Online Adaptive Optimal Control Algorithm Based on Synchronous Integral Reinforcement Learning With Explorations

Lei Guo¹⁰, Member, IEEE, and Han Zhao¹⁰, Student Member, IEEE

Abstract—In this paper, we present a novel algorithm named synchronous integral Q-learning, which is based on synchronous policy iteration, to solve the continuous-time infinite horizon optimal control problems of input-affine system dynamics. The integral reinforcement is measured as an excitation signal in this method to estimate the solution to the Hamilton–Jacobi–Bellman equation. Moreover, the proposed method is completely modelfree, i.e. no *a priori* knowledge of the system is required. Using policy iteration, the actor and critic neural networks can simultaneously approximate the optimal value function and policy. The persistence of excitation condition is required to guarantee the convergence of the two networks. Unlike in traditional policy iteration algorithms, the restriction of the initial admissible policy is relaxed in this method. The effectiveness of the proposed algorithm is verified through numerical simulations.

Index Terms—Synchronous integral reinforcement learning, Policy iteration, Persistence of excitation, Adaptive control.

I. INTRODUCTION

O PTIMAL control [1] and adaptive control [2] are two important concepts in modern control theory. The main goal of the optimal/adaptive controller is to reach the control objective with the minimal performance index/the unknown system structures or parameters. The method that combines the advantages of both methods is called reinforcement learning (RL, [3]) in the computational intelligence field or adaptive dynamic programming (ADP, [4]) in control theory (also known as approximate dynamic programming [5], neurodynamic programming [6] and adaptive critic design [7]), and it has been widely studied (See [8] for the latest survey on ADP).

The key problem of optimal control/ADPRL methods is how to solve the Hamilton–Jacobi–Bellman equation (HJBE) or Bellman equation, which is the discrete-time (DT) version of the HJBE and often used in the RL literature. The optimal policy and the corresponding representation of its quality, i.e. the value function (VF), can be solved from the HJBE. Owing to the phenomenon known as "curses of dimensionality" [9], the exact solution of the HJBE is usually difficult to find. The approximation method is often used, e.g. iterative methods using neural networks (NNs) [10]. The well-known actor-critic structure is generally used in ADPRL methods to simultaneously approximate the optimal policy and its VF.

Meanwhile, model-based methods may be difficult to implement in real-world control problems owing to the difficulties in mechanism modelling and the uncertainties of the dynamics system, which are called "curses of modelling" [6]. In studies on DT Markov decision process, model-free methods in ADPRL and deep RL based on deep NNs have achieved considerable success [11]–[13]. In the continuous-time (CT) domain, however, the effective methods in DT systems, e.g. action-dependent heuristic dynamic programming [14] or Qlearning [11], are difficult to implement because a priori knowledge and partial difference forms are required in the CT HJBE. [15] proposed an advantage updating algorithm to approximately compute the derivative of the VF. A modelfree estimation method of the VF was also proposed in [16]; however, an approximation or measurement of the differential term in these two methods is needed.

To solve the aforementioned problem, [17] proposed the concept of integral RL (IRL) and an algorithm to solve the CT optimal control problem of linear systems. The temporal difference (TD, [18]) estimation was introduced into the IRL algorithms by solving the integral form of the HJBE. The requirement that the system dynamics must be fully known is relaxed in [17]. Under the persistence of excitation (PE, [2]) condition, the VF of the current policy can be estimated in a model-free manner, and the drift dynamics of the system are not used in policy updates. However, satisfaction with the PE requirement cannot be guaranteed during the estimation of the parameters. [19] added the exploration signal to the input to excite the system and removed the restriction of the a priori knowledge of the input gain matrix. For nonlinear problems, [20] used an exploration method that was extended and improved in [21].

These aforementioned IRL algorithms are based on policy iteration (PI), which is an iterative method of dynamic programming (DP). The latest summary of the PI algorithms in ADP field can be found in [22]. To guarantee the convergence of the weights in NNs, the PI algorithm require an admissible controller at the beginning of the iteration. However, it is difficult to design one if the dynamics of the system are completely unknown. Furthermore, the weight updating method is in the least-squares sense, bringing a DT weight controller into the actual CT dynamics systems. [23] used the gradient descent method to solve the integral-TD (I-TD) equation and update the weights in NNs. The algorithm in [23] is called

This work was supported by National Natural Science Foundation of China under Grant 61105103. (Corresponding author: Lei Guo.)

The authors are with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, P.O. Box 108, No.10 Xi Tucheng Road, Haidian, Beijing 100876, China (e-mail: guolei@bupt.edu.cn; han_zhao@bupt.edu.cn).

synchronous IRL, and it is a partially model-free algorithm for solving nonlinear optimal control problems. Synchronous IRL is based on the concept of synchronous PI [24], which can be regraded as an extended implementation of general PI (GPI, [3]) (the value iteration method [25], [26] is also a special case of GPI). The initial admissible policy is not required in synchronous IRL; however, full information of the input gain matrix is still required. In [27], a Q-learning method for CT linear systems was proposed. This is a completely modelfree method that is implemented by estimating the Q function instead of the original VF.

For the optimal control problems of input-affine nonlinear systems, the application of the IRL algorithms is limited by several shortcomings. Focusing on these limitations, we propose a novel algorithm called synchronous integral Q-learning as a solution. Because of the combination of the exploration term and the synchronous learning structure, the actor and critic NNs can simultaneously and continuously update their weights to approximately solve the exploration-HJBE and guarantee the closed-loop stability and the convergence of NNs under the PE condition. The main contributions of this study are summarised as follows:

- The proposed algorithm is a completely model-free method that can estimate the parameters without requiring any *a priori* knowledge (except for the information that the system dynamics should be input-affine) or using an identifier NN [28].
- The initial admissible control policy in traditional PI methods is not needed owing to the characteristics of the synchronous IRL algorithm.
- The hybrid system structure is avoided in this algorithm because the weights are updated continuously.

