

Article

High-Order Control Lyapunov–Barrier Functions for Real-Time Optimal Control of Constrained Non-Affine Systems

Alaa Eddine Chriat * and Chuangchuang Sun

Aerospace Engineering Department, Mississippi State University, Starkville, MS 39762, USA;
csun@ae.msstate.edu

* Correspondence: aec652@msstate.edu

Abstract: This paper presents a synthesis of higher-order control Lyapunov functions (HOCLFs) and higher-order control barrier functions (HOCBFs) capable of controlling nonlinear dynamic systems while maintaining safety. Building on previous Lyapunov and barrier formulations, we first investigate the feasibility of the Lyapunov and barrier function approach in controlling a non-affine dynamic system under certain convexity conditions. Then we propose an HOCLF form that ensures convergence of non-convex dynamics with convex control inputs to target states. We combine the HOCLF with the HOCBF to ensure forward invariance of admissible sets and guarantee safety. This online non-convex optimal control problem is then formulated as a convex Quadratic Program (QP) that can be efficiently solved on board for real-time applications. Lastly, we determine the HOCLBF coefficients using a heuristic approach where the parameters are tuned and automatically decided to ensure the feasibility of the QPs, an inherent major limitation of high-order CBFs. The efficacy of the suggested algorithm is demonstrated on the real-time six-degree-of-freedom powered descent optimal control problem, where simulation results were run efficiently on a standard laptop.

Keywords: control barrier functions; safety-critical control; real-time control; powered descent

MSC: 49-00



Citation: Chriat, A.E.; Sun, C. High-Order Control Lyapunov–Barrier Functions for Real-Time Optimal Control of Constrained Non-Affine Systems. *Mathematics* **2024**, *12*, 4015. <https://doi.org/10.3390/math12244015>

Academic Editors: Zhijia Zhao and Asier Ibeas

Received: 13 November 2024

Revised: 6 December 2024

Accepted: 20 December 2024

Published: 21 December 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Since the mid-1950s, fast technological advancements have been significantly influenced by the demands of the automotive, robotics, and aerospace industries for precision, reliability, and robustness. With this advancement, modern control theory emerged with multiple techniques to handle complex uncertain systems, leveraging digital computers to perform complex calculations. Unlike classical control theory, which relies heavily on model simplifications and frequency domain techniques, modern control theory focuses on using state-space methods and various estimation techniques like optimal and adaptive control to control complex dynamic systems. Modern techniques were enhanced to be more robust and reliable over time, such as nonlinear model predictive control or model-free adaptive control [1–4]. Nevertheless, these techniques still have some shortcomings such as model dependency, computational complexity, and implementation challenges.

Researchers have since worked on new forms of model predictive control which leverage the use of a sequence of techniques like linearization and discretization to generate an optimal control problem that can be iteratively solved like model predictive control. These techniques offer excellent reliability and robustness but still suffer from computational cost and complexity in real-time implementation. The work in [5–7] synthesizes the second-order cone program (SOCP), conditionally enforced constraints, and linearization and discretization procedures to transform highly complex nonlinear systems with non-convex state and control into a convexified problem and demonstrates its performance on the 6DoF powered descent guidance problem as a feed-forward trajectory generation problem.

Although successive convexification (SCvx) can provide a real-time control solution for complex systems, it has some significant drawbacks. First, linearization and discretization in a real-time implementation abstract the higher-order features of the problem, rendering the convex model inadequate outside the nominal states. This abstraction possibly raises convergence issues, lowers the confidence in maintaining safety constraints, or even leads to unstable control behavior. Second, SCvx can encounter infeasibility and thus requires good initialization, which can be challenging to acquire for some nominal states, especially for systems with fast dynamics, making SCvx vulnerable to disturbances and model uncertainties. Ultimately, as SCvx requires iteratively solving a sequence of convex optimization problems at each step, real-time implementation can impose a significant computational burden when handling complex problems with large state and control spaces, which may lead to delays in decision-making.

In recent years, a new theory emerged that shows the capability of controlling nonlinear systems while ensuring safety despite non-convex constraints. Based on the work in [8–10], which lays the mathematical foundation for Lyapunov functions and their application in null-asymptotic controllability, researchers have synthesized a control Lyapunov function and control barrier function approach to design safe, deployable controllers. The work in [11,12] uses barrier functions as a novel methodology for specifying and strictly enforcing safety conditions. When unified with control Lyapunov functions in the context of a quadratic program, CLFs and CBFs provide real-time optimization-based controllers capable of controlling and ensuring the safety of nonlinear dynamics and non-convex safety constraints in conjunction with performance objectives. This approach proved effective in numerous applications, including automotive control problems like adaptive cruise control and lane-keeping. This approach was extended for higher-order systems [13] and infers asymptotic stability by examining the second- and third-order derivatives of the Lyapunov functions and argues the crucial role that higher-order derivatives play in establishing analytical stability. Recently, this result has been adapted and generalized to higher-order derivatives and time-varying systems by parameterizing Lyapunov functions and imposing some conditions on their higher-order derivatives [14], while, for safety, the authors of [15,16] proposed a sequence of time-varying barrier functions to ensure the intersection of all subsets of the safety constraint remains forward invariant. This approach extends the concept of Lyapunov and barrier functions to higher-order control Lyapunov barrier functions (HOCLBFs), designed specifically for safely constraining higher-order systems subject to high-relative-degree constraints.

Adopting the control Lyapunov and barrier functions approach proves advantageous for designing online real-time controllers that can enforce safety constraints and manage non-convex constraints and dynamics, as the control problem reduces solving a straightforward, computationally efficient quadratic program at each time step. Nonetheless, the existing literature requires a control affine system for using CLFs and CBFs. Additionally, the unification of these control functions suffers several limitations when subject to different constraints, which renders the quadratic program infeasible or inaccurate or results in divergent behaviors. Finally, this approach has been adopted for simple low-dimensional problems such as 2D robots or adaptive cruise control and has yet to prove its efficacy in complex systems such as aerospace applications with highly nonlinear dynamics and uncertain environments that demand real-time, adaptive, and reliable control systems.

1.1. Motivation and Contributions

Our motivation in this work is to attract more research on the foundation of the Lyapunov and barrier functions and their feasibility in controlling complex systems, as generalizing the approach to a broader scope of problems such as non-affine systems can reveal more insights into the theory and design of robust controllers in addition to the broader scope of problems that can be tackled using this approach. In this work, we aim to enhance the current advances and overcome the drawbacks of this method. Finally, we aim to demonstrate the feasibility of this approach in controlling a highly complex control

problem. In this paper, we start initially by investigating the applicability of CLFs and CBFs to non-affine systems and then contemplate the current forms of CLFs and CBFs and enhance the formulation to require less tunable parameters. Finally, we apply this approach to the 6DoF powered descent problem to showcase the ability of CLFs and CBFs to control high-dimensional complex systems. Our contribution can be summarized in the following:

1. We establish sufficient conditions for generalizing CLFs and CBFs to non-affine control systems.
2. We propose a computationally efficient higher-order CLF formulation and address the different infeasibilities that arise from unifying high-order CLFs and CBFs due to the large number of tunable parameters using fewer parameters and a heuristic approach to tune them.
3. We illustrate the efficacy of our approach by devising an online real-time optimal controller for six-degree-of-freedom powered descent with non-affine inputs, considering a range of non-convex safety constraints.

