An Improved DDPG and Its Application Based on the Double-Layer BP Neural Network

MINGLI ZHANG[1,2], YIJIE ZHANG [1], ZHENGJIE GAO[3], AND XIAOLONG HE[3]
[1]School of Economics and Management, Yanshan University, Qinhuangdao 066004, China
[2]School of Graduate Studies, Hebei University of Chinese Medicine, Shijiazhuang 050200, China
[3]School of Mechanical Engineering, Yanshan University, Qinhuangdao 066004, China

Corresponding author: Yijie Zhang (shizi114@163.com)

ABSTRACT This paper focused on three application problems of the traditional Deep Deterministic Policy Gradient(DDPG) algorithm. That is, the agent exploration is insufficient, the neural network performance is unsatisfied, the agent output fluctuates greatly. In terms of agent exploration strategy, network training algorithm and overall algorithm implementation, an improved DDPG method based on double-layer BP neural network is proposed. This method introduces fuzzy algorithm and BFGS algorithm based on Armijo-Goldstein criterion, improves the exploration efficiency, learning efficiency and convergence of BP neural network, increases the number of layers of BP neural network to improve the fitting ability of the network, and adopts periodic update to ensure the stable operation of the algorithm. The experimental results show that the deep learning network based on the improved DDPG algorithm has greatly improved the performance compared with the traditional method after multiple rounds of self-learning under variable working conditions. This study lays a theoretical and experimental foundation for the extended application of deep learning algorithm.

INDEX TERMS Deep deterministic policy gradient (DDPG), BP neural network, fuzzy algorithm, improved DDPG.

I. INTRODUCTION

The overall advantages of the legged robot lies in that its has good motion performance in unstructured environment, strong terrain adaptability and load capacity, and have broad application prospects in complex working environments such as security patrol inspection, field transportation, fire rescue, and geological exploration [1]–[4]. Since the 21st century, a number of advanced quadruped bionic robots have been born in the field of robotics. BigDog, a four-legged robot developed by Boston Dynamics, is the most typical robot. With strong comprehensive performance, BigDog can easily walk in the ice, hills and jungle, and it is the industry benchmark for four-legged bionic robots [5]. Later, Boston Dynamics launched Cheetah [6], WildCat [7], LS3 [8] and other more advanced quadruped robots. Cheetah, as the fastest robot in the world, can run at an amazing speed of 44.8km/h. The above robots have their own application backgrounds, and although the active motion joints of their

The associate editor coordinating the review of this manuscript and approving it for publication was Ning Sun [iD].

legs are different, they are all developed based on the study of SLIP model. Contact angle is one of the key control parameters of SLIP model, so the position control plays a crucial role in whether the four-legged robot based on SLIP model can achieve the expected control effect.

The traditional classical position control algorithm is generally improved based on the proportional-integration-differential (PID) controller with constant parameter structure. Its implementation method is simple and its stable performance is widely used in engineering applications. But the control performance of control method based on the constant parameters PID controller is closely related to the control parameter, when the state of the control loop changes, servo systems often need to adjust the parameters, in order to satisfy the demands of high precision position control in all kinds of cases [9], readjust parameter means shutdown and debugging which will affect system efficiency, also waste time and energy. Researchers have proposed many solutions to solve the problems of variable working conditions and inertial disturbance in the application of hydraulic servo system, such as fuzzy logic, synovial membrane, adaptive

control and so on. Gyan Wrata *et al.* applied the PID controller set by the fuzzy controller to excavate electromechanical hydraulic actuators to improve the control accuracy, compensate the leakage of proportional flow control valve, and reduce the energy loss [10]. Wei Sun *et al.* designed a reduced adaptive fuzzy system together with a compensation term which can reduce possible chattering phenomena and achieve better control performance [11]. Triet Hung Ho *et al.* proposed an adaptive fuzzy slide control method, which reduced the chattering of tracking error and control effect [12]. Although these methods reduce the following error of the system and increase the adaptive ability of the system to a certain extent, they are not widely used due to the complex parameter debugging and slow convergence speed of the control system.

By combining reinforcement learning with DBN (Deep Belief Network), CNN (Convolutional Neural Network), RNN(Recurrent Neural Network), BP Neural Network and other deep neural networks, NatureDQN (improved Depth Q Network), DDPG, TRPO(Trust Region Policy Optimization), A3C(Asynchronous Advantage Actor-Critic), DPPO(Deep Proximal Policy Optimization) and other deep reinforcement learning algorithms have been developed, and remarkable achievements have been made in many fields. In the field of e-sports, the robot developed by OpenAI can defeat most human players in games such as DOTA2 and StarCraft [13]. In the field of autonomous driving, The Wayve company, in British, has successfully used the deep reinforcement learning method to make the intelligent body learn to drive a car in one day [14]. In the field of robot control, Swiss Federal Institute of Technology developed ANYMAL, a four-legged robot, and adopted deep reinforcement learning to enable the robot to learn to maintain body balance in various complex terrains [15]. Carlucho, Paula et al proposed an adaptive PID control algorithm based on incremental Q learning and applied it to mobile robots [16]. Runsheng Yu et al proposed an optimal tracking control method for underwater robot based on DDPG algorithm, which has a higher control accuracy than traditional PID algorithm [17].

DDPG is a Deep Deterministic Policy Gradient algorithm, it is a data-driven algorithm, which can self-learn the mathematical model of the controlled system according to the input and the output data of the system and realize the optimal control of the controlled system through system feedback [18]. DDPG algorithm is a combination of DQN and policy gradient algorithm. So it can update like DQN single step, but also has the high rate utilization of deterministic policy gradient data, the convergence of good advantages. It is suitable for the complex control system based on the continuous state and action Spaces, can realize effective information iteration and self-improve to achieve better control performance in the process of interaction with the system, in no model and prior knowledge of human cases, It is a true sense of the intelligent control method. BFGS method is a quasi-Newtonian algorithm [19], it is proposed to improve Newton's method, which needs to calculate Hessian matrix

with high computational resources. Constructing approximate Hessian to reduce the computation is the best algorithm to balance the convergence performance and computational resource requirements. Armijo-goldstein search criterion is a major criterion in the field of imprecise one-dimensional search [20]. Following these criteria can make the algorithm converge quickly (to find the optimal solution), make the step size of our imprecise one-dimensional search meet certain rules, and prevent the subsequent process of finding the optimal solution from failing to converge because the step size is too large or too small.

Hydraulic driven legged robot has the potential of high speed and high load crossing in complex environment due to its strong load capacity, fast response speed and high control accuracy. Hydraulic rive unit (HDU), as the leg ''muscle'' of that robot, needs to ensure the high-precision position control effect in complex working conditions to ensure the stable operation of the robot. As a hotspot of control research, deep reinforcement learning, if applied to the position control of HDU, will greatly improve the robot's adaptive ability to complex working conditions and ensure the robot's control accuracy.