The remainder of this paper is organized as follows. In Section II, the infinite horizon optimal control problem in CT input-affine nonlinear systems is formulated. The performance index used to evaluate the quality of a controller is presented, and the basic offline PI method and the model-free PI algorithm based on IRL and exploration are also introduced in this section. Section III provides the VF approximation design of our method and the online weight tuning law based on the actor-critic NNs. Then, the closed-loop stability and the convergence of NNs are proved. Numerical simulations that show the effectiveness of the proposed method are described in Section IV. Finally, Section V presents the conclusions of the study.

For the notations, we use ||X|| to denote the Euclidean norm, $\sqrt{X^{\top}X}$, of the vector or the Frobenius norm, $\sqrt{tr(X^{\top}X)}$, of matrix X. $X \otimes Y$ denotes the Kronecker product of matrices X and Y. The function of time, x(t), is also written as x_t or x, and the function of other variables, f(x), can be written as f in short.

II. OPTIMAL CONTROL PROBLEM AND PI ALGORITHMS

A. Problem formulation

Let us consider a CT input-affine nonlinear system:

$$\dot{x} = f(x(t)) + g(x(t))u(t) \quad x(0) = \xi,$$
(1)

where $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ are the fully observable state and the control input, respectively. ξ is the initial state of the system. Let us assume that f(x) + g(x)u is Lipschitz on compact set Ω and satisfies f(0) = 0.

We define the integral form of the infinite horizon performance index as

$$J(x,u) = \int_0^\infty r(x,u)d\tau,$$
 (2)

where $r(x, u) = S(x) + u^{\top} R u$ with S(x) > 0 and R > 0. In this study, our goal is to design an optimal control law u^* that stabilizes the system at x = 0 and minimizes the index (2). We use the following VF to represent the quality of a policy:

$$V^{\mu}(x) = J(x, u)|_{u=\mu(x)}, V^{\mu}(0) = 0,$$
(3)

where $\mu(x)$ is a feedback control law with $\mu(0) = 0$, and in the remainder of the paper, it is also called policy. With the admissibility of the policy, the VF of the policy is well-defined.

Definition 1: ([10], Admissible control) Policy $\mu(x)$ is said to be admissible on Ω , denoted by $\mu \in \mathcal{A}(\Omega)$, iff the following are satisfied:

1) This policy stabilizes (1) on Ω , i.e.

$$\lim_{t \to \infty} (f + g\mu) d\tau = 0. \tag{4}$$

2) $V^{\mu}(\xi)$ is bounded for any state $\xi \in \Omega$.

Here, Ω and $\mathcal{A}(\Omega)$ are the admissible region of (1) and the admissible control set, respectively.

Let us assume that the admissible control set $\mathcal{A}(\Omega)$ of system (1) is not empty and $V^{\mu} \in C^{1}(\Omega)$. According to Definition 1, it is easy to conclude that there exists an optimal control law, $\mu^{*}(x)$, such that

$$V^{\mu^*}(\xi) = \min_{\mu(x) \in \mathcal{A}(\Omega)} \int_0^\infty r(x(\tau), u(\tau)) d\tau$$

$$\leq V^{\mu}(\xi), \forall \xi \in \Omega.$$

It can be seen clearly that the optimal VF satisfies $V^*(\xi) = V^{\mu^*}(\xi)$. In Section II-B, we introduce several methods for solving the optimal VF. Without special instructions, the problem discussed in this paper is limited to a compact set, $x \in \Omega$.

B. HJBE and PI

According to the definition of the VF, the infinitesimal version of (3) can be obtained as

$$0 = r(x, \mu(x)) + (\nabla V^{\mu})^{\top} (f(x) + g(x)\mu(x)),$$
 (5)

where ∇V^{μ} denotes the gradient of V^{μ} and (5) is called the Lyapunov equation of the system (1). From (3) and (5), we can infer that

$$V^{\mu}(x) \ge 0, \tag{6}$$

$$\dot{V}^{\mu}(x) = -r(x,\mu) \le 0.$$
 (7)

Here, $V^{\mu}(x)$ is regarded as the Lyapunov function of system (1). The optimal control problem can be converted to an optimization problem under the constraint of the state equation. We define the Hamiltonian as follows:

$$H(x, u, \nabla V^{\mu}) = r(x, u) + \left(\nabla V^{\mu}\right)^{\top} (f(x) + g(x)u), \quad (8)$$

Algorithm 1 Offline PI

1. Initialization

Given the initial admissible policy, $\mu_0(x)$, set $i \leftarrow 0$. 2. **Policy Evaluation**

Solve the Lyapunov equation according to $\mu_i(x)$:

$$H(x, \mu_i(x), \nabla V^{\mu_i}) = 0$$

V^{*}(0) = 0. (13)

3. Policy Improvement

Update the control policy

$$\mu_{i+1} = \arg \min_{\mu \in \mathcal{A}(\Omega)} H(x, \mu, \nabla V^{\mu_i}).$$
(14)

For input-affine system (1), this policy can be explicitly represented as

$$\mu_{i+1} = -\frac{1}{2} R^{-1} g^{\top}(x) \nabla V^{\mu_i}(x).$$
(15)

4. Set $i \leftarrow i + 1$.

5. Repeat step 2-4 until convergence.

where ∇V^{μ} is also the Lagrange multiplier for this problem. For the optimal policy and its VF, the following Lyapunov equation is satisfied:

$$H(x, \mu^*, \nabla V^{\mu^*}) = 0.$$
 (9)

The optimal policy can be obtained by minimizing the Hamiltonian:

$$\mu^* = \arg\min_{\mu \in \mathcal{A}(\Omega)} H(x, \mu, \nabla V^{\mu^*}).$$
(10)

Owing to the input-affine characteristic of system (1), the optimal policy can be explicitly given as

$$\mu^* = -\frac{1}{2}R^{-1}g^{\top}(x)\nabla V^{\mu^*}(x).$$
(11)

Substituting (9) into (8), we can obtain the well-known HJBE:

$$0 = S(x) + (\nabla V^{\mu^*})^{\top}(x)f(x) - \frac{1}{4}(\nabla V^{\mu^*})^{\top}(x)g(x)R^{-1}g^{\top}(x)\nabla V^{\mu^*}(x)$$
(12)
$$V^*(0) = 0$$

With the linear system dynamics and the quadratic form of the performance index, i.e. the linear quadratic regulator (LQR) problem, the HJBE becomes the Riccati equation, which is relatively easy to solve. However, in the general nonlinear case, it is usually extremely difficult or even not possible to find the solution for the HJBE.

