Safe Control Synthesis via Input Constrained Control Barrier Functions

Devansh Agrawal & Dimitra Panagou
University of Michigan

- Background and Problem Statement
- Motivating Idea
- Formal Construction
- Simulation Results

- Background and Problem Statement
- Motivating Idea (ex. Adapative Cruise Control)
- Formal Construction
- Simulation Results (ex. Autonomous Rendezvous)

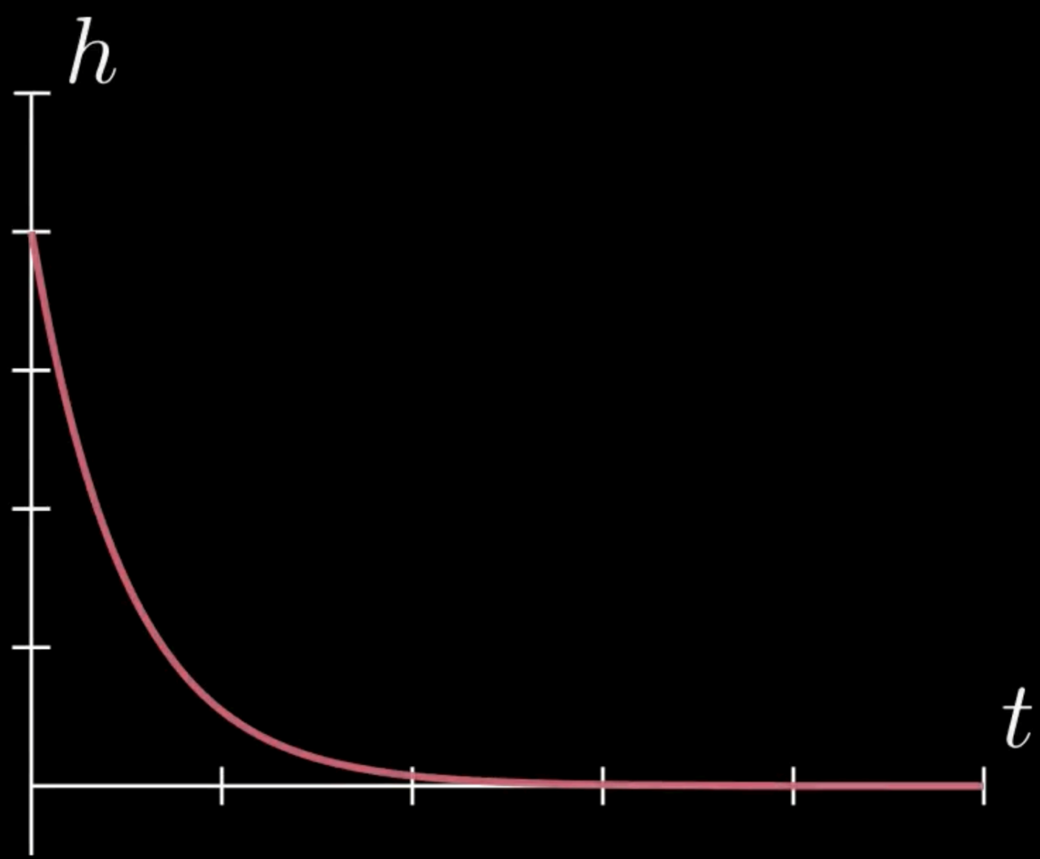
Safety Critical Controls

$$\dot{x} = f(x) + g(x)u \qquad \begin{matrix} (x \in \mathcal{X} \subset \mathbb{R}^n) \\ (u \in \mathcal{U} \subset \mathbb{R}^m) \end{matrix}$$

$$\mathcal{S} = \{x : h(x) \geq 0\} \subset \mathcal{X}$$

Which set of control inputs u ensures system stays safe?

Background: Control Barrier Functions

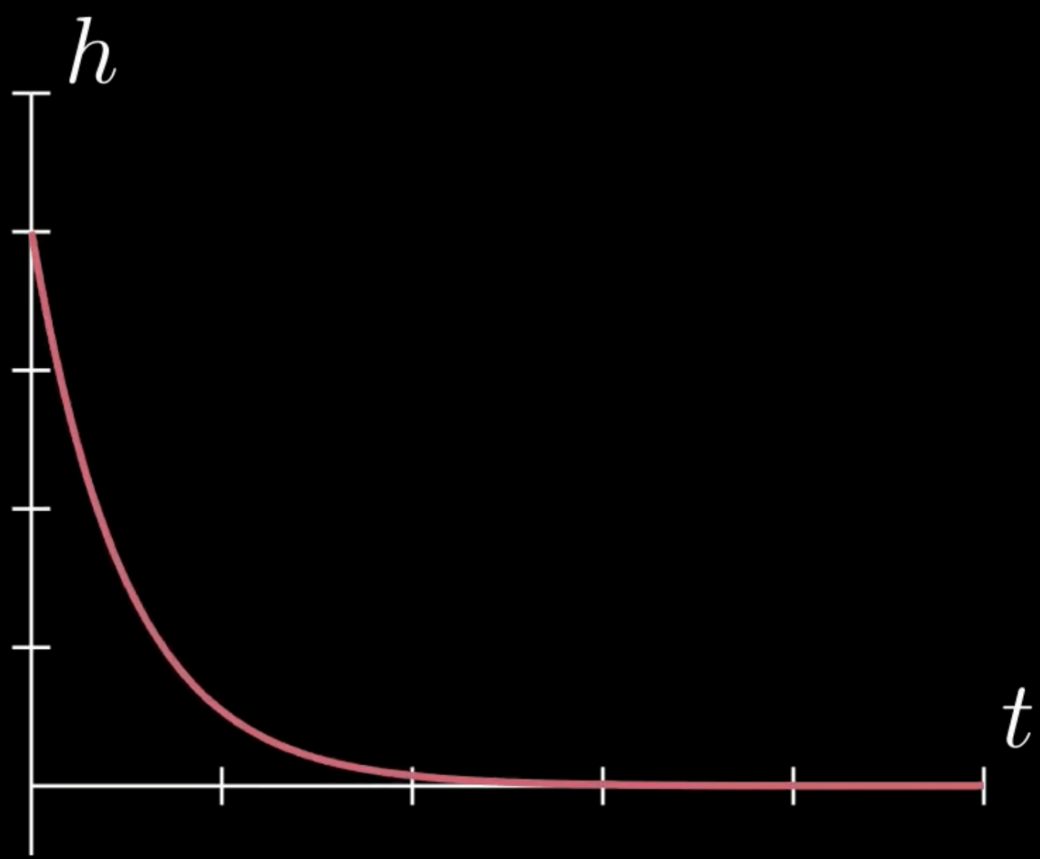


$$\alpha(r) = 1.00r$$

Let $\alpha \in \mathcal{K}$.
If $\forall x \in \mathcal{S}$, $\exists u \in \mathcal{U}$ such that
$$\dot{h}(x, u) \geq -\alpha(h(x))$$
then h is a CBF for $\mathcal{S}$

[Ames, 2019]

Background: Control Barrier Functions



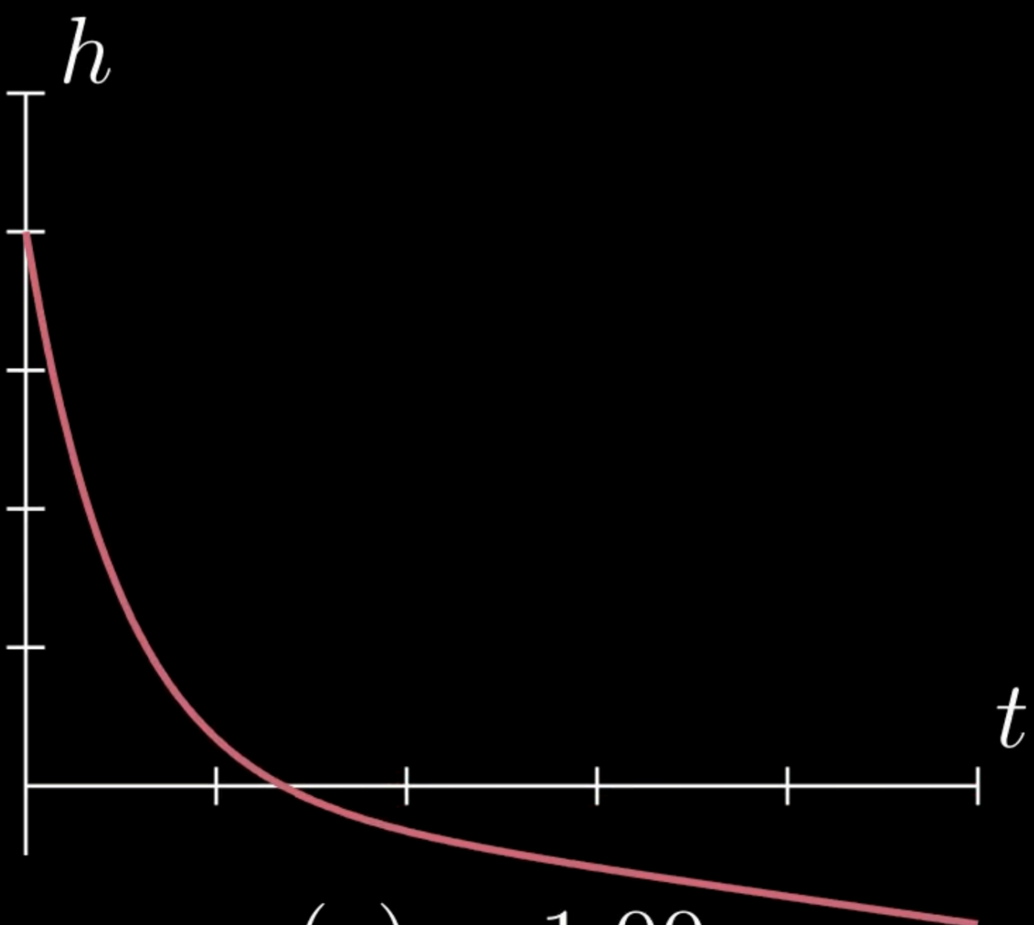
$\alpha(r) = 1.00r$

Let $\alpha \in \mathcal{K}$.
If $\forall x \in \mathcal{S}$, $\exists u \in \mathcal{U}$ such that
$\dot{h}(x, u) \geq -\alpha(h(x))$
then h is a CBF for $\mathcal{S}$

If h is a CBF for $\mathcal{S}$,
any Lips. controller $\pi : \mathcal{S} \to \mathcal{U}$
where $\dot{h}(x, \pi(x)) \geq -\alpha(h(x))$
ensures $\mathcal{S}$ is forward invariant

[Ames, 2019]

Background: Control Barrier Functions



$\alpha(r) = 1.00r$

$u \in \mathcal{U}$

Let $\alpha \in \mathcal{K}$.
If $\forall x \in \mathcal{S}, \exists u \in \mathcal{U}$ such that
$\dot{h}(x, u) \geq -\alpha(h(x))$
then h is a CBF for $\mathcal{S}$

If h is a CBF for $\mathcal{S}$,
any Lips. controller $\pi : \mathcal{S} \to \mathcal{U}$
where $\dot{h}(x, \pi(x)) \geq -\alpha(h(x))$
ensures $\mathcal{S}$ is forward invariant

[Ames, 2019]

Background: Problem Statement

Given:

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

$$u \in \mathcal{U}$$

$$\mathcal{S} = \{x : h(x) \geq 0\}$$

[Cortez 2020]

[Wu 2015]

[Breeden 2021]

Construct:

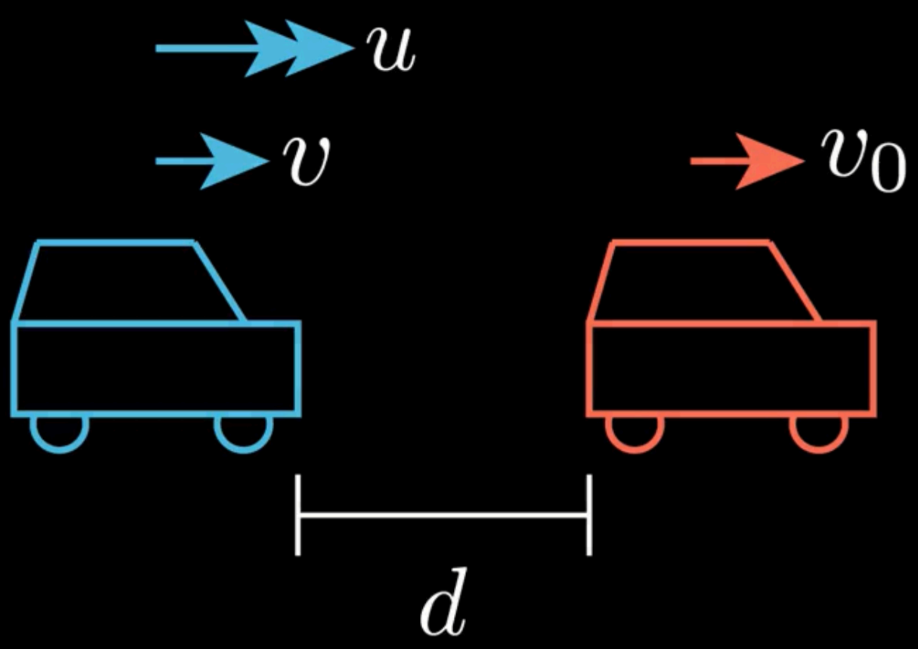
Inner Safe Set: $\mathcal{C}^* \subset \mathcal{S}$

