2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

# Safe Q-learning for continuous-time linear systems

Soutrik Bandyopadhyay and Shubhendu Bhasin

*Abstract*— **Q-learning is a promising method for solving optimal control problems for uncertain systems without the explicit need for system identification. However, approaches for continuous-time Q-learning have limited provable safety guarantees, which restrict their applicability to real-time safety-critical systems. This paper proposes a safe Q-learning algorithm for partially unknown linear time-invariant systems to solve the linear quadratic regulator problem with user-defined state constraints. We frame the safe Q-learning problem as a constrained optimal control problem using reciprocal control barrier functions and show that such an extension provides a safety-assured control policy. To the best of our knowledge, Q-learning for continuous-time systems with state constraints has not yet been reported in the literature.**

## I. Introduction

Reinforcement learning(RL) has a strong inter-relationship with the theory of adaptive optimal control [1]. In particular, RL algorithms have seen reasonable success in solving continuous-time optimal control problems for systems with uncertain/unknown dynamics (see [2]–[6] and references therein for some examples). Stemming from the theory of dynamic programming for continuous-time systems, such approaches typically solve the Hamilton-Jacobi-Bellman(HJB) equations [7] under uncertain system dynamics by observing system trajectories. However, unlike its discrete-time counterpart, the Bellman equation, the HJB equation requires accurate knowledge of the system dynamics. Thus, solving HJB equations for continuous-time uncertain systems involve some degree of system identification to identify the unknown/uncertain system dynamics.

One promising approach to solving optimal control problems without exact knowledge of the system dynamics is Q-learning [8]. Inspired by algorithms for discrete-time Q-learning [9]–[13], significant research effort is directed towards extending Q-learning to continuous-time optimal control problems [14]–[20]. Such approaches have shown promising results in learning optimal control policies without needing to know the exact system dynamics. However, applying such algorithms to real-time safety critical systems is still an open challenge due to lack of safety guarantees.

Formally, the notion of safety of dynamical systems is the certification of forward invariance [21] of state and actuation constraint sets. Under this definition of safety, the safe RL problem is the mathematical construct to solve optimal control problems under user-defined state and actuation constraints. In the literature, some common approaches to ensure safety, include model predictive control (MPC) [22]–[24], reachability analysis [25], [26] and control barrier functions [27], [28] to name a few. Control barrier functions have gained popularity recently because they provide a Lyapunov-like analysis to study a system's safety without the need to compute the system trajectories.

In the literature, the state and input-constrained linear quadratic regulation (LQR) problem has been extensively studied [29]–[32]. Further, approaches that combine adaptive and optimal control theory to solve the LQR problem for uncertain systems have also been reported [33]–[36]. However, solving constrained LQR problems under uncertain system dynamics is still an open challenge.

A particular class of solutions for the constrained adaptive optimal control problem can be found in [37]–[40] where control policies are learned via a constrained approximate dynamic programming approach. However, all these approaches typically require an online system identification to identify uncertain system dynamics. This requirement for system identification comes at a price of increased computation complexity for these approaches. As discussed before, some continuous-time Q-learning approaches have shown the ability to learn optimal control policies without needing this online system identification and thus, they are computationally cheaper.

In the context of continuous-time Q learning, the authors of [41] have applied MPC to Q-learning in order to incorporate actuation constraints. The authors of [42] discuss an integral reinforcement learning technique with input constraints. Continuous-time Q-learning has been used for kino-dynamic motion planning in [43].

To the best of our knowledge, continuous-time Q-learning with state constraints has not been reported in the literature. In this paper, we propose a safe Q-learning algorithm to handle user-defined state constraints using reciprocal control barrier functions [27].

### A. Contributions

This work extends the continuous-time Q-learning framework to incorporate state constraints. The distinct advantage of the proposed method over constrained approximate dynamic programming approaches is that it does not require an explicit system identification step while safely learning the optimal control policy.

We first formulate the safe Q-learning problem as the constrained optimization of the Q function with respect to the control policy, subject to the constraint on the time derivative of a reciprocal control barrier function. We subsequently formulate the Lagrangian of the optimization problem and use an analytical solution to compute a constrained optimal

The authors are with the Department of Electrical Engineering, Indian Institute of Technology Delhi, New Delhi, India. {Soutrik.Bandyopadhyay,sbhasin}@ee.iitd.ac.in

control policy. We show that the proposed method bridges the gap between constrained adaptive optimal control and the ad-hoc method of safeguarding controllers [38]. We then extend the integral reinforcement learning technique to safely learn the optimal control policy online and show that the proposed method satisfies the user-defined state constraints.

### B. Mathematical notations used

In this paper, we use $\succcurlyeq$ and $\succ$ to denote square matrices' semi-definite and definite ordering, respectively. For a function $(\cdot) : \mathbb{R}^n \to \mathbb{R}$, $\nabla_x(\cdot)$ denotes $\frac{\partial(\cdot)}{\partial x}$. We use $\mathbb{N}_n$ to denote the set of all natural numbers up to and including $n$. We use $\lambda_{\max}(\cdot)$ and $\lambda_{\min}(\cdot)$ to refer to the maximum and minimum eigenvalues of a square matrix, respectively. We use $\|\cdot\|$ to denote the 2-norm for vectors and the corresponding induced norm for matrices. Additionally, $\text{tr}(\cdot)$ denotes the trace of a square matrix.