PI is a DP algorithm used to iteratively solve the optimal control problem by alternately taking two steps, namely, policy evaluation and policy improvement. The procedure for offline PI is shown in Algorithm II-B.

Remark 1: In the optimal control problem, the convergence of the PI can be guaranteed if the algorithm starts with an initial admissible policy. Under this condition, the convergence to the optimal policy and VF has been proven. See [10] for the detailed proof.

With regard to the LQR problem of linear time-invariant

systems, Algorithm II-B becomes the Kleinman algorithm [29]. In the case of high–order and complex nonlinear systems, the PI algorithm is still difficult to implement. The solution to (13) is often approximated by NNs [10], Galerkin approximation [30], and other approximation methods. The system dynamics need to be fully known in this algorithm.

C. IRL with explorations

[17] proposed an algorithm framework called the IRL. By integrating (7) into time interval [t - T, t], we can obtain the I-TD equation as

$$V^{\mu_i}(x(t-T)) = \int_{t-T}^t r(x,\mu_i) d\tau + V^{\mu_i}(x(t)).$$
(16)

Note that there is no *a priori* knowledge of the system in (16); the first term on the right-hand side of this equation can be collected online. For sufficient groups of integral data, the critic NN and the least-squares method can be used to approximate the computation of the solution to (13) and finish the policy evaluation. The policy can be updated by using (15), and thus, the requirement of the known system drift dynamics f(x) is dismissed.

Unlike in offline PI, the PE condition is required to guarantee the uniqueness of ∇V^{μ_i} . However, it cannot re-excite the system when the state has been stabilized at the origin. Thus, the convergence to the optimal solution may not be guaranteed in real-world implementations. [20] improved the policy evaluation step and solved the input-affine optimal control problem. By adding a bounded piecewise continuous nonzero probing signal e_{τ} , we can transform (1) into

$$\dot{x} = f(x) + g(x)(u+e).$$
 (17)

The online Lyapunov equation (16) can be obtained as follows after adding the term with e_{τ} :

$$V^{\mu_{i}}(x(t-T)) + \int_{t-T}^{t} (\nabla V^{\mu_{i}})^{\mathsf{T}} g(x) e_{\tau} d\tau$$

$$= \int_{t-T}^{t} r(x,\mu_{i}) d\tau + V^{\mu_{i}}(x(t)).$$
(18)

Remark 2: Compared with the method in [17], this method does not require additional information on the system dynamics. The designed signal e_{τ} is added to ensure that the PE condition is satisfied without generating an estimation bias. The concept of the probing signal is equivalent to the exploration [3] in the RL literature.

By further substituting (15) into (18), we can obtain the following equation:

$$V^{\mu_{i}}(x(t-T)) - \int_{t-T}^{t} 2\mu_{i+1}^{\top} Re_{\tau} d\tau$$

$$= \int_{t-T}^{t} r(x,\mu_{i}) d\tau + V^{\mu_{i}}(x(t)).$$
(19)

Note that (19) can simultaneously evaluate and improve the present policy. During the iteration, no *a priori* knowledge of the system is required. If $e_{\tau} \equiv 0$, (19) is equivalent to (13). The exploration signal can both guarantee the PE condition and relax the requirement of g(x), making it a completely

model-free algorithm. However, two issues exist in this algorithm:

- Because of the nature of PI algorithms, both (18) and (19) still require an initial admissible policy; this might be difficult to implement when the system dynamics are partially or even completely unknown.
- The algorithm updates the VF and the policy based on the batch or recursive least-squares method, which brings a DT weight tuning controller to the CT system. The hybrid system structure increases the burden on the computing unit.

In Section III, we present a novel algorithm that combines the concepts of IRL, exploration, and synchronous RL to solve the aforementioned issues. We call this algorithm synchronous integral Q-learning because it is a GPI implementation of the algorithm in [21].

III. SYNCHRONOUS INTEGRAL REINFORCEMENT LEARNING BASED ON EXPLORATIONS

A. Synchronous integral Q-learning

Eq. (19) shows that the optimal policy and its corresponding VF satisfy

$$V^{\mu^{*}}(x(t-T)) - \int_{t-T}^{t} 2\mu^{*\top} Re_{\tau} d\tau$$

= $\int_{t-T}^{t} r(x,\mu^{*}) d\tau + V^{\mu^{*}}(x(t)).$ (20)

The exploration-HJBE can be approximately solved using the actor-critic NNs. First, we consider the VF approximation. We assume that the optimal VF can be denoted as an NN:

$$V^{\mu^*}(x) = w_c^{*\top} \phi_c(x) + \varepsilon_c(x), \qquad (21)$$

where $\phi_c : \mathbb{R}^n \to \mathbb{R}^{N_c}$, w_c^* and ε_c are the activation function, weight and reconstruction error of the NN, respectively. N_c is the number of hidden layers in the critic NN. Because ε_c is bounded on a compact set, the activation function can be selected properly to create a complete set of basis functions such that $V^*(x)$ and its gradient

$$\nabla V^{\mu^*} = \nabla \phi_c^\top w_c^* + \nabla \varepsilon_c \tag{22}$$

are uniformly approximated [31]. According to the Weierstrass high order approximation theorem [10], such a set of basis functions exists if the VF is sufficiently smooth. Moreover, ε_c and its gradient $\nabla \varepsilon_c$ are bounded when N_c is a constant and $\varepsilon_c \rightarrow 0$ uniformly when $N_c \rightarrow \infty$.