Safe Control Inputs: $K_{ICCBF}(x)$

Such that,

Forward invariance of $\mathcal{C}^*$ is guaranteed for any Lips. cont. controller $\pi(x) \in K_{ICCBF}(x)$

Example: Adaptive Cruise Control



$$\frac{d}{dt}\begin{bmatrix} d \\ v \end{bmatrix} = \begin{bmatrix} v_0 - v \\ -F(v)/m \end{bmatrix} + \begin{bmatrix} 0 \\ g_0 \end{bmatrix} u,$$

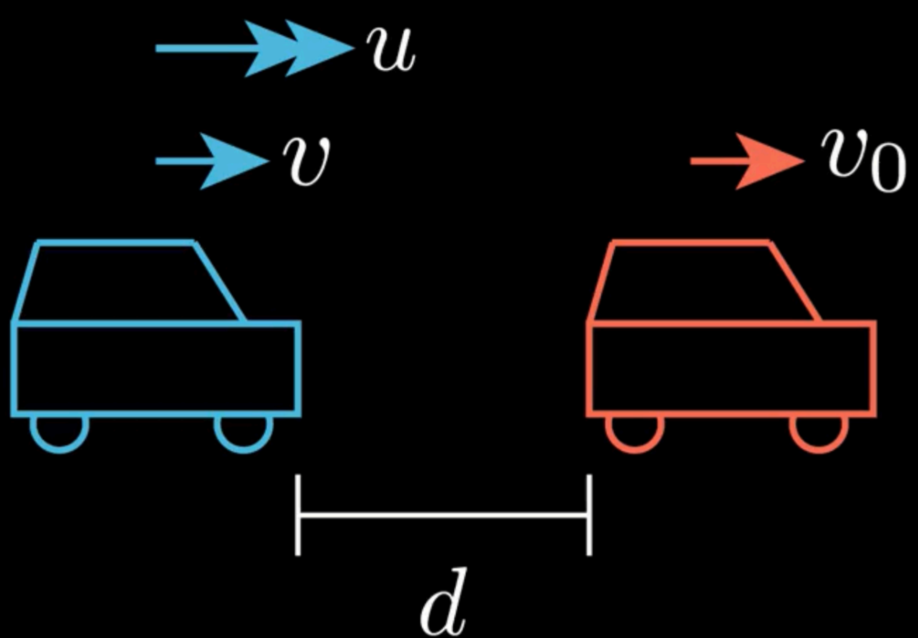
$$|u| \leq 0.25$$

$$\text{safe if } d \geq 1.8v$$

$$\therefore h(x) = d - 1.8v$$

[Ames, 2019]

