

Safety Verification and Controller Synthesis for Systems with Input Constraints

Han Wang*, Kostas Margellos*, Antonis Papachristodoulou*

* *Department of Engineering Science, University of Oxford, Oxford,
United Kingdom. E-mails: {han.wang, kostas.margellos,
antonis}@eng.ox.ac.uk*

Abstract: In this paper we consider the safety verification and safe controller synthesis for nonlinear control systems. The Control Barrier Certificates (CBC) approach is proposed as an extension to the barrier certificates approach. Our approach can be used to characterize control invariance of a given set in terms of safety of a general nonlinear control system subject to input constraints. From the point of view of controller design, the proposed method provides an approach to synthesize a safe control law that guarantees that the trajectories of the system starting from a given initial set do not enter an unsafe set. Unlike the related control barrier functions formulations, our formulation only considers the vector field within the tangent cone of the zero level set defined by the certificates, and is shown to be less conservative by means of numerical evidence. For polynomial systems with semi-algebraic initial and safe sets, CBCs and safe control laws can be synthesized using sum-of-squares decomposition and semi-definite programming. Numerical examples demonstrate the efficacy of our approach.

Copyright © 2023 The Authors. This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Keywords: safety verification, nonlinear system, sum-of-squares programming

1. INTRODUCTION

Safety-critical systems are commonly used in modern autonomous applications, such as unmanned aerial vehicles, autonomous driving and surgical robotics (Guiochet et al., 2017). Their safety-critical nature requires the behaviour of these systems to remain within a given safe set for an infinite time horizon. Such a property is directly related to reachability analysis and reach-avoid games (Lygeros, 2004), (Margellos and Lygeros, 2011), i.e. determining an initial set so that trajectories reach a target set without entering an unsafe region. However, verifying safety for general nonlinear systems using these methods is hampered computationally due to the need of solving the underlying Hamilton Jacobi Isaacs PDE.

To overcome this issue, the connection between forward invariance and safety has been established in (Prajna and Rantzer, 2005). Forward invariance is a system-set property which guarantees that the trajectories entering a set cannot escape it (Blanchini, 1999). By finding an invariant subset of a safe region, the system is ensured to be safe. To identify a candidate invariant set, the barrier certificates approach which considers the invariant set as the certificate's sub-zero level set, was proposed in (Prajna et al., 2007), (Prajna and Jadbabaie, 2004). Although the properties of this framework have been demonstrated for autonomous systems with and without stochasticity, there is no systematic formulation for the case where control inputs are present. To address this issue, the so called control barrier functions (CBF) approach was proposed in (Ames et al., 2016).

Control barrier functions are a class of functions that are negative in the unsafe regions, and can be used to verify

the safety property. Unlike Lyapunov-like barrier certificates, control barrier functions are less restrictive by introducing an additional relaxation term in the constraint. Forward invariance is proven by satisfying the constraints and utilizing the comparison lemma (Vidyasagar, 2002). The approach can be easily combined with the control Lyapunov functions approach (Freeman and Primbbs, 1996) under a unified quadratic programming framework that trades safety with controller performance (Ames et al., 2016). It was also shown to be applicable and promising in many applications such as adaptive cruise control (Xu et al., 2017), bipedal robots (Hsu et al., 2015), multi-robot collision avoidance (Chen et al., 2017) and others.

Several attempts have been encountered to improve the feasibility when input limits are taken into account, such as adaptive CBF (Xiao et al., 2021), (Zeng et al., 2021), higher relative degree CBF (Xiao and Belta, 2019), backup CBF (Chen et al., 2021), singular CBF (Tan et al., 2021). These methods aim at addressing the cases where the CBF based quadratic programming (QP) is infeasible. Often a CBF is assumed to be constructed directly from a physical property such as kinodynamics of the vehicle. How to synthesize a CBF efficiently is still an open question, and has attracted significant attention in recent years.

Direct numerical synthesis approaches by sum-of-squares programming (Wang et al., 2018), (Xu et al., 2017), machine learning (Srinivasan et al., 2020), and deep learning (Robey et al., 2020) have been proposed. All these methods, either via convex optimisation, or learning techniques, consider the synthesis of a CBF with a relaxation term included in the synthesis procedure. From the standpoint of control invariant sets, it is guaranteed that there exists a class- \mathcal{K} relaxation term to bound the safety variation, but

imposing such a term at every point inside the set during the control synthesis introduces conservativeness. The synthesis procedure without the presence of this term has been considered in (Clark, 2021), and using Positivstellensatz, a weaker condition on invariance is imposed for systems without input limits.