Similarly, the optimal policy can be approximated by an actor NN:

$$\mu^*(x) = -\frac{1}{2}R^{-1}g^{\top}(x)\nabla\phi_c^{\top}(x)w_c^* - \frac{1}{2}R^{-1}g^{\top}(x)\nabla\varepsilon_c \qquad (23)$$
$$= w_a^{*\top}\phi_a(x) + \varepsilon_a(x),$$

where $\phi_a : \mathbb{R}^n \to \mathbb{R}^{N_a}$, w_a^* and ε_a are similar to the parameters in the critic NN, which can also enable the actor

NN uniformly approximate the optimal policy. Using the actor critic NNs, we can define the approximation error of (20) as

$$\int_{t-T}^{t} (S(x) + u^{*\top} R u^*) d\tau$$

$$+ w_c^{*\top} \phi_c(x(t)) - w_c^{*\top} \phi_c(x(t-T)) \qquad (24)$$

$$+ \operatorname{col} \{w_a^*\}^{\top} \int_{t-T}^{t} 2\phi_a(x) \otimes (Re_{\tau}) d\tau \equiv \varepsilon_B.$$

By defining the integral reinforcement

$$p(x,u) = \int_{t-T}^{t} r(x_{\tau}, u_{\tau}) d\tau, \qquad (25)$$

we can write (24) as

$$\varepsilon_B - \rho = W^{*\top} \delta, \tag{26}$$

where $W^* = [w_c^{*\top}, \operatorname{col}\{w_a^*\}^\top]^\top$ and

ĥ

$$\begin{split} \delta &= [\delta_c^\top, \delta_a^\top]^\top \\ &= \operatorname{col}\left\{\phi_c(x)\big|_{t-T}^t, \int_{t-T}^t 2\phi_a(x) \otimes (Re_\tau)d\tau\right\} \end{split}$$

Under the assumption that f(x) + g(x)u is Lipschitz, the residual error ε_B is bounded on a compact set.

Remark 3: When $N_c, N_a \to \infty, \varepsilon_B \to 0$ uniformly.

B. Actor-critic networks and the weight tuning law

We use the critic and actor NNs to approximate the optimal VF and policy, respectively, according to (24), and we define the approximate exploration-HJBE as

$$\int_{t-T}^{t} \left(-S(x) - \phi_a^{\top}(x) w_a^* R w_a^{*\top} \phi_a(x) + \varepsilon_{HJB}(x) \right) d\tau$$

= $W^{*\top} \delta$, (27)

where $\varepsilon_{HJB}(x)$ is the approximation error arising from the NNs. Because the optimal weights w_c^* and w_a^* are unknown, a parameter estimation method is required. The estimation of the VF can be obtained as

$$\hat{V}(x) = \hat{w}_c^\top \phi_c(x), \tag{28}$$

and the estimation of the policy is

$$\hat{\mu}(x) = \hat{w}_a^{\mathsf{T}} \phi_a(x), \tag{29}$$

where \hat{w}_c and \hat{w}_a are the estimations of the parameters. The approximation Bellman error can be obtained from (24) as

$$E = \hat{W}^{\top} \delta + \rho, \qquad (30)$$

where $\hat{W} = [\hat{w}_c^{\top}, \operatorname{col}\{\hat{w}_a\}^{\top}]^{\top}$. To minimize the squared residual error

$$K = \frac{1}{2}E^{\top}E, \qquad (31)$$

we can use the gradient-based methods to update the weights of both the two NNs. By using the normalized gradient descent algorithm [2] and (27), we can obtain the weights tuning law

$$\dot{\hat{W}} = -\alpha \frac{\partial K}{\partial \hat{W}} = -\alpha \frac{\delta}{(1+\delta^{\top}\delta)^2} E,$$
(32)



Fig. 1. Control scheme of synchronous integral Q-learning algorithm.

where $\alpha > 0$ is the learning rate that determines the convergence speed of the parameters. The entire control scheme of the algorithm is shown in Fig. 1.

We define $\overline{\delta} = \delta/(1 + \delta^{T} \delta)$. Before analyzing the convergence of the parameters, we need to review the PE conditions in this section.

Definition 2: ([2], PE) At any given time, signal $\overline{\delta}$ is said to be persistently excited over interval [t - T, t] if there exist constants $\beta_1 > 0$ and $\beta_2 > 0$, such that

$$\beta_1 I \leq \int_{t-T}^t \overline{\delta}(\tau) \overline{\delta}^\top(\tau) d\tau \leq \beta_2 I, \qquad (33)$$

The PE condition is widely used in adaptive control and system identification methods to guarantee the convergence of the parameters.

Defining the estimation error of the weights as $\tilde{W} := W^* - \hat{W}$, we can express the error dynamics as

$$\begin{cases} \tilde{\tilde{W}} = -\alpha \overline{\delta} \cdot \overline{\delta}^{\top} \tilde{W} + \alpha \overline{\delta} \frac{\varepsilon_B}{m_s} \\ y = \overline{\delta}^{\top} \tilde{W} \end{cases}$$
(34)

where $m_s = 1 + \delta^{\top} \delta$. According to (34) and (33), we can obtain the following lemma.

Lemma 1: Assume that the control policy is admissible and that $\overline{\delta}$ is persistently excited for all t > 0. If the residual error satisfies $\|\varepsilon_B\| \le \varepsilon_{\max}$, the norm of the estimation error $\|\tilde{W}\|$ converges exponentially to a residual set:

$$\tilde{W} \le \frac{\sqrt{\beta_2 T}}{\beta_1} \{ [1 + \eta \beta_2 \alpha] \varepsilon_{\max} \}, \tag{35}$$

where η is a positive constant of the order of 1. **Proof.** See [24].

Lemma 1 proves that, under the admissible control condition, the weights can converge exponentially to a neighborhood of the optimal weights when the reconstruction error exists. This is important for evaluating the performance of the algorithm.

We assume the following.