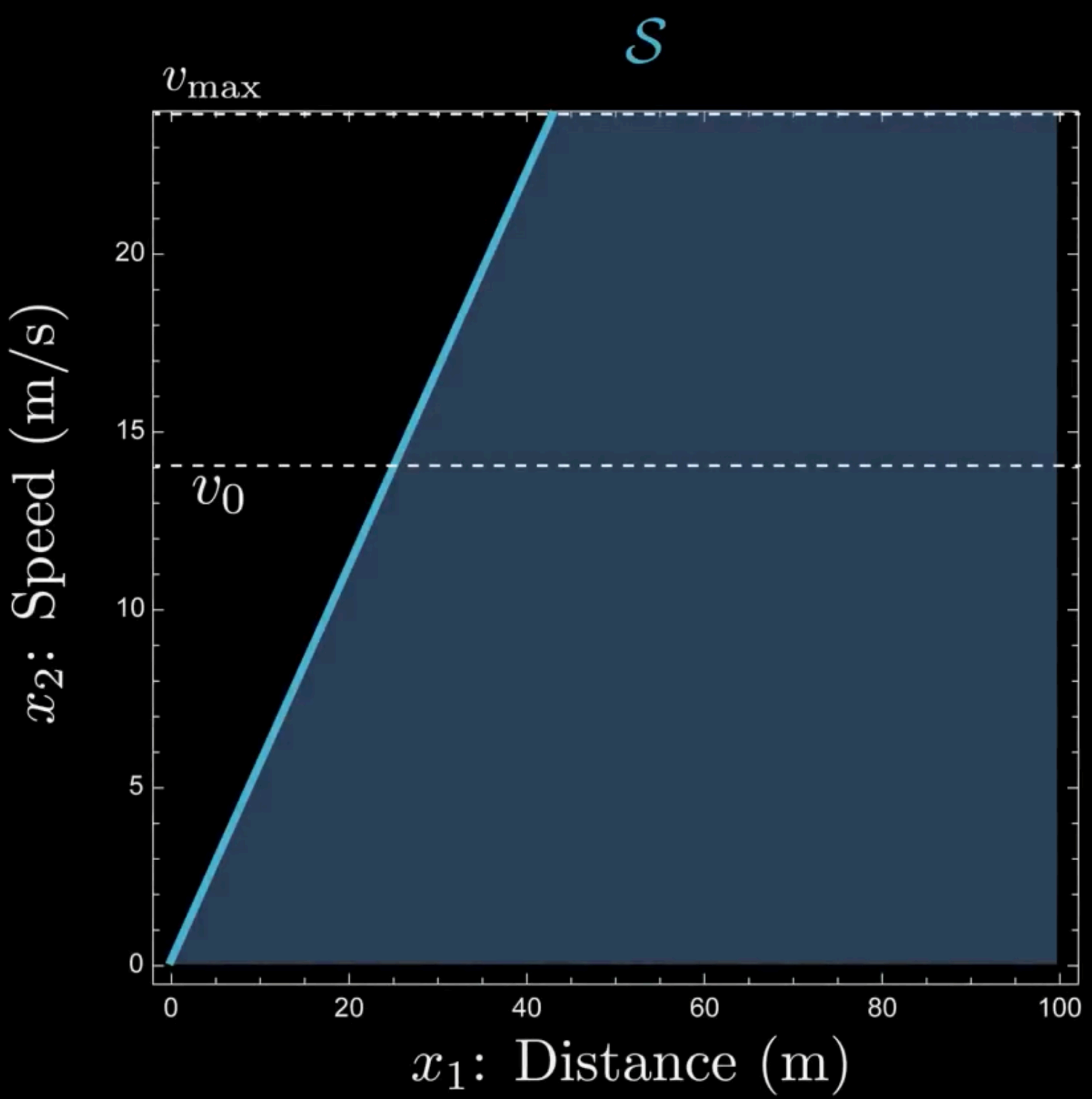
Example: Adaptive Cruise Control



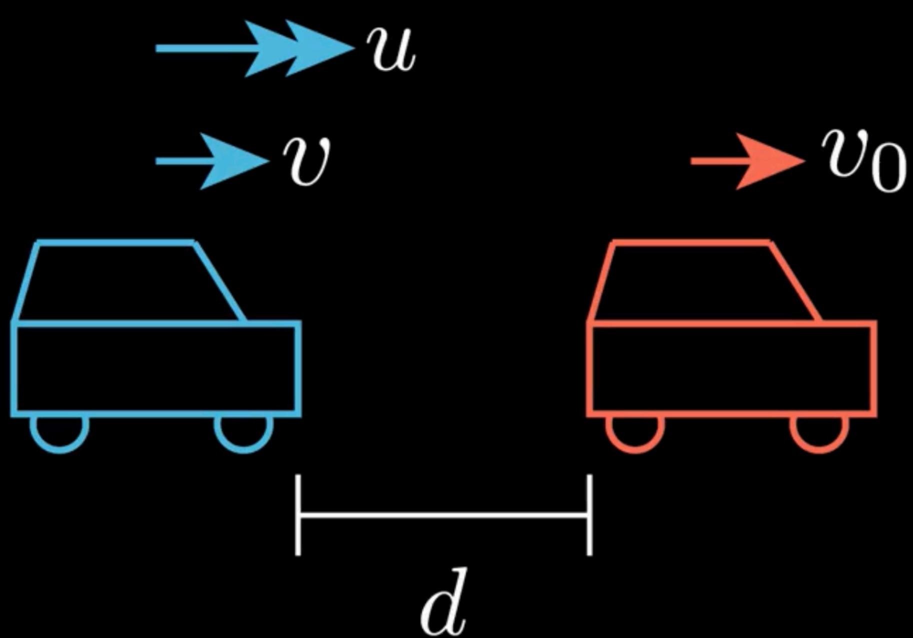
$$\dot{x} = f(x) + g(x)u$$

$$|u| \leq 0.25$$

$$h(x) = d - 1.8v$$



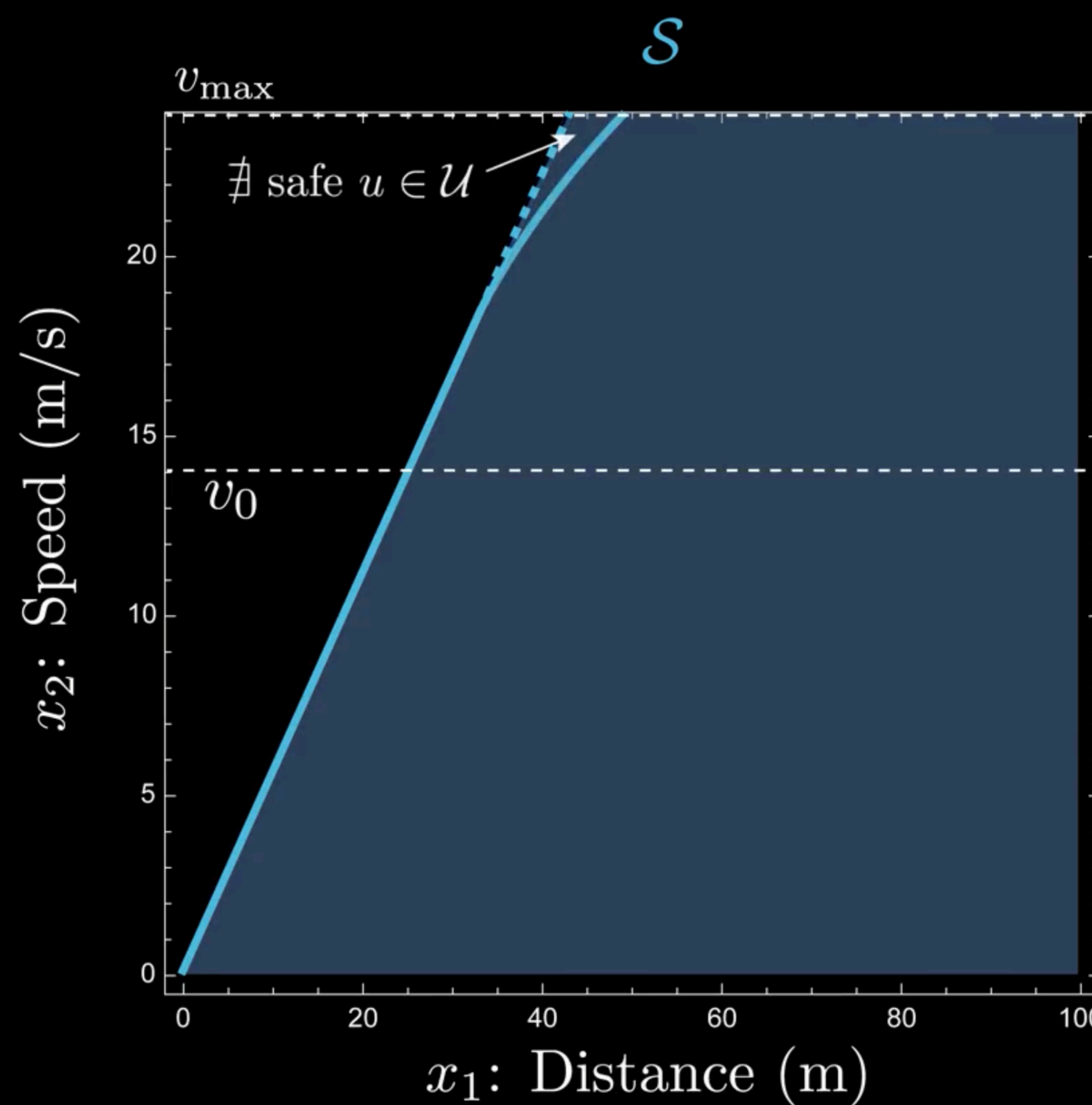
Example: Adaptive Cruise Control



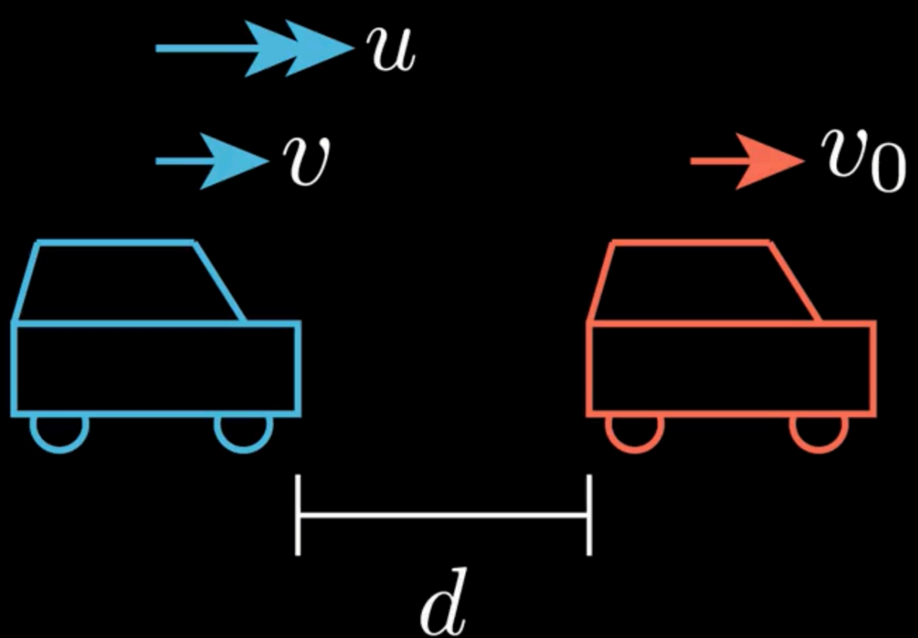
$$\dot{x} = f(x) + g(x)u$$

$$|u| \leq 0.25$$

$$h(x) = d - 1.8v$$



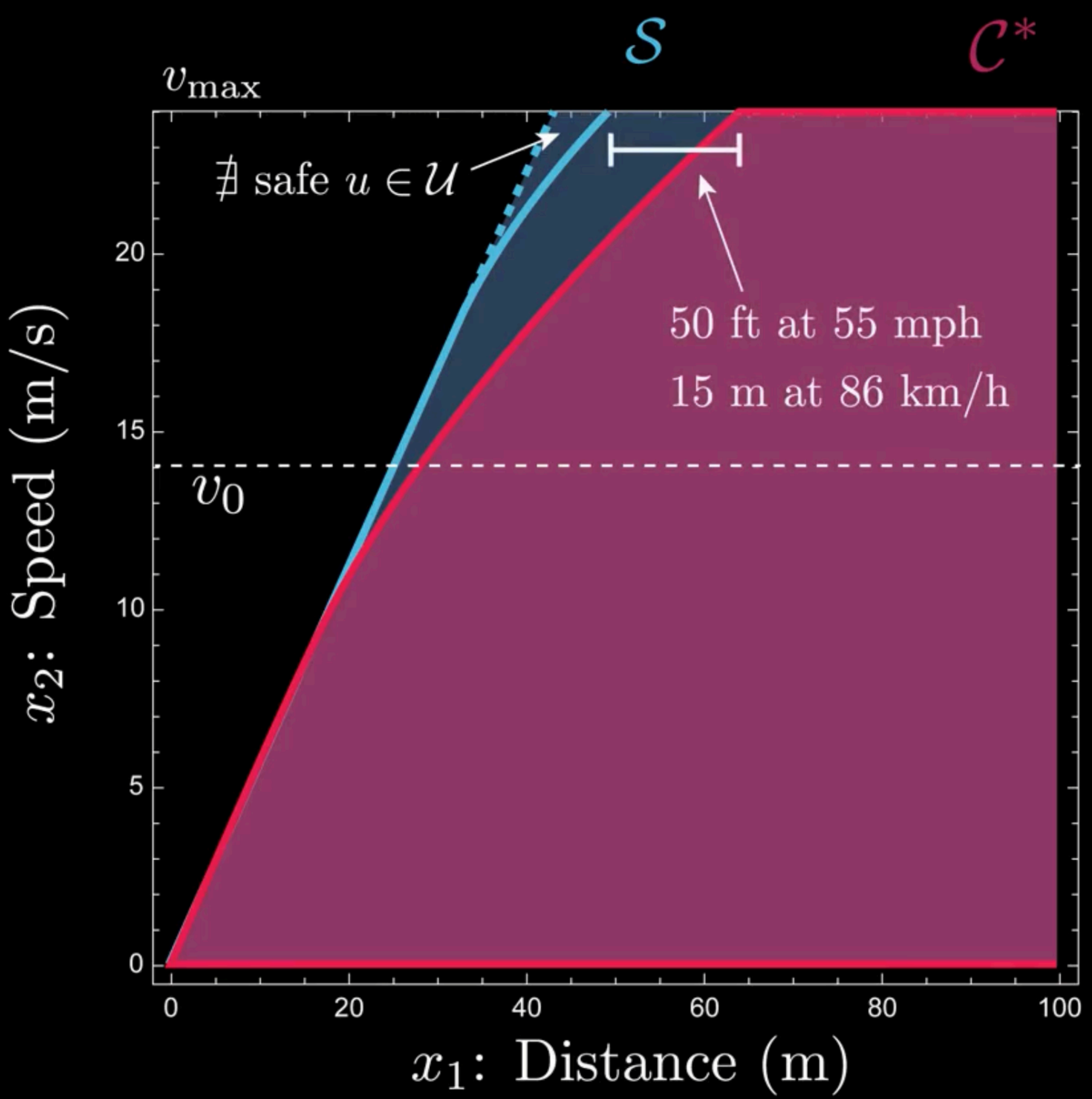
Example: Adaptive Cruise Control



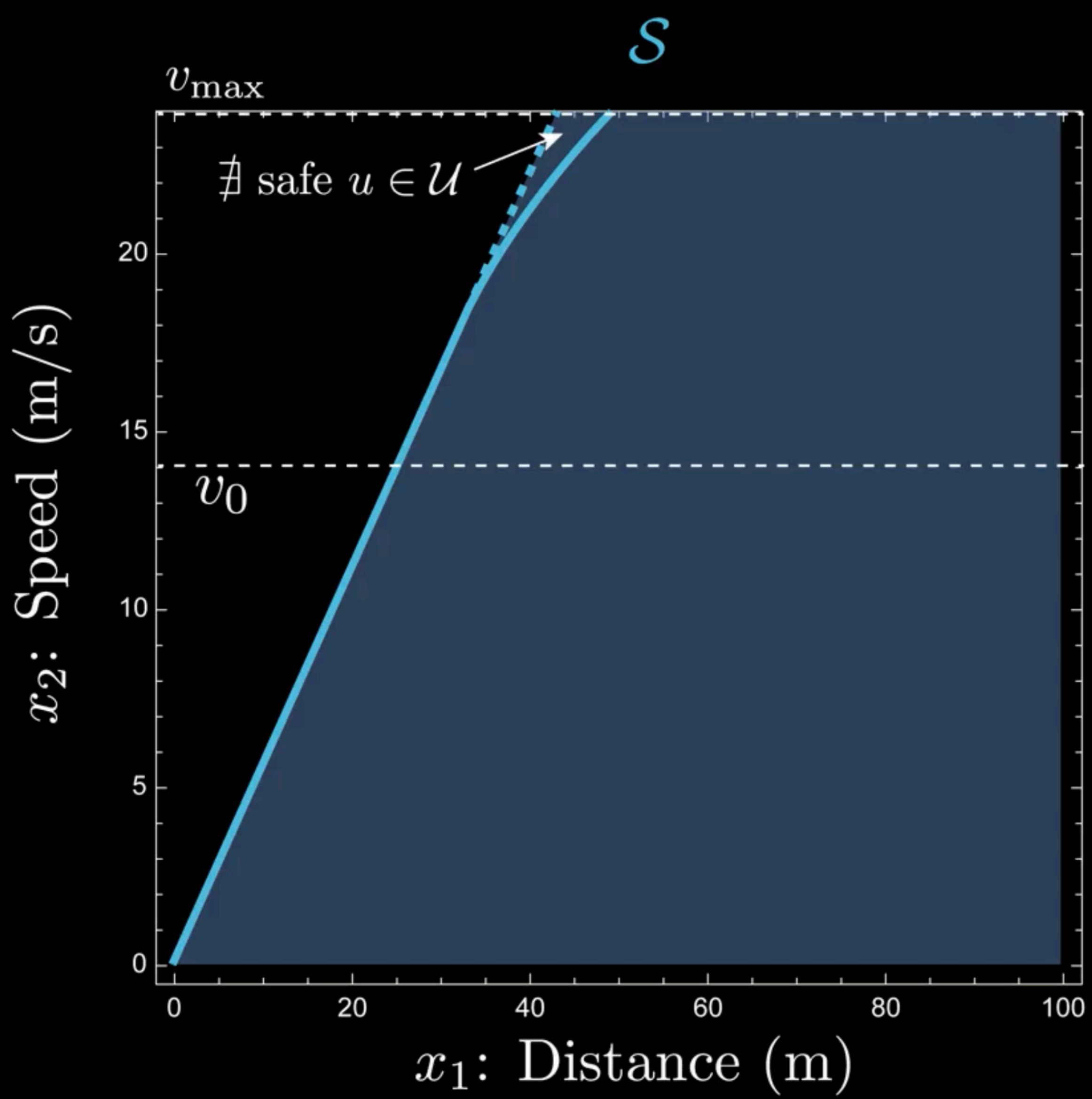
$$\dot{x} = f(x) + g(x)u$$

$$|u| \leq 0.25$$

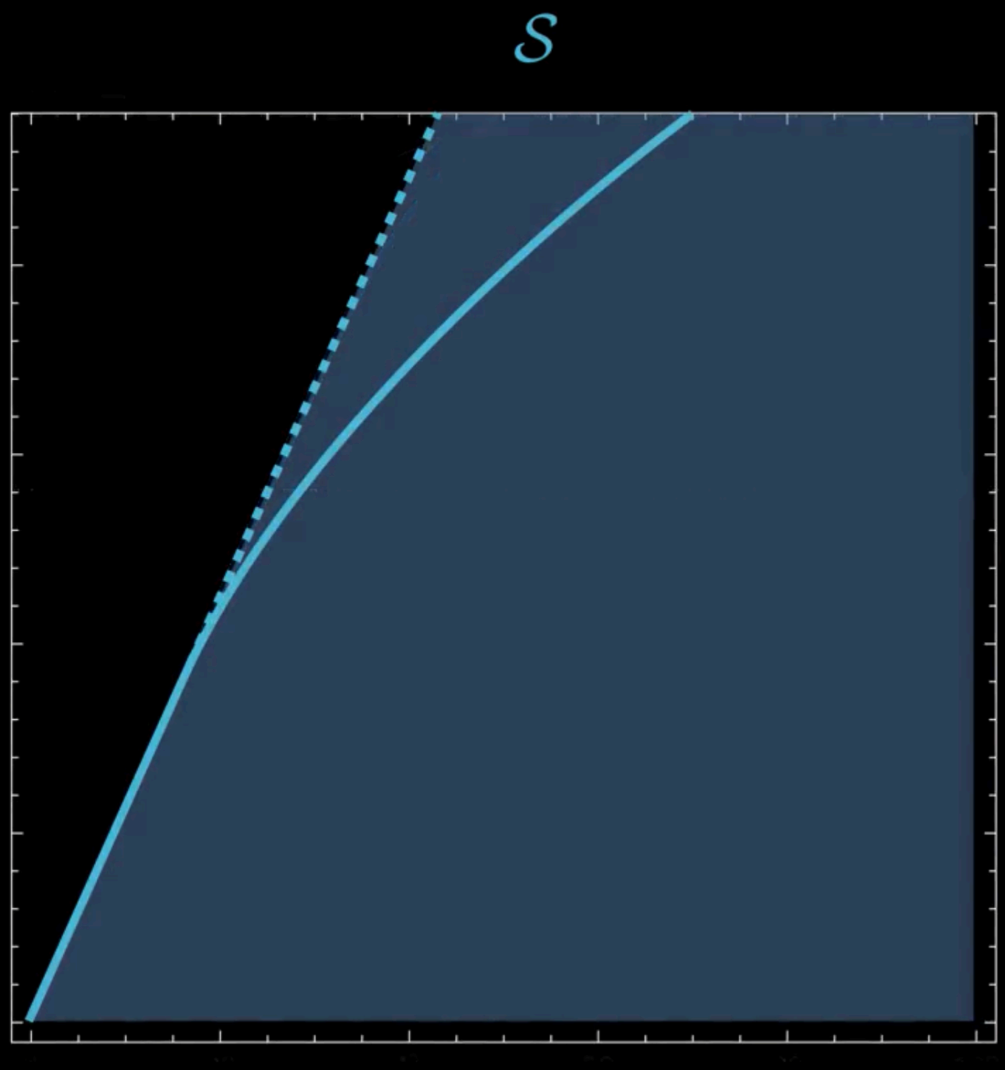
$$h(x) = d - 1.8v$$



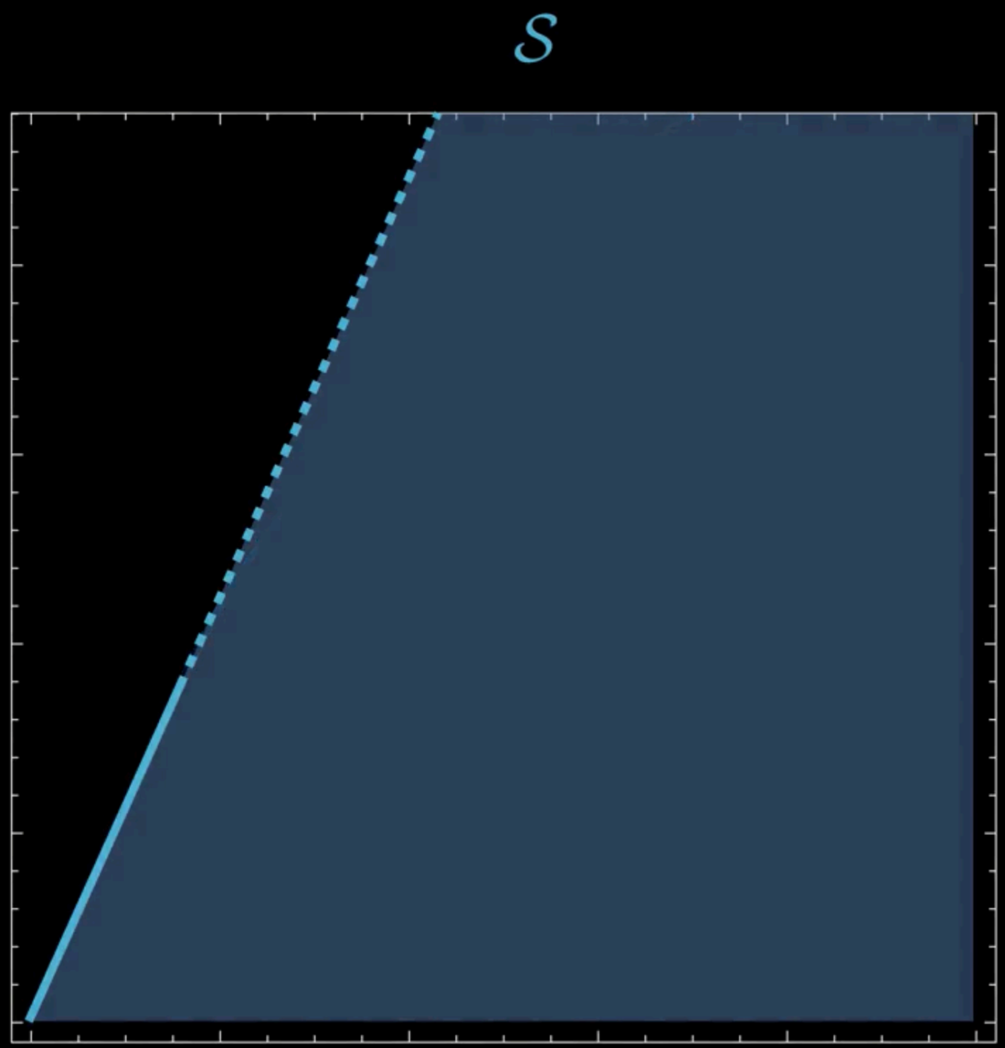
Motivating Idea



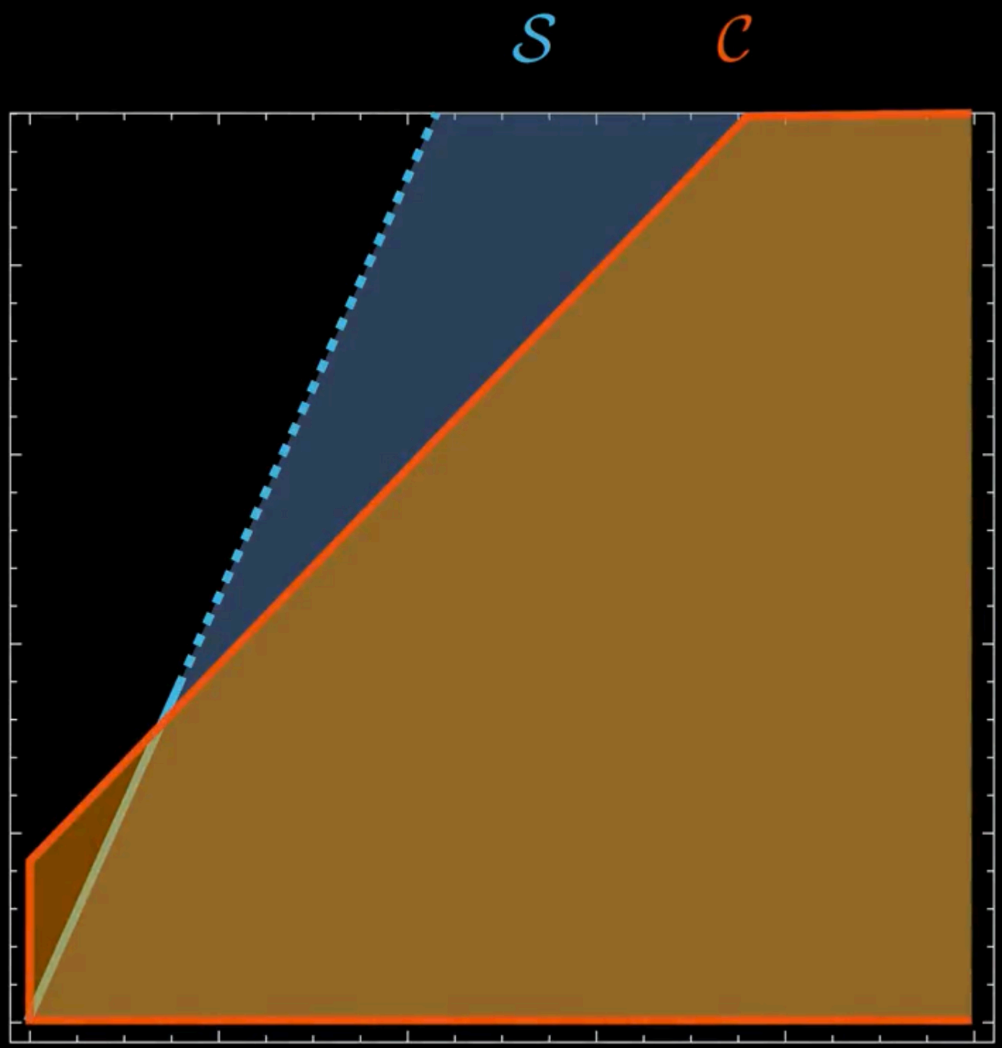
Motivating Idea



Motivating Idea



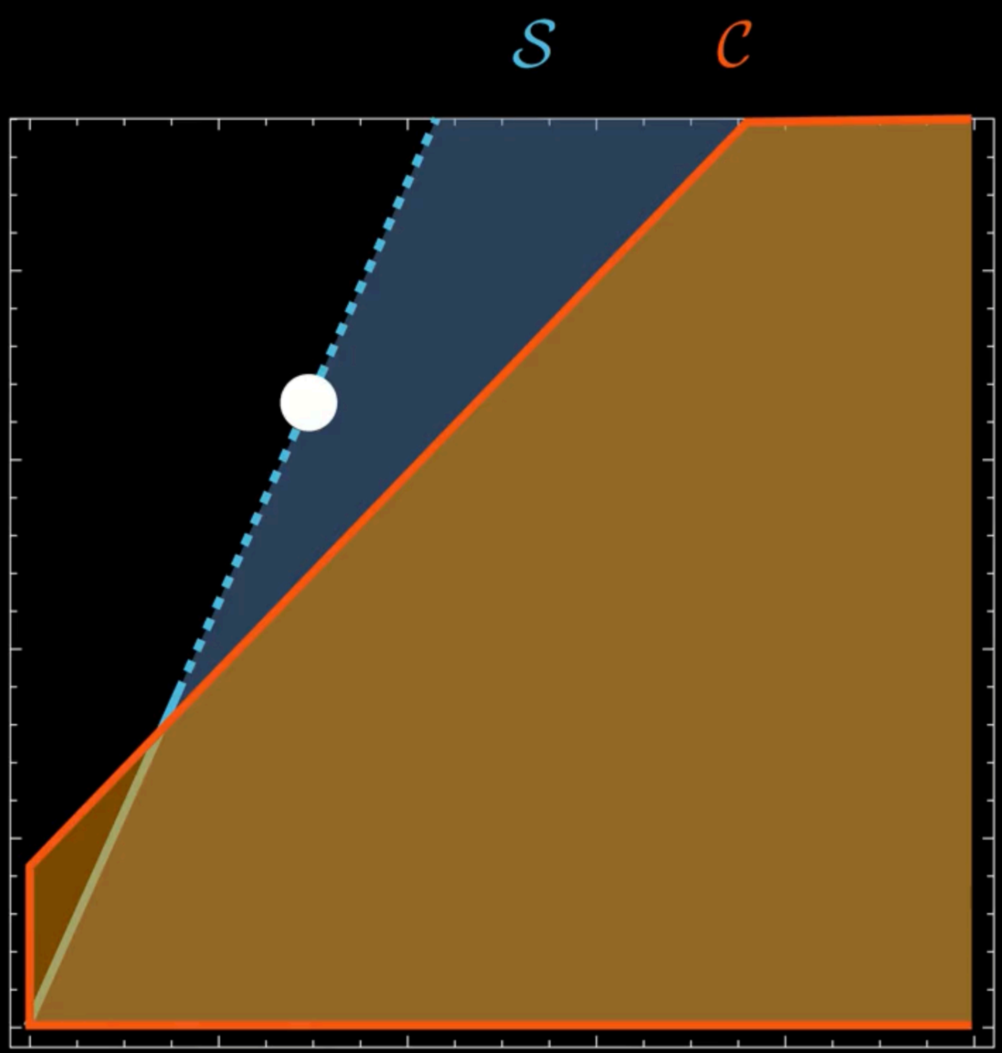