The authors' cooperation team has carried out preliminary studies on the mathematical modeling and sensitivity analysis of HDU position control, which lays an application foundation for the research method in this paper [21]. In addition, related control methods based on constant parameter PID control were also studied in the previous study by the authors' cooperation team [22]. However, during the control process, problems such as shutdown to adjust parameters and unsatisfied adaptability of the control method in multiple working conditions are often encountered, so it is urgent to introduce machine learning related theories and combine them with the authors' preliminary research basis.

Based on the existing research, this paper conducts exploratory research on the intellectualization of HDU position control, and the main contributions are as follows: the learning performance and exploration depth of the agent are insufficient to solve the problem of control divergence which caused by large fluctuation of output control signal in HDU position control based on traditional DDPG deep learning algorithm. In this paper, an improved DDPG algorithm based on double-layer BP neural network is proposed and applied to HDU position control system. In terms of network structure, this method increases the number of layers of BP neural network to improve the fitting ability of the network. In terms of network training method, the BFGS algorithm based on Armijo-Goldstein criterion is adopted to improve the convergence of the network. In terms of reinforcement learning algorithm implementation, the network learning adopts periodic updating to reduce the performance requirements of reinforcement learning algorithm on computer and controller, enhance the stability of the algorithm in experiment, and optimize the system control law.

The specific organization of this paper is as follows: In section 2, the composition of traditional DDPG algorithm

and the realization principle of each part are firstly analyzed, and the problems existing in the application of traditional DDPG algorithm in the HDU position control system are analyzed. In section 3, the design method of improved DDPG algorithm based on BP neural network is presented in detail. In section 4, the research method of this paper is applied to the experimental research of legged robot HDU.

II. THE APPLICATION OF TRADITIONAL DDPG METHOD IN HYDRAULIC POSITION CONTROL SYSTEM

In the traditional DDPG method, the action strategy to be optimized by an agent is represented by a deep neural network, with the network input is the system state and the output is the corresponding action. In this way, a continuous non-linear action space can be learned to meet the requirements of complex tasks such as multi-joint robot. The algorithm structure is similar to DQN. There are two networks for the representation of the strategy, one for the target network to be eventually learned and improved, and the other for calculating the Temporal-Difference (TD) error when estimating the value function. The whole learning process takes random data from the memory bank. The brief structure of DDPG algorithm is shown in Fig.1.
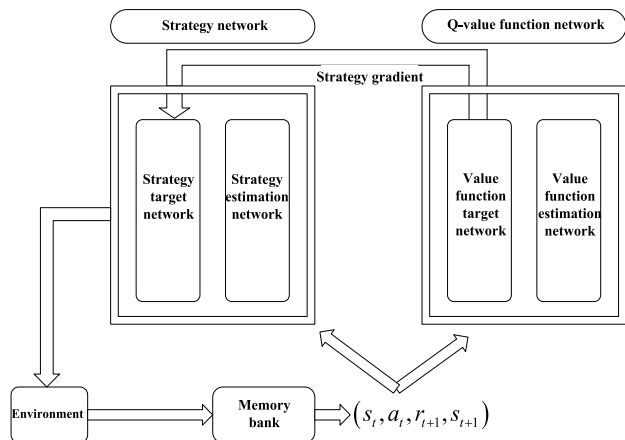

FIGURE 1. DDPG algorithm structure framework.

The goal of traditional DDPG is to continuously improve the strategy so that the agent can get higher and higher returns. The increasing long-term return is reflected by the increasing value of the optimal state-action value function (Q) of the actions performed by the agent. The Q-value function estimation method is as follows:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_{t+1}$$
$$+ \gamma \max Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)) \quad (1)$$

In the Eq.(1), r refers to the corresponding return value fed back by actions, s refers to the state, a refers to the action strategy taken, α refers to learning rate, γ refers to reward decay coefficient. The principle is as follows: according to the next state s_{t+1}, $Q(s_{t+1} + a_{t+1})$ in the previous Q table is selected as the estimated value of Q, where the maximum

value of $Q(s_{t+1} + a_{t+1})$ is selected to multiplied by the decay coefficient γ then plus the real return value r_{t+1} as the actual value of Q, and the difference between the actual value of Q and the estimated value of Q is used for continuous learning. In traditional DDPG, the strategy gradient that strategy improvement depends on is derived from Q-value function for many times. The way to update the parameters of the strategy target network is to gradient up the mapping of the target network of the value function, so that the action strategy changes towards the direction of the Q-value function increasing. The strategy gradient can be expressed as follows:

$$\frac{dQ(s, a)}{d\mu} = \frac{dQ(s, a)}{da} \cdot \frac{da}{d\mu} \quad (2)$$

In the equation, μ refers to strategy target network parameters, Q refers to value function target network.

The pseudo-code of the traditional DDPG algorithm is shown in Table 1.

TABLE 1. DDPG Algorithm pseudocode.

DDPG algorithm
Initialize value function target network $Q(s, a; \boldsymbol{\theta})$, Strategy target network $A(s; \boldsymbol{\mu})$
Use the two networks in the previous step to de-initialize the corresponding TD error estimation network $Q' \leftarrow Q$, $A' \leftarrow A$
Initialize the memory bank: M
for epoch=1: N
Initialize behavioral noise: NS
Initialize system state
for t=1: T
$$a_t = A(s_i; \boldsymbol{\theta}) + NS_t$$
Perform action a_t on the environment. get state s_{t+1} and return r_{t+1}
Store data in memory: $M \leftarrow (s_i, a_i, s_{i+1}, r_{i+1})$
Randomly select a minibatch containing group K samples from memory bank M
Make
$$y_i = r_{i+1} + \gamma Q'(s_{i+1}, A'(s_{i+1}); \boldsymbol{\theta}')$$
Update network parameters according to the target network loss function and value function:
$$CLoss = \frac{1}{K} \sum_{i=1}^{K} \left(y_i - \frac{1}{2} Q(s_i, a_i; \boldsymbol{\theta})^2 \right)$$
Update the target network parameters according to the strategy gradient:
$$PG = \frac{1}{K} \sum_{i=1}^{K} \frac{dQ(s_i, a_i)}{da_i} \cdot \frac{da_i}{d\boldsymbol{\mu}}$$
Update TD error estimation network:
$$\boldsymbol{\theta}' \leftarrow \tau\boldsymbol{\theta}' + (1-\tau)\boldsymbol{\theta}'$$ $$\boldsymbol{\mu}' \leftarrow \tau\boldsymbol{\mu}' + (1-\tau)\boldsymbol{\mu}'$$
end
end

According to the pseudo-code, the HDU position control system based on traditional DDPG algorithm deep learning is built. And xe_t is defined as the position error of the control system at time t, dxe_t as the velocity error of the control system at time t, and $ddxe_t$ as the acceleration error of the control system at time t. The input of DDPG agent is xe_t, dxe_t, and the output is the voltage signal given by the controller to the servo valve amplifier plate.

