DOI:
10.1109/TCSII.2019.2947682

Document Version
Peer reviewed version

Reinforcement Learning-based Control of Nonlinear Systems using Lyapunov Stability Concept and Fuzzy Reward Scheme

Ming Chen, Hak-Keung Lam, *Senior Member, IEEE*, Qian Shi and Bo Xiao, *Member, IEEE*

***Abstract*—In this paper, a reinforcement learning-based control approach for nonlinear systems is presented. The proposed control approach offers a design scheme of the adjustable policy learning rate (APLR) to reduce the influence imposed by negative or large advantages, which improves the learning stability of the proximal policy optimization (PPO) algorithm. Besides, this paper puts forward a Lyapunov-fuzzy reward system to further promote the learning efficiency. In addition, the proposed control approach absorbs the Lyapunov stability concept into the design of the Lyapunov reward system and a particular fuzzy reward system is set up using the knowledge of the cart-pole inverted pendulum and fuzzy inference system (FIS). The merits of the proposed approach are validated by simulation examples.**

***Index Terms*—Proximal policy optimization (PPO), adjustable policy learning rate (APLR), Lyapunov reward system, fuzzy reward system, cart-pole inverted pendulum.**

I. Introduction

THE real-world control systems are full of different categories of nonlinearities, such as inverted pendulum [1], continuum manipulator [2], chemical reactor [3], quadcopter [4] etc.. As one of the most effective ways to achieve the nonlinear control objective, reinforcement learning (RL) based control strategy trains an agent that is able to learn the optimal control policy by directly interacting with environment in a trial-and-error manner [5], [6]. In 1989, Q-learning algorithm, which is an off-policy RL algorithm was developed for optimal control [7]. Nevertheless, basic Q-learning algorithm has to check the state-action values in the Q-table to take the optimal action at every state. The Q-table is formed by state spaces and action spaces, which are finite and discrete to store all of the state-action values as a look-up table. Due to the limitations of Q-table, basic Q-learning algorithm cannot be directly applied to the control problems which have continuous and high-dimensional state and action spaces. The high dimensions of the state and action spaces raise the computational burden and cause the problem that is known as curse of dimensionality (CoD). To address this issue, approximate RL can be considered, in which the value function is approximated by neural networks (NNs). With the tremendous breakthrough on deep learning in recent years [8], deep Q-network (DQN) was developed for handling a variety of difficult assignments through learning, which reaches or even surpasses the human counterpart [9]. Although DQN can be utilized in the continuous and high-dimensional state case with the assistance of deep neural network (DNN). However, the action spaces of DQN are still discrete and finite, which might encounter the CoD problem as well. [10] presented a deep deterministic policy gradient (DDPG) algorithm which is an actor-critic approach with DNN function approximators, where discretization of action spaces is not required. However, learning rate has a tremendous impact on DDPG that small learning rate will slow down the convergence while large learning rate might result in the terrible performance [11]. [12] presented a trust region policy optimization (TRPO) algorithm which takes Kullback–Leibler (KL) divergence constraint into the consideration for circumventing this problem. In 2017, proximal policy optimization (PPO) algorithm was proposed, which is inspired by TRPO but simpler and has excellent performance [13].

In this paper, PPO is applied to the benchmark control problem of the cart-pole inverted pendulum, which is a classical RL problem [14], [15]. However, PPO will encounter two main challenges when it is applied to deal with the benchmark control problem of the cart-pole inverted pendulum. The first challenge is the instability during the learning process, which results from negative or large advantages [16], [17]. The second challenge is the design of the reward system, which will have a dramatic influence on the learning effect [18].

Hence, this paper proposes a scheme of the adjustable policy learning rate (APLR). Comparatively small policy learning rate is used to adjust policy slightly when actions with negative expectation of advantages are learned, which encourages the learning of correct actions and mitigates the learning instability. In addition, APLR will be constructed according to the size of the expectation of advantages. For example, the policy learning rate will be designed to be small by the proposed approach for guaranteeing the stable optimization of the policy gradient if the expectation of advantages is too large. For the second challenge, this paper proposes a comprehensive reward system which incorporates the concepts of the Lyapunov stability and the fuzzy inference system (FIS), as well as the knowledge of the cart-pole inverted pendulum [19]. Lyapunov reward system is based on the concept of stability from control theory, which guides the learning process by the Lyapunov stability theory instead of learning by pure trial-and-error.

