Control Barrier Functions: Theory Overview and Design Using Dynamic Safety Margins

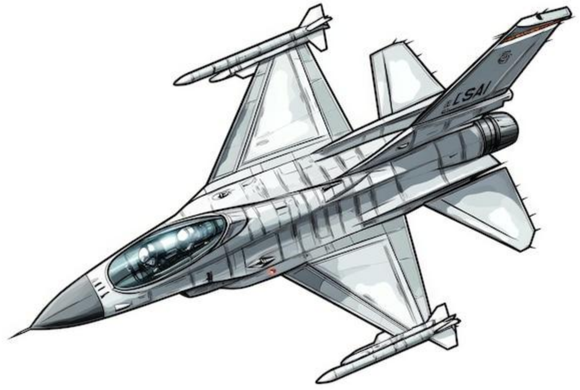
Victor Freire

EuroGNC 2026, Madrid

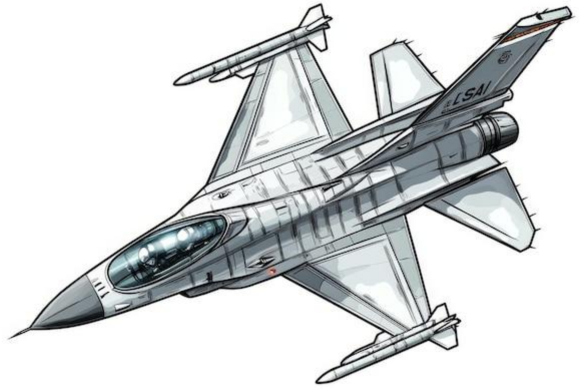
May 4, 2026

University of Colorado Boulder

Introduction



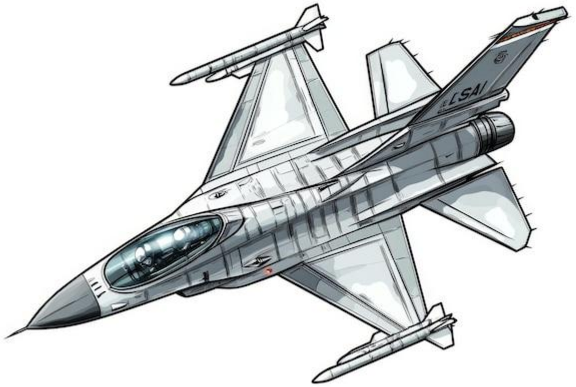
Introduction



Stability

Essentially solved

Introduction



Stability

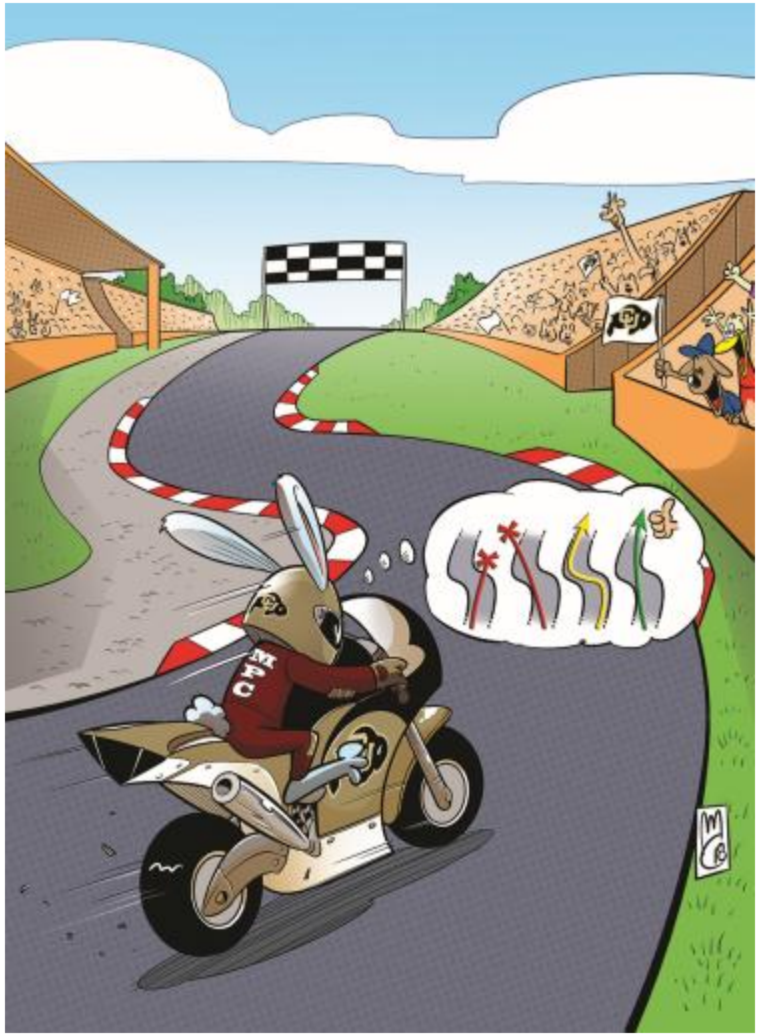
Essentially solved

Safety

Under study
(Workshop)

Constrained Control

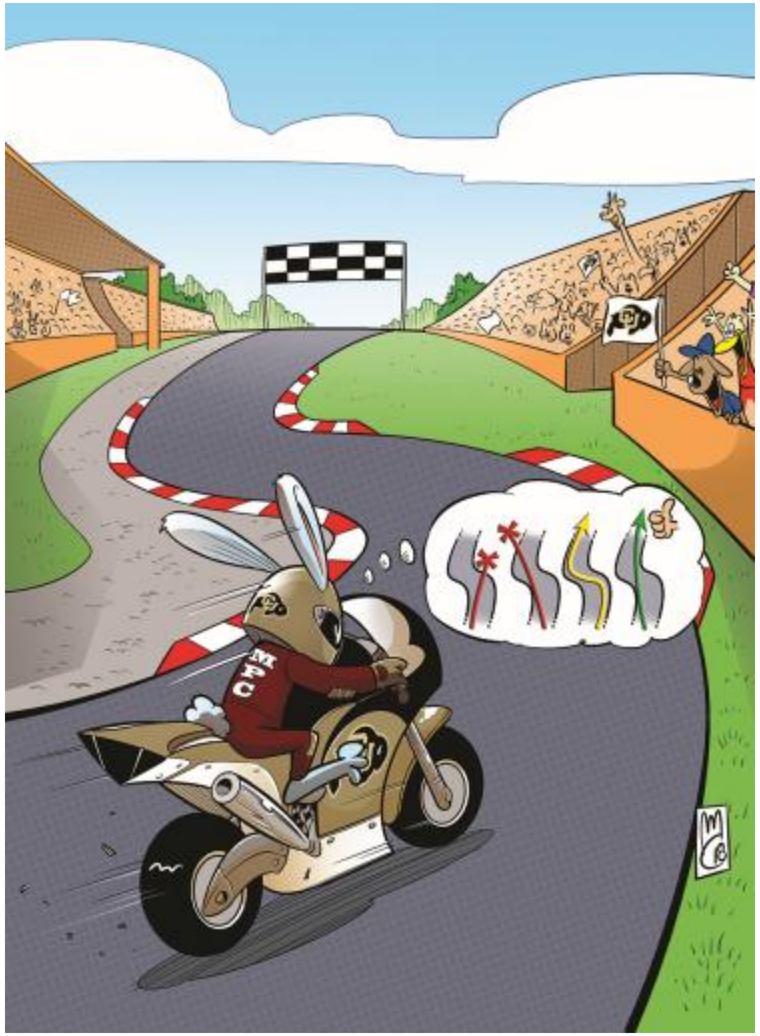
- **Model Predictive Control**
  Full control law redesign
  High computational cost

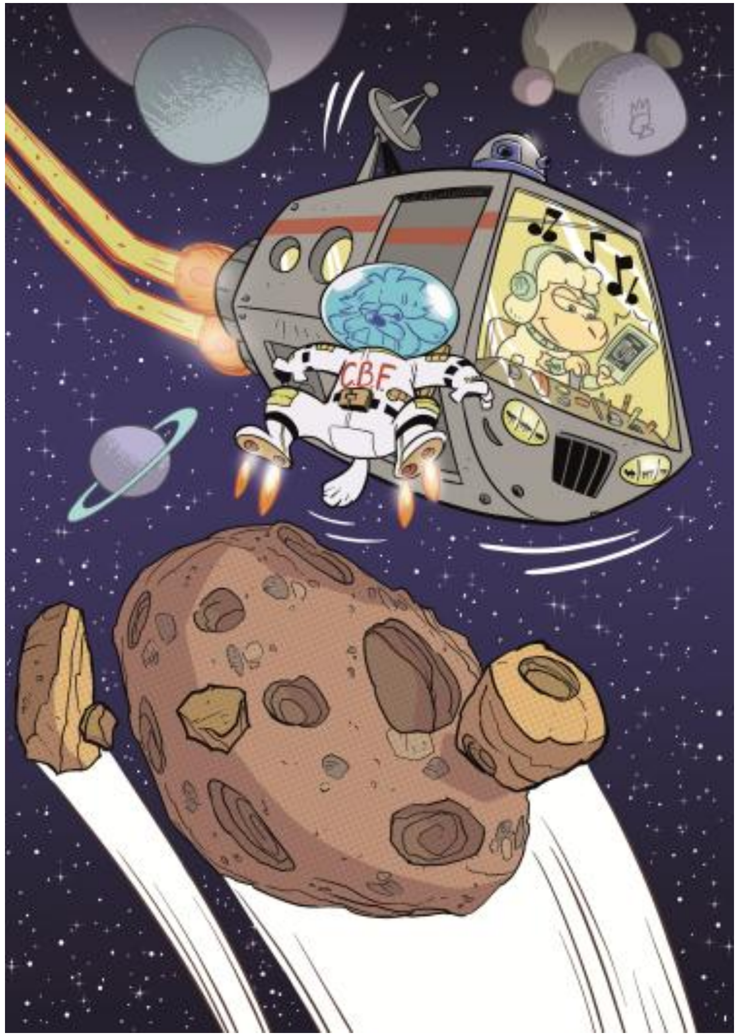


Constrained Control

- **Model Predictive Control**
  Full control law redesign
  High computational cost

- **Safety Filters**
  Keep legacy controller
  Lower computational cost





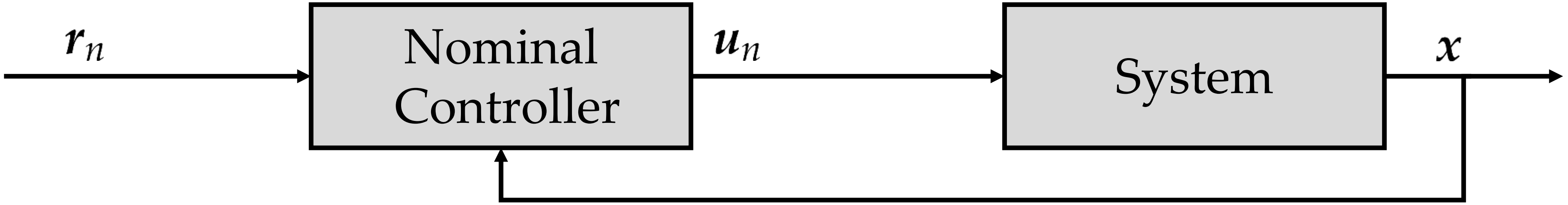
Safety Filters

Family of constrained controllers that use a two-step approach:

Safety Filters

Family of constrained controllers that use a two-step approach:

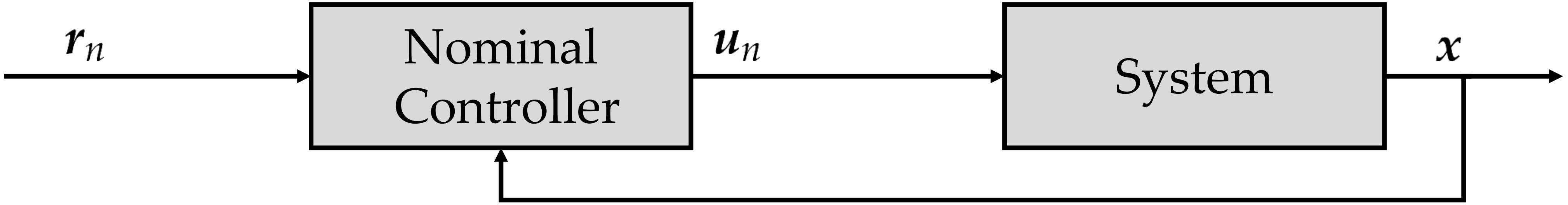
1. Design a control law for the **Unconstrained** system,



Safety Filters

Family of constrained controllers that use a two-step approach:

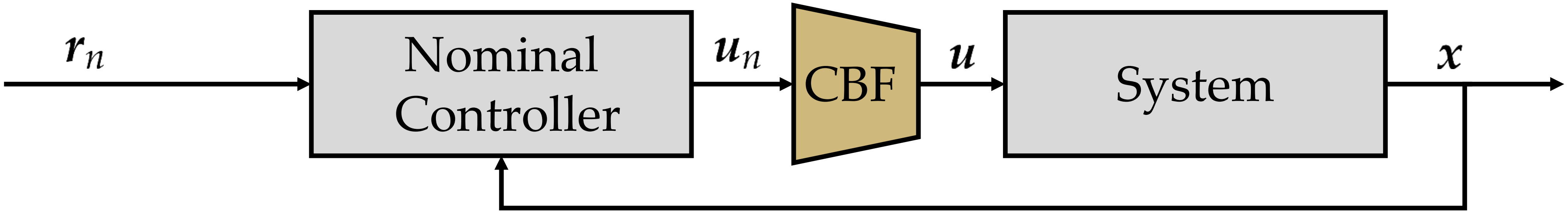
1. Design a control law for the **Unconstrained** system,
2. Introduce an **Add-on** unit for constraint enforcement.



Safety Filters

Family of constrained controllers that use a two-step approach:

1. Design a control law for the **Unconstrained** system,
2. Introduce an **Add-on** unit for constraint enforcement.
   - **Control Barrier Function:** Filter nominal controller



Index

Problem: *Constrained Control*

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

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

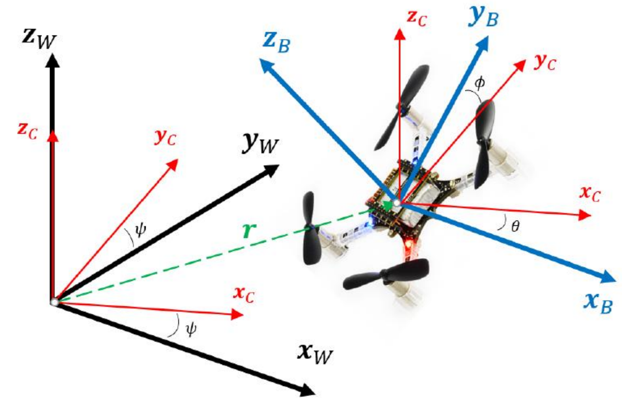
Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

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

Quadcopter Dynamics



$$x = \begin{bmatrix} \boldsymbol{r} \\ \xi \\ \dot{\boldsymbol{r}} \end{bmatrix} \qquad \xi = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}$$

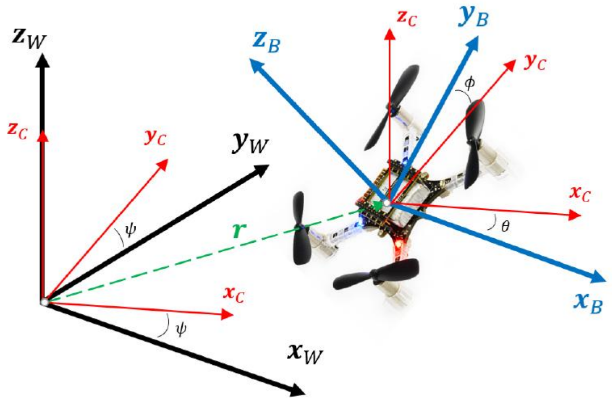
Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

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

Quadcopter Dynamics



$$G_\xi(\xi) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\phi \\ 0 & c\phi & -s\phi \\ 0 & \frac{s\phi}{c\theta} & \frac{c\phi}{c\theta} \end{bmatrix}$$

$$G_T(\xi) = \begin{bmatrix} s\phi s\psi + c\phi s\theta c\psi \\ c\phi s\theta s\psi - s\phi c\psi \\ c\phi c\theta \end{bmatrix} \qquad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{\xi} \\ \dot{\boldsymbol{r}} \end{bmatrix} \qquad \boldsymbol{\xi} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}$$

$$f(\boldsymbol{x}) = \begin{bmatrix} \dot{\boldsymbol{r}} \\ 0 \\ -g_a z_W \end{bmatrix} \qquad g(\boldsymbol{x}) = \begin{bmatrix} 0 & 0 \\ 0 & G_\xi(\boldsymbol{\xi}) \\ G_T(\boldsymbol{\xi}) & 0 \end{bmatrix}$$

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system
$$\dot{x}(t) = f\big(x(t)\big) + g\big(x(t)\big)u(t)$$

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix}\theta & \dot{\theta} & \alpha_w\end{bmatrix}^\top$$
$$\boldsymbol{u} = \begin{bmatrix}\delta_e & \delta_f\end{bmatrix}^\top$$

θ pitch $\quad$ α_w angle of attack $\quad$ δ_e, δ_f control surfaces

Problem: *Constrained Control*

Given a control-affine system

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

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix}\theta & \dot{\theta} & \alpha_w\end{bmatrix}^\top$$

$$\boldsymbol{u} = \begin{bmatrix}\delta_e & \delta_f\end{bmatrix}^\top$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix}0 & 1 & 0\\ 0 & -0.87 & 43.22\\ 0 & 0.99 & -1.34\end{bmatrix} \quad B = \begin{bmatrix}0 & 0\\ -17.25 & -1.58\\ -0.17 & -0.25\end{bmatrix}$$

Short period approximation

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system
$$\dot{x}(t) = f\left(x(t)\right) + g\left(x(t)\right)u(t)$$
and constraints
$$\mathcal{X} \subset \mathbb{R}^n,\ \mathcal{U} \subset \mathbb{R}^m,$$

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix}\theta & \dot{\theta} & \alpha_w\end{bmatrix}^\top$$
$$\boldsymbol{u} = \begin{bmatrix}\delta_e & \delta_f\end{bmatrix}^\top$$
$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

Pitch pointing constraint
$$\mathcal{X} = \{\boldsymbol{x} \in \mathbb{R}^3 \mid |\alpha_w| \leq 4 \text{ deg}\}$$

Control surface deflection limits
$$\mathcal{U} = \{\boldsymbol{u} \in \mathbb{R}^2 \mid |\delta_e| \leq 25 \text{ deg}, |\delta_f| \leq 20 \text{ deg}\}$$

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

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

and constraints

$$\mathcal{X} \subset \mathbb{R}^n, \ \mathcal{U} \subset \mathbb{R}^m,$$

find a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

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

and constraints

$$\mathcal{X} \subset \mathbb{R}^n, \ \mathcal{U} \subset \mathbb{R}^m,$$

find a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Bonus: Trajectory $x(t)$ achieves goal.

(Performance)

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

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

and constraints

$$\mathcal{X} \subset \mathbb{R}^n, \; \mathcal{U} \subset \mathbb{R}^m,$$

find a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

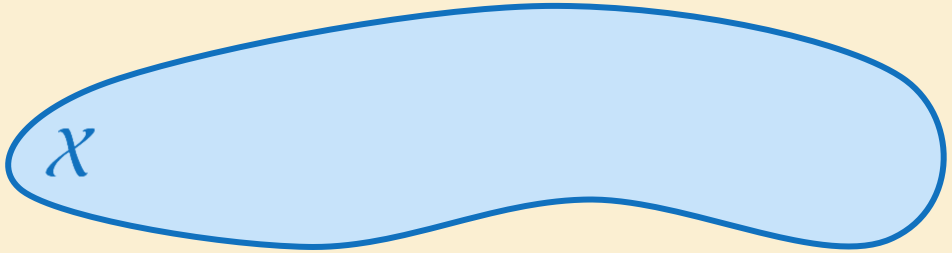
Is there a solution?

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system
$$\dot{x}(t) = f\big(x(t)\big) + g\big(x(t)\big)u(t)$$
and constraints
$$\mathcal{X} \subset \mathbb{R}^n,\ \mathcal{U} \subset \mathbb{R}^m,$$
find a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Definition: *Control Invariance*



$\mathcal{X}$ is **control invariant** if for any initial condition $x_0 \in \mathcal{X}$, there exists a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$.

Solution exists if $\mathcal{X}$ is **control invariant**.