The concrete form of the design return function $r(s)$ can be expressed as follows:

$$\begin{cases} r = r1 + r2 \\ r1 = \begin{cases} -2\,|xe|, & |xe| < 0.05 \\ 0, & \text{other} \end{cases} \\ r2 = \begin{cases} -1, & |xe_t| > |xe_{t-1}| \\ 0, & |xe_t| \le |xe_{t-1}| \end{cases} \end{cases} \quad (3)$$

The value function network consists of input layer, 1 layer of hidden layer and output layer. The input is a three dimensional vector $[xe_t, dxe_t, a_t]$, and the number of neurons in the hidden layer is 10, it uses the Tanh activation function. The output is the value function of the state action pair, namely the Q-value.

The structure of the strategy network is similar to that of the value function network. The input is a two-dimensional vector $[xe_t, dxe_t]$ and the output is the voltage signal of the servo valve. In essence, the direct control system is the strategy target network that has been learning and improving in the DDPG algorithm. The control law can be expressed as follows:

$$u_t = A(s_t, a_t; \boldsymbol{\theta}) \quad (4)$$

where, u_t is the target strategy network output, that is, the servo valve voltage signal. A is the mapping formed by the strategy target network. θ is the network parameter.

Fig.2 shows the simulation tracking curve of HDU position control using traditional DDPG method. It can be seen in the figure, that after the start of the simulation, intelligent body output during the positive voltage signal, the piston is moving in the positive direction. However, when approaching the target curve, the controlled quantity does not decrease, it leads to the advance of the actual position compared with the expected position. At this point, in order to slow down the piston movement speed, the controlled quantity is reduced sharply, piston does backward movement, the curve is completely deviate from the expectation curve.
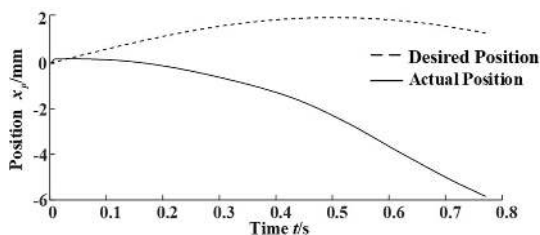

FIGURE 2. Traditional DDPG control method simulates position tracking.

After multiple learning processes under different working conditions, the control effect has not been improved. After analysis, the traditional DDPG method in the application of HDU servo control mainly has the following three problems:

(1) Insufficient exploration of agent. In the deterministic strategy, the output of the strategy network is a certain value, and the exploration of the agent is limited. After adding noise to the output, the exploration direction changes randomly, and

the depth and breadth of the exploration are insufficient, and it is difficult for the agent to learn rules from it.

(2) The performance of the neural network is poor and the convergence is too slow. The network in the DDPG algorithm needs to be updated synchronously in real time to achieve convergence quickly. Otherwise, in the early stage of learning, it will easily lead to the divergence of the control system, cause control failure and made the agent in trouble.

(3) The direct output signal of the strategy network is easy to lead to control failure. The servo valve voltage signal controls the opening size of the servo valve. The input signal of the agent learning in the early stage may not be of the appropriate order of magnitude of the controlled system, and the output fluctuates greatly, which is easy to lead to control failure.

III. DESIGN OF THE IMPROVED DDPG
Based on the three shortcomings proposed in the second section, this section intends to improve from three aspects: Firstly, the exploration strategy of the control method is improved to make the control method more likely to be successful, make the gradient information of the control method more regular, and reduce the learning difficulty of the agent. secondly, improving the convergence speed and fitting accuracy of neural network is very important for all artificial intelligence problems, especially for the learning of such online control system, the performance of neural network determines the success of the whole control algorithm. thirdly, the learning algorithm should have good stability, so that it can be applied in the actual control system, and the output should be kept within the safe range at all times, without causing hidden dangers to the controlled system.

Specific improvement measures are as follows:

1) For the exploration strategy of the agent, the fuzzy rule exploration strategy is designed to help the agent find the direction of learning more quickly.

2) For the performance of the neural network, the number of layers of BP neural network is increased in the aspect of network structure to improve the fitting ability of the network. In the aspect of network training method, the BFGS algorithm based on Armijo-Goldstein criterion is adopted to improve the convergence of the network.

3) In terms of reinforcement learning algorithm implementation, periodic updating is adopted in network learning to reduce the performance requirements of reinforcement learning algorithm on computer and controller, and to enhance the stability of the algorithm in experiments. Meanwhile, the control law of the system is also changed.

A. AN AGENT EXPLORATION STRATEGY BASED ON FUZZY OPTIMIZATION
Fuzzy concept reflects the human thinking process of classifying, judging and making decisions. For example, in the process of receiving water into a container, when there is little water in the container, turn the faucet on larger. when there is much water, turn the faucet on smaller. "little", "larger",

"much" and "smaller" are fuzzy concepts. Fuzzy thinking is applied to the control system to realize the classification and judgment of state quantities, and the results are used to control decisions to improve the adaptability of the system.

In the process of learning, the agent will constantly explore at random. In deterministic strategy reinforcement learning, Uhlenbeck-Ornstein (UO) random process is commonly used to generate noise. This method has good time sequence, and the generated random signals have strong temporal correlation, which is suitable for the environment with momentum properties such as the agent exploration control system. However, in order to realize the most efficient learning process, random exploration alone is not enough. A mentor is needed to guide the agent's exploration to the direction of the optimal solution and increase the probability of the agent finding better actions. Therefore, the fuzzy thinking is used to improve the exploration direction of agent, and the current system state is considered comprehensively to lead the exploration of agent to a direction that is relatively easy to obtain more returns. The basic principle of fuzzy exploration is that when the actual position of the system is ahead of the expected position, the controlled quantity should be reduced to reduce the speed of the hydraulic cylinder. when the actual position of the system falls behind the expected position, the controlled quantity should be increased to increase the speed of the cylinder, and the gap between the expected position should be narrowed. when the error is large, the scope of exploration should be expanded to find a better action. when the error is small, the exploration scope should be narrowed. Specific fuzzy rules are designed as follows:

The entire fuzzy exploration process is designed on the basis of UO random process, and the mean value and range of the output value of the random process are changed with fuzzy rules. The input of fuzzy rules is position error e, and the output is mean value mu and range of variation sig of the agent UO random process.

In view of the sinusoidal movement of the HDU controlled in this paper, the position tracking error is generally at the order of 10^{-1}mm. Considering that the initial learning error will be relatively large, the basic field of position tracking error e is set as [−2mm, 2mm]. For the convenience of using fuzzy language, the basic domain is divided into 7 parts according to the following boundary values:

$$\{-2, -1, -0.4, -0.1, 0.1, 0.4, 1, 2\}$$

When the actual situation exceeds its range, the maximum value is 2 or the minimum value is −2.