This research was supported by King's College London. And this study was also financially supported by the grants of China Scholarship Council. (Corresponding author: Hak-Keung Lam.)

Ming Chen, Hak-Keung Lam and Qian Shi are with the Department of Engineering, King's College London, Strand, London, WC2R 2LS, United Kingdom. (e-mail: {ming.1.chen, hak-keung.lam, qian.shi}@kcl.ac.uk).

Bo Xiao is with the Hamlyn Centre for Robotic Surgery and Department of Computing, Imperial College London, London, SW7 2AZ, United Kingdom. (e-mail: b.xiao@imperial.ac.uk).

Furthermore, a specific fuzzy reward system is designed using the knowledge of the cart-pole inverted pendulum and FIS. Lyapunov reward system and fuzzy reward system will be integrated into a comprehensive reward system through the weighted connection, which can achieve more proper reward from the system stability and behavior point of view.

The rest of the paper is organized as follows. Section II introduces the concepts of PPO and FIS. Section III describes the proposed schemes of the APLR and Lyapunov-fuzzy reward system. Simulation parameters and results are shown in Section IV. Section V draws a conclusion.

II. Preliminary

A. Proximal policy optimization

PPO clips the objective function to restrict the policy update for improving the learning performance:

$$J_t{}^{clip}(\theta) = \mathbb{E}[min(\rho_t(\theta)\hat{A}_t, clip(\rho_t(\theta), 1-\xi, 1+\xi)\hat{A}_t)], \quad (1)$$

where $\mathbb{E}[.]$ denotes the empirical expectation over a finite batch of samples, $clip(.)$ denotes the clipping function, which is defined as follows, if $\rho_t(\theta) > (1+\xi)$, then $\rho_t(\theta)$ will be clipped by $(1+\xi)$; if $\rho_t(\theta) < (1-\xi)$, then $\rho_t(\theta)$ will be clipped by $(1-\xi)$; otherwise $\rho_t(\theta)$ will not be clipped. $\rho_t(\theta)$ indicates $\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta old}(a_t|s_t)}$ which is the probability ratio. As defined in [5], $\pi(a_t|s_t)$ denotes the selected probability of action a given state s at time t with stochastic policy π. θ denotes the vector of policy weights and θ_{old} denotes the vector of policy weights ahead of the update, ξ denotes the hyper-parameter, $\hat{A}_t$ is constructed by the generalized advantage estimation (GAE) technique [20].

B. Fuzzy inference system

FIS is a system processing input information using human spirit represented by fuzzy logic to general decision. It consists of four parts which are fuzzifier, knowledge base involving rule base or database, fuzzy inference engine and defuzzifier. Fuzzifier converts the crisp (real-valued) input into a fuzzy set which is represented by membership functions (MFs). Knowledge base is the database made up of linguistic rules which can be described as the following format: IF antecedent condition THEN consequence. Fuzzy inference engine generates the fuzzy output according to the fuzzy inputs on the basis of the knowledge base. The function of defuzzifiers is to transform the fuzzy output to a crisp (real-valued) output.

III. Methodology

A. Adjustable policy learning rate

The control performance depends on the selection of actions that correct action is able to achieve better control performance. However, the range of the proper actions which can realize the control purpose might not be wide, let alone the precise action, which means RL agent will learn improper actions more. The learning of actions with negative advantages will drive the policy away from improper actions, but superfluous learning of improper actions will lead to the instability [16]. Consequently, this paper presents heterogeneous learning rates in Algorithm 1 for making policy more focus on actions which can bring out positive advantages and decrease the harm of the learning of actions with negative advantages.