In this work we revisit the barrier certificates approach, and extend it for nonlinear control systems with actuation constraints. Our formulation provides a direct interpretation of control invariance and safety, thus alleviating conservativeness. The existence of a control barrier certificate (CBC) is sufficient to guarantee safety, hence the approach can be used for safety verification. For systems with polynomial dynamics and semi-algebraic safe and initial sets, we use sum-of-squares programming and the generalised S-procedure to synthesize a CBC, as well as a Lipschitz state feedback safe control law which satisfies the actuation constraints.

The remainder of this paper is organized as follows. The notion of control barrier certificates is introduced in Section 2. The computational methods based on sum-of-squares programming and the S-procedure are presented in Section 3. Several simulation results on synthesizing CBCs and safe controllers are shown in Section 4. Section 5 concludes the paper.

Notations: \mathbb{R} represents the space of real numbers, and \mathbb{R}^n denotes the n -dimensional real space. For a set S , $\text{Int}S$, ∂S , \bar{S} are the interior, boundary and complementary set, respectively. $A \succeq 0$ means matrix A is positive semi-definite. $\Sigma[x]$ and $\mathcal{R}[x]$ denote the set of sum-of-squares polynomials and polynomials in x with real coefficients.

2. CONTROL BARRIER CERTIFICATES

In this section, we consider the controller synthesis problem under the realm of safety for nonlinear systems. Here, we extend the results on barrier certificates to control barrier certificates for safety verification and safe controller design, where the safe controller is designed in the feedback form, simultaneously with the control barrier certificate. We also compare our results with the CBF approach.

We start the formulation for a continuous-time nonlinear system for generality. The system is described by an ordinary differential equation:

$$\dot{x} = f(x, u), \quad (1)$$

where $x(t) \in \mathbb{R}^n$ denotes the state vector, and for all time instances t , $u(t) \in \mathcal{U} \subseteq \mathbb{R}^m$ is the control input, where \mathcal{U} is a bounded set denoting actuation limits and $f(x, u)$ is a locally Lipschitz continuous vector field. We assume that the solution to (1) is unique.

We now define the notion of *Control Barrier Certificates* (CBC) which is an extension to barrier certificates (Prajna and Jadbabaie, 2004).

Definition 1 (Control Barrier Certificates). *Let a continuous time control system denoted by $\dot{x} = f(x, u)$, with initial set $I \subseteq \mathbb{R}^n$, safe set $S \subseteq \mathbb{R}^n$, and input constraints $\mathcal{U} \subseteq \mathbb{R}^m$. A C^1 function $B : \mathbb{R}^n \rightarrow \mathbb{R}$ is called a Control Barrier Certificate (CBC) if*

$$B(x) < 0, \quad \forall x \in \bar{S}, \quad (2a)$$

$$B(x) \geq 0, \quad \forall x \in I, \quad (2b)$$

$$\sup_{u \in \mathcal{U}} \frac{\partial B(x)}{\partial x} f(x, u) > 0, \quad \forall x, \text{ such that } B(x) = 0. \quad (2c)$$

Let

$$K_{CBC}(x) := \begin{cases} \left\{ u \mid \frac{\partial B(x)}{\partial x} f(x, u) > 0 \right\} \cap \mathcal{U}, & \text{if } B(x) = 0 \\ \mathcal{U}, & \text{otherwise} \end{cases} \quad (3)$$

denote the admissible set of control inputs for a CBC $B(x)$. Let $\mathcal{B} := \{x \mid B(x) \geq 0\}$ denote the zero-super level set of $B(x)$. Different from a control barrier function, a control barrier certificate $B(x)$ is only required to be nonnegative on the initial set I . Similarly to the proof for the existence of a barrier certificate is sufficient for safety (Prajna and Jadbabaie, 2004, Theorem 1), the existence of a CBC is also sufficient for safety.

In the case where the control barrier certificate is unknown, synthesizing the control barrier certificates together with the safe controller design becomes an issue. To this end, $B(x)$ and $u(x)$ are parameterized by

$$B(x) = \sum_{i=1}^k p_k \Lambda_k(x), u(x) = \sum_{i=1}^l b_i \zeta'_i(x), \quad (4)$$

where $p := \{p_1, \dots, p_k\}$ and $b := \{b_1, \dots, b_l\}$ denotes a series of parameters which will be decision variables in an optimisation problem, and $\Lambda_1(x), \dots, \Lambda_k(x)$, $\Lambda'_1(x), \Lambda'_l(x)$ are two classes of function basis. The new optimisation problem for constructing the CBC and controller is

$$\begin{aligned} &\text{find } b, p \\ &\text{s.t. (2), (4),} \\ &u(x) \in K_{CBC}(x). \end{aligned} \quad (5)$$

Compared to CBF-QP with known control barrier certificates, (5) is computationally harder since it involves solving an infinitely constrained optimisation problem. But it gives more flexibility for safe controller design, as $B(x)$ is also a variable now. We will show how to solve (5) efficiently using sum-of-squares programming in Section 3.