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

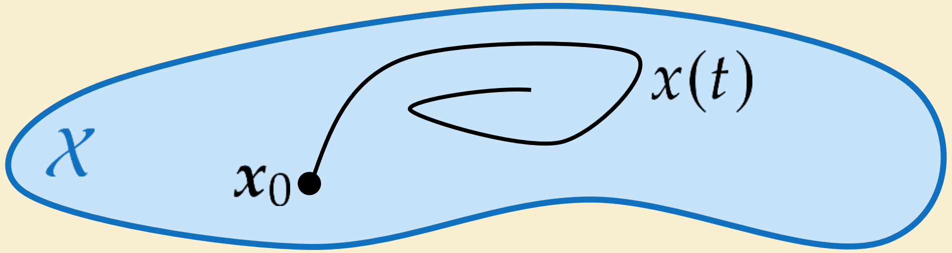
$$\dot{x}(t) = f\big(x(t)\big) + g\big(x(t)\big)u(t)$$

and constraints

$$\mathcal{X} \subset \mathbb{R}^n, \ \mathcal{U} \subset \mathbb{R}^m,$$

find a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Definition: *Control Invariance*



$\mathcal{X}$ is **control invariant** if for any initial condition $x_0 \in \mathcal{X}$, there exists a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$.

Solution exists if $\mathcal{X}$ is **control invariant**.

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

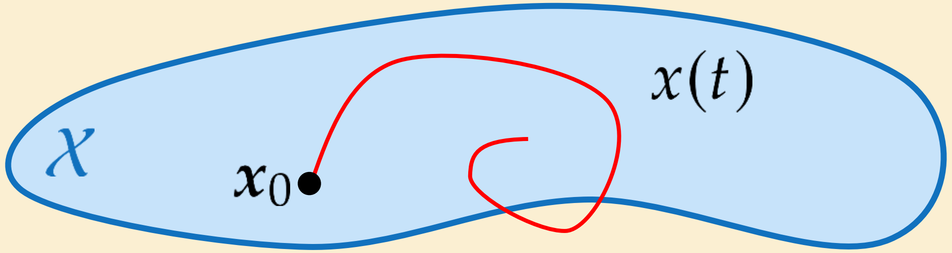
$$\dot{x}(t) = f\big(x(t)\big) + g\big(x(t)\big)u(t)$$

and constraints

$$\mathcal{X} \subset \mathbb{R}^n, \ \mathcal{U} \subset \mathbb{R}^m,$$

find a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Definition: *Control Invariance*



$\mathcal{X}$ is **control invariant** if for any initial condition $x_0 \in \mathcal{X}$, there exists a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$.

What if $\mathcal{X}$ is not ***control invariant?***

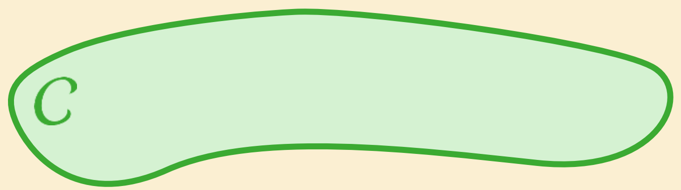
Blanchini, *Automatica* 1999

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system
$$\dot{x}(t) = f\big(x(t)\big) + g\big(x(t)\big)u(t)$$
and constraints
$$\mathcal{X} \subset \mathbb{R}^n, \; \mathcal{U} \subset \mathbb{R}^m,$$
find a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Definition: *Control Invariance*



C is **control invariant** if for any initial condition $x_0 \in C$, there exists a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in C$.

What if $\mathcal{X}$ is not ***control invariant?***

Safety and Invariance

Problem: *Constrained Control*

Given a control-affine system

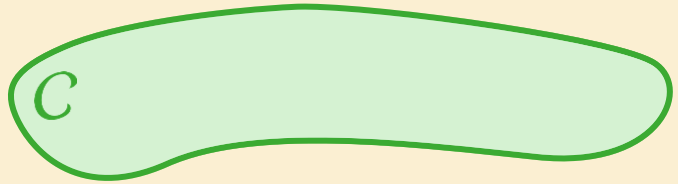
$$\dot{x}(t) = f\big(x(t)\big) + g\big(x(t)\big)u(t)$$

and constraints

$$\mathcal{X} \subset \mathbb{R}^n, \ \mathcal{U} \subset \mathbb{R}^m,$$

find a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

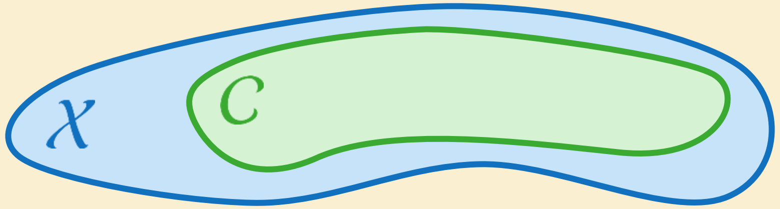
Definition: *Control Invariance*



C is **control invariant** if for any initial condition $x_0 \in C$, there exists a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in C$.

Solution exists if there exists C **control invariant** and **admissible** (i.e., **safe**).

A set C is **admissible** if $C \subset \mathcal{X}$.



Safety and Invariance

Theorem: *Nagumo (simplified)*

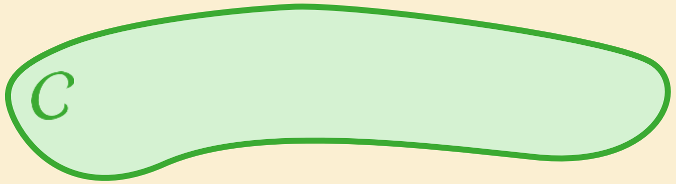
Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$\forall \boldsymbol{x} \in \partial C,\ \exists \boldsymbol{u} \in \mathcal{U},\ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$

Definition: *Control Invariance*



C is **control invariant** if for any initial condition $\boldsymbol{x}_0 \in C$, there exists a signal $u : [0, \infty) \to \mathcal{U}$ such that $x(t) \in C$.

Safety and Invariance

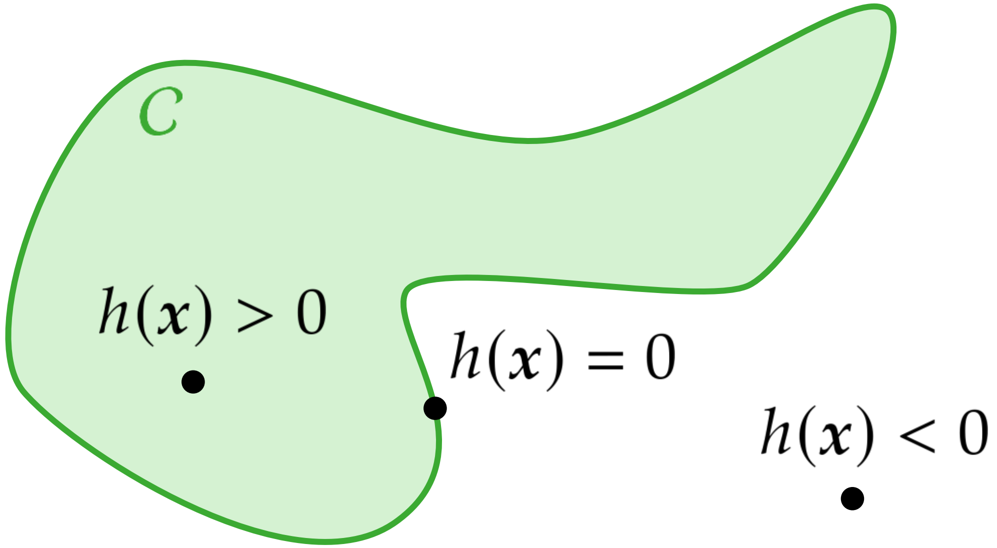
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$$\forall \boldsymbol{x} \in \partial C,\ \exists \boldsymbol{u} \in \boldsymbol{\mathcal{U}},\ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$$



Safety and Invariance

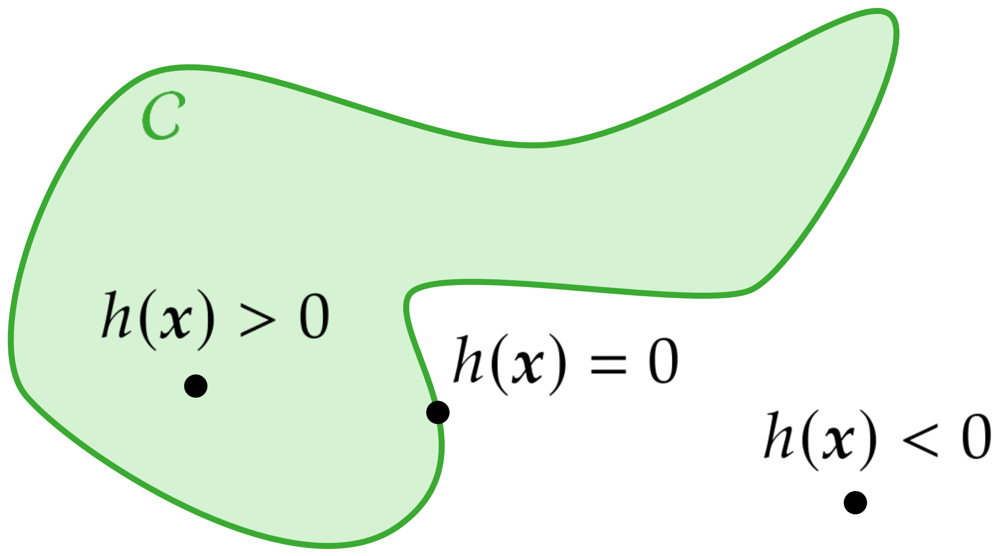
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$$\forall \boldsymbol{x} \in \partial C,\ \exists \boldsymbol{u} \in \mathcal{U},\ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$$



Safety and Invariance

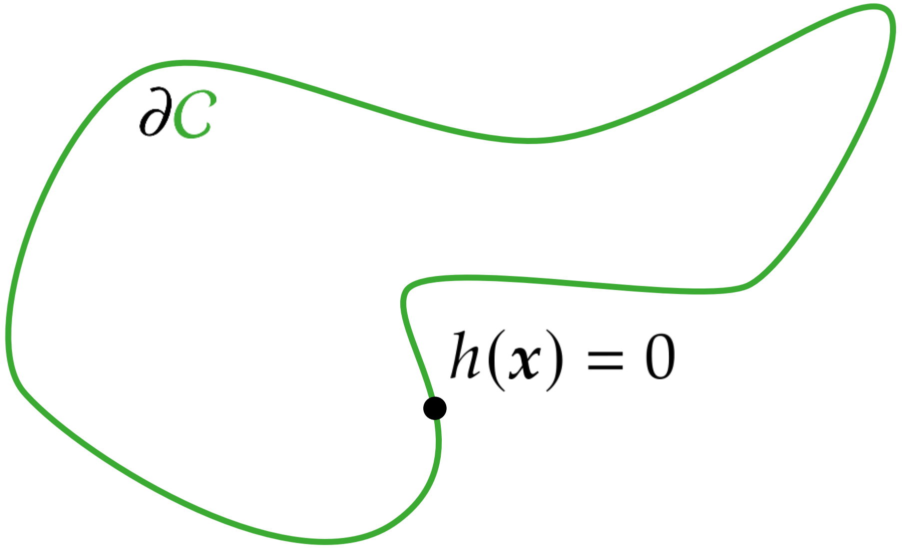
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$$\forall \boldsymbol{x} \in \partial C, \ \exists \boldsymbol{u} \in \boldsymbol{\mathcal{U}}, \ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$$



Safety and Invariance

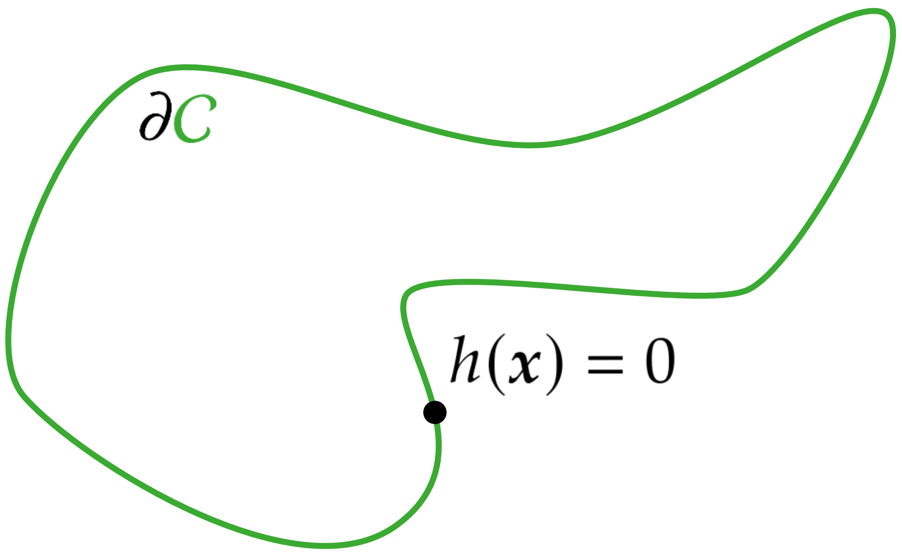
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$\forall \boldsymbol{x} \in \partial C, \exists \boldsymbol{u} \in \mathcal{U}, \; L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$



$L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$

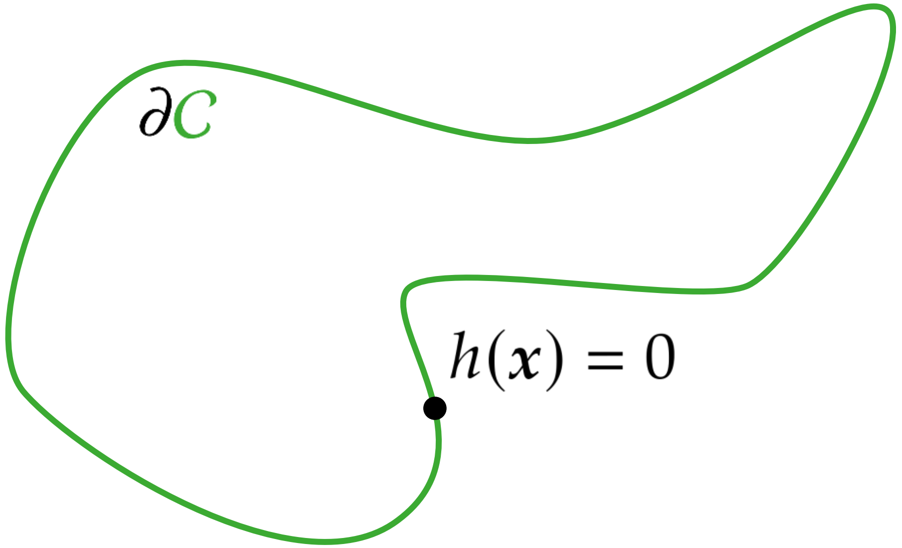
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$$\forall \boldsymbol{x} \in \partial C, \ \exists \boldsymbol{u} \in \mathcal{U}, \ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$$



$$L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) + \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})\boldsymbol{u}$$

"Lie derivatives" (Notation)

$$L_f h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) \qquad L_g h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})$$

Nagumo, *Phys.-Math. Soc. Jpn.* 1942
Aubin, Frankowska, **Set-Valued Analysis** 2009

Safety and Invariance

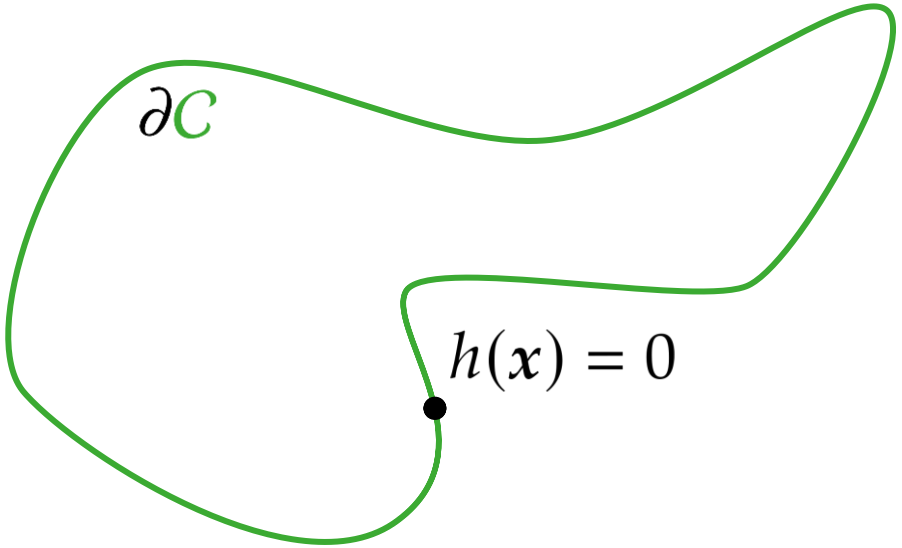
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \rightarrow \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$$\forall \boldsymbol{x} \in \partial C, \ \exists \boldsymbol{u} \in \mathcal{U}, \ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$$



"Lie derivatives" (Notation)

$$L_f h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) \qquad L_g h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})$$

$$L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) + \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}} \big(f(\boldsymbol{x}) + g(\boldsymbol{x})\boldsymbol{u}\big)$$

Nagumo, *Phys.-Math. Soc. Jpn.* 1942

Aubin, Frankowska, **Set-Valued Analysis** 2009

Safety and Invariance

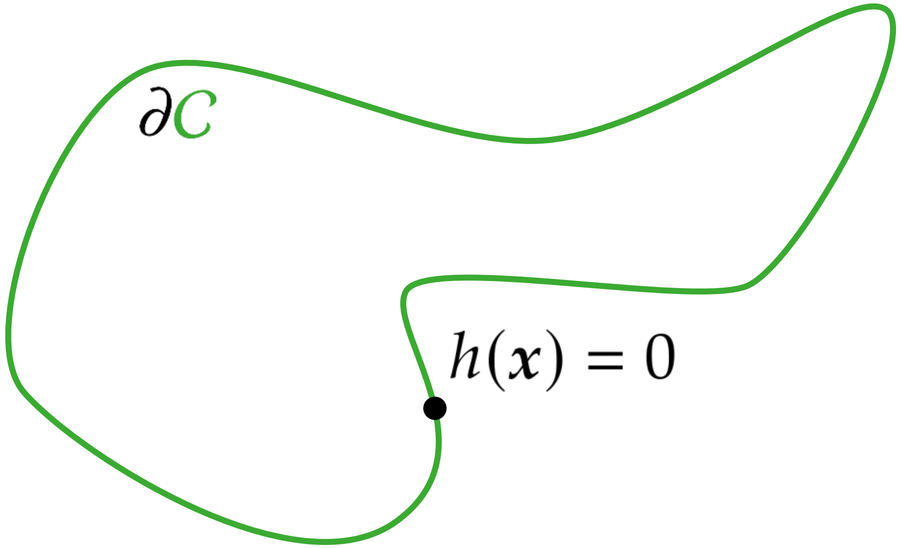
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$\forall \boldsymbol{x} \in \partial C,\ \exists \boldsymbol{u} \in \mathcal{U},\ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$



$$L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) + \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})\boldsymbol{u}$$
$$= \frac{\partial h}{\partial \boldsymbol{x}}\big(f(\boldsymbol{x}) + g(\boldsymbol{x})\boldsymbol{u}\big)$$
$$= \frac{\partial h}{\partial \boldsymbol{x}}\dot{\boldsymbol{x}}$$

"Lie derivatives" (Notation)

$$L_f h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) \qquad L_g h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})$$

Safety and Invariance

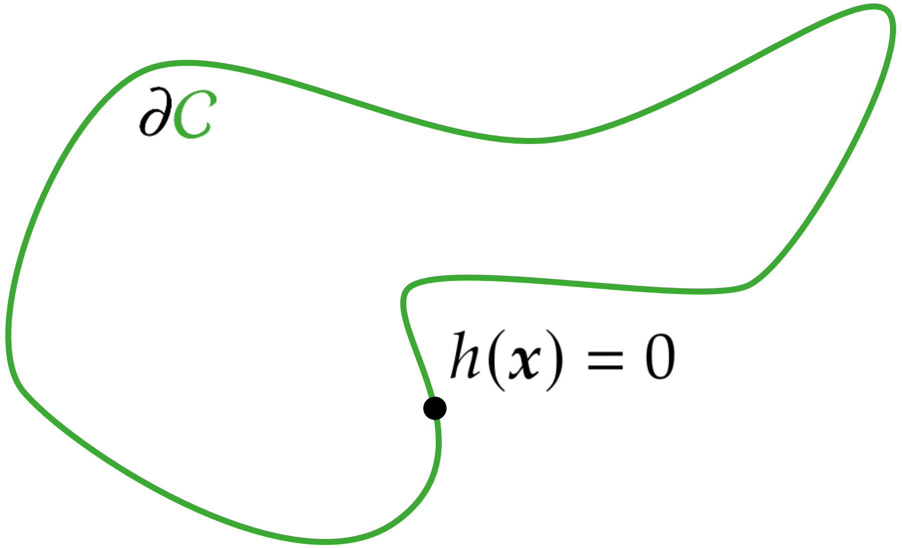
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$\forall \boldsymbol{x} \in \partial C,\ \exists \boldsymbol{u} \in \mathcal{U},\ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$



