A Safety-Certified Policy Iteration Algorithm For Control of Constrained Nonlinear Systems

Navid Moshtaghi Yazdani, Reihaneh Kardehi Moghaddam, Bahare Kiumarsi, *Member, IEEE,* and Hamidreza Modares, *Senior Member, IEEE*

***Abstract*— This paper considers designing safe controllers with guaranteed performance for continuous-time systems under state constraints. While existing safe control frameworks mainly ignore the performance of the controller, and existing optimal control frameworks ignore the safety of the controller, the presented approach brings the best of the two worlds of safe control design and optimal control design together. To this end, a novel safe policy iteration algorithm is presented that iteratively finds an optimal controller while certifying the safety of the improved control policy at each iteration. To avoid solving a Hamilton-Jacobi Bellman (HJB) equality for finding the optimal control policy, which does not guarantee safety, an HJB inequality is solved that can be integrated with barrier certificates to verify the safety of the control policy. Sum-of-Squares program is employed to implement the presented policy iteration algorithms and thus iteratively find a safe control solution with guaranteed performance. A simulation example is provided to verify the effectiveness of the proposed approach.**

***Index Terms*— Optimal Safe Control, Systems Under State Constraints, HJB Equality, Sum-of-Squares Program.**

I. Introduction

With the ever-increasing demand for building the next-generation safety-critical systems that can safely interact with their environment (e.g., self-driving cars or assistive robots), there is an urgent need for developing safe controllers that respect systems constraints. Safety, however, is the bare minimum requirement for any safety-critical system, and it is mainly desired to design controllers that also satisfy desired performance specifications. Performance guarantee can be provided by solving an optimal control problem for which a pre-defined cost function is optimized. Despite the importance of designing safe optimal controllers, state-feedback safe control design and optimal control design are typically separated in the literature. This gap motivates this paper to develop safe controllers with a performance guarantee over a long horizon.

A. Related work

Finding optimal feedback controllers for general nonlinear systems requires solving the so-called Hamilton-Jacobi-Bellman (HJB) equations. Several results are presented to approximate solutions to HJB equations using Galerkin method [1] and neural networks [2]–[8]. However, system state constraints are not incorporated when solving HJB equations. Optimal control of systems with state constraints using Pontryagin's minimum principle [9] has also been widely considered. Nevertheless, solutions are usually computed in open-loop form which lack robustness and might be computationally infeasible to be implemented online. A state-feedback solution for constraint optimal control is provided in [10]. Although elegant, it is limited to linear systems and input constraints.

Model predictive control (MPC) [11] takes into account the system constraints while optimizing a performance function. However, since MPC uses short-horizon performance functions, it results in myopic control strategies for which it is hard to guarantee stability and feasibility of the solution. Optimal control of constrained systems using penalty functions in the performance function has also been considered [12]. However, these methods can only handle limited types of sate constraints (e.g. linear constraints). Reference governor [13] let the system apply a nominal performance-oriented control policy and modify the control action whenever it is not safe. This framework, however, does not consider safety in the design phase and the corrective controller can keep intervene with the nominal controller and jeopardize the system performance.

On the other hand, safety verification and safe control design using barrier certificate and control barrier function has also been widely and successfully employed to guarantee satisfaction of system's constraints [14]–[21]. Since safety and stability might be conflicting, in [22] a slag variable is added to the stability condition to relax stability and prioritize safe when needed. However, stability only guarantees asymptotic target reaching and does not guarantee an acceptable transient response for the system. Barrier certificate and optimal control are combined in [23]–[25] to find optimal safe controllers. However, the optimisation is over a finite horizon and must be solved for every initial condition to find the entire optimal state and input trajectories. To our knowledge, state-feedback safe optimal control design over a long horizon in the design phase is not considered in the literature.

B. Contributions and Outline

In this paper, designing state-feedback safe controllers with guaranteed performance for systems under state constraints is considered. A novel safe policy iteration algorithm

N. Moshtaghi Yazdani and R. Kardehi Moghaddam are with the Department of Electrical Engineering, Mashhad branch, Islamic Azad University, Mashhad, Iran e-mail: navid.moshtaghi@ut.ac.ir

Bahare Kiumarsi is with Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, USA email: kiumarsi@msu.edu.

