

# IEEE TRANSACTIONS ON CYBERNETICS

A PUBLICATION OF THE IEEE SYSTEMS, MAN, AND CYBERNETICS SOCIETY

Indexed in PubMed® and MEDLINE®, products of the United States National Library of Medicine



JANUARY 2015

VOLUME 45

NUMBER 1

ITCEB8

(ISSN 2168-2267)

---

## REGULAR PAPERS

Automatic Programming via Iterated Local Search for Dynamic Job Shop Scheduling .....	1
.....S. Nguyen, M. Zhang, M. Johnston, and K. C. Tan	
Joint Local and Global Consistency on Interdocument and Interword Relationships for Co-Clustering .....	15
.....B.-K. Bao, W. Min, T. Li, and C. Xu	
Optimal Classification by Mixed-Initiative Nested Thresholding .....	29
.....B. Hyun, P. Kabamba, and A. Girard	
ASMiGA: An Archive-Based Steady-State Micro Genetic Algorithm .....	40
.....K. Nag, T. Pal, and N. R. Pal	
An Incremental and Distributed Inference Method for Large-Scale Ontologies Based on MapReduce Paradigm .....	53
.....B. Liu, K. Huang, J. Li, and M. Zhou	
Off-Policy Reinforcement Learning for $H_\infty$ Control Design .....	65
.....B. Luo, H.-N. Wu, and T. Huang	
Stochastic Abstract Policies: Generalizing Knowledge to Improve Reinforcement Learning .....	77
.....M. L. Koga, V. Freire, and A. H. R. Costa	
Hierarchical Ensemble of Background Models for PTZ-Based Video Surveillance .....	89
.....N. Liu, H. Wu, and L. Lin	
Bio-inspired Group Modeling and Analysis for Intruder Detection in Mobile Sensor/Robotic Networks .....	103
.....B. Fu, Y. Xiao, X. Liang, and C. L. P. Chen	
Intuitionistic Fuzzy Interaction Bonferroni Means and Its Application to Multiple Attribute Decision Making .....	116
.....Y. He, Z. He, and H. Chen	
Semi-Flocking Algorithm for Motion Control of Mobile Sensors in Large-Scale Surveillance Systems .....	129
.....S. Hosseini Semnani and O. A. Basir	
Adaptive Fuzzy Output Feedback Dynamic Surface Control of Interconnected Nonlinear Pure-Feedback Systems .....	138
.....Y. Li, S. Tong, and T. Li	

---

# Off-Policy Reinforcement Learning for $H_\infty$ Control Design

Biao Luo, Huai-Ning Wu, and Tingwen Huang

**Abstract**—The  $H_\infty$  control design problem is considered for nonlinear systems with unknown internal system model. It is known that the nonlinear  $H_\infty$  control problem can be transformed into solving the so-called Hamilton–Jacobi–Isaacs (HJI) equation, which is a nonlinear partial differential equation that is generally impossible to be solved analytically. Even worse, model-based approaches cannot be used for approximately solving HJI equation, when the accurate system model is unavailable or costly to obtain in practice. To overcome these difficulties, an off-policy reinforcement learning (RL) method is introduced to learn the solution of HJI equation from real system data instead of mathematical system model, and its convergence is proved. In the off-policy RL method, the system data can be generated with arbitrary policies rather than the evaluating policy, which is extremely important and promising for practical systems. For implementation purpose, a neural network (NN)-based actor-critic structure is employed and a least-square NN weight update algorithm is derived based on the method of weighted residuals. Finally, the developed NN-based off-policy RL method is tested on a linear F16 aircraft plant, and further applied to a rotational/translational actuator system.

**Index Terms**— $H_\infty$  control design, Hamilton–Jacobi–Isaacs equation, neural network, off-policy learning, reinforcement learning.

## I. INTRODUCTION

**R**EINFORCEMENT learning (RL) is a machine learning technique that has been widely studied from the computational intelligence and machine learning scope in the

artificial intelligence community [1]–[4]. RL technique refers to an actor or agent that interacts with its environment and aims to learn the optimal actions, or control policies, by observing their responses from the environment. Sutton and Barto [2] suggested a definition of RL method, i.e., any method that is well suited for solving RL problem can be regarded as a RL method, where the RL problem is defined in terms of optimal control of Markov decision processes. This obviously established the relationship between the RL method and control community. Moreover, RL methods have the ability to find an optimal control policy from unknown environment, which makes RL a promising method for control design of real systems.

In the past few years, many RL approaches [5]–[23] have been introduced for solving the optimal control problems. Especially, some extremely important results were reported by using RL for solving the optimal control problem of discrete-time systems [7], [10], [14], [17], [18], [22]. Such as, Liu and Wei [17] suggested a finite-approximation-error based iterative adaptive dynamic programming approach [17], and a novel policy iteration (PI) method [22] for discrete-time nonlinear systems. For continuous-time systems, Murray *et al.* [5] presented two PI algorithms that avoid the necessity of knowing the internal system dynamics. Vrabie *et al.* [8], [9], [13] introduced the thought of PI and proposed an important framework of integral reinforcement learning (IRL). Modares *et al.* [21] developed an experience replay-based IRL algorithm for nonlinear partially unknown constrained-input systems. In [19], an online neural network (NN) based decentralized control strategy was developed for stabilizing a class of continuous-time nonlinear interconnected large-scale systems. In addition, it is worth mentioning that the thought of RL methods have also been introduced to solve the optimal control problem of partial differential equation (PDE) systems [6], [12], [15], [16], [23]. However, for most of practical real systems, the existence of external disturbances is usually unavoidable.

To reduce the effects of disturbance, robust controller is required for disturbance rejection. One effective solution is the  $H_\infty$  control method, which achieves disturbance attenuation in the  $L_2$ -gain setting [24]–[26], that is, to design a controller such that the ratio of the objective output energy to the disturbance energy is less than a prescribed level. Over the past few decades, a large number of theoretical results on nonlinear  $H_\infty$  control have been reported [27]–[29], where the  $H_\infty$  control problem can be transformed into how to solve the so-called Hamilton–Jacobi–Isaacs (HJI) equation. However, the HJI equation is a nonlinear PDE, which is difficult or

Manuscript received November 24, 2013; revised March 28, 2014 and April 2, 2014; accepted April 18, 2014. Date of publication May 9, 2014; date of current version December 15, 2014. This work was supported in part by the National Basic Research Program of China under 973 Program of Grant 2012CB720003, in part by the National Natural Science Foundation of China under Grant 61121003, and in part by the General Research Fund project from Science and Technology on Aircraft Control Laboratory of Beihang University under Grant 9140C480301130C48001. This publication was also made possible by NPRP grant # NPRP 4-1162-1-181 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors. This paper was recommended by Associate Editor H. M. Schwartz.

B. Luo is with the Science and Technology on Aircraft Control Laboratory, Beihang University (Beijing University of Aeronautics and Astronautics), Beijing 100191, China, and also with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: biao.luo@hotmail.com).

H.-N. Wu is with the Science and Technology on Aircraft Control Laboratory, Beihang University (Beijing University of Aeronautics and Astronautics), Beijing 100191, China (e-mail: whn@buaa.edu.cn).

T. Huang is with the Texas A&M University at Qatar, Doha 5825, Qatar (e-mail: tingwen.huang@qatar.tamu.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCYB.2014.2319577

impossible to solve, and may not have global analytic solutions even in simple cases.

Thus, some works have been reported to solve the HJI equation approximately [27], [30]–[35]. In [27], it was shown that there exists a sequence of PIs on the control input such that the HJI equation is successively approximated with a sequence of HJB-like equations. Then, the methods for solving HJB equation can be extended for the HJI equation. In [36], the HJB equation was successively approximated by a sequence of linear PDEs, which were solved with Galerkin approximation in [30], [37], and [38]. In [39], the successive approximation method was extended to solve the discrete-time HJI equation. Similar to [30], a PI scheme was developed in [31] for the constrained input system. For implementation purpose of this scheme, a neuro-dynamic programming approach was introduced in [40] and an online adaptive method was given in [41]. This approach suits for the case that the saddle point exists, thus a situation that the smooth saddle point does not exist was considered in [42]. In [32], a synchronous PI method was developed, which is the extension of the work [43]. To improve the efficiency for computing the solution of HJI equation, Luo and Wu [44] proposed a computationally efficient simultaneous policy update algorithm (SPUA). In addition, in [45] the solution of the HJI equation was approximated by the Taylor series expansion, and an efficient algorithm was furnished to generate the coefficients of the Taylor series. It is observed that most of these methods [27], [30]–[33], [35], [40], [44], [45] are model-based, where the full system model is required. However, the accurate system model is usually unavailable or costly to obtain for many practical systems. Thus, some RL approaches have been proposed for  $H_\infty$  control design of linear systems [46], [47] and nonlinear systems [48] with unknown internal system model. But these methods are on-policy learning approaches [32], [41], [46]–[49], where the cost function should be evaluated by using system data generated with policies being evaluated. It is found that there are several drawbacks (to be discussed in Section III) to apply the on-policy learning to solve real  $H_\infty$  control problem.

To overcome this problem, this paper introduces an off-policy RL method to solve the nonlinear continuous-time  $H_\infty$  control problem with unknown internal system model. The rest of the paper is rearranged as follows. Sections II and III present the problem description and the motivation. The off-policy learning methods for nonlinear systems and linear systems are developed in Sections IV and V, respectively. The simulation studies are conducted in Section VI and a brief conclusion is given in Section VII.

*Notations:*  $\mathbb{R}$ ,  $\mathbb{R}^n$  and  $\mathbb{R}^{n \times m}$  are the set of real numbers, the  $n$ -dimensional Euclidean space and the set of all real matrices, respectively.  $\|\cdot\|$  denotes the vector norm or matrix norm in  $\mathbb{R}^n$  or  $\mathbb{R}^{n \times m}$ , respectively. The superscript  $T$  is used for the transpose and  $I$  denotes the identity matrix of appropriate dimension.  $\nabla \triangleq \partial/\partial x$  denotes a gradient operator notation. For a symmetric matrix  $M$ ,  $M > (\geq) 0$  means that it is a positive (semi-positive) definite matrix.  $\|v\|_M^2 \triangleq v^T M v$  for some real vector  $v$  and symmetric matrix  $M > (\geq) 0$  with appropriate dimensions.  $C^1(\mathcal{X})$  is function space on  $\mathcal{X}$  with first derivatives are continuous.  $L_2[0, \infty)$  is a Banach space, for

$\forall w(t) \in L_2[0, \infty)$ ,  $\int_0^\infty \|w(t)\|^2 dt < \infty$ . Let  $\mathcal{X}$ ,  $\mathcal{U}$  and  $\mathcal{W}$  be compact sets, denote  $\mathcal{D} \triangleq \{(x, u, w) | x \in \mathcal{X}, u \in \mathcal{U}, w \in \mathcal{W}\}$ . For column vector functions  $s_1(x, u, w)$  and  $s_2(x, u, w)$ , where  $(x, u, w) \in \mathcal{D}$  define inner product  $\langle s_1(x, u, w), s_2(x, u, w) \rangle_{\mathcal{D}} \triangleq \int_{\mathcal{D}} s_1^T(x, u, w) s_2(x, u, w) d(x, u, w)$  and norm  $\|s_1(x, u, w)\|_{\mathcal{D}} \triangleq (\int_{\mathcal{D}} s_1^T(x, u, w) s_1(x, u, w) d(x, u, w))^{1/2}$ .  $H^{m,p}(\mathcal{X})$  is a Sobolev space that consists of functions in space  $L_p(\mathcal{X})$  such that their derivatives of order at least  $m$  are also in  $L_p(\mathcal{X})$ .