3. COMPUTATION METHOD BASED ON SUM-OF-SQUARES PROGRAMMING

In this section we show how to construct the control barrier certificates and design a safe controller for polynomial systems with semi-algebraic safe and initial sets. The nonlinear control affine system is represented by

$$\dot{x} = f(x) + g(x)u, \quad (6)$$

where $f(x)$ and $g(x)$ are locally smooth polynomial functions and $u \in \mathcal{U} := \{u \mid Au + b \geq 0\}$.

Even for such a simplified system model, solving the parametric optimisation problem (4) – (5) involves solving an infinite set of non-negative inequalities. However, for systems with polynomial functions $f(x)$, $g(x)$ and semi-algebraic sets I , S , a tractable method for tackling the infinite inequalities is *sum-of-squares* (SOS) programming, which is a convex relaxation method based on the sum-of-squares decomposition of multivariate polynomials and semidefinite programming.

Theorem 1. *Consider a polynomial nonlinear system (6), semi-algebraic safe set $S = \{x \mid s(x) \geq 0\}$, initial set $I = \{x \mid w(x) \geq 0\}$, and control admissible set $\mathcal{U} :=$*

$\{u|Au + b \geq 0\}$, where $A \in \mathbb{R}^{h \times m}$, and $b \in \mathbb{R}^h$. If there exists multipliers $\sigma_{\text{safe}} \in \Sigma[x]$, $\sigma_{\text{init}} \in \Sigma[x]$, $\lambda_1 \in \mathcal{R}[x]$, $\lambda_2 \in \mathcal{R}[x]^h$, polynomials $B(x) \in \mathcal{R}[x]$, $u(x) \in \mathcal{R}[x]$, predefined small positive real scalars $\epsilon_1 > 0$, $\epsilon_2 > 0$, such that

$$-B(x) + \sigma_{\text{safe}}s(x) - \epsilon_1 \in \Sigma[x], \quad (7a)$$

$$B(x) - \sigma_{\text{init}}w(x) \in \Sigma[x], \quad (7b)$$

$$\frac{\partial B(x)}{\partial x}(f(x) + g(x)u(x)) + \lambda_1 B(x) - \epsilon_2 \in \Sigma[x], \quad (7c)$$

$$-\lambda_2 B(x) + Au(x) + b \in \Sigma[x]^h, \quad (7d)$$

then $B(x)$ fulfills the conditions (2) and $\mathcal{B} = \{x|B(x) \geq 0\}$ is a control invariant set with respect to vector field $f(x) + g(x)u(x)$.

Proof. Condition (7a) indicates that for any x , $-B(x) + \sigma_{\text{safe}}s(x) - \epsilon_1 \geq 0$, thus for any x , $-B(x) + \sigma_{\text{safe}}s(x) > 0$. Therefore, for any $x \in \tilde{S}$, we directly have that $\sigma_{\text{safe}}s(x) \leq 0$, and further $B(x) < 0$, i.e., (2a) holds. Similarly (7b) can be shown to satisfy (2b) following the same arguments. Based on the S-procedure, condition (7c) implies condition (2c), because when $B(x) = 0$, $\frac{\partial B(x)}{\partial x}(f(x) + g(x)u(x)) - \epsilon_2 \geq 0$, and thus $\frac{\partial B(x)}{\partial x}(f(x) + g(x)u(x)) > 0$. Condition (7d) implies that $Au(x) + b$ is element-wise nonnegative for $x \in \partial\mathcal{B}$. The small positive real scalars ϵ_1, ϵ_2 ensure strict inequality for (2a) and (2c). \square

We note that in Theorem 1 we only require a polynomial multiplier λ , but not a SOS one since the condition $\frac{\partial B(x)}{\partial x}(f(x) + g(x)u(x)) \geq 0$ is only imposed on the boundary $B(x) = 0$. Condition (7c) introduces products of decision variables, i.e. $\lambda B(x)$, which results in bilinearity. However, there is no guaranteed solver for nonconvex, or specifically bilinear constrained SOS programs. Here, like existing work of using SOS to synthesize barrier certificates, we use an iterative procedure for control barrier certificate synthesis and safe control law design. Different from the iterative algorithm for barrier certificate synthesis, our problem involves an additional polynomial variable u in the SOS program. Thus, an additional round for controller synthesis is required in our algorithm.