The basic field of mean mu of UO random process is set as [−3, 3], and the basic field is divided into 7 parts according to the following boundary value:

$$\{-3, -2, -1, -0.5, 0.5, 1, 2, 3\}$$

The basic field of change range sig of UO random process is set as [0, 1], and the basic field is divided into four parts according to the following boundary value:

$$\{0, 0.15, 0.3, 0.6, 1\}$$

The exact quantity is transformed into fuzzy quantity by fuzzy classification. In fuzzy systems, fuzzy language is used to express fuzzy cognition of quantity, and general use "big", "medium", "small" and other living speech to construct fuzzy evaluation rule. In the control system, the input quantity and the output quantity generally have the direction of positive and negative, therefore, considering "positive and negative", the expression of fuzzy language is as follows:

{negative-big, negative-middle, negative-big, zero, positive-small, positive- middle, positive-big}

The abbreviations are commonly used in algorithms:

$$\{NB, NM, NS, ZO, PS, PM, PB\}$$

The subordinating degree functions of the three variables all adopt the triangle membership function, and the method of solving fuzziness is centroid method. The specific fuzzy rules are listed as follows:

TABLE 2. Fuzzy rules for random processes of UO.

| The random process | Position error | | | | | | |
mean of UO	PB	PM	PS	ZO	NS	NM	NB
Direction of piston motion in HDU — Positive	PB	PM	PS	ZO	NS	NM	NB
Negative	NB	NM	NS	ZO	PS	PM	PB

TABLE 3. Fuzzy rules Table for UO random process variation range.

Position error	PB	PM	PS	ZO	NS	NM	NB
The range of random processes of UO	PB	PM	PS	ZO	PS	PM	PB

B. NEURAL NETWORK TRAINING ALGORITHM BASED ON ARMIJO-GOLDSTEIN CRITERION AND BFGS METHOD

1) ARMIJO-GOLDSTEIN CRITERION

The learning process of neural network is actually a nonlinear optimization problem. The weight of the network is changed through iterative optimization. The ideal state is that the loss function is 0. The most common method is the constant step gradient descent method, which is simple to implement. However, the performance fluctuates greatly during the learning process, and the learning variance is relatively large, so it cannot ensure that each step is approaching to the direction of convergence. Therefore, some search methods are needed to improve the convergence. The search methods can be divided

into accurate search and inaccurate search. Accurate search has the highest precision, but it requires a lot of computing resources. Inexact search, which does not require every step to be precision The number of search steps increases relatively, but the overall convergence speed is fast and the computing resources are relatively less.

Armijo-Goldstein's search guideline is an imprecise search process proposed by Armijo and Goldstein. Its core ideas are as follows: (1) The value of the target function has to go down enough to get close enough to the target. (2) The search step size should not be too small. The mathematical expression of Armijo-Goldstein criterion can be expressed as follows:

$$\begin{cases} f(x_k + \alpha_k d_k) \leq f(x_k) + \alpha_k \rho g_k^T d_k \\ f(x_k + \alpha_k d_k) \geq f(x_k) + \alpha_k (1-\rho) g_k^T d_k \end{cases} \quad (5)$$

where, $f()$ is the objective function. α_k is the step size. ρ is a constant, $0 < \rho < \frac{1}{2}$.

2) BFGS ALGORITHM

Neural network is trained by modifying network parameters so that the loss function is 0, it is essentially an unconstrained nonlinear optimization problem. BFGS method is a Quasi-Newton Methods. It is proposed to improve the shortcoming of Newtonian method that Hessian matrix occupies high computational resources, and construct approximate Hessian to reduce the computational workload. It is the best algorithm that balances the convergence performance and computational resource requirements at present. Its algorithm principle can be expressed as follows:

Assuming that $L(\theta)$ is the loss function of the neural network and θ is the parameter of the neural network, the learning problem of the neural network can be expressed as the unconstrained minimum problem under the determined data set:

$$\min_{\theta} f(\boldsymbol{\theta}) \quad (6)$$

By Taylor expansion of $f(\theta)$, only second-order and second-order terms are retained, and the following equation can be obtained:

$$\varphi(\boldsymbol{\theta}) = f(\theta_k) + \nabla f(\theta_k) \cdot (\boldsymbol{\theta} - \theta_k) \\ + \frac{1}{2} \cdot (\boldsymbol{\theta} - \theta_k)^T \cdot \nabla^2 f(\theta_k) \cdot (\boldsymbol{\theta} - \theta_k) \quad (7)$$

where, ∇f is the gradient of f, and the equation can be expressed as follows:

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial \theta_1} \\ \frac{\partial f}{\partial \theta_2} \\ \vdots \\ \frac{\partial f}{\partial \theta_N} \end{bmatrix} \quad (8)$$

$\nabla^2 f$ is Hessian matrix of f, and the equation can be expressed as follows:

$$\nabla^2 f = \begin{bmatrix} \frac{\partial^2 f}{\partial \theta_1^2} & \frac{\partial^2 f}{\partial \theta_1 \theta_2} & \frac{\partial^2 f}{\partial \theta_1 \partial \theta_N} \\ \frac{\partial^2 f}{\partial \theta_2 \partial \theta_1} & \frac{\partial^2 f}{\partial \theta_2^2} & \frac{\partial^2 f}{\partial \theta_2 \partial \theta_N} \\ \vdots & & \ddots & \vdots \\ \frac{\partial^2 f}{\partial \theta_N \partial \theta_1} & \frac{\partial^2 f}{\partial \theta_N \partial \theta_2} & \cdots & \frac{\partial^2 f}{\partial \theta_N^2} \end{bmatrix}_{N \times N} \quad (9)$$

Marking ∇f and $\nabla^2 f$ as g and H, respectively.

The necessary condition for the extremum is:

$$\varphi'(\boldsymbol{\theta}) = 0 \quad (10)$$

Simultaneous equation (8) and (9), and H is a nonsingular matrix, the equation can be obtained as follows:

$$\boldsymbol{\theta} = \boldsymbol{\theta}_k - \boldsymbol{H}_k^{-1} \cdot \boldsymbol{g}_k \quad (11)$$

Given the initial value θ_0, the iterative equation of Newton's method can be obtained as follows:

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \boldsymbol{H}_k^{-1} \cdot \boldsymbol{g}_k \quad (k = 0, 1, 2, \cdots) \quad (12)$$

The Quasi-Newton Method is used to construct positive definite matrix approximating Hessian matrix, and the complex calculation of Hessian is omitted.

