Off-Policy Reinforcement Learning for Partially Unknown Nonzero-Sum Games

Qichao Zhang[1,2], Dongbin Zhao[1,2] ⋆, and Sibo Zhang[3]

1. The state Key Laboratory of Management and Control for Complex Systems,
Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China,
2. University of Chinese Academy of Sciences, Beijing, 100049, China,
3. University of Illinois Urbana-Champaign, Champaign, IL, 61801, USA.

Abstract. In this paper, the optimal control problem of nonzero-sum (NZS) games with partially unknown dynamics is investigated. The off-policy reinforcement learning (RL) method is proposed to approximate the solution of the coupled Hamilton-Jacobi (HJ) equations. A single critic network structure for each player is constructed using neural network (NN) technique. To improve the applicability of the off-policy RL method, the tuning laws of critic weights are designed based on the offline learning and online learning methods, respectively. The simulation study demonstrates the effectiveness of the proposed algorithms.

Keywords: Internal reinforcement learning, nonzero-sum games, optimal control, partially unknown dynamics, offline and online learning

1 Introduction

The game theory for continuous-time systems, which is called as differential game[1], have received spreading attention in the optimal control field[2]. In general, differential games can be divided into three categories: zero-sum (ZS) games[3], nonzero-sum (NZS) games[4] and fully cooperative (FC) games[5]. The players in the NZS game can be either cooperative or competitive to maximize their own interest. In order to obtain the optimal controllers, it is desired to obtain the Nash equilibrium[6] by solving the HJ equations. However, it is difficult to obtain the analytic solution of the HJ equations for nonlinear systems.

To approach the Nash equilibrium of the differential game, many model-based or model-free reinforcement learning (RL) and adaptive dynamic programming (ADP) algorithms have been presented[7, 8]. For the model-based RL which requires full knowledge of system dynamics, an online synchronous policy iteration algorithm with actor-critic NN structure was proposed in [9]. For the model-based RL using partially knowledge of system dynamics, the internal RL (IRL)[10] is the main technique to relax the knowledge of the internal dynamics.

⋆ This research is supported by National Natural Science Foundation of China (NSFC) under Grants No. 61573353, No. 61533017, by the National Key Research and Development Plan under Grants 2016YFB0101000. (E-mail: dongbin.zhao@ia.ac.cn.)

For the partially unknown nonlinear NZS games, a concurrent learning-based actor-critic-identifier (ACI) structure was presented in [11], where the unknown internal dynamics was identified using NN. Song *et al.*[12] investigated the off-policy IRL algorithm with actor-critic structure for completely unknown NZS games, where the convergence analysis of the proposed algorithm was proved.

To the best of our knowledge, there are still no IRL algorithms for general NZS games with partially unknown dynamics. Motivated by [12] and [13], a novel off-policy IRL algorithm with single-critic structure is presented to solve the coupled HJ equations in this paper. Then, the NN-based offline iterative learning and online iterative learning algorithms are employed for the off-policy IRL, respectively. Simulation results show the effectiveness of the proposed scheme.

2 Problem Statement

Consider the N-player nonzero-sum differential games given by

$$\dot{x} = f\left(x(t)\right) + \sum_{j=1}^{N} g_j\left(x(t)\right) u_j(t), \tag{1}$$

where $x \in R^n$ is the state, $u_j \in R^{m_j}$ is the control input, $f(\cdot) \in R^n$, $g_j(\cdot) \in R^{n\times m_j}$ are smooth nonlinear dynamics. $f(\cdot)$ is Lipschitz continuous on a compact set $\Omega \subseteq R^n$ with $f(0) = 0$. In this paper, the internal system dynamics $f(x)$ is assumed to be unknown. Define the set of players as $\mathbf{N} = \{1, ..., N\}$, and the supplementary set of player i as $u_{-i} = \{u_j \mid j \in \{1, ..., i-1, i+1, ..., N\}\}$.