Assumption 1: For a given compact set $\Omega \in \mathbb{R}^n$: a. $f(\cdot)$ is Lipschitz and $g(\cdot)$ is bounded by a constant

$$||f(x)|| < b_f ||x||, ||g(x)|| \le b_g$$

b. The reconstruction error of the NNs and the gradient of the critic NN error are bounded so that

$$\|\varepsilon_c\| < b_{\varepsilon_c}, \|\varepsilon_c\| < b_{\varepsilon_c} \\ \|\nabla \varepsilon_c\| < b_{\varepsilon_{cx}}.$$

c. The activation functions of the NNs and the gradients of the critic NN activation functions are bounded so that

$$\|\phi_{c}(x)\| < b_{\phi_{c}}, \|\phi_{a}(x)\| < b_{\phi_{a}} \\ \|\nabla\phi_{c}(x)\| < b_{\phi_{cx}}.$$

d. The optimal weights of the NNs are bounded so that

$$\|W^*\| < W^*_{\max}.$$

Theorem 1: Let all the assumptions in this paper hold, and let the tuning law and the parameters be selected as detailed in the proof. Then, there exists a number N_0 such that, for the number of hidden layer units of both the two NNs N_c , $N_a > N_0$, the closed loop system state and the NN approximation error \tilde{W} are uniformly ultimately bounded (UUB). **Proof.** See Appendix A.

C. Implementation of the algorithm for LQR problems

Let us consider the widely studied CT LQR problem, i.e. f(x) = Ax, g(x) = B, where A and B are matrices that do not depend on x. Specially, we define the performance index as $J = x^{T}Sx + u^{T}Ru$ with S > 0. According to the basic LQR theory, the optimal VF is quadratic to x and the optimal policy is the linear feedback control of x

$$V^* = w_c^{*\top} \phi_c(x) = x^{\top} P^* x,$$

$$\mu^* = w_a^{*\top} x = -R^{-1} B^{\top} P^* x,$$

where $\phi_c(x) = x \otimes x$. The exploration-HJBE (20) in the LQR problem becomes

$$\int_{t-T}^{T} (-x^{\top}Sx - x^{\top}w_a^*Rw_a^{*\top}x)d\tau = W^{*\top}\delta.$$
 (36)

Note that the approximation error $\varepsilon_{HJB}(x)$ does not occur in (36). Similarly, the approximation NNs can be written as

$$\hat{V}(x) = \hat{w}_c^{\top}(x \otimes x), \hat{\mu}(x) = \hat{w}_a^{\top}x.$$

Then, using the weight tuning law (32), we can solve the LQR problem online. The policy is also globally optimal for linear systems, and the approximation error is guaranteed to converge exponentially to zero owing to the non-existence of the reconstruction error of the NNs.

Remark 4: For Linear systems, the exploration can be chosen as a sum of sinusoidal signals that have sufficient richness (the number of the frequency components must be larger than or equal to the number of estimated parameters) to satisfy the PE condition. However, in nonlinear problems, no verifiable method exists to ensure that [23].

Remark 5: After the exploration signal is added, both the actor and the critic NN can update their weights by solving the same equation and the state-value function is approximated in this algorithm instead of directly estimating the Q function. Thus, the proposed method is different from the Q-learning approaches in [27], [32], [33].

IV. NUMERICAL SIMULATIONS

To show the effectiveness of the proposed method, we set a second order nonlinear system as a benchmark, which has been used in several studies [21], [24], [28]. The system dynamics are as follows:

$$f(x) = \begin{bmatrix} -x_1 + x_2 \\ -0.5x_1 - 0.5x_2(1 - (\cos(2x_1) + 2)^2) \end{bmatrix}, \quad (37)$$

$$g(x) = \begin{bmatrix} 0\\ \cos(2x_1) + 2 \end{bmatrix}.$$
 (38)

The cost function is selected as

$$S(x) = x_1^2 + x_2^2, R = 1.$$

According to the converse HJB approach [34], the optimal VF and policy can be respectively obtained as

$$V^*(x) = \frac{1}{2}x_1^2 + x_2^2 \tag{39}$$

and

$$\mu^*(x) = -(\cos(2x_1) + 2)x_2. \tag{40}$$

Here, we present two cases of this example to show the approximation performance of the two NNs.

A. Case 1: exact parameterization

Now, let us assume that the VF and the policy are parameterized exactly. In this case, the algorithm is used to estimate the parameters in a grey-box fashion. We choose the activation function as follows:

$$\phi_c(x) = [x_1^2, x_1 x_2, x_2^2]^{\top}, \phi_a(x) = [x_1 \cos(2x_1), x_1, x_2 \cos(2x_1), x_2]^{\top}$$

The optimal weights can be obtained from (39) and (40) and are

$$w_c^* = [0.5, 0, 1]^\top,$$

 $w_a^* = [0, 0, -1, -2]^\top.$
(41)

We choose the initial state as $x(0) = [0,0]^{\top}$ and the initial weights of the NNs as $\hat{w}_c(0) = [1,1,1]^{\top}$ and $\hat{w}_a(0) = [0.5, -0.5 - 0.5, -0.5]^{\top}$. The learning rate is set as $\alpha = 1000$.

The design of the exploration signal determines the level of excitation, which also affects the performance of the algorithm. In this case we choose the exploration signal as

$$e(t) = \sum_{k=1}^{100} \sin(\omega_k t),$$

where ω_k is uniformly sampled from [-50, 50]. The exploration is added to $t \in [0, 90]$. After 90 s, the exploration is ended and the simulation stops at $t_f = 100$ s. The length of the sampling interval is T = 0.025 s. The trajectories of x_1 and x_2 are shown in Fig. 2. After the exploration is stopped, the state can be stabilized near the origin.

As shown in Figs. 3 and 4, all the weights in the critic and actor NNs are close to the optimal value. After 100 s of training, the weights of the two NNs converge to

$$\hat{w}_c(t_f) = [0.5000, -0.0001, 1.0000]^{\top},$$

 $\hat{w}_a(t_f) = [0.0000, 0.0001, -1.0000, -2.0000]^{\top},$



Fig. 2. Case 1: trajectories of (a) x_1 and (b) x_2 .