Hamidreza Modares is with Department of Mechanical Engineering, Michigan State University, East Lansing, MI, USA email: modaresh@msu.edu

is presented that guarantees safety and performance during the design phase. The proposed policy iteration algorithm consists of two main parts: 1) a policy evaluation step that finds the value function corresponding to a safe policy and 2) a policy improvement step that finds a policy with improved performance for which its safety is verified by incorporating a control barrier function. Possible conflict between safety and stability is taken into account and the conflict is minimized at each iteration. To guarantee performance, a HJB inequality is solved, and is integrated with barrier certificate to assure safety. Sum-of-Squares program is employed to iteratively find an optimal safe control solution. A simulation example is provided to verify the effectiveness of the proposed approach.

The remaining of the paper is organized as follows: Section 2 formulates the problem and introduces some basic results regarding nonlinear optimal control. Section 3 presents a novel safe optimal framework. Section 4 provides numerical examples to validate the efficiency and effectiveness of the proposed method.

C. Notations

Let C^1 denote the set of all continuously differentiable functions. Then, $\mathcal{P}$ denotes the set of all functions in C^1 that are also positive definite and proper. A polynomial $p(x)$ is an Sum-of-Squares (SOS) polynomial, i.e, $p(x) \in \mathcal{P}^{SOS}$ where $\mathcal{P}^{SOS}$ is a set of SOS polynomial, if $p(x) = \sum_1^m p_i^2(x)$ where $p_i(x) \in \mathcal{P}$, $i = 1, 2, ..., m$. The function $K : \mathbb{R}^n \to \mathbb{R}$ is an extended class $\mathcal{K}$ function if it is a strictly increasing function and $K(0) = 0$. ∇V refers to the gradient of a function $V : \mathbb{R}^n \to \mathbb{R}$. The Lie derivative of function h with respect to f is defined as $\mathcal{L}_f h(x) = \frac{\partial h}{\partial x} f(x)$. Moreover, $\|u\|_R = u^T R u$.

II. PRELIMINARIES AND PROBLEM FORMULATION

This section presents preliminary results on stability, safety and optimality of control systems. The problem of optimal safe control design is then formulated.

A. Optimal Control of Dynamical Systems

Consider the nonlinear system

$$\dot{x} = f(x) + g(x)u \tag{1}$$

where $x \in \mathbb{R}^n$ is the vector of system states, $u \in \mathbb{R}^m$ is the vector of control inputs, and $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are both locally Lipschitz continuous with $f(0) = 0$. We assume that the system is stabilizable.

The common objective in standard optimal control design is to find a control policy that minimizes a predefined performance index over the trajectories of the system (1) defined as

$$J(x_0, u) = \int_0^\infty r(x(t), u(t))dt \tag{2}$$

where $r(x, u) = q(x) + u^T R u$ is the reward function, with $q(x)$ as a positive-definite function, and R as a symmetric-positive-definite matrix. The reward function $r(x, u)$ is defined such that optimizing (2) guarantees achieving the designer's intention (e.g. minimize the control effort or achieve a desirable transient response) as well as system's stability. Existence of a stabilizing optimal solution is guaranteed under mild assumptions on the system dynamics and the reward function (see [19]).

Assumption 1. Consider the system (1). There exists a Lyapunov function $V \in \mathcal{P}$ and a feedback control policy u, such that

$$\mathcal{L}(V, u) = -(\mathcal{L}_f V(x) + \mathcal{L}_g V(x)u) - r(x, u) \geqslant 0 \quad \forall x \in \mathbb{R}^n$$

This assumption guarantees the stability of the system and is satisfied for stabilizing systems.

Theorem 1. *[9, Theorem 10.1.2] Consider the system* (1) *with the performance function* (2)*. Suppose that there exists a positive semi-definite function $V^*(x) \in C^1$ that satisfies the Hamilton-Jacobi-Bellman (HJB) equation given as*

$$H(V^*) = 0$$

where

$$H(V) = q(x) + \mathcal{L}_f V(x) - \frac{1}{4}\mathcal{L}_g V(x) R^{-1} (\mathcal{L}_g V(x))^T. \tag{3}$$

Then, the feedback control

$$u^*(x) = \frac{1}{2} R^{-1} (\mathcal{L}_g V^*(x))^T(x) \tag{4}$$

optimizes the performance index (2) *and achieves asymptotic stability of the equilibrium $x = 0$. Moreover, the optimal value function is given by*

$$V^*(x_0) = \min_u J(x_0, u) = J(x_0, u^*), \quad \forall x_0 \in \mathbb{R}^n$$

Theorem 1 shows that to find an optimal control solution, one needs to solve the HJB equation (3). Policy iteration algorithms are presented to iteratively solve this HJB equation [22]. However, there is no safety guarantees for existing policy iteration algorithms.