For the admissible policy u_i defined in [9], the system (1) is stabilized on the compact set Ω, denoted by $u_i \in \Phi(\Omega)$. Define the value functions for any N-tuple of admissible strategies $u_i(x), i \in \mathbf{N}$ as

$$\begin{aligned} V_i\left(x, u_i, u_{-i}\right) &= \int_t^\infty \left(Q_i\left(x(\tau)\right) + \sum_{j=1}^{N} u_j^T(\tau) R_{ij} u_j(\tau)\right) d\tau \\ &= \int_t^\infty r_i\left(x(\tau), u_i(\tau), u_{-i}(\tau)\right) d\tau,\ i \in \mathbf{N}, \end{aligned} \tag{2}$$

where $Q_i(x) = x^T Q_i x$, $Q_i \geq 0$ and $R_{ii} \geq 0$ are positive symmetric matrices, and $R_{ij} > 0$ are positive semidefinite symmetric. For the nonzero-sum differential games, it aims to find an Nash equilibrium defined as follows.

Definition 1 (Nash Equilibrium: An N-tuple of admissible policies $\{u_i^*, u_{-i}^*\}$ is said to constitute a Nash equilibrium solution for an N-player nonzero-sum game, if $J_i^*(u_1^*, ..., u_i^*, ..., u_N^*) \leq J_i(u_1^*, ..., u_i, ..., u_N^*), i \in \mathbf{N}$.

To obtain the Nash equilibrium of the nonzero-sum games, we should solve the so-called HJ equations, which is described as follows.

$$\begin{aligned} &Q_i\left(x\right) + (\nabla V_i^*)^T f(x) - \tfrac{1}{2}(\nabla V_i^*)^T \sum_{j=1}^{N} g_j(x) R_{jj}^{-1} g_j^T(x) \\ &\times(\nabla V_j^*) + \tfrac{1}{4}\sum_{j=1}^{N} (\nabla V_j^*)^T g_j(x) R_{jj}^{-1} R_{ij} R_{jj}^{-1} g_j^T(x) \nabla V_j^* = 0 \end{aligned} \tag{3}$$

where $V_i^*(x)$ is the optimal value function with $V_i^*(x) \geq 0, V_i(0) = 0$, and $\nabla V_i = \frac{\partial V_i(x)}{\partial x}$. The optimal state feedback control policy for each player i is $u_i^*(x) = -\frac{1}{2} R_{ii}^{-1} g_i^T(x) \nabla V_i^*, i \in \mathbf{N}$.

3 Off-Policy IRL for Partially Unknown NZS Games

3.1 Off-Policy IRL Method

With an arbitrary admissible control policy $u_j \in \Phi(\Omega), j \in \mathbf{N}$, the system (1) can be rewritten as

$$\dot{x} = f(x) + \sum_{j=1}^{N} g_j(x)(u_j - u_j^k) + \sum_{j=1}^{N} g_j(x) u_j^k, \tag{4}$$

with $u_i^{k+1}(x) = -\frac{1}{2} R_{ii}^{-1} g_i^T(x) \nabla V_i^{k+1}(x)$. The derivative of $V_i^{k+1}(x)$ with respect to time along the system trajectory (4) equals to

$$\begin{aligned} \frac{dV_i^{k+1}(x)}{dt} =& (\nabla V_i^{k+1})^T (f + \sum_{j=1}^{N} g_j(x) u_j^k) + (\nabla V_i^{k+1})^T \sum_{j=1}^{N} g_j(x)(u_j - u_j^k) \\ =& - r_i(x, u_i^k, u_{-i}^k) + (\nabla V_i^{k+1})^T \sum_{j=1}^{N} g_j(x)(u_j - u_j^k). \end{aligned} \tag{5}$$

Based on the IRL, we have the integral form of equation (5) along the time interval $[t, t + \Delta t]$

$$\begin{aligned} & V_i^{k+1}(x(t)) - V_i^{k+1}(x(t + \Delta t)) \\ & + \int_t^{t+\Delta t} \Big(\nabla V_i^{k+1}(x(\tau)) \Big)^T \sum_{j=1}^{N} g_j(x(\tau)) \big(u_j(\tau) - u_j^k(\tau) \big) d\tau. \end{aligned} \tag{6}$$

3.2 NN-Based Off-Policy IRL Algorithm

In this subsection, the NN approximation is introduced to solve (6) for $V_i^{k+1}(x)$ based on a single-critic network structure. The value function is described as

$$V_i{}^k(x) = w_{i,k}^T \phi_i(x) + \varepsilon_{i,k}, i \in \mathbf{N}, \tag{7}$$

where $\phi_i : R^n \to R^{K_{i,k}}$ is the activation functions, $w_{i,k} \in R^{K_{i,k}}$ is the unknown coefficient vector with $K_{i,k}$ the numbers of hidden neurons, $\varepsilon_{i,k}$ is the reconstruction error with appropriate dimensions.