By Taylor expansion of $f(\theta)$, only second-order and second-order terms are retained, and the following equation can be obtained:

$$\varphi(\boldsymbol{\theta}) = f(\boldsymbol{\theta}_k) + \nabla f(\boldsymbol{\theta}_k) \cdot (\boldsymbol{\theta} - \boldsymbol{\theta}_k) + \frac{1}{2} \cdot (\boldsymbol{\theta} - \boldsymbol{\theta}_k)^T \\ \cdot \nabla^2 f(\boldsymbol{\theta}_k) \cdot (\boldsymbol{\theta} - \boldsymbol{\theta}_k) \quad (13)$$

By taking the gradient from both sides of equation (13), it can be expressed as follows:

$$\boldsymbol{g}_{k+1} - \boldsymbol{g}_k \approx \boldsymbol{H}_{k+1} \cdot (\boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k) \quad (14)$$

Let $s_k = \theta_{k+1} - \theta_k, y_k = g_{k+1} - g_k$ to obtain the Quasi-Newton condition as follows:

$$\boldsymbol{s}_k \approx \boldsymbol{H}_{k+1}^{-1} \cdot \boldsymbol{y}_k \quad (15)$$

The core idea of this algorithm is to find the approximation of Hessian matrix H_k by matrix B_k.

$$\boldsymbol{B}_{k+1} = \boldsymbol{B}_k + \Delta \boldsymbol{B}_k \quad (16)$$

The following main work is to find the correction matrix ΔB_k, which is assumed to be a symmetric matrix in the form of Equation (16):

$$\Delta \boldsymbol{B}_k = \alpha \boldsymbol{u} \boldsymbol{u}^T + \beta \boldsymbol{v} \boldsymbol{v}^T \quad (17)$$

Simultaneous equation (15), (16) and (17), it can be obtained that:

$$\boldsymbol{y}_k = \boldsymbol{B}_k \boldsymbol{s}_k + (\alpha \boldsymbol{u}^T \boldsymbol{s}_k) \boldsymbol{u} + (\beta \boldsymbol{v}^T \boldsymbol{s}_k) \boldsymbol{v} \quad (18)$$

By the above equation, let $\alpha u^T s_k = 1$, $\beta v^T s_k = -1$, $u = y_k$ and $v = B_k s_k$, it can be obtained as follows:

$$\alpha = \frac{1}{y_k^T s_k}, \quad \beta = -\frac{1}{s_k^T B_k s_k} \tag{19}$$

The correction matrix can be obtained as follows:

$$\Delta B_k = \frac{y_k y_k^T}{y_k^T s_k} - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} \tag{20}$$

To sum up, the iterative equation of approximate Hessian matrix is expressed as follows:

$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} \tag{21}$$

3) DESIGN OF POSITION CONTROL METHOD BASED ON IMPROVED DDPG ALGORITHM

In combination with Armijo-Goldstein criterion and BFGS algorithm, the neural network training algorithm steps in this section are as follows:

1. Initialize the network parameters and give the convergence precision.

2. Determine the direction of search.

3. Search according to Armijo-Goldstein criterion and determine the optimal step size.

4. Calculate the approximate Hessian matrix.

5. Judge whether it converges according to the given conditions. If yes, end the algorithm. If no, go back to step 2.

The pseudo-code of the improved DDPG algorithm is shown in Table 4.

The improved DDPG method proposed in this paper has the following limitations:

1. The improved DDPG method has randomness in the learning process, so unstable learning situation will occur occasionally. By adjusting hyperparameters can reduce the probability of instability, but it cannot completely avoid the occurrence of instability.

2. Due to the difference between the accuracy of the simulation mathematical model of HDU position control system and the experimental system, the learning results in the simulation cannot be directly applied to the experimental system. Instead, the learning steps should be followed step by step in the experimental system.

3. The improved DDPG method used in this paper is designed for HDU position control system. Whether this method has good performance in other systems remains to be done by future work.

IV. SIMULATION ON HDU POSITION CONTROL BASED ON IMPROVED DDPG ALGORITHM

The improved DDPG deep learning network is used to replace the controller. The overall control structure diagram of HDU position control method based on the improved DDPG algorithm is shown in Fig.3.

In order to improve the stability of the learning algorithm in the actual system, reduce the experimental risk, and ensure

TABLE 4. Improved DDPG Algorithm pseudocode.

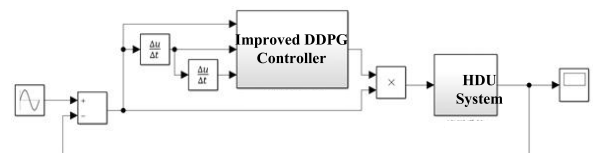| Improved DDPG algorithm |
| --- |
| Initialize value function target network $Q(s,a;\theta)$, Strategy target network $A(s;\mu)$ |
| Use the two networks in the previous step to de-initialize the corresponding TD error estimation network $Q' \leftarrow Q$, $A' \leftarrow A$ |
| Initialize the memory bank: M |
| for epoch=1: N (Each epoch is a round of learning) |
| Initialize behavioral noise: NS |
| Initialize system state |
| for t=1: T (T is the time length of each task cycle) |
| According to the fuzzy rules to calculate the exploration noise NS_t |
| $$a_t = A(s_t;\theta) + NS_t$$ |
| Perform action a_t on the environment. get state s_{t+1} and return r_{t+1} |
| Store data in memory: $M \leftarrow (s_t,a_t,s_{t+1},r_{t+1})$ |
| end |
| Randomly select a minibatch containing group K samples from memory bank M |
| Make |
| $$y_i = r_{i+1} + \gamma Q'(s_{i+1}, A'(s_{i+1});\theta')$$ |
| Update the network parameters according to the value function and the target network loss function until the fitting error meets the requirements: |
| $$CLoss = \frac{1}{K}\sum_{i=1}^{K}\left(y_i - \frac{1}{2}Q(s_i,a_i;\theta)\right)^2$$ |
| Update the target network parameters according to the strategy gradient: |
| $$PG = \frac{1}{K}\sum_{i=1}^{K}\frac{dQ(s_i,a_i)}{da_i}\cdot\frac{da_i}{d\mu}$$ |
| Update TD error estimation network: |
| $$\theta' \leftarrow \tau\theta' + (1-\tau)\theta'$$ $$\mu' \leftarrow \tau\mu' + (1-\tau)\mu'$$ |
| end |

FIGURE 3. Block diagram of HDU position control system based on the improved DDPG.

the safety of human and equipment, the output of DDPG agent multiplied by the current system error is output to the hydraulic system as the final control quantity. Then, the input of DDPG control module is position error e, velocity error de, acceleration error dde, the output is gain coefficient, and the control law of the system is:

$$u = A(s) \cdot (x_r - x_p) \tag{22}$$

where, A is the mapping formed by the policy target network, s is the input of DDPG agent, x_r is the expected position of the system, and x_p is the actual position of the system.