## II. PROBLEM DESCRIPTION

Let us consider the following affine nonlinear continuous-time dynamical system:

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + k(x(t))w(t) \quad (1)$$

$$z(t) = h(x) \quad (2)$$

where  $[x_1 \dots x_n]^T \in \mathcal{X} \subset \mathbb{R}^n$  is the state,  $u = [u_1 \dots u_m]^T \in \mathcal{U} \subset \mathbb{R}^m$  is the control input and  $u(t) \in L_2[0, \infty)$ ,  $w = [w_1 \dots w_q]^T \in \mathcal{W} \subset \mathbb{R}^q$  is the external disturbance and  $w(t) \in L_2[0, \infty)$ ,  $z = [z_1 \dots z_p]^T \in \mathbb{R}^p$  is the objective output.  $f(x)$  is Lipschitz continuous on a set  $\mathcal{X}$  that contains the origin,  $f(0) = 0$ .  $f(x)$  represents the internal system model which is assumed to be unknown in this paper.  $g(x)$ ,  $k(x)$ , and  $h(x)$  are known continuous vector or matrix functions of appropriate dimensions.

The  $H_\infty$  control problem under consideration is to find a state feedback control law  $u(x)$  such that the system (1) is closed-loop asymptotically stable, and has  $L_2$ -gain less than or equal to  $\gamma$ , that is

$$\int_0^\infty (\|z(t)\|^2 + \|u(t)\|_R^2) dt \leq \gamma^2 \int_0^\infty \|w(t)\|^2 dt \quad (3)$$

for all  $w(t) \in L_2[0, \infty)$ ,  $R > 0$  and  $\gamma > 0$  is some prescribed level of disturbance attenuation. From [27], this problem can be transformed to solve the so-called HJI equation, which is summarized in Lemma 1.

*Lemma 1:* Assume the system (1) and (2) is zero-state observable. For  $\gamma > 0$ , suppose there exists a solution  $V^*(x)$  to the HJI equation

$$\begin{aligned} & [\nabla V^*(x)]^T f(x) + h^T(x)h(x) \\ & - \frac{1}{4} [\nabla V^*(x)]^T g(x)R^{-1}g^T(x)\nabla V^*(x) \\ & + \frac{1}{4\gamma^2} [\nabla V^*(x)]^T k(x)k^T(x)\nabla V^*(x) = 0 \end{aligned} \quad (4)$$

where  $V^*(x) \in C^1(\mathcal{X})$ ,  $V^*(x) \geq 0$  and  $V^*(0) = 0$ . Then, the closed-loop system with the state feedback control

$$u(t) = u^*(x(t)) = -\frac{1}{2}R^{-1}g^T(x)\nabla V^*(x) \quad (5)$$

has  $L_2$ -gain less than or equal to  $\gamma$ , and the closed-loop system (1), (2) and (5) (when  $w(t) \equiv 0$ ) is locally asymptotically stable.

### III. MOTIVATION FROM INVESTIGATION OF RELATED WORK

From Lemma 1, it is noted that the  $H_\infty$  control (5) relies on the solution of the HJI equation (4). Therefore, a model-based iterative method was proposed in [30], where the HJI equation is successively approximated by a sequence of linear PDEs

$$[\nabla V^{(i,j+1)}]^T (f + gu^{(i)} + kw^{(i,j)}) + h^T h + \|u^{(i)}\|_R^2 - \gamma^2 \|w^{(i,j)}\|^2 = 0 \quad (6)$$

and then update control and disturbance policies with

$$w^{(i,j+1)} \triangleq \frac{1}{2} \gamma^{-2} k^T \nabla V^{(i,j+1)} \quad (7)$$

$$u^{(i+1)} \triangleq -\frac{1}{2} R^{-1} g^T \nabla V^{(i+1)} \quad (8)$$

with  $V^{(i+1)} \triangleq \lim_{j \rightarrow \infty} V^{(i,j)}$ . From [27] and [30], it was indicated that the  $V^{(i,j)}$  can converge to the solution of the HJI equation, i.e.,  $\lim_{i,j \rightarrow \infty} V^{(i,j)} = V^*$ .

*Remark 1:* Note that the key point of the iterative scheme (6)–(8) depends on the solution of the linear PDE (6). Thus, several related methods were developed, such as, Galerkin approximation [30], synchronous PI [32], neuro-dynamic programming approach [31], [40], and online adaptive control method [41] for constrained input systems, and Galerkin approximation method for discrete-time systems [39]. Obviously, the iteration (6)–(8) will generate two iterative loops since the control and disturbance policies are updated at the different iterative steps, i.e., the inner loop for updating disturbance policy  $w$  with index  $j$ , and the outer iterative loop for updating control policy  $u$  with index  $i$ . The outer loop will not be activated until the inner loop is convergent, which results in low efficiency. Therefore, Luo and Wu [44] SPUA, where the control and disturbance policies are updated at the same iterative step, and thus only one iterative loop is required. It worth noting that the word “simultaneous” in [44] and the word “synchronous/simultaneous” in [32] and [41] represent different meanings. The former emphasizes the same “iterative step,” while the latter emphasizes the same “time instant.” In other words, the SPUA in [44] updates control and disturbance policies at the “same” iterative step, while the algorithms in [32] and [41] update control and disturbance policies at the “different” iterative steps.

The procedure of model-based SPUA is given in Algorithm 1.

It is worth noting that Algorithm 1 is an infinite iterative procedure, which is used for theoretical analysis rather than for implementation purpose. That is to say, Algorithm 1 will converge to the solution of the HJI equation (4) when the iteration goes to infinity. By constructing a fixed point equation, the convergence [44] of Algorithm 1 is established by proving that it is essentially a Newton’s iteration method for finding the fixed point. With the increase of index  $i$ , the sequence  $V^{(i)}$  obtained by the SPUA with (9)–(11) can converge to the solution of HJI equation (4), i.e.,  $\lim_{i \rightarrow \infty} V^{(i)} = V^*$ .

*Remark 2:* It is necessary to explain the rationale of using (9) and (10) for control and disturbance policies update. The  $H_\infty$  control problem (1)–(3) can be viewed as a two-players zero-sum differential game problem [26], [32]–[34], [40], [41],

---

#### Algorithm 1 Model-Based SPUA

---

- *Step 1:* Give an initial function  $V^{(0)} \in \mathbb{V}_0$  ( $\mathbb{V}_0 \subset \mathbb{V}$  is determined by [44, Lemma 5]). Let  $i = 0$ ;
- *Step 2:* Update the control and disturbance policies with

$$u^{(i)} \triangleq -\frac{1}{2} R^{-1} g^T \nabla V^{(i)} \quad (9)$$

$$w^{(i)} \triangleq \frac{1}{2} \gamma^{-2} k^T \nabla V^{(i)} \quad (10)$$

- *Step 3:* Solve the following linear PDE for  $V^{(i+1)}(x)$  :
 
$$[\nabla V^{(i+1)}]^T (f + gu^{(i)} + kw^{(i)}) + h^T h + \|u^{(i)}\|_R^2 - \gamma^2 \|w^{(i)}\|^2 = 0; \quad (11)$$
 where  $V^{(i+1)}(x) \in C^1(\mathcal{X})$ ,  $V^{(i+1)}(x) \geq 0$  and  $V^{(i+1)}(0) = 0$ .
  - *Step 4:* Let  $i = i + 1$ , go back to Step 2 and continue.
- 

[47]. The game problem is a minimax problem, where the control policy  $u$  acts as the minimizing player and the disturbance policy  $w$  is the maximizing player. The game problem aims at finding the saddle point  $(u^*, w^*)$ , where  $u^*$  is given by expression (5) and  $w^*$  is given by  $w^*(x) \triangleq \frac{1}{2} \gamma^{-2} k^T (x) \nabla V^*(x)$ . Correspondingly, for the  $H_\infty$  control problem (1)–(3),  $u^*$  and  $w^*$  are the associated  $H_\infty$  control policy and the worst disturbance signal [26], [31], [32], [34], [40], [47], respectively. Thus, it is reasonable using expressions (9) and (10) (that are consistent with  $u^*$  and  $w^*$  in form) for control and disturbance policies update. Similar control and disturbance policy update method could be found in [27], [30], [31], [34], [40], and [41].

Observe that both iterative equations (6) and (11) require the full system model. For the  $H_\infty$  control problem that the internal system dynamic  $f(x)$  is unknown, data-based methods [47], [48] were suggested to solve the HJI equation online. However, most of related existing online methods are on-policy learning approaches [32], [41], [47]–[49]. From the definition of on-policy learning [2], the cost function should be evaluated with the data generated from the evaluating policies. For example,  $V^{(i,j+1)}$  in (6) is the cost function of the policies  $w^{(i,j)}$  and  $u^{(i)}$ , which means that  $V^{(i,j+1)}$  should be evaluated with system data by using evaluating policies  $w^{(i,j)}$  and  $u^{(i)}$ . It is observed that these on-policy learning approaches for solving the  $H_\infty$  control problem have several drawbacks.

- 1) For real implementation of on-policy learning methods [32], [41], [48], [49], the approximate evaluating control and disturbance policies (rather than the actual policies) are used to generate data for learning their cost function. In other words, the on-policy learning methods using the “inaccurate” data to learn their cost function, which will increase the accumulated error. For example, to learn the cost function  $V^{(i,j+1)}$  in (6), some approximate policies  $\hat{w}^{(i,j)}$  and  $\hat{u}^{(i)}$  (rather than its actual policies  $w^{(i,j)}$  and  $u^{(i)}$ , which are usually unknown because of estimate error) are employed to generate data.
- 2) The evaluating control and disturbance policies are required to generate data for on-policy learning, thus disturbance signal should be adjustable, which is usually impractical for most of real systems.



- 3) It is known [2], [50] that the issue of “exploration” is extremely important in RL for learning the optimal control policy, and the lack of exploration during the learning process may lead to divergency. Nevertheless, for on-policy learning, exploration is restricted because only the evaluating policies can be used to generate data. From the literature investigation, it is found that the “exploration” issue is rarely discussed in existing work that using RL techniques for control design.
- 4) The implementation structure is complicated, such as in [32] and [41], three NNs are required for approximating cost function, control, and disturbance policies, respectively.
- 5) Most of existing approaches [32], [41], [47]–[49] are implemented online, thus they are difficult for real-time control because the learning process is often time-consuming. Furthermore, online control design approaches just use current data while discard past data, which implies that the measured system data is used only once and thus results in low utilization efficiency.

To overcome the drawbacks mentioned above, we propose an off-policy RL approach to solve the  $H_\infty$  control problem with unknown internal system dynamic  $f(x)$ .

#### IV. OFF-POLICY REINFORCEMENT LEARNING FOR $H_\infty$ CONTROL

In this section, an off-policy RL method for  $H_\infty$  control design is derived and its convergence is proved. Then, a NN-based critic-actor structure is developed for implementation purpose.

