ME 6402 – Lecture 24

CONTROL BARRIER FUNCTIONS

April 8 2025

Overview:

- Extend barrier functions to Control Barrier Functions

Additional Reading:

- A. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control Barrier Functions: Theory and Applications," IEEE Transactions on Automatic Control, 2019.

Control Barrier Functions

Consider a control-affine system

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

and a given set $\mathcal{C} = \{x \text{ s.t. } h(x) \geq 0\}$. How can we choose a controller $u(x)$ such that $\mathcal{C}$ is positively invariant?

Recall Barrier Functions:

$$\dot{h}(x) = \nabla h(x)^T f(x) \geq -\alpha(h(x)) \quad \text{for all } x \in \mathbb{R}^n \tag{2}$$

Recall that we also have the following theorem for barrier functions:
Theorem: Barrier Function. *If h is a barrier function, then $\mathcal{C} = \{x : h(x) \geq 0\}$ is positively invariant.*

Definition: *Control Barrier Function.* A function h with $\mathcal{C} = \{x \text{ s.t. } h(x) \geq 0\}$ is a *control barrier function (CBF)* for a control-affine system $\dot{x} = f(x) + g(x)u$ if there exists a locally Lipschitz function $\alpha : \mathbb{R} \to \mathbb{R}$ satisfying $\alpha(0) = 0$ such that

$$\sup_{u \in \mathbb{R}^m} \nabla h(x)^T (f(x) + g(x)u) \geq -\alpha(h(x)) \quad \text{for all } x \in \mathbb{R}^n. \tag{3}$$

The supremum is the smallest number that is greater than or equal to every element in the set. The supremum must be a real number (cannot be infinity).

We can also write (3) using Lie derivative notation:

$$\sup_{u \in \mathbb{R}^m} L_f h(x) + L_g h(x)u \geq -\alpha(h(x)) \tag{4}$$

Define

$$U(x) = \{u \in \mathbb{R}^m \text{ s.t. } \nabla h(x)^T (f(x) + g(x)u) \geq -\alpha(h(x))\}. \tag{5}$$

Theorem: Invariance from CBF. *If h is a control barrier function for (1), then the following hold:*

1. *$U(x) \neq \emptyset$ for all x;*
2. *Any Lipschitz feedback control $u : \mathbb{R}^n \to \mathbb{R}^m$ satisfying $u(x) \in U(x)$ renders $\mathcal{C}$ invariant;*
3. *A feedback control is given by*

$$u^*(x) = \begin{cases} 0 \text{ if } \nabla h(x)^T f(x) + \alpha(h(x)) \geq 0 \\ \frac{-\nabla h(x)^T f(x) - \alpha(h(x))}{\|\nabla h(x)^T g(x)\|_2^2} (g(x)^T \nabla h(x)) \\ \qquad \text{otherwise.} \end{cases} \tag{6}$$

this is the same thing as writing:

$$u^*(x) = \begin{cases} 0 & \text{if } L_f h(x) + \alpha(h(x)) \geq 0 \\ \frac{-(L_f h(x) + \alpha(h(x))) L_g h(x)^T}{L_g h(x) L_g h(x)^T} & \text{otherwise} \end{cases} \tag{7}$$

A sufficient condition for $u^*(x)$ to be Lipschitz on some domain is that $\nabla h(x)^T g(x) \neq 0$ everywhere on the domain.

Proof. The proof of all three parts is as follows:

1. If $\sup_{u \in \mathbb{R}^m} \nabla h(x)^T (f(x) + g(x)u) < \infty$, then the sup is attained for some u.
2. h becomes a (regular) barrier function for $\tilde{f}(x) = f(x) + g(x)u(x)$ and theorem from previous lecture applies.
3. (Sketch) First, note that $u^*(x)$ is well-defined since $\nabla h(x)^T g(x) \neq 0$ whenever $h(x)^T f(x) + \alpha(h(x)) < 0$ by CBF condition. $u^*(x)$ can be considered as a composition of 3 Lipschitz functions and is therefore Lipschitz. Finally, we can verify that