## II. Problem Formulation and Preliminaries

Consider the linear time-invariant system

$$\dot{x}(t) = Ax(t) + Bu(t), \;\; x(0) = x_0, \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the system state, $u(t) \in \mathbb{R}^m$ is the control input, $A \in \mathbb{R}^{n\times n}$ is the uncertain system matrix, and $B \in \mathbb{R}^{n\times m}$ is the input matrix. We assume that pair $(A, B)$ is controllable and $B$ is full rank and known. For the system in (1), we seek to solve the infinite-horizon linear quadratic regulation (LQR) problem by minimizing the cost functional

$$J(x(0), u) \triangleq \int_0^\infty c(x(\tau), u(\tau))d\tau, \tag{2}$$

with respect to the control policy $u$, where $c(x, u) \triangleq \frac{1}{2}x^\top Mx + \frac{1}{2}u^\top Ru$, with $M \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ being the state and input weighing matrices respectively. The matrix $M$ is positive semi-definite and the pair $(\sqrt{M}, A)$ is detectable. The control weighing matrix $R$ is positive definite. Additionally, we impose the following safety constraints on the state trajectory of the system

$$x(t) \in \mathcal{S} \;\; \forall\, t \in \mathbb{R}_{\geq 0}, \tag{3}$$

where the set $\mathcal{S}$ is a user-defined compact set containing the origin. In other words, the control policy must ensure the forward invariance of the set $\mathcal{S}$ [21]. For the rest of the paper, we suppress the time dependence of the signals $x(\cdot)$ and $u(\cdot)$ for notational brevity.

### A. Unconstrained optimal control

For the system in (1) and the cost functional in (2), the Hamiltonian [7] $H : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$, is defined as

$$H(x, u, \nabla_x V_s^*(x)) \triangleq c(x, u) + \nabla_x V_s^*(x)^\top (Ax + Bu), \tag{4}$$

where $V_s^*(x) : \mathbb{R}^n \to \mathbb{R}$ is the optimal value function defined as

$$V_s^*(x(t)) \triangleq \min_{u(\tau)\forall\tau\in\mathbb{R}_{\geq 0}} J(x(t), u) \tag{5}$$

The optimal control law for the unconstrained system is obtained by minimizing the Hamiltonian with respect to (w.r.t.) the control action for each state $x \in \mathbb{R}^n$, i.e., $u^*(x) = \arg\min_u H(x, u, \nabla_x V_s^*(x)) = -R^{-1}B^\top \nabla_x V_s^*(x)$. For the case of LTI systems, under quadratic integral cost functionals, it is well known that the value function is a quadratic function of the state [7], i.e., $V_s^*(x) = \frac{1}{2}x^\top Px$, where $P \in \mathbb{R}^{n\times n}$ is a unique positive definite symmetric matrix obtained by solving the algebraic Riccati equation (ARE)

$$A^\top P + PA - PBR^{-1}B^\top P + M = 0, \tag{6}$$

and the optimal control takes the form $u^*(x) = -R^{-1}B^\top Px$. The solution to the ARE in (6) and the corresponding optimal control law require complete knowledge of the system matrices $A$ and $B$. To solve optimal control problems for systems with uncertain/unknown dynamics, continuous-time approximate dynamic programming (ADP) approaches have been proposed in the literature [2], [3], [33].

### B. Continuous-time Q-learning

A notable approach for solving the ARE in a model-free setting is to define the so-called "$Q$-function" inspired by the field of reinforcement learning in discrete-time setting [8], [13], [44]. The advantage of this method is that the optimal control policy can be learned online from the state observations without needing to know the system dynamics.

In the present work, we define the function $Q : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ as [15], [45]

$$Q(x, u) \triangleq V_s^*(x) + H(x, u, \nabla_x V_s^*). \tag{7}$$

Substituting the value function from (5) and the Hamiltonian yields

$$Q(x, u) = \frac{1}{2}X^\top \overline{Q} X, \tag{8}$$

where $X \triangleq [x^\top \; u^\top]^\top$, $\overline{Q} \triangleq [Q_{11}, Q_{12}; Q_{21}, Q_{22}]$, $Q_{11} \triangleq PA + A^\top P + P + M$, $Q_{12} = Q_{21}^\top \triangleq PB$, and $Q_{22} \triangleq R$ are matrices of appropriate dimensions (cf. [15]). Based on this definition of the $Q$ function, the optimal control law $u^* : \mathbb{R}^n \to \mathbb{R}^m$, can be written as

$$u^*(x) = \arg\min_u Q(x, u) = -Q_{22}^{-1}Q_{21}x. \tag{9}$$

The expression in (9) offers a possible way to approximate the optimal control in a model-free way using the estimates of $Q_{22}$ and $Q_{21}$ [15]. In this paper, we extend the above formulation to incorporate user-defined safety constraints by utilizing Lyapunov-like control barrier functions.