B. Control Barrier Functions for Safe Control of Dynamical Systems

In a safety critical system, it is of vital importance for the system to prevent its state starting from any initial condition $\mathcal{X}_0$ from entering some certain unsafe regions $\mathcal{X}_u \in \mathcal{X}$. To design a safe controller, control barrier functions (CBFs), inspired by control Lyapunov function, can be used. Consider the nonlinear system (1) and let there exist a function $h : R^n \to R$ such that

$$\begin{aligned} h(x) &\geqslant 0, \ \forall x \in \mathcal{X}_0 \\ h(x) &< 0, \forall x \in \mathcal{X}_u \end{aligned} \tag{5}$$

Define

$$\ell = \{x \in \mathcal{X} \mid h(x) \geqslant 0\} \tag{6}$$

where $\mathcal{X}$ is the set of allowable states. Then, h is called a Zeroing CBF (ZCBF) if there exists a extended class function $K \in \mathcal{K}$ that satisfies

$$\sup_{u\in U} \{\mathcal{L}_f h(x) + \mathcal{L}_g h(x)u + K(h(x))\} \geqslant 0, \forall x \in \mathcal{X} \tag{7}$$

Having a ZCBF $h(x)$, the admissible control space $S(x)$ is defined as

$$S(x) = \{u \in U \mid \mathcal{L}_f h(x)$$

$$+ \mathcal{L}_g h(x)u + K(h(x)) \geqslant 0\}, \forall x \in \mathcal{X} \quad (8)$$

The following theorem shows how to design a controller using the ZCBF concept to guarantee that the safe set l is forward invariant and thus the system is safe.

Assumption 2. The admissible control space $S(x)$ is nonempty.

Remark 1. Assumption 1 is satisfied if there is a stabilizing controller and Assumption 2 is satisfied if there is a safe controller. Assumption 1 is standard for any control system design and Assumption 2 is standard for safe control design (see for example [22]). These two Assumption are not necessarily in conflict. But if safety and stability cannot be satisfied simultaneously at some region of the state space, it is standard (see [22] for example) to sacrifice on stability momentarily to guarantee safety.

Theorem 2. *[26] Given a set $\ell \subset R^n$ defined in* (6) *and a ZCBF h defined in* (7)*, any controller $u \in S(x)$ for the system* (1) *renders the safe set ℓ forward invariant.*

Conditions (5) and (7) guarantee that if the system starts from any initial condition $\forall x \in \mathcal{X}_0$ within the safe set, its future trajectories will not enter the unsafe region $\mathcal{X}_u$. This is because the condition (7) makes the safe set l invariant.

C. Safe Optimal Control: A Novel Framework

While an optimal controller designed based on the solution to the HJB (3) guarantees performance, it cannot assure safety. On the other hand, the control designed based on the CBF satisfying (8) guarantees safety but might result in a poor performance. Safety is the bare minimum requirement in the next-generation safety-critical systems. To bring the best of the both worlds together, in this section, we aim to design stabilizing safe controllers that provide a guaranteed performance within the volume of the certified safe area. It is well known that solving the HJB equation (3) without even considering constraints is hard and requires approximating the optimal solution. Iterative policy iteration algorithms are designed to solve the HJB equations without constraints [27]–[29]. Considering safety constrains, however, makes existing policy iteration algorithms invalid, as they cannot guarantee safety. In this paper, to find a safe control policy with guaranteed performance, we first relax solving the optimal control problem by solving an HJB inequality instead of the HJB equation (3). This allows us to obtain a suboptimal solution to the minimization problem (2) subject to (1). This formulation also allows us to include safety constraints satisfaction by including a CBF as another inequality to be satisfied. The relaxed optimal control problem for the system (1) with the performance (2) is first reviewed as follows.