"Lie derivatives" (Notation)

$$L_f h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) \qquad L_g h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})$$

$$\begin{aligned} L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} &= \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) + \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})\boldsymbol{u} \\ &= \frac{\partial h}{\partial \boldsymbol{x}} \big(f(\boldsymbol{x}) + g(\boldsymbol{x})\boldsymbol{u}\big) \\ &= \frac{\partial h}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}} = \dot{h}(\boldsymbol{x}, \boldsymbol{u}) \end{aligned}$$

Safety and Invariance

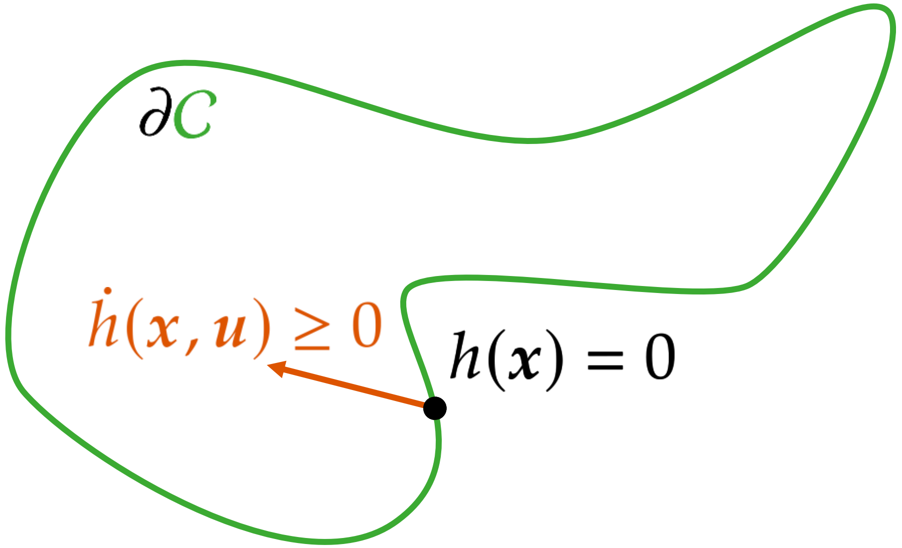
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$$\forall \boldsymbol{x} \in \partial C, \; \exists \boldsymbol{u} \in \mathcal{U}, \; L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$$



$$L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) + \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}}\left(f(\boldsymbol{x}) + g(\boldsymbol{x})\boldsymbol{u}\right) = \frac{\partial h}{\partial \boldsymbol{x}}\dot{\boldsymbol{x}} = \dot{h}(\boldsymbol{x}, \boldsymbol{u})$$

"Lie derivatives" (Notation)

$$L_f h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} f(\boldsymbol{x}) \qquad L_g h(\boldsymbol{x}) \triangleq \frac{\partial h}{\partial \boldsymbol{x}} g(\boldsymbol{x})$$

Nagumo, *Phys.-Math. Soc. Jpn.* 1942

Aubin, Frankowska, **Set-Valued Analysis** 2009

Safety and Invariance

Theorem: *Nagumo (simplified)*

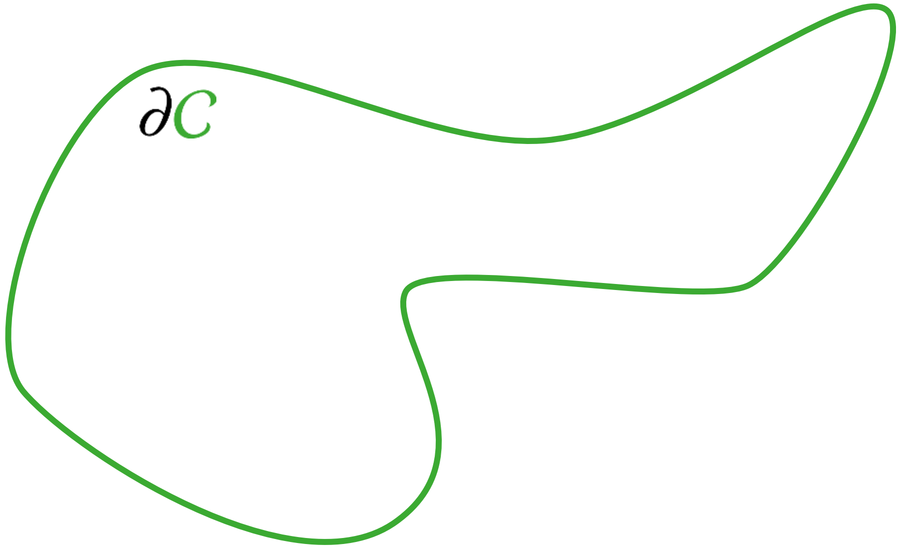
Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$\forall \boldsymbol{x} \in \partial C,\ \exists \boldsymbol{u} \in \boldsymbol{\mathcal{U}},\ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$

Measure-zero.
Difficult to check numerically.



Safety and Invariance

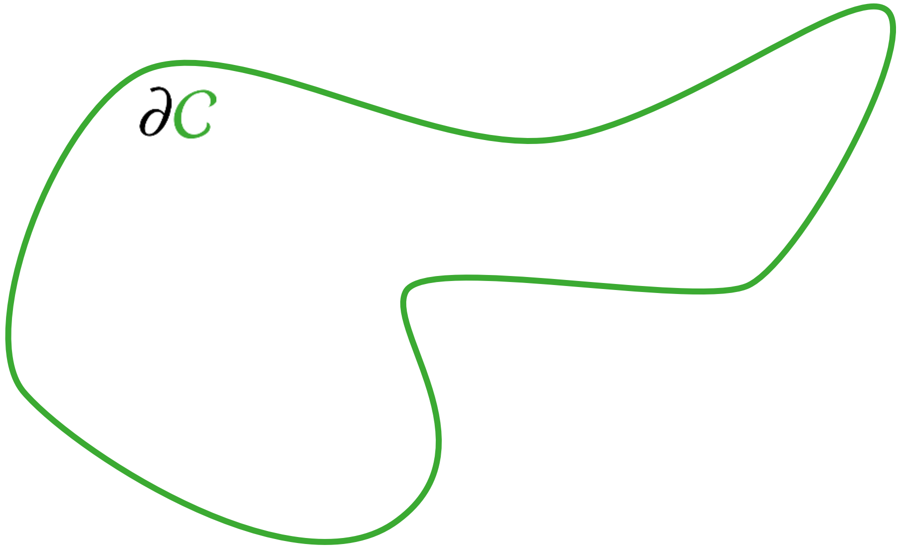
Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{x \in \mathbb{R}^n \mid h(x) \geq 0\},$$

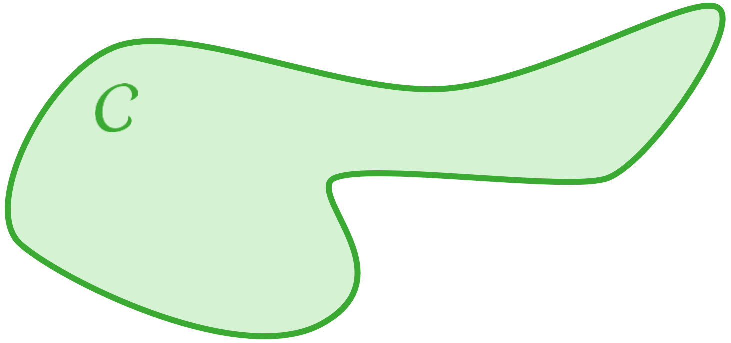
is **control invariant** if and only if

$$\forall x \in \partial C,\ \exists u \in \mathcal{U},\ L_f h(x) + L_g h(x) u \geq 0.$$



Theorem: *CBF Condition (2017)*

$$\forall x \in C,\ \exists u \in \mathcal{U},\ L_f h(x) + L_g h(x) u + \alpha\big(h(x)\big) \geq 0.$$



Index

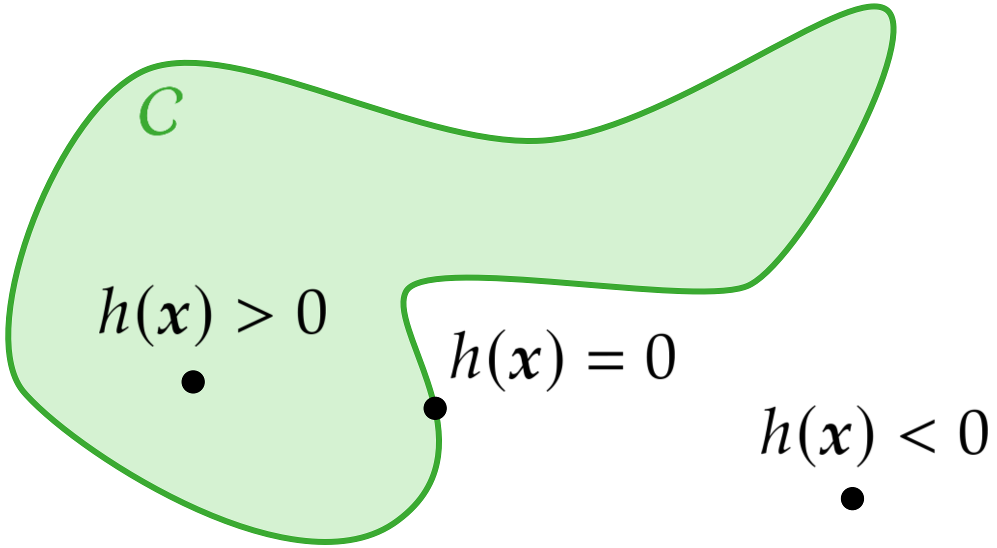
Control Barrier Functions (CBFs)

Definition: *CBF*

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \longrightarrow \mathbb{R}$ such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$



Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \longrightarrow \mathbb{R}$
such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C,\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.

Ames, Xu, Grizzle, Tabuada, *TAC* 2017

Ames, Coogan, Egerstedt, Notomista, Sreenath, Tabuada, *ECC* 2019 (survey)

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF

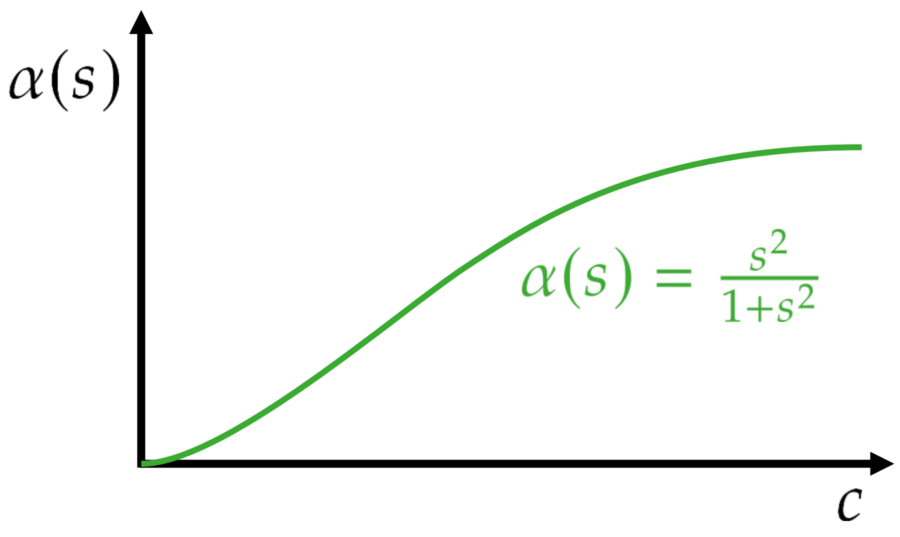
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$ such that $C = \{x \in \mathbb{R}^n \mid h(x) \geq 0\}$

CBF Condition

There exists $\alpha \in \mathcal{K}$ such that

$$\forall x \in C,\ \exists u \in \mathcal{U},\ \dot{h}(x,u) + \alpha(h(x)) \geq 0,$$

where $\dot{h}(x,u) = L_f h(x) + L_g h(x) u$.



$\alpha \in \mathcal{K}$ if

- continuous
- strictly increasing
- $\alpha(0) = 0$

Control Barrier Functions (CBFs)

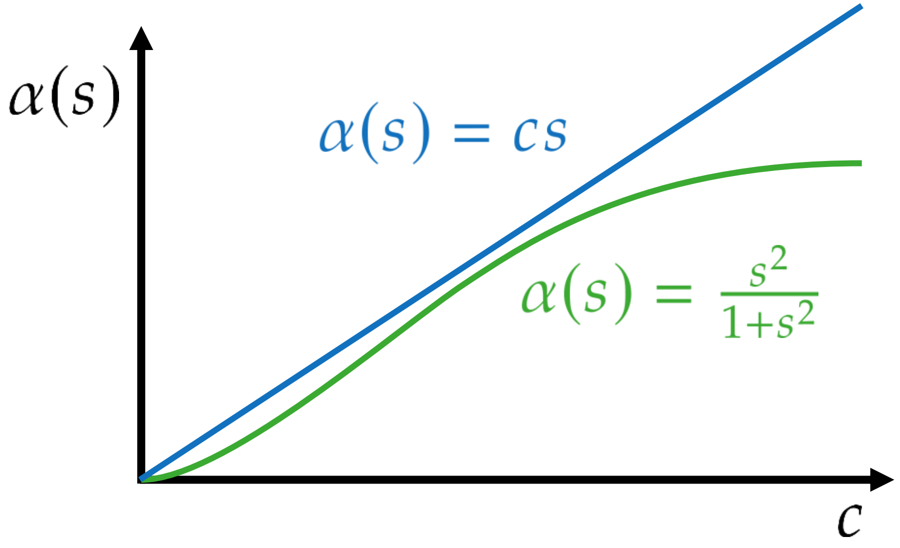
Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$
such that $C = \{x \in \mathbb{R}^n \mid h(x) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall x \in C,\ \exists u \in \mathcal{U},\ \dot{h}(x,u) + \alpha(h(x)) \geq 0,$$

where $\dot{h}(x,u) = L_f h(x) + L_g h(x) u$.



$\alpha \in \mathcal{K}$ if

- continuous
- strictly increasing
- $\alpha(0) = 0$

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF

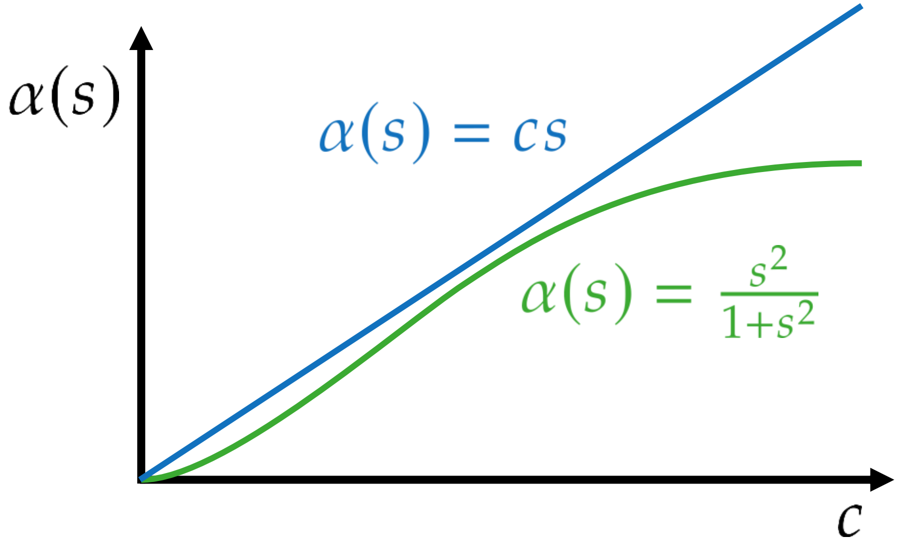
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$
such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition

There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C,\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x},\boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x},\boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.



Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF

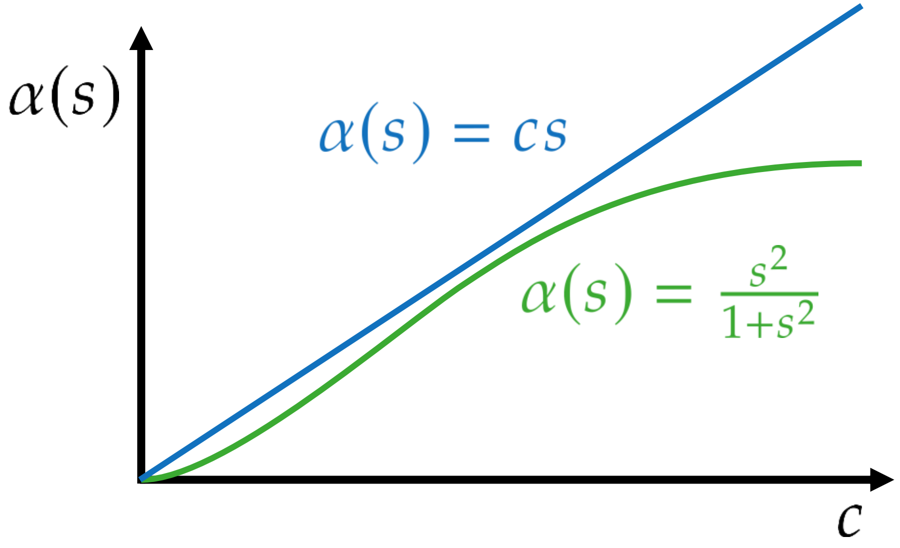
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$ such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition

There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C, \; \exists \boldsymbol{u} \in \mathcal{U}, \; \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha(h(\boldsymbol{x})) \geq 0,$$

where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.



$\boldsymbol{x} \in \partial C$

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF

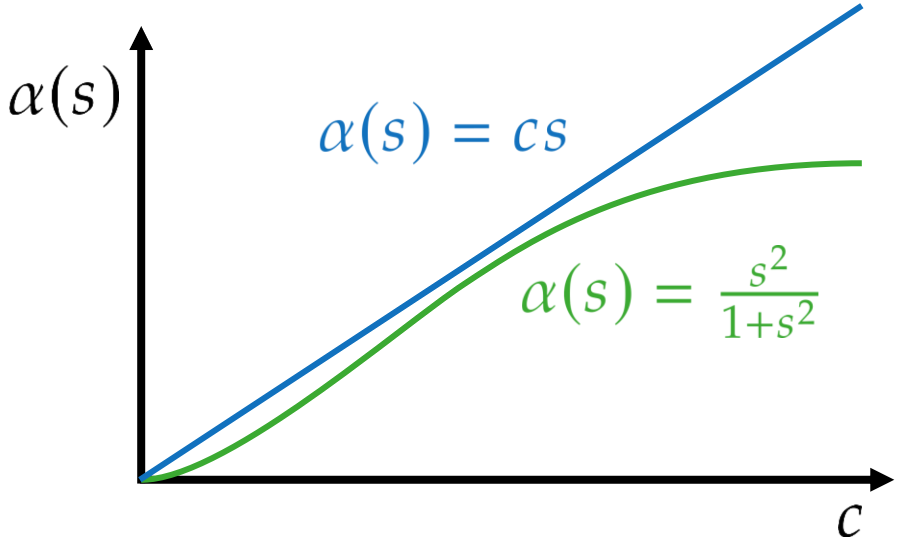
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$ such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition

There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C,\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x},\boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x},\boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.



$$\boldsymbol{x} \in \partial C \implies h(\boldsymbol{x}) = 0$$

Control Barrier Functions (CBFs)

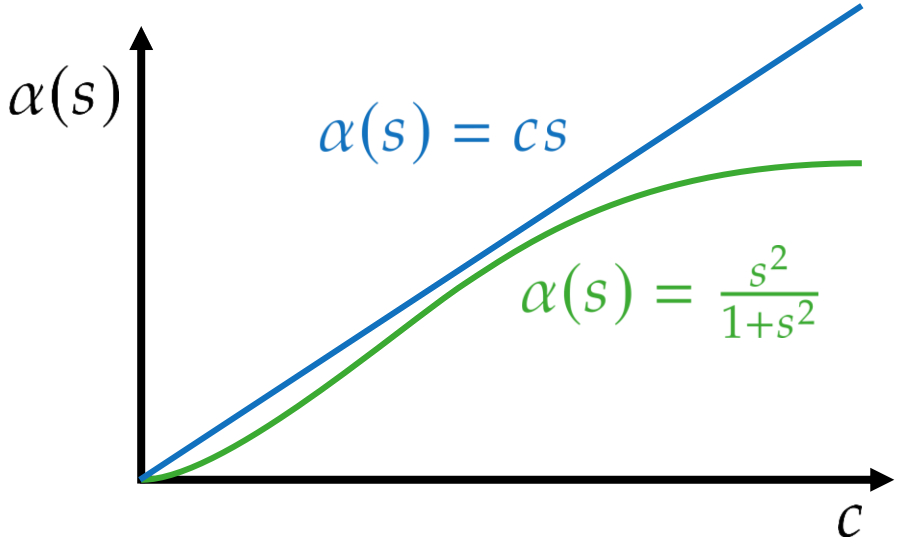
Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$
such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C,\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x},\boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x},\boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.



$$\boldsymbol{x} \in \partial C \implies h(\boldsymbol{x}) = 0 \implies \alpha\big(h(\boldsymbol{x})\big) = 0$$

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF

A smooth function $h : \mathbb{R}^n \to \mathbb{R}$ such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition

There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C,\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha(h(\boldsymbol{x})) \geq 0,$$

where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.