##### A. Off-Policy Reinforcement Learning

To derive the off-policy RL method, we rewrite the system (1) as

$$\dot{x} = f + gu^{(i)} + kw^{(i)} + g[u - u^{(i)}] + k[w - w^{(i)}] \quad (12)$$

for  $\forall u \in \mathcal{U}$ ,  $w \in \mathcal{W}$ . Let  $V^{(i+1)}(x)$  be the solution of the linear PDE (11), then taking derivative along the state of system (12) yields

$$\begin{aligned} \frac{dV^{(i+1)}(x)}{dt} &= [\nabla V^{(i+1)}]^T (f + gu^{(i)} + kw^{(i)}) \\ &+ [\nabla V^{(i+1)}]^T g[u - u^{(i)}] + [\nabla V^{(i+1)}]^T k[w - w^{(i)}]. \end{aligned} \quad (13)$$

With the linear PDE (11), conducting integral on both sides of (13) in time interval  $[t, t + \Delta t]$  and rearranging terms yield

$$\begin{aligned} \int_t^{t+\Delta t} [\nabla V^{(i+1)}(x(\tau))]^T g(x(\tau)) [u(\tau) - u^{(i)}(x(\tau))] d\tau \\ + \int_t^{t+\Delta t} [\nabla V^{(i+1)}(x(\tau))]^T k(x(\tau)) [w(\tau) - w^{(i)}(x(\tau))] d\tau \\ + V^{(i+1)}(x(t)) - V^{(i+1)}(x(t + \Delta t)) \\ = \int_t^{t+\Delta t} (h^T(x(\tau))h(x(\tau)) + \|u^{(i)}(x(\tau))\|_R^2 \\ - \gamma^2 \|w^{(i)}(x(\tau))\|^2) d\tau. \end{aligned} \quad (14)$$

It is observed from the equation (14) that the cost function  $V^{(i+1)}$  can be learned by using arbitrary input signals  $u$  and  $w$ , rather

than the evaluating policies  $u^{(i)}$  and  $w^{(i)}$ . Then, replacing linear PDE (11) in Algorithm 1 with (14) results in the off-policy RL method. To show its convergence, Theorem 1 establishes the equivalence between iterative equations (11) and (14).

*Theorem 1:* Let  $V^{(i+1)}(x) \in C^1(\mathcal{X})$ ,  $V^{(i+1)}(x) \geq 0$  and  $V^{(i+1)}(0) = 0$ .  $V^{(i+1)}(x)$  is the solution of (14) iff (if and only if) it is the solution of the linear PDE (11), i.e., (14) is equivalent to the linear PDE (11).

*Proof:* From the derivation of (14), it is concluded that if  $V^{(i+1)}(x)$  is the solution of the linear PDE (11), then  $V^{(i+1)}(x)$  also satisfies (14). To complete the proof, we have to show that  $V^{(i+1)}(x)$  is the unique solution of (14). The proof is by contradiction.

Before starting the contradiction proof, we derive a simple fact. Consider

$$\begin{aligned} \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_t^{t+\Delta t} h(\tau) d\tau \\ = \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \left( \int_0^{t+\Delta t} h(\tau) d\tau - \int_0^t h(\tau) d\tau \right) \\ = \frac{d}{dt} \int_0^t h(\tau) d\tau \\ = h(t). \end{aligned} \quad (15)$$

From (14), we have

$$\begin{aligned} \frac{dV^{(i+1)}(x)}{dt} &= \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} [V^{(i+1)}(x(t + \Delta t)) - V^{(i+1)}(x(t))] \\ &= \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_t^{t+\Delta t} [\nabla V^{(i+1)}(x(\tau))]^T g(x(\tau)) \\ &\quad [u(\tau) - u^{(i)}(x(\tau))] d\tau \\ &\quad + \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_t^{t+\Delta t} [\nabla V^{(i+1)}(x(\tau))]^T k(x(\tau)) \\ &\quad [w(\tau) - w^{(i)}(x(\tau))] d\tau \\ &\quad - \lim_{\Delta t \rightarrow 0} \frac{1}{\Delta t} \int_t^{t+\Delta t} [h^T(x(\tau))h(x(\tau)) + \|u^{(i)}(x(\tau))\|_R^2 \\ &\quad - \gamma^2 \|w^{(i)}(x(\tau))\|^2] d\tau. \end{aligned} \quad (16)$$

By using the fact (15), (16) is rewritten as

$$\begin{aligned} \frac{dV^{(i+1)}(x)}{dt} &= [\nabla V^{(i+1)}(x(t))]^T g(x(t)) [u(t) - u^{(i)}(x(t))] \\ &\quad + [\nabla V^{(i+1)}(x(t))]^T k(x(t)) [w(t) - w^{(i)}(x(t))] \\ &\quad - [h^T(x(t))h(x(t)) + \|u^{(i)}(x(t))\|_R^2 - \gamma^2 \|w^{(i)}(x(t))\|^2]. \end{aligned} \quad (17)$$

Suppose that  $W(x) \in C^1(\mathcal{X})$  is another solution of (14) with boundary condition  $W(0) = 0$ . Thus,  $W(x)$  also satisfies (17), i.e.,

$$\begin{aligned} \frac{dW(x)}{dt} &= [\nabla W(x(t))]^T g(x(t)) [u(t) - u^{(i)}(x(t))] \\ &\quad + [\nabla W(x(t))]^T k(x(t)) [w(t) - w^{(i)}(x(t))] \\ &\quad - [h^T(x(t))h(x(t)) + \|u^{(i)}(x(t))\|_R^2 - \gamma^2 \|w^{(i)}(x(t))\|^2]. \end{aligned} \quad (18)$$

Substituting (18) from (17) yields

$$\begin{aligned} & \frac{d}{dt} \left( V^{(i+1)}(x) - W(x) \right) \\ &= \left[ \nabla \left( V^{(i+1)}(x) - W(x) \right) \right]^T g(x) \left[ u - u^{(i)}(x) \right] \\ &+ \left[ \nabla \left( V^{(i+1)}(x) - W(x) \right) \right]^T k(x) \left[ w - w^{(i)}(x) \right]. \end{aligned} \quad (19)$$

This means that (19) holds for  $\forall u \in \mathcal{U}, w \in \mathcal{W}$ . If letting  $u = u^{(i)}, w = w^{(i)}$ , we have

$$\frac{d}{dt} \left[ V^{(i+1)}(x) - W(x) \right] = 0. \quad (20)$$

Then,  $V^{(i+1)}(x) - W(x) = c$  for  $\forall x \in \mathcal{X}$ , where  $c$  is a real constant, and  $c = V^{(i+1)}(0) - W(0) = 0$ . Thus,  $V^{(i+1)}(x) - W(x) = 0$ , i.e.,  $W(x) = V^{(i+1)}(x)$  for  $\forall x \in \mathcal{X}$ . This completes the proof. ■

*Remark 3:* It follows from Theorem 1 that the solution of (14) is equivalent to (11), and thus the convergence of the off-policy RL is guaranteed, i.e., the solution of the iterative equation (14) will converge to the solution of HJI equation (4) as iteration step  $i$  increases. Different from the equation (11) in Algorithm 1, the off-policy RL with (14) uses system data instead of the internal system dynamic  $f(x)$ . Hence, the off-policy RL can be regarded as a direct learning method for  $H_\infty$  control design, which avoids the identification of  $f(x)$ . In fact, the information of  $f(x)$  is embedded in the measurement of system data. That is to say, the lack of knowledge about  $f(x)$  does not have any impact on the off-policy RL to obtain the solution of HJI equation (4) and the  $H_\infty$  control policy. It worth pointing out that the equation (14) is similar with the form of the IRL [8], [9], which is an important framework for control design of continuous-time systems. The IRL in [8] and [9] is an online optimal control learning algorithm for partially unknown systems.

### B. Implementation Based on Neural Network

To solve (14) for the unknown function  $V^{(i+1)}(x)$  based on system data, we develop a NN-based actor-critic structure. From the well-known high-order Weierstrass approximation theorem [51], a continuous function can be represented by an infinite-dimensional linearly independent basis function set. For real practical application, it is usually required to approximate the function in a compact set with a finite-dimensional function set. We consider the critic NN for approximating the cost function on a compact set  $\mathcal{X}$ . Let  $\varphi(x) \triangleq [\varphi_1(x) \dots \varphi_L(x)]^T$  be the vector of linearly independent activation functions for critic NN, where  $\varphi_l(x): \mathcal{X} \mapsto \mathbb{R}, l = 1, \dots, L, L$  is the number of critic NN hidden layer neurons. Then, the output of critic NN is given by

$$\widehat{V}^{(i)}(x) = \sum_{l=1}^L \theta_l^{(i)} \varphi_l(x) = \varphi^T(x) \theta^{(i)} \quad (21)$$

for  $\forall i = 0, 1, 2, \dots$ , where  $\theta^{(i)} \triangleq [\theta_1^{(i)} \dots \theta_L^{(i)}]^T$  is the critic NN weight vector. It follows from (9), (10), and (21) that the disturbance and control policies are given by:

$$\widehat{u}^{(i)}(x) = -\frac{1}{2} R^{-1} g^T(x) \nabla \varphi^T(x) \theta^{(i)} \quad (22)$$

$$\widehat{w}^{(i)}(x) = \frac{1}{2} \gamma^{-2} k^T(x) \nabla \varphi^T(x) \theta^{(i)} \quad (23)$$

for  $\forall i = 0, 1, 2, \dots$ , and  $\nabla \varphi(x) \triangleq [\partial \varphi_1 / \partial x \dots \partial \varphi_L / \partial x]^T$  is the Jacobian of  $\varphi(x)$ . Expressions (22) and (23) can be viewed as actor NNs for the disturbance and control policies respectively, where  $-\frac{1}{2} R^{-1} g^T(x) \nabla \varphi^T(x)$  and  $\frac{1}{2} \gamma^{-2} k^T(x) \nabla \varphi^T(x)$  are the activation function vectors and  $\theta^{(i)}$  is the actor NN weight vector.