In order to improve the fitting performance of the neural network, the double layer neural network is designed to learn the value function and the action strategy. The BFGS algorithm based on Armijo-Goldstein criterion is used for the training of the value function network, the training of the

strategy network adopts the gradient descent algorithm. After many experiments with different parameters, the balance fitting performance and convergence speed are balanced, and the structural parameters of the final value function network and the strategy network are determined as shown in Table 5 and Table 6.

TABLE 5. Structure parameter of value function network.

Number of hidden layer	Number of neurons per layer	Hidden layer neuron activation function	Output layer neuron activation function
2	[4,6,10,1]	Tanh	no

TABLE 6. Structure parameters of strategy network.

Number of hidden layer	Number of neurons per layer	Hidden layer neuron activation function	Output layer neuron activation function
2	[4,6,10,1]	Tanh	Logistic

In order to better guide the learning of agents, the working conditions in this section are designed from easy to difficult, as shown in Table 7.

TABLE 7. Simulation working condition of the improved DDPG.

| Number of working conditions	Sinusoidal input frequency /Hz	Sinusoidal input amplitude /mm
①	0.5	2
②	1	4
③	2	6

The initial position of the piston of the HDU is 25mm. The improved DDPG control method is used to learn the simulation condition ①-③ in Table 7 in turn, and the learning effect is observed.

First, the simulation working condition ① is learned. The output after the sixth round is basically unchanged. It can be considered that the learning has basically converged. Due to space limitation, only the first two and the sixth rounds of the agent learning effect are shown. The position tracking curve of the HDU and the DDPG strategy network output curves are shown in Fig. 4 to Fig. 6.

In the first round of learning, the output of the strategy network of the agent is roughly between 3 and 3.5. With the increase of learning times, the output of the strategy network gradually increases. When it comes to the sixth round, the system achieves the optimal control performance and the output of the strategy network is constant. The maximum tracking error of the system is from 0.042mm in the first round to 0.023 in the sixth round. The agent reduces the maximum tracking error by 45.8% through learning, indicating that the trained agent is capable of independent exploration and self-learning in the absence of system model and human
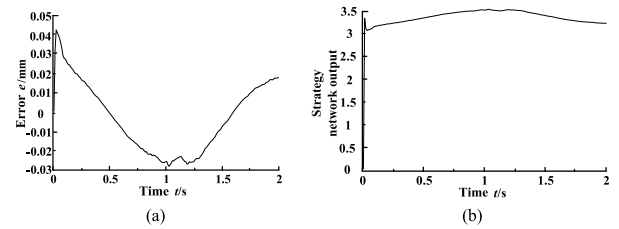
FIGURE 4. Training results of the first round in working condition ①. (a) Position tracking error. (b) Strategy network output.
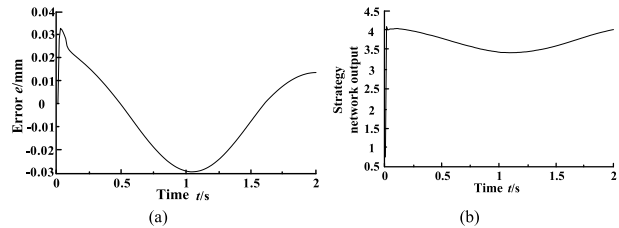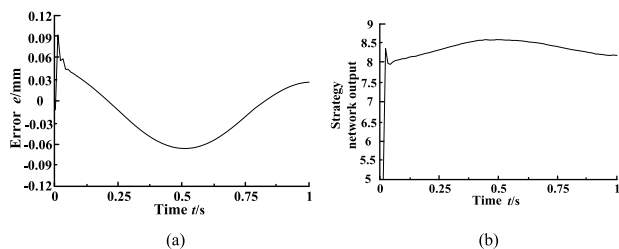

FIGURE 5. Training results of the second round in working condition ①. (a) Position tracking error. (b) Strategy network output.
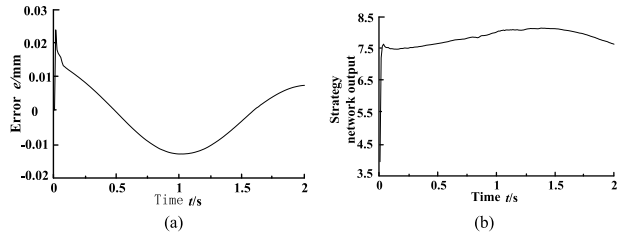
FIGURE 6. Training results of the sixth round in working condition ①. (a) Position tracking error. (b) Strategy network output.

prior knowledge. The output curve of the strategy network and the position tracking curve of the HDU after the convergence of agent learning working conditions ② and ③ are shown in Fig. 7 and Fig. 8.

FIGURE 7. Training results of working condition ②. (a) Position tracking error. (b) Strategy network output.

The simulation results show that the control method based on improved DDPG can significantly improve the control effect through exploration and learning without prior knowledge, and finally achieve the performance close to the variable value PID method. It improves the adaptive ability of the HDU under unknown working conditions, which has practical significance, and provides theoretical and practical reference for improving the control performance of the robot under unknown complex environment.
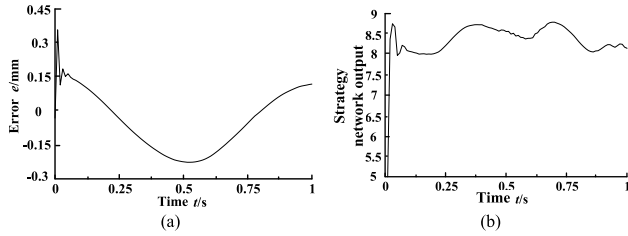
FIGURE 8. Training results of working condition ③. (a) Position tracking error. (b) Strategy network output.

V. EXPERIMENT

A. EXPERIMENTAL RESULTS

The method proposed in this paper is intended to conduct experimental research on the HDU performance test platform of legged robot [22]. The platform is mainly installed by two HDUs in the form of opposite top. The left HDU adopts position closed-loop control as the system to be tested, while the right HDU adopts force closed-loop control as the loading system. In the experiment, the left HDU performs performance tests of relevant control algorithms, while the right HDU performs zero force servo control to ensure the same working conditions as the left HDU in each experiment. In order to test the learning ability of HDU position control system under different working conditions, experimental working conditions is the same as Table 7.

In order to comprehensively verify the performance of the improved DDPG control method, the working conditions shown in Table 6 are used for experimental verification on the performance test bench of the HDU, and the value error curve obtained by the constant value PID and variable value PID control method is introduced to compare the control performance of the test system under different conditions. The above two PID algorithms have been studied in the previous research of the author's team. This paper will not give specific theories due to the limited space. The initial position of the HDU piston is 25mm, and the system oil source pressure is 7MPa.