This work puts an effort toward the unification of the CLF and CBF theory and achieving a concise robust form, inviting more future research in developing robust controllers and coupling these controllers with machine learning algorithms. Figure 1 illustrates the HOCLBF approach in the general context of optimal control theory.

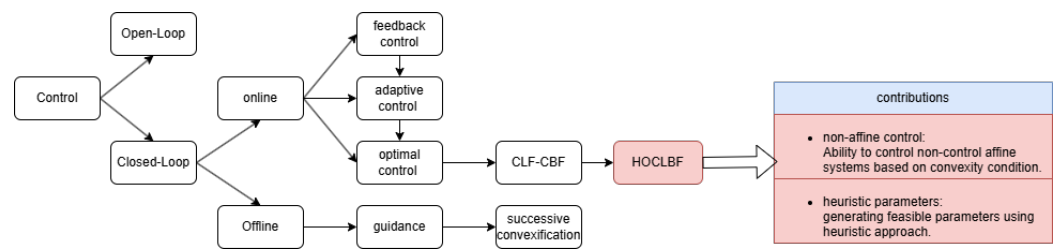


Figure 1. Flowchart representing a control theory overview situated in the general context and explaining the contributions made.

1.2. Organization

The rest of the paper is organized as follows. Section 2 reviews the mathematical background for control Lyapunov and barrier functions. In Section 3, we expand the scope of the CLFs and CBFs beyond control affine systems, suggest a new high-order CLF formulation, synthesize the CLF-CBF approach, and address the potential infeasibility sources using both heuristic and classical approaches. In Section 4, we demonstrate the efficiency of the HOCLBF approach in controlling 6-DOF powered descent, along with numerical simulation results presented in Section 4.1.5. Conclusions and possible future directions of this work are detailed in Section 5.

2. Preliminaries

This section briefly reviews some fundamental definitions of stabilizing control Lyapunov and barrier functions, while we refer the reader to [8–16] for a deeper understanding of these concepts. Consider the nonlinear control-affine system:

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

where f and g are globally Lipschitz, $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ are the states and control inputs, respectively, constrained in closed sets, with initial condition $x(t_0) = x_0$.

Definition 1 ([10–12]). A continuous differentiable function $V : \mathbb{R}^n \rightarrow \mathbb{R}^+$ is an exponentially stabilizing control Lyapunov function if it is positive, proper, infinitesimally decreasing, and there exists $c_1, c_2, c_3 > 0$, such that for any solution $x(t)$:

$$\begin{aligned} c_1 \|x_1\|^2 &\leq V(x) \leq c_2 \|x\|^2 \\ \inf_{u \in U} [L_f V(x) + L_g V(x)u + c_3 V(x)] &\leq 0 \end{aligned} \quad (2)$$

The Lyapunov function yields a feedback controller that exponentially stabilizes and drives the system to the desired output/zero dynamics, with c_3 representing the rate of convergence of the solution:

$$\|x(t)\| \leq \sqrt{\frac{c_2}{c_1}} e^{-\frac{c_3 t}{2}} \|x(0)\|, \quad (3)$$

and this controller is equivalent to a min-norm controller that can be solved via a quadratic program, which proved efficient in experimental application [17,18].

$$\begin{aligned} \min_{u \in U} u^T u \\ \text{s.t. } L_f V(x) + L_g V(x)u + c_3 V(x) &\leq 0 \end{aligned} \quad (4)$$

Definition 2 ([19]). *The relative degree of a differentiable function refers to the number of differentiations required to obtain the control input u explicitly shown in the control function.*

Since most systems are not first-order in control inputs, we need to ensure stability using the higher-order derivatives of the Lyapunov function. We can infer asymptotic stability by examining up to the third derivative of the Lyapunov function.

Definition 3 ([13]). *If the function $V(x)$ is three times differentiable, and there exist $a_1, a_2 > 0$ such that:*

$$a_1 \ddot{V}(x) + a_2 \dot{V}(x) + \dot{V}(x) \leq 0 \quad (5)$$

then $V(x)$ is an asymptotic stabilizing Lyapunov function.

In addition to stability requirements, the nonlinear system (1) is often subject to critical safety measures. These safety measures can be stated as a set C defined by some continuous function $h(x)$ as:

$$\begin{aligned} C &= \{x \in \mathbb{R}^n : h(x) \geq 0\} \\ \partial C &= \{x \in \mathbb{R}^n : h(x) = 0\} \\ \text{Int}(C) &= \{x \in \mathbb{R}^n : h(x) > 0\} \end{aligned} \quad (6)$$

where ∂C is the boundary of the set representing the critical safety limits, and $\text{Int}(C)$ is the interior of the set constituting all the permissible/safe states. We can use control barrier functions to ensure the solutions $x(t)$ will not leave the safe set C , $\forall t \geq 0$ [20]. The CBF is a new mathematical technique used to guarantee the safety and stability of a dynamic system while achieving the desired goals. CBFs are scalar functions that quantify the system's safety in the current state. The control law is then designed to keep the system within the safety region defined by the CBFs. Mathematically, consider the nonlinear control affine system (1).

Definition 4 ([12]). *If there exists a continuous function $\alpha(\bullet) : [-b, a] \in \mathbb{R} \rightarrow \mathbb{R}$ that is strictly increasing and $\alpha(0) = 0$, then α belongs to the extended $\mathcal{K}$ class functions.*

Definition 5 ([12]). *The function $h : \mathbb{R}^n \rightarrow \mathbb{R}$ is a zeroing barrier function for the safe set $C = \{x \in \mathbb{R}^n : h(x) \geq 0\}$ if there exists an extended $\mathcal{K}$ class function $\alpha(\bullet)$ such that:*

$$\begin{aligned} \sup_{u \in U} [L_f h(x) + L_g h(x)u + \alpha(h(x))] &\geq 0 \\ \inf_{\text{int}(C)} [\alpha(h(x))] &\geq 0 \quad \text{and} \quad \lim_{\partial C} \alpha(h(x)) = 0 \end{aligned} \quad (7)$$

The existence of the CBF $h(x)$ renders the set C forward invariant.

In the same way as Lyapunov functions, control barrier functions require that the control input u appears in $h(x)$. In [16], they propose a higher-order CBF to handle higher-relative-order constraints.