### C. Control barrier functions

A versatile approach to ensure the safety of dynamical systems is via control barrier functions, which are Lyapunov-like functions used to provide safety certificates to control policies [27], [28], [46]. Specifically, let there exist a continuously differentiable function $h(x) : \mathbb{R}^n \to \mathbb{R}$, such that $\mathcal{S} = \{x \in \mathbb{R}^n : h(x) \geq 0\}, \text{Int}(\mathcal{S}) = \{x \in \mathbb{R}^n : h(x) > 0\}, \partial\mathcal{S} = \{x \in \mathbb{R}^n : h(x) = 0\}$. where $\text{Int}(\mathcal{S})$ and $\partial\mathcal{S}$ are non-empty sets defined as the interior and the boundary of the compact set $\mathcal{S}$, respectively. The function $h(x)$ is often referred to as the "zeroing" control barrier function

(ZCBF). In this paper, we consider another type of control barrier function, namely - reciprocal control barrier function (RCBF) [27] due to its similarities with Lyapunov functions. The RCBF is defined as

*Definition 1 (Reciprocal control barrier function [27]):* A continuously differentiable function $B_s(x) : \text{Int}(\mathcal{S}) \to \mathbb{R}$ is said to be a RCBF for the system in (1) if there exist class $\mathcal{K}$ functions $\alpha_1, \alpha_2, \alpha_3$ such that

$$1/\alpha_1(h(x)) \leq B_s(x) \leq 1/\alpha_2(h(x)), \quad (10)$$

$$\inf_u [\nabla_x B_s(x)^\top (Ax + Bu)] \leq \alpha_3(h(x)), \ \ \forall\, x \in \mathcal{S}. \quad (11)$$

Provided a valid RCBF $B_s(x)$ exists, a control policy $u(x) : \text{Int}(\mathcal{S}) \to \mathbb{R}^m$ satisfying

$$\nabla_x B_s(x)^\top [Ax + Bu(x)] \leq \gamma(1/B_s(x)) \ \ \forall\, x \in \text{Int}(\mathcal{S}), \quad (12)$$

for some class $\mathcal{K}$ function $\gamma(\cdot)$; renders the set $\mathcal{S}$ forward invariant for the system (1) [27]. We now use RCBF to ensure safe online training of the continuous-time Q-learning algorithm.

## III. Safe Q-Learning

We now detail the main contribution of the present work. The objective of the proposed safe Q-learning algorithm is to modify the optimal control policy of the unconstrained problem to ensure safety. Thus, we qualify the optimization problem in (9) by the safety constraint (12) and formulate the safe Q-learning problem as

$$u^*_{\text{safe}}(x) = \arg\min_u \quad Q(x, u), \quad (13a)$$

$$\text{s.t.} \quad \nabla_x B_s(x)^\top [Ax + Bu] \leq \gamma(1/B_s(x)), \quad (13b)$$

$$x(0) \in \text{Int}(\mathcal{S}), \quad (13c)$$

where $B_s(x)$ is a user-defined candidate Lyapunov-like barrier function for the constraint set $\mathcal{S}$ and $\gamma : \mathbb{R} \to \mathbb{R}$ is a class $\mathcal{K}$ function. Under the structure of value function in (5), we observe that the optimization problem outlined in (13) is convex in the decision variable $u$. We formulate the Lagrangian function $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ as

$$\mathcal{L}(x, u, \nu) = Q(x, u) + \nu[\nabla_x B_s^\top [Ax + Bu] - \gamma(1/B_s(x))], \quad (14)$$

where $\nu \in \mathbb{R}_{\geq 0}$ is the Lagrange multiplier. The optimal control for the constrained system can be obtained from $\frac{\partial \mathcal{L}}{\partial u} = 0$, as

$$u^*_{\text{safe}}(x) = -Q_{22}^{-1} Q_{21} x - \nu^*(x) R^{-1} B^\top \nabla_x B_s(x), \quad (15)$$

where $\nu^*(x) : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is the Lagrange multiplier derived from the Karush-Kuhn-Tucker(KKT) conditions [47, Section 5.3.3], defined as

$$\nu^*(x) = \max(C_b(x)/R_b(x), 0), \quad (16)$$

where $C_b(x) \triangleq -\nabla_x B_s(x)^\top B R^{-1} B^\top \nabla_x V_s^*(x) + \nabla_x B_s(x)^\top Ax + \gamma(1/B_s(x))$ (cf. [48], [49]), and $R_b(x) \triangleq \nabla_x B_s(x)^\top B R^{-1} B^\top \nabla_x B_s(x)$. We observe that the expression for the optimal Lagrange multiplier contains unknown terms of the system matrix $A$ and the matrix $P$ of the optimal value function. To make the control law implementable and to simplify the analysis, we estimate the Lagrange multiplier by a user-defined positive constant $\nu = k_{sb} \in \mathbb{R}_{>0}$. The certainty equivalence controller thus becomes

$$\hat{u}_{\text{safe}}(x) = -\hat{Q}_{22}^{-1} \hat{Q}_{21} x - k_{sb} R^{-1} B^\top \nabla_x B_s(x), \quad (17)$$

where $\hat{Q}_{21}$ and $\hat{Q}_{22}$ denote the online estimates for $Q_{21}$ and $Q_{22}$, respectively with appropriate dimensions.