Based on (7), the iteration equation (6) can be rewritten as

$$\begin{aligned} \zeta_{i,k+1}(x(t)) =& (\phi_i(x + \Delta t) - \phi_i(x))^T w_{i,k+1} - \int_t^{t+\Delta t} \sum_{j=1}^{N} \Big(g_j(x)(u_j(\tau) - u_j^k(\tau)) \Big)^T \\ & \times \nabla \phi_i^T(x) w_{i,k+1} d\tau + \int_t^{t+\Delta t} Q_i(x) + \sum_{j=1}^{N} \Big((u_j^k(\tau))^T R_{ij} u_j^k(\tau) \Big) d\tau \end{aligned} \tag{8}$$

Let $\hat{w}_{i,k}$ be the estimations of the unknown coefficients $w_{i,k}$. The actual output of the NN approximation can be presented as $\hat{V_i}^k(x) = \hat{w}_{i,k}^T \phi_i(x)$. Then, we can obtain the approximated control policies $\hat{u}_i^k(x) = -\frac{1}{2} R_{ii}^{-1} g_i^T(x) \nabla \phi_i^T(x) \hat{w}_{i,k}$. Using $\hat{V_i}^{k+1}(x)$ instead of $V_i^{k+1}(x)$ in equation (6), the residual error is given by

$$\begin{aligned}
&e_i^{k+1}(x(\tau), u_i(\tau), u_{-i}(\tau)) \triangleq e_i^{k+1}(t)\\
&= \big(\phi_i\big(x(t)\big) - \phi_i\big(x(t+\Delta t)\big)\big)^T \hat{w}_{i,k+1} + \int_t^{t+\Delta t} \sum_{j=1}^{N} \Big(g_j(x)(u_j(\tau) - u_j^k(\tau))\Big)^T\\
&\quad \nabla\phi_i^T(x) \hat{w}_{i,k+1} d\tau - \int_t^{t+\Delta t} Q_i(x) d\tau - \int_t^{t+\Delta t} \sum_{j=1}^{N} \big((u_j^k(\tau))^T R_{ij} u_j^k(\tau)\big) d\tau.
\end{aligned} \tag{9}$$

Let

$$\begin{aligned}
&\rho_i\big(x(t), u_i(t), u_{-i}(t)\big)\\
&\triangleq \big(\phi_i\big(x(t)\big) - \phi_i\big(x(t+\Delta t)\big)\big)^T + \int_t^{t+\Delta t} \sum_{j=1}^{N} \Big(g_j(x)(u_j(\tau) - u_j^k(\tau))\Big)^T \nabla\phi_i^T(x) d\tau,\\
&\pi_i(x(t)) \triangleq \int_t^{t+\Delta t} Q_i(x) d\tau + \sum_{j=1}^{N} \big((u_j^k(\tau))^T R_{ij} u_j^k(\tau)\big) d\tau.
\end{aligned} \tag{10}$$

For notation simplicity, define

$$\begin{aligned}
&D_{i,j}(x) \triangleq \nabla\phi_j(x) g_j(x) R_{jj}^{-1} g_j^T(x) \nabla\phi_i^T(x),\\
&E_{i,j}(x) \triangleq \nabla\phi_j(x) g_j(x) R_{jj}^{-1} R_{ij} R_{jj}^{-1} g_j^T(x) \nabla\phi_j^T(x),\\
&\eta_1(x(t)) \triangleq \big(\phi_i\big(x(t)\big) - \phi_i\big(x(t+\Delta t)\big)\big)^T,\\
&\eta_2(x(t), u_i, u_{-i}) \triangleq \int_t^{t+\Delta t} \big(\sum_{j=1}^{N} u_j^T(\tau) g_j^T(x)\big) \nabla\phi_i^T(x) d\tau,
\end{aligned}$$

$$\eta_3(x(t)) \triangleq \begin{bmatrix} \int_t^{t+\Delta t} D_{i1}(x) d\tau \\ \vdots \\ \int_t^{t+\Delta t} D_{iN}(x) d\tau \end{bmatrix}, \eta_4(x(t)) \triangleq \begin{bmatrix} \int_t^{t+\Delta t} E_{i,1}(x) d\tau & 0 & 0 \\ 0 & \ddots & \vdots \\ 0 & \cdots & \int_t^{t+\Delta t} E_{i,N}(x) d\tau \end{bmatrix},$$