First of all, the working condition 1 is studied. After a total of 9 rounds of learning, network parameters basically did not change. The position control system error and strategy network output results are shown in Fig.9.
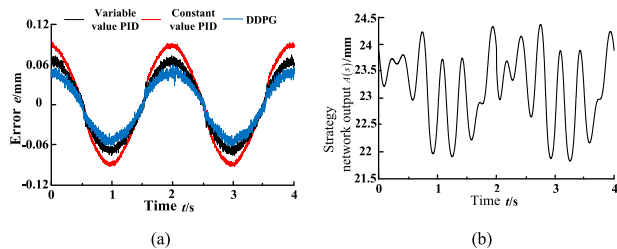

FIGURE 9. Experimental error curve and agent control signal in working condition 1. a) Position tracking error comparison in working condition ①. b) Strategy network output.

Table 8 shows the HDU position control system error table in Fig.9.

TABLE 8. Control system error table.

Control method	Constant value PID	Variable value PID	DDPG
Maximum error /mm	0.09	0.07	0.05
Maximum relative error	4.5%	3.5%	2.5%

According to Fig.9 (a) and Table 8, after many rounds of learning, the agent's output basically does not change anymore, so it can be considered that the learning has converged. At this time, the output of the strategy network fluctuates around 23, and the maximum position tracking error after convergence is about 2.5% of the amplitude of the input sinusoidal signal, indicating that the improved DDPG control method designed in this section has preliminarily possessed the ability of self-exploration and self-learning without any prior knowledge. In terms of control accuracy, compared with the constant value PID method under the same working condition ①, the control error of the improved DDPG method after training is 44.4% lower than that of the constant value PID method and 28.6% lower than that of the variable value PID method.

Then, the learning process of working condition ② and ③ is similar to that of working condition ①. In order to simplify the content of the paper, only the learning results after the final convergence are shown. When the network parameters are basically unchanged, the error of the position control system and the output result of the strategy network are shown in Fig.10.
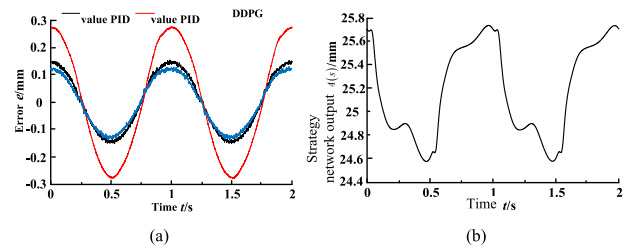

FIGURE 10. Experiment error curve and agent control signal in working condition ②. a) Position tracking error comparison in working condition ②. b) Strategy network output.

Table 9 shows the HDU position control system error table in Fig.10.

When the network parameters are basically unchanged, the error of the position control system and the output result of the strategy network are shown in Fig.11.

Table 10 shows the HDU position control system error table in Fig.11.

It can be seen from Table 7 and 8 that the deep learning position control algorithm based on DDPG under working conditions ② and ③ with the sinusoidal frequency and sinusoidal amplitude increased finally reached the training results

TABLE 9. Control system error table.

Control method	Constant value PID	Variable value PID	DDPG
Maximum error /mm	0.28	0.14	0.12
Maximum relative error	7%	3.5%	3%
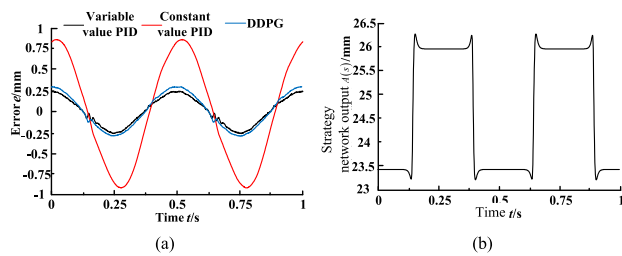
FIGURE 11. Experiment error curve and agent control signal in working condition ③ a) Position tracking error comparison in working condition ③. b) Strategy network output.

TABLE 10. Control system error Table.

Control method	Constant value PID	Variable value PID	DDPG
Maximum error /mm	0.9	0.25	0.31
Maximum relative error	15%	4.17%	5.2%

with the maximum relative error no more than 4% and 6% respectively. Under working condition ②, the control error of the improved DDPG method after training is 57.1% lower than that of the constant value PID method and 14.3% lower than that of the variable value PID method. Under working condition ③, the control error of the improved DDPG method after training is 65.3% lower than that of the constant value PID method and 24.8% higher than that of the variable value PID method.

The experimental results show that when the amplitude and frequency of the experimental signal increase, along with the maximum relative error up to 15%, the constant value PID method has the worst control effect, that seriously affects the normal work of HDU and does not have self-adaptability. The variable value PID method has the best control effect with the maximum relative error of 4.17%. The relative constant value PID method has a great improvement in the control accuracy. However, the control effect of this method depends on the working experience of the experimenters and their deep understanding of the system. It requires a lot of time to stop and debug before the experiment of each condition, so it has no self-adaptability. The maximum relative error of

the control system trained by the improved DDPG method is 5.2%, and its control effect is better than that of the constant value PID and variable value PID methods in working condition ① and ②, but slightly worse than that of the variable value PID method in working condition ③. However, the overall control effect is similar to the variable value PID, which can achieve high-precision control effect under the condition of avoiding downtime and debugging, and has a good self-adaptability.

It can be seen from the training results of the three working conditions that the improved DDPG control method is greatly improved compared with that before the improvement, and the algorithm becomes stable, so that it can be safely applied in the actual system. It can be seen from the training results of each working condition that the control method can continuously learn by itself in different working conditions to improve the control performance of the system. However, due to the limited fitting accuracy of the neural network for the dynamic system, if the error is reduced to a certain extent, the performance difference brought by the change of the output of the strategy network will be greatly reduced until it is less than the learning accuracy of the neural network. At this time, the network cannot generate effective strategy gradient signals to promote the learning of agent, so the learning accuracy of agent will be limited. It can be seen from the output curve of the policy network during the learning process that the policy network calculates the output amount according to the real-time state of the system. In each sinusoidal period, the network output also fluctuates periodically. The higher the input sinusoidal signal frequency is and greater the sinusoidal signal amplitude is, the smaller the fluctuation of the network output will. However, the mode of fluctuation in each working condition is not regular. Therefore, considering that the fluctuation is generated from two aspects: First, the sensor noise has a random influence on the output of the strategy network. When the amplitude and frequency of the input sinusoidal signal are large, the value of several state variables of the system will also be large, but the influence of the sensor noise will be relatively small. Second, the network learning accuracy is limited. When the error is reduced to a certain extent, the network can not provide effective strategy gradient information to guide the agent to continue learning, and the agent output will fluctuate within a certain range.