*Remark 1:* The optimal Lagrange multiplier $\nu^*(\cdot)$ in (15) is a state-varying gain that switches between zero and $C_b(x)/R_b(x)$ depending upon the sign of $C_b(\cdot)$. If the first term of the control input is sufficient to ensure safety (i.e., satisfies (13b)) at a given state $x \in \text{Int}(\mathcal{S})$, then $C_b(x) \leq 0$ and consequently $\nu^*(x) = 0$ (this property of Lagrange multipliers is termed as complementary slackness, see [47]). Additionally, $\nu^*(\cdot)$ is non-zero when the first term of control input in (15) is unable to satisfy the constraint on its own. Thus, the second term also becomes active and $\nu^*(\cdot)$ provides a way to ensure safety. However, as discussed above, $\nu^*(\cdot)$ contains terms of unknown/uncertain matrices $A$ and $P$. To make the controller implementable, we approximate the multiplier $\nu^*(x)$ by a constant $k_{sb}$. Under this approximation, there is no way to switch-off the safety-inducing term (second term of (17)). Thus, the proposed approximate control law is only sub-optimal, with the optimality gap dependent on the choice of $k_{sb}$. Further, we show that the satisfaction of the safety constraint is not compromised under the approximation of the Lagrange multiplier by the constant $k_{sb}$.

*Remark 2:* The second term in (17) closely resembles a "safeguarding controller" coined in [38]. The developments in the present paper aim to bridge the gap between the ad-hoc approach of safeguarding controllers and the theory of constrained optimal control.

We now extend the actor-critic learning algorithm from [15] to learn the controller in (17) online.

### A. Actor-Critic based online learning

The optimal $Q(x, u^*)$ function from (8) can be parameterized as

$$Q(x, u^*) = \frac{1}{2} X^\top \overline{Q} X = \frac{1}{2} \text{vech}(\overline{Q})^\top \phi(X), \quad (18)$$

where $\text{vech}(\overline{Q}) \in \mathbb{R}^p$ with $p \triangleq (n+m)(n+m+1)/2$, denotes the half-vectorization of $\overline{Q}$ yielding a column vector containing the upper-triangular elements of $\overline{Q}$, where elements off the diagonal are considered to be $2\overline{Q}_{ij}$; and $\phi : \mathbb{R}^{n+m} \to \mathbb{R}^p$ denotes the quadratic basis function yielding a vector containing the elements $\{X_i X_j\}_{i \in \mathbb{N}_n, j \in \mathbb{N}_m}$. For notational brevity, we define $W_c \triangleq \frac{1}{2}\text{vech}(\overline{Q})$. Since the matrix $\overline{Q}$ is unknown, the vector $W_c$ and control law $u^*(\cdot)$ are unknown and unimplementable, and thus require corresponding estimators. The "critic" estimator approximating the $Q$ function is given by

$$\hat{Q}(x, \hat{u}_{\text{safe}}) = \hat{W}_c^\top \phi(X), \quad (19)$$

where $\hat{W}_c \in \mathbb{R}^p$ is the weight estimate for the critic and the estimated control law ("actor") is given by

$$\hat{u}_{\text{safe}}(x) = \hat{W}_a^\top x - k_{sb} R^{-1} B^\top \nabla_x B_s(x), \tag{20}$$

where $\hat{W}_a \in \mathbb{R}^{n\times m}$ is the actor weight estimate. The objective of the actor and critic components of the algorithm is to minimize the estimation errors defined as $\tilde{W}_a(t) \triangleq -Q_{12}Q_{22}^{-1} - \hat{W}_a(t)$ and $\tilde{W}_c(t) \triangleq W_c - \hat{W}_c(t)$ respectively. To learn the ideal weights $W_c$ and $W_a(\triangleq -Q_{12}Q_{22}^{-1})$, we write the fixed point optimality equation based on integral reinforcement learning [15] as

$$Q(x(t), u^*(t)) = Q(x(t-T), u^*(t-T)) - \int_{t-T}^{t} c(x,u)d\tau, \tag{21}$$

where $T \in \mathbb{R}_{>0}$ is a fixed time interval. The expression in (21) is the continuous-time equivalent of the Bellman optimality equation for discrete-time reinforcement learning. Now, under the uncertainties in the system matrix $A$ and the ARE solution $P$, we define the temporal difference (TD) error in the estimate of the $Q$ function as

$$\begin{aligned} e_c(t) \triangleq\ & \hat{Q}(x(t), \hat{u}(t)) - \hat{Q}(x(t-T), \hat{u}(t-T)) \\ & + \int_{t-T}^{t} c(x,\hat{u})d\tau = \hat{W}_c^\top \psi(t) + \int_{t-T}^{t} c(x,\hat{u})d\tau, \end{aligned} \tag{22}$$

where $\psi(t) \triangleq \phi(X(t)) - \phi(X(t-T))$. To learn the ideal weights online, we define the squared norm of the critic error as $\delta_c \triangleq \frac{1}{2}\|e_c\|^2$, and subsequently, write the gradient descent-based update law for the critic as

$$\dot{\hat{W}}_c(t) = -\eta_c \frac{\psi(t)}{(1+\psi(t)^\top \psi(t))^2} e_c(t), \tag{23}$$

where $\eta_c \in \mathbb{R}_{>0}$ is a user-defined gain. The update law for the actor is given by

$$\dot{\hat{W}}_a(t) = \text{proj}(-\eta_a(\hat{Q}_{21}(t)^\top \hat{Q}_{22}(t)^{-1} + \hat{W}_a)), \tag{24}$$

where the estimates $\hat{Q}_{22}(t)$ and $\hat{Q}_{21}(t)$ are extracted from $\hat{W}_c(t)$, proj$(\cdot)$ denotes the projection operator [50] that ensures $\|\hat{W}_a(t)\| \leq \overline{W}_a \ \ \forall\, t \in \mathbb{R}_{\geq 0}$ where $\overline{W}_a \in \mathbb{R}_{>0}$ is a user-defined bound and $\eta_a \in \mathbb{R}_{>0}$ is the user-defined actor gain. Since the update law for actor depends on the estimate of the critic, the critic's learning rate must be substantially larger.