Definition 6 ([16]). For the nonlinear system (1) with the m^{th} differentiable constraint $h(x)$, we define a sequence of functions ψ_i starting from $\psi_0 = h(x)$ and a sequence of safe sets $C_i(t)$ associated with each ψ_i , with $i \in \{1, 2, \dots, m\}$:

$$\begin{aligned}\psi_i(x, t) &= \dot{\psi}_{i-1}(x, t) + \alpha_i(\psi_{i-1}(x, t)) \\ C_i(t) &= \{x \in \mathbb{R}^n : \psi_{i-1}(x, t) \geq 0\}\end{aligned}\quad (8)$$

The function ψ_m is a high-order control barrier function if there exist extended $\mathcal{K}$ class functions $\alpha_i(\bullet)$ such that:

$$\sup_{u \in U} [L_f^m h(x) + L_g L_f^{m-1} h(x, t) u + \frac{\partial^m h(x, t)}{\partial t^m} + \mathcal{O}(h(x, t)) + \alpha_m(\psi_{m-1}(x, t))] \geq 0 \quad (9)$$

$\forall x(t) \in C_1(t) \cap \dots \cap C_m(t), \forall t > 0$, under the condition that $x(0) \in C_1(0) \cap \dots \cap C_m(0)$.

Ensuring $\psi_m \geq 0$ guarantees the forward invariance of the set C_{m+1} and, subsequently, the forward invariance of all the sets $C_{m-i}(t)$, hence ensuring the system's safety within the set $C_1(t)$, similar to Definition 5. To guarantee the feasibility of the HOCBF, the authors of [16] suggested multiplying the class $\mathcal{K}$ functions $\alpha(\bullet)$ by a penalty term p_i and solving iteratively to relax the problem when needed:

$$\psi_i = \dot{\psi}_{i-1} + p_i \alpha_i(\psi_{i-1}) \quad (10)$$

3. A QP Formulation Based on High-Order Control Lyapunov–Barrier Functions

In this section, we synthesize HOCLFs and HOCBFs to control nonlinear systems. We first establish some conditions to extend the use of CLFs and CBFs for controlling and constraining non-affine nonlinear systems. Next, we propose a new form for higher-order CLFs inspired by the analytical first-order CLF in [10–12], explicitly displaying the rate of convergence and the computational high-order CLF proposed in [13] incorporating the higher-order derivative of the Lyapunov function. Finally, we suggest a classical controller to address the contrast between the CLFs and a heuristic approach to handle the infeasibility in the QP arising from CLF and CBF conflicts.

Starting from the first formulation of Lyapunov functions [8], Artstein's theorem considers a general dynamic system of the form $\dot{x} = f(x, u)$ and argues its global asymptotic stability under the condition of finding a continuous function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ such that:

$$\begin{aligned}V(0) &= 0 \quad x = 0 \\ V(x) &> 0 \quad \forall x \neq 0 \\ \inf_{u \in U} [\nabla V \cdot f(x, u)] &< 0 \quad \forall x \neq 0\end{aligned}\quad (11)$$

A smooth mapping $v(x) : \mathbb{R} \rightarrow U$ is constructed to achieve the global asymptotic stability of the system with the feedback law $\dot{x} = f(x, v(x))$, where $v(x)$ belongs to a convex set and U is a convex polytope $U = \{u \in \mathbb{R}^m \mid A_0 u \leq b_0\}$. It is also suggested to consider a piece-wise linear function to ensure Lipschitz continuity and guarantee the existence of a control law $v(x)$. The linearity in input was further established in the work conducted in [9], where a universal Artstein theorem is constructed for systems with affine inputs. Since Artstein's theorem guarantees the stability of a general dynamic system, we argue that we can use analytical Lyapunov and barrier functions to stabilize non-affine nonlinear systems using convex optimization under certain conditions.

In this work, we consider a general dynamic system of the form:

$$\dot{x}(t) = f(x(t)) + g(x(t), u(t)) \quad (12)$$

where $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ are the states and control inputs, respectively. f and g are globally Lipschitz, and g is at least twice differentiable. We also consider a continuously differentiable positive-definite function $V : \mathbb{R}^n \rightarrow \mathbb{R}_+$ to be an exponentially stabilizing control Lyapunov function yielding the following controller:

$$\begin{aligned} \min_{u \in U} \quad & u^T u \\ \text{s.t.} \quad & [\nabla V \cdot \dot{x} + cV(x)] \leq 0 \end{aligned} \quad (13)$$

Theorem 1. For a general Lyapunov function $V : \mathbb{R}^n \rightarrow \mathbb{R}_+$, the min-norm controller (13) is a convex quadratic program if $[\nabla V(x) \cdot g(x, u)]$ is convex, requiring that $\nabla V(x)$ and $g(x, u)$ are both convex and positive functions (or concave and negative) and have the same monotonicity. Since $\nabla V(x)$ is a function of the states only, it is sufficient that $[\nabla V(x) \cdot \frac{\partial^2 g(x, u)}{\partial u^2}] \geq 0$. That is, $\nabla V(x)$ is positive and $g(x, u)$ is convex (or negative and concave) in $\text{dom}(V) \cap \text{dom}(g)$, respectively.

Proof. We prove the first statement by verifying Jensen's inequality for the product $[\nabla V(x) \cdot g(x, u)]$. If $\nabla V(x)$ and $g(x, u)$ are positive, convex, and have the same monotonicity, then for $0 \leq \theta \leq 1$, we find:

$$\nabla V(x)g(x, \theta u_1 + (1 - \theta)u_2) \leq \theta \nabla V(x)g(x, u_1) + (1 - \theta)\nabla V(x)g(x, u_2)$$

We prove the second statement by verifying the second-order convexity condition. Taking the first derivative of $[\nabla V(x) \cdot \dot{x}]$ with respect to the control input eliminates $[\nabla V(x) \cdot f(x)]$ and yields

$$\frac{d[\nabla V(x) \cdot \dot{x}]}{du} = \frac{\partial \nabla V(x)}{\partial u} \cdot g(x, u) + \nabla V(x) \cdot \frac{\partial g(x, u)}{\partial u}$$

Taking the second derivative yields

$$\frac{d^2[\nabla V(x) \cdot \dot{x}]}{du^2} = \frac{\partial^2 \nabla V(x)}{\partial u^2} \cdot g(x, u) + \nabla V \cdot \frac{\partial^2 g(x, u)}{\partial u^2} + 2 \frac{\partial \nabla V(x)}{\partial u} \cdot \frac{\partial g(x, u)}{\partial u},$$

and canceling the term involving $\partial \nabla V(x)$, we obtain

$$\frac{\partial^2[\nabla V(x) \cdot \dot{x}]}{\partial u^2} = \nabla V(x) \cdot \frac{\partial^2 g(x, u)}{\partial u^2}.$$

Therefore, the convexity of the feasible set in the quadratic program depends on the sign of $\nabla V(x) \cdot \frac{\partial^2 g(x, u)}{\partial u^2}$. $\square$

Remark 1. With similar reasoning, we can prove that the CBF in (7) defines a convex feasible set for the quadratic program if $[\nabla h(x) \cdot \frac{\partial^2 g(x, u)}{\partial u^2}] \leq 0$.