1) Initialization: We first fix the degree of polynomials $B(x)$, σ_{safe} , σ_{init} , λ_1 , λ_2 and $u(x)$. The polynomial/monomial scalar/vector basis for $B(x)$ and $u(x)$ have degree upper bounded by the aforementioned degrees of $B(x)$. ϵ_1 and ϵ_2 are chosen to be small real numbers. Unlike the iterative procedure proposed in (Xu et al., 2017) which initializes the control law by a scaled LQR controller, we find the initialized feasible control input $u^0(x)$ by solving a feasibility SOS program.

$$\begin{aligned} &\text{find } k_1, \dots, k_l, \sigma_{\text{cont}} \\ &\text{s.t. } A(k_0 + \sum_{j=1}^l k_j v_j(x)) + b \cdot \sigma_{\text{cont}} \in \Sigma[x]^h. \end{aligned} \quad (8)$$

We note here that there is no control barrier certificate $B(x)$ at this stage of finding the initial feasible control input $u^0(x)$. Therefore, $u^0(x)$ can not be restricted to the domain of $\partial\mathcal{B}$ as that in (7d). Other than directly interpreting $A(k_0 + \sum_{j=1}^l k_j v_j) + b \in \Sigma[x]^h$, we add an additional positive multiplier σ_{cont} which satisfies $\sigma_{\text{cont}} -$

$\epsilon_3 \in \Sigma[x]$, $\epsilon_3 > 0$ to avoid introducing constant terms in the SOS constraints, as well as improving feasibility. The resulting initial controller $u^0(x)$ is derived by the parameters k_1, \dots, k_l and the scaled term σ_{cont} from the solution of (8)

$$u^0(x) = \frac{1}{\sigma_{\text{cont}}} \cdot (k_0 + \sum_{j=1}^l k_j v_j(x)). \quad (9)$$

Given initial input $u^0(x)$, the corresponding scaled multiplier σ_{cont} , the initial control barrier certificate $B^0(x)$ can be found by solving an initial feasibility SOS program as

$$\begin{aligned} &\text{find } p_0, \dots, p_m, \sigma_{\text{safe}}, \sigma_{\text{init}} \\ &\text{s.t. } -B(x) + \sigma_{\text{safe}}s(x) - \epsilon_1 \in \Sigma[x], \\ &\quad B(x) - \sigma_{\text{init}}w(x) \in \Sigma[x], \\ &\quad \sigma_{\text{cont}} \cdot \frac{\partial B(x)}{\partial x}(f(x) + g(x)u^0(x)) - \epsilon_2 \in \Sigma[x], \end{aligned} \quad (10)$$

The boundary condition (7c) is strengthened to be $\frac{\partial B(x)}{\partial x}(f(x) + g(x)u(x)) - \epsilon_2 \in \Sigma[x]$ for convexity and simplicity of computing. Compared with the condition (2c), every super level set of this constructed $B(x)$ is invariant. This condition is also referred to be the weak barrier certificate in (Prajna and Jadbabaie, 2004). $\sigma_{\text{cont}} \cdot \frac{\partial B(x)}{\partial x}(f(x) + g(x)u^0(x)) - \epsilon_2$ is guaranteed to be a polynomial, since $\sigma_{\text{cont}} \cdot u^0(x)$ is a polynomial.

After obtaining a feasible initial control input $u^0(x)$ and control barrier certificate $B^0(x)$, the problem of control barrier certificates synthesis can be regarded as a barrier certificates synthesis problem with vector field $f(x) + g(x)u^0(x)$. The multipliers λ_1^0, λ_2^0 are fixed to be 0 or 1 in initialization for simplicity. The initial control barrier certificate $B^0(x)$ is used to enlarge the size of the control invariant set incrementally. The following steps of the algorithm iteratively solve the SOS program to address the bisecting terms $\lambda_1 B(x)$ and $\frac{\partial B(x)}{\partial x}(f(x) + g(x)u(x))$ in (7c).

2) Update the control input $u^k(x)$: At iteration k , given a control barrier certificate from (10) (when $k = 1$) or (12) (when $k \geq 2$), the controller synthesis is constrained to (7d). Fixing $B(x) = B^{k-1}(x)$, a convex programming synthesis procedure for $u^k(x)$ is

$$\begin{aligned} &\text{find } k_0, \dots, k_l, \lambda_1, \lambda_2 \\ &\text{s.t. } -\lambda_2 B^{k-1}(x) + A(k_0 + \sum_{j=1}^l k_j v_j) + b \in \Sigma[x]^h, \\ &\quad \frac{\partial B^{k-1}(x)}{\partial x}(f(x) + g(x)u(x)) + \lambda_1 B^{k-1}(x) - \epsilon_2 \in \Sigma[x], \end{aligned} \quad (11)$$

and we have that $u^k(x) = (k_0 + \sum_{j=1}^l k_j v_j)$. Here we use λ_1 other than λ_1^{k-1} since $B(x)$ has been substituted by $B^k(x)$, thus there is no bilinear term anymore. By limiting the domain of the controller to $\partial\mathcal{B}$, there is no need to have additional multiplier σ_{cont} as that has been used in initial controller design for feasibility.