$$\eta_5(x(t)) \triangleq \int_t^{t+\Delta t} Q_i(x) d\tau.$$

Next, we have

$$\begin{aligned}
\rho_i\big(x(t), u_i(t), u_{-i}(t)\big) =& \eta_1(x(t)) + \eta_2(x(t), u_i, u_{-i}) + \frac{1}{2} \hat{W}_k^T \eta_3(x(t)),\\
\pi_i(x(t)) =& \frac{1}{4} \hat{W}_k^T \eta_4(x(t)) \hat{W}_k + \eta_5(x(t)),
\end{aligned}$$

where $\hat{W}_k = [\hat{w}_{1,k}^T, ..., \hat{w}_{N,k}^T]^T$.

Then, (9) can be rewritten as

$$e_i^{k+1}(t) = \rho_i\big(x(t), u_i(t), u_{-i}(t)\big)\hat{w}_{i,k+1} - \pi_i(x(t)). \qquad (11)$$

Note that the equation (11) is the key for the off-policy IRL algorithm for NZS games with partially unknown dynamics.

4 Offline Iterative Learning Algorithm

For the designed offline iterative learning algorithm, critic weights are updated based on least-square (LS) scheme. Define a strictly increasing time sequence $\{t_m\}_{m=0}^q$ for a large time interval with the number of collected samples $q > 0$. Let the sample set $M_i = \{(x_m, u_{i,m}, u_{-i,m})\}_{m=0}^q$. In fact, each time interval $[t_m, t_{m+1}]$ is equivalent to the one $[t, t + \Delta t]$ in (9).

Define $\rho_{i,m} = \rho_i(x_m, u_{i,m}, u_{-i,m})$ and $\pi_{i,m} = \pi_i(x_m)$. To guarantee the convergence of $\hat{w}_{i,k+1}$, the persistency of excitation (PE) assumption which is usually needed in adaptive control algorithms is given.

Assumption 1: Let the signal $\rho_{i,m}$ be persistently existed, that is there exist $q_0 > 0$ and $\delta > 0$ such that for all $q \leq q_0$, we have $\frac{1}{q}\sum_{k=0}^{q-1}\rho_{i,m}\rho_{i,m}^T \geq \delta I_{i,m}$, where $I_{i,m}$ is the identity matrix of appropriate dimensions.

According to the LS principle, it is desired to determine the estimated weighting function vector $\hat{w}_{i,k+1}$ by minimizing $\min\limits_{\hat{w}_{i,k+1}} \frac{1}{2}(e_{i,m}^{k+1})^T e_{i,m}^{k+1}$. According to the Monte Carlo integration method in [13], the solution to this LS problem yields

$$\hat{w}_{i,k+1} = [P_i^T P_i]^{-1} P_i^T \varPi_i, \qquad (12)$$

where $P_i = [\rho_{i,0}, ..., \rho_{i,q-1}]^T$, $\varPi_i = [\pi_{i,0}, ..., \pi_{i,q-1}]^T$.

Based on the update rule (12), the NN-based offline iterative learning algorithm for the off-policy IRL is presented in Algorithm 1. Note that it can be divided into two phases, i.e. the measurement phase of step 1 to collect the system data and the offline learning phase of step 2-4 to approximate the ideal critic weights.

Algorithm 1 (Offline iterative learning for NZS games)

```
1: Select the initial admissible control policies $\{u_i, u_{-i}\}$. Collect real system data $(x_m, u_i, u_{-i})$ for sample set $M$, then compute $\eta_1(x_m), \eta_2(x_m, u_i, u_{-i}), \eta_3(x_m), \eta_4(x_m)$ and $\eta_5(x_m)$;
2: Select the initial critic NN weight vector $\hat{w}_{i,0}$ for each player. Let $k = 0$;
3: Compute $P_i$ and $\varPi_i$, and update $\hat{w}_{i,k+1}$ for each player using (12);
4: Let $k = k + 1$, if $\|\hat{w}_{i,k+1} - \hat{w}_{i,k}\| \leq \epsilon$ ($\epsilon$ is a small positive number to stop the process with a finite number of iterations), else go back to Step 3 and continue.
```

5 Online Iterative Learning Algorithm

For the online iterative learning algorithm, the gradient descent method is utilisable to update the weights of critic NNs. According to the ER technique, the past system data is also improvable to approach the critic NNs' weights. As the critic weights are updated continuously in the online learning algorithm, we use w_i, e_i, K_i to replace $w_{i,k+1}, e_i^{k+1}, K_{i,k+1}$, respectively.