Remark 2. A control affine system is a special case as $[\nabla V \cdot \frac{\partial^2 g(x, u)}{\partial u^2}] = [\nabla h(x) \cdot \frac{\partial^2 g(x, u)}{\partial u^2}] = 0$.

Example 1. The following example demonstrates an extreme case where both $[\nabla V(x) \cdot g(x, u)]$ change sign within $\text{dom}(V) \cap \text{dom}(g)$ while keeping the convexity condition valid. Consider the system

$$\dot{x} = 10x^2 - 5x\sqrt{u} \quad (14)$$

We can drive and stabilize this system from an arbitrary x_0 to the unstable equilibrium point at $x_f = [0 \ 0]$ by using the Lyapunov function $V(x) = x^2$. This results in

$$\nabla V(x) = 2x, \frac{\partial^2 g(x, u)}{\partial u^2} = \frac{x}{4\sqrt{u^3}}.$$

By examining $\nabla V(x)$ and $\frac{\partial^2 g(x, u)}{\partial u^2}$, we can see a change of sign between $x \in \mathbb{R}_+^n$ and $x \in \mathbb{R}_-^n$; nevertheless, $[\nabla V(x) \cdot \frac{\partial^2 g(x, u)}{\partial u^2}] \geq 0$ for $x \in \mathbb{R}^n$. Hence, we conclude controllability via the control Lyapunov function

$$\text{CLF} := 20x^3 - 10x^2\sqrt{u} + cx^2 \leq 0$$

Figure 2 shows the state trajectory, control trajectory, and phase portrait of the unforced system.

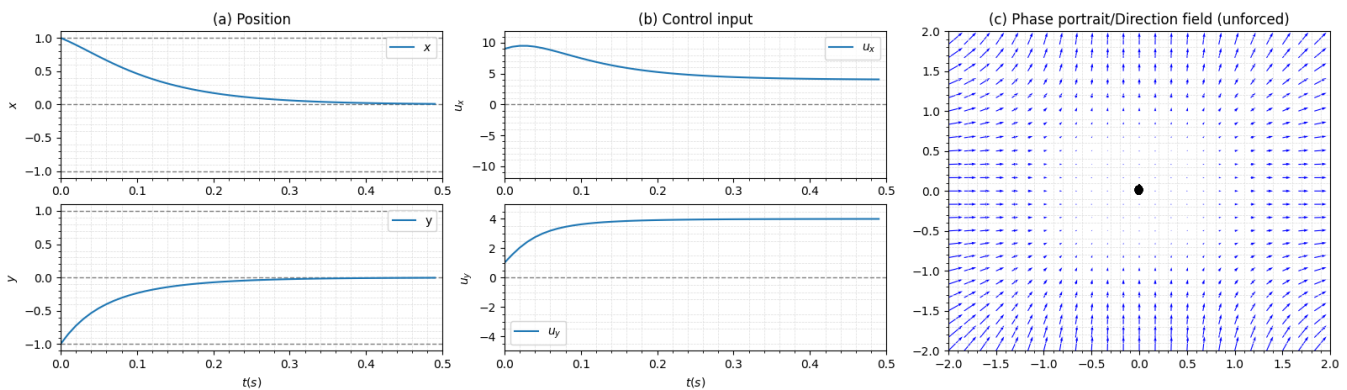


Figure 2. (a) The state trajectory starting from $x_0 = [1 \ -1]$ and reaching $x_f = [0 \ 0]$ for $c = 20$. (b) The control trajectory driving the system to $x_f = [0 \ 0]$. (c) The phase portrait of the unforced system, with blue arrows indicating the existence of an unstable equilibrium point in $x_{eq} = [0 \ 0]$.

Next, we propose a new computationally efficient form for higher-order CLFs that can exponentially stabilize nonlinear systems. Starting from the HOCLF proposed in [13], we ensure the satisfaction of positiveness and zero dynamics by the initial choice of the Lyapunov function, e.g., taking a quadratic function of the difference between the desired and actual solution guarantees $V(x) > 0, \forall t$ and $V(0) = 0$.

$$V(x, x_d) = (x_{desired}(t_f) - x(t))^2 \quad (15)$$

We also omit the scalars τ_i and add the Lyapunov function multiplied by a scalar c that represents the rate of convergence of the CLF to the sum of Lyapunov derivatives and obtain the following HOCLF:

$$\inf_{u \in U} [V^m + V^{m-1} + \dots + \dot{V} + cV] \leq 0 \quad (16)$$

Combining high-order Lyapunov and barrier functions can often encounter infeasibilities. The first source of infeasibility comes from the conflicts of goals between the CLFs and CBFs. This problem was overcome in [11,12,21] by adding a relaxing (softening) term δ to the CLFs, which allows them to diverge from the desired states to guarantee the satisfaction of CBF constraints. The second infeasibility arises from the HOCLF and HOCBF coefficients when their constraints conflict with the constraint of the control input bounds. Although [16] suggested an iterative approach to change the parameters until they make the QP feasible, it relaxes the safety constraints and has a marginally incremental impact on computation time and power. Alternatively, we adopt a heuristic approach where the quadratic program solves for approximate solutions when an optimal solution is impractical or impossible to find, and this is achieved by maximizing the rates of convergence for the CLF and the Kappa function coefficients with a tunable upper bound

while minimizing the relaxing terms for the CLF. The last infeasibility to overcome stems from conflicts between the multiple goals of CLFs; e.g., a position CLF driving the system towards a desired position conflicts with the velocity CLF trying to attenuate the velocity to zero, resulting in a quadratic program infeasibility. To overcome this issue, we suggest using state-varying desired states. In other words, instead of using the final desired states in the CLFs, we compute intermediate desired states between the initial and final states. We suggest two methods for determining intermediate states.

Navigational Planner: We can use a navigation algorithm between the initial and final states and obtain a sequence of plausible intermediate states; nonetheless, this method has some distinctive drawbacks. Employing the navigation algorithm offline leaves it myopic to the current state and trajectory; therefore, its intermediate states can become unreachable as the current trajectory progresses, especially for highly nonlinear coupled dynamics. On the other hand, using the algorithm online can generate excellent intermediate states; however, it requires significantly higher computational power and time.