$$\begin{aligned} &\min_V \int_\Omega V(x)dx \quad (9)\\ &s.t. \qquad H(V) \leqslant 0\\ &\qquad\quad V \in \mathcal{P} \end{aligned}$$

where $H(V)$ is defined as (3), and $\Omega \subset \mathbb{R}^n$ is an arbitrary compact set containing the origin. Problem 1 actually solves a relaxed version of the HJB (3) in which the the HJB equality is relaxed with HJB inequality. It is shown in [27] that the solution to Problem 1 is unique and if V^* is a solution to (9), then,

$$u^p = -\frac{1}{2}R^{-1}(\mathcal{L}_g V(x)^*)^T \quad (10)$$

guarantees stability and V^* can be viewed as an upper bound or an overestimate of the actual cost. The superscript p is used here to denote that u^p is a performance-oriented controller. This control policy, however, does not certify safety of the system. Using this relax optimal control formulation, we now propose the following optimization framework in which both performance and safety are considered. The safety is guaranteed by adding a CBF inequality to the relaxed optimal control problem formulation. The proposed safe optimization (7) is given by:

Problem 1 (Safe Optimal Control): find a controller that solves

$$\begin{aligned} &min \int_\Omega Vdx + K_\delta \delta^2\\ &s.t. \qquad H(V) \leqslant \delta\\ &\qquad \mathcal{L}_f h(x) + \mathcal{L}_g h(x)u + K(h(x)) \geqslant 0 \end{aligned}$$

where Ω is the area in which system performance is expected to be improved, $K_\delta > 0$ is a design parameter that trades off between the system's aggressiveness toward performance and safety, and δ is a stability relaxation factor.

Note that δ can be interpreted by the aspiration level for the performance that shows how much we sacrifice the performance when both safety and performance cannot be satisfied together. This parameter, however, is minimized to get as much performance as possible.

Theorem 3. *Under Assumptions 1 and 2, the Safe Optimization Problem 1 has a feasible solution.*

Proof: Based on Assumption 2, a safe control policy u exists. Let's write this control policy as $u = u^p + u^{safe}$ where $u^p = -\frac{1}{2}R^{-1}(\mathcal{L}_g V(x)^*)^T$ is part of the controller that is used to optimize the performance without concerning safety, and is given by (10), and u^{safe} is added to u^p to guarantee safety. Then,

$$\begin{aligned} H(V^*) &= q(x) + \mathcal{L}_f V^*(x) - \frac{1}{4}\mathcal{L}_g V^*(x)R^{-1}(\mathcal{L}_g V^*(x))^T\\ &= \mathcal{L}_f V^* + \mathcal{L}_g V^* u^p + r(x, u^p)\\ &= \mathcal{L}_f V^* + \mathcal{L}_g V^* u + r(x,u) - \mathcal{L}_g V^* u^{safe}\\ &\quad + u^p R u^p - u^T R u\\ &= \mathcal{L}_f V^* + \mathcal{L}_g V^* u + r(x,u) - \left\|u^{safe}\right\|_R^2 \end{aligned}$$

while u^p is stabilizing. If adding the safe controller u^{safe} violates stability at some points, u might not be stable for the entire region of attraction, i.e., $\mathcal{L}_f V^* + \mathcal{L}_g V^* u + r(x,u) < 0$ might not be satisfied at some points in the region of attraction. By choosing an appropriate slack variable $\delta(x)$,

to compensate for the conflict between safety and stability, however, one has

$$H(V^*) - \delta = \mathcal{L}_f V^* + \mathcal{L}_g V^* u + r(x,u) - \left\|u^{safe}\right\|_R^2 - \delta \leqslant 0$$

for some δ, which is generally a function of x. On the other hand, since u is safe, based on the converse control barrier Lyapunov (CBL) theorem, there exits a barrier certificate $h(x)$ satisfying

$$\mathcal{L}_f h(x) + \mathcal{L}_g h(x) u + K(h(x)) \geqslant 0$$

This completes the proof.

Solving this optimization problem is non-trivial in general. If both HJB inequality and CBL inequality constraints are restricted to SOS constraints, SOS programs can be used to significantly reduce the computational burden in finding a solution to this optimization problem. However, since $H(V)$ is bilinear in V, it makes the optimization problem hard or even impossible to solve using SOS. Therefore, we propose a safe policy iteration algorithm that iterates on a Bellman inequality, which is linear in V, instead of directly solving for $H(V) \leqslant \delta$. Using this Bellman inequality, a policy evaluation step that will find the value function V^i corresponding to a safe control policy u^i and a policy improvement step will find an improved policy u^{i+1} for which its safety is certified by adding the CBF inequality. We assume that an initial safe control policy u^0 is given, which can be found by a control policy that only satisfies the stability without any concern about optimality. To evaluate a given policy u^i .i.e, to find the value function V^i corresponding to it, the following policy evaluation step is proposed.