Even though PPO has tried to improve the stability by the limitation on the change in probability ratio, the large case of $\hat{A}_t$ is still a key matter which will affect the learning stability.

PPO presented to optimize the objective function with respect to policy weights by adaptive moment estimation (Adam) [21]:

$$\theta \leftarrow \theta + \alpha^{policy} \cdot \text{Adam}(J_t{}^{clip}(\theta)), \quad (2)$$

where α^{policy} denotes the policy learning rate.

Therefore, we propose to circumvent the problem of large $\hat{A}_t$ by the design of APLR, which is shown in Algorithm 1.

Algorithm 1 The design scheme of the APLR

if $\mathbb{E}[\hat{A}_t] \geq 0$ **then**
  if $|\mathbb{E}[\hat{A}_t]| > \eta_1$ (where $|\cdot|$ denotes the absolute value operator) **then**
    $\alpha_2^{policy} = \frac{\alpha_1^{policy}\eta_2(2-\eta_3^{(\eta_2-|\mathbb{E}[\hat{A}_t]|)})}{|\mathbb{E}[\hat{A}_t]|}$
    (where α_1^{policy} denotes the original policy learning rate, $\eta_1,\ \eta_2,\ \eta_3 > 0$)
  else
    $\alpha_2^{policy} = \alpha_1^{policy}$
  end if
else
  if $|\mathbb{E}[\hat{A}_t]| > \eta_4$ **then**
    $\alpha_2^{policy} = \frac{\alpha_1^{policy}\eta_5(2-\eta_6^{(\eta_5-|\mathbb{E}[\hat{A}_t]|)})}{|\mathbb{E}[\hat{A}_t]|}$
    (where $\eta_4,\ \eta_5,\ \eta_6 > 0$)
  else
    $\alpha_2^{policy} = \alpha_1^{policy}$
  end if
  $\alpha_2^{policy} = \alpha_2^{policy}\eta_7$
  (where η_7 is a small positive number)
end if

When $\mathbb{E}[\hat{A}_t]$ is not negative, if the absolute value of $\mathbb{E}[\hat{A}_t]$ is more than the threshold (η_1) which means the the absolute value is so large that the stability might be affected, new policy learning rate will replace the original policy learning rate for guaranteeing the learning stability. In Algorithm 1, $\frac{\alpha_1^{policy}\eta_2(2-\eta_3^{(\eta_2-|\mathbb{E}[\hat{A}_t]|)})}{|\mathbb{E}[\hat{A}_t]|} < \alpha_1^{policy}$ if overall $\hat{A}_t$ in the objective function is excessively large, which demonstrates the policy learning rate will diminish for offsetting the impact imposed by large $\hat{A}_t$. The design scheme has the following property: $\eta_2(2-\eta_3^{(\eta_2-\eta_1)}) = \eta_1$, and $\eta_2(2-\eta_3^{(\eta_2-|\mathbb{E}[\hat{A}_t]|)}) < |\mathbb{E}[\hat{A}_t]|$ for the case $\eta_1 < |\mathbb{E}[\hat{A}_t]|$; $\eta_5(2-\eta_6^{(\eta_5-\eta_4)}) = \eta_4$, and $\eta_5(2-\eta_6^{(\eta_5-|\mathbb{E}[\hat{A}_t]|)}) < |\mathbb{E}[\hat{A}_t]|$ for the case $\eta_4 < |\mathbb{E}[\hat{A}_t]|$. In the negative condition, α_2^{policy} should multiply a small positive value for avoiding the occurrence of the instability resulting from the learning of actions with negative advantages.