Motivating Idea



Consider $b(x) \triangleq \inf_{u \in \mathcal{U}} \dot{h}(x,u) + \alpha(h(x))$

$$\mathcal{C} = \{x : b(x) \geq 0\}$$

Motivating Idea



$$\text{Consider } b(x) \triangleq \inf_{u \in \mathcal{U}} \dot{h}(x,u) + \alpha(h(x))$$

$$\mathcal{C} = \{x : b(x) \geq 0\}$$

Useful Property 1: If $x \in \partial S$, and $\nexists$ safe u

$$h(x) = 0 \text{ and } \dot{h}(x,u) < 0 \ \forall u \in \mathcal{U}$$

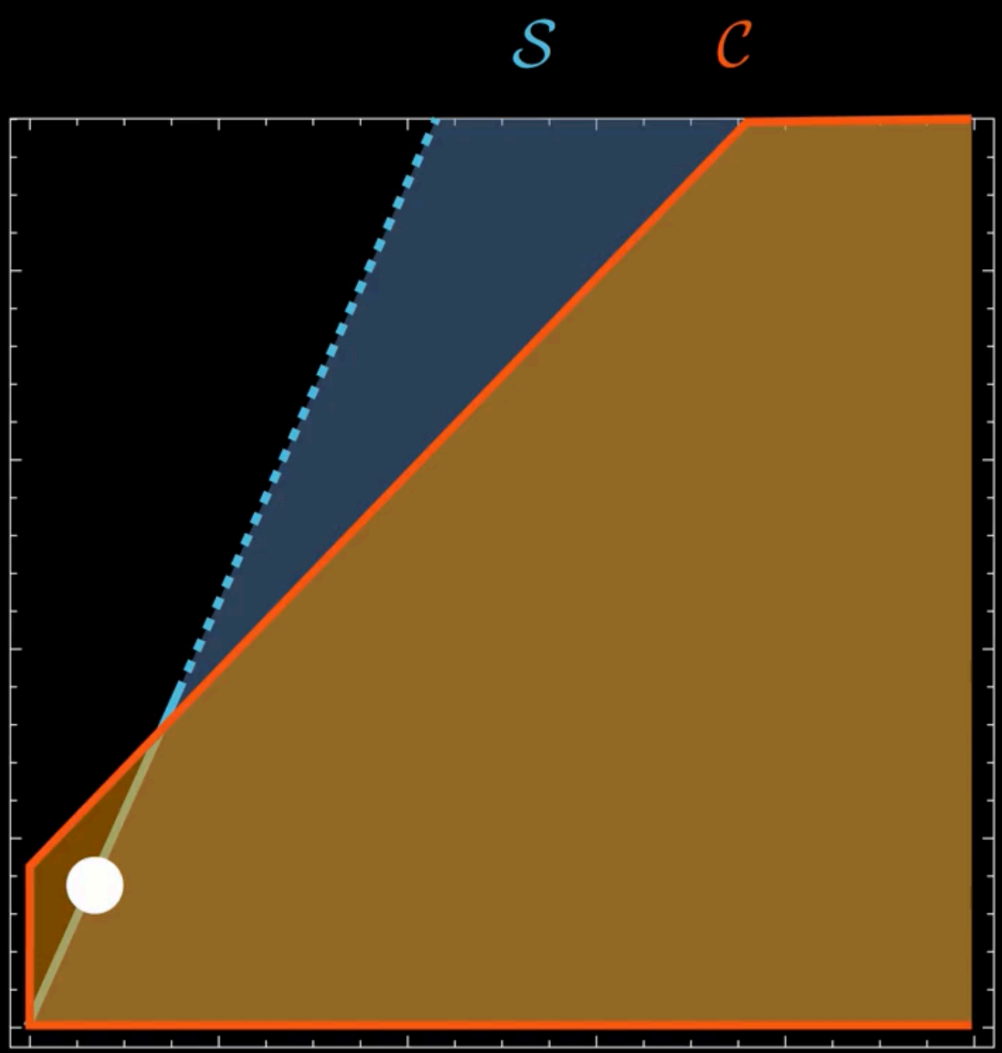
$$h(x) = 0 \text{ and } \inf_{u \in \mathcal{U}} \dot{h}(x,u) < 0$$

$$h(x) = 0 \text{ and } \inf_{u \in \mathcal{U}} \dot{h}(x,u) + \alpha(h(x)) < 0$$

$$h(x) = 0 \text{ and } b(x) < 0$$

$\therefore x \notin \mathcal{C}$ $\quad\quad$ $\therefore$ unsafe $\partial\mathcal{S}$ not in $\mathcal{C}$