Fig. 3. Case 1: evolution of the critic weights.

which are extremely close to the optimal value (41). Figs. 5 and 6 show the approximation errors of the critic and actor NNs, respectively. In the region of $x_1, x_2 \in [-1, 1]$, the maximum approximation error of the VF is approximately 10^{-4} and that of the policy is approximately 5×10^{-5} , indicating the excellent approximation performance of the trained NN.

B. Case 2: fully unknown dynamics

In case 1, the policy is assumed to satisfy the condition of the exact parameterization, which cannot be generalized to the case in which the information on the system is completely unknown. In case 2, we choose the following activation function to approximate the optimal policy:

$$\phi_a(x) = [x_1, x_1^2, \dots, x_1^5, x_2, x_1x_2, \dots, x_1^4x_2]^{\top}.$$

In the neighborhood of the origin, the optimal weight of the policy can be obtained as

$$\phi_a(x) = [0, 0, 0, 0, 0, -3, 0, 2, 0, -2/3]^{\top}$$



Fig. 4. Case 1: evolution of the actor weights.



Fig. 5. Case 1: approximation error of the critic network.

because the Taylor expansion of $cos(2x_1) + 2$ at $x_1 = 0$ is

$$\cos(2x_1) + 2 = 3 - 2x_1^2 + \frac{2}{3}x_1^4 + O(x_1^6)$$

After the training, the weights converge to

_

$$\hat{w}_c(t_f) = [0.5007, 0.0011, 0.9997]^+,$$

 $\hat{w}_a(t_f) = [-0.0021, -0.0007, 0.0040, 0.0016, -0.0001, -3.0011, 0.0023, 1.9986, 0.0040, -0.5758]^+.$

The approximation errors of the optimal VF and policy are shown in Figs. 7 and 8, respectively. The errors of both the NNs are less than 10^{-2} .

Remark 6: Because of the existence of the reconstruction error and the different structures of the actor NN between case 1 and case 2, the results of case 2 are worse but can show the convergence of the algorithm.

Remark 7: Compared with similar methods, the algorithm proposed in this paper does not require an extra identifier NN [28]. In addition, case 1 shows that our method obtains a



Fig. 6. Case 1: approximation error of the actor network.

Value Function Approximation Error



Fig. 7. Case 2: approximation error of the critic network.



Fig. 8. Case 2: approximation error of the actor network.

smaller approximation error than does the method in [24] for the same training time.

V. CONCLUSIONS

In this paper, we presented a novel algorithm using the concepts of IRL and synchronous RL to solve the CT optimal control problems. It does not require any *a priori* knowledge or an identifier NN. Moreover, an admissible control is not needed for its implementation. The design of the exploration to achieve safe learning is a meaningful future research direction. In [21], the invariant exploration method is implemented in the PI algorithm; however, it has not been proven to guarantee stability in the GPI method. The extension of our method to multi-agent or nonaffine nonlinear control problems is also worth investigating. In addition, it is important to explore the application of the proposed method to real-world high-order systems, e.g. in designing the controller for robots and aircraft.

APPENDIX Proof of Theorem 1

We define the approximation errors $\tilde{w}_c = w_c^* - \hat{w}_c$ and $\tilde{w}_a = w_a^* - \hat{w}_a$ and consider the Lyapunov function,

$$L = V^*(x) + \frac{1}{2}\tilde{W}^{\top}\tilde{W}, t \ge 0.$$
 (42)

The derivative of (42) to time t is

$$\dot{L} = \dot{V}^*(x) + \frac{1}{2}\tilde{W}^\top \dot{\tilde{W}},\tag{43}$$

Substituting the error dynamics (34), we can obtain the derivative as

$$\dot{L} = \nabla V^{*\top}(x)(f(x) + g(x)\hat{w}_a^{\top}\phi_a(x) + e) - \alpha \tilde{W}^{\top}\overline{\delta} \cdot \overline{\delta}^{\top}\tilde{W}.$$
(44)

Eq. (44) can be written as two terms, i.e. $\dot{L} = L_1 + L_2$, where

$$L_1 = \nabla V^{*\top}(x) \left(f(x) + g(x) \hat{w}_a^{\top} \phi_a(x) + e \right),$$
(45)

$$L_2 = -\alpha \tilde{W}^{\mathsf{T}} \overline{\delta} \cdot \overline{\delta}^{\mathsf{T}} \tilde{W}. \tag{46}$$

The first term is

$$L_{1} = w_{c}^{*\top} \left(\nabla \phi_{c}(x) f(x) + \nabla \phi_{c}(x) g(x) w_{a}^{*\top} \phi_{a}(x) \right) - \nabla \phi_{c}(x) g(x) \tilde{w}_{a}^{\top} \phi_{a}(x) + \nabla \phi_{c}(x) g(x) e \right) + \varepsilon_{1}(x),$$

$$(47)$$

where

$$\varepsilon_1(x) = \nabla \varepsilon_c^{\top} \left(f(x) + g(x) w_a^{*\top} \phi_a(x) + g(x) e -g(x) \tilde{w}_a^{\top} \phi_a(x) \right).$$
(48)