B. Cart-pole inverted pendulum

The dynamic model of the cart-pole inverted pendulum [22] is given by:

$$\dot{x}_1(t) = x_2(t), \tag{3}$$

$$\dot{x}_2(t) = \frac{\begin{pmatrix} g\sin(x_1(t)) - \varphi m_p l (x_2(t))^2 \sin(2x_1(t))/2 \\ -\varphi\cos(x_1(t))u(t) \end{pmatrix}}{4l/3 - \varphi m_p l(\cos(x_1(t)))^2}, \tag{4}$$

where $x_1(t)$ denotes the angular displacement of the inverted pendulum (rad), $x_2(t)$ denotes the angular velocity of the inverted pendulum (rad/s), g is the gravity acceleration which is 9.8m/s^2, m_p is the mass of the inverted pendulum which is 1kg, M_c is the mass of the cart which is 18kg, $\varphi = \frac{1}{m_p + M_c}$, l is the distance from the centre of mass of the inverted pendulum to the shaft axis which is 0.5m, $u(t)$ denotes the force which is applied to the cart (N). The control objective is to stabilise the cart-pole inverted pendulum, i.e., $\lim_{t\to\infty} x_1(t) = 0$ and $\lim_{t\to\infty} x_2(t) = 0$.

C. Scheme of the reward system

A design scheme of the specific reward system is presented for the cart-pole inverted pendulum which consists of the Lyapunov reward system and fuzzy reward system.

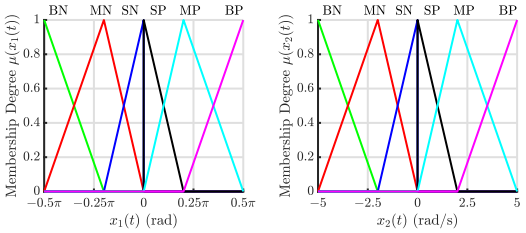
1) Lyapunov reward system: the Lyapunov reward system is based on the Lyapunov stability concept. The architecture of the Lyapunov reward system is shown in Algorithm 2.

Algorithm 2 Lyapunov reward system

if $\mathscr{V}(\vec{0}) = 0$ and $\dot{\mathscr{V}}(\vec{0}) = 0$ **then**
  R_L=5 (where R_L denotes the Lyapunov reward)
else if $\mathscr{V}(\vec{x}(t)) > 0$ and $\dot{\mathscr{V}}(\vec{x}(t)) < 0$ for all $\vec{x}(t) \neq \vec{0}$ **then**
  R_L=1
else
  R_L=0
end if

In the simulation on the cart-pole inverted pendulum, $\vec{x}(t)$ is defined as $[x_1(t)\ x_2(t)]^T$. According to the Lyapunov stability concept [23], [24], $\mathscr{V}(\vec{x}(t)) = \vec{x}(t)^T \mathscr{P}\vec{x}(t)$, $\mathscr{P} \in \Re^{n\times n}$ is a symmetric constant matrix to be determined satisfying $\mathscr{P} = \mathscr{P}^T > 0$, $n > 0$. Lyapunov reward will be set as 5 if the conditions that $\mathscr{V}(\vec{0}) = 0$ and $\dot{\mathscr{V}}(\vec{0}) = 0$ are matched, which denotes the equilibrium point is reached. Before reaching the equilibrium point, Lyapunov reward will be 1 if the control of the cart-pole inverted pendulum can follow the asymptotically stable law, otherwise no incentive will be given.

2) Fuzzy reward system: The design of the fuzzy reward system is based on the FIS and the knowledge of the cart-pole inverted pendulum. The first step is to implement the fuzzification process. We describe the operating scopes of $x_1(t)$ and $x_2(t)$ of the cart-pole inverted pendulum by 6 fuzzy sets respectively, which are BN (Big Negative), MN (Medium Negative), SN (Small Negative), SP (Small Positive), MP (Medium Positive) and BP (Big Positive). The MFs concerning $x_1(t)$ and $x_2(t)$ are illustrated in Fig. 1.



(a) MFs of $x_1(t)$. (b) MFs of $x_2(t)$.

Fig. 1. Antecedent MFs.

The consequence (C) in every fuzzy rule is the sum of an ordinary value (C^O) and a bonus value (C^B) which are both constants defined by designers. The ordinary values are set according to the relationships between angular displacement and angular velocity. The bonus value is utilized to prevent valleys from appearing between peaks of the reward and guarantee the growth of the maximum reward when angular displacement is approaching 0.

The rules of $x_1(t)$, $x_2(t)$ and the corresponding outcomes are summarized in Table I.

