ADRC. (d) TAE-ADRC.

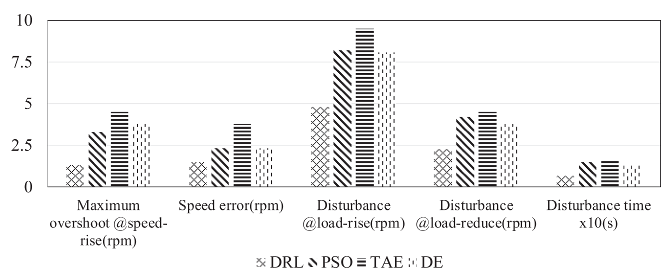

Fig. 22. Comparison between optimization methods.

dq-axis current of DE and PSO-ADRC has obvious jitter under no-load, and the jitter is smaller under load. The i_q of TAE shakes obviously. The current curve of DRL-ADRC is smoother, which can prove the stability of the proposed method.

Fig. 22 shows the advantages of DRL-ADRC more clearly. It can be seen that the recovery speed of TAE-ADRC in case of disturbance is close to that of the other three. The disturbance rejection ability of DE-ADRC is better than PSO-ADRC, but the response speed is slower than that of PSO-ADRC. Although PSO-ADRC has a relatively close control effect to DRL-ADRC during simulation, however, the difference is obvious in the experiment. ADRC optimized by DRL has better adaptability and control effect, which proves the effectiveness of DRL-ADRC in the experiments.

VI. Conclusion

The article proposes a novel DRL-ADRC method used in flux weakening control system for MEA motors. The novel method employs the DRL as the automatic tune mechanism for the parameters of ADRC, and the DDPG is used as the algorithm to make the optimization process converge. According to the setting of optimization objectives, DRL is trained to interact with the model and converge to the optimum parameters. The simulations which are compared with heuristic algorithms and disturbance rejection methods show that the DRL can avoid being trapped into the local optimum and premature convergence, and is more suitable for multiobjective strong coupling system. Experimental results prove that DRL-ADRC has smaller overshoot, higher work efficiency, faster response speed, and better disturbance rejection ability than that of other methods, which can better meet the requirement of MEA motors. The proposed method can be used in other applications besides the MEA motor.

References

[1] B. Sarlioglu and C. T. Morris, "More electric aircraft: Review, challenges, and opportunities for commercial transport aircraft," *IEEE Trans. Transp. Electrific.*, vol. 1, no. 1, pp. 54–64, Jun. 2015.

[2] M. Hirst et al., "Demonstrating the more electric engine: A step towards the power optimised aircraft," *IET Elect. Power Appl.*, vol. 5, no. 1, pp. 3–13, 2011.

[3] S. Sue and C. Pan, "Voltage-constraint-tracking-based field-weakening control of IPM synchronous motor drives," *IEEE Trans. Ind. Electron.*, vol. 55, no. 1, pp. 340–347, Jan. 2008.

[4] P. Vaclavek and P. Blaha, "Interior permanent magnet synchronous machine field weakening control strategy - the analytical solution," in *Proc. SICE Annu. Conf.*, Aug. 2008, pp. 753–757.

[5] F. Liu, L. Xu, R. Liu, and Y. Li, "Impedance and stability analysis of a permanent magnet synchronous generator system for more electric aircraft," in *Proc. IEEE Int. Conf. Electr. Syst. Aircr., Railway, Ship Propuls. Road Veh., Int. Transp. Electrific. Conf.*, 2018, pp. 1–6.

[6] S. Chi and L. Xu, "A special flux weakening control scheme of PMSM - incorporating and adaptive to wide-range speed regulation," in *Proc. Conf. Proc. CES/IEEE Int. Power Electron. Motion Control*, 2006, pp. 1–6.

[7] H. Eldeeb, C. M. Hackl, L. Horlbeck, and J. Kullick, "A unified theory for optimal feedforward torque control of anisotropic synchronous machines," *Int. J. Control*, vol. 91, no. 10, pp. 2273–2302, 2017.