PID Planner: We can use an online PID planner to generate the subsequent intermediary desired state based on our current and final state. We can define the error between the current and final state as $e_x(t) = x_{desired}(t_f) - x(t)$, yielding a PID planner as follows:

$$x_{desired}(t + dt) = K_P e_x(t) + K_I \int_0^{t_f} e_x(\tau) d\tau + K_D \frac{de_x(t)}{dt} \quad (17)$$

where K_P, K_I, K_D are tunable parameters. We adopt this approach in our work as it provides an exceptional compromise between generating online, plausible, reachable intermediate states and considerable computational efficacy. The final form of the optimal quadratic program controller we suggest is as follows:

$$\begin{aligned} \min_{u, \delta, q, c} : & u^T H(x) u + \delta^T Q(x) \delta - P^T(x) q - R^T(x) c \\ \text{s.t.} : & V_i^m(x_{PID}) + V_i^{m-1}(x_{PID}) + \dots + \dot{V}_i(x_{PID}) + C_i V_i(x_{PID}) \leq \delta_i & \text{HOCLF} \\ & \dot{V}_j(x_{PID}) + C_j V_j(x_{PID}) \leq \delta_j & \text{CLFs} \\ & \psi_{m,0,k}(x_{PID}) + \psi_{m,1,k}(x_{PID}) + \alpha_{m,k}(\psi_{m-1,k})(x_{PID}) \geq 0 & \text{HO-ZCBFs} \\ & \Phi_{0,0,p}(x_{PID}) + \Phi_{0,1,p}(x_{PID}) + \alpha_p(\Phi_0)(x_{PID}) \geq 0 & \text{ZCBFs} \\ & A_0 u \leq b_0 & \text{Control limitation} \\ & \text{Other DCP constraints} \end{aligned} \quad (18)$$

where H, Q, P , and R are weighing coefficients used to tune the system's behavior.

4. Simulations and Results

4.1. 6-DOF Optimal Powered Descent

To demonstrate the proposed HOCLBF framework on a more complex system, we address the problem of controlling a 6-DOF powered descent. The model we use in this paper is inspired by [7,22,23] but tailored to accommodate our application. In our model, we assume a single 2D gimbal engine that can continuously vary between minimum and maximum thrusts; the rocket is a rigid body with a constant CG and inertia and we neglect any deformations. We assume the dynamics are mainly governed by inertial and thrust forces and neglect aerodynamic forces based on the findings of [7]. Lastly, we ignore planetary motion and assume a constant gravity throughout the rocket trajectory.

4.1.1. Dynamic Model

To fully describe the dynamics of the 6-DOF rocket, similar to [7,23] we assume the fuel dynamics are governed by an affine function of the thrust magnitude and neglect the atmospheric back pressure:

$$\dot{m}(t) = -\alpha_m \|T_B\|_2 \quad (19)$$

where $\alpha_{\dot{m}} = \frac{1}{I_{sp}g_0}$. I_{sp} is the vacuum specific impulse of the engine, and g_0 is standard Earth gravity. We describe the translational and rotational dynamics with six differential equations as follows.

$$\begin{aligned}
 \dot{r}_I(t) &= C_{I \leftarrow B}(q_I(t))v_b \\
 \dot{v}_B(t) &= \frac{1}{m(t)}[T(t) + A(t)] + E_{B \leftarrow I}(q_I(t))g_I - \omega_B \times v_B \\
 \ddot{r}_I(t) &= C_{I \leftarrow B}(q_I(t))\dot{v}_B + \dot{C}_{I \leftarrow B}(q_I(t))v_B \\
 \dot{q}_I(t) &= E_{I \leftarrow B}(q_I(t))\omega_B \\
 \dot{\omega}_B(t) &= \tilde{J}_B^{-1} \left[r_{T,B} \times T_B(t) + r_{cp,B} \times A(t) \right] - \tilde{J}_B^{-1}(\omega_B \times \tilde{J}\omega_B) \\
 \ddot{q}_I(t) &= E_{I \leftarrow B}(q_I(t))\dot{\omega}_B + \dot{E}_{I \leftarrow B}(q_I(t))\omega_B
 \end{aligned} \tag{20}$$

We add the second derivative of the position and attitude in the inertial frame. We make these adjustments to accommodate the Lyapunov approach efficiently.

4.1.2. High-Order Control Lyapunov Functions

To drive the states to the desired ones, we need to formulate positive Lyapunov functions that are twice differentiable as the order of the position dynamics is $m = 2$. Hence, we use the following quadratic Lyapunov function which satisfies positiveness and the zero dynamics point and yields:

$$\begin{aligned}
 V_i &= \frac{1}{2} \|s_i(t) - s_{id}\|_2^2 \\
 \dot{V}_i &= \dot{s}_i(t)(s_i(t) - s_{id}) \\
 \ddot{V}_i &= \ddot{s}_i(t)(s_i(t) - s_{id}) + \dot{s}_i^2(t) \\
 CLF &:= \ddot{V}_{r_I} + \dot{V}_{r_I} + c_{r_I} V_{r_I} \leq \delta_{r_I}
 \end{aligned} \tag{21}$$

Position CLF:

$$\begin{aligned}
 \ddot{V}_{r_I} &= \frac{1}{m(t)}(r_I(t) - r_d)C_{I \leftarrow B}(q_I(t))T(t) + (r_I(t) - r_d)C_{I \leftarrow B}(q_I(t)) \\
 &\quad \left[\frac{1}{m(t)}A(t) + E_{B \leftarrow I}(q_I(t))g_I - \omega_B(t) \times v_B(t) \right] + (r_I(t) - r_d) \\
 &\quad \dot{C}_{I \leftarrow B}(q_I(t))v_B(t) + \dot{r}_I(t)^2
 \end{aligned}$$

Velocity CLF:

$$\begin{aligned}
 \dot{V}_{v_B} &= \frac{1}{m(t)}(v_B(t) - v_d)T(t) + (v_B(t) - v_d) \\
 &\quad [E_{B \leftarrow I}(q_I(t))g_I - \omega_B(t) \times v_B(t)]
 \end{aligned} \tag{22}$$

Attitude CLF:

$$\begin{aligned}
 \ddot{V}_{q_I} &= (q_I(t) - q_d)E_{B \leftarrow I}(q_I(t))\tilde{J}^{-1}(r_T \times T(t)) + (q_I(t) - q_d)E_{B \leftarrow I}(q_I(t)) \\
 &\quad [\tilde{J}^{-1}(r_{cp} \times A(t) - \omega_B(t) \times \tilde{J}\omega_B(t))] + (q_I(t) - q_d)\dot{E}_{B \leftarrow I}(q_I(t))\omega_B(t) \\
 &\quad + \dot{q}_I(t)^2
 \end{aligned}$$

Angular velocity CLF:

$$\begin{aligned}
 \dot{V}_{\omega_B} &= (\omega_B(t) - \omega_d)\tilde{J}^{-1}(r_T \times T_B(t)) + (\omega_B(t) - \omega_d) \\
 &\quad \left[\tilde{J}^{-1}(r_{cp} \times A(t) - \omega_B(t) \times \tilde{J}\omega_B(t)) \right]
 \end{aligned}$$

Formulating all the Lyapunov functions with the relaxing terms and PID planner ensures convergence to the desired states while maintaining flexibility to diverge when a conflict of goals happens based on the weights of the relaxing terms. Now that smooth convergence is guaranteed, it is crucial to maintain the states within their designated safe

spaces, as described by the safety constraints. This is accomplished through implementing control barrier functions without relaxation to enforce forward invariance for the safe sets.