Motivating Idea



Consider $b(x) \triangleq \inf_{u \in \mathcal{U}} \dot{h}(x, u) + \alpha(h(x))$

$$\mathcal{C} = \{x : b(x) \geq 0\}$$

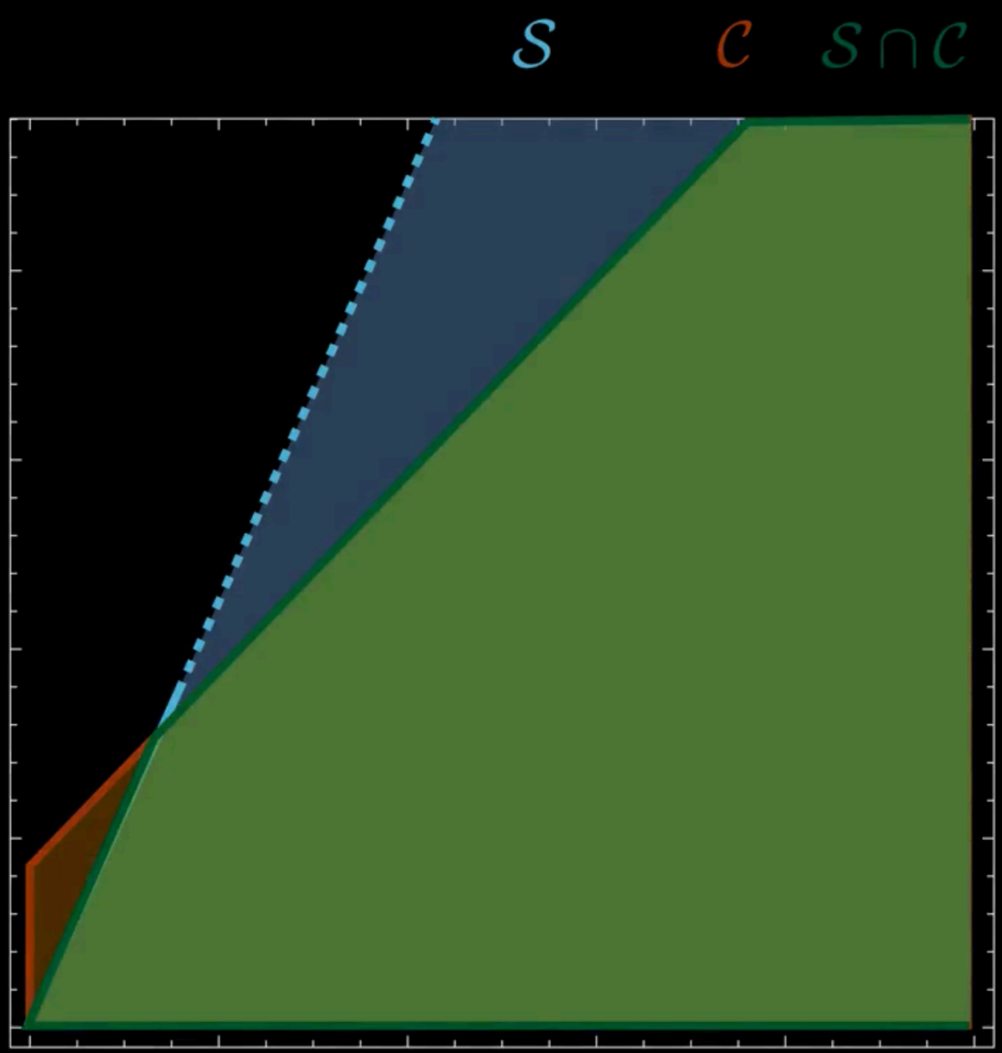
Useful Property 2: If $x \in \mathcal{C}$ and $x \in \partial S$,

$$b(x) \geq 0, \text{ and } h(x) = 0.$$

$$b(x) = \inf_{u \in \mathcal{U}} \dot{h}(x, u) \cancel{+ \alpha(h(x))} \geq 0$$

$$\therefore \forall u \in \mathcal{U} : \dot{h}(x, u) \geq 0$$

Motivating Idea



Consider $b(x) \triangleq \inf_{u \in \mathcal{U}} \dot{h}(x,u) + \alpha(h(x))$

$$\mathcal{C} = \{x : b(x) \geq 0\}$$

Useful Property 2: If $x \in \mathcal{C}$ and $x \in \partial S$,

$$b(x) \geq 0, \text{ and } h(x) = 0.$$

$$b(x) = \inf_{u \in \mathcal{U}} \dot{h}(x,u) \cancel{+ \alpha(h(x))} \geq 0$$

$$\therefore \forall u \in \mathcal{U} : \dot{h}(x,u) \geq 0$$

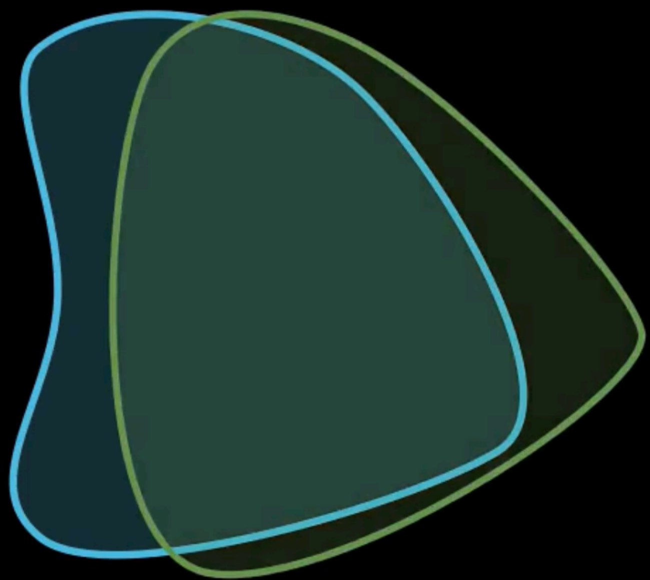
If $u = \pi(x)$ keeps $x \in \mathcal{C}$,
then $\mathcal{S} \cap \mathcal{C}$ is forward invariant

Formal Construction (Main Result)

$$\mathcal{S} = \{x : h(x) \geq 0\}$$

$$b_0(x) = h(x)$$

Formal Construction (Main Result)



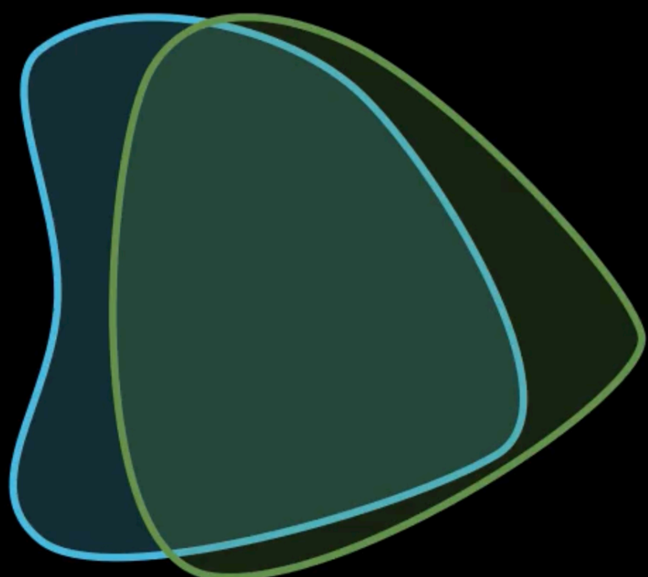
$\mathcal{S} = \{x : h(x) \geq 0\}$

$\mathcal{C}_1 = \{x : b_1(x) \geq 0\}$

$$b_0(x) = h(x)$$

$$b_1(x) = \inf_{u \in \mathcal{U}} \dot{b}_0(x, u) + \alpha_0(b_0(x))$$

Formal Construction (Main Result)

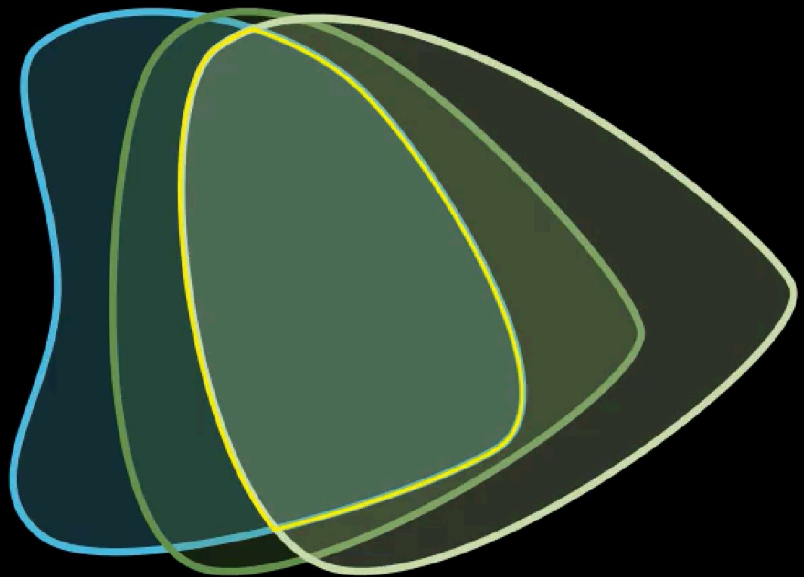


$$b_0(x) = h(x)$$

$$\begin{aligned} b_1(x) &= \inf_{u \in \mathcal{U}} \dot{b}_0(x, u) + \alpha_0(b_0(x)) \\ &= \inf_{u \in \mathcal{U}} L_f b_0(x) + L_g b_0(x) u + \alpha_0(b_0(x)) \end{aligned}$$

($b_i(x)$ does not depend on u)

Formal Construction (Main Result)



$$\mathcal{S} = \{x : h(x) \geq 0\}$$

$$\mathcal{C}_1 = \{x : b_1(x) \geq 0\}$$

$$\dots$$

$$\mathcal{C}_N = \{x : b_N(x) \geq 0\}$$

$$\boxed{\mathcal{C}^* = \mathcal{S} \cap \mathcal{C}_1 \cdots \cap \mathcal{C}_N}$$

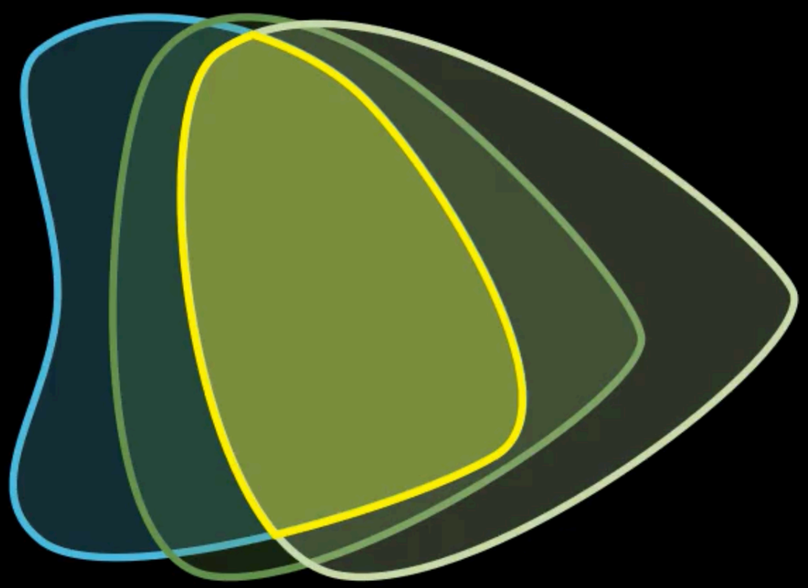
$$b_0(x) = h(x)$$

$$b_1(x) = \inf_{u \in \mathcal{U}} \dot{b}_0(x, u) + \alpha_0(b_0(x))$$

$$\dots$$

$$b_N(x) = \inf_{u \in \mathcal{U}} \dot{b}_{N-1}(x, u) + \alpha_{N-1}(b_{N-1}(x))$$

Formal Construction (Main Result)



$$\mathcal{S} = \{x : h(x) \geq 0\}$$

$$\mathcal{C}_1 = \{x : b_1(x) \geq 0\}$$

$$\cdots$$

$$\mathcal{C}_N = \{x : b_N(x) \geq 0\}$$

$$\boxed{\mathcal{C}^* = \mathcal{S} \cap \mathcal{C}_1 \cdots \cap \mathcal{C}_N}$$

$$b_0(x) = h(x)$$

$$b_1(x) = \inf_{u \in \mathcal{U}} \dot{b}_0(x, u) + \alpha_0(b_0(x))$$

$$\cdots$$

$$b_N(x) = \inf_{u \in \mathcal{U}} \dot{b}_{N-1}(x, u) + \alpha_{N-1}(b_{N-1}(x))$$

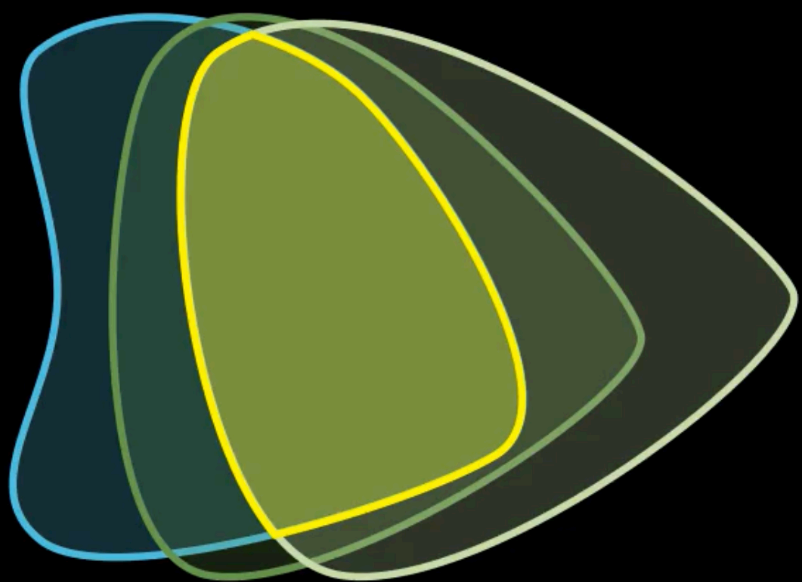
Def:

If $\exists \alpha_N \in \mathcal{K}$ s.t. $\forall x \in \mathcal{C}^*$,

$$\sup_{u \in \mathcal{U}} \dot{b}_N(x, u) + \alpha_N(b_N(x)) \geq 0$$

then b_N is an ICCBF.