TABLE I
THE RULES OF ANTECEDENT CONDITIONS AND CONSEQUENCES OF THE CART-POLE INVERTED PENDULUM.

C / $x_2(t)$ \ $x_1(t)$	BN	MN	SN	SP	MP	BP
BN	$C^O(0)+C^B(0)$	$C^O(\frac{1}{3})+C^B(0)$	$C^O(\frac{2}{3})+C^B(0)$	$C^O(1)+C^B(0)$	$C^O(\frac{4}{3})+C^B(0)$	$C^O(1)+C^B(0)$
MN	$C^O(\frac{1}{3})+C^B(0.5)$	$C^O(\frac{2}{3})+C^B(0.5)$	$C^O(1)+C^B(0.5)$	$C^O(\frac{4}{3})+C^B(0.5)$	$C^O(\frac{4}{3})+C^B(0.5)$	$C^O(\frac{4}{3})+C^B(0.5)$
SN	$C^O(\frac{2}{3})+C^B(1)$	$C^O(1)+C^B(1)$	$C^O(\frac{4}{3})+C^B(1)$	$C^O(\frac{5}{3})+C^B(1)$	$C^O(\frac{4}{3})+C^B(1)$	$C^O(1)+C^B(1)$
SP	$C^O(1)+C^B(1)$	$C^O(\frac{4}{3})+C^B(1)$	$C^O(\frac{5}{3})+C^B(1)$	$C^O(\frac{4}{3})+C^B(1)$	$C^O(1)+C^B(1)$	$C^O(\frac{2}{3})+C^B(1)$
MP	$C^O(\frac{4}{3})+C^B(0.5)$	$C^O(\frac{5}{3})+C^B(0.5)$	$C^O(\frac{4}{3})+C^B(0.5)$	$C^O(1)+C^B(0.5)$	$C^O(\frac{2}{3})+C^B(0.5)$	$C^O(\frac{1}{3})+C^B(0.5)$
BP	$C^O(1)+C^B(0)$	$C^O(\frac{4}{3})+C^B(0)$	$C^O(1)+C^B(0)$	$C^O(\frac{2}{3})+C^B(0)$	$C^O(\frac{1}{3})+C^B(0)$	$C^O(0)+C^B(0)$

The minimum C^O is 0 and the maximum C^O is $\frac{5}{3}$, the difference of C^O is $\frac{1}{3}$. The maximum C^O is supposed to occur in the rule where $x_1(t)$ and $x_2(t)$ have the most appropriate relationship. The exceptional circumstance is that big velocity is not considered as the most proper angular velocity for big angle, otherwise $x_2(t)$ will more possibly breach the constraint. Nevertheless, if larger angular velocity can help to reach the control purpose better, the larger velocity will be preferred to learn because $\hat{A}_t$ is not only based on immediate reward but also future rewards [20]. The minimum C^B is 0 and the maximum C^B is 1, the difference of C^B is 0.5. When absolute $x_1(t)$ is smaller, the C^B will become larger.

The defuzzified reward is defined as the following:

$$R_F = \frac{\sum_{i=1}^{36} C_i \min(\mu_i(x_1(t)), \mu_i(x_2(t)))}{\sum_{i=1}^{36} \min(\mu_i(x_1(t)), \mu_i(x_2(t)))}, \tag{5}$$

where R_F denotes the fuzzy reward and μ_i denotes the membership degree.

The ultimate outcomes after the defuzzification are shown in Fig. 2.

3) Lyapunov-fuzzy reward system: The Lyapunov reward system and fuzzy reward system are connected by weights

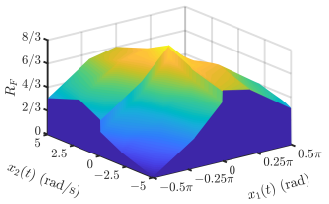


Fig. 2. Overall surface of the fuzzy reward.