4.1.3. Safety Constraints

To ensure the safety of the descent, we need to ensure the satisfaction of several non-convex constraints; we start by setting a lower bound on the vehicle mass:

$$m(t) \geq m_{dry} \in R_{++} \Rightarrow m(t) - m_{dry} \geq 0 \quad (23)$$

We also want the rocket to stay inside a glide slope cone with half angle $\gamma_{gs} \in (0, 90^\circ]$ and an origin that coincides with the inertial frame:

$$\hat{e}_3 \cdot \vec{r}_I(t) \geq \cot(\gamma_{gs}) \|\hat{H}_\gamma \vec{r}_I(t)\|_2 \Rightarrow r_I M_{gs} r_I \geq 0 \quad (24)$$

where $\hat{H}_\gamma = [\hat{e}_1, \hat{e}_2]^T$ and $M_{gs} = [[-\cot(\gamma_{gs})^2 \ 0 \ 0]^T [0 \ -\cot(\gamma_{gs})^2 \ 0]^T [0 \ 0 \ 1]^T]$.

To constrain the angular velocities of the rocket, we set an upper bound ω_{max} :

$$\|\omega_B(t)\|_2 \leq \omega_{max} \Rightarrow \omega_{max}^2 - \|\omega_B(t)\|_2^2 \geq 0 \quad (25)$$

Since there are physical limitations on the controls, we need to set upper and lower bounds on the thrust generated by the engine and ensure that the thrust remains within a gimbal angle dictated by the engine:

$$\begin{aligned} 0 \leq T_{min} \leq \|T_B\|_2 \leq T_{max} \\ T_z \geq \cos(\delta_{max}) \|T_B\|_2 \end{aligned} \quad (26)$$

Next, we can specify different boundary conditions reflecting several scenarios for our rocket trajectory. We start from a specified initial rocket mass, position, velocity, and angular velocity:

$$m(0) = m_{in}, \quad r_I(0) = r_{I_{in}}, \quad v_B(0) = v_{B_{in}}, \quad \omega_B(0) = \omega_{B_{in}} \quad (27)$$

and we aim to control the rocket to satisfy the desired final conditions:

$$r_I(t_f) = r_{I_{LS}}, \quad v_B(t_f) = v_{B_{LS}}, \quad q_I(t_f) = q_{I_{LS}}, \quad \omega_B(t_f) = \omega_{B_{LS}} \quad (28)$$

4.1.4. Safety Constraints as Control Barrier Functions

In this section, we formulate a set of first-order and second-order zeroing control barrier functions to confine the system's states within the designated safe sets. We define the extended class $\mathcal{K}$ function as a linear function of the safety function itself $\alpha_i(\psi_j) = \psi_j$ with a coefficient of 1, as it will be multiplied by an optimized parameter to ensure feasibility. Formulating the highest order CBF yields:

$$\begin{aligned} \psi_{1_m} &:= -\alpha_m \|T_B(t)\|_2 \geq -p_{1_m} (m(t) - m_{dry}) \\ \psi_{1_{\omega_B}} &:= -\omega_B(t) \cdot \tilde{J}^{-1}(r_T \times T(t)) \geq -\tilde{J}^{-1}(\omega_B(t) \times \tilde{J}\omega_B) \cdot \omega_B(t) - \frac{p_{1_{\omega_B}}}{2} (\omega_{max}^2 - \|\omega_B\|_2^2) \\ \psi_{2_{gs}} &:= \frac{1}{m(t)} r_I M_{gs} C_{I \leftarrow B}(q_I(t)) T(t) \geq r_I M_{gs} C_{I \leftarrow B}(q_I(t)) [\omega_B(t) \times v_B(t) - E_{B \leftarrow I}(q_I(t)) g_I] \\ &\quad - r_I M_{gs} C_{I \leftarrow B}(q_I(t)) v_B - \dot{r}_I M_{gs} \dot{r}_I - \left(\frac{p_{1_{gs}}}{2} + p_{2_{gs}} \right) \dot{\psi}_{0_{gs}} - \frac{p_{1_{gs}} p_{2_{gs}}}{2} (\psi_{0_{gs}}) \end{aligned} \quad (29)$$

The last step for the barrier functions is replacing the sum and multiplications of parameters $\left(\frac{p_{1_{gs}}}{2} + p_{2_{gs}}\right)$ and $\frac{p_{1_{gs}} p_{2_{gs}}}{2}$ with single parameters $\{p_{1_{gs}}, p_{2_{gs}}\}$, respectively. That is to keep the barrier functions convex in parameters too, enabling the optimization of the parameters inside the quadratic program. Algorithm 1 summarizes the HOCLBF method

of solving a sequence of online convex quadratic programs to control the optimal 6-DOF powered descent in real-time. Model parameters were taken from [7,22,23], with additional parameters either manually tuned or heuristically optimized to ensure adequate performance. Note that we strongly suggest using an adaptive fourth Runge-Kutta for time integration to minimize higher-order truncation errors.

Algorithm 1 HOCLBF for 6-DOF powered descent control.

```

1: Initialize:  $x = x_{initial}, x_{final}, p_{i,min}, c_{i,min}, p_{i,max}, c_{i,max}, dt, \varepsilon, K_{p_i}, K_{I_i}, K_{D_i}$ 
2: while ( $\|x_{final} - x\|_2 \geq \varepsilon$ ) do
3:    $e_x = x_{final} - x$ 
4:    $x_{desired} = K_p e_x + K_I \int e_x d\tau + K_D \frac{de_x}{dt}$ 
5:    $u^* = \arg \min_{u,c,\delta,p} : u^T H u + \delta^T Q \delta - P^T p - R^T c$ 
6:   s.t.:
7:      $\ddot{V}_{r_I} + \dot{V}_{r_I} + c_{r_I} V_{r_I} \leq \delta_{r_I}$  ▷ Position HOCLF (22)
8:      $\ddot{V}_{v_B} + c_{v_B} V_{v_B} \leq \delta_{v_B}$  ▷ Velocity CLF (22)
9:      $\ddot{V}_{q_I} + \dot{V}_{q_I} + c_{q_I} V_{q_I} \leq \delta_{q_I}$  ▷ Attitude HOCLF (22)
10:     $\ddot{V}_{\omega_B} + c_{\omega_B} V_{\omega_B} \leq \delta_{\omega_B}$  ▷ Angular velocity CLF (22)
11:     $c_{(r,v,q,\omega)_{max}} \leq I_4 c_{(r,v,q,\omega)} \leq c_{(r,v,q,\omega)_{max}}$  ▷ Convergence rate bounds
12:     $\psi_{\dot{m},1} + p_{1_m} \psi_{\dot{m},0} \geq 0$  ▷ Dry mass CBF (23)
13:     $\psi_{gs,2} + p_{1_{gs}} \psi_{gs,1} + p_{2_{gs}} \psi_{gs,0} \geq 0$  ▷ Glide slope HOCLBF (24)
14:     $\psi_{\omega_B,1} + p_{1_{\omega_B}} \psi_{\omega_B,0} \geq 0$  ▷ Angular velocity CBF (25)
15:     $\hat{e}_3 \cdot u \geq \cos(\delta_{max}) \|u\|_2$  ▷ Gimbal thrust constraint (26)
16:     $A_T u \leq b_T$  ▷ Thrust magnitude bounds (26)
17:     $x = x + \dot{x}(x, u^*) dt$  or  $x = x + RK_4(\dot{x}, x, u^*, dt)$  ▷ Time propagation (19) and (20)
18: end while