By substituting the exploration-HJBE (20), we can obtain

$$L_{1} = \left(-S(x) - \phi_{a}^{\top} w_{a}^{*} R w_{a}^{*\top} \phi_{a} + \varepsilon_{HJB} - w_{c}^{*\top} \nabla \phi_{c}(x) g(x) \tilde{w}_{a}^{\top} \phi_{a}(x)\right) + \varepsilon_{1}(x).$$

$$(49)$$

Because S(x) > 0, there exists matrix q on Ω such that $x^{\top}qx < S(x)$. Substituting q and the relationship between the two NNs, we can write the first term of \dot{L} as

$$L_{1} \leq \left(-x^{\top}qx - \phi_{a}^{\top}w_{a}^{*}Rw_{a}^{*\top}\phi_{a} + \varepsilon_{HJB} + (2\phi_{a}^{\top}w_{a}^{*}R + 2\varepsilon_{a}^{\top}R + \nabla\varepsilon_{c}^{\top}g(x))\tilde{w}_{a}^{\top}\phi_{a}\right) + \varepsilon_{1}(x).$$

$$(50)$$

Using Young's inequality, we can express (50) as

$$L_{1} \leq -\sigma_{\min}(q) \|x\|^{2} + \phi_{a}^{\top} \tilde{w}_{a} R \tilde{w}_{a}^{\top} \phi_{a} + \varepsilon_{HJB} + (2\varepsilon_{a}^{\top} R + \nabla \varepsilon_{c}^{\top} g(x)) \tilde{w}_{a}^{\top} \phi_{a} + \varepsilon_{1}(x).$$
(51)

We select proper N_0 such that $\sup_{x \in \Omega} \|\varepsilon_{HJB}\| < \varepsilon$. According to (48) and Assumption 1, we can obtain

$$L_{1} \leq -\sigma_{\min}(q) \|x\|^{2} + \sigma_{\max}(R) \|\tilde{w}_{a}^{\top}\phi_{a}\|^{2} + 2b_{\varepsilon_{a}}\sigma_{\max}(R) \|\tilde{w}_{a}^{\top}\phi_{a}\| + \varepsilon + b_{\varepsilon_{c}} \left(b_{f} \|x\| + b_{g}b_{\phi_{a}} \|w_{a}^{*}\| + b_{g}\|e\|\right).$$
(52)

By using the characteristics of the norm, we can write (52)

$$L_{1} \leq -\sigma_{\min}(q) \|x\|^{2} + \sigma_{\max}(R) b_{\phi_{a}}^{2} \|\tilde{W}\|^{2}$$

+ $2b_{\varepsilon_{a}} \sigma_{\max}(R) b_{\phi_{a}} \|\tilde{W}\| + b_{\varepsilon_{c}} b_{f} \|x\|$
+ $\varepsilon + b_{\varepsilon_{c}} (b_{g} b_{\phi_{a}} \|w_{a}^{*}\| + b_{g} \|e\|).$ (53)

According to the proof of Lemma 1, the second term satisfies

$$L_{2} \leq -\alpha \left\| \frac{\delta}{m_{s}} \right\|^{2} \|\tilde{W}\|^{2} + \alpha \left\| \frac{\delta}{m_{s}} \right\| \left\| \frac{\varepsilon_{2}}{m_{s}} \right\| \|\tilde{W}\|, \quad (54)$$

where

as

$$\varepsilon_2(x) = \nabla \varepsilon_c^{\top} \left(f(x) + g(x) w_a^{*\top} \phi_a(x) + g(x) e \right).$$
(55)

We add (53) and (54) and then substitute (55) and inequality $\left\|\frac{\delta}{m^2}\right\| < 1$ so that

$$\begin{split} \dot{L} &\leq -\sigma_{\min}(q) \|x\|^{2} \\ &+ \left(\sigma_{\max}(R) b_{\phi_{a}}^{2} - \alpha \left\| \frac{\delta}{m_{s}} \right\|^{2} \right) \|\tilde{W}\|^{2} \\ &+ b_{\varepsilon_{cx}} b_{f} \|x\| \|\tilde{W}\| \\ &+ b_{\varepsilon_{c}} b_{f} \|x\| \\ &+ \left(2b_{\varepsilon_{a}} \sigma_{\max}(R) b_{\phi_{a}} + \alpha b_{\varepsilon_{cx}} b_{g} (b_{\phi_{a}} \|w_{a}^{*}\| + \|e\|) \right) \|\tilde{W}\| \\ &+ \varepsilon + b_{\varepsilon_{c}} \left(b_{g} b_{\phi_{a}} \|w_{a}^{*}\| + b_{g} \|e\| \right). \end{split}$$

$$(56)$$

Let

$$a = \sigma_{\max}(R)b_{\phi_a}^2 - \alpha \left\| \frac{\delta}{m_s} \right\|^2,$$

$$c = b_{\varepsilon_c}b_g \left(b_{\phi_a} \| w_a^* \| + \| e \| \right),$$

and

$$\begin{split} d &= \left[\begin{array}{c} b_{\varepsilon_c} b_f \\ 2b_{\varepsilon_a} \sigma_{\max}(R) b_{\phi_a} + \alpha b_{\varepsilon_{cx}} b_g(b_{\phi_a} \| w_a^* \| + \| e \|) \end{array} \right]. \\ \tilde{Z} &= \left[\begin{array}{c} \| x \| \\ \| \tilde{W} \| \end{array} \right], \end{split}$$

Then, (56) becomes

Ĺ

$$\leq -\tilde{Z}^{\top}M\tilde{Z} + d^{\top}\tilde{Z} + c + \varepsilon, \qquad (57)$$

...2

where

$$M = \begin{bmatrix} \sigma_{\max}(q) & -\frac{b_{\varepsilon_{cx}}b_f}{2} \\ -\frac{b_{\varepsilon_{cx}}b_f}{2} & -a \end{bmatrix}.$$

To let *M* be a positive definite matrix, we choose a sufficiently large learning rate α if $||\delta|| \neq 0$. The norm of δ can easily maintain a non-zero value under the PE assumption and

a proper value of T during the learning phase.

$$det(M) = -a\sigma_{\max}(q) - \frac{b_{\varepsilon_{cx}}^2 b_f^2}{4} > 0.$$
 (58)

Then (57) becomes

$$\dot{L} \le -\sigma_{\min}(M) \|\tilde{Z}\|^2 + \|d\| \|\tilde{Z}\| + c + \varepsilon, \qquad (59)$$

According to (59), the Lyapunov function is negative if

$$\|\tilde{Z}\| > \frac{\|d\|}{2\sigma_{\min}(M)} + \sqrt{\frac{\|d\|^2}{4\sigma_{\min}^2(M)}} + \frac{c+\varepsilon}{\sigma_{\min}(M)} \equiv B_Z.$$
(60)

The inequality shows that \dot{L} is negative if L exceeds a certain bound. Then, according to the Lyapunov analysis, the state and the weights are UUB. Under the ideal condition, i.e. $N_c, N_a \rightarrow \infty$ or both the optimal VF and the corresponding policy are under the exact parameterization assumption, and the state and the approximation error are stabilized at the origin.