*Assumption 1:* The signal $\frac{\psi(t)}{1+\psi(t)^\top\psi(t)}$ is persistently exciting (PE).

*Remark 3:* The update laws in (23) and (24) are implemented in continuous-time by maintaining a buffer of the past trajectory data and computing the signal $e_c$ accordingly.

### B. Safety and Stability Analysis

*Theorem 1:* For the system in (1) and under the critic and actor update laws in (23) and (24) respectively, the control law in (20) ensures that the state $x$, the actor and critic weight estimation errors ($\tilde{W}_a$ and $\tilde{W}_c$) are uniformly ultimately bounded (UUB), and the set $\mathcal{S}$ is forward invariant.

*Proof:* According to the definition of forward invariance [21], we initialize the state $x$ such that $x(0) \in \text{Int}(\mathcal{S})$. We now consider the positive definite candidate Lyapunov function $\mathcal{V} : \mathcal{D} \to \mathbb{R}$, where $\mathcal{D} \triangleq \text{Int}(\mathcal{S}) \times \mathbb{R}^{p+nm}$, defined as

$$\mathcal{V}(\zeta) = \frac{1}{2}x^\top P x + k_{sb} B_s(x) + \frac{1}{2}\|\tilde{W}_c\|^2 + \frac{1}{2}\text{tr}(\tilde{W}_a^\top \tilde{W}_a), \tag{25}$$

where $\zeta \triangleq [x^\top\ \tilde{W}_c^\top\ \text{vec}(\tilde{W}_a)^\top]^\top$ is the augmented state vector for the overall closed loop system. Using (10) one can show that there exist two class $\mathcal{K}$ functions $\alpha_l$, $\alpha_u$ such that $\alpha_l(\|\zeta\|) \leq \mathcal{V}(\zeta) \leq \alpha_u(\|\zeta\|) \ \ \forall\, \zeta \in \mathcal{D}$. In other words, $\mathcal{V}(\zeta)$ is a valid candidate Lyapunov function [51]. Computing the time derivative of $\mathcal{V}$ and substituting the control law (20) and the weight update laws (23), (24), we obtain the bound

$$\begin{aligned} \dot{\mathcal{V}}(\zeta) \leq\ & -\kappa_x\|x\|^2 - \kappa_c\|\tilde{W}_c\|^2 - \kappa_a\|\tilde{W}_a\|^2 \\ & - \kappa_b\|B^\top \nabla_x B_s(x)\|^2 + k_{sb}\overline{\mathcal{B}}\|B^\top \nabla_x B_s(x)\| + \iota_c, \end{aligned} \tag{26}$$

where $\kappa_c \triangleq \frac{1}{8}\eta_c$, $\kappa_a \triangleq \eta_a$, $\kappa_x \triangleq \frac{1}{2}\lambda_{\min}(M + Q_{12}Q_{22}^{-1}Q_{21})$, $\kappa_b \triangleq k_{sb}^2\lambda_{\min}(R^{-1})$, $\overline{\mathcal{B}} \triangleq \sup_{x\in\text{Int}(\mathcal{S})}\|Ax\|/\|B\| + \|\overline{W}_a x\|$, $\iota_c \triangleq \frac{2\eta_a^2\overline{W}_a^2\|R^{-1}\|^2}{\eta_c} + \sup_{x\in\text{Int}(\mathcal{S})}\frac{\overline{W}_a^2\|x\|^2}{2}$ are positive constants. Completing the squares, we write

$$\begin{aligned} \dot{\mathcal{V}}(\zeta) \leq\ & -\kappa_x\|x\|^2 - \kappa_c\|\tilde{W}_c\|^2 - \kappa_a\|\tilde{W}_a\|^2 \\ & - \frac{\kappa_b}{2}\|B^\top \nabla_x B_s(x)\|^2 + \iota, \end{aligned} \tag{27}$$

where $\iota \triangleq \frac{\overline{\mathcal{B}}^2}{2k_{sb}\lambda_{\min}(R^{-1})} + \iota_c$ is a finite positive constant. It can be shown that there exists a class $\mathcal{K}$ function $\alpha_v(\cdot)$ such that $\alpha_v(\|\zeta\|) \leq \kappa_x\|x\|^2 + \kappa_c\|\tilde{W}_c\|^2 + \kappa_a\|\tilde{W}_a\|^2 + \frac{\kappa_b}{2}\|B^\top\nabla_x B_s(x)\|^2$. We thus write $\dot{\mathcal{V}}(\zeta) \leq -\alpha_v(\|\zeta\|) + \iota$. Now, since $x(0) \in \text{Int}(\mathcal{S})$, $\mathcal{V}(\zeta(0))$ is a finite quantity. Additionally, we observe that $\dot{\mathcal{V}}(\cdot) < 0$ outside the compact set $\Omega_v \triangleq \{\zeta \in \mathcal{D} : \|\zeta\| \leq \alpha_v^{-1}(\iota)\}$. Thus using [51, Theorem 4.18] it can be shown that $\zeta$ is uniformly ultimately bounded (UUB). Since $x(0) \in \text{Int}(\mathcal{S}) \implies \mathcal{V}(\cdot) < \infty \ \ \forall\, t \in \mathbb{R}_{\geq 0}$, the RCBF $B_s(x(t)) < \infty \ \ \forall\, t \in \mathbb{R}_{\geq 0}$. By definition, at no point in time does the state trajectory intersect the boundary of the safe set $\partial\mathcal{S}$ [52]. Thus the state $x(t) \in \mathcal{S} \ \ \forall\, t \in \mathbb{R}_{\geq 0}$ and the set $\mathcal{S}$ is forward invariant for the system in (1). ■