Based on (11), define the residual errors at the past internal $[t_d, t_{d+1}]$ as $e_i(t_d) = \rho_i\big(t_d\big)\hat{w}_i + \pi_i(t_d)$. It is desired to minimize the following square error $E_i = \frac{1}{2}(e_i(t))^T e_i(t) + \frac{1}{2}\sum\limits_{d=1}^{l}(e_i(t_d))^T e_i(t_d)$.

Condition 1: Let $D_i = [\rho_i(t_d), \rho_i(t_{d+1}), ..., \rho_i(t_{d+l})]$ be the recorded data corresponding to each critic NN's weights. Then D_i contains as many linearly independent elements as the number of corresponding critic NNs hidden neurons, i.e., $rank(D_i) = K_i$.

The adaptation law for the critic weights based on gradient descent method and ER is given by

$$\dot{\hat{w}}_i = -\alpha_i\left[\frac{\rho_i^T(t)}{\big(1+\rho_i^T(t)\rho_i(t)\big)^2}\left(\rho_i\hat{w}_i + \pi_i(t)\right) + \sum_{d=1}^{l}\frac{\rho_i^T(t_d)}{\big(1+\rho_i^T(t_d)\rho_i(t_d)\big)^2}\left(\rho_i(t_d)\hat{w}_i + \pi_i(t_d)\right)\right] \tag{13}$$

6 Simulation Study

Consider the following two-player affine nonlinear nonzero-sum game system [9]:

$$\dot{x} = f(x) + g(x)u + k(x)w \tag{14}$$

where

$$\begin{aligned} f(x) &= \begin{bmatrix} x_2 \\ -x_2 - 0.5x_1 + 0.25x_2(\cos(2x_1)+2)^2 \\ +0.25x_2(\sin(2x_1)+2)^2 \end{bmatrix}. \\ g(x) &= \begin{bmatrix} 0 \\ \cos(2x_1)+2 \end{bmatrix},\ k(x) = \begin{bmatrix} 0 \\ \sin\left(4x_1^2\right)+2 \end{bmatrix} \end{aligned}$$

$x = [x_1, x_2]^T \in R^2$ and $u, w \in R$ are state and control variables, respectively.

Select $Q_1(x) = 2x^T x$, $Q_2(x) = x^T x$, $R_{11} = R_{12} = 2I$, and $R_{21} = R_{22} = I$, where I is an identity matrix. The optimal value functions are $V_1^*(x) = 0.5x_1^2 + x_2^2$ and $V_2^*(x) = 0.25x_1^2 + 0.5x_2^2$. For the offline and online iterative learning, the activation functions of the critic NNs of two players are selected as $\phi_{c1}(x) = \phi_{c2}(x) = [x_1^2\ x_1x_2\ x_2^2]^T$. Thus, the ideal weights of critic NNs are $w_{c1} = [0.5\ 0.0\ 1.0]^T; w_{c2} = [0.25\ 0.0\ 0.5]^T$.

6.1 Offline Iterative Learning

The initial state vector is chosen as $x_0 = [2, -2]^T$. Set the the convergence threshold $\varepsilon = 10^{-6}$. The integral time interval is chosen as 0.1s. Let the length index $q = 200$, which means the online data collection phase is terminated after 20s. The convergence curves of w_{ci} are shown in Fig. 1. The critic NNs weights $w_{ci,k+1}$ converge to $\hat{w}_{c1} = [0.4956\ 0.098\ 1.0613]^T; \hat{w}_{c2} = [0.2356\ 0.063\ 0.5223]^T$ at the fourth iteration, which are nearly the ideal values above. Compared with [9], the knowledge of internal dynamics is relaxed in the proposed offline algorithm.