```

4.1.5. Numerical Results

In this section, we assess the performance of the HOCLBF in comparison to the base CLF-CBF and the successive convexification approach. Our goal is to demonstrate the advantage of using the HOCLBF in generating feasible solutions compared to the base CLF-CBF with constant parameters, as well as compare its trajectory to the robust SC algorithm. The comparison is carried out under the condition that all other parameters are kept identical to ensure a consistent assessment. We run our algorithm on a standard laptop using a commercial solver “ECOS” [24] suited for embedded applications and obtain an average runtime of less than 1.1 ms. We start by comparing the high-velocity entry and landing between the three approaches, starting from $r_I(0) = [3.0, 3.0, 4.0]$ and landing on a circular pad $C([0, 0, 0], \rho_v = 0.5)$ in the presence of a 60° glide slope constraint. In Figure 3, we can see that the base CLF-CBF generates infeasible solutions as soon as it gets closer to the glide slope constraint, while the HOCLBF managed to reach the landing site successfully. We can also see some fluctuations in the HOCLBF compared to the smooth SC trajectory.

Figure 4 illustrates the 3D trajectories where we can see that the HOCLBF reaches the landing site successfully, while the base CLF-CBF faces infeasibilities arising from the fixed parameters.

To demonstrate the efficiency of the CBF approach in ensuring the safety of the rocket landing, we generated a trajectory with no glide slope $\gamma_{gs} = 90^\circ$, then for $\gamma_{gs} = 45^\circ, 60^\circ, 75^\circ$ (Figure 5). For the tighter glide slopes, we can see that despite starting outside the safe region, the rocket moves horizontally at a slower descent rate until it enters the safe region and then starts descending faster and successfully reaches the landing site with no infeasibilities in the QP while keeping the safety constraints satisfied.

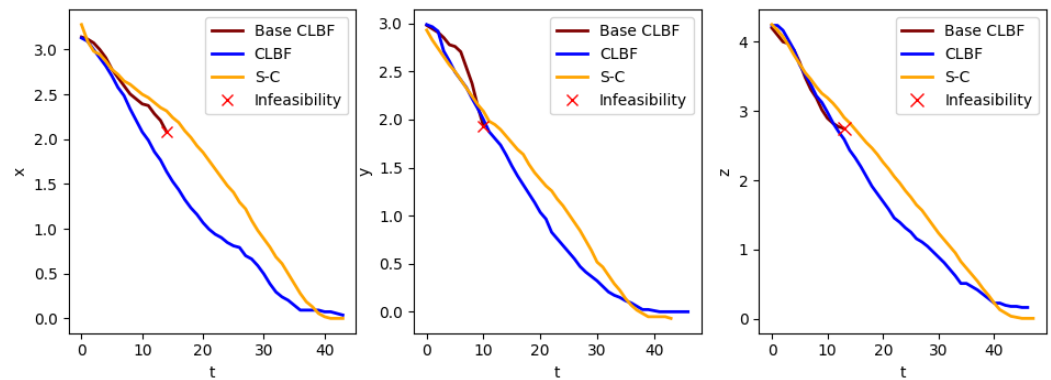


Figure 3. Trajectory comparison between base CLBF, HOCLBF, and successive convexification with a 60° glide slope constraint. The red 'x' marks solution infeasibility.

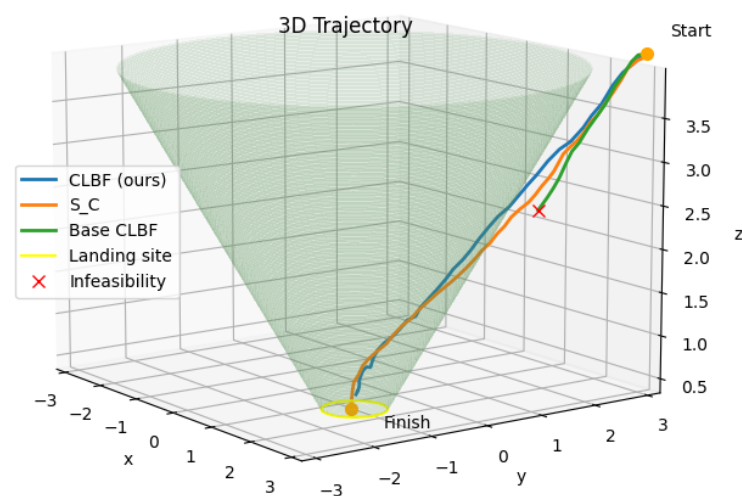


Figure 4. Three-dimensional trajectory comparison between base CLBF, HOCLBF, and successive convexification with a 60° glide slope constraint.

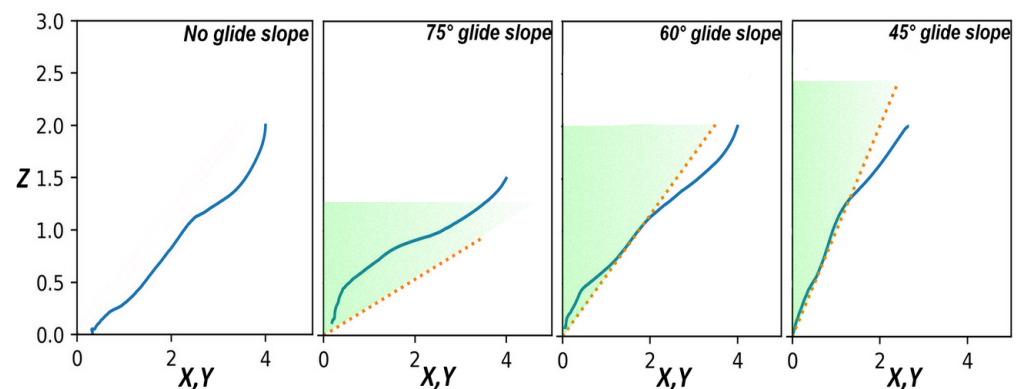


Figure 5. Trajectories for different glide slope angles in blue; The orange dotted line with the green region illustrates the safe area within the prescribed cone.