*Remark 4:* The ultimate bound for the system depends on the term $\iota$ which can be reduced by choosing the critic gain $\eta_c$ to be much larger than $\eta_a$, and choosing an appropriate safety gain $k_{sb}$. However, upon increasing $k_{sb}$, it can be shown that the control effort required increases. Thus there exists a trade-off between the safety and control effort objectives.

## IV. Simulation Results

To demonstrate the safety and performance of the proposed control algorithm, we consider the linear system with $A = [0, 1; 1.6, 2.8]$ and $B = [0; 1]$. We seek to solve the linear quadratic regulator problem with the matrices $M = \mathbb{I}_2$ and $R = 0.1$. We impose a constraint on the norm of the state $x$, i.e., $\|x(t)\| \leq 1.5 \ \ \forall\, t \in \mathbb{R}_{\geq 0}$. To incorporate the constraint we construct the candidate RCBF as $B_s(x) = (\frac{1.5^2}{1.5^2 - x^\top x} - 1)^2$. The actor gain ($\eta_a$) was chosen to be 0.05, and the critic gain ($\eta_c$) was chosen to be 20. To enforce the safety constraints, the gain $k_{sb}$ was chosen to be 0.2.

The integration window $T$ was set to $0.01s$. We apply an exploration noise to the control input for the first $10s$ of the simulation to ensure sufficient excitation and enable state exploration.

Fig. 1(a) shows the state trajectory of the system under the influence of the proposed control law starting from the initial condition $x_0 = [1; 1]$. We observe that the proposed controller meets the regulation objectives and brings the state to the origin within approximately 10 seconds after removing the exciting signal. Fig. 1(b) shows the plot for the actor weight estimate ($\hat{W}_a$). We observe that the estimated actor weight converges close to the true control gain ($W_a$). However, there is a small steady-state error in the estimate $\hat{W}_{a2}$ (i.e., the weight corresponding to $x_2$). The plot for the norm of the state $x$ is shown in Fig. 1(c) along with the plot corresponding to the controller in [15]. We observe that during the initial phase of the online RL training, controller from [15] violates the safety constraint, whereas the proposed controller meets the safety constraints at all times. We now study the effect of variation of the safety

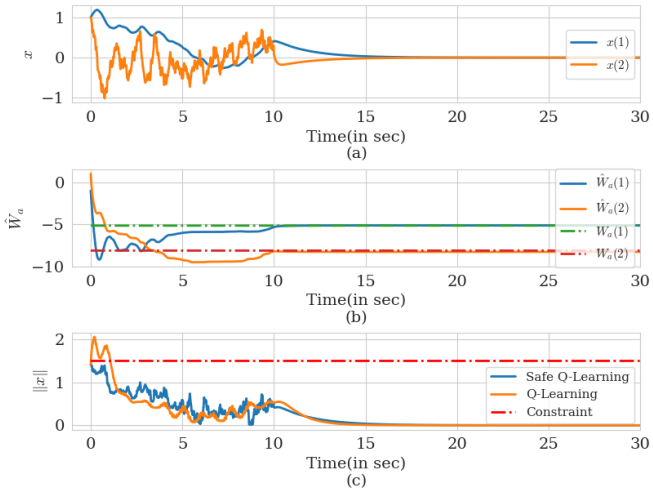


Fig. 1. (a) State Trajectory for the proposed algorithm (b) Estimated actor weights ($\hat{W}_a$) compared with the true control gains ($W_a$) (c) Plot of the norm of state $x$ compared with algorithm from [15].

TABLE I

PERFORMANCE ANALYSIS UNDER DIFFERENT SAFETY GAINS

| Safety gain $k_{sb}$ | 0.01 | 0.1 | 0.2 | 0.3 | 0.5 |
|---|---|---|---|---|---|
| Total Cost | 43.652 | 40.631 | 40.021 | 39.833 | 39.293 |
| Peak Control effort | 18.746 | 18.45 | 18.39 | 24 | 40 |

gain $k_{sb}$ on the performance of the proposed algorithm. To demonstrate the same, we simulate the linear system under different values of $k_{sb}$ starting from the same initial condition $x_0 = [1; 1]$. The exploration signal for all the cases was kept the same to enable a fair comparison. The plot for the norm of the state, along with the constraint bound, is shown in Fig. 2. We observe that the state ventures closer to the constraint boundary upon decreasing the value of $k_{sb}$. In other words, upon increasing $k_{sb}$, the algorithm becomes more conservative.