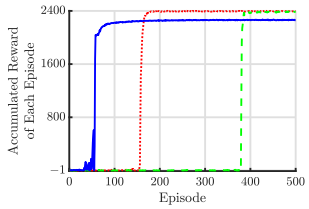


Fig. 3. Learning curves. (Solid line in blue: PPO (APLR) with the Lyapunov-fuzzy reward system. Dotted line in red: PPO (APLR) with the existing reward scheme. Dashed line in green: PPO with the existing reward scheme.)

used to balance the impacts of Lyapunov reward and fuzzy reward in the comprehensive reward system, which is shown as the following equation:

$$R = W_L R_L + W_F R_F, \tag{6}$$

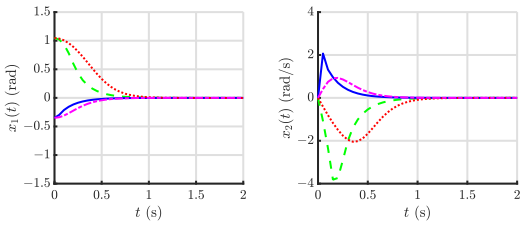
where W_L denotes the weight of the Lyapunov reward and W_F denotes the weight of the fuzzy reward, which are set as 0.1 and $\frac{3}{8}$ respectively in this paper. Lyapunov-fuzzy reward system takes effect only when the boundary of any variable is not exceeded, otherwise penalty -1 will be given.

IV. Simulation

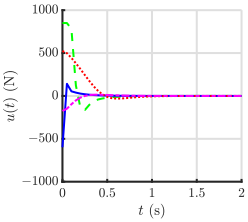
A. The settings of simulation parameters

In this section, we compare learning curves of the PPO (APLR), PPO under the guidance of the existing reward scheme, as well as the simulation results of the PPO (APLR) which is guided by the Lyapunov-fuzzy reward system. The inputs of three RL approaches are $x_1(t)$ and $x_2(t)$, the output is the force applied to the cart. In the learning process, the initial state for each episode is set as follows, $x_1(t)$ is a random value from $[-\frac{\pi}{3}, \frac{\pi}{3}]$ and $x_2(t)$ is 0. The operating scopes of $x_1(t)$ and $x_2(t)$ are $[-\frac{\pi}{2}, \frac{\pi}{2}]$ and $[-5, 5]$, respectively. Besides, the range of the force is $[-850, 850]$. Units of the hidden layer in the value network and policy network are both 100. The activation functions of the value network and policy network are rectified linear unit (ReLU) and tanh function, respectively. The value learning rate for the update of the value network is 0.001 and the policy learning rate for the update of the policy network is 0.0001. η_1 to η_7 for the APRL are 2, 2, 1.5, 1, 1, 1.5, 0.0001, respectively. The size of the mini-batch is 32 and epochs for the multiple update are 10. The sampling interval is 0.05s, time steps are 2400 and there are 500 episodes. $\mathscr{P}$ of the Lyapunov reward system is chosen as $\begin{bmatrix} 11 & 0 \\ 0 & 1 \end{bmatrix}$. PPO in [13] only has one whole function, so we set learning rate 0.0005. The existing reward scheme is defined as [25], which is employed for comparison purposes: reward is 1 when $|x_1(t)| < \frac{6\pi}{180}$rad, if the constraint is violated, -1 will be given, otherwise the reward will be 0.

In addition, we compare the control performance of the conventional proportional-integral-derivative (PID) controller with the policy learned by the PPO (APLR) with the Lyapunov-fuzzy reward system. Proportional, integral and derivative gains are -500, -1 and -100. These gains are acquired by trial-and-error, which can obtain the best results (with the smallest fluctuation and shortest settling time) in this case .



(a) Response of $x_1(t)$. (b) Response of $x_2(t)$.



(c) Control signal $u(t)$.

Fig. 4. Response curves. (Dashed line in green: policy learned by the PPO (APLR) with the Lyapunov-fuzzy reward system ($\frac{60\pi}{180}$, 0). Dotted line in red: conventional PID controller ($\frac{60\pi}{180}$, 0). Solid line in blue: policy learned by the PPO (APLR) with the Lyapunov-fuzzy reward system ($-\frac{20\pi}{180}$, 0). Dash-dot line in magenta: conventional PID controller ($-\frac{20\pi}{180}$, 0).)