Theorem: *Nagumo (simplified)*

Let $h : \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable with regular value 0. The set

$$C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\},$$

is **control invariant** if and only if

$$\forall \boldsymbol{x} \in \partial C,\ \exists \boldsymbol{u} \in \mathcal{U},\ L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq 0.$$

$$\boldsymbol{x} \in \partial C \implies h(\boldsymbol{x}) = 0 \implies \alpha\big(h(\boldsymbol{x})\big) = 0$$

Ames, Xu, Grizzle, Tabuada, *TAC* 2017

Ames, Coogan, Egerstedt, Notomista, Sreenath, Tabuada, *ECC* 2019 (survey)

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$ such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C,\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.

Theorem: *CBF Condition (2017)*

If h is a CBF, then C is control invariant.

Ames, Xu, Grizzle, Tabuada, *TAC* 2017
Ames, Coogan, Egerstedt, Notomista, Sreenath, Tabuada, *ECC* 2019 (survey)

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$ such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C, \; \exists \boldsymbol{u} \in \mathcal{U}, \; \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.

Theorem: *CBF Condition (2017)*

If h is a CBF, then C is control invariant.

What about the other direction?
If C is control invariant, then h is a CBF.

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$ such that $C = \{x \in \mathbb{R}^n \mid h(x) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall x \in C,\ \exists u \in \mathcal{U},\ \dot{h}(x,u) + \alpha\big(h(x)\big) \geq 0,$$

where $\dot{h}(x,u) = L_f h(x) + L_g h(x) u$.

Theorem: *CBF Condition (2017)*

If h is a CBF, then C is control invariant.

Theorem: *CBF Condition (2017)*

Assume (a) C is compact and (b) there exists a continuous selection $\mu : C \to \mathcal{U}$ such that $\mu(x) \in \mathcal{K}(x)$.
If C is control invariant, then h is a CBF.

$$\mathcal{K}(x) = \{u \in \mathcal{U} \mid L_f h(x) + L_g h(x) u \geq 0\}$$

Control Barrier Functions (CBFs)

Definition: CBF

Candidate CBF
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$
such that $\mathcal{C} = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in \mathcal{C},\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.

Problem: Constrained Control

Given a control-affine system
$$\dot{x}(t) = f\big(x(t)\big) + g\big(x(t)\big)u(t)$$
and constraints
$$\mathcal{X} \subset \mathbb{R}^n,\ \mathcal{U} \subset \mathbb{R}^m,$$
find a signal $u : [0, \infty) \to \mathcal{U}$
such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Solution exists if there exists $\mathcal{C}$ **control invariant** and **admissible** (i.e., **safe**).

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$
such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C,\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.

Problem: *Constrained Control*

Given a control-affine system
$$\dot{x}(t) = f\big(x(t)\big) + g\big(x(t)\big)u(t)$$
and constraints
$$\mathcal{X} \subset \mathbb{R}^n,\ \mathcal{U} \subset \mathbb{R}^m,$$
find a signal $u : [0, \infty) \to \mathcal{U}$
such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Solution exists if there exists a CBF h
such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\} \subset \mathcal{X}$.

Control Barrier Functions (CBFs)

Definition: *CBF*

Candidate CBF
A smooth function $h : \mathbb{R}^n \to \mathbb{R}$
such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition
There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C,\ \exists \boldsymbol{u} \in \mathcal{U},\ \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.

Problem: *Constrained Control*

Given a control-affine system

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

and constraints

$$\mathcal{X} \subset \mathbb{R}^n,\ \mathcal{U} \subset \mathbb{R}^m,$$

find a signal $u : [0, \infty) \to \mathcal{U}$
such that $x(t) \in \mathcal{X}$ for all $t \geq 0$.

Solution exists if there exists a CBF h
such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\} \subset \mathcal{X}$.

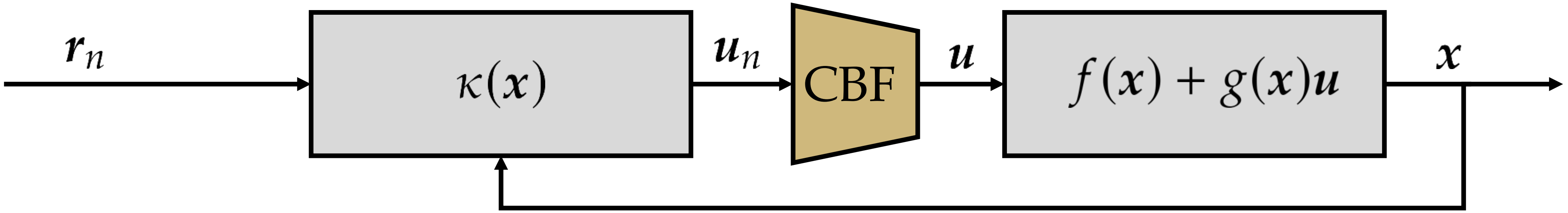
Given a CBF h, how to find a solution $u(t)$?

Index

Safety Filters

Family of constrained controllers that use a two-step approach:

1. Design a control law for the **Unconstrained** system,
2. Introduce an **Add-on** unit for constraint enforcement.
   - **Control Barrier Function:** Filter nominal controller



CBF-based Safety Filter

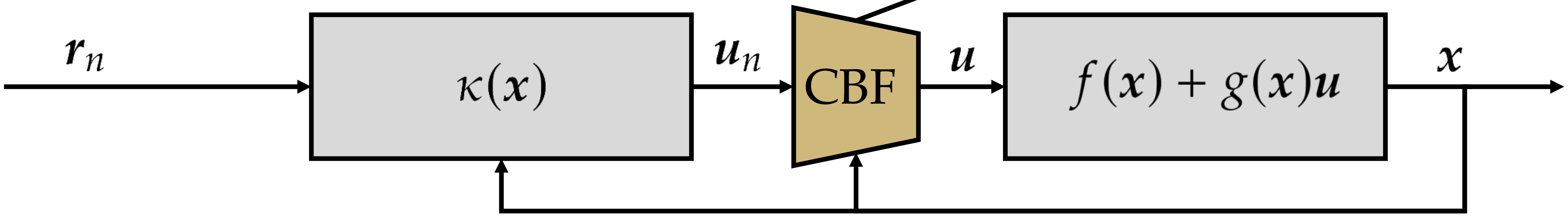
Safety Filters

Family of constrained controllers that use a two-step approach:

1. Design a control law for the **Unconstrained** system,
2. Introduce an **Add-on** unit for constraint enforcement.
   - **Control Barrier Function:** Filter nominal controller

CBF-based Safety Filter

$$\min_{\boldsymbol{u}\in\mathcal{U}} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad \dot{h}(\boldsymbol{x},\boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$$



CBF-based Safety Filter

$$\begin{aligned} \min_{\boldsymbol{u} \in \mathcal{U}} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big) \end{aligned}$$

CBF-based Safety Filter

Optimization Problem (Program)

$$\min_{u \in \mathcal{U}} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$$

=

Find the vector $\boldsymbol{u} \in \mathcal{U}$ that minimizes $\|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$ subject to $L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$.

CBF-based Safety Filter

Optimization Problem (Program)

$$\min_{u \in \mathcal{U}} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$$

Given $\boldsymbol{x}$, these terms are constant.

=

Find the vector $\boldsymbol{u} \in \mathcal{U}$ that minimizes $\|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$ subject to $L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$.

CBF-based Safety Filter

Optimization Problem (Program)

$$\min_{\boldsymbol{u}\in\mathcal{U}} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$$

=

Find the vector $\boldsymbol{u} \in \mathcal{U}$ that minimizes $\|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$ subject to $L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$.

Given $\boldsymbol{x}$, these terms are constant.

If $\mathcal{U}$ is polyhedral $\mathcal{U} = \{\boldsymbol{u} \mid M\boldsymbol{u} \leq \boldsymbol{b}\}$,
then the safety filter is a **quadratic program** (QP).

CBF-based Safety Filter

Optimization Problem (Program)

$$\min_{\boldsymbol{u}\in\mathcal{U}} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$$

=

Find the vector $\boldsymbol{u} \in \mathcal{U}$ that minimizes $\|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$ subject to $L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$.

Given $\boldsymbol{x}$, these terms are constant.

If $\mathcal{U}$ is polyhedral $\mathcal{U} = \{\boldsymbol{u} \mid M\boldsymbol{u} \leq \boldsymbol{b}\}$,
then the safety filter is a **quadratic program** (QP).

Commercial QP Solvers

mosek

GUROBI OPTIMIZATION

COPT Cardinal Optimizer

IBM CPLEX

CBF-based Safety Filter

Optimization Problem

$$\min_{u \in \mathcal{U}} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$$

Is there a solution?

(feasibility)

CBF-based Safety Filter

Optimization Problem

$$\begin{aligned} \min_{u \in \mathcal{U}} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big) \end{aligned}$$

Theorem: *CBF Safety Filter*

If h is a CBF, then the CBF-based safety filter is feasible for all $\boldsymbol{x} \in \mathcal{C}$.

CBF-based Safety Filter

Optimization Problem

$$\min_{u \in \mathcal{U}} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$$

Theorem: *CBF Safety Filter*

If h is a CBF, then the CBF-based safety filter is feasible for all $\boldsymbol{x} \in C$.

Definition: *CBF*

Candidate CBF

A smooth function $h : \mathbb{R}^n \to \mathbb{R}$ such that $C = \{\boldsymbol{x} \in \mathbb{R}^n \mid h(\boldsymbol{x}) \geq 0\}$

CBF Condition

There exists $\alpha \in \mathcal{K}$ such that

$$\forall \boldsymbol{x} \in C, \ \exists \boldsymbol{u} \in \mathcal{U}, \ \dot{h}(\boldsymbol{x}, \boldsymbol{u}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0,$$

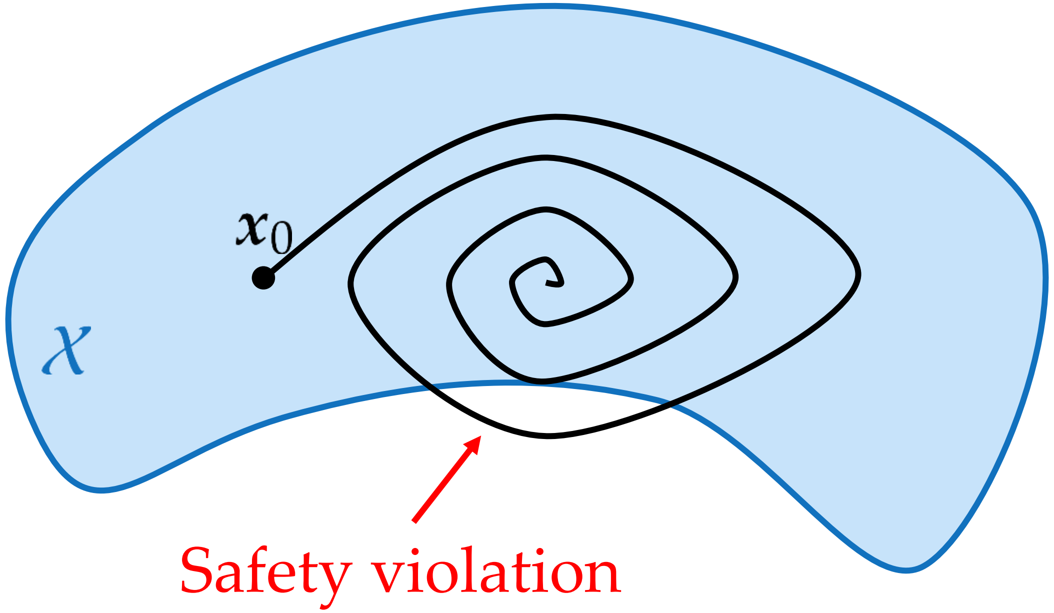
where $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$.

Ames, Xu, Grizzle, Tabuada, *TAC* 2017

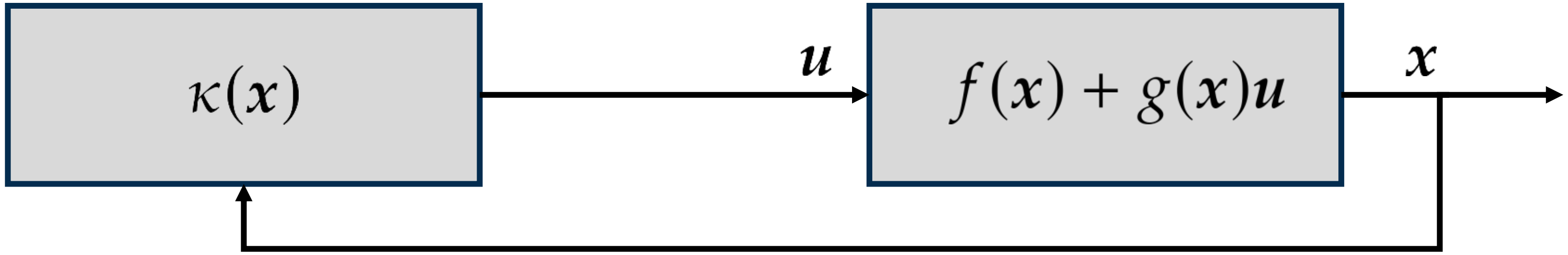
CBF-based Safety Filter

Safety Filters

1. Design a control law for the **Unconstrained** system,



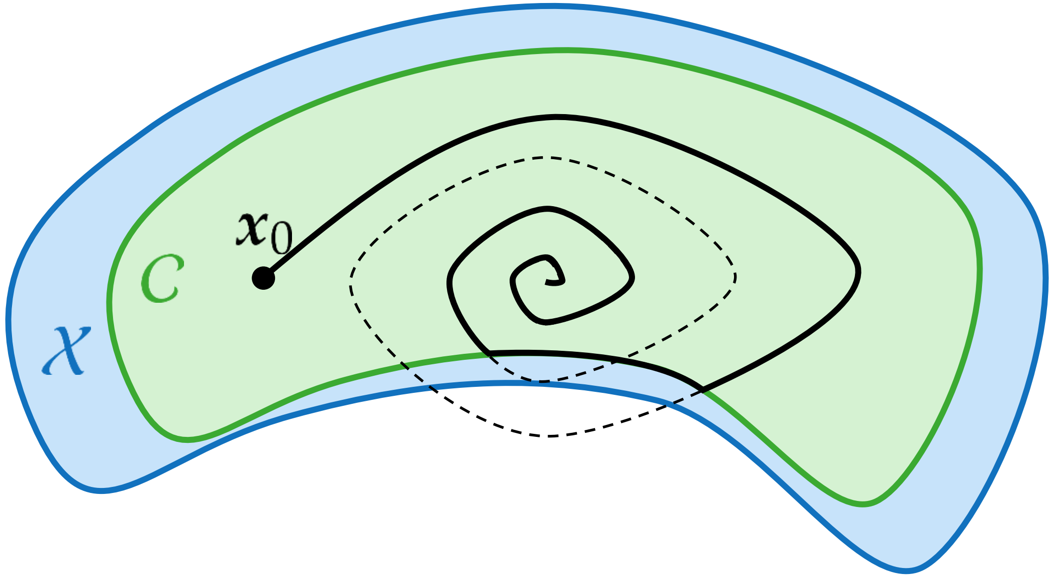
Unsafe



CBF-based Safety Filter

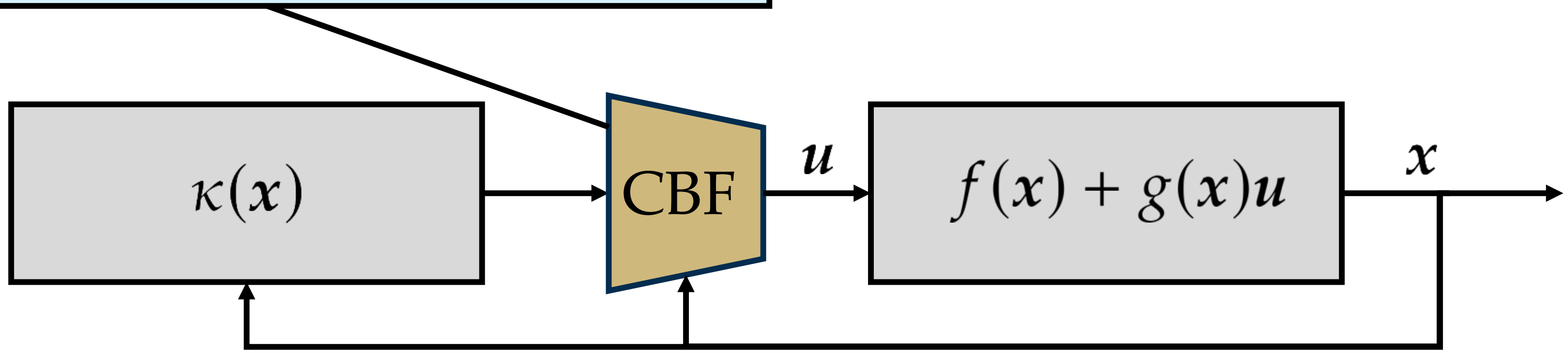
Safety Filters

1. Design a control law for the **Unconstrained** system,
2. Introduce an **Add-on** unit for constraint enforcement.



$$\begin{aligned} \min_{\boldsymbol{u}\in\mathcal{U}} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big) \end{aligned}$$

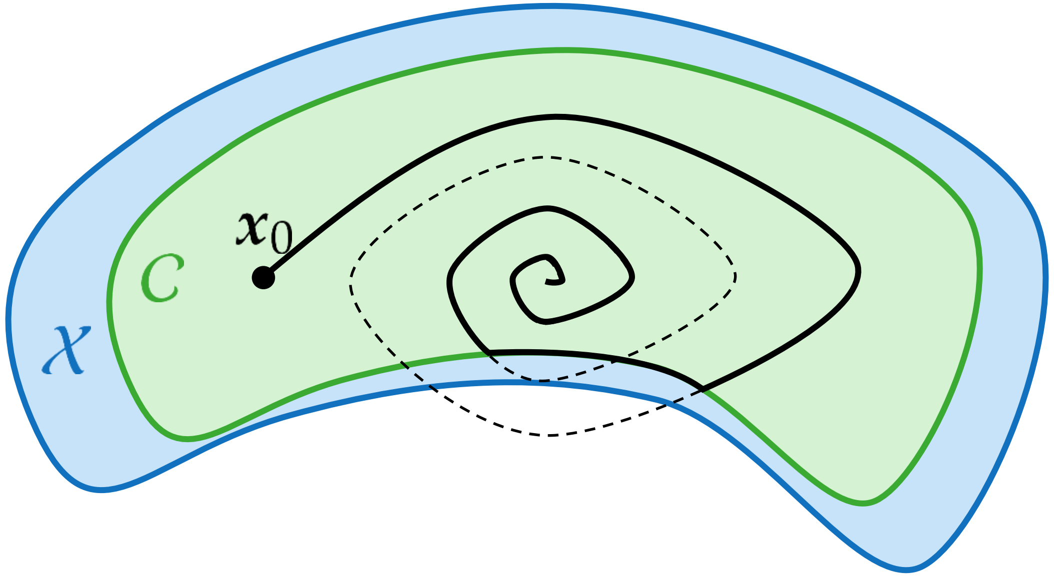
Safe



CBF-based Safety Filter

Safety Filters

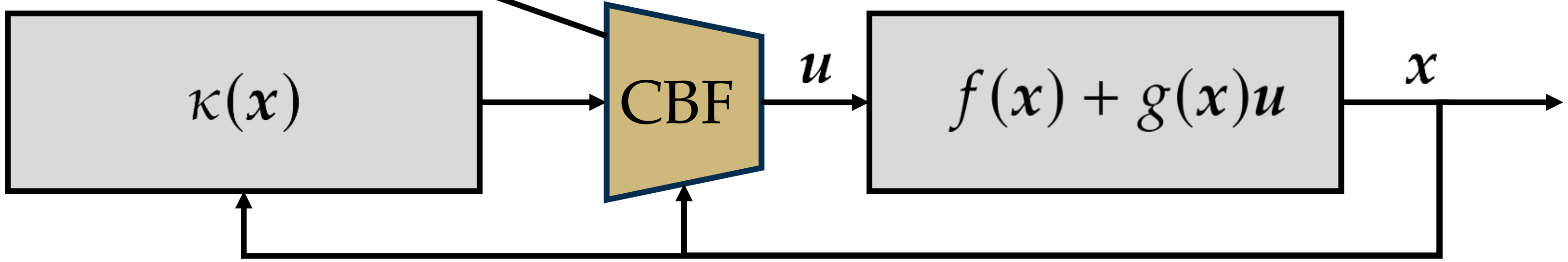
1. Design a control law for the **Unconstrained** system,
2. Introduce an **Add-on** unit for constraint enforcement.



$$\min_{\boldsymbol{u}\in\mathcal{U}} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u} \geq -\alpha\big(h(\boldsymbol{x})\big)$$

How to find a CBF $h(\boldsymbol{x})$?