VI. CONCLUSION
In this paper, an improved DDPG control method is designed. Focus on the exploration strategy of the agent, fuzzy thinking is adopted to optimize the exploration direction of the agent under different states. To solve the problem of network performance, the number of network hidden layers is increased, and the BFGS method based on Armijo-Goldstein criterion is adopted as the training algorithm of value function network. To solve the problem of algorithm stability, periodic updating strategy is adopted in network learning. The experiments results show that: The improved DDPG control method can reduce the control error of the controlled system by learning

the control error and quickly setting the controlled quantity. After the agent convergence, the control error of the system can be reduced by an average of 55.6% compared with the constant value PID method, which is similar to the effect of variable value PID. It can not only achieve the high precision control of manual debugging, but also avoid the waste of time caused by downtime debugging, and it can show good adaptive ability under complex and continuous working conditions.

Future work: First, the control method proposed in this paper is applied to the prototype of quadruped robot, and attention should be paid to the control effect and working condition adaptability. Second, the application of deep learning in HDU force control system will be explored. Third, the method will be extended to other fields.

REFERENCES

[1] M. Focchi, V. Barasuol, I. Havoutis, J. Buchli, C. Semini, and D. G. Caldwell, "Local reflex generation for obstacle negotiation in quadrupedal locomotion," in *Proc. Int. Conf. Climbing Walking Robots Support Technol (CLAWAR)*, Sydney, NSW, Australia, 2013, pp. 443–450.

[2] L. Lyu, Z. Chen, and B. Yao, "Development of pump and valves combined hydraulic system for both high tracking precision and high energy efficiency," *IEEE Trans. Ind. Electron.*, vol. 66, no. 9, pp. 7189–7198, Sep. 2019.

[3] J. Luo, Y. Su, L. Ruan, Y. Zhao, D. Kim, L. Sentis, and C. Fu, "Robust bipedal locomotion based on a hierarchical control structure," *Robotica*, vol. 37, no. 10, pp. 1750–1767, Oct. 2019.

[4] J. Cho, J. T. Kim, S. Park, and K. Kim, "Dynamic walking of JINPOONG on the uneven terrain," in *Proc. 10th Int. Conf. Ubiquitous Robots Ambient Intell. (URAI)*, Jeju, South Korea, Oct. 2013, pp. 468–469.

[5] M. Raibert, "BigDog, the rough-terrain quadruped robot," in *Proc. 17th World Congr.*, 2008, pp. 1–5.

[6] Boston Dynamics. (2012). *CHEETAH-Fastest Legged Robot, Dynamics*. Accessed: Oct. 17, 2012. [Online]. Available: http://www.bostondynamics.com/robot_cheetah.html

[7] (Oct. 2013). *Introducing WildCat*. [Online]. Available: http://www.youtube.com/watch?v=wE3fmFTtp9g

[8] C. D. Remy, O. Baur, M. Latta, A. Lauber, M. Hutter, M. A. Hoepflinger, C. Pradalier, and R. Siegwart, "Walking and crawling with ALoF: A robot for autonomous locomotion on four legs," *Ind. Robot, Int. J.*, vol. 38, no. 3, pp. 264–268, May 2011.

[9] Y. Ye, C.-B. Yin, Y. Gong, and J.-J. Zhou, "Position control of nonlinear hydraulic system using an improved PSO based PID controller," *Mech. Syst. Signal Process.*, vol. 83, pp. 241–259, Jan. 2017.

[10] G. Wrat, M. Bhola, P. Ranjan, S. K. Mishra, and J. Das, "Energy saving and fuzzy-PID position control of electro-hydraulic system by leakage compensation through proportional flow control valve," *ISA Trans.*, vol. 101, pp. 269–280, Jun. 2020.

[11] W. Sun, J.-W. Lin, S.-F. Su, N. Wang, and M. J. Er, "Reduced adaptive fuzzy decoupling control for lower limb exoskeleton," *IEEE Trans. Cybern.*, early access, Feb. 26, 2020, doi: 10.1109/TCYB.2020.2972582.

[12] T. H. Ho and K. K. Ahn, "Speed control of a hydraulic pressure coupling drive using an adaptive fuzzy sliding-mode control," *IEEE/ASME Trans. Mechatronics*, vol. 17, no. 5, pp. 976–986, Oct. 2012.

[13] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, Jan. 2016.

[14] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J.-M. Allen, V.-D. Lam, A. Bewley, and A. Shah, "Learning to drive in a day," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, Montreal, QC, Canada, May 2019, pp. 8248–8254.

[15] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Sci. Robot.*, vol. 4, no. 26, Jan. 2019, Art. no. eaau5872.

[16] I. Carlucho, M. De Paula, S. A. Villar, and G. G. Acosta, "Incremental Q-learning strategy for adaptive PID control of mobile robots," *Expert Syst. Appl.*, vol. 80, pp. 183–199, Sep. 2017.

[17] R. Yu, Z. Shi, C. Huang, T. Li, and Q. Ma, "Deep reinforcement learning based optimal trajectory tracking control of autonomous underwater vehicle," in *Proc. 36th Chin. Control Conf. (CCC)*, Dalian, China, Jul. 2017, pp. 26–28.

[18] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," in *Proc. ICLR*, London, U.K., Sep. 2015, pp. 1–14.

[19] C. Shen, C. Fan, Y. Wang, and W. Xue, "Limited memory BFGS algorithm for the matrix approximation problem in frobenius norm," *Comput. Appl. Math.*, vol. 39, no. 2, p. 43, Feb. 2020.

[20] N. Deng and Z. Li, "Convergence properties of a modified BFGS algorithm for minimization with Armijo-Goldstein steplengths," *J Comput. Math.*, vol. 17, no. 6, pp. 645–652, 1999.

[21] K.-X. Ba, G.-L. Ma, B. Yu, Z.-G. Jin, Z.-P. Huang, J.-X. Zhang, and X.-D. Kong, "A nonlinear model-based variable impedance parameters control for position-based impedance control system of hydraulic drive unit," *Int. J. Control, Autom. Syst.*, vol. 18, no. 7, pp. 1806–1817, Jul. 2020.

[22] K.-X. Ba, B. Yu, X.-D. Kong, C.-H. Li, Q.-X. Zhu, H.-L. Zhao, and L.-J. Kong, "Parameters sensitivity characteristics of highly integrated valve-controlled cylinder force control system," *Chin. J. Mech. Eng.*, vol. 31, p. 43, 2018.

MINGLI ZHANG received the Ph.D. degree from Yanshan University, in 2013. She is currently a Professor with the School of Economics and Management, Yanshan University. Her research interests include complex network information dissemination, innovation and entrepreneurship management, and industrial cluster management.

YIJIE ZHANG is currently pursuing the Ph.D. degree with Yanshan University, China. Her main research interests include complex network information dissemination and industrial cluster management.

ZHENGJIE GAO is currently pursuing the master's degree with Yanshan University, China. His main research interests include electro-hydraulic servo control systems and robot design and control.

XIAOLONG HE is currently pursuing the Ph.D. degree with Yanshan University, China. His main research interests include electro-hydraulic servo control systems and robot design and control.

● ● ●