Safe policy evaluation step: Given a safe control policy u^i , find V^i and δ_i that solve the following optimization problem

$$\begin{aligned}
&\min_{V^i,\delta^i} \int_\Omega V^i dx + K_\delta \delta_i^2 \\
&\mathcal{L}(V^i,u^i) = -\mathcal{L}_f V^i - \mathcal{L}_g V^i u^i - r(x,u^i) \geqslant -\delta_i, \forall x \in \mathbb{R}^n \\
&V^{i-1} - V^i \geqslant 0
\end{aligned} \tag{11}$$

In terms of SOS, this optimization problem is transformed into

$$\begin{aligned}
&\min_{V^i,h^i} \int_\Omega V^i dx + K_\delta \delta_i^2 \\
&\mathcal{L}(V^i,u^i) + \delta_i \;\; is \;\; SOS \qquad \forall x \in \mathbb{R}^n \\
&V^{i-1} - V^i \;\; is \;\; SOS
\end{aligned} \tag{12}$$

In the policy evaluation step (12), the value function corresponding to a given policy is found while minimizing the relaxation factor δ^i. Note that since a safe control policy u^i might not necessarily be stabilizing, therefore $\mathcal{L}(V^i,u^i)$ might not be positive semidefinite. Once the value function V^i is found, the following policy improvement step finds an improved certified control policy.

To find an improved policy, we use the stationarity condition bellow [9]

$$u^{i+1} = \underset{u}{\operatorname{argmin}} \mathcal{L}(V^i,u) \tag{13}$$

which hints at a sufficient condition for global minimality of the Bellman equation and take into account satisfaction of the CBF when improving the policy. Note that the Bellman equation can be written as

$$\begin{aligned}
\mathcal{L}(V^i,u^{i+1}) &= -\mathcal{L}_f V^i - \mathcal{L}_g V^i u^{i+1} - r(x,u^{i+1}) \\
&= -(\mathcal{L}_f V^i + \mathcal{L}_g V^i (u^p)^{i+1})) \\
&\quad - r(x,(u^p)^{i+1}) + \left\|u^{safe}\right\|_R^2
\end{aligned} \tag{14}$$

where $u = u^p + u^{safe}$. Minimizing the term $-(\mathcal{L}_f V^i + \mathcal{L}_g V^i (u^p)^{i+1}) - r(x,(u^p)^{i+1})$ using stationarity condition results in $(u^p)^{i+1} = -\frac{1}{2}R^{-1}(\mathcal{L}_g V(x)^i)^T$. Therefore, minimizing $\left\|u^{safe}\right\|_R$ as the second term while setting $u^{i+1} = u^{safe} - \frac{1}{2}R^{-1}(\mathcal{L}_g V(x)^i)^T$ optimizes the performance. Since the controller must certify safety constraint, the CBL inequality must also be considered. This leads to the following policy improvement step.

Safe policy improvement step: Given a value function V^i find a certified improved control policy u^{i+1} by solving the following optimization problem

$$\begin{aligned}
&\min_{h^{i+1},u^{safe},Z} \left\|(u^{safe})\right\|^2 \\
&u^{i+1} = u^{safe} - \frac{1}{2}R^{-1}(\mathcal{L}_g V^i)^T, \quad \forall x \in \mathbb{R}^n \\
&\mathcal{L}_f h^{i+1} + \mathcal{L}_g h^{i+1} u^{i+1} + Z h^{i+1}(x) \;\; is \;\; SOS \\
&Z \;\; is \;\; SOS
\end{aligned} \tag{15}$$

Note that the safe policy improvement step (15) finds a safe controller that has minimum intervention with the performance-driven controller. The SOS program (15) involves bilinear decision variables. Therefore, to perform the policy improvement step, one needs to split it into several smaller SOS programs, which leads to an iterative search algorithm that first fixes h^{i+1} and finds u^{safe} and Z, and then fixes u^{safe} and Z and finds h^{i+1}.