$$\nabla h(x)^T (f(x) + g(x)u^*(x)) + \alpha(h(x)) \tag{8}$$

$$= \begin{cases} \nabla h(x)^T f(x) + \alpha(h(x)) & \text{if } \nabla h(x)^T f(x) + \alpha(h(x)) \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

$$\geq 0. \tag{10}$$

Minimum Effort Control

From the above proof, specifically, the condition

$$\nabla h(x)^T (f(x) + g(x)u^*(x)) + \alpha(h(x)) \tag{11}$$

$$= \begin{cases} \nabla h(x)^T f(x) + \alpha(h(x)) & \text{if } \nabla h(x)^T f(x) + \alpha(h(x)) \geq 0 \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

we conclude that $u^*(x)$ is the "minimum effort" controller, *i.e.*, $u^*(x) = \text{argmin}_{u \in U(x)} \|u\|_2^2$.

Example (Cart-Pole System): :

Recall the model of the cart-pole system from Lecture 16 (take $m = M = \ell = 1$):

$$\begin{aligned} \ddot{y} = \dot{v} &= \frac{1}{1+\sin^2\theta}\Big(u + \dot{\theta}^2 \sin\theta - g \sin\theta\cos\theta\Big) \\ \ddot{\theta} &= \frac{1}{1+\sin^2\theta}\Big(-u\cos\theta - \dot{\theta}^2\cos\theta\sin\theta + 2g\sin\theta\Big) \end{aligned} \tag{13}$$

where $v = \dot{y}$ is velocity. Take as the state $x = [y\ v\ \theta\ \dot{\theta}]^T$. Suppose we want v to satisfy

$$-L \le v \le L.$$

Choose

$$h(x) = \frac{1}{2}(-v^2 + L^2) \tag{14}$$

$$\alpha(s) = \gamma s, \quad \gamma > 0. \tag{15}$$

Then

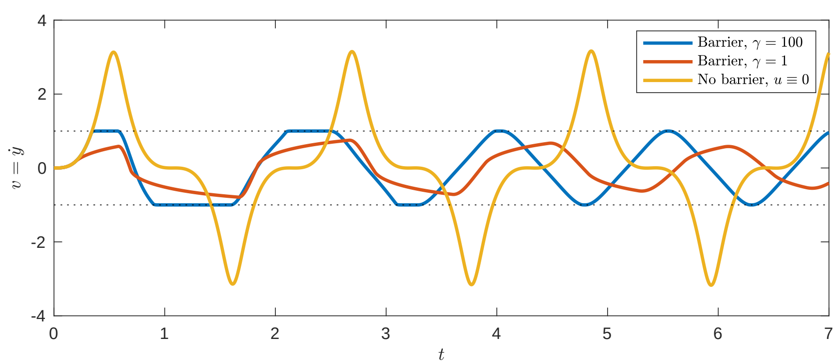
$$\nabla h(x)^T f(x) = L_f h(x) = \frac{-v}{1+\sin^2(\theta)}\Big(\dot{\theta}^2\sin\theta - g\sin\theta\cos\theta\Big) \tag{16}$$

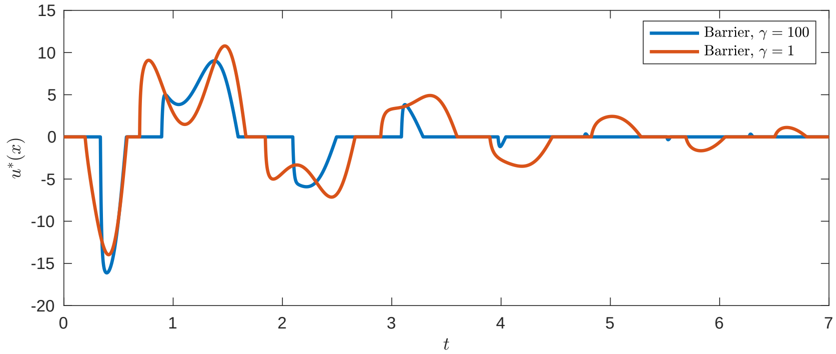
$$\nabla h(x)^T g(x) = L_g h(x) = \frac{-v}{1+\sin^2(\theta)} \tag{17}$$