4.1.6. Limitations and Future Work

In this work, we attempt to generalize the use of high-order CLFs and CBFs to a wider set of problems and demonstrate the potential of HOCLBFs in safely controlling complex high-dimensional problems. In this work, we overcome the infeasibility issues that arise from the coupling of CLFs and CBFs by using heuristic parameters and we obtain successful results comparable to the robust successive convexification algorithm. Nonetheless, this approach still requires adequate analytical design for the Lyapunov and

barrier functions, and some parameters still need initial tuning and are prone to diverge in the presence of high disturbances. Therefore, our next goal is to incorporate some machine learning algorithms alongside differentiable optimization to enhance the performance of the HOCLBF and make it robust to parameter tuning and environmental disturbances.

5. Conclusions

In this paper, we generalize control Lyapunov and barrier functions to general non-linear dynamic systems with convex input under some conditions and suggest a new formulation of the higher-order control Lyapunov function that guarantees convergence for higher-order dynamics. We unite control Lyapunov and barrier functions using a heuristic approach and an online PID planner ensuring the feasibility of the Quadratic Program. We showcase the efficacy of our HOCLBF approach by devising a real-time six-degree-of-freedom powered descent controller as a convex Quadratic Program. Simulation results for different maneuvers under different safety settings were presented to demonstrate the effectiveness of the HOCLBF in controlling a highly nonlinear coupled system while maintaining safety. In this work, we overcome the infeasibility issues that arise from the coupling of CLFs and CBFs by using heuristic parameters, and we obtain successful results comparable to the robust successive convexification algorithm. Nonetheless, this approach still requires an adequate analytical design for the Lyapunov and barrier functions; some parameters still need initial tuning and are prone to diverge in the presence of high disturbances. Therefore, The next phase of this research will focus on leveraging differentiable convex optimization and reinforcement learning approaches to achieve automatic parameter tuning and intelligent safety constraints. We are also looking into distributional robust optimization to make the HOCLBF robust to model uncertainty and environmental disturbances.

Author Contributions: Conceptualization, methodology, writing, A.E.C. and C.S.; formal analysis, software, validation, visualization, writing—original draft preparation, A.E.C.; supervision, project administration, writing—review and editing, C.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All data can be simulated and reproduced from our dynamics models and control algorithms.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Allgower, F.; Findeisen, R.; Nagy, Z.K. Nonlinear model predictive control: From theory to application. *J.-Chin. Inst. Chem. Eng.* **2004**, *35*, 299–316.
2. Findeisen, R.; Allgöwer, F. An introduction to nonlinear model predictive control. In Proceedings of the 21st Benelux Meeting on Systems and Control, Veldhoven, The Netherlands, 19–21 March 2002 ; Volume 11, pp. 119–141.
3. Åström, K.J. Adaptive control. In *Mathematical System Theory: The Influence of RE Kalman*; Springer: Berlin/Heidelberg, Germany, 1995; pp. 437–450.
4. Tao, G. *Adaptive Control Design and Analysis*; John Wiley & Sons: Hoboken, NJ, USA, 2003; Volume 37.
5. Acikmese, B.; Ploen, S.R. Convex programming approach to powered descent guidance for mars landing. *J. Guid. Control. Dyn.* **2007**, *30*, 1353–1366. [[CrossRef](#)]
6. Mao, Y.; Dueri, D.; Szmuk, M.; Açikmeşe, B. Successive convexification of non-convex optimal control problems with state constraints. *IFAC-PapersOnLine* **2017**, *50*, 4063–4069. [[CrossRef](#)]
7. Szmuk, M.; Reynolds, T.P.; Açikmeşe, B. Successive convexification for real-time six-degree-of-freedom powered descent guidance with state-triggered constraints. *J. Guid. Control Dyn.* **2020**, *43*, 1399–1413. [[CrossRef](#)]
8. Artstein, Z. Stabilization with relaxed controls. *Nonlinear Anal. Theory Methods Appl.* **1983**, *7*, 1163–1173. [[CrossRef](#)]
9. Sontag, E.D. A ‘universal construction of Artstein’s theorem on nonlinear stabilization. *Syst. Control Lett.* **1989**, *13*, 117–123. [[CrossRef](#)]
10. Sontag, E.D. Control-lyapunov functions. In *Open problems in Mathematical Systems and Control Theory*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 211–216.

11. Ames, A.D.; Grizzle, J.W.; Tabuada, P. Control barrier function based quadratic programs with application to adaptive cruise control. In Proceedings of the 53rd IEEE Conference on Decision and Control, Los Angeles, CA, USA, 15–17 December 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 6271–6278.
12. Ames, A.D.; Xu, X.; Grizzle, J.W.; Tabuada, P. Control barrier function based quadratic programs for safety critical systems. *IEEE Trans. Autom. Control* **2016**, *62*, 3861–3876. [[CrossRef](#)]
13. Butz, A. Higher order derivatives of Lyapunov functions. *IEEE Trans. Autom. Control* **1969**, *14*, 111–112. [[CrossRef](#)]
14. Ahmadi, A.A.; Parrilo, P.A. On higher order derivatives of Lyapunov functions. In Proceedings of the 2011 American Control Conference, San Francisco, CA, USA, 29 June–1 July 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1313–1314.
15. Xiao, W.; Belta, C. Control barrier functions for systems with high relative degree. In Proceedings of the 2019 IEEE 58th Conference on Decision and Control (CDC), Nice, France, 11–13 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 474–479.
16. Xiao, W.; Belta, C. High-Order Control Barrier Functions. *IEEE Trans. Autom. Control* **2022**, *67*, 3655–3662. [[CrossRef](#)]
17. Mehra, A.; Ma, W.L.; Berg, F.; Tabuada, P.; Grizzle, J.W.; Ames, A.D. Adaptive cruise control: Experimental validation of advanced controllers on scale-model cars. In Proceedings of the 2015 American Control Conference (ACC), Chicago, IL, USA, 1–3 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1411–1418.
18. Ames, A.D.; Powell, M. Towards the unification of locomotion and manipulation through control lyapunov functions and quadratic programs. In *Control of Cyber-Physical Systems*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 219–240.
19. Khalil, H. *Nonlinear Systems*; Prentice Hall: Hoboken, NJ, USA, 2002.
20. Blanchini, F. Set invariance in control. *Automatica* **1999**, *35*, 1747–1767. [[CrossRef](#)]
21. Romdlony, M.Z.; Jayawardhana, B. Uniting control Lyapunov and control barrier functions. In Proceedings of the 53rd IEEE Conference on Decision and Control, Los Angeles, CA, USA, 15–17 December 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 2293–2298.
22. Lee, U.; Mesbahi, M. Optimal power descent guidance with 6-DoF line of sight constraints via unit dual quaternions. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, Kissimmee, FL, USA, 5–9 January 2015; p. 0319.
23. Lee, U.; Mesbahi, M. Dual quaternions, rigid body mechanics, and powered-descent guidance. In Proceedings of the 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), Maui, HI, USA, 10–13 December 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 3386–3391.
24. Domahidi, A.; Chu, E.; Boyd, S. ECOS: An SOCP solver for embedded systems. In Proceedings of the 2013 European Control Conference (ECC), Zurich, Switzerland, 17–19 July 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 3071–3076.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.