[8] S. Wang, J. Kang, M. Degano, A. Galassini, and C. Gerada, "An accurate wide-speed range control method of IPMSM considering resistive voltage drop and magnetic saturation," *IEEE Trans. Ind. Electron.*, vol. 67, no. 4, pp. 2630–2641, Apr. 2020.

[9] W. Wang, "Design of flux weakening control system of PMSM based on the fuzzy self-tuning PID controller," in *Proc. Int. Conf. Consum. Electron., Commun. Netw.*, 2011, pp. 226–229.

[10] A. Wang, X. Jia, and L. Zhang, "A new flux weakening control strategy considering voltage saturation for IPMSM drives," in *Proc. Chin. Control Conf.*, Jul. 2012, pp. 4295–4299.

[11] X. Li, C. Liu, S. Wu, S. Chi, and P. C. Loh, "Sliding-Mode flux-weakening control with only single current regulator for permanent magnet synchronous motor," *IEEE Access*, vol. 7, pp. 131616–131626, 2019.

[12] Z. Zheng, D. Sun, M. Wang, and H. Nian, "A dual two-vector-based model predictive flux control with field-weakening operation for OW-PMSM drives," *IEEE Trans. Power Electron.*, vol. 36, no. 2, pp. 2191–2200, Feb. 2021.

[13] B. Guo and F. Jin, "Sliding mode and active disturbance rejection control to stabilization of one-dimensional anti-stable wave equations subject to disturbance in boundary input," *IEEE Trans. Autom. Control*, vol. 58, no. 5, pp. 1269–1274, May 2013.

[14] Z. Zhang, Z. Wang, X. Wei, Z. Liang, R. Kennel, and J. Rodriguez, "Space-vector-optimized predictive control for dual three-phase pmsm with quick current response," *IEEE Trans. Power Electron.*, vol. 37, no. 4, pp. 4453–4462, Apr. 2022.

[15] A. Aboelhassan, A. M. Diab, M. Galea, and S. Bozhko, "Investigating electrical drive performance employing model predictive control and active disturbance rejection control algorithms," in *Proc. 23rd Int. Conf. Electr. Mach. Syst.*, 2020, pp. 1379–1384.

[16] J. Han, "From PID to active disturbance rejection control," *IEEE Trans. Ind. Electron.*, vol. 56, no. 3, pp. 900–906, Mar. 2009.

[17] Z.-Q. Gao, S.-H. Hu, and F.-J. Jiang, "A novel motion control design approach based on active disturbance rejection," in *Proc. IEEE Conf. Decis. Control*, 2001, vol. 5, pp. 4877–4882.

[18] H. Feng and B. Guo, "A new active disturbance rejection control to output feedback stabilization for a one-dimensional anti-stable wave equation with disturbance," *IEEE Trans. Automat. Control*, vol. 62, no. 8, pp. 3774–3787, Aug. 2017.