The comparison of the cost and the peak control effort under different values of $k_{sb}$ is given in Table I. We observe that upon increasing $k_{sb}$, the total cost decreases, but the peak control effort required increases (although for lower values of $k_{sb}$, the exploration control input dictates the peak control effort). Thus we observe a trade-off between the cost and the control effort required. However, the safety constraint is never violated for all values of $k_{sb}$.

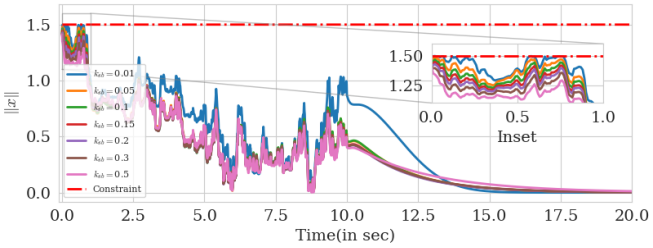


Fig. 2. The plot of the norm of the state $x$ for the proposed controller under different values of $k_{sb}$.

## V. CONCLUSIONS

In this paper, we propose a safe Q-learning algorithm utilizing reciprocal control barrier functions. Such an approach has the distinct advantage of learning optimal control policies for uncertain LTI systems with user-defined state constraints. We formulate the safe Q-learning problem as a constrained optimization problem involving a constraint on the time derivative of the RCBF. Subsequently, we derive adaptation laws based on integral reinforcement learning for the actor and critic estimators to estimate the constrained optimal control law online. We prove that under the proposed control law, the user-defined constraint set is forward invariant. Additionally, the state and the estimation errors are shown to be uniformly ultimately bounded via a Lyapunov analysis. We subsequently demonstrate the safety and stability performance in a simulation study. Future extensions to the present work could include extending the safe Q-learning algorithm to consider the dynamics of the Lagrange multiplier generated by the KKT conditions and extending the proposed algorithm to include both actuation and state constraints.

## REFERENCES


[1] R. S. Sutton, A. G. Barto, and R. J. Williams, "Reinforcement learning is direct adaptive optimal control," *IEEE control systems magazine*, vol. 12, no. 2, pp. 19–22, 1992.

[2] S. Bhasin, R. Kamalapurkar, M. Johnson, K. G. Vamvoudakis, F. L. Lewis, and W. E. Dixon, "A novel actor–critic–identifier architecture for approximate optimal control of uncertain nonlinear systems," *Automatica*, vol. 49, no. 1, pp. 82–92, 2013.

[3] 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.

[4] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, "Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers," *IEEE Control Systems Magazine*, vol. 32, no. 6, pp. 76–105, 2012.

[5] R. Kamalapurkar, J. A. Rosenfeld, and W. E. Dixon, "Efficient model-based reinforcement learning for approximate online optimal control," *Automatica*, vol. 74, pp. 247–258, 2016.

[6] R. Kamalapurkar, P. Walters, and W. E. Dixon, "Model-based reinforcement learning for approximate optimal regulation," *Automatica*, vol. 64, pp. 94–104, 2016.

[7] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal control*. John Wiley & Sons, third ed., 2012.

[8] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, pp. 279–292, 1992.

[9] H. Hasselt, "Double q-learning," *Advances in neural information processing systems*, vol. 23, 2010.

[10] J. N. Tsitsiklis, "Asynchronous stochastic approximation and q-learning," *Machine learning*, vol. 16, pp. 185–202, 1994.

[11] J. Abounadi, D. P. Bertsekas, and V. Borkar, "Stochastic approximation for nonexpansive maps: Application to q-learning algorithms," *SIAM Journal on Control and Optimization*, vol. 41, no. 1, pp. 1–22, 2002.

[12] S. Bhatnagar, R. S. Sutton, M. Ghavamzadeh, and M. Lee, "Natural actor–critic algorithms," *Automatica*, vol. 45, no. 11, pp. 2471–2482, 2009.

[13] S. Bradtke, "Reinforcement learning applied to linear quadratic regulation," *Advances in neural information processing systems*, vol. 5, 1992.

[14] S. K. Jha and S. Bhasin, "On-policy q-learning for adaptive optimal control," in *2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, pp. 1–6, IEEE, 2014.

[15] K. G. Vamvoudakis, "Q-learning for continuous-time linear systems: A model-free infinite horizon optimal control approach," *Systems & Control Letters*, vol. 100, pp. 14–20, 2017.

[16] P. Mehta and S. Meyn, "Q-learning and pontryagin's minimum principle," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pp. 3598–3605, IEEE, 2009.

[17] J. Kim, J. Shin, and I. Yang, "Hamilton-jacobi deep q-learning for deterministic continuous-time systems with lipschitz continuous controls," *The Journal of Machine Learning Research*, vol. 22, no. 1, pp. 9363–9396, 2021.

[18] J. Kim and I. Yang, "Hamilton-jacobi-bellman equations for q-learning in continuous time," in *Learning for Dynamics and Control*, pp. 739–748, PMLR, 2020.

[19] J. Y. Lee, J. B. Park, and Y. H. Choi, "Integral q-learning and explorized policy iteration for adaptive optimal control of continuous-time linear systems," *Automatica*, vol. 48, no. 11, pp. 2850–2859, 2012.