3) Synthesize the control barrier certificate $B^k(x)$: After obtaining a feasible control input $u^{k-1}(x)$, the synthesis of a control barrier certificate $B^k(x)$ relies on fixed multipliers $\lambda_1^{k-1}, \lambda_2^{k-1}$ to bypass the bilinear terms.

Searching for $B^k(x)$ and the remaining multipliers follows the following SOS program

$$\begin{aligned}
 & \text{find } p_0, \dots, p_m, \sigma_{\text{safe}}, \sigma_{\text{init}}, \sigma_{\text{enl}} \\
 & \text{s.t. } -B(x) + \sigma_{\text{safe}}s(x) - \epsilon_1 \in \Sigma[x], \\
 & \quad B(x) - \sigma_{\text{init}}w(x) \in \Sigma[x], \\
 & \quad \frac{\partial B(x)}{\partial x}(f(x) + g(x)u^k(x)) + \lambda_1^{k-1}B(x) - \epsilon_2 \in \Sigma[x], \\
 & \quad -\lambda_2^{k-1}B(x) + Au^k(x) + b \in \Sigma[x]^h, \\
 & \quad B(x) - \sigma_{\text{enl}}B^{k-1}(x) \in \Sigma[x],
 \end{aligned} \tag{12}$$

where $\sigma_{\text{enl}} \in \Sigma[x]$. Here the control law $u^{k-1}(x)$ is substituted for the variable u , and the multipliers λ_1 λ_2 are substituted by λ_1^{k-1} and λ_2^{k-1} , respectively. We introduce additional constraints $B(x) - \sigma_{\text{enl}}B^{k-1}(x) \in \Sigma[x]$ to enlarge the volume of the control invariant set \mathcal{B}^k by enforcing $\mathcal{B}^{k-1} \subseteq \mathcal{B}^k$. A similar technique is also used in (Cunis and Kolmanovsky, 2021).

4) Update the multipliers: The multiplier λ_1^k updates rely on a fixed control barrier certificate $B^k(x)$ and input $u^k(x)$. Clearly, there is no bilinearity in the control input update procedure (11). The multipliers λ_1^k and λ_2^k are obtained by directly solving it, while there is no need to fix $B(x)$ and re-solve the SOS programming problem.

Remark. For the case where (8) or (10) is infeasible, there are two ways to ensure feasibility: (i) Increase the degree of the polynomial bases $v_1, \dots, v_l, b_1, \dots, b_m$; (ii) Re-solve the problem (8) with an alternative objective function for a different initialization. The algorithm terminates upon convergence in two consecutive iterations, i.e. $B^k(x) = B^{k-1}(x)$.

4. SIMULATION RESULTS AND DISCUSSION

In this section we show numerical simulation results on synthesizing control barrier certificates and safe controllers under different system settings. The SOS toolbox SOS-TOOLS (Prajna et al., 2002), (Papachristodoulou et al., 2013) is used with version v401 for parsing the SOS programs, while SeDuMi is used for solving the resulting semidefinite program (Sturm, 1999). We also compare the CBC proposed in this paper and CBF mainly from the view point of synthesis.

4.1 Nonlinear Control Affine Systems

We first consider a general second order polynomial nonlinear control affine system. This system is defined by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_1 + \frac{1}{3}x_1^3 + x_2 \end{bmatrix} + \begin{bmatrix} x_1^2 + x_2 + 1 \\ x_2^2 + x_1 + 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \tag{13}$$

where the control input is box constrained, i.e. $u_1 \in [-1.5, 1.5]$, $u_2 \in [-1.5, 1.5]$. The safe set is defined by a disc $S = \{x|x_1^2 + x_2^2 - 3 \leq 0\}$, and initial set defined by $I = \{x|(x_1 - 0.4)^2 + (x_2 - 0.4)^2 - 0.16 \leq 0\}$. We leverage the control barrier certificates synthesis procedures (7) to find a polynomial CBC $B_1(x)$, and compare the results with the CBF synthesis procedure proposed in (Xu et al., 2017). To synthesize a candidate CBF $B_2(x)$, an alternative constraint for (7c) is introduced

$$\frac{\partial B(x)}{\partial x}(f(x) + g(x)u(x)) - \sigma_{\text{cbf}}B(x) + \alpha B(x) - \epsilon_2 \in \Sigma[x], \tag{14}$$

where the class- \mathcal{K} function is selected to be $\alpha B(x)$ with $\alpha > 0$, and $\sigma_{\text{cbf}} \in \Sigma[x]$ is a SOS multiplier. Here we restrict the definition domain for CBF to be \mathcal{B}_2 . ϵ_2 is set to be the same with that in (7c). Instead of the feasibility SOS program used for CBC, we set an objective function α which is maximized for CBF as in (Xu et al., 2017).