Safe



Index

Example: Safety Filter for F-16

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix}\theta & \dot{\theta} & \alpha_w\end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix}\delta_e & \delta_f\end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -0.87 & 43.22 \\ 0 & 0.99 & -1.34 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ -17.25 & -1.58 \\ -0.17 & -0.25 \end{bmatrix}$$

θ pitch $\quad \alpha_w$ angle of attack $\quad \delta_e, \delta_f$ control surfaces

Example: Safety Filter for F-16

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

Actuator Inner-loop (first-order lag)

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

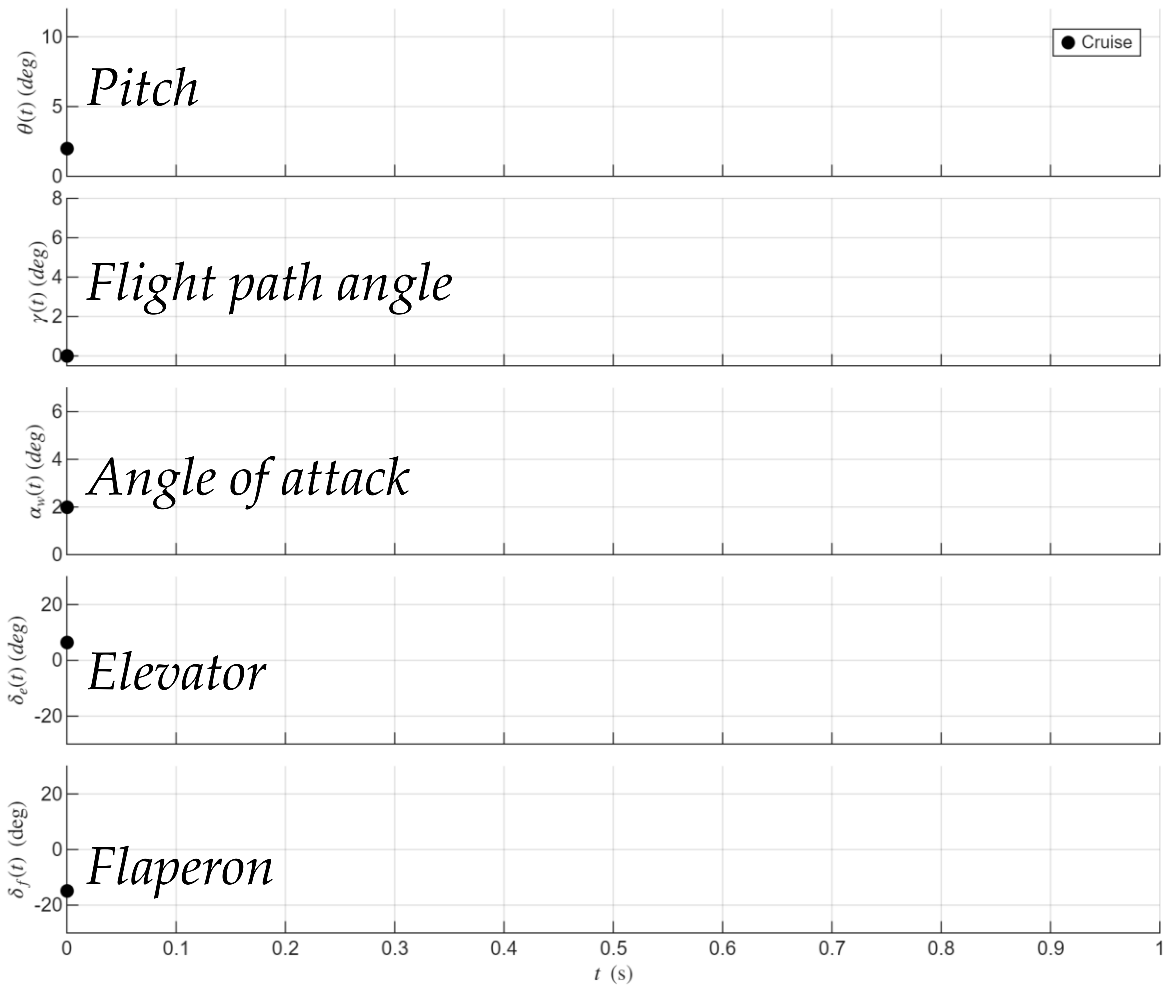
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -0.87 & 43.22 \\ 0 & 0.99 & -1.34 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ -17.25 & -1.58 \\ -0.17 & -0.25 \end{bmatrix}$$

θ pitch $\quad \alpha_w$ angle of attack $\quad \delta_e, \delta_f$ control surfaces

Example: Safety Filter for F-16



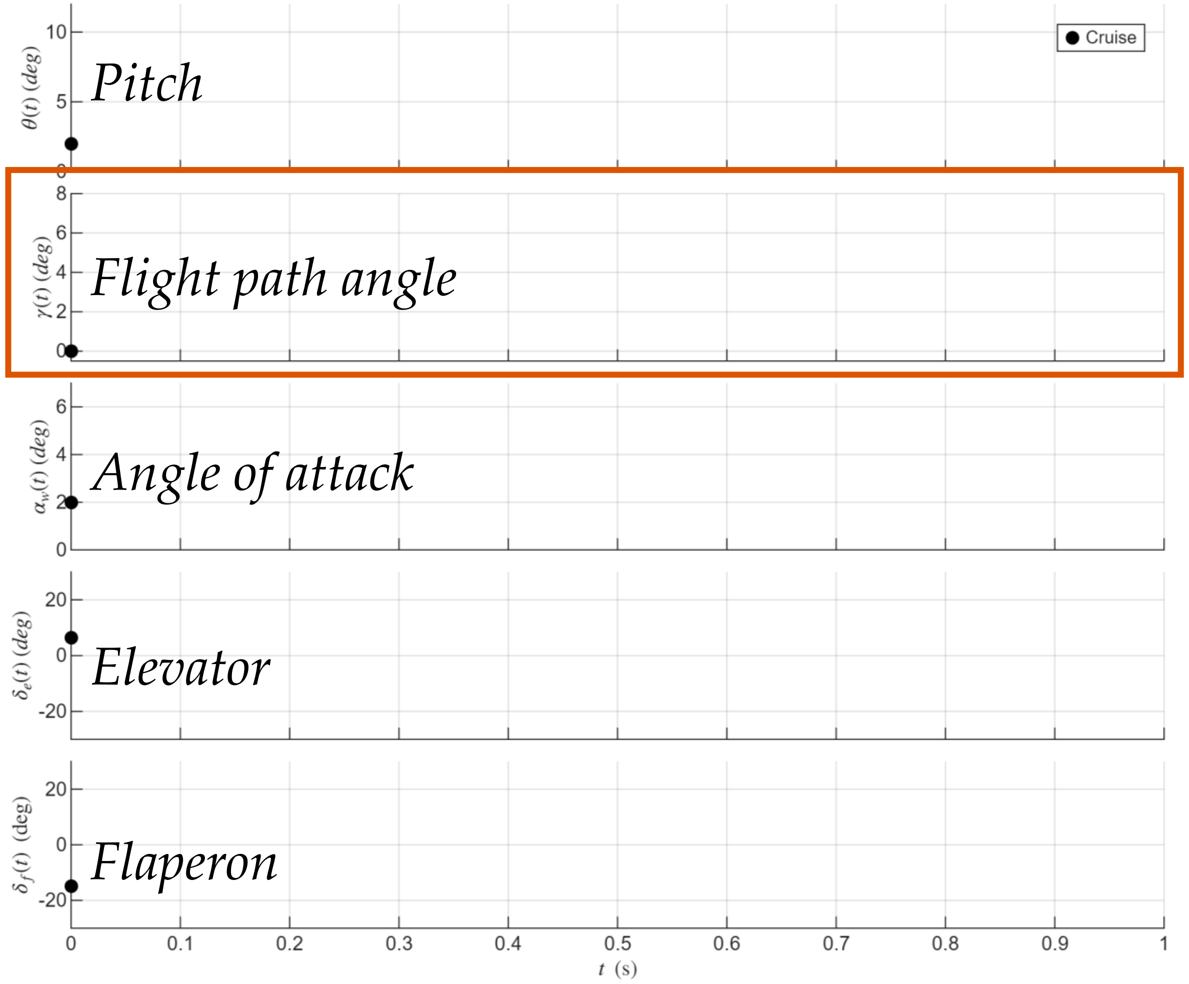
F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Cruise: $\theta = 2,\ \gamma = 0$

Example: Safety Filter for F-16



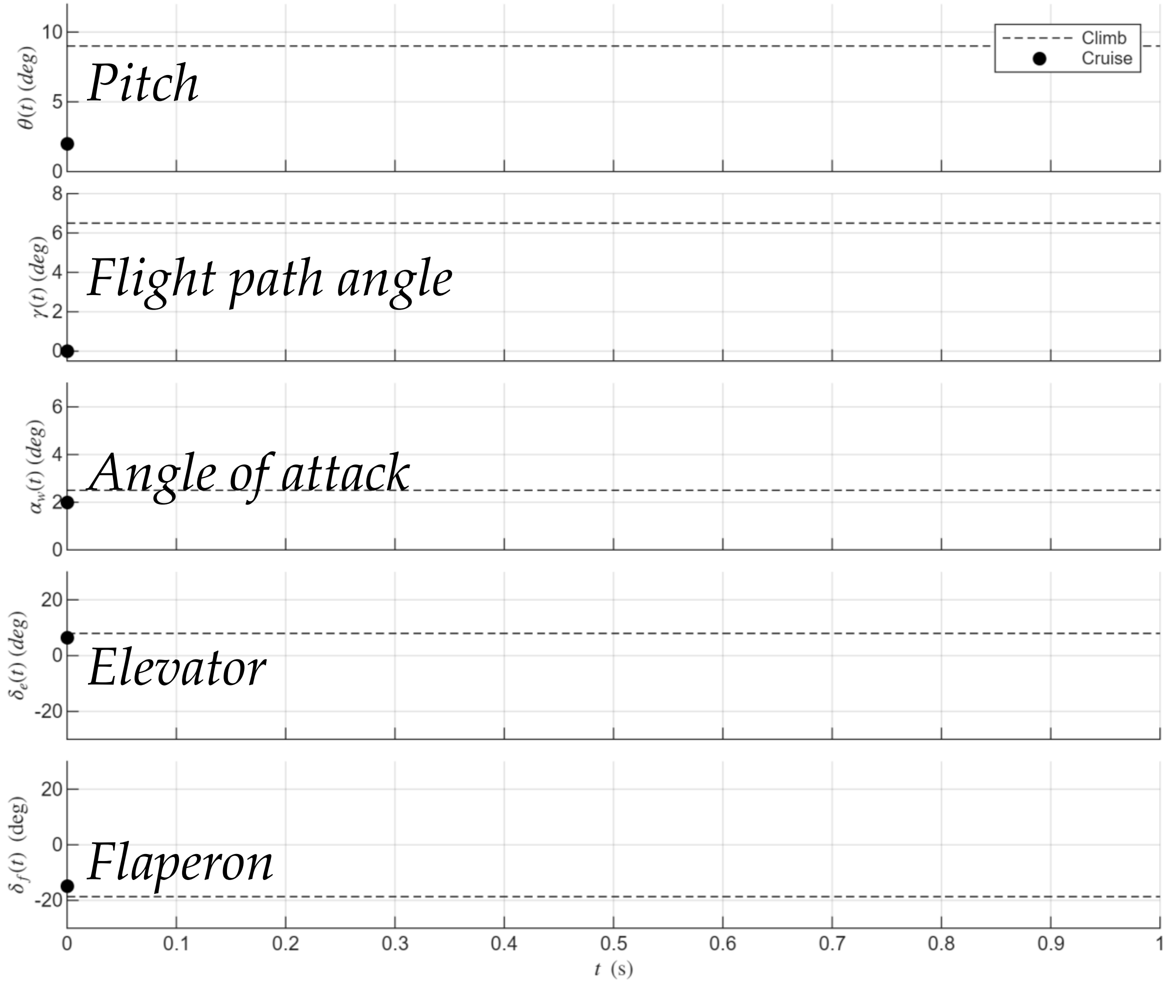
F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Cruise: $\theta = 2$, $\gamma = 0$ $\quad \gamma \approx \theta - \alpha_w$

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

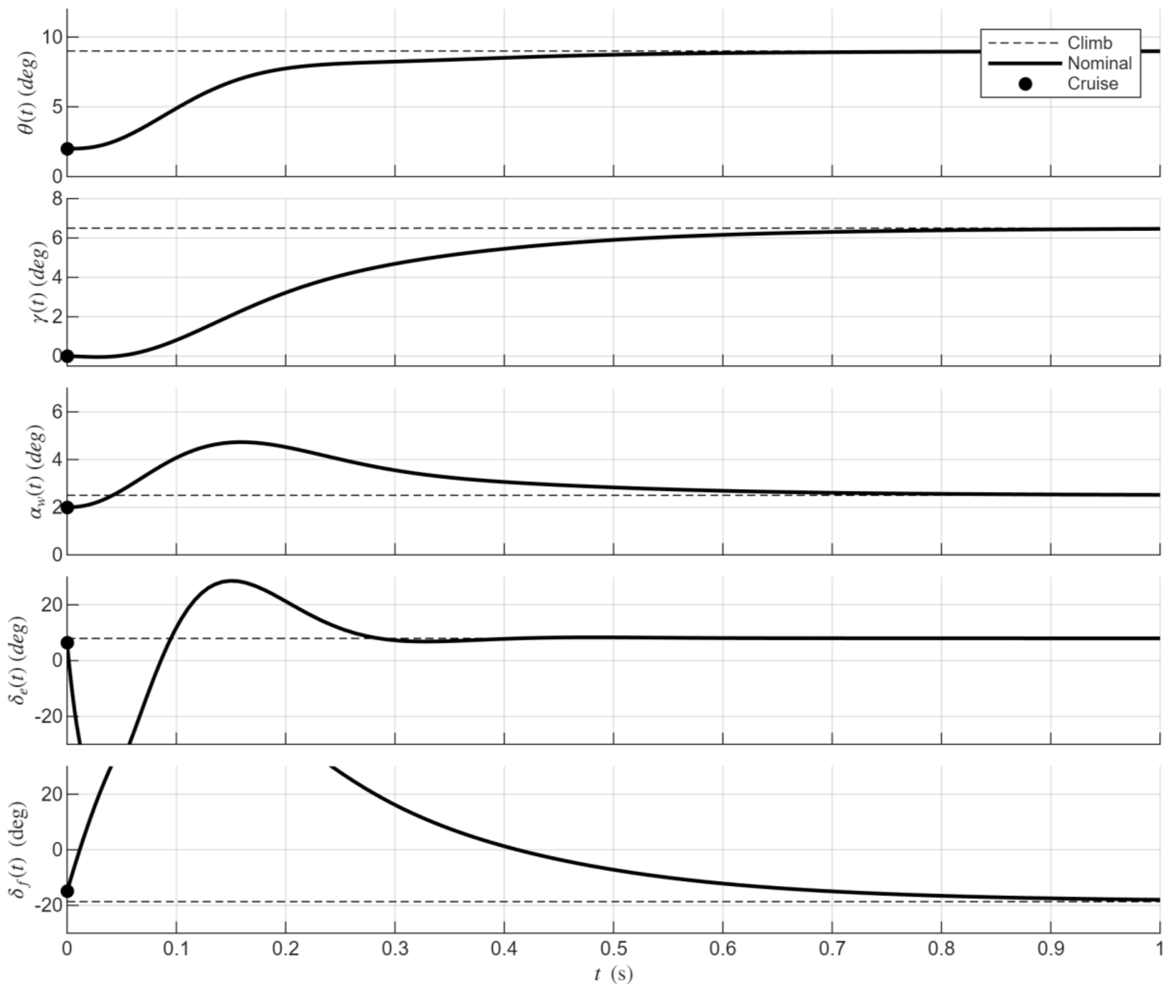
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Cruise: $\theta = 2,\ \gamma = 0$

↓ Goal

Climb: $\theta = 9,\ \gamma = 6.5$

Example: Safety Filter for F-16



F-16 Pitch Dynamics

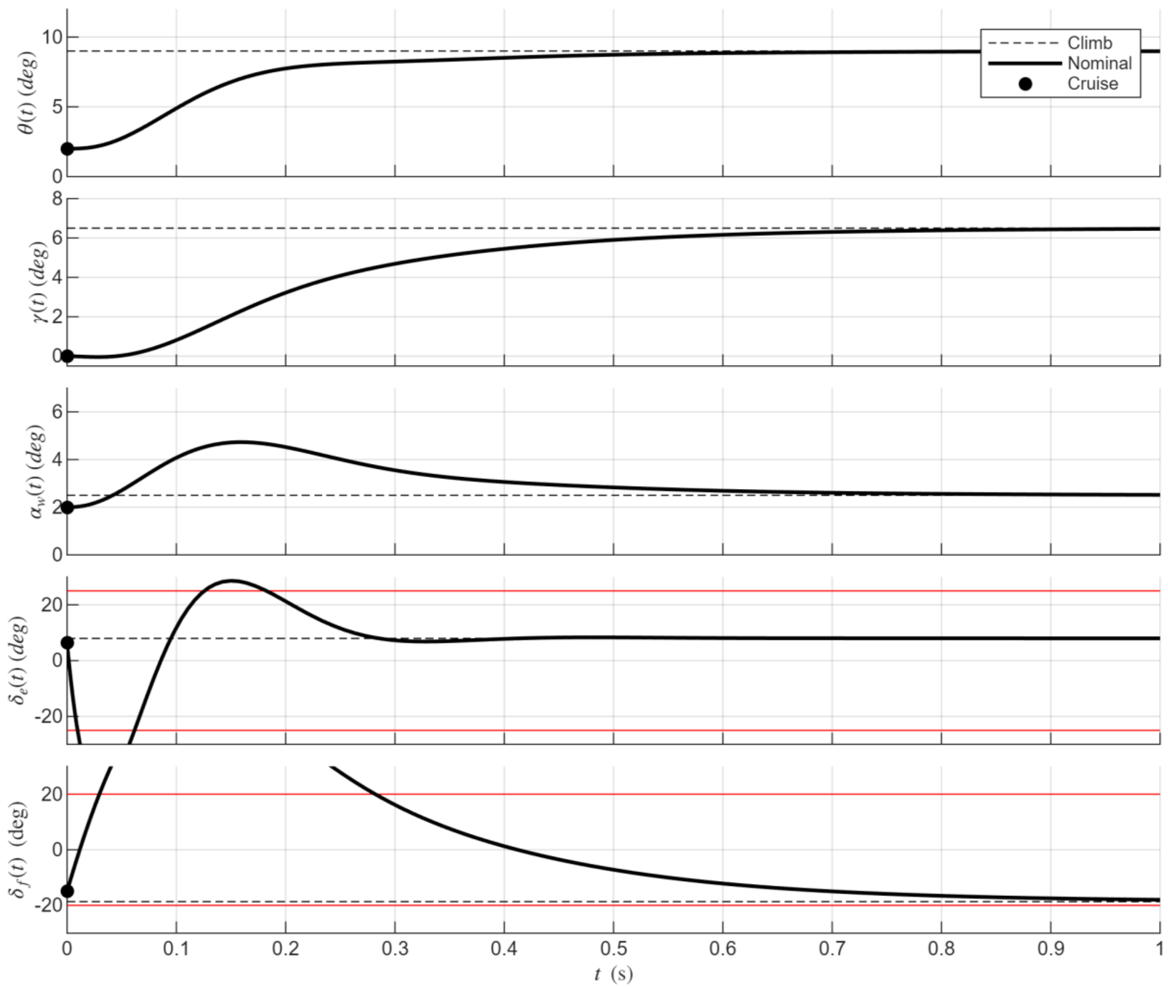
$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