Algorithm 1 presents the proposed policy iteration algorithm. Theorem 4 shows that Algorithm 1 finds a safe improved policy at every stops and stops if no further improvement can be achieved.

Algorithm 1: Safe policy Iteration

1: **procedure**
2: **Initialization:** Start with a safe and possibly conservative control policy
3: **Step 1 (Safe Policy Evaluation):** Fix the control policy u^i and solve (11) for V^i
4: **Step 2 (Safe Policy Improvement):** Repeat the following sub-steps until a stopping condition is met
5: Step 2.1. Fix V^i and h^{i+1} and solve (15) for u^{safe} and Z
6: Step 2.2. Fix V^i, u^{safe} and Z and solve (15) for h^{i+1}.
7: Repeat Steps 1 and 2 until it converges.
8: **end procedure**

Theorem 4. *Consider the dynamical control system* (1) *and Let Assumptions 1 and 2 be satisfied. Let Algorithm 1*

start from a safe control policy u^0 with a value function V^o. Then, the performance improves in each iteration and u^i is safe for all i.

Proof: The safe policy evaluation step in Algorithm 1 is conditioned on $V^{i-1} - V^i \geqslant 0$ and searches among policies that are stabilizing as much as possible and maximizing the performance. Moreover, based on (14) the policy improvement step in Algorithm 1 minimizes $-(\mathcal{L}_f V^i + \mathcal{L}_g V^i (u^p)^{i+1}) - r(x,(u^p)^{i+1})$ using the stationarity condition which results in $u^p = -\frac{1}{2}R^{-1}(\mathcal{L}_g V^i)^T$. Therefore, minimizing $\left\|u^{safe}\right\|$ as the second term while setting $u^{i+1} = u^{safe} - \frac{1}{2}R^{-1}(\mathcal{L}_g V^i)^T$, as performed in policy improvement step of Algorithm 1, optimizes the performance in each iteration. Since the CBL inequality is also considered while finding an improved policy, the safety of the improved policy is guaranteed while the intervention of the safe control part with the performance is minimized by minimizing u^{safe}. This completes the proof.

D. Simulation results

To verify the effectiveness of the proposed approach, a simulation example is considered in this section.

Example 1. *consider an autonomous dynamical system as follows*

$$\begin{bmatrix}\dot{x_1}\\ \dot{x_2}\end{bmatrix} = \begin{bmatrix} x_2 + 0.6x_2^2 + 0.4u_2 \\ -x_1 + x_2 + x_1^2 x_2 + 0.6u_1 + 0.4u_2\end{bmatrix}$$

where $x = [x_1, x_2]$ and u are the state and control of the system. The unsafe area is encoded with polynomial inequalities $X_u = \{x \in R^2 | t_i(x) < 0, i = 1,2,3\}$ where

$$t_1 = -0.5 + (x_1+2)^2 + (x_2+2)^2 < 0$$
$$t_2 = -0.5 + (x_1+3)^2 + (x_2-1.5)^2 < 0$$
$$t_3 = -0.5 + (x_1-3)^2 + (x_2-1)^2 < 0$$

To find an initial policy to start with, using SOS-related techniques presented in [30], the following robust stabilizing control policy is used.

$$u^1 = \begin{bmatrix}u_1^1\\ u_2^1\end{bmatrix} = \begin{bmatrix}8.316x_1 + 11.295x_2 \\ -8.2x_1 - 1.331x_2\end{bmatrix}$$

Our control objective is to find improved safe control policies using the proposed safe policy iteration algorithm. By solving the following feasibility problem using SOS-TOOLS

$$\begin{aligned} & V \in \mathbb{R} \\ & \mathcal{L}(V^1, u^1) \quad is \quad SOS \end{aligned} \tag{16}$$

we obtain the value function corresponding to (16) *as*

$$V^1 = 1.343x_1^2 + 0.5155x_1x_2 + 1.1152x_2^2$$

The control policy found by the proposed algorithm is

$$\begin{aligned} u_1^{1*}(x) = & -0.00427x_1^3 - 0.000728x_1^2x_2 + 0.071x_1^2 \\ & - 0.121x_1x_2^2 + 0.0428x_1x_2 - 0.166x_1 \\ & - 0.0362x_2^3 - 0.0535x_2^2 - 1.931x_2 \\ u_2^{1*}(x) = & -0.00549x_1^3 - 0.00891x_1^2x_2 - 0.086x_1^2 \\ & + 0.0049x_1x_2^2 - 0.1079x_1x_2 - 1.556x_1 \\ & - 0.0387x_2^3 - 0.0496x_2^2 - 2.777x_2 \end{aligned}$$