[19] C. Liu, G. Luo, X. Duan, Z. Chen, Z. Zhang, and C. Qiu, "Adaptive LADRC-Based disturbance rejection method for electromechanical servo system," *IEEE Trans. Ind. Appl.*, vol. 56, no. 1, pp. 876–889, Jan./Feb. 2020.
[20] W.-Q. Xun, Z.-K. Yu, and D.-X. Dong, "An exoskeleton joint output force control technology based on improved ADRC," in *Proc. Int. Conf. Robot. Autom. Eng.*, Dec. 2017, pp. 146–150.
[21] L. Sun, W. Xue, D. Li, H. Zhu, and Z.-G. Su, "Quantitative tuning of active disturbance rejection controller for FOPTD model with application to power plant control," *IEEE Trans. Ind. Electron.*, vol. 69, no. 1, pp. 805–815, Jan. 2022.
[22] S. Das and B. Subudhi, "A two-degree-of-freedom internal model-based active disturbance rejection controller for a wind energy conversion system," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 8, no. 3, pp. 2664–2671, Sep. 2020.
[23] Z.-Q. Gao, "Scaling and bandwidth-parameterization based controller tuning," in *Proc. AmER. Control Conf.*, Jun. 2003, pp. 4989–4996.
[24] 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.
[25] H. Huang, "SoPC-based parallel ACO algorithm and its application to optimal motion controller design for intelligent omnidirectional mobile robots," *IEEE Trans. Ind. Inf.*, vol. 9, no. 4, pp. 1828–1835, Nov. 2013.
[26] Y. Gao, J. Wang, S. Gao, and Y. Cheng, "An integrated robust design and robust control strategy using the genetic algorithm," *IEEE Trans. Ind. Inf.*, vol. 17, no. 12, pp. 8378–8386, Dec. 2021.
[27] J. Yi, J. Bai, W. Zhou, H. He, and L. Yao, "Operating parameters optimization for the aluminum electrolysis process using an improved quantum-behaved particle swarm algorithm," *IEEE Trans. Ind. Inf.*, vol. 14, no. 8, pp. 3405–3415, Aug. 2018.
[28] Z. Yin, C. Du, J. Liu, X. Sun, and Y. Zhong, "Research on auto disturbance-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.
[29] 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.
[30] Y. Qin, W. Zhang, J. Shi, and J. Liu, "Improve PID controller through reinforcement learning," in *Proc. IEEE CSAA Guid., Navig. Control Conf.*, 2018, pp. 1–6.
[31] Y. Tange, S. Kiryu, and T. Matsui, "Model predictive control based on deep reinforcement learning method with discrete-valued input," in *Proc. IEEE Conf. Control Technol. Appl.*, 2019, pp. 308–313.
[32] M. Nicola and C. I. Nicola, "Improvement of PMSM control using reinforcement learning deep deterministic policy gradient agent," in *Proc. 21st Int. Symp. Power Electron.*, Oct. 2021, pp. 1–6.
[33] Q. Zhang, M. Wang, H. Li, and T. Cui, "Research on control characteristics of three-phase permanent magnet synchronous motor for electric aircraft propulsion," in *Proc. Chin. Control Conf.*, Jul. 2020, pp. 2676–2681.
[34] J. X. J. Bannwarth, Z. J. Chen, K. A. Stol, and B. A. MacDonald, "Disturbance accomodation control for wind rejection of a quadcopter," in *Proc. Int. Conf. Unmanned Aircr. Syst.*, Jun. 2016, pp. 695–701.
[35] J. Wang, D. Huang, S. Fang, Y. Wang, and W. Xu, "Model predictive control for arc motors using extended state observer and iterative learning methods," *IEEE Trans. Energy Conver.*, vol. 37, no. 3, pp. 2217–2226, Sep. 2022, doi: 10.1109/TEC.2022.3159834.

Yicheng Wang received the B. Eng. degree from Southwest Jiaotong University, Chengdu, China in 2018, and the M.S. degree in 2021 from Southeast University, Nanjing, China, where he is currently working toward the Ph.D. degree in electrical engineering.

His research interests include control strategies for permanent magnet machines and power electronics.

Shuhua Fang (Senior Member, IEEE) received the M.S. degree from Shandong University of Science and Technology, Jinan, China, in 2004 and the Ph.D. degree from Southeast University, Nanjing, China, in 2008, respectively, both in electrical engineering.

From 1998 to 2001, he was an Associate Engineer with the Xuzhou Coal and Mine Machinery Factory, where his research activities were primarily in the area of the design, analysis and control of electrical apparatus for coal and mine. Since 2008, he has been with Southeast University, where he is currently a Full Professor with the School of Electrical Engineering. From 2013 to 2014, he was a Visiting Professor with the Wisconsin Electric Machine and Power Electronics Consortium, University of Wisconsin-Madison. His research interests include design, simulation and control for both permanent magnet actuator and intelligent apparatus.

Jianxiong Hu (Student Member, IEEE) received the B. Eng. degree from Tianjin University, China, in 2018, and the M.S. degree in 2021 from Southeast University, Nanjing, China, where he is currently working toward the Ph.D. degree in electrical engineering.

His research interests include power system modelling, simulation, and power system stability analysis and control.