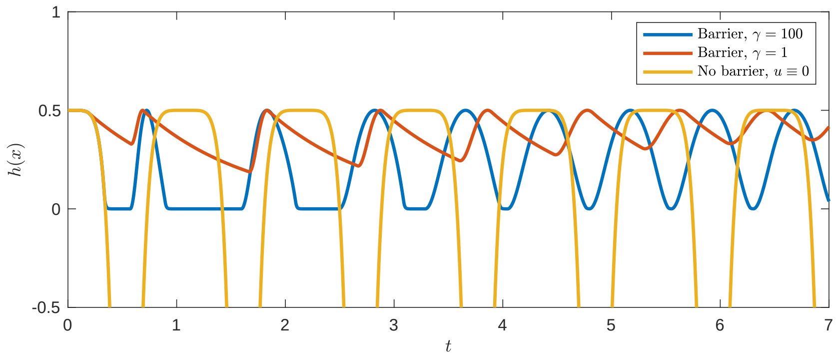
$$\alpha(h(x)) = \gamma h(x) \tag{18}$$

and $u^*(x)$ constructed as above.

The figures below show results for $x_0 = [y_0\ v_0\ \theta_0\ \dot{\theta}_0]^T = [0\ 0\ \pi/2\ 0]^T$ using the $u^*(x)$ from the theorem.







Controller Synthesis as Optimization Problem

For fixed x, the CBF constraint is *affine* in u! Then we can define a *convex* program to compute a control input at each time instant:

$$\begin{aligned} u(x) = &\arg\min_{\mu} && C(\mu, x) \\ &\text{subject to} && \nabla h(x)^T f(x) + \nabla h(x)^T g(x)\mu \geq -\alpha(h(x)) \end{aligned} \tag{19}$$

where $C(\mu, x)$ is some cost function that is convex in μ for each fixed state x.

Example 1: Suppose $k(x) : \mathbb{R}^n \to \mathbb{R}^m$ is some nominal feedback controller designed for some other purpose (e.g., performance objectives). Can choose $C(\mu, x) = \|\mu - k(x)\|_2^2$. The result is a quadratic program (with affine constraints) to compute $u(x)$ at each x.

- Raises questions about solving a QP in real-time online, care must be taken with discretization values, etc.
- Convex solvers are fast enough that they can be included "in-the-loop" and have been for applications like stable bipedal locomotion, quadrotor control

Example 2: Consider our controlled pendulum:

$$\dot{x} = \frac{d}{dt}\begin{bmatrix}\theta \\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}\dot{\theta} \\ -\frac{g}{l}\sin(\theta) + u\end{bmatrix} \quad \Longrightarrow \quad \dot{x} = \begin{bmatrix}x_2 \\ -\frac{g}{l}\sin(x_1) + u\end{bmatrix}$$

Let's assume that we want to limit the velocity of the pendulum, we can define our safe set as:

$$\mathcal{C} = \{x \text{ s.t. } h(x) := v_{\text{max}}^2 - x_2^2 \geq 0\}$$

Taking the derivative of $h(x)$ we have:

$$\begin{aligned}\dot{h} &= -2x_2\dot{x_2} = -2x_2(-\frac{g}{l}\sin(x_1) + u) \\ &= \underbrace{2x_2\frac{g}{l}\sin(x_1)}_{L_f h}\underbrace{-2x_2}_{L_g h}u\end{aligned}$$

Thus, safety can be enforced via the CBF condition:

$$L_f h + L_g h u \geq -\alpha(h)$$

where $\alpha(h) = \gamma h$ for some $\gamma > 0$. This can be enforced along with a tracking controller on the pendulum via the aforementioned QP formulation[1]:

[1] An implementation of this example is provided online.

$$\begin{aligned}u^* = \underset{u}{\text{minimize}} \; & \|u - u_{des}\|_2^2 \\ \text{subject to } & L_f h + L_g h u + \gamma h \geq 0\end{aligned}$$

Note: For CBFs to be implemented in this way, we need to ensure that $\nabla h(x)^T g(x) = L_g h(x) \neq 0$ for all x in the domain of interest. However, this would preclude us from selecting a CBF to limit the pendulum *position*, i.e., $h(x) := \theta_{\text{max}}^2 - x_1^2$. In this case, we need *higher-order* CBFs. We will cover these in the next lecture.