Due to estimation errors of the critic and actor NNs (21)-(23), the replacement of  $V^{(i+1)}, w^{(i)}$  and  $u^{(i)}$  in the iterative equation (14) with  $\widehat{V}^{(i+1)}, \widehat{w}^{(i)}$  and  $\widehat{u}^{(i)}$  respectively, yields the following residual error:

$$\begin{aligned} & \sigma^{(i)}(x(t), u(t), w(t)) \\ & \triangleq \int_t^{t+\Delta t} \left[ u(\tau) - \widehat{u}^{(i)}(x(\tau)) \right]^T g^T(x(\tau)) \nabla \varphi^T(x(\tau)) \theta^{(i+1)} d\tau \\ & + \int_t^{t+\Delta t} \left[ w(\tau) - \widehat{w}^{(i)}(x(\tau)) \right]^T k^T(x(\tau)) \nabla \varphi^T(x(\tau)) \theta^{(i+1)} d\tau \\ & + [\varphi(x(t)) - \varphi(x(t + \Delta t))]^T \theta^{(i+1)} \\ & - \int_t^{t+\Delta t} \left[ h^T(x(\tau)) h(x(\tau)) + \|\widehat{u}^{(i)}(x(\tau))\|_R^2 \right. \\ & \quad \left. - \gamma^2 \|\widehat{w}^{(i)}(x(\tau))\|^2 \right] d\tau \\ & = \int_t^{t+\Delta t} u^T(\tau) g^T(x(\tau)) \nabla \varphi^T(x(\tau)) \theta^{(i+1)} d\tau \\ & + \frac{1}{2} \int_t^{t+\Delta t} \left( \theta^{(i)} \right)^T \nabla \varphi(x(\tau)) g(x(\tau)) R^{-1} g^T(x(\tau)) \\ & \quad \nabla \varphi^T(x(\tau)) \theta^{(i+1)} d\tau \\ & + \int_t^{t+\Delta t} w^T(\tau) k^T(x(\tau)) \nabla \varphi^T(x(\tau)) \theta^{(i+1)} d\tau \\ & - \frac{1}{2} \gamma^{-2} \int_t^{t+\Delta t} \left( \theta^{(i)} \right)^T \nabla \varphi(x(\tau)) k(x(\tau)) k^T(x(\tau)) \\ & \quad \nabla \varphi^T(x(\tau)) \theta^{(i+1)} d\tau + [\varphi(x(t)) - \varphi(x(t + \Delta t))]^T \theta^{(i+1)} \\ & - \frac{1}{4} \int_t^{t+\Delta t} \left( \theta^{(i)} \right)^T \nabla \varphi(x(\tau)) g(x(\tau)) R^{-1} g^T(x(\tau)) \\ & \quad \nabla \varphi^T(x(\tau)) \theta^{(i)} d\tau \\ & + \frac{1}{4} \gamma^{-2} \int_t^{t+\Delta t} \left( \theta^{(i)} \right)^T \nabla \varphi(x(\tau)) k(x(\tau)) k^T(x(\tau)) \\ & \quad \nabla \varphi^T(x(\tau)) \theta^{(i)} d\tau - \int_t^{t+\Delta t} h^T(x(\tau)) h(x(\tau)) d\tau. \end{aligned} \quad (24)$$

For notation simplicity, define

$$\begin{aligned} \rho_{\Delta \varphi}(x(t)) & \triangleq [\varphi(x(t)) - \varphi(x(t + \Delta t))]^T \\ \rho_{g \varphi}(x(t)) & \triangleq \int_t^{t+\Delta t} \nabla \varphi(x(\tau)) g(x(\tau)) R^{-1} g^T(x(\tau)) \\ & \quad \nabla \varphi^T(x(\tau)) d\tau \\ \rho_{k \varphi}(x(t)) & \triangleq \int_t^{t+\Delta t} \nabla \varphi(x(\tau)) k(x(\tau)) k^T(x(\tau)) \\ & \quad \nabla \varphi^T(x(\tau)) d\tau \\ \rho_{u \varphi}(x(t), u(t)) & \triangleq \int_t^{t+\Delta t} u^T(\tau) g^T(x(\tau)) \nabla \varphi^T(x(\tau)) d\tau \\ \rho_{w \varphi}(x(t), w(t)) & \triangleq \int_t^{t+\Delta t} w^T(\tau) k^T(x(\tau)) \nabla \varphi^T(x(\tau)) d\tau \\ \rho_h(x(t)) & \triangleq \int_t^{t+\Delta t} h^T(x(\tau)) h(x(\tau)) d\tau \end{aligned}$$

then, expression (24) is rewritten as

$$\begin{aligned} \sigma^{(i)}(x(t), u(t), w(t)) &= \rho_{u\varphi}(x(t), u(t))\theta^{(i+1)} + \frac{1}{2}(\theta^{(i)})^T \rho_{g\varphi}(x(t))\theta^{(i+1)} \\ &+ \rho_{w\varphi}(x(t), w(t))\theta^{(i+1)} - \frac{1}{2}\gamma^{-2}(\theta^{(i)})^T \rho_{k\varphi}(x(t))\theta^{(i+1)} \\ &+ \rho_{\Delta\varphi}\theta^{(i+1)} - \frac{1}{4}(\theta^{(i)})^T \rho_{g\varphi}(x(t))\theta^{(i)} \\ &+ \frac{1}{4}\gamma^{-2}(\theta^{(i)})^T \rho_{k\varphi}(x(t))\theta^{(i)} - \rho_h(x(t)). \end{aligned} \quad (25)$$

For description convenience, expression (25) is represented as a compact form

$$\sigma^{(i)}(x(t), u(t), w(t)) = \bar{\rho}^{(i)}(x(t), u(t), w(t))\theta^{(i+1)} - \pi^{(i)}(x(t)) \quad (26)$$

where

$$\begin{aligned} \bar{\rho}^{(i)}(x(t), u(t), w(t)) &\triangleq \rho_{u\varphi}(x(t), u(t)) + \frac{1}{2}(\theta^{(i)})^T \rho_{g\varphi}(x(t)) \\ &+ \rho_{w\varphi}(x(t), w(t)) - \frac{1}{2}\gamma^{-2}(\theta^{(i)})^T \rho_{k\varphi}(x(t)) + \rho_{\Delta\varphi} \\ \pi^{(i)}(x(t)) &\triangleq \frac{1}{4}(\theta^{(i)})^T \rho_{g\varphi}(x(t))\theta^{(i)} \\ &- \frac{1}{4}\gamma^{-2}(\theta^{(i)})^T \rho_{k\varphi}(x(t))\theta^{(i)} + \rho_h(x(t)). \end{aligned}$$

For description simplicity, denote  $\bar{\rho}^{(i)} = [\bar{\rho}_1^{(i)} \dots \bar{\rho}_L^{(i)}]^T$ . Based on the method of weighted residuals [52], the unknown critic NN weight vector  $\theta^{(i+1)}$  can be computed in such a way that the residual error  $\sigma^{(i)}(x, u, w)$  (for  $\forall t \geq 0$ ) of (26) is forced to be zero in some average sense. Thus, projecting the residual error  $\sigma^{(i)}(x, u, w)$  onto  $d\sigma^{(i)}/d\theta^{(i+1)}$  and setting the result to zero on domain  $\mathcal{D}$  using the inner product,  $\langle \cdot, \cdot \rangle_{\mathcal{D}}$

$$\left\langle d\sigma^{(i)}/d\theta^{(i+1)}, \sigma^{(i)}(x, u, w) \right\rangle_{\mathcal{D}} = 0. \quad (27)$$

Then, the substitution of (26) into (27) yields

$$\begin{aligned} \left\langle \bar{\rho}^{(i)}(x, u, w), \bar{\rho}^{(i)}(x, u, w) \right\rangle_{\mathcal{D}} \theta^{(i+1)} \\ - \left\langle \bar{\rho}^{(i)}(x, u, w), \pi^{(i)}(x) \right\rangle_{\mathcal{D}} = 0 \end{aligned}$$

where the notations  $\langle \bar{\rho}^{(i)}, \bar{\rho}^{(i)} \rangle_{\mathcal{D}}$  and  $\langle \bar{\rho}^{(i)}, \pi^{(i)} \rangle_{\mathcal{D}}$  are given by

$$\begin{aligned} \left\langle \bar{\rho}^{(i)}, \bar{\rho}^{(i)} \right\rangle_{\mathcal{D}} &\triangleq \begin{bmatrix} \left\langle \bar{\rho}_1^{(i)}, \bar{\rho}_1^{(i)} \right\rangle_{\mathcal{D}} & \dots & \left\langle \bar{\rho}_1^{(i)}, \bar{\rho}_L^{(i)} \right\rangle_{\mathcal{D}} \\ \vdots & \dots & \vdots \\ \left\langle \bar{\rho}_L^{(i)}, \bar{\rho}_1^{(i)} \right\rangle_{\mathcal{D}} & \dots & \left\langle \bar{\rho}_L^{(i)}, \bar{\rho}_L^{(i)} \right\rangle_{\mathcal{D}} \end{bmatrix} \text{ and} \\ \left\langle \bar{\rho}^{(i)}, \pi^{(i)} \right\rangle_{\mathcal{D}} &\triangleq \begin{bmatrix} \left\langle \bar{\rho}_1^{(i)}, \pi^{(i)} \right\rangle_{\mathcal{D}} & \dots & \left\langle \bar{\rho}_L^{(i)}, \pi^{(i)} \right\rangle_{\mathcal{D}} \end{bmatrix}^T. \end{aligned}$$

Thus,  $\theta^{(i+1)}$  can be obtained with

$$\begin{aligned} \theta^{(i+1)} &= \left\langle \bar{\rho}^{(i)}(x, u, w), \bar{\rho}^{(i)}(x, u, w) \right\rangle_{\mathcal{D}}^{-1} \\ &\quad \left\langle \bar{\rho}^{(i)}(x, u, w), \pi^{(i)}(x) \right\rangle_{\mathcal{D}}. \end{aligned} \quad (28)$$

The computation of inner products  $\langle \bar{\rho}^{(i)}(x, u, w), \bar{\rho}^{(i)}(x, u, w) \rangle_{\mathcal{D}}$  and  $\langle \bar{\rho}^{(i)}(x, u, w), \pi^{(i)}(x) \rangle_{\mathcal{D}}$  involve many numerical integrals on domain  $\mathcal{D}$ , which

are computationally expensive. Thus, the Monte Carlo integration method [53] is introduced, which is especially competitive on multidimensional domain. We now illustrate the Monte Carlo integration for computing  $\langle \bar{\rho}^{(i)}(x, u, w), \bar{\rho}^{(i)}(x, u, w) \rangle_{\mathcal{D}}$ . Let  $I_{\mathcal{D}} \triangleq \int_{\mathcal{D}} d(x, u, w)$ , and  $\mathcal{S}_M \triangleq \{(x_m, u_m, w_m) | (x_m, u_m, w_m) \in \mathcal{D}, m = 1, 2, \dots, M\}$  be the set that sampled on domain  $\mathcal{D}$ , where  $M$  is size of sample set  $\mathcal{S}_M$ . Then,  $\langle \bar{\rho}^{(i)}(x, u, w), \bar{\rho}^{(i)}(x, u, w) \rangle_{\mathcal{D}}$  is approximately computed with

$$\begin{aligned} \left\langle \bar{\rho}^{(i)}(x, u, w), \bar{\rho}^{(i)}(x, u, w) \right\rangle_{\mathcal{D}} &= \int_{\mathcal{D}} \left( \bar{\rho}^{(i)}(x, u, w) \right)^T \bar{\rho}^{(i)}(x, u, w) d(x, u, w) \\ &= \frac{I_{\mathcal{D}}}{M} \sum_{m=1}^M \left( \bar{\rho}^{(i)}(x_m, u_m, w_m) \right)^T \bar{\rho}^{(i)}(x_m, u_m, w_m) \\ &= \frac{I_{\mathcal{D}}}{M} \left( Z^{(i)} \right)^T Z^{(i)} \end{aligned} \quad (29)$$

where  $Z^{(i)} \triangleq \left[ \left( \bar{\rho}^{(i)}(x_1, u_1, w_1) \right)^T \dots \left( \bar{\rho}^{(i)}(x_M, u_M, w_M) \right)^T \right]^T$ . Similarly,

$$\begin{aligned} \left\langle \bar{\rho}^{(i)}(x, u, w), \pi^{(i)}(x) \right\rangle_{\mathcal{D}} &= \frac{I_{\mathcal{D}}}{M} \sum_{m=1}^M \left( \bar{\rho}^{(i)}(x_m, u_m, w_m) \right)^T \pi^{(i)}(x_m) \\ &= \frac{I_{\mathcal{D}}}{M} \left( Z^{(i)} \right)^T \eta^{(i)} \end{aligned} \quad (30)$$

where  $\eta^{(i)} \triangleq [\pi^{(i)}(x_1) \dots \pi^{(i)}(x_M)]^T$ . Then, the substitution of (29) and (30) into (28) yields

$$\theta^{(i+1)} = \left[ \left( Z^{(i)} \right)^T Z^{(i)} \right]^{-1} \left( Z^{(i)} \right)^T \eta^{(i)}. \quad (31)$$

It is noted that the critic NN weight update rule (31) is a least-square scheme. Based on the update rule (31), the procedure for  $H_{\infty}$  control design with NN-based off-policy RL is presented in Algorithm 2.