Formal Construction (Main Result)



Def:

If $\exists \alpha_N \in \mathcal{K}$ s.t. $\forall x \in \mathcal{C}^*$,

$$\sup_{u \in \mathcal{U}} \dot{b}_N(x, u) + \alpha_N(b_N(x)) \geq 0$$

then b_N is an ICCBF.

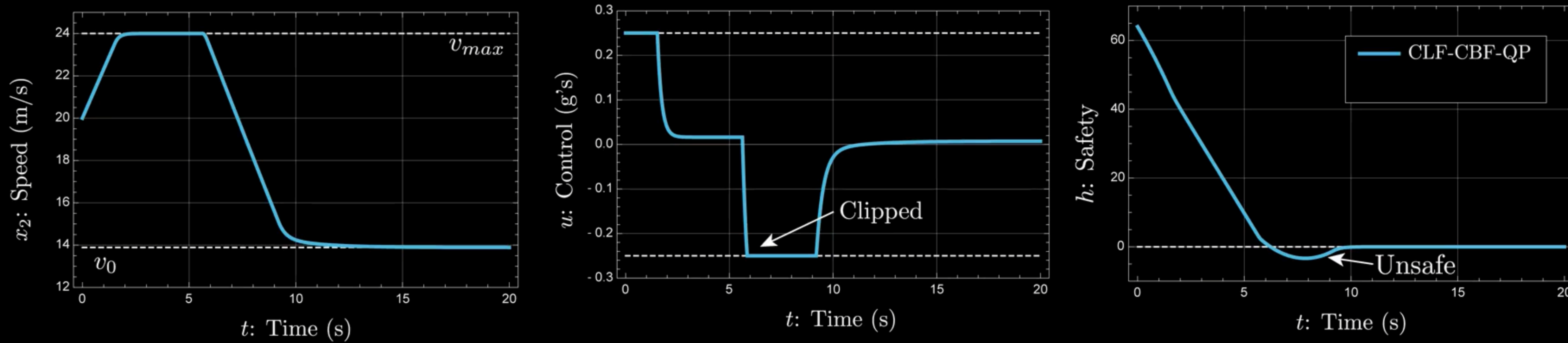
Theorem:

If b_N is an ICCBF, any Lips. controller π s.t. $\pi(x) \in \{u \in \mathcal{U} : \dot{b}_N(x, u) + \alpha_N(b_N(x)) \geq 0\}$ renders $\mathcal{C}^* \subset \mathcal{S}$ forward invariant.

Simulation Results: Adaptive Cruise Control

$$h(x) = x_1 - 1.8x_2$$

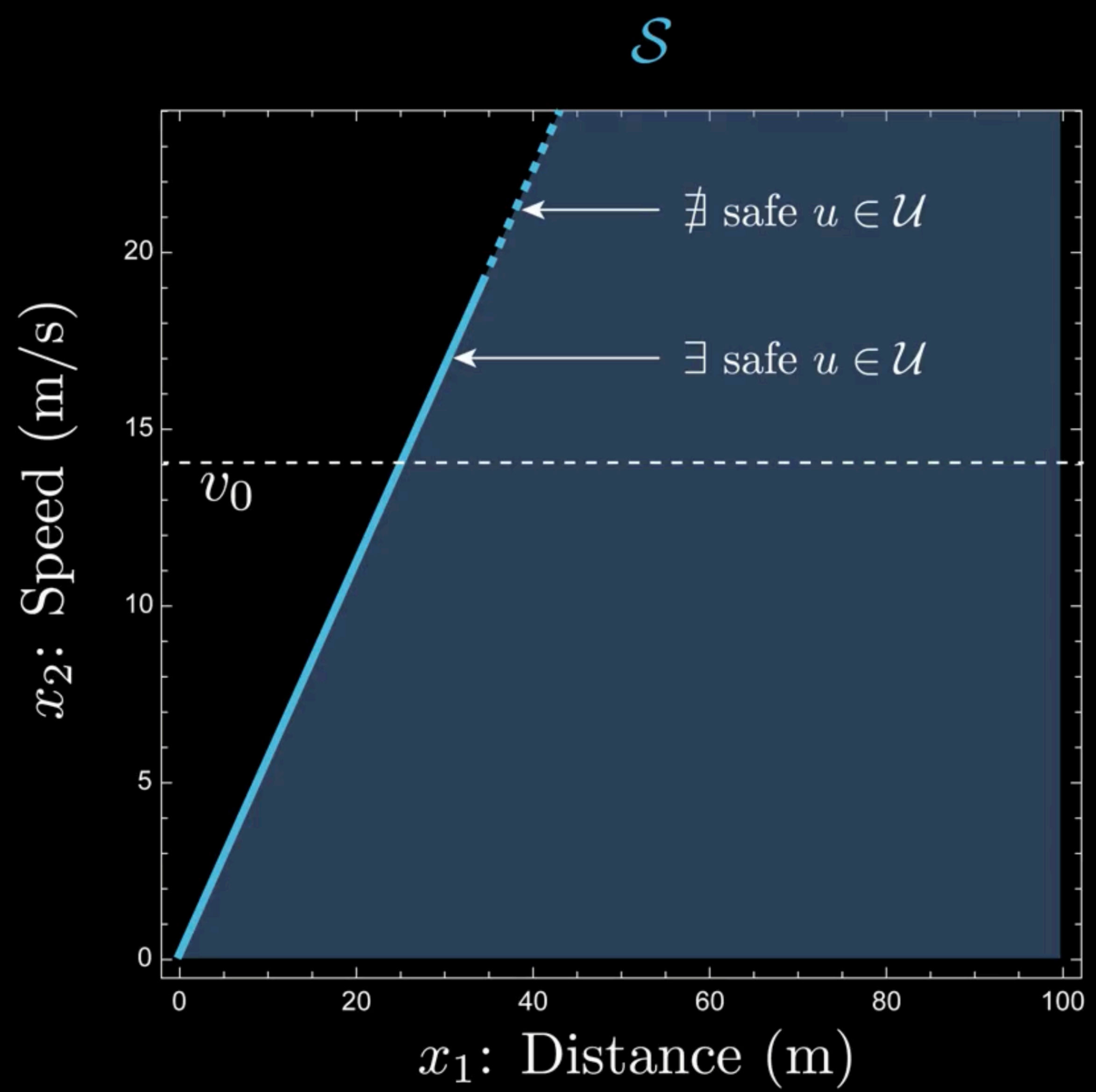
$$\pi(x) = \text{clip}[\ \text{argmin}\, ||u||\ \text{s.t.}\ \ L_f h(x) + L_g h(x)u + \alpha h(x) \geq 0\ ]$$



Applying clipped CLF-CBF-QP controller is not safe since $h(x)$ is not a valid CBF, with input constraints

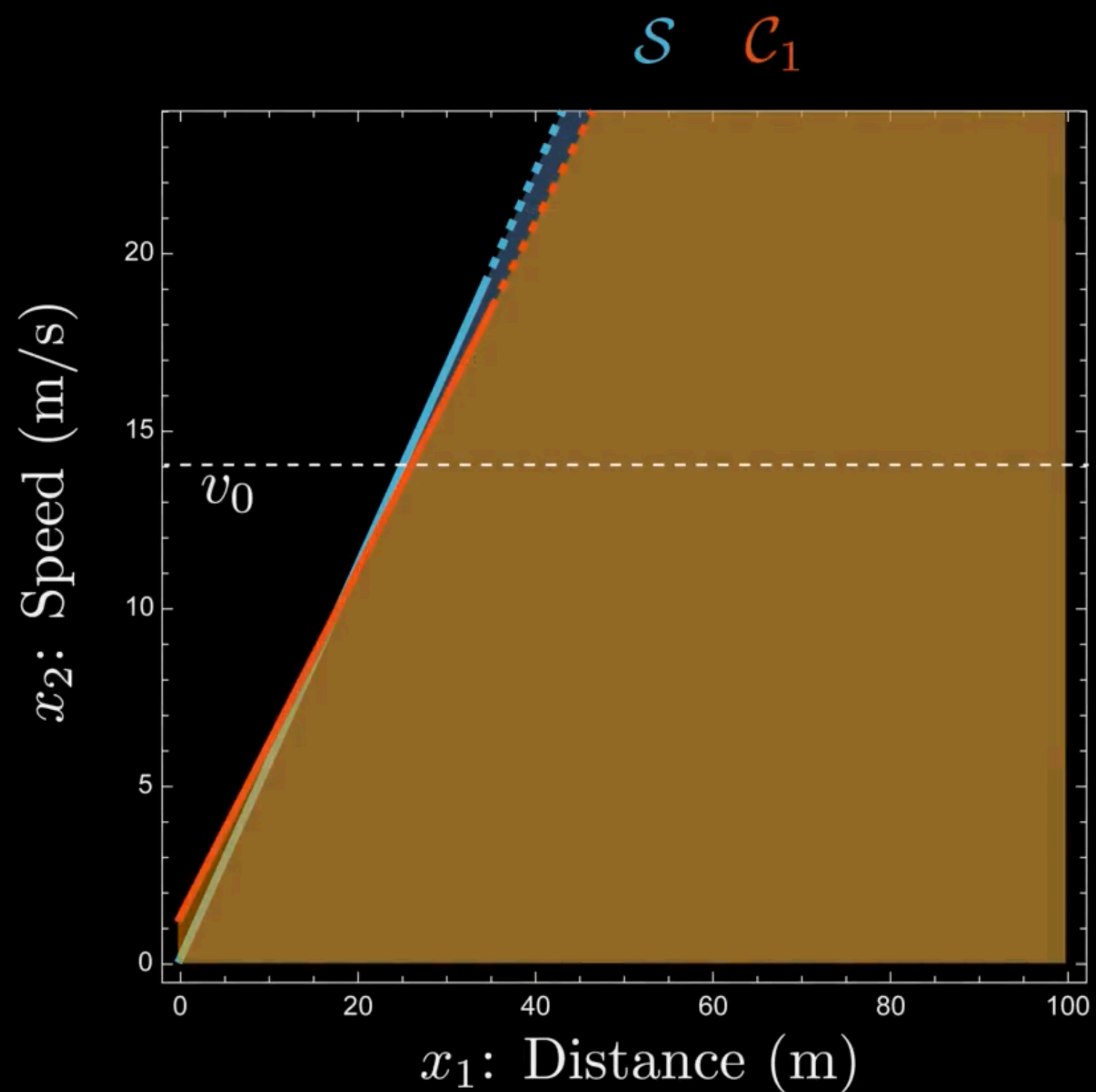
[Ames, 2019]