B. Simulation results

Fig. 3 shows the learning curves of three RL methods. Although the learning curve of the PPO with the existing reward scheme can converge, the convergence episode is around 390 far more than others. The convergence speed of the PPO (APLR) with the existing reward scheme is faster than the PPO with the existing reward scheme, which requires about 180 episodes. Nevertheless, the convergence speed of the PPO (APLR) with the Lyapunov-fuzzy reward system is the fastest in all methods, which just needs around 100 episodes. The maximum accumulated reward of the existing reward scheme is higher than the Lyapunov-fuzzy reward system due to the loose reward setting of the existing reward scheme. However, it is also the reason why the existing reward scheme is inefficient. It is shown in Fig. 3 that the proposed PPO (APLR) with the Lyapunov-fuzzy reward system can learn the policy with the least number of episodes among all RL methods.

We compared the response curves of the cart-pole inverted pendulum controlled by the conventional PID controller with the policy learned by the PPO (APLR) with the Lyapunov-fuzzy reward system, which is illustrated in Fig. 4. In this simulation experiment, the settling time of $x_1(t)$ is set as the

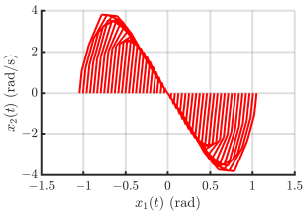


Fig. 5. Phase portraits of $x_1(t)$ and $x_2(t)$ under different initial conditions.

time demanded for the response curve of $x_1(t)$ to arrive at and remain within a scope of $[-0.1, 0.1]$. In terms of the initial state $(\frac{60\pi}{180}, 0)$, the settling time of $x_1(t)$ under the control of the conventional PID controller is about 0.7s which is more than the learned policy which is around 0.5s. In terms of the initial state $(-\frac{20\pi}{180}, 0)$, the settling time of $x_1(t)$ under the control of the conventional PID controller is about 0.35s which is still more than the learned policy which is around 0.25s. In two initial conditions, response curves of $x_2(t)$ under the control of the conventional PID controller and learned policy are all within the operating scope.

In addition, we investigated the phase portraits of $x_1(t)$ and $x_2(t)$. The range of initial $x_1(t)$ is $[-\frac{60\pi}{180}, \frac{60\pi}{180}]$, and initial $x_2(t)$ is 0. As shown in Fig. 5, the phase flow with any initial state condition in the setting range has the tendency to reach the origin. Furthermore, $x_1(t)$ and $x_2(t)$ never exceed the operating scopes, which shows that the constraints are not violated.

V. Conclusion

A reinforcement learning-based control approach is proposed to handle the benchmark control problem of the cart-pole inverted pendulum under the guidance of the Lyapunov-fuzzy reward system. In this paper, APLR is proposed to improve the learning stability of the PPO by limiting effects of negative or large advantages. In addition, Lyapunov-fuzzy reward system is proposed to guide the learning process more efficiently, and the resultant comprehensive reward system can better evaluate the generated action.

Learning curves of three RL methods have clarified the proposed PPO (APLR) with the Lyapunov-fuzzy reward system is the most efficient method. As seen from the response curves, the proposed PPO (APLR) with the Lyapunov-fuzzy reward system is able to learn the policy which can achieve the control objective quicker, and the control performance of the learned policy is better than the conventional PID controller in terms of the settling time of $x_1(t)$. Moreover, the phase portraits demonstrate that the learned policy can achieve the control objective without violating constraints.

References

[1] B. Xiao, H. K. Lam, and H. Li, "Stabilization of interval type-2 polynomial-fuzzy-model-based control systems," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 1, pp. 205–217, 2017.

[2] Y. Yu, Q. Shi, and H. K. Lam, "Fuzzy sliding mode control of a continuum manipulator," in *Proc. of 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2018, pp. 2057–2062.