*Remark 4:* In the least-square scheme (31), it is required to compute the inverse of matrix  $(Z^{(i)})^T Z^{(i)}$ . This means that the matrix  $Z^{(i)}$  should be full column rank, which depends on the richness of the sampling data set  $\mathcal{S}_M$  and its size  $M$ . To attain this goal in real implementation, it would be useful by increasing the size  $M$ , starting from different initial states, and using rich input signals, such as random noises, sinusoidal function noises with enough frequencies. Of course, it would be nice, if possible but is not a necessity, to use the persistent exciting input signals, while it is still a difficult issue [54], [55] that requires further investigation. In a word, the choices of rich input signals and the size  $M$  are generally experience-based.

Note that Algorithm 2 has two parts: the first part is Step 1 for data processing, i.e., measure system data  $(x, u, w)$  for computing  $\rho_{\Delta\varphi}$ ,  $\rho_{g\varphi}$ ,  $\rho_{k\varphi}$ ,  $\rho_{u\varphi}$ ,  $\rho_{w\varphi}$  and  $\rho_h$ ; the second part is Steps 2–4 for offline iterative learning the solution of the HJI equation (4).

*Remark 5:* From Theorem 1, the proposed off-policy RL is mathematically equivalent to the model-based SPUA

**Algorithm 2** NN-Based Off-Policy RL for  $H_\infty$  Control Design

- *Step 1:* Collect real system data  $(x_m, u_m, w_m)$  for sample set  $\mathcal{S}_M$ , and then compute  $\rho_{\Delta\varphi}(x_m)$ ,  $\rho_{g\varphi}(x_m)$ ,  $\rho_{k\varphi}(x_m)$ ,  $\rho_{u\varphi}(x_m, u_m)$ ,  $\rho_{w\varphi}(x_m, w_m)$  and  $\rho_h(x_m)$ ;
- *Step 2:* Select initial critic NN weight vector  $\theta^{(0)}$  such that  $\widehat{V}^{(0)} \in \mathbb{V}_0$ . Let  $i = 0$ ;
- *Step 3:* Compute  $Z^{(i)}$  and  $\eta^{(i)}$ , and update  $\theta^{(i+1)}$  with (31);
- *Step 4:* Let  $i = i + 1$ . If  $\|\theta^{(i)} - \theta^{(i-1)}\| \leq \xi$  ( $\xi$  is a small positive number), stop iteration and  $\theta^{(i)}$  is employed to obtain the  $H_\infty$  control policy with (22), else go back to Step 3 and continue.

(i.e., Algorithm 2), which is proved to be a Newton's method [44]. Hence, the off-policy RL have the same advantages and disadvantages as the Newton's method. That is to say, the off-policy RL is a local optimization method, and thus there exists a problem that an initial critic NN weight vector  $\theta^{(0)}$  should be given such that the initial solution  $\widehat{V}^{(0)}$  locates in a neighborhood  $\mathbb{V}_0$  of the HJI equation (4). In fact, this problem also widely arises in many existing works for solving optimal and  $H_\infty$  control problems of either linear or nonlinear systems through the observations from computer simulation, such as [5], [8], [9], [27], [30], [31], [36], [40], [42], and [56]–[59]. Till present, it is still a difficult issue for finding proper initializations or developing global approaches. There is no exception for the proposed off-policy RL algorithm, where the selection of initial weight vector  $\theta^{(i)}$  is still experience-based and requires further investigation.

Algorithm 2 can be viewed as an off-policy learning method according to [2], [60], and [61], which overcomes the drawbacks mentioned in Section III.

- 1) In the off-policy RL algorithm (i.e., Algorithm 2), the control  $u$  and disturbance  $w$  can be arbitrarily on  $\mathcal{U}$  and  $\mathcal{W}$ , where no error occurs during the process of generating data, and thus the accumulated error (exists in the on-policy learning methods mentioned in Section III) can be reduced.
- 2) In the Algorithm 2, the control  $u$  and disturbance  $w$  can be arbitrarily on  $\mathcal{U}$  and  $\mathcal{W}$ , and thus disturbance  $w$  does not required to be adjustable.
- 3) In the Algorithm 2, the cost function  $V^{(i+1)}$  of control and disturbance policies  $(u^{(i)}, w^{(i)})$  can be evaluated by using system data generated with other different control and disturbance signals  $(u, w)$ . Thus, the obvious advantage of the developed off-policy RL method is that it can learn the cost function and control policy from system data that are generated according to a more exploratory or even random policies.
- 4) The implementation of Algorithm 2 is very simple, in fact only one NN is required, i.e., critic NN. This means that once the critic NN weight vector  $\theta^{(i+1)}$  is computed via (31), the action NNs for control and disturbance policies can be obtained based on (22) and (23) accordingly.
- 5) The developed off-policy RL method learns the  $H_\infty$  control policy offline, which is then used for

real-time control. Thus, it is much more practical than online control design methods since less computational load will generate during real-time application. Meanwhile, note that in Algorithm 2, once the terms  $\rho_{\Delta\varphi}$ ,  $\rho_{g\varphi}$ ,  $\rho_{k\varphi}$ ,  $\rho_{u\varphi}$ ,  $\rho_{w\varphi}$  and  $\rho_h$  are computed with sample set  $\mathcal{S}_M$  (i.e., Step 1 is finished), no extra data is required for learning the  $H_\infty$  control policy (in Steps 2–4). This means that the collected data set can be utilized repeatedly, and thus the utilization efficiency is improved compared to the online control design methods.

*Remark 6:* Observe that the experience replay based IRL method [21] can be viewed as an off-policy method based on its definition [2]. There are three obvious differences between the method and the work of this paper. Firstly, the method in [21] is for solving the optimal control problem without external disturbance, while the off-policy RL algorithm in this paper is for solving the  $H_\infty$  control problem with external disturbance. Secondly, the method in [21] is online adaptive control approach. The off-policy RL algorithm in this paper uses real system information, and learns the  $H_\infty$  control policy by using an offline process. After the learning process is finished, the convergent control policy is employed for real system control. Thirdly, the method in [21] involves two NNs (i.e., one critic NN and one actor NN) for adaptive optimal control realization, while only one NN (i.e., critic NN) is required in the algorithm of this paper.

### C. Convergence Analysis for NN-Based Off-Policy RL

It is necessary to analyze the convergence of the NN-based off-policy RL algorithm. From Theorem 1, (14) in the off-policy RL is equivalent to the linear PDE (11), which means that the derived least-square scheme (31) is essentially for solving the linear PDE (11). In [57], a similar least-square method was suggested to solve the first order linear PDE directly, wherein some theoretical results are useful for analyzing the convergence of the proposed NN-based off-policy RL algorithm. The following Theorem 2 is given to show the convergence of critic NN and actor NNs.

*Theorem 2:* For  $i = 0, 1, 2, \dots$ , assume that  $V^{(i+1)} \in H^{1,2}(\mathcal{X})$  is the solution of (14), the critic NN activation functions  $\varphi_l(x) \in H^{1,2}(\mathcal{X})$ ,  $l = 1, 2, \dots, L$  are selected such that they are complete when  $L \rightarrow \infty$ ,  $V^{(i+1)}$  and  $\nabla V^{(i+1)}$  can be uniformly approximated, and the set  $\{\varphi_l(x_1) - \varphi_l(x_2)\}_{l=1}^L$  is linearly independent and complete for  $\forall x_1, x_2 \in \mathcal{X}, x_1 \neq x_2$ . Then

$$\sup_{x \in \mathcal{X}} |\widehat{V}^{(i+1)}(x) - V^{(i+1)}(x)| \rightarrow 0 \quad (32)$$

$$\sup_{x \in \mathcal{X}} |\nabla \widehat{V}^{(i+1)}(x) - \nabla V^{(i+1)}(x)| \rightarrow 0 \quad (33)$$

$$\sup_{x \in \mathcal{X}} |\widehat{u}^{(i+1)}(x) - u^{(i+1)}(x)| \rightarrow 0 \quad (34)$$

$$\sup_{x \in \mathcal{X}} |\widehat{w}^{(i+1)}(x) - w^{(i+1)}(x)| \rightarrow 0. \quad (35)$$

*Proof:* The proof procedure of the above results is very similar with that in [57], and thus some similar proof steps will



be omitted for avoidance of repetition. To use the theoretical results in [57], we firstly prove the  $\{\nabla\varphi_l(f + gu^{(i)} + kw^{(i)})\}_{l=1}^L$  is linear independent by contradiction. Assume this is not true, then there exists a vector  $\theta \triangleq [\theta_1 \dots \theta_L]^T \neq 0$  such that

$$\sum_{l=1}^L \theta_l \nabla \varphi_l (f + gu^{(i)} + kw^{(i)}) = 0$$

which means that for  $\forall x \in \mathcal{X}$

$$\begin{aligned} & \int_t^{t+\Delta t} \sum_{l=1}^L \theta_l \nabla \varphi_l (f + gu^{(i)} + kw^{(i)}) d\tau \\ &= \int_t^{t+\Delta t} \theta_l \frac{d\varphi_l}{d\tau} d\tau \\ &= \sum_{l=1}^L \theta_l [\varphi_l(x(t + \Delta t)) - \varphi_l(x(t))] \\ &= \sum_{l=1}^L \theta_l \varpi_l(x(t + \Delta t), x(t)) \\ &= 0. \end{aligned}$$

This contradicts the fact that the set  $\{\varpi_l\}_{l=1}^L$  is linearly independent, which implies that the set  $\{\nabla\varphi_l(f + gu^{(i)} + kw^{(i)})\}_{l=1}^L$  is linear independent.

From Theorem 1,  $V^{(i+1)}$  is the solution of the linear PDE (11). Then, with the same procedure used in Theorem 2 and [57, Corollary 2], the results (32)–(34) can be proven. And the result (35) can be proven in a similar way for (34). ■

The results (33)–(35) in Theorem 2 imply that the critic NN and actor NNs are convergent. In the following Theorem 3, we prove that the NN-based off-policy RL algorithm converges uniformly to the solution of the HJI equation (4) and the  $H_\infty$  control policy (5).