Simulation Results: Adaptive Cruise Control



$$h(x) = x_1 - 1.8x_2$$

Simulation Results: Adaptive Cruise Control

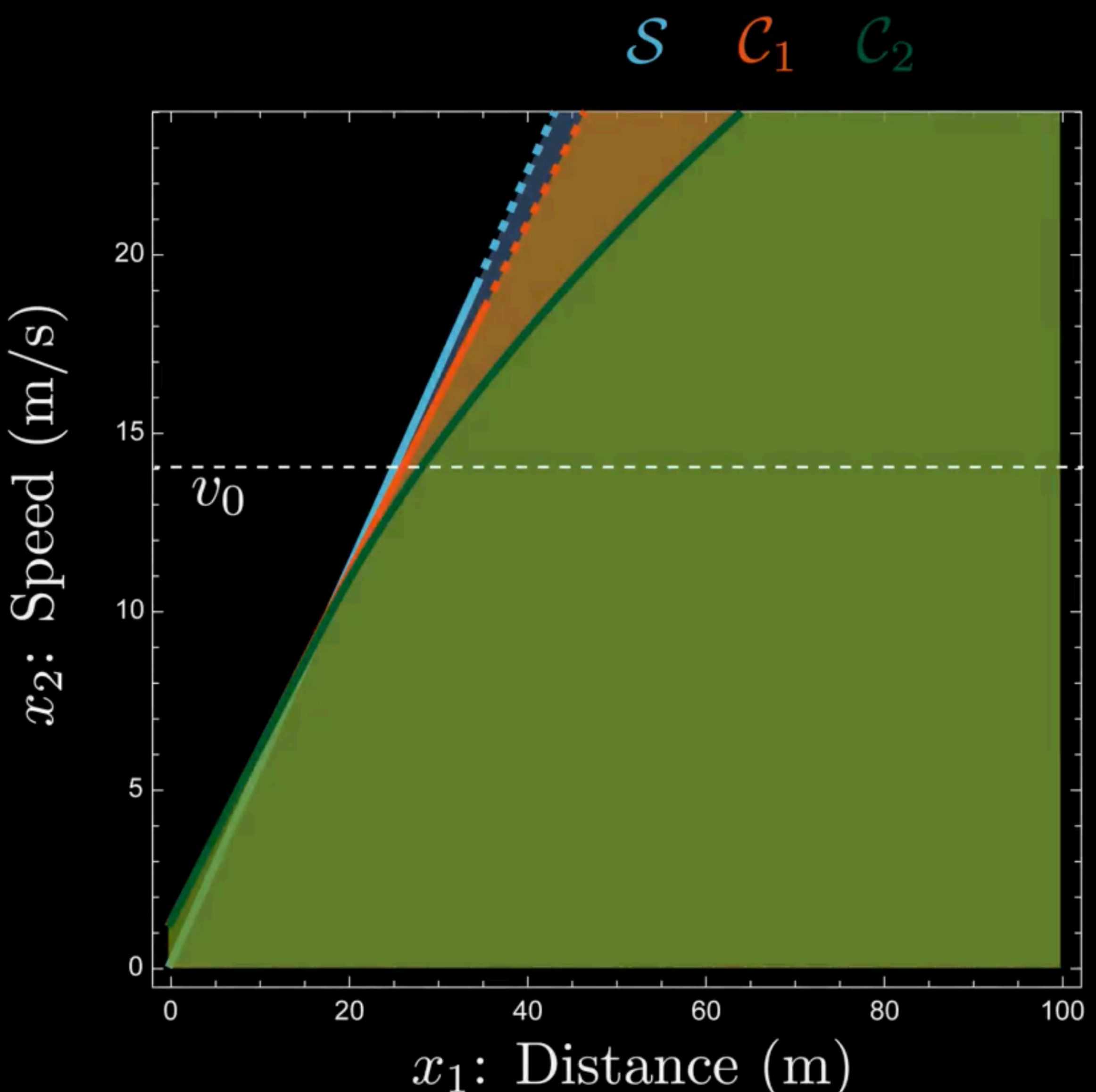


$$h(x) = x_1 - 1.8x_2$$

$$b_1(x) = \inf_{u \in \mathcal{U}} \dot{h} + \alpha_0(h(x))$$

$$(\alpha_0(r) = 4r)$$

Simulation Results: Adaptive Cruise Control



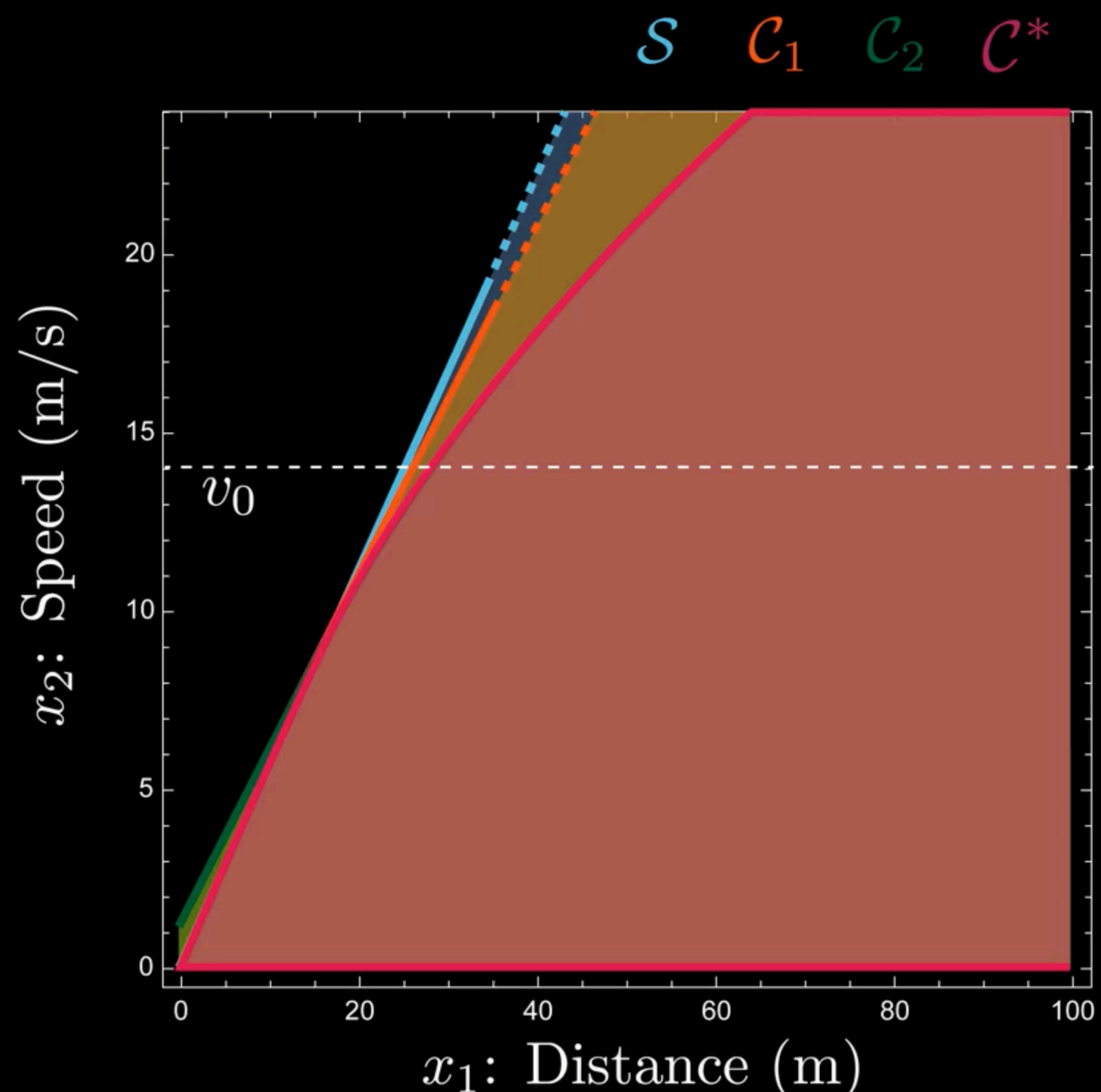
$$h(x) = x_1 - 1.8x_2$$

$$b_1(x) = \inf_{u \in \mathcal{U}} \dot{h} + 4h(x)$$

$$b_2(x) = \inf_{u \in \mathcal{U}} \dot{b}_1 + \alpha_1(b_1(x))$$

$$(\alpha_1(r) = 7\sqrt{r})$$

Simulation Results: Adaptive Cruise Control



$$h(x) = x_1 - 1.8x_2$$

$$b_1(x) = \inf_{u \in \mathcal{U}} \dot{h} + 4h(x)$$

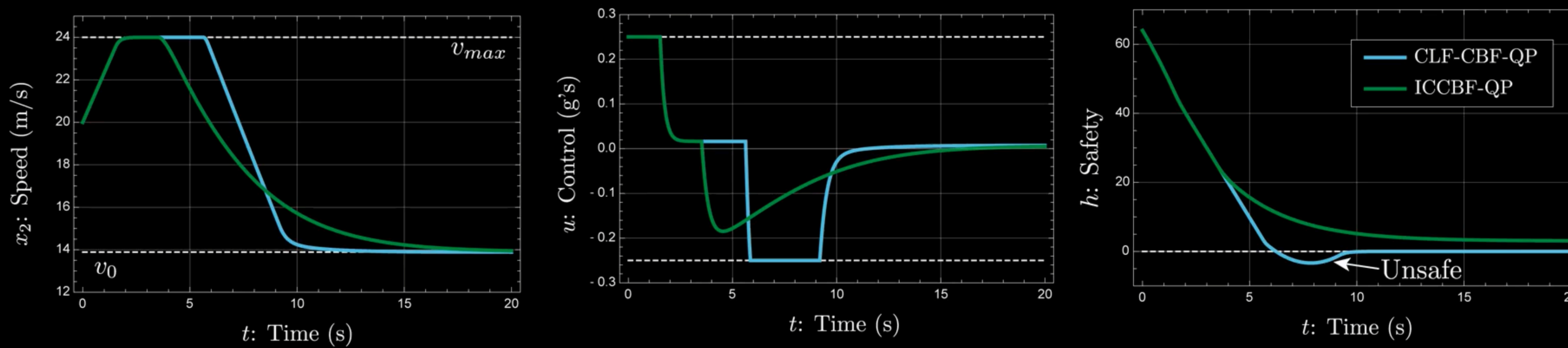
$$b_2(x) = \inf_{u \in \mathcal{U}} \dot{b}_1 + 7\sqrt{b_1(x)}$$

$$\mathcal{C}^* = \mathcal{S} \cap \mathcal{C}_1 \cap \mathcal{C}_2$$

$$\text{Safe } u : \dot{b}_2 + \alpha_2(b_2(x)) \geq 0$$

$$(\alpha_2(r) = 2r)$$

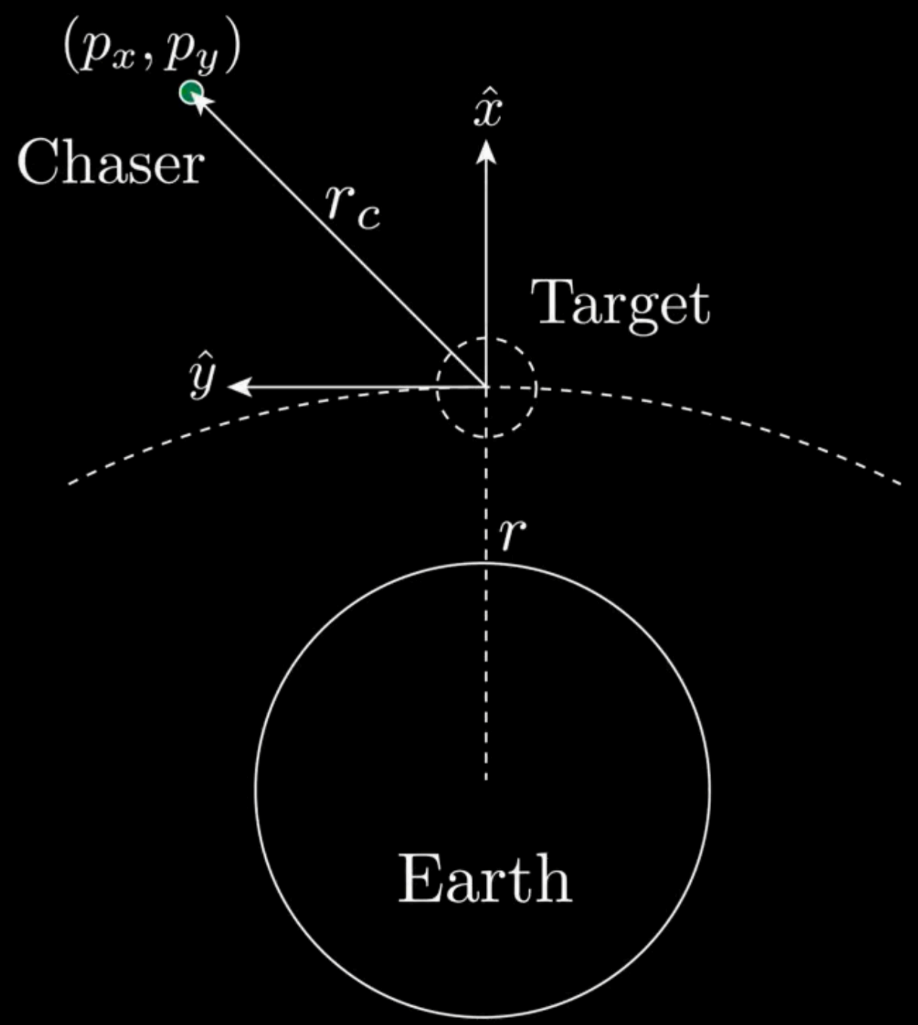
Simulation Results: Adaptive Cruise Control