[20] M. Palanisamy, H. Modares, F. L. Lewis, and M. Aurangzeb, "Continuous-time q-learning for infinite-horizon discounted cost linear quadratic regulator problems," *IEEE transactions on cybernetics*, vol. 45, no. 2, pp. 165–176, 2014.

[21] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[22] M. Zanon and S. Gros, "Safe reinforcement learning using robust mpc," *IEEE Transactions on Automatic Control*, vol. 66, no. 8, pp. 3638–3652, 2020.

[23] K. P. Wabersich, L. Hewing, A. Carron, and M. N. Zeilinger, "Probabilistic model predictive safety certification for learning-based control," *IEEE Transactions on Automatic Control*, vol. 67, no. 1, pp. 176–188, 2021.

[24] Z. Li, U. Kalabić, and T. Chu, "Safe reinforcement learning: Learning with supervision using a constraint-admissible set," in *2018 Annual American Control Conference (ACC)*, pp. 6390–6395, IEEE, 2018.

[25] J. F. Fisac, A. K. Akametalu, M. N. Zeilinger, S. Kaynama, J. Gillula, and C. J. Tomlin, "A general safety framework for learning-based control in uncertain robotic systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 7, pp. 2737–2752, 2018.

[26] J. F. Fisac, N. F. Lugovoy, V. Rubies-Royo, S. Ghosh, and C. J. Tomlin, "Bridging hamilton-jacobi safety analysis and reinforcement learning," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 8550–8556, IEEE, 2019.

[27] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[28] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European control conference (ECC)*, pp. 3420–3431, IEEE, 2019.

[29] P. O. Scokaert and J. B. Rawlings, "Constrained linear quadratic regulation," *IEEE Transactions on automatic control*, vol. 43, no. 8, pp. 1163–1169, 1998.

[30] T. A. Johansen, I. Petersen, and O. Slupphaug, "On explicit suboptimal lqr with state and input constraints," in *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, vol. 1, pp. 662–667, IEEE, 2000.

[31] S. Dean, S. Tu, N. Matni, and B. Recht, "Safely learning to control the constrained linear quadratic regulator," in *2019 American Control Conference (ACC)*, pp. 5582–5588, IEEE, 2019.

[32] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[33] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.

[34] S. K. Jha, S. B. Roy, and S. Bhasin, "Data-driven adaptive lqr for completely unknown lti systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 4156–4161, 2017.

[35] S. K. Jha, S. B. Roy, and S. Bhasin, "Initial excitation-based iterative algorithm for approximate optimal control of completely unknown lti systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 12, pp. 5230–5237, 2019.

[36] H. Modares and F. L. Lewis, "Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning," *IEEE Transactions on Automatic control*, vol. 59, no. 11, pp. 3051–3056, 2014.

[37] M. H. Cohen and C. Belta, "Approximate optimal control for safety-critical systems with control barrier functions," in *2020 59th IEEE Conference on Decision and Control (CDC)*, pp. 2062–2067, IEEE, 2020.

[38] M. H. Cohen and C. Belta, "Safe exploration in model-based reinforcement learning using control barrier functions," *Automatica*, vol. 147, p. 110684, 2023.

[39] Z. Marvi and B. Kiumarsi, "Safe reinforcement learning: A control barrier function optimization approach," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 6, pp. 1923–1940, 2021.

[40] S. N. Mahmud, K. Hareland, S. A. Nivison, Z. I. Bell, and R. Kamalapurkar, "A safety aware model-based reinforcement learning framework for systems with uncertainties," in *2021 American Control Conference (ACC)*, pp. 1979–1984, IEEE, 2021.

[41] H. Zhang, S. Li, and Y. Zheng, "Q-learning-based model predictive control for nonlinear continuous-time systems," *Industrial & Engineering Chemistry Research*, vol. 59, no. 40, pp. 17987–17999, 2020.

[42] 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.

[43] G. P. Kontoudis and K. G. Vamvoudakis, "Kinodynamic motion planning with continuous-time q-learning: An online, model-free, and safe navigation framework," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 12, pp. 3803–3817, 2019.

[44] C. J. C. H. Watkins, "Learning from delayed rewards," 1989.

[45] K. G. Vamvoudakis, "Non-zero sum nash q-learning for unknown deterministic continuous-time linear systems," *Automatica*, vol. 61, pp. 274–281, 2015.

[46] A. J. Taylor and A. D. Ames, "Adaptive safety with control barrier functions," in *2020 American Control Conference (ACC)*, pp. 1399–1405, IEEE, 2020.

[47] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[48] H. Almubarak, E. A. Theodorou, and N. Sadegh, "Hjb based optimal safe control using control barrier functions," in *2021 60th IEEE Conference on Decision and Control (CDC)*, pp. 6829–6834, IEEE, 2021.

[49] X. Tan, W. S. Cortez, and D. V. Dimarogonas, "High-order barrier functions: Robustness, safety, and performance-critical control," *IEEE Transactions on Automatic Control*, vol. 67, no. 6, pp. 3021–3028, 2021.

[50] E. Lavretsky and K. A. Wise, "Robust adaptive control," in *Robust and adaptive control*, pp. 317–353, Springer, 2013.

[51] H. K. Khalil, "Nonlinear systems," *Prentice Hall, Upper Saddle River*, 2002.

[52] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier lyapunov functions for the control of output-constrained nonlinear systems," *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.