**Theorem 3:** If the conditions in Theorem 2 hold, then, for  $\forall \epsilon > 0$ ,  $\exists i_0, L_0$ , when  $i \geq i_0$  and  $L \geq L_0$ , we have

$$\sup_{x \in \mathcal{X}} |\widehat{V}^{(i)}(x) - V^*(x)| < \epsilon \quad (36)$$

$$\sup_{x \in \mathcal{X}} |\widehat{u}^{(i)}(x) - u^*(x)| < \epsilon \quad (37)$$

$$\sup_{x \in \mathcal{X}} |\widehat{w}^{(i)}(x) - w^*(x)| < \epsilon. \quad (38)$$

*Proof:* By following the same proof procedures in [57, Ths. 3 and 4], the results (36)–(38) can be proven directly. Similar to (37), the result (38) can also be proven. ■

**Remark 7:** The proposed off-policy RL method is to learn the solution of the HJI equation (4) and the  $H_\infty$  control policy (5). It follows from Theorem 3 that the control policy  $\widehat{u}^{(i)}$  designed by the off-policy RL will uniformly converge to the  $H_\infty$  control policy  $u^*$ . With the  $H_\infty$  control policy, it is noted from Lemma 1 that the closed-loop system (1) with  $w(t) \equiv 0$  is locally asymptotically stable. Furthermore, it is observed from (3) that for the closed-loop system with disturbance  $w(t) \in L_2[0, \infty)$ , the output  $z(t)$  is in  $L_2[0, \infty)$  [62], i.e., the closed-loop system is (bounded-input bounded-output) stable.

## V. OFF-POLICY REINFORCEMENT LEARNING FOR LINEAR $H_\infty$ CONTROL

In this section, the developed NN-based off-policy RL method (i.e., Algorithm 2) is simplified for linear  $H_\infty$  control design. Consider the linear system

$$\dot{x}(t) = Ax(t) + B_2 u(t) + B_1 w(t) \quad (39)$$

$$z(t) = Cx \quad (40)$$

where  $A \in \mathbb{R}^{n \times n}$ ,  $B_1 \in \mathbb{R}^{n \times q}$ ,  $B_2 \in \mathbb{R}^{n \times m}$  and  $C \in \mathbb{R}^{p \times n}$ . Then, the HJI equation (4) of the linear system (39) and (40) results in an algebraic Riccati equation (ARE) [46], [62]

$$A^T P + PA + Q + \gamma^{-2} PB_1 B_1^T P - PB_2 R^{-1} B_2^T P = 0 \quad (41)$$

where  $Q = C^T C$ . If ARE (41) has a stabilizing solution  $P \geq 0$ , the solution of the HJI equation (4) of the linear system (39) and (40) is  $V^*(x) = x^T P x$ , and then the linear  $H_\infty$  control policy (5) is accordingly given by

$$u^*(x) = -R^{-1} B_2^T P x. \quad (42)$$

Consequently,  $V^{(i)}(x) = x^T P^{(i)} x$ , then the iterative equations (9)–(11) in Algorithm 1 are respectively represented with

$$u^{(i)} = -R^{-1} B_2^T P^{(i)} x \quad (43)$$

$$w^{(i)} = \gamma^{-2} B_1^T P^{(i)} x \quad (44)$$

$$\bar{A}_i^T P^{(i+1)} + P^{(i+1)} \bar{A}_i + \bar{Q}^{(i)} = 0 \quad (45)$$

where  $\bar{A}_i \triangleq A + \gamma^{-2} B_1 B_1^T P^{(i)} - B_2 R^{-1} B_2^T P^{(i)}$  and  $\bar{Q}^{(i)} \triangleq Q - \gamma^{-2} P^{(i)} B_1 B_1^T P^{(i)} + P^{(i)} B_2 R^{-1} B_2^T P^{(i)}$ .

Similar to the derivation of the off-policy RL method for nonlinear  $H_\infty$  control design in Section IV, rewrite the linear system (39) as

$$\dot{x} = Ax + B_2 u^{(i)} + B_1 w^{(i)} + B_2 [u - u^{(i)}] + B_1 [w - w^{(i)}]. \quad (46)$$

Based on equations (43)–(46), (14) is given by

$$\begin{aligned} & \int_t^{t+\Delta t} x^T(\tau) P^{(i+1)} B_2 [u(\tau) + R^{-1} B_2^T P^{(i)} x(\tau)] d\tau \\ &+ \int_t^{t+\Delta t} x^T(\tau) P^{(i+1)} B_1 [w(\tau) - \gamma^{-2} B_1^T P^{(i)} x(\tau)] d\tau \\ &+ [x(t) - x(t + \Delta t)]^T P^{(i+1)} [x(t) - x(t + \Delta t)] \\ &= \int_t^{t+\Delta t} x^T(\tau) \bar{Q}^{(i)} x(\tau) d\tau \end{aligned} \quad (47)$$

where  $P^{(i+1)}$  is a  $n \times n$  unknown matrix to be learned. For notation simplicity, define

$$\begin{aligned} \rho_{\Delta x}(x(t)) &\triangleq x(t) - x(t + \Delta t) \\ \rho_{xx}(x(t)) &\triangleq \int_t^{t+\Delta t} x(\tau) \otimes x(\tau) d\tau \\ \rho_{ux}(x(t), u(t)) &\triangleq \int_t^{t+\Delta t} u(\tau) \otimes x(\tau) d\tau \\ \rho_{wx}(x(t), w(t)) &\triangleq \int_t^{t+\Delta t} w(\tau) \otimes x(\tau) d\tau \end{aligned}$$

where  $\otimes$  denotes Kronecker product. Each term of (47) can be written as

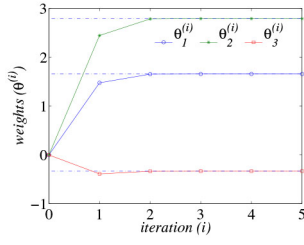


Fig. 1. For the linear F16 aircraft plant, the critic NN weights  $\theta_1^{(i)} \sim \theta_3^{(i)}$  at each iteration.

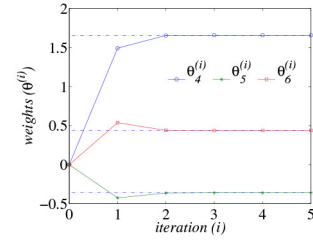


Fig. 2. For the linear F16 aircraft plant, the critic NN weights  $\theta_4^{(i)} \sim \theta_6^{(i)}$  at each iteration.

$$\begin{aligned}
 & \int_t^{t+\Delta t} x^T(\tau) P^{(i+1)} B_2 u(\tau) d\tau \\
 &= \rho_{ux}^T(x(t), u(t)) (B_2^T \otimes I) \text{vec}(P^{(i+1)}) \\
 & \int_t^{t+\Delta t} x^T(\tau) P^{(i+1)} B_2 R^{-1} B_2^T P^{(i)} x(\tau) d\tau \\
 &= \rho_{xx}^T(x(t)) (P^{(i)} B_2 R^{-1} B_2^T \otimes I) \text{vec}(P^{(i+1)}) \\
 & \int_t^{t+\Delta t} x^T(\tau) P^{(i+1)} B_1 w(\tau) d\tau \\
 &= \rho_{wx}^T(x(t), w(t)) (B_1^T \otimes I) \text{vec}(P^{(i+1)}) \\
 & \gamma^{-2} \int_t^{t+\Delta t} x^T(\tau) P^{(i+1)} B_1 B_1^T P^{(i)} x(\tau) d\tau \\
 &= \gamma^{-2} \rho_{xx}^T(x(t)) (P^{(i)} B_1 B_1^T \otimes I) \text{vec}(P^{(i+1)}) \\
 & [x(t) - x(t + \Delta t)]^T P^{(i+1)} [x(t) - x(t + \Delta t)] \\
 &= \rho_{\Delta x}^T(x(t)) \text{vec}(P^{(i+1)}) \\
 & \int_t^{t+\Delta t} x^T(\tau) \bar{Q}^{(i)} x(\tau) d\tau = \rho_{xx}^T(x(t)) \text{vec}(\bar{Q}^{(i)})
 \end{aligned}$$

where  $\text{vec}(P)$  denotes the vectorization of the matrix  $P$  formed by stacking the columns of  $P$  into a single column vector. Then, (47) can be rewritten as

$$\bar{\rho}^{(i)}(x(t), u(t), w(t)) \text{vec}(P^{(i+1)}) = \pi^{(i)}(x(t)) \quad (48)$$

with

$$\begin{aligned}
 \bar{\rho}^{(i)}(x(t), u(t), w(t)) &= \rho_{ux}^T(x(t), u(t)) (B_2^T \otimes I) \\
 &+ \rho_{wx}^T(x(t), w(t)) (B_1^T \otimes I) + \rho_{\Delta x}^T(x(t)) \\
 &- \gamma^{-2} (P^{(i)} B_1 B_1^T \otimes I) \\
 \pi^{(i)}(x(t)) &= \rho_{xx}^T(x(t)) \text{vec}(\bar{Q}^{(i)}).
 \end{aligned}$$

It is noted that (48) is equivalent to the equation (26) with residual error  $\sigma^{(i)} = 0$ . This is because no cost function approximation is required for linear systems. Then, by collecting sample set  $\mathcal{S}_M$  for computing  $\rho_{ux}$ ,  $\rho_{wx}$ ,  $\rho_{xx}$ , and  $\rho_{\Delta x}$ , a more simpler least-square scheme (31) can be derived to obtain the unknown parameter vector  $\text{vec}(P^{(i+1)})$  accordingly.

## VI. SIMULATION STUDIES

In this section, the efficiency of the developed NN-based off-policy RL method is tested on a F16 aircraft plant. Then, it is applied to the rotational/translational actuator (RTAC) nonlinear benchmark problem.

### A. Efficiency Test on Linear F16 Aircraft Plant

Consider a F16 aircraft plant that used in [32], [46], [48], and [63], where the system dynamics is described by a linear continuous-time model

$$\begin{aligned}
 \dot{x} &= \begin{bmatrix} -1.01887 & 0.90506 & -0.00215 \\ 0.82225 & -1.07741 & -0.17555 \\ 0 & 0 & -1 \end{bmatrix} x \\
 &+ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} w \\
 z &= x
 \end{aligned} \quad (49)$$

where the system state vector is  $x = [\alpha \ q \ \delta_e]^T$ ,  $\alpha$  denotes the angle of attack,  $q$  is the pitch rate and  $\delta_e$  is the elevator deflection angle. The control input  $u$  is the elevator actuator voltage and the disturbance  $w$  is wind gusts on angle of attack. Select  $R = 1$  and  $\gamma = 5$  for the  $L_2$ -gain performance (3). Then, solve the associated ARE (41) with the MATLAB command CARE, we obtain

$$P = \begin{bmatrix} 1.6573 & 1.3954 & -0.1661 \\ 1.3954 & 1.6573 & -0.1804 \\ -0.1661 & -0.1804 & 0.4371 \end{bmatrix}.$$

For linear systems, the solution of the HJI equation is  $V^*(x) = x^T P x$ , thus the complete activation function vector for critic NN is  $\varphi(x) = [x_1^2 \ x_1 x_2 \ x_1 x_3 \ x_2^2 \ x_2 x_3 \ x_3^2]^T$  of size  $L = 6$ . Then, the idea critic NN weight vector is  $\theta^* = [p_{11} \ p_{12} \ p_{13} \ p_{22} \ p_{23} \ p_{33}]^T = [1.6573 \ 2.7908 \ -0.3322 \ 1.6573 \ -0.3608 \ 0.4371]^T$ . Letting initial critic NN weight  $\theta_l^{(0)} = 0$  ( $l = 1, \dots, 6$ ), iterative stop criterion  $\xi = 10^{-7}$  and integral time interval  $\Delta t = 0.1s$ , Algorithm 2 is applied to learn the solution of the ARE. To generate sample set  $\mathcal{S}_M$ , let sample size  $M = 100$  and generate random noise in interval  $[0, 0.1]$  as input signals. Figs. 1 and 2 give the critic NN weight  $\theta^{(i)}$  at each iteration, in which the dash lines represent idea values of  $\theta^*$ . It is observed from the figures that the critic NN weight vector converges to the idea values of  $\theta^*$  at  $i = 5$  iteration. Then, the efficiency of the developed off-policy RL method is verified.