LQR Controller $\kappa(\boldsymbol{x}) = \bar{u}(\boldsymbol{r}) - K(\boldsymbol{x} - \bar{x}(\boldsymbol{r}))$

Desired **Climb** operating conditions

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

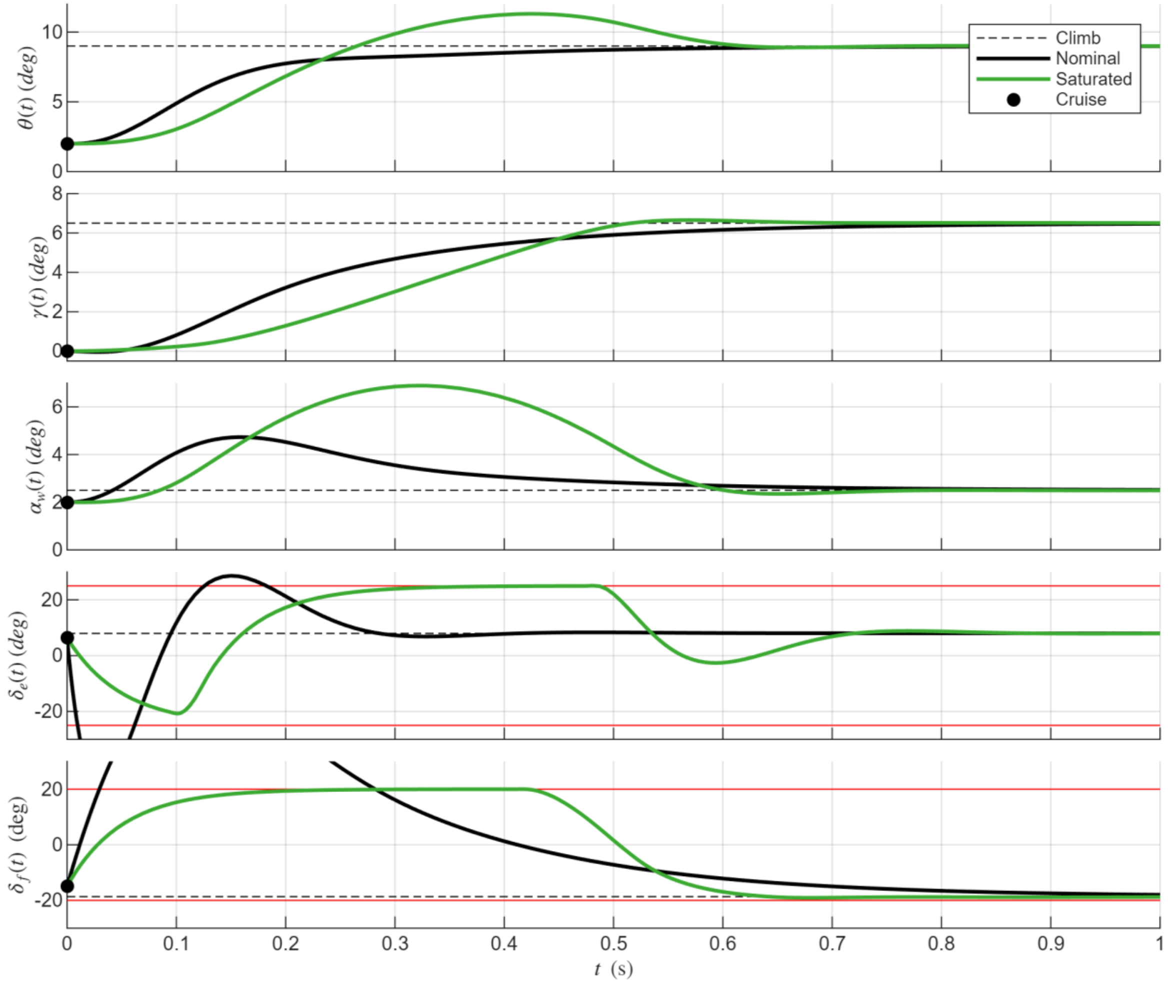
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

LQR Controller $\kappa(\boldsymbol{x}) = \bar{u}(\boldsymbol{r}) - K(\boldsymbol{x} - \bar{x}(\boldsymbol{r}))$

Control surface deflection limits

$|\delta_e| \leq 25$ deg, $|\delta_f| \leq 20$ deg

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

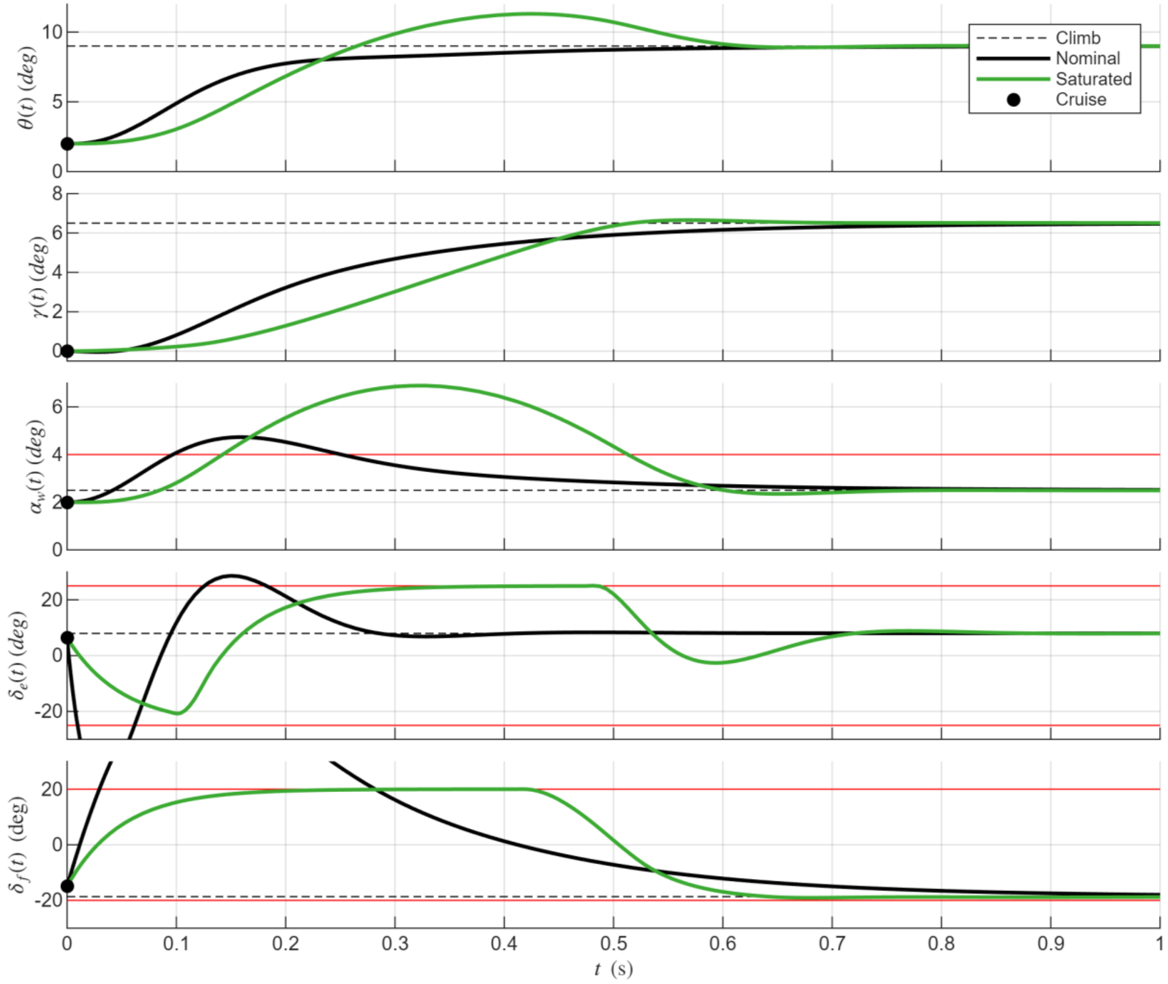
LQR Controller $\kappa(\boldsymbol{x}) = \bar{u}(\boldsymbol{r}) - K(\boldsymbol{x} - \bar{x}(\boldsymbol{r}))$

Saturated LQR $\boldsymbol{u} = \mathrm{sat}(\kappa(\boldsymbol{x}))$

Control surface deflection limits

$|\delta_e| \leq 25$ deg, $|\delta_f| \leq 20$ deg

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix}\theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f\end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix}\delta_{e,c} & \delta_{f,c}\end{bmatrix}$$

LQR Controller $\kappa(\boldsymbol{x}) = \bar{u}(\boldsymbol{r}) - K(\boldsymbol{x} - \bar{x}(\boldsymbol{r}))$

Saturated LQR $\boldsymbol{u} = \mathrm{sat}\big(\kappa(\boldsymbol{x})\big)$

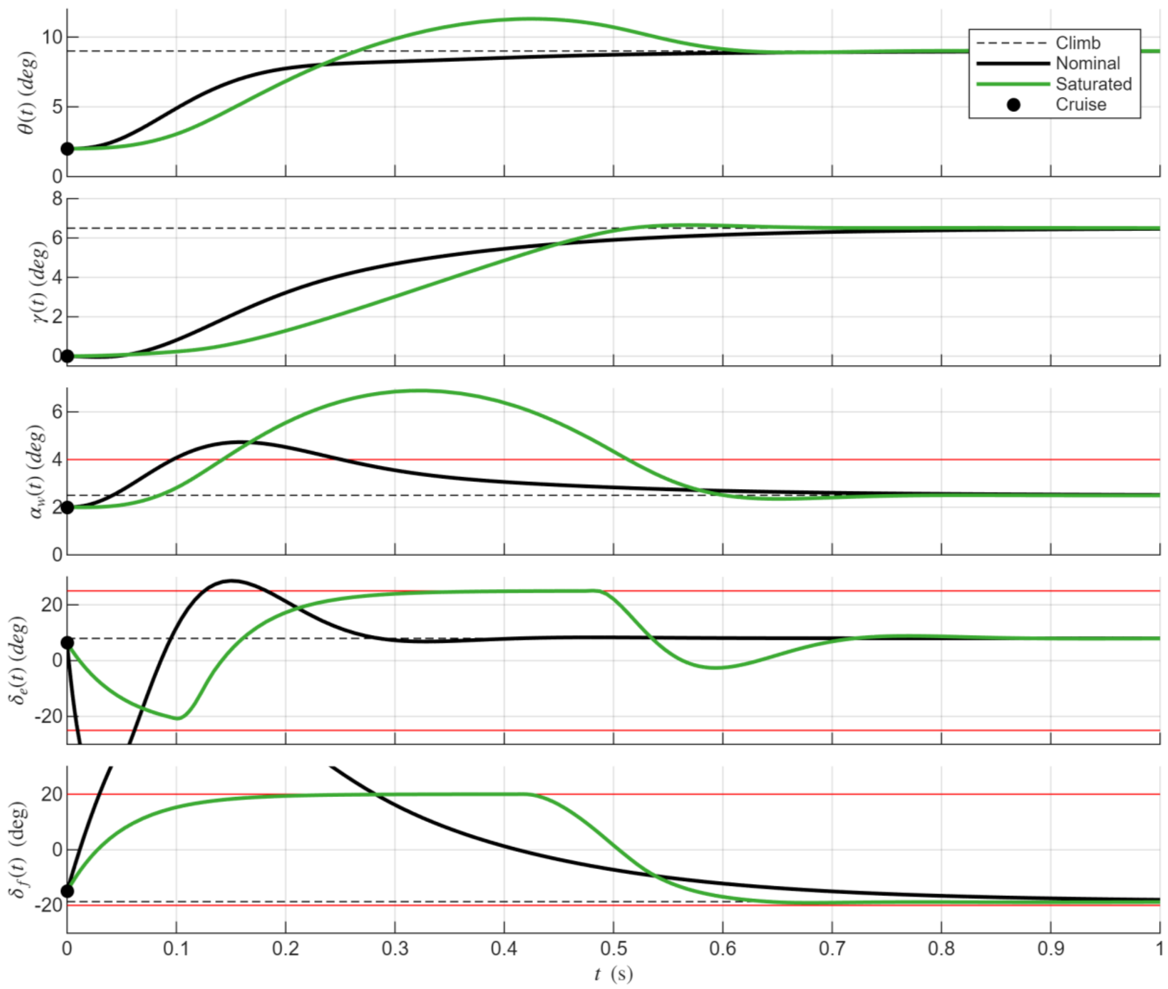
Pitch pointing constraint

$|\alpha_w| \leq 25$ deg

Control surface deflection limits

$|\delta_e| \leq 25$ deg, $|\delta_f| \leq 20$ deg

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

LQR Controller $\kappa(\boldsymbol{x}) = \bar{u}(\boldsymbol{r}) - K(\boldsymbol{x} - \bar{x}(\boldsymbol{r}))$

Saturated LQR $\boldsymbol{u} = \mathrm{sat}(\kappa(\boldsymbol{x}))$

Saturation is not good enough.
Let's try CBFs

Example: Safety Filter for F-16

Naïve CBF Design (elevator)

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$25 \text{ deg} \geq \delta_\mathrm{e}$$

Example: Safety Filter for F-16

Naïve CBF Design (elevator)

Elevator constraint: $h(\boldsymbol{x}) = 25 - \delta_e \geq 0$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$25 \text{ deg} \geq \delta_{\mathrm{e}}$

Example: Safety Filter for F-16

Naïve CBF Design (elevator)

Elevator constraint: $h(\boldsymbol{x}) = 25 - \delta_e \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 \end{bmatrix}$

⚠ This procedure yields ***candidate CBFs***. They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (elevator)

Elevator constraint: $h(\boldsymbol{x}) = 25 - \delta_e \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} A\boldsymbol{x}$$

$$L_g h(\boldsymbol{x})\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}} B\boldsymbol{u}$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$

$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$

$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (elevator)

Elevator constraint: $h(\boldsymbol{x}) = 25 - \delta_e \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} A\boldsymbol{x} = 20\delta_e$$

$$L_g h(\boldsymbol{x})\boldsymbol{u} = \frac{\partial h}{\partial \boldsymbol{x}} B\boldsymbol{u} = -20\delta_{e,c}$$

⚠ This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$

$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$

$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (elevator)

Elevator constraint: $h(\boldsymbol{x}) = 25 - \delta_e \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = 20(\delta_e - \delta_{e,c})$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (elevator)

Elevator constraint: $h(\boldsymbol{x}) = 25 - \delta_e \geq 0$

Differentiate: $\frac{\partial h}{\partial \boldsymbol{x}} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = 20(\delta_e - \delta_{e,c})$

Enforce: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\begin{aligned} \min_{\boldsymbol{u} \in \mathbb{R}^2} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & \underbrace{20(\delta_e - \delta_{e,c})}_{\dot{h}(\boldsymbol{x}, \boldsymbol{u})} \geq -\alpha \underbrace{(25 - \delta_e)}_{h(\boldsymbol{x})} \end{aligned}$$

Example: Safety Filter for F-16

Naïve CBF Design (elevator)

Elevator constraint: $h(\boldsymbol{x}) = 25 - \delta_e \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = 20(\delta_e - \delta_{e,c})$

Enforce: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Tune: $\alpha > 0$

Safety ------------------------------ Performance

$\alpha = 0.1$ $\alpha = 100$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

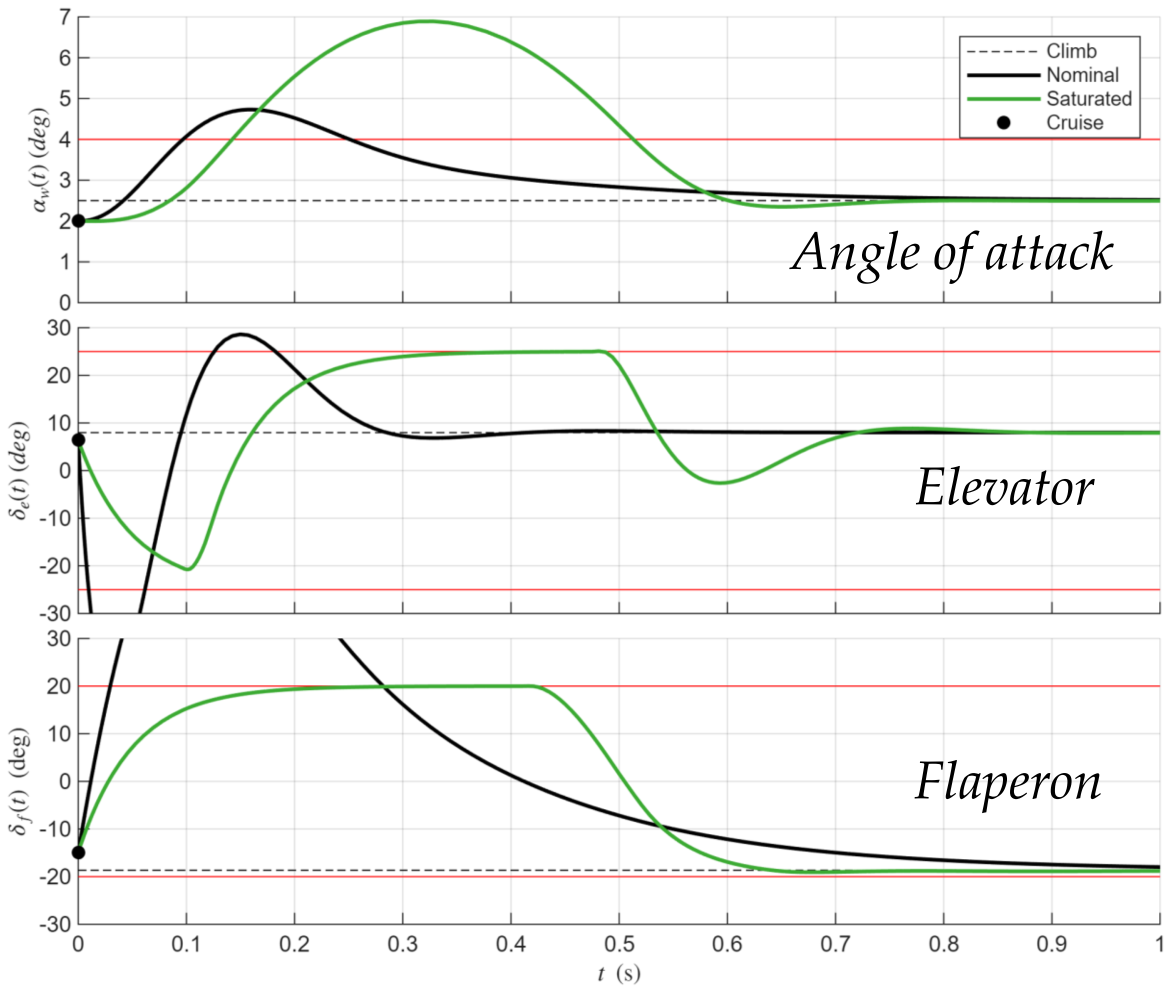
$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\min_{\boldsymbol{u} \in \mathbb{R}^2} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$

$$\text{s.t.} \quad \underbrace{20(\delta_e - \delta_{e,c})}_{\dot{h}(\boldsymbol{x}, \boldsymbol{u})} \geq -\alpha \underbrace{(25 - \delta_e)}_{h(\boldsymbol{x})}$$