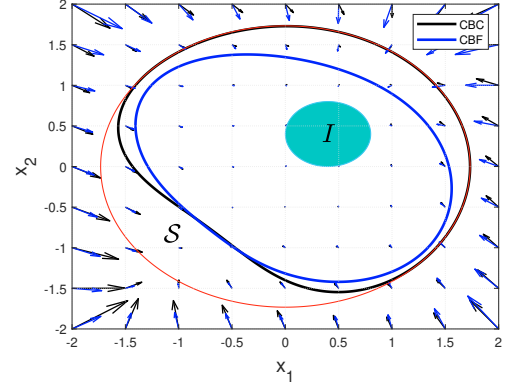


Fig. 1. Control invariant sets defined by CBC or CBF

Figure 1 shows the control invariant sets defined by CBF and CBC. The red and light blue disc represent the safe and initial sets, respectively. The interior of the deep blue curve is the invariant set \mathcal{B}_2 defined by CBF, and the interior of the black curve is the invariant set \mathcal{B}_1 defined by CBC. It can be seen from the figure that \mathcal{B}_1 is “larger” than \mathcal{B}_2 . Actually we have $\mathcal{B}_2 \subset \mathcal{B}_1$, which is proved by there exists a SOS multiplier σ , such that $B_1(x) - \sigma B_2(x) \in \Sigma[x]$. The reason is that, we trivially have $\sigma_{\text{cbf}} + \alpha \in \mathcal{R}[x]$. A larger search area enables us to find a larger control invariant set. On the other hand, the additional term $\lambda_1 B_1(x)$ can be regarded as an adapted relaxation term compared to a fixed class- \mathcal{K} function used in CBF approach. By using a zeroth order base for the polynomial multiplier λ_1 and expanding the definition domain of CBF to the whole real space, our formulation is equivalent to CBF. Higher order basis selections hereby reduce conservativeness.

Figure 2 shows the value of the relaxation coefficient λ_1 and α . The multiplier λ_1 includes the following monomial basis: $[x_1^2, x_1x_2, x_2^2, x_1, x_2, 1]$. It can be seen that λ_1 varies in the control invariant set, which therefore endows the formulation flexibility. An interesting property here is that α cannot be too large, this is because for $x \in \mathcal{B}_1$, $\alpha B_2(x) < 0$. In addition, with a non-empty safe set $S \subset \mathbb{R}^n$, we directly have $B_1(x) \notin \Sigma[x]$, and $\alpha B_1(x) \notin \Sigma[x]$.

The control invariant set \mathcal{B}_1 obtained by the CBC design and the values of the safe controllers are shown in Figure 3. The vector field, which is represented by the arrows in Figure 3(a) point inside \mathcal{B}_1 on $\partial\mathcal{B}_1$. The value of the polynomial control law $u(x)$ is within $[-1.5, 1.5]$ in both coordinates.

4.2 LTI Systems

Consider a second order linear model

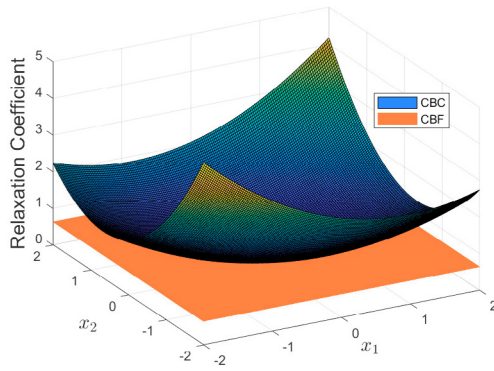


Fig. 2. Relaxation coefficients $\lambda(x)$ for CBC, and α for CBF

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad (15)$$

where $u_1 \in [-2.5, 2.5]$, $u_2 \in [-2.5, 2.5]$. The system is unstable since the eigenvalues of the state matrix $\begin{bmatrix} 2 & 1 \\ 3 & 1 \end{bmatrix}$ are 3.3 and -0.3 , while it is locally stabilizable. The safe set is defined by a disc $S = \{x | x_1^2 + x_2^2 - 3 \leq 0\}$. The trajectories of the system start from the following initial set $I = \{x | (x_1 - 0.4)^2 + (x_2 - 0.4)^2 - 0.16 \leq 0\}$. Clearly, all trajectories starting from the initial set tend to infinity, since the system is unstable. Safety is therefore violated with a closed safe region set.

Figure 4(a) shows the zero level set of the quadratic CBC $B_1(x)$. With controller $u_1(x)$ and $u_2(x)$, vector field in (15) guarantees safety with avoiding the unsafe set. For this case, the system admits an ellipsoidal control invariant set. The level sets of $u_1(x)$ and $u_2(x)$ are shown in Figure 4(b)-4(c). It can be seen that $u(x) \in \mathcal{U}$ for any $x \in B_1$.