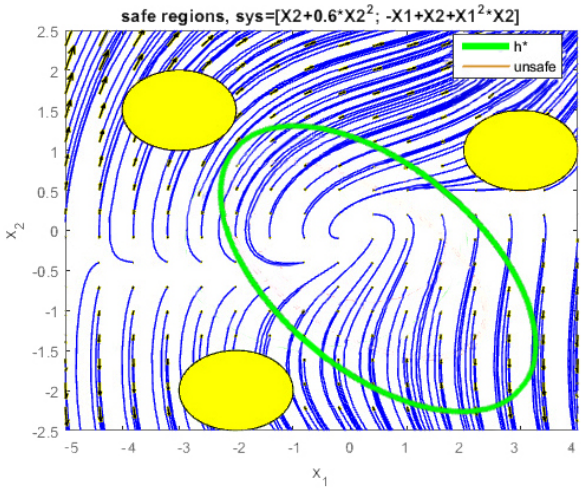

Fig. 1. The estimated safe region using the proposed optimal safe controller

Moreover, the safe set is

$$\ell = \{x \in \mathbb{R}^2 | h(x) \geqslant 0\}$$

where

$$\begin{aligned} h(x) = & 0.588 - 0.066x_1^2 - 0.0476x_1 - 0.121x_2 \\ & - 0.0274x_1x_2 - 0.025x_2^2 \end{aligned}$$

Note that the safe set if not know a priori and is a subset of the complementary set of the unsafe area. That is, the safe set must not only be in the complementary set of the unsafe set be should also be invariant in the sense that it never leaves the set in the future. The safe set is found using the CBF $h(x)$. Note that the barrier certificate is restricted to be second order polynomial. The estimated safe sets for the initial control policy and the optimal control policy are illustrated in Figure 1.

The state trajectories of the system for both proposed safe optimal control policy, and safe control using CBF are shown in Figure 2 and Figures 3. One can observes that the performance of the proposed controller is significantly better than the safe controller designed based on only CBF.

III. Conclusion

A safe optimization is proposed for control of dynamics systems under state constraints. To guarantee performance and safety, a Hamilton-Jacobi-Bellman (HJB) inequality replaces the HJB equality and a safe policy iteration algorithm is presented that certifies the safety of the improved policy and finds a value function corresponding to it. SOS is then used to implement the proposed safe policy iteration algorithm. Simulation examples verify the effectiveness of the proposed safe algorithm.

References

[1] 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.

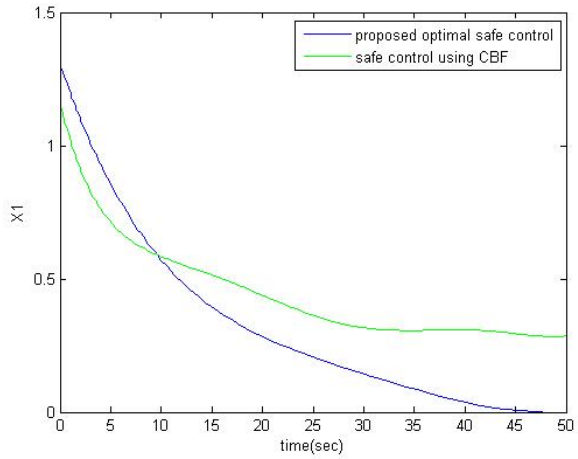

Fig. 2. System state trajectory x_1

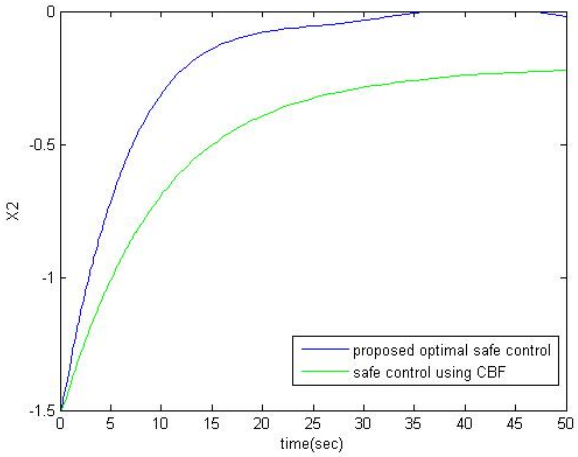

Fig. 3. System state trajectory x_2

[2] 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.