Example: Safety Filter for F-16



F-16 Pitch Dynamics

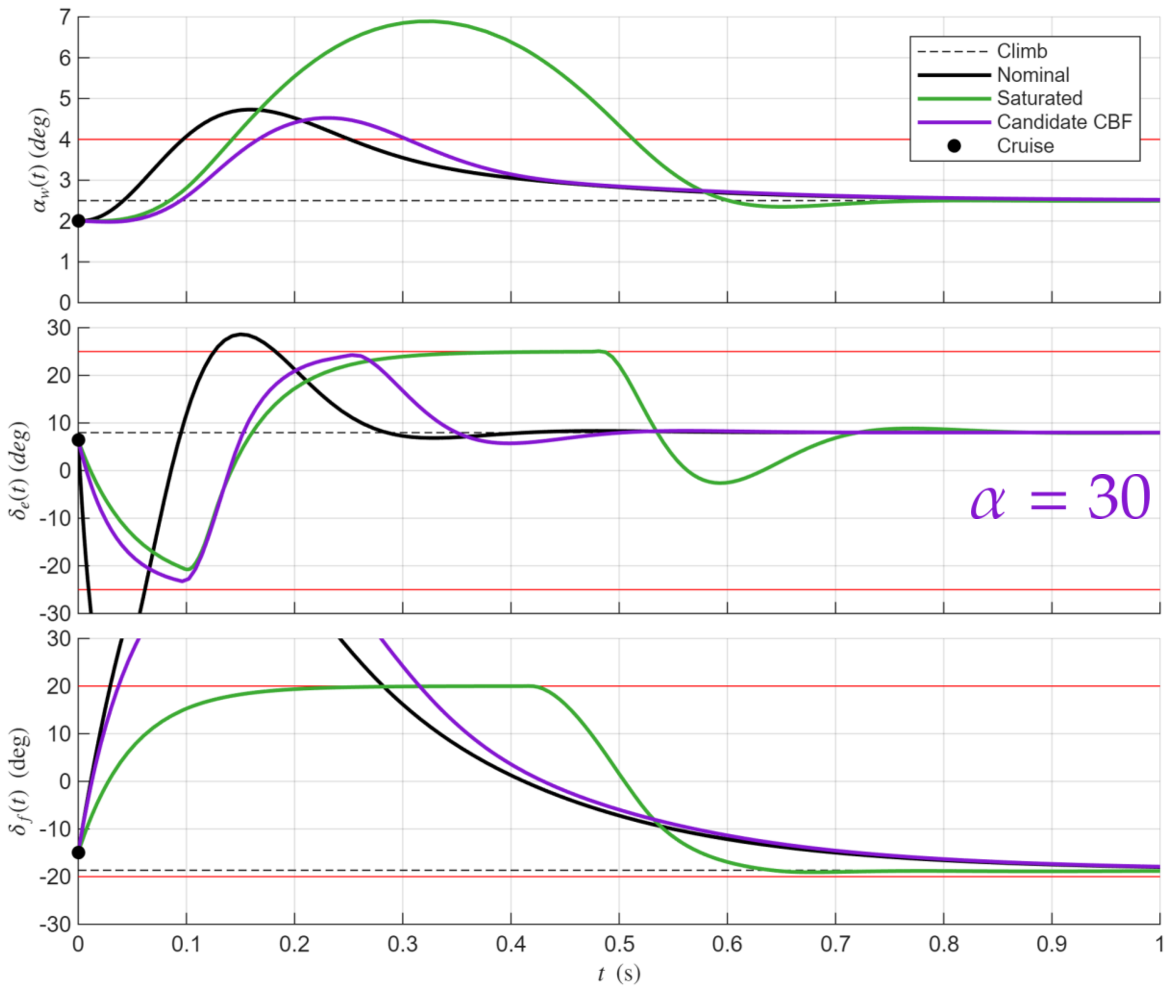
$$x = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$u = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

LQR Controller $\kappa(x) = \bar{u}(r) - K(x - \bar{x}(r))$

Saturated LQR $u = \mathrm{sat}\big(\kappa(x)\big)$

Example: Safety Filter for F-16



F-16 Pitch Dynamics

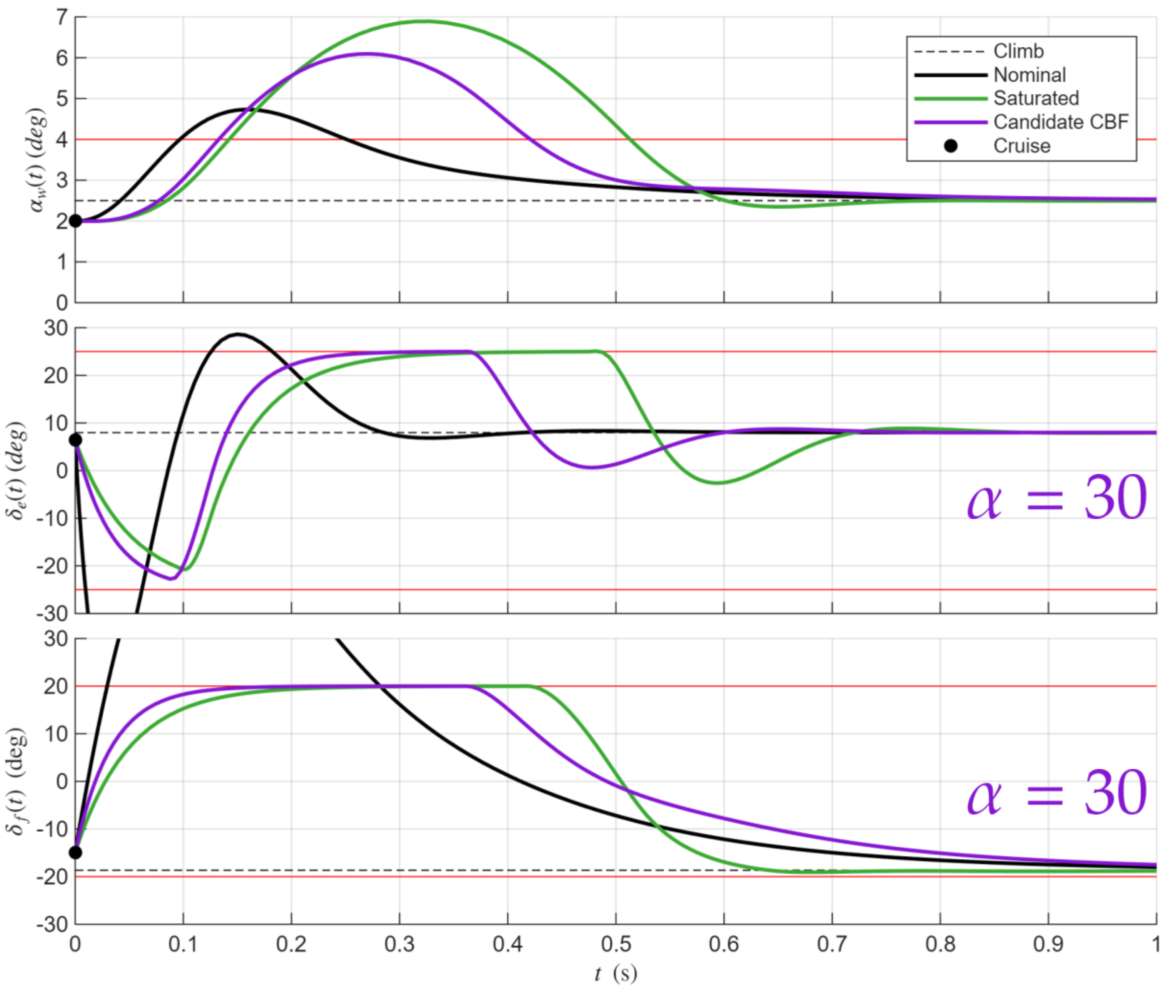
$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\min_{\boldsymbol{u} \in \mathbb{R}^2} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$

$$\text{s.t.} \quad 20(\delta_e - \delta_{e,c}) \geq -30(25 - \delta_e)$$

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\begin{aligned} \min_{\boldsymbol{u} \in \mathbb{R}^2} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & 20(\delta_e - \delta_{e,c}) \geq -30(25 - \delta_e) \\ & 20(\delta_f - \delta_{f,c}) \geq -30(20 - \delta_f) \end{aligned}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$4\ \text{deg} \geq \alpha_{\text{w}}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$4 \text{ deg} \geq \alpha_{\text{w}}$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} A\boldsymbol{x}$$

$$L_g h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} B$$

This procedure yields ***candidate CBFs***. They provide **no safety guarantees**.

F-16 Pitch Dynamics

$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$

$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$

$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} A\boldsymbol{x}$$

$$= -\dot{\theta} + 1.3\alpha_w + 0.2\delta_e + 0.3\delta_f$$

$$L_g h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} B$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x})\boldsymbol{u}$

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} A\boldsymbol{x}$$
$$= -\dot{\theta} + 1.3\alpha_w + 0.2\delta_e + 0.3\delta_f$$

$$L_g h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} B = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$
$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + L_g h(\boldsymbol{x}) \boldsymbol{u}$

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} A \boldsymbol{x}$$

$$= -\dot{\theta} + 1.3\alpha_w + 0.2\delta_e + 0.3\delta_f$$

$$L_g h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} B = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

Relative degree > 1

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x}) + \cancel{L_g h(\boldsymbol{x})}\boldsymbol{u}$

$$L_f h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} A\boldsymbol{x}$$
$$= -\dot{\theta} + 1.3\alpha_w + 0.2\delta_e + 0.3\delta_f$$

$$L_g h(\boldsymbol{x}) = \frac{\partial h}{\partial \boldsymbol{x}} B = \begin{bmatrix} 0 & 0 \end{bmatrix}$$

Lost Control Authority

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$

$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$

$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Differentiate: $\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$

Compute: $\dot{h}(\boldsymbol{x}, \boldsymbol{u}) = L_f h(\boldsymbol{x})$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Lost Control Authority

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$

$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

⚠ This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix}\theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f\end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix}\delta_{e,c} & \delta_{f,c}\end{bmatrix}$$

Nguyen, Sreenath, *ACC* 2016

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

⚠ This procedure yields ***candidate CBFs***. They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

⚠ This procedure yields *candidate CBFs*. They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

$$\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

Nguyen, Sreenath, *ACC* 2016

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x},\boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

$$L_g h_1(\boldsymbol{x}) = \frac{\partial h_1}{\partial \boldsymbol{x}} B$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

$$\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \boxed{\frac{\partial h}{\partial x}(A + \alpha I_n)}$

Compute: $\dot{h}_1(\boldsymbol{x},\boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

$$L_g h_1(\boldsymbol{x}) = \frac{\partial h_1}{\partial \boldsymbol{x}} B = \boxed{\frac{\partial h}{\partial \boldsymbol{x}}(A + \alpha I_n)} B$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

$$\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

$$L_g h_1(\boldsymbol{x}) = \frac{\partial h_1}{\partial \boldsymbol{x}} B = \frac{\partial h}{\partial \boldsymbol{x}} AB + \alpha \frac{\partial h}{\partial \boldsymbol{x}} B$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

$$\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

$$L_g h_1(\boldsymbol{x}) = \frac{\partial h_1}{\partial \boldsymbol{x}} B = \frac{\partial h}{\partial \boldsymbol{x}} AB + \cancel{\alpha \frac{\partial h}{\partial \boldsymbol{x}} B}$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$f(\boldsymbol{x}) = A\boldsymbol{x} \quad g(\boldsymbol{x}) = B$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -0.87 & 43.22 & -17.25 & -1.58 \\ 0 & 0.99 & -1.34 & -0.17 & -0.25 \\ 0 & 0 & 0 & -20 & 0 \\ 0 & 0 & 0 & 0 & -20 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20 & 0 \\ 0 & 20 \end{bmatrix}$$

$$\frac{\partial h}{\partial x} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \end{bmatrix}$$

Nguyen, Sreenath, *ACC* 2016

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

$$L_g h_1(\boldsymbol{x}) = \frac{\partial h_1}{\partial \boldsymbol{x}} B = \begin{bmatrix} 3.4 & 5.0 \end{bmatrix}$$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Recovered Control Authority

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint...

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

Enforce: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha_1\big(h_1(\boldsymbol{x})\big)$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\min_{\boldsymbol{u} \in \mathbb{R}^2} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad \dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha_1\big(h_1(\boldsymbol{x})\big)$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

Enforce: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha_1\big(h_1(\boldsymbol{x})\big)$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\min_{\boldsymbol{u} \in \mathbb{R}^2} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad A_{QP}\boldsymbol{u} \leq b_{QP}(\boldsymbol{x})$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x},\boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

Enforce: $\dot{h}_1(\boldsymbol{x},\boldsymbol{u}) \geq -\alpha_1\big(h_1(\boldsymbol{x})\big)$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\min_{\boldsymbol{u} \in \mathbb{R}^2} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad A_{QP}\boldsymbol{u} \leq b_{QP}(\boldsymbol{x})$$

$$A_{QP} = -L_g h_1(\boldsymbol{x}) = -\begin{bmatrix} 3.4 & 5.0 \end{bmatrix}$$

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x},\boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

Enforce: $\dot{h}_1(\boldsymbol{x},\boldsymbol{u}) \geq -\alpha_1\big(h_1(\boldsymbol{x})\big)$

This procedure yields ***candidate CBFs***. They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix}\theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f\end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix}\delta_{e,c} & \delta_{f,c}\end{bmatrix}$$

$$\begin{aligned} \min_{\boldsymbol{u}\in\mathbb{R}^2} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & A_{QP}\boldsymbol{u} \leq b_{QP}(\boldsymbol{x}) \end{aligned}$$

$$A_{QP} = -L_g h_1(\boldsymbol{x}) = -\begin{bmatrix}3.4 & 5.0\end{bmatrix}$$

$$b_{QP}(\boldsymbol{x}) = (\alpha + \alpha_1)\frac{\partial h}{\partial x}A\boldsymbol{x} + \alpha\alpha_1 h(\boldsymbol{x}) + \frac{\partial h}{\partial x}A^2\boldsymbol{x}$$

Nguyen, Sreenath, *ACC* 2016

Example: Safety Filter for F-16

Naïve CBF Design (alpha)

Stall constraint: $h(\boldsymbol{x}) = 4 - \alpha_w \geq 0$

Enforce: $L_f h(\boldsymbol{x}) \geq -\alpha\big(h(\boldsymbol{x})\big)$

Treat as new constraint…

Define: $h_1(\boldsymbol{x}) = L_f h(\boldsymbol{x}) + \alpha\big(h(\boldsymbol{x})\big) \geq 0$

Differentiate: $\frac{\partial h_1}{\partial x} = \frac{\partial h}{\partial x}(A + \alpha I_n)$

Compute: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) = L_f h_1(\boldsymbol{x}) + L_g h_1(\boldsymbol{x})\boldsymbol{u}$

Enforce: $\dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha_1\big(h_1(\boldsymbol{x})\big)$

Tune: $\alpha, \alpha_1 > 0$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix}\theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f\end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix}\delta_{e,c} & \delta_{f,c}\end{bmatrix}$$

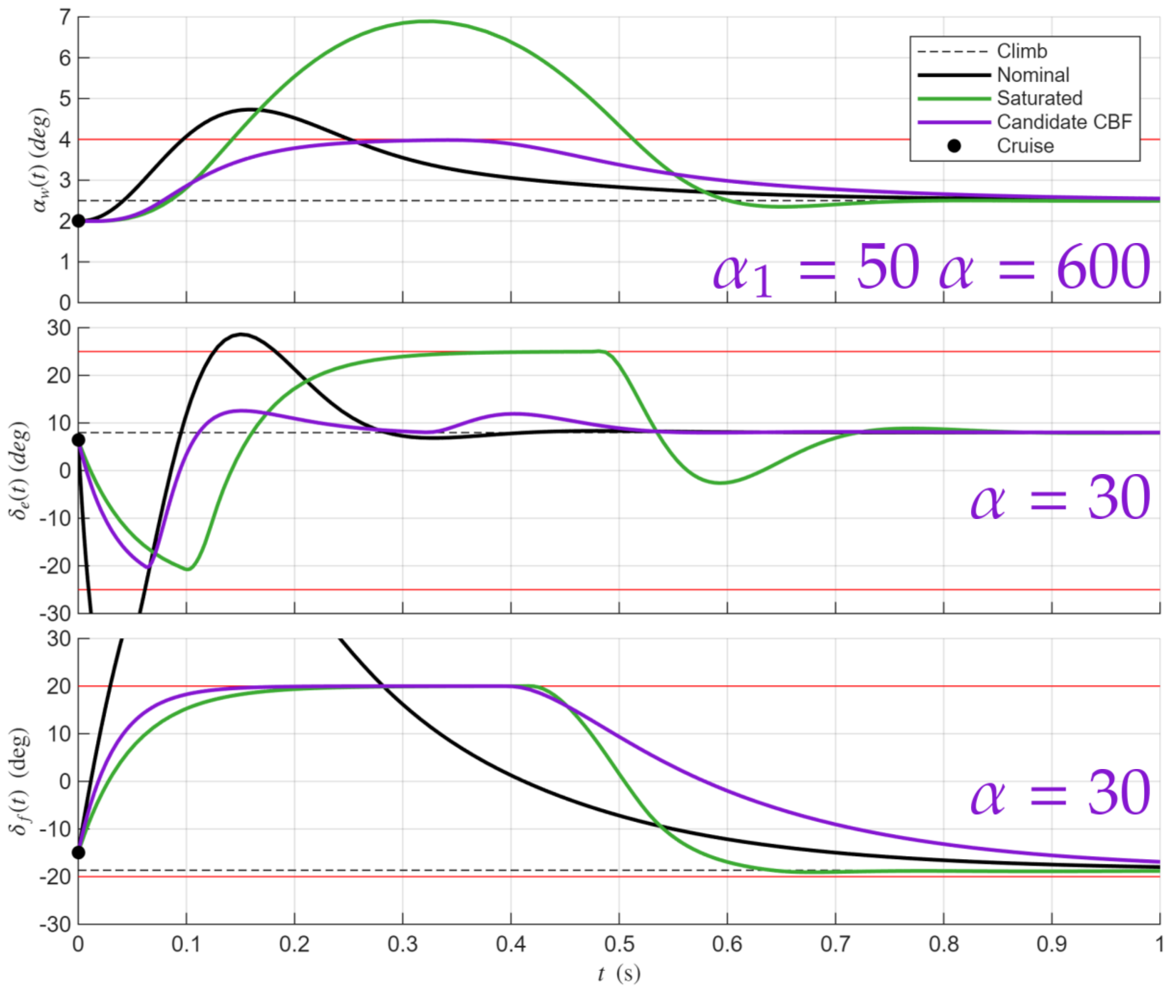
Roots of polynomial, eigenvalues (2x2)

$$\min_{\boldsymbol{u}\in\mathbb{R}^2} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$
$$\text{s.t.} \quad A_{QP}\boldsymbol{u} \leq b_{QP}(\boldsymbol{x})$$

$$A_{QP} = -L_g h_1(\boldsymbol{x}) = -\begin{bmatrix}3.4 & 5.0\end{bmatrix}$$

$$b_{QP}(\boldsymbol{x}) = \boxed{(\alpha + \alpha_1)}\frac{\partial h}{\partial x}A\boldsymbol{x} + \boxed{\alpha\alpha_1}h(\boldsymbol{x}) + \frac{\partial h}{\partial x}A^2\boldsymbol{x}$$