This completes the proof.

REFERENCES

- [1] F. Lewis and V. Syrmos, Optimal Control. John Wiley, 1995.
- [2] P. Ioannou and B. Fidan, Adaptive control tutorial. SIAM, 2006.
- [3] R. Sutton and A. Barto, *Reinforcement learning: an introduction*. Cambridge University Press, 1998.
- [4] P. Werbos, "Beyond regression: new tools for prediction and analysis in the behavioural sciences," Ph.D. dissertation, Harvard University, 1974.
- [5] —, "Advanced forecasting methods for global crisis warning and models of intelligence," *General Systems Yearbook*, vol. 22, pp. 25–38, 1977.
- [6] D. Bertsekas and J. Tsitsiklis, *Neuro-dynamic programming*. Athena Scientific, 1996.
- [7] D. Prokhorov and D. Wunsch, "Adaptive critic designs," *IEEE Transactions on Neural Networks*, vol. 8, pp. 997–1007, 1997.
- [8] D. Liu, S. Xue, B. Zhao, B. Luo, and Q. Wei, "Adaptive dynamic programming for control: a survey and recent advances," *IEEE Transactions* on Systems, Man, and Cybernetics: Systems, vol. 51, pp. 142–160, 2021.
- [9] R. Bellman, *Dynamic programming*. New Jersey: Princeton University Press, 1957.
- [10] M. Abu-Khalaf and F. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, pp. 779–791, 2005.
- [11] C. Watkins, "Learning from delayed rewards," Ph.D. dissertation, King's College of Cambridge, 1989.
- [12] T. Lillicrap, J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2016.
- [13] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017.
- [14] P. Werbos, "Neural networks for control and system identification," in Proceedings of IEEE conference on decision and control, 1989, pp. 260– 265.
- [15] L. B. III, "Reinforcement learning in continuous time: advantage updating," in *Proceedings of IEEE international conference on neural networks*, 1994, pp. 2448–2453.
- [16] D. Liu, S. Xue, B. Zhao, B. Luo, and Q. Wei, "Adaptive dynamic programming," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 32, pp. 140–153, 2002.
- [17] D. Vrabie, O. Pastravanu, F. Lewis, and M. Abu-Khalaf, "Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica*, vol. 45, pp. 477–484, 2009.
- [18] R. Sutton, "Learning to predict by the methods of temporal differences," *Machine Learning*, vol. 3, pp. 9–44, 1988.
- [19] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, pp. 2699–2704, 2012.

- [20] J. Lee, J. Park, and Y. Choi, "Integral reinforcement learning with explorations for continuous-time nonlinear systems," in *Proceedings of the 2012 International Joint Conference on Neural Networks (IJCNN)*, 2012, pp. 1–6.
- [21] —, "Integral reinforcement learning for continuous-time input-affine nonlinear systems with simultaneous invariant explorations," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, pp. 916–932, 2015.
- [22] J. Lee and R. Sutton, "Policy iterations for reinforcement learning problems in continuous time and space–fundamental theory and methods," *Automatica*, vol. 126, 2021.
- [23] K. Vamvoudakis, D. Vrabie, and F. Lewis, "Online adaptive algorithm for optimal control with integral reinforcement learning," *International Journal of Robust and Nonlinear Control*, vol. 24, pp. 2686–2710, 2013.
- [24] K. Vamvoudakis and F. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, pp. 878–888, 2010.
- [25] T. Bian and Z.-P. Jiang, "Value iteration, adaptive dynamic programming, and optimal control of nonlinear systems," in *Proceedings of IEEE* 55th Conference on Decision and Control (CDC), 2016, pp. 3375–3380.
- [26] —, "Reinforcement learning and adaptive optimal control for continuous-time nonlinear systems: a value iteration approach," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.
- [27] K. Vamvoudakis and F. Lewis, "Q-learning for continuous-time linear systems: a model-free infinite horizon optimal control approach," Systems & Control Letters, vol. 100, pp. 14–20, 2017.
- [28] S. Bhasin, R. Kamalapurkar, M. Johnson, K. Vamvoudakis, F. Lewis, and W. Dixon, "A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems," *Automatica*, vol. 49, pp. 82–92, 2013.
- [29] D. Kleinman, "On an iterative technique for Riccati equation computations," *IEEE Transactions on Automatic Control*, vol. 13, pp. 114–115, 1968.
- [30] R. Beard, G. Saridis, and J. Wen, "Galerkin approximations of the generalized Hamilton-Jacobi-Bellman equation," *Automatica*, vol. 33, pp. 2159–2177, 1997.
- [31] K. Hornik, M. Stinchcombe, and H. White, "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks," *Neural Networks*, vol. 3, pp. 551–560, 1990.
- [32] H. Lee, S.-H. Kim, and Y. Kim, "Policy gradient-based integral reinforcement learning for optimal control design of nonaffine morphing aircraft systems," in *Proceedings of the 28th Mediterranean Conference* on Control and Automation (MED), 2020, pp. 218–223.
- [33] A. Chen and G. Herrmann, "Adaptive optimal control via continuoustime Q-learning for unknown nonlinear affine systems," in *Proceedings* of IEEE 58th Conference on Decision and Control (CDC), 2019, pp. 1007–1012.
- [34] V. Nevistic and J. Primbs, "Constrained nonlinear optimal control: a converse HJB approach," California Institute of Technology, Pasadena, CA 91125, Tech rep. CIT-CDS 96-021, Tech. Rep., 1996.