[3] F. L. Lewis and K. G. Vamvoudakis, "Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, pp. 14–25, Feb 2011.

[4] B. Kiumarsi and F. L. Lewis, "Actor–critic-based optimal tracking for partially unknown nonlinear discrete-time systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, pp. 140–151, Jan 2015.

[5] 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.

[6] D. Wang, D. Liu, Y. Zhang, and H. Li, "Neural network robust tracking control with adaptive critic framework for uncertain nonlinear systems," *Neural Networks*, vol. 97, pp. 11 – 18, 2018.

[7] 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, no. 1, pp. 82 – 92, 2013.

[8] W. Gao and Z. Jiang, "Learning-based adaptive optimal tracking control of strict-feedback nonlinear systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, pp. 2614–2624, June 2018.

[9] F. L. Lewis, D. L. Vrabie, and V. L. Syrmos, *Optimal Control*. John Wiley, 2012.

[10] W. Gao, Z. Jiang, and K. Ozbay, "Data-driven adaptive optimal control of connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, pp. 1122–1133, May 2017.

[11] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit solution of model predictive control via multiparametric quadratic programming," in *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*, vol. 2, pp. 872–876, June 2000.

[12] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal state feedback control of constrained nonlinear systems using a neural networks hjb approach," *Annual Reviews in Control*, vol. 28, no. 2, pp. 239 – 251, 2004.

[13] E. Garone, S. Di Cairano, and I. Kolmanovsky, "Reference and command governors for systems with constraints: A survey on theory and applications," *Automatica*, vol. 75, pp. 306–328, 2017.

[14] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *53rd IEEE Conference on Decision and Control*, pp. 6271–6278, Dec 2014.

[15] 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, pp. 3861–3876, Aug 2017.

[16] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *2016 American Control Conference (ACC)*, pp. 322–328, July 2016.

[17] M. Z. Romdlony and B. Jayawardhana, "Stabilization with guaranteed safety using control lyapunov–barrier function," *Automatica*, vol. 66, pp. 39 – 47, 2016.

[18] M. Z. Romdlony and B. Jayawardhana, "Uniting control lyapunov and control barrier functions," in *53rd IEEE Conference on Decision and Control*, pp. 2293–2298, Dec 2014.

[19] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," *International Workshop on Hybrid Systems: Computation and Control*, p. 477–492, 2004.

[20] 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.

[21] L. Wang, A. D. Ames, and M. Egerstedt, "Multi-objective compositions for collision-free connectivity maintenance in teams of mobile robots," *CoRR*, vol. abs/1608.06887, 2016.

[22] L. Wang, D. Han, and M. Egerstedt, "Permissive barrier certificates for safe stabilization using sum-of-squares," in *2018 Annual American Control Conference (ACC)*, pp. 585–590, IEEE, 2018.

[23] W. Xiao, C. Belta, and C. G. Cassandras, "Decentralized merging control in traffic networks: A control barrier function approach," in *Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems*, ICCPS '19, (New York, NY, USA), p. 270–279, Association for Computing Machinery, 2019.

[24] R. Harvey, Z. Qu, and T. Namerikawa, "An optimized input/output-constrained control design with application to microgrid operation," *IEEE Control Systems Letters*, vol. 4, pp. 367–372, April 2020.

[25] J. Hauser and A. Saccon, "A barrier function method for the optimization of trajectory functionals with constraints," in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 864–869, Dec 2006.

[26] L. Wang, A. Ames, and M. Egerstedt, "Safety barrier certificates for heterogeneous multi-robot systems," in *2016 American Control Conference (ACC)*, pp. 5213–5218, IEEE, 2016.

[27] Y. Jiang and Z. Jiang, "Global adaptive dynamic programming for continuous-time nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 60, pp. 2917–2929, Nov 2015.

[28] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, "Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 10, pp. 1513–1525, 2013.

[29] 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.

[30] J. Xu, L. Xie, and Y. Wang, "Simultaneous stabilization and robust control of polynomial nonlinear systems using sos techniques," *IEEE Transactions on Automatic Control*, vol. 54, pp. 1892–1897, Aug 2009.