4.3 Comparison with Control Barrier Functions

From the point of view of set invariance, the zero-super level set of both CBC and CBF are control invariant. CBC, which is a direct interpretation of control invariance to ensure safety, takes initial conditions into consideration as well - without initial conditions, the CBC formulation is equivalent to CBF. Although the definition of CBF involves the existence of a class- \mathcal{K} function, this, however is a straightforward property that holds for both CBC and CBF.

From the aspect of controller design, the CBF-QP approach relies on a given safe control invariant set, which is free for our approach (5). For the case where the control invariant set is constructed *a priori*, although the CBF approach endows Lipschitz continuity for the resulting controller, it also introduces unnecessary conservatism since $\dot{B}_2(x)$ is bounded by a *fixed* additional relaxation term.

5. CONCLUSION

In this paper we investigate the problem of safety verification and controller design for safety critical systems. Our approach depends on the evaluation of a control invariant set which encloses the initial set while avoiding

the unsafe set. This formulation only imposes boundary conditions, thus alleviating conservatism. For polynomial systems with semi-algebraic initial and safe sets, we propose an iterative procedure using SOS programming to synthesize a CBC, where the control input is restricted in axis-aligned hyper-rectangular sets. We also show that the CBC has less conservative compared to CBF by means of numerical simulations. Future work aims at extending the formulation to discrete time systems.

REFERENCES

- Ames, A.D., Xu, X., Grizzle, J.W., and Tabuada, P. (2016). Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8), 3861–3876.
- Blanchini, F. (1999). Set invariance in control. *Automatica*, 35(11), 1747–1767.
- Chen, Y., Jankovic, M., Santillo, M., and Ames, A.D. (2021). Backup control barrier functions: Formulation and comparative study. *arXiv preprint arXiv:2104.11332*.
- Chen, Y., Peng, H., and Grizzle, J. (2017). Obstacle avoidance for low-speed autonomous vehicles with barrier function. *IEEE Transactions on Control Systems Technology*, 26(1), 194–206.
- Clark, A. (2021). Verification and synthesis of control barrier functions. *arXiv preprint arXiv:2104.14001*.
- Cunis, T. and Kolmanovsky, I. (2021). Viability, viscosity, and storage functions in model-predictive control with terminal constraints. *Automatica*, 131, 109748.
- Freeman, R.A. and Primbs, J.A. (1996). Control lyapunov functions: New ideas from an old source. In *Proceedings of 35th IEEE Conference on Decision and Control*, volume 4, 3926–3931. IEEE.
- Guiochet, J., Machin, M., and Waeselynck, H. (2017). Safety-critical advanced robots: A survey. *Robotics and Autonomous Systems*, 94, 43–52.
- Hsu, S.C., Xu, X., and Ames, A.D. (2015). Control barrier function based quadratic programs with application to bipedal robotic walking. In *2015 American Control Conference (ACC)*, 4542–4548. IEEE.
- Lygeros, J. (2004). On reachability and minimum cost optimal control. *Automatica*, 40(6), 917–927.
- Margellos, K. and Lygeros, J. (2011). Hamilton-jacobi formulation for reach-avoid differential games. *IEEE Transactions on automatic control*, 56(8), 1849–1861.
- Papachristodoulou, A., Anderson, J., Valmorbida, G., Prajna, S., Seiler, P., and Parrilo, P.A. (2013). *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*. <http://arxiv.org/abs/1310.4716>. Available from <http://www.eng.ox.ac.uk/control/sostools>, <http://www.cds.caltech.edu/sostools> and <http://www.mit.edu/~parrilo/sostools>.
- Prajna, S. and Jadbabaie, A. (2004). Safety verification of hybrid systems using barrier certificates. In *International Workshop on Hybrid Systems: Computation and Control*, 477–492. Springer.
- Prajna, S., Jadbabaie, A., and Pappas, G.J. (2007). A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8), 1415–1428.
- Prajna, S., Papachristodoulou, A., and Parrilo, P.A. (2002). Introducing sostools: A general purpose sum