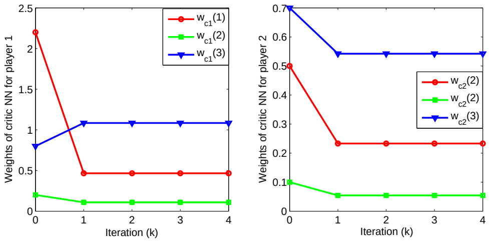

Fig. 1. The weights w_{c1} and w_{c2} of critic NNs for player 1 and 2

6.2 Online Iterative Learning

Select the same activation functions of critic NNs. Set the initial state vector as $x_0 = [1, -1]^T$. The experience set size selects $l = 10$ and the integral time interval is also 0.1s. Note that we remove the initial probing control inputs at 80s. The learning rates $\alpha_1 = 2, \alpha_2 = 4$. The final critic weights for player 1 and player 2 are $\hat{w}_{c1} = [0.5156\ 0.0114\ 0.9906]^T; \hat{w}_{c2} = [0.2592\ 0.0111\ 0.4901]^T$, which are shown in Fig. 2. The simulation results prove the effectiveness of the proposed online off-policy method.

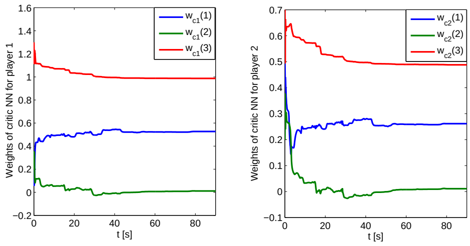

Fig. 2. The weights w_{c1} and w_{c2} of critic NNs for player 1 and 2

7 Conclusion

In this paper, we investigate the off-policy IRL technique for the nonlinear nonzero-sum games with unknown internal dynamics. To implement the proposed method, a NN-based offline and online learning with a single critic NN structure are proposed. For the online iterative learning algorithm, the ER technique is introduced to improve the convergence rate. Finally, simulation results demonstrate the effectiveness of the proposed algorithms.

References

1. A. Friedman, *Differential games.* Mineola, New York, USA: Courier Corporation, 2013.
2. Q. Zhang, D. Zhao, and Y. Zhu, “Event-triggered h_∞ control for continuous-time nonlinear system via concurrent learning,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, DOI: 10.1109/TSMC.2016.2531680, 2016.
3. Y. Zhu , D. Zhao, X. Li, “Iterative adaptive dynamic programming for solving unknown nonlinear zero-sum game based on online data,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 3, pp. 714–725, 2017.
4. A. W. Starr and Y.-C. Ho, “Nonzero-sum differential games,” *Journal of Optimization Theory and Applications*, vol. 3, no. 3, pp. 184–206, 1969.
5. Q. Zhang, D. Zhao, and Y. Zhu, “Data-driven adaptive dynamic programming for continuous-time fully cooperative games with partially constrained inputs,” *Neurocomputing*, vol. 238, pp. 377–386, 2017.
6. J. Nash, “Non-cooperative games,” *Annals of mathematics*, pp. 286–295, 1951.
7. D. Zhao , Q. Zhang, D. Wang, et al, “Experience replay for optimal control of nonzero-sum game systems with unknown dynamics,” *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 854–865, 2016.
8. Y. Zhu , D. Zhao, H. He, et al, “Event-triggered optimal control for partially unknown constrained-input systems via adaptive dynamic programming,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4101–4109, 2017.2017, 64(5): 4101-4109.
9. K. G. Vamvoudakis and F. L. Lewis, “Multi-player non-zero-sum games: online adaptive learning solution of coupled hamilton–jacobi equations,” *Automatica*, vol. 47, no. 8, pp. 1556–1569, 2011.
10. D. Vrabie and F. Lewis, “Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems,” *Neural Networks*, vol. 22, no. 3, pp. 237–246, 2009.
11. R. Kamalapurkar, J. R. Klotz, and W. E. Dixon, “Concurrent learning-based approximate feedback-nash equilibrium solution of n-player nonzero-sum differential games,” *Automatica Sinica, IEEE/CAA Journal of*, vol. 1, no. 3, pp. 239–247, 2014.
12. R. Song, F. L. Lewis, and Q. Wei, “Off-policy integral reinforcement learning method to solve nonlinear continuous-time multiplayer nonzero-sum games,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 3, pp. 704–713, 2017.
13. B. Luo, H.-N. Wu, and T. Huang, “Off-policy reinforcement learning for h_∞ control design,” *IEEE Transactions on Cybernetics*, vol. 45, no. 1, pp. 65–76, 2015.