In addition, to test the influence of the parameter  $\Delta t$  to Algorithm 2, we reconduct simulation with different parameter cases:  $\Delta t = 0.2, 0.3, 0.4, 0.5s$ , and the results show that the critic NN weight vector  $\theta^{(i)}$  still converges to the idea values of  $\theta^*$  at  $i = 5$  iteration for all cases. This implies that the developed off-policy RL algorithm 2 is insensitive to the parameter  $\Delta t$ .

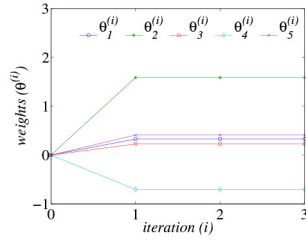


Fig. 3. For the RTAC system, the critic NN  $\theta_1^{(i)} \sim \theta_5^{(i)}$  weights at each iteration.

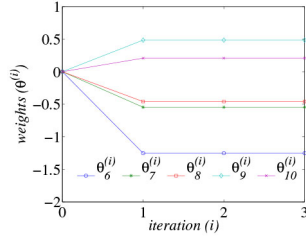


Fig. 4. For the RTAC system, the critic NN  $\theta_6^{(i)} \sim \theta_{10}^{(i)}$  weights at each iteration.

### B. Application to the Rotational/Translational Actuator Nonlinear Benchmark Problem

The RTAC nonlinear benchmark problem [40], [44], [64] has been widely used to test the abilities of control methods. The dynamics of this nonlinear plant poses challenges because the rotational and translation motions are coupled. The RTAC system is given as follows:

$$\dot{x} = \begin{bmatrix} \frac{x_2}{1 - \zeta^2 \cos^2 x_3} \\ \frac{-x_1 + \zeta x_4^2 \sin x_3}{1 - \zeta^2 \cos^2 x_3} \\ \frac{x_4}{1 - \zeta^2 \cos^2 x_3} \\ \frac{\zeta \cos x_3 (x_1 - \zeta x_4^2 \sin x_3)}{1 - \zeta^2 \cos^2 x_3} \end{bmatrix} + \begin{bmatrix} 0 \\ -\zeta \cos x_3 \\ 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \\ \frac{-\zeta \cos x_3}{1 - \zeta^2 \cos^2 x_3} \\ 0 \end{bmatrix} w \quad (51)$$

$$z = \sqrt{0.1}x \quad (52)$$

where  $\zeta = 0.2$ . For the  $L_2$ -gain performance (3), let  $R = 1$  and  $\gamma = 6$ . Then, the developed off-policy RL method is used to solve the nonlinear  $H_\infty$  control problem of system (51) and (52). Select the critic NN activation function vector as

$$\varphi(x) = \begin{bmatrix} x_1^2 & x_1 x_2 & x_1 x_3 & x_1 x_4 & x_2^2 & x_2 x_3 & x_2 x_4 & x_3^2 & x_3 x_4 \\ x_4^2 & x_1^3 x_2 & x_1^3 x_3 & x_1^3 x_4 & x_1^2 x_2^2 & x_1^2 x_2 x_3 & x_1^2 x_2 x_4 \\ x_1^2 x_3^2 & x_1^2 x_3 x_4 & x_1^2 x_4^2 & x_1 x_2^3 \end{bmatrix}^T$$

of size  $L = 20$ . With the initial critic NN weight  $\theta_l^{(0)} = 0$  ( $l = 1, \dots, 20$ ), iterative stop criterion  $\xi = 10^{-7}$  and integral time interval  $\Delta t = 0.033s$ , Algorithm 2 is applied to learn the solution of the HJI equation. To generate sample set  $\mathcal{S}_M$ , let

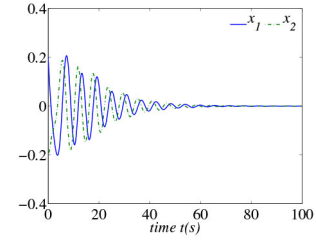


Fig. 5. State trajectories  $x_1(t), x_2(t)$  of the closed-loop RTAC system.

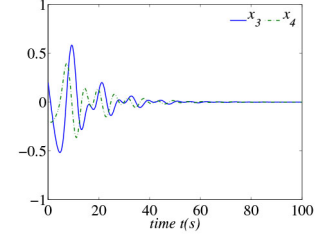


Fig. 6. State trajectories  $x_3(t), x_4(t)$  of the closed-loop RTAC system.

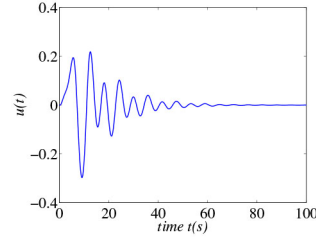


Fig. 7. Control trajectory  $u(t)$  of the closed-loop RTAC system.

sample size  $M = 300$  and generate random noise in interval  $[0, 0.5]$  as input signals. It is found that the critic NN weight vector converges fast to

$$\theta^{(3)} = [0.3285 \ 1.5877 \ 0.2288 \ -0.7028 \ 0.4101 \ -1.2514 \\ -0.5448 \ -0.4595 \ 0.4852 \ 0.2078 \ -1.3857 \ 1.7518 \\ 1.1000 \ 0.5820 \ 0.1950 \ -0.0978 \ -1.0295 \ -0.2773 \\ -0.2169 \ 0.2463]^T$$

at  $i = 3$  iteration. Figs. 3 and 4 show first 10 critic NN weights (i.e.,  $\theta_1^{(i)} \sim \theta_{10}^{(i)}$ ) at each iteration. With the convergent critic NN weight vector  $\theta^{(3)}$ , the  $H_\infty$  control policy can be computed with (22). Under the disturbance signal  $w(t) = 0.2r_1(t)e^{-0.2t}\cos(t)$ , ( $r_1(t) \in [0, 1]$  is a random number), closed-loop simulation is conducted with the  $H_\infty$  control policy. Figs. 5–7 give the trajectories of state and control policy. To show the relationship between  $L_2$ -gain and time, define the following ratio of disturbance attenuation as:

$$r_d(t) = \left( \frac{\int_0^t (\|z(\tau)\|^2 + \|u(\tau)\|_R^2) d\tau}{\int_0^t \|w(\tau)\|^2 d\tau} \right)^{\frac{1}{2}}.$$

Fig. 8 shows the curve of  $r_d(t)$ , where it converges to 3.7024 ( $< \gamma = 6$ ) as time increases, which implies that the designed  $H_\infty$  control law can achieve a prescribed  $L_2$ -gain performance level  $\gamma$  for the closed-loop system.

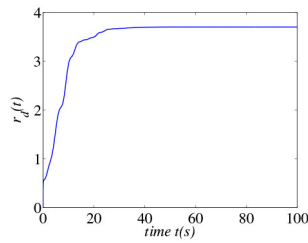


Fig. 8. Curve  $r_d(t)$  of the closed-loop RTAC system.

## VII. CONCLUSION

A NN-based off-policy RL method has been developed to solve the  $H_\infty$  control problem of continuous-time systems with unknown internal system model. Based on the model-based SPUA, an off-policy RL method is derived, which can learn the solution of HJI equation from the system data generated by arbitrary control and disturbance signals. The implementation of the off-policy RL method is based on an actor-critic structure, where only one NN is required for approximating the cost function, and then a least-square scheme is derived for NN weights update. The effectiveness of the proposed NN-based off-policy RL method is tested on a linear F16 aircraft plant and a nonlinear RTAC problem.

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable comments.