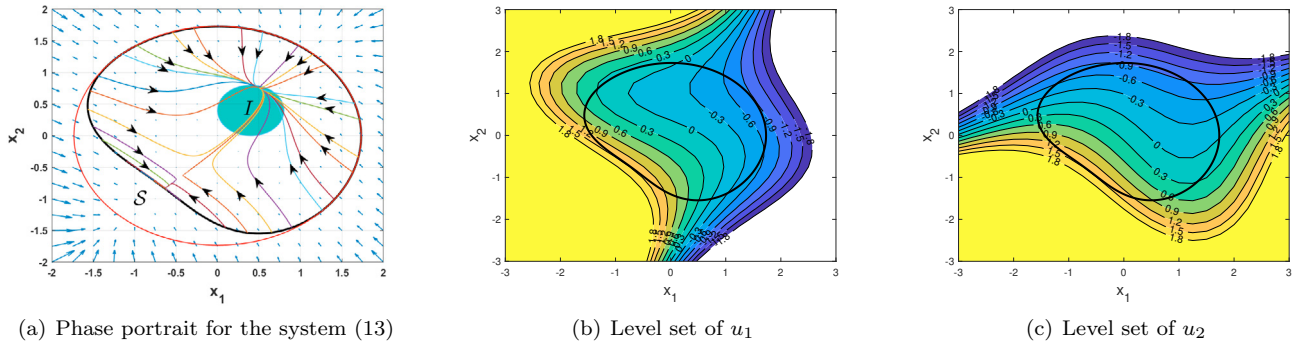


Fig. 3. The interior of the red disc represents the safe set, the interior of the blue disc represents the initial set from which the trajectories start. The black closed curve encircling the initial set is the control invariant set, defined by the super-zero level set of $B_1(x)$. The arrows in the figure represent the vector field. The colorful lines are the trajectories starting from ∂B_1 .

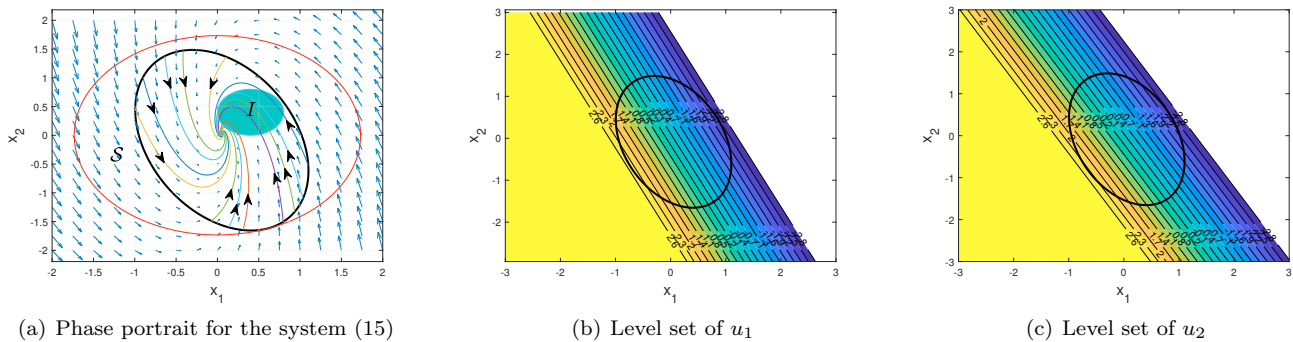


Fig. 4. The safe and initial set are defined to be the same as in Figure 3. Safety is ensured with the polynomial control law.

- of squares programming solver. In *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, volume 1, 741–746. IEEE.
- Prajna, S. and Rantzer, A. (2005). On the necessity of barrier certificates. *IFAC Proceedings Volumes*, 38(1), 526–531.
- Robey, A., Hu, H., Lindemann, L., Zhang, H., Dimarogonas, D.V., Tu, S., and Matni, N. (2020). Learning control barrier functions from expert demonstrations. In *2020 59th IEEE Conference on Decision and Control (CDC)*, 3717–3724. IEEE.
- Srinivasan, M., Dabholkar, A., Coogan, S., and Vela, P.A. (2020). Synthesis of control barrier functions using a supervised machine learning approach. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 7139–7145. IEEE.
- Sturm, J.F. (1999). Using sedumi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4), 625–653. doi:10.1080/10556789908805766. URL <https://doi.org/10.1080/10556789908805766>.
- Tan, X., Cortez, W.S., and Dimarogonas, D.V. (2021). High-order barrier functions: Robustness, safety and performance-critical control. *IEEE Transactions on Automatic Control*.
- Vidyasagar, M. (2002). *Nonlinear systems analysis*. SIAM.
- Wang, L., Han, D., and Egerstedt, M. (2018). Permissive barrier certificates for safe stabilization using sum-of-squares. In *2018 Annual American Control Conference (ACC)*, 585–590. IEEE.
- Xiao, W. and Belta, C. (2019). Control barrier functions for systems with high relative degree. In *2019 IEEE 58th conference on decision and control (CDC)*, 474–479. IEEE.
- Xiao, W., Belta, C., and Cassandras, C.G. (2021). Adaptive control barrier functions. *IEEE Transactions on Automatic Control*.
- Xu, X., Grizzle, J.W., Tabuada, P., and Ames, A.D. (2017). Correctness guarantees for the composition of lane keeping and adaptive cruise control. *IEEE Transactions on Automation Science and Engineering*, 15(3), 1216–1229.
- Zeng, J., Zhang, B., Li, Z., and Sreenath, K. (2021). Safety-critical control using optimal-decay control barrier function with guaranteed point-wise feasibility. In *2021 American Control Conference (ACC)*, 3856–3863. IEEE.