[3] J. Zhai, "Dynamic output-feedback control for nonlinear time-delay systems and applications to chemical reactor systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, pp. 1–1, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8599076

[4] S. Kim and C. K. Ahn, "Auto-tuner based controller for quadcopter attitude tracking applications," *IEEE Transactions on Circuits and Systems II: Express Briefs*, pp. 1–1, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8630662

[5] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[6] D. Liu, X. Yang, D. Wang, and Q. Wei, "Reinforcement-learning-based robust controller design for continuous-time uncertain nonlinear systems subject to input constraints," *IEEE Transactions on Cybernetics*, vol. 45, no. 7, pp. 1372–1385, 2015.

[7] C. J. C. H. Watkins, "Learning from delayed rewards," Ph.D. dissertation, King's College, Cambridge, 1989.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[9] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[10] 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," *arXiv preprint arXiv:1509.02971*, 2015. [Online]. Available: https://arxiv.org/abs/1509.02971

[11] S. Choi, T. P. Le, Q. D. Nguyen, M. A. Layek, S. Lee, and T. Chung, "Toward self-driving bicycles using state-of-the-art deep reinforcement learning algorithms," *Symmetry*, vol. 11, no. 2, 2019. [Online]. Available: https://www.mdpi.com/2073-8994/11/2/290

[12] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. of International Conference on Machine Learning*, 2015, pp. 1889–1897.

[13] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017. [Online]. Available: https://arxiv.org/abs/1707.06347

[14] N. D. Nguyen, T. Nguyen, and S. Nahavandi, "System design perspective for human-level agents using deep reinforcement learning: A survey," *IEEE Access*, vol. 5, pp. 27 091–27 102, 2017.

[15] J. Kim, H. Lim, C. Kim, M. Kim, Y. Hong, and Y. Han, "Imitation reinforcement learning-based remote rotary inverted pendulum control in openflow network," *IEEE Access*, vol. 7, pp. 36 682–36 690, 2019.

[16] P. Hämäläinen, A. Babadi, X. Ma, and J. Lehtinen, "PPO-CMA: proximal policy optimization with covariance matrix adaptation," *arXiv preprint arXiv:1810.02541*, 2018. [Online]. Available: http://arxiv.org/abs/1810.02541

[17] M. Pirotta, M. Restelli, and L. Bascetta, "Adaptive step-size for policy gradient methods," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 1394–1402.

[18] J. Xu, Z. Hou, W. Wang, B. Xu, K. Zhang, and K. Chen, "Feedback deep deterministic policy gradient with fuzzy reward for robotic multiple peg-in-hole assembly tasks," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1658–1667, 2019.

[19] P. Kofinas, G. Vouros, and A. I. Dounis, "Energy management in solar microgrid via reinforcement learning using fuzzy reward," *Advances in Building Energy Research*, vol. 12, no. 1, pp. 97–115, 2018.

[20] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," *arXiv preprint arXiv:1506.02438*, 2015. [Online]. Available: https://arxiv.org/abs/1506.02438

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. [Online]. Available: https://arxiv.org/abs/1412.6980

[22] B. Xiao, H. K. Lam, Y. Yu, and Y. Li, "Sampled-data output-feedback tracking control for interval type-2 polynomial fuzzy systems," *IEEE Transactions on Fuzzy Systems*, pp. 1–1, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8673900

[23] H. K. Lam, "A review on stability analysis of continuous-time fuzzy-model-based control systems: From membership-function-independent to membership-function-dependent analysis," *Engineering Applications of Artificial Intelligence*, vol. 67, pp. 390–408, 2018.

[24] J. Zhai, "Adaptive control for nonlinear systems with uncertain output function and unknown homogenous growth rate," *IEEE Transactions on Circuits and Systems II: Express Briefs*, pp. 1–1, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8822386

[25] Q. Shi, H. K. Lam, B. Xiao, and S. H. Tsai, "Adaptive PID controller based on Q-learning algorithm," *CAAI Transactions on Intelligence Technology*, vol. 3, no. 4, pp. 235–244, 2018.