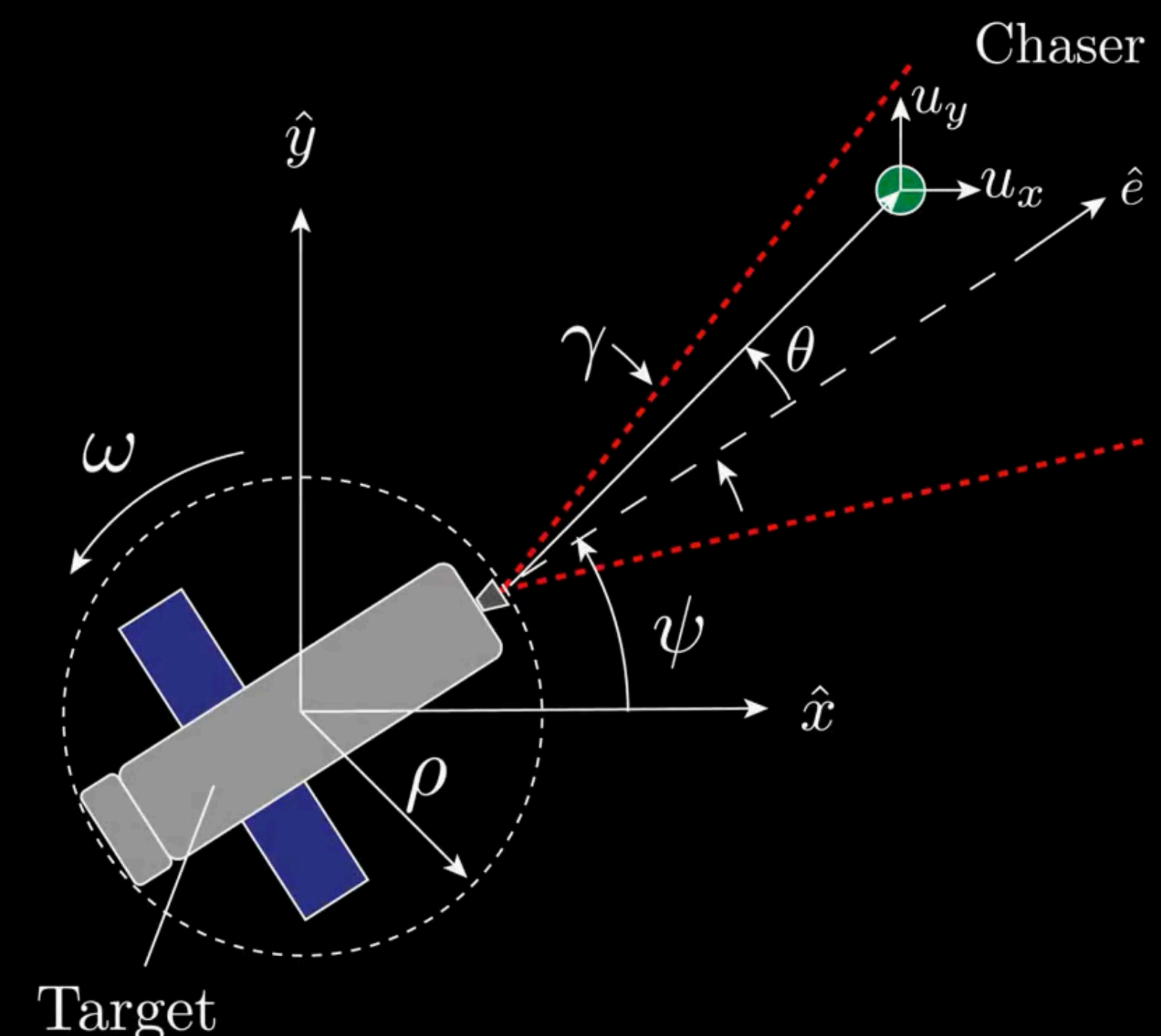


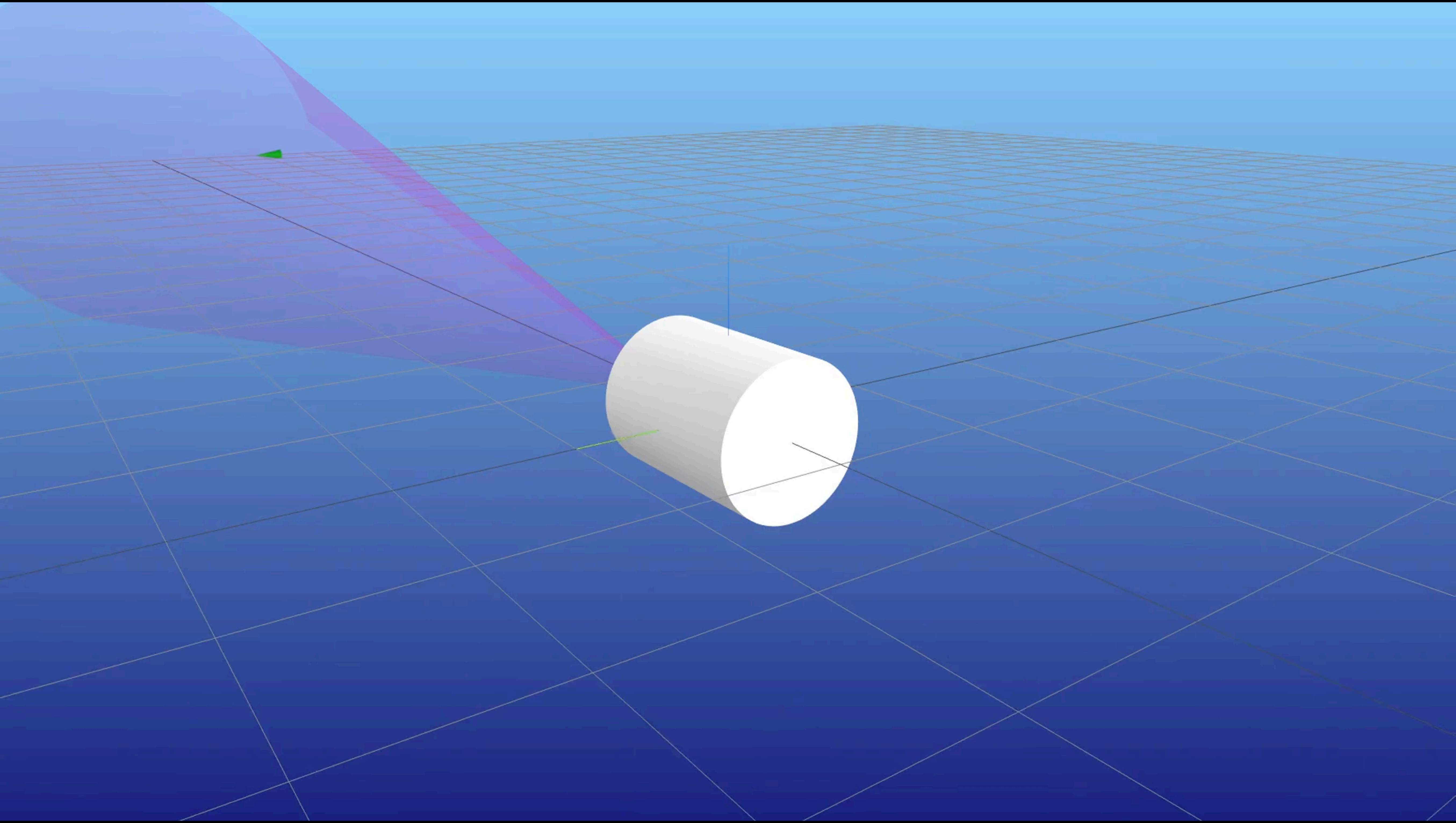
Using the ICCBF, the controller starts braking earlier to maintain safety

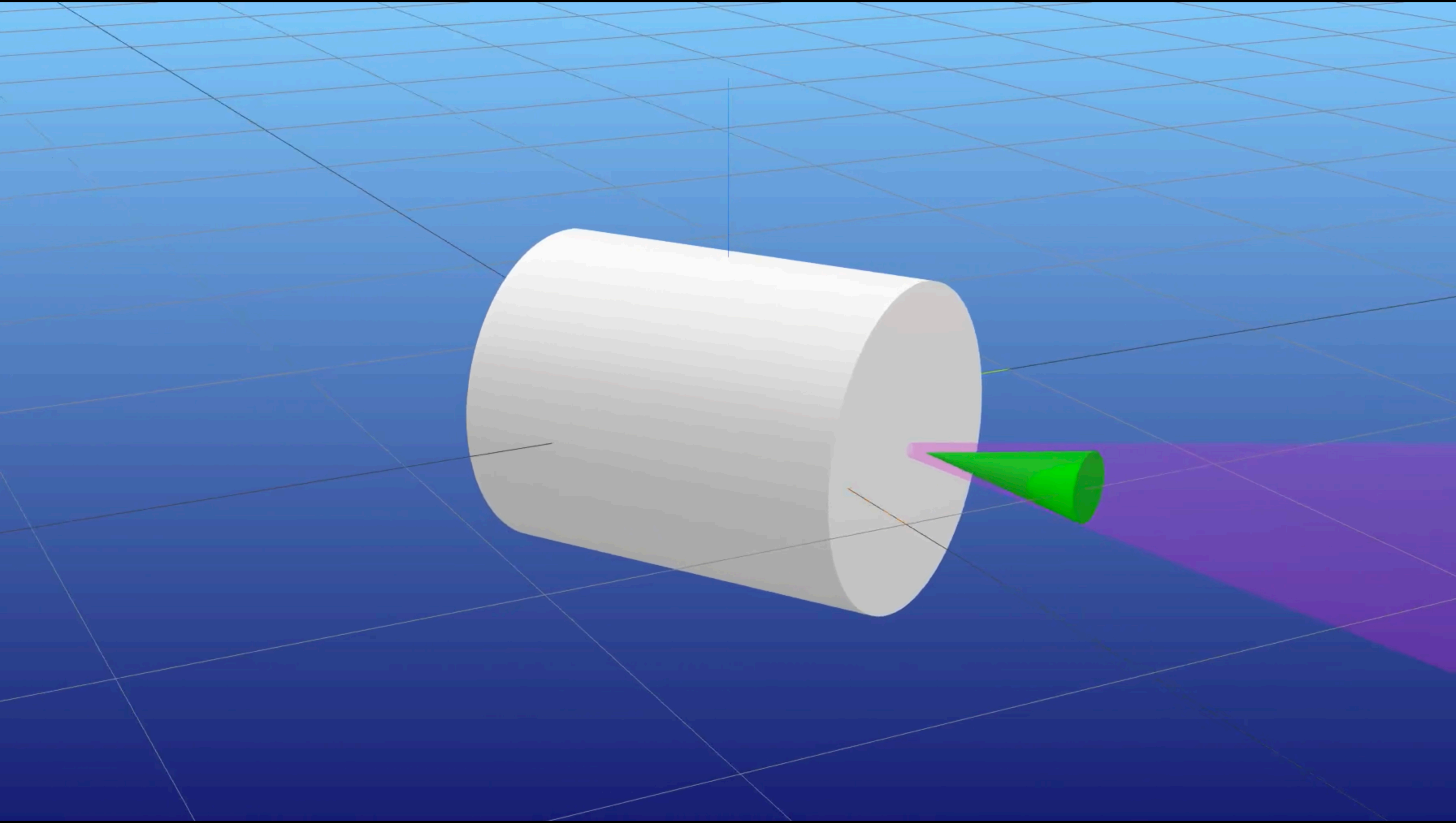
- Background and Problem Statement
- Motivating Idea
- Formal Construction
- Simulation Results

Simulation Results: Docking









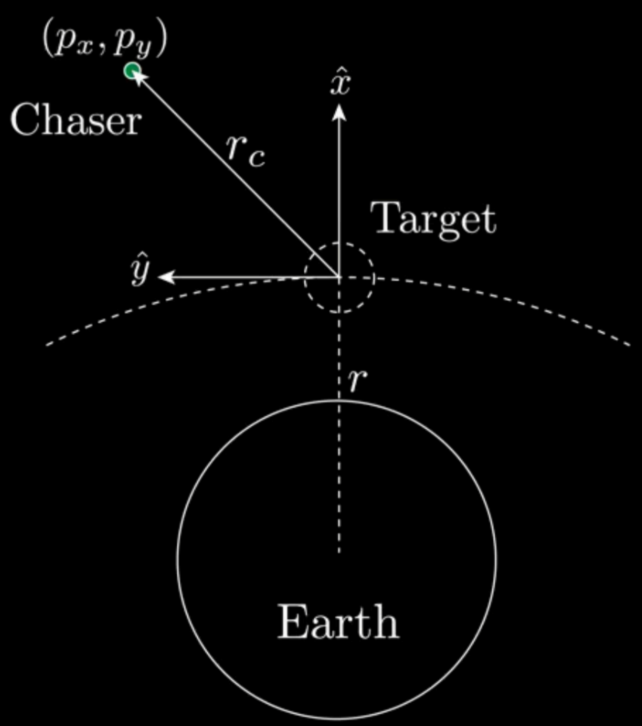
Conclusions: Input Constrained Control Barrier Functions

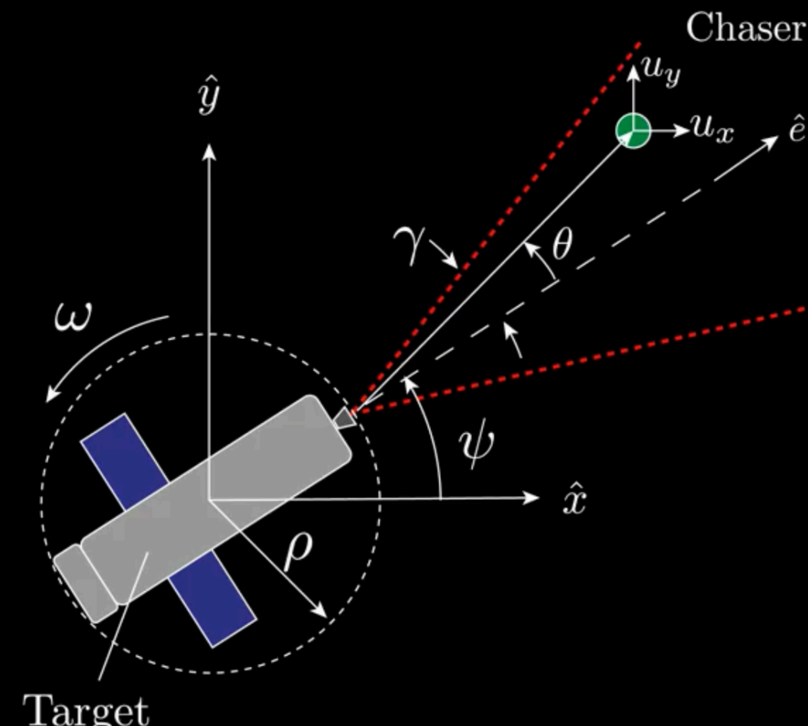
- Method to construct ICCBFs
- Simulation Results
- Remarks
  - Generalisation of Higher Order CBFs [Xiao, 2019]
  - Simple ICCBFs
- Future Directions
  - Noise, Robustness
  - Verification

Appendix

Simulation Results: Docking

$$\frac{d}{dt}\begin{bmatrix} p_x \\ p_y \\ v_x \\ v_y \\ \psi \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ n^2 p_x + 2nv_y + \frac{\mu}{r^2} - \frac{\mu(r+p_x)}{r_c^3} \\ n^2 p_y - 2nv_x - \frac{\mu p_y}{r_c^3} \\ \omega \end{bmatrix} + \frac{1}{m_c}\begin{bmatrix} 0 \\ 0 \\ u_x \\ u_y \\ 0 \end{bmatrix}$$





$$|u_x| + |u_y| \leq 0.25 \text{ kN}$$
$$h(x) = \cos\theta - \cos 10^\circ$$