## REFERENCES

- [1] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Intell. Res.*, vol. 4, no. 1, pp. 237–285, Jan. 1996.
- [2] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
- [3] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Nashua, NH, USA: Athena Scientific, 2005.
- [4] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*. Hoboken, NJ, USA: Wiley, 2007.
- [5] J. J. Murray, C. J. Cox, G. G. Lendaris, and R. Saeks, "Adaptive dynamic programming," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 32, no. 2, pp. 140–153, May 2002.
- [6] V. Yadav, R. Padhi, and S. Balakrishnan, "Robust/optimal temperature profile control of a high-speed aerospace vehicle using neural networks," *IEEE Trans. Neural Netw.*, vol. 18, no. 4, pp. 1115–1128, Jul. 2007.
- [7] H. Zhang, Q. Wei, and Y. Luo, "A novel infinite-time optimal tracking control scheme for a class of discrete-time nonlinear systems via the greedy HDP iteration algorithm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 4, pp. 937–942, Aug. 2008.
- [8] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica*, vol. 45, no. 2, pp. 477–484, 2009.
- [9] D. Vrabie and F. L. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Netw.*, vol. 22, no. 3, pp. 237–246, 2009.
- [10] H. Zhang, Y. Luo, and D. Liu, "Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints," *IEEE Trans. Neural Netw.*, vol. 20, no. 9, pp. 1490–1503, Sep. 2009.
- [11] H. Zhang, L. Cui, X. Zhang, and Y. Luo, "Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2226–2236, Dec. 2011.
- [12] B. Luo and H.-N. Wu, "Online policy iteration algorithm for optimal control of linear hyperbolic PDE systems," *J. Process Control*, vol. 22, no. 7, pp. 1161–1170, 2012.
- [13] D. Vrabie, F. L. Lewis, and K. G. Vamvoudakis, *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*. Stevenage, England: IET Press, 2012.
- [14] D. Liu, D. Wang, D. Zhao, Q. Wei, and N. Jin, "Neural-network-based optimal control for a class of unknown discrete-time nonlinear systems using globalized dual heuristic programming," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 3, pp. 628–634, Jul. 2012.
- [15] B. Luo and H.-N. Wu, "Approximate optimal control design for nonlinear one-dimensional parabolic PDE systems using empirical eigenfunctions and neural network," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 6, pp. 1538–1549, Dec. 2012.
- [16] H.-N. Wu and B. Luo, "Heuristic dynamic programming algorithm for optimal control design of linear continuous-time hyperbolic PDE systems," *Ind. Eng. Chem. Res.*, vol. 51, no. 27, pp. 9310–9319, 2012.
- [17] D. Liu and Q. Wei, "Finite-approximation-error-based optimal control approach for discrete-time nonlinear systems," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 779–789, Apr. 2013.
- [18] Q. Wei and D. Liu, "A novel iterative  $\theta$ -adaptive dynamic programming for discrete-time nonlinear systems," *IEEE Trans. Autom. Sci. Eng.*, 2013, to be published.
- [19] D. Liu, D. Wang, and H. Li, "Decentralized stabilization for a class of continuous-time nonlinear interconnected systems using online learning optimal control approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 418–428, Feb. 2014.
- [20] B. Luo, H.-N. Wu, T. Huang, and D. Liu, "Data-based approximate policy iteration for nonlinear continuous-time optimal control design," *arXiv preprint arXiv:1311.0396*, 2013 [Online]. Available: <http://arxiv.org/abs/1311.0396>
- [21] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, "Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems," *Automatica*, vol. 50, no. 1, pp. 193–202, 2014.
- [22] D. Liu and Q. Wei, "Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 621–634, Mar. 2014.
- [23] B. Luo, H.-N. Wu, and H.-X. Li, "Data-based suboptimal neuro-control design with reinforcement learning for dissipative spatially distributed processes," *Ind. Eng. Chem. Res.*, 2014, to be published.
- [24] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. Upper Saddle River, NJ, USA: Prentice Hall, 1996.
- [25] A. V. D. Schaft,  *$L_2$ -Gain and Passivity in Nonlinear Control*. New York, NY, USA: Springer-Verlag, 1996.
- [26] T. Başar and P. Bernhard,  *$H_\infty$  Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*. Berlin, Germany: Springer, 2008.
- [27] A. V. D. Schaft, " $L_2$ -gain analysis of nonlinear systems and nonlinear state-feedback  $H_\infty$  control," *IEEE Trans. Autom. Control*, vol. 37, no. 6, pp. 770–784, Jun. 1992.
- [28] A. Isidori and W. Kang, " $H_\infty$  control via measurement feedback for general nonlinear systems," *IEEE Trans. Autom. Control*, vol. 40, no. 3, pp. 466–472, Mar. 1995.
- [29] A. Isidori and A. Astolfi, "Disturbance attenuation and  $H_\infty$ -control via measurement feedback in nonlinear systems," *IEEE Trans. Autom. Control*, vol. 37, no. 9, pp. 1283–1293, Sep. 1992.
- [30] R. W. Beard, "Successive Galerkin approximation algorithms for nonlinear optimal and robust control," *Int. J. Control*, vol. 71, no. 5, pp. 717–743, 1998.
- [31] M. Abu-Khalaf, F. L. Lewis, and J. Huang, "Policy iterations on the Hamilton–Jacobi–Isaacs equation for  $H_\infty$  state feedback control with input saturation," *IEEE Trans. Autom. Control*, vol. 51, no. 12, pp. 1989–1995, Dec. 2006.
- [32] K. G. Vamvoudakis and F. L. Lewis, "Online solution of nonlinear two-player zero-sum games using synchronous policy iteration," *Int. J. Robust Nonlinear Control*, vol. 22, no. 13, pp. 1460–1483, 2012.
- [33] Y. Feng, B. Anderson, and M. Rotkowitz, "A game theoretic algorithm to compute local stabilizing solutions to HJB equations in nonlinear  $H_\infty$  control," *Automatica*, vol. 45, no. 4, pp. 881–888, 2009.
- [34] D. Liu, H. Li, and D. Wang, "Neural-network-based zero-sum game for discrete-time nonlinear systems via iterative adaptive dynamic programming algorithm," *Neurocomputing*, vol. 110, no. 13, pp. 92–100, 2013.
- [35] N. Sakamoto and A. V. D. Schaft, "Analytical approximation methods for the stabilizing solution of the Hamilton–Jacobi equation," *IEEE Trans. Autom. Control*, vol. 53, no. 10, pp. 2335–2350, Nov. 2008.
- [36] G. N. Saridis and C.-S. G. Lee, "An approximation theory of optimal control for trainable manipulators," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 3, pp. 152–159, Mar. 1979.
- [37] R. W. Beard, G. N. Saridis, and J. T. Wen, "Galerkin approximations of the generalized Hamilton–Jacobi–Bellman equation," *Automatica*, vol. 33, no. 12, pp. 2159–2177, 1997.



- [38] R. W. Beard, G. Saridis, and J. Wen, "Approximate solutions to the time-invariant Hamilton–Jacobi–Bellman equation," *J. Optim. Theory Appl.*, vol. 96, no. 3, pp. 589–626, 1998.
- [39] S. Mehraeen, T. Dierks, S. Jagannathan, and M. L. Crow, "Zero-sum two-player game theoretic formulation of affine nonlinear discrete-time systems using neural networks," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1641–1655, Dec. 2013.
- [40] M. Abu-Khalaf, F. L. Lewis, and J. Huang, "Neurodynamic programming and zero-sum games for constrained control systems," *IEEE Trans. Neural Netw.*, vol. 19, no. 7, pp. 1243–1252, Jul. 2008.
- [41] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, "Online solution of nonquadratic two-player zero-sum games arising in the  $H_\infty$  control of constrained input systems," *Int. J. Adapt. Control*, vol. 28, nos. 3–5, pp. 232–254, Mar.–May 2014.
- [42] H. Zhang, Q. Wei, and D. Liu, "An iterative adaptive dynamic programming method for solving a class of nonlinear zero-sum differential games," *Automatica*, vol. 47, no. 1, pp. 207–214, 2011.
- [43] K. G. Vamvoudakis and F. L. Lewis, "Online actor–Critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, 2010.
- [44] B. Luo and H.-N. Wu, "Computationally efficient simultaneous policy update algorithm for nonlinear  $H_\infty$  state feedback control with Galerkin's method," *Int. J. Robust Nonlinear Control*, vol. 23, no. 9, pp. 991–1012, 2013.
- [45] J. Huang and C.-F. Lin, "Numerical approach to computing nonlinear  $H_\infty$  control laws," *J. Guid. Control Dynam.*, vol. 18, no. 5, pp. 989–994, 1995.
- [46] H.-N. Wu and B. Luo, "Simultaneous policy update algorithms for learning the solution of linear continuous-time  $H_\infty$  state feedback control," *Inf. Sci.*, vol. 222, pp. 472–485, Feb. 2013.
- [47] D. Vrabie and F. Lewis, "Adaptive dynamic programming for online solution of a zero-sum differential game," *J. Control Theory Appl.*, vol. 9, no. 3, pp. 353–360, 2011.
- [48] H.-N. Wu and B. Luo, "Neural network based online simultaneous policy update algorithm for solving the HJI equation in nonlinear  $H_\infty$  control," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 12, pp. 1884–1895, Dec. 2012.
- [49] H. Zhang, L. Cui, and Y. Luo, "Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network ADP," *IEEE Trans. Cybern.*, vol. 43, no. 1, pp. 206–216, Feb. 2013.
- [50] S. B. Thrun, "Efficient exploration in reinforcement learning," Carnegie Mellon University, Pittsburgh, PA, USA, Tech. Rep. CMU-CS-92-102, 1992.
- [51] R. Courant and D. Hilbert, *Methods of Mathematical Physics*. New York, NY, USA: Wiley, 2004.
- [52] B. A. Finlayson, *The Method of Weighted Residuals and Variational Principles: With Applications in Fluid Mechanics, Heat and Mass Transfer*. New York, NY, USA: Academic Press, Inc., 1972.
- [53] G. Peter Lepage, "A new algorithm for adaptive multidimensional integration," *J. Comput. Phys.*, vol. 27, no. 2, pp. 192–203, 1978.
- [54] J. A. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control: Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches*. New York, NY, USA: Wiley, 2006.
- [55] J.-J. E. Slotine *et al.*, *Applied Nonlinear Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1991.
- [56] D. Kleinman, "On an iterative technique for Riccati equation computations," *IEEE Trans. Autom. Control*, vol. 13, no. 1, pp. 114–115, Feb. 1968.
- [57] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, no. 5, pp. 779–791, 2005.
- [58] Z. Chen and S. Jagannathan, "Generalized Hamilton–Jacobi–Bellman formulation-based neural network control of affine nonlinear discrete-time systems," *IEEE Trans. Neural Netw.*, vol. 19, no. 1, pp. 90–106, Jan. 2008.
- [59] 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.
- [60] D. Precup, R. S. Sutton, and S. Dasgupta, "Off-policy temporal-difference learning with function approximation," in *Proc. 18th ICML*, 2001, pp. 417–424.
- [61] H. R. Maei, C. Szepesvári, S. Bhatnagar, and R. S. Sutton, "Toward off-policy learning control with function approximation," in *Proc. 27th ICML*, 2010, pp. 719–726.
- [62] M. Green and D. J. Limebeer, *Linear Robust Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1995.
- [63] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*. Hoboken, NJ, USA: Wiley-Interscience, 2003.
- [64] G. Escobar, R. Ortega, and H. Sira-Ramirez, "An  $L_2$  disturbance attenuation solution to the nonlinear benchmark problem," *Int. J. Robust Nonlinear Control*, vol. 8, nos. 4–5, pp. 311–330, 1999.



**Biao Luo** received the B.E. degree in measuring and control technology and instrumentations, and the M.E. degree in control theory and control engineering from Xiangtan University, Xiangtan, China, in 2006 and 2009, respectively. He is currently pursuing the Ph.D. degree in control science and engineering from Beihang University (Beijing University of Aeronautics and Astronautics), Beijing, China.

From February 2013 to August 2013, he was a Research Assistant at the Department of System Engineering and Engineering Management, City University of Hong Kong, Kowloon, Hong Kong. From September 2013 to December 2013, he was a Research Assistant at the Department of Mathematics and Science, Texas A&M University at Qatar, Doha, Qatar. His current research interests include distributed parameter systems, optimal control, data-based control, fuzzy/neural modeling and control, hypersonic entry/reentry guidance, learning and control from big data, reinforcement learning, approximate dynamic programming, and evolutionary computation.

Mr. Luo was a recipient of the Excellent Master Dissertation Award of Hunan Province in 2011.



**Huai-Ning Wu** was born in Anhui, China, on November 15, 1972. He received the B.E. degree in automation from Shandong Institute of Building Materials Industry, Jinan, China, and the Ph.D. degree in control theory and control engineering from Xi'an Jiaotong University, Xi'an, China, in 1992 and 1997, respectively.

From August 1997 to July 1999, he was a Post-Doctoral Researcher at the Department of Electronic Engineering, Beijing Institute of Technology, Beijing, China. In August 1999, he joined the School of Automation Science and Electrical Engineering, Beihang University (formerly Beijing University of Aeronautics and Astronautics), Beijing. From December 2005 to May 2006, he was a Senior Research Associate at the Department of Manufacturing Engineering and Engineering Management (MEEM), City University of Hong Kong, Kowloon, Hong Kong. From October to December during 2006–2008, and from July to August in 2010, he was a Research Fellow with the Department of MEEM, City University of Hong Kong. From July to August in 2011 and 2013, he was a Research Fellow with the Department of Systems Engineering and Engineering Management, City University of Hong Kong. He is currently a Professor with Beihang University. His current research interests include robust control, fault-tolerant control, distributed parameter systems, and fuzzy/neural modeling and control.

Dr. Wu serves as Associate Editor of the IEEE Transactions on Systems, Man & Cybernetics: Systems. He is a member of the Committee of Technical Process Failure Diagnosis and Safety, Chinese Association of Automation.



**Tingwen Huang** received the B.S. degree from Southwest Normal University (currently Southwest University), Chongqing, China, the M.S. degree from Sichuan University, Sichuan, China, and the Ph.D. degree from Texas A&M University, College Station, TX, USA, in 1990, 1993, and 2002, respectively.

He was a Visiting Assistant Professor at Texas A&M University. In August 2003, he joined Texas A&M University at Qatar as an Assistant Professor, where he was promoted to Professor in 2013. His current research interests include neural networks, chaotic dynamical systems, complex networks, optimization and control. He has authored and co-authored over 100 refereed journal papers.