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

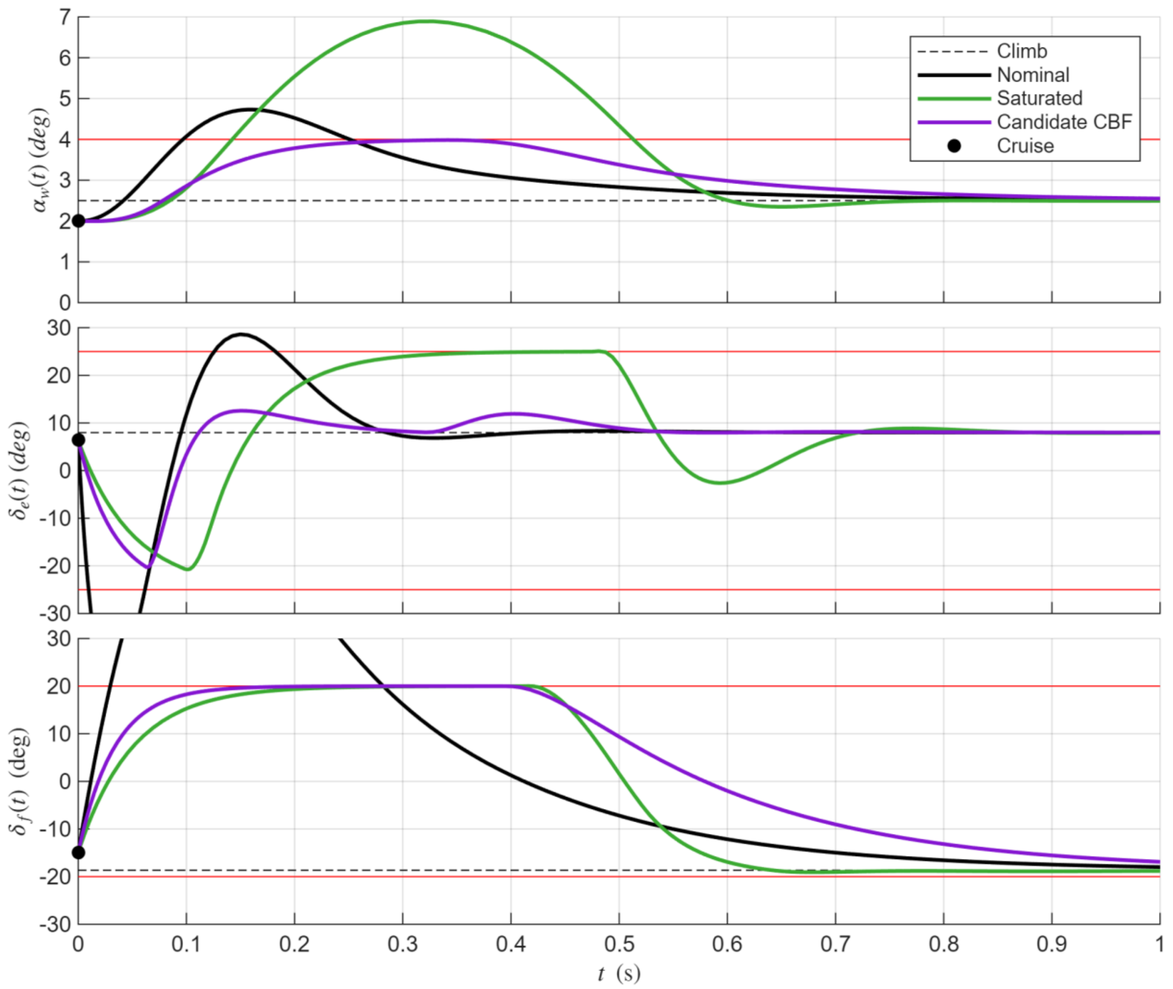
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\begin{aligned} \min_{\boldsymbol{u} \in \mathbb{R}^2} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & 20(\delta_e - \delta_{e,c}) \geq -30(25 - \delta_e) \\ & 20(\delta_f - \delta_{f,c}) \geq -30(20 - \delta_f) \\ & \dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha_1\big(h_1(\boldsymbol{x})\big) \end{aligned}$$

mosek

3.5 ms solvetime

Example: Safety Filter for F-16



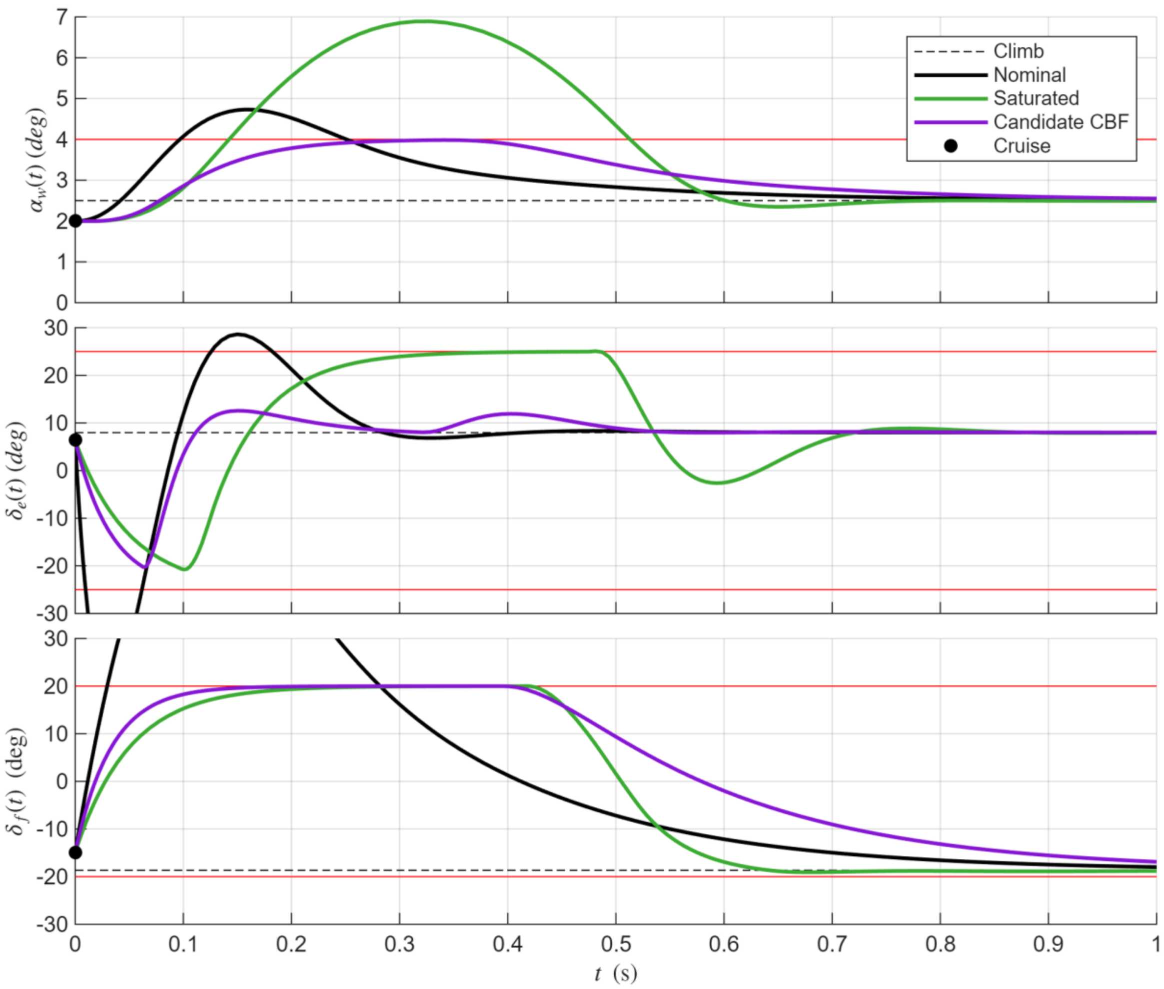
F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

This procedure yields *candidate CBFs*.
They provide **no safety guarantees**.

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

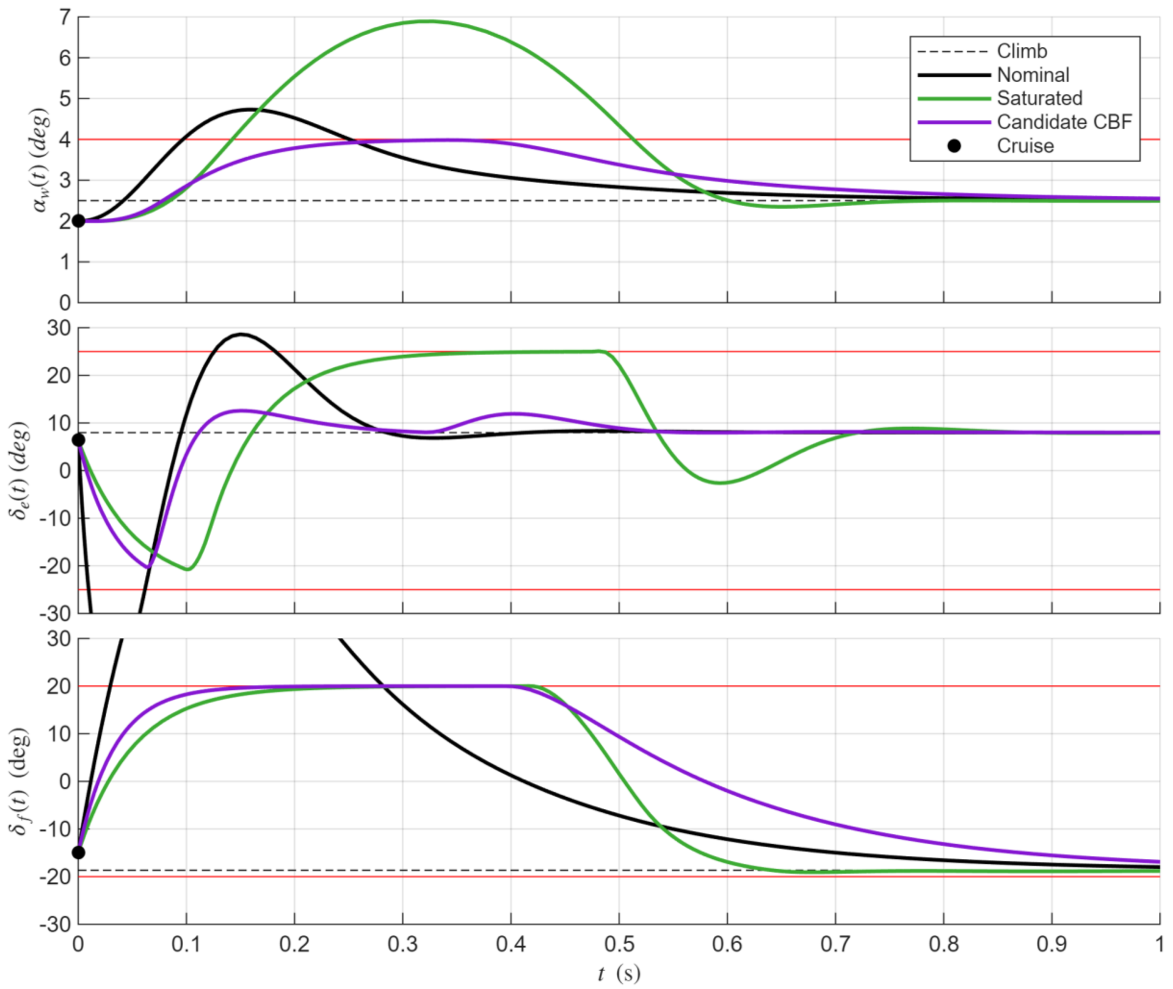
Cruise: $\theta = 2, \ \gamma = 0$

↓ Goal

Climb: $\theta = 9, \ \gamma = 6.5$

This procedure yields ***candidate CBFs***. They provide **no safety guarantees**.

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

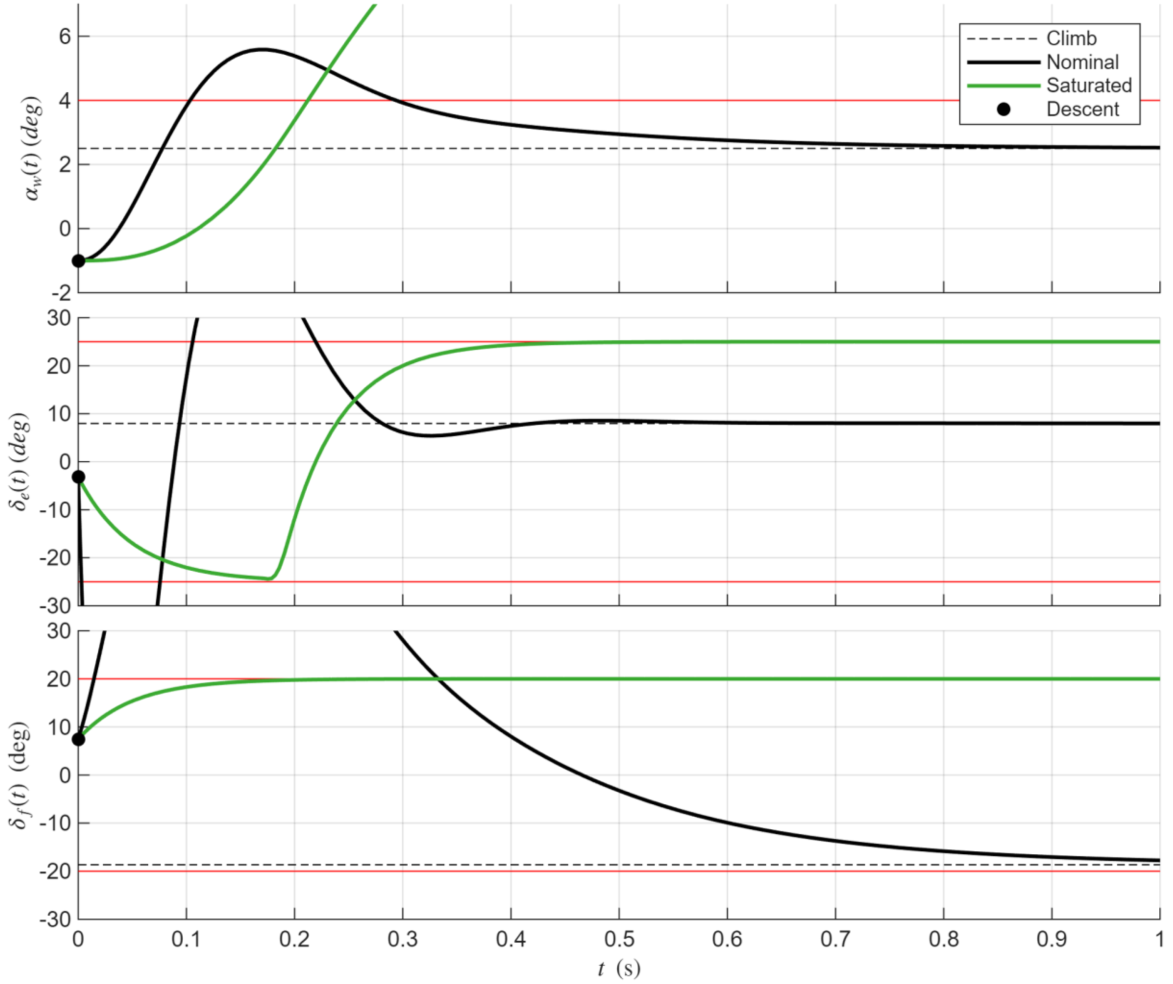
Cruise: $\theta = 2,\ \gamma = 0$

Goal

Climb: $\theta = 9,\ \gamma = 6.5$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

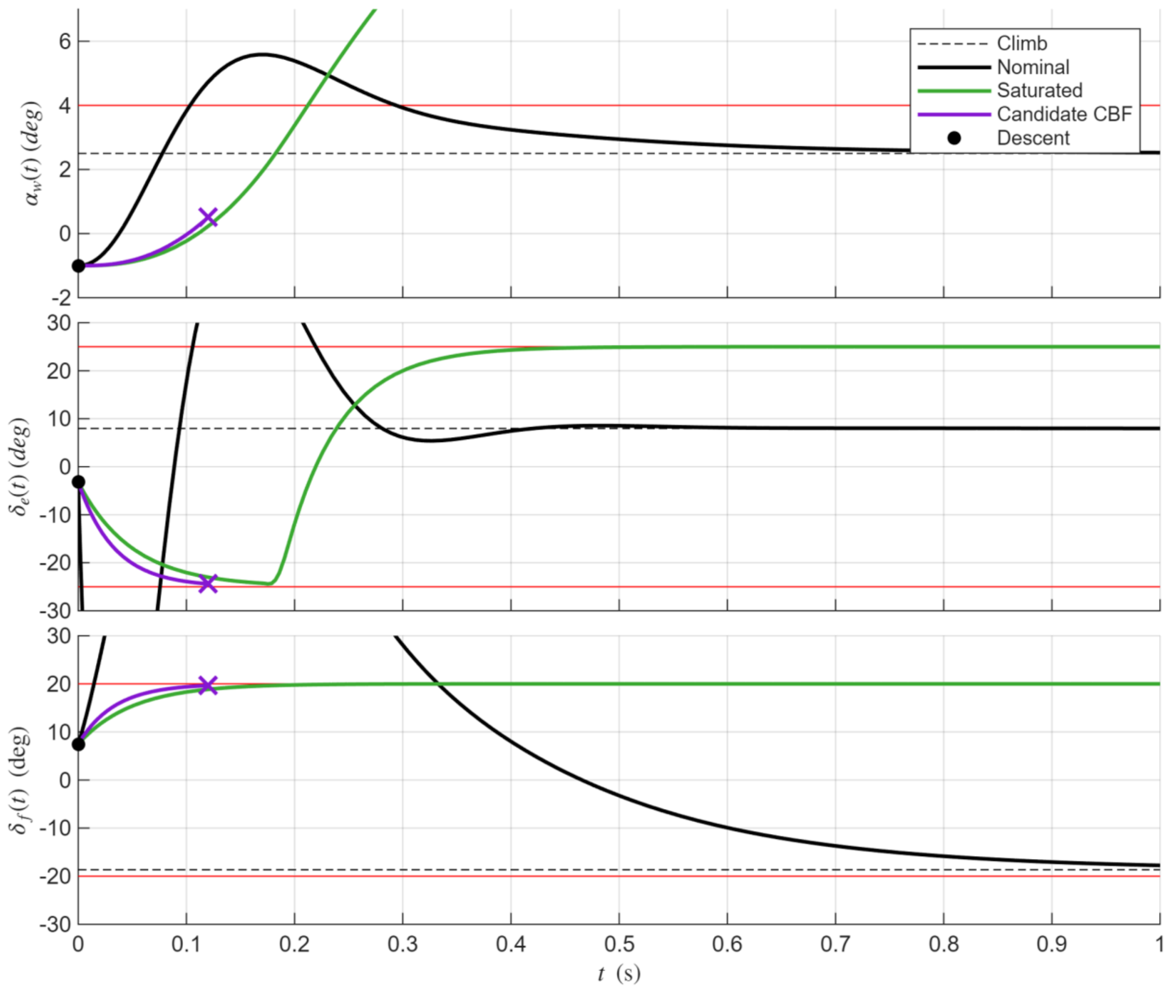
Descent: $\theta = -3, \ \gamma = -2$

↓ Goal

Climb: $\theta = 9, \ \gamma = 6.5$

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

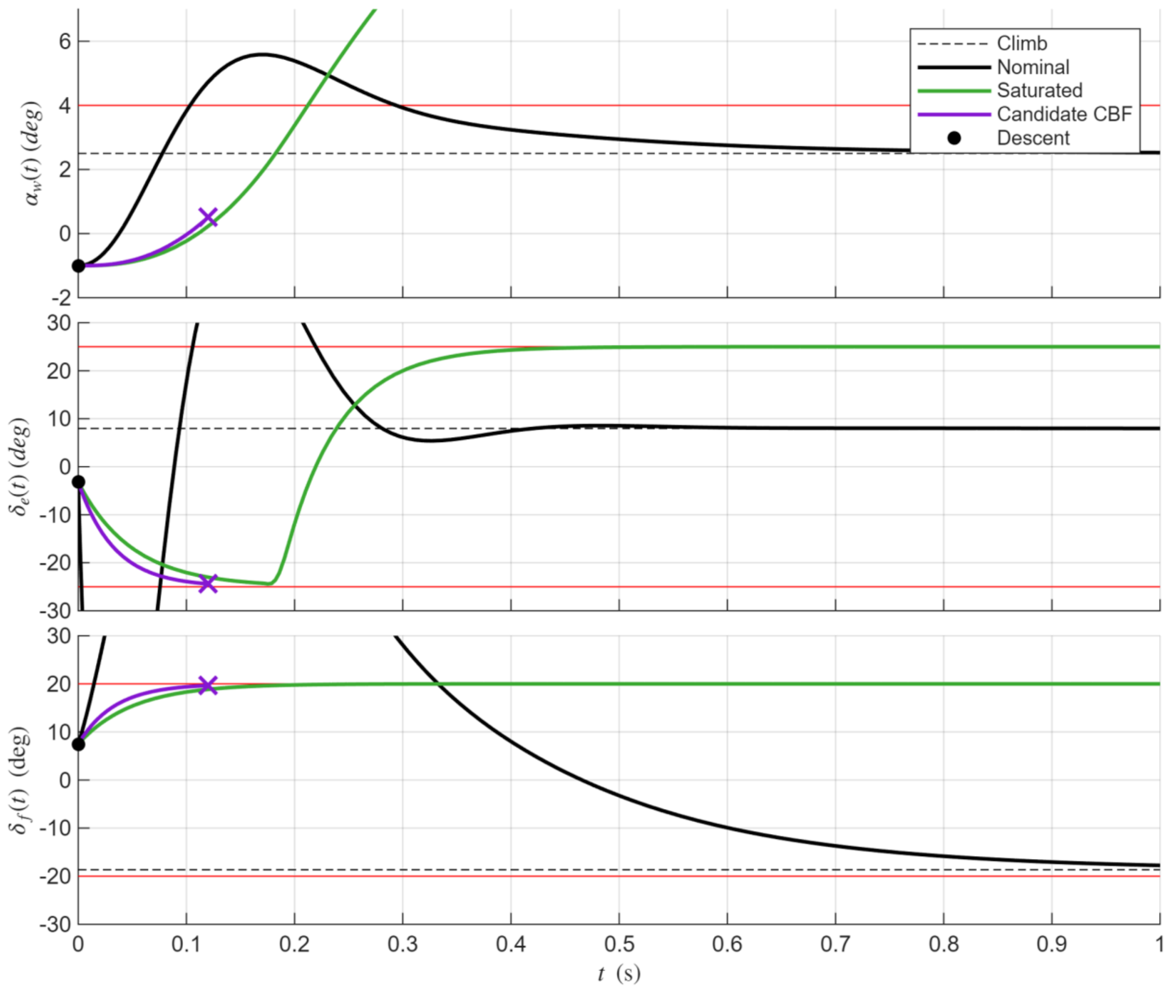
Descent: $\theta = -3,\ \gamma = -2$

Goal

Climb: $\theta = 9,\ \gamma = 6.5$

This procedure yields ***candidate CBFs***. They provide **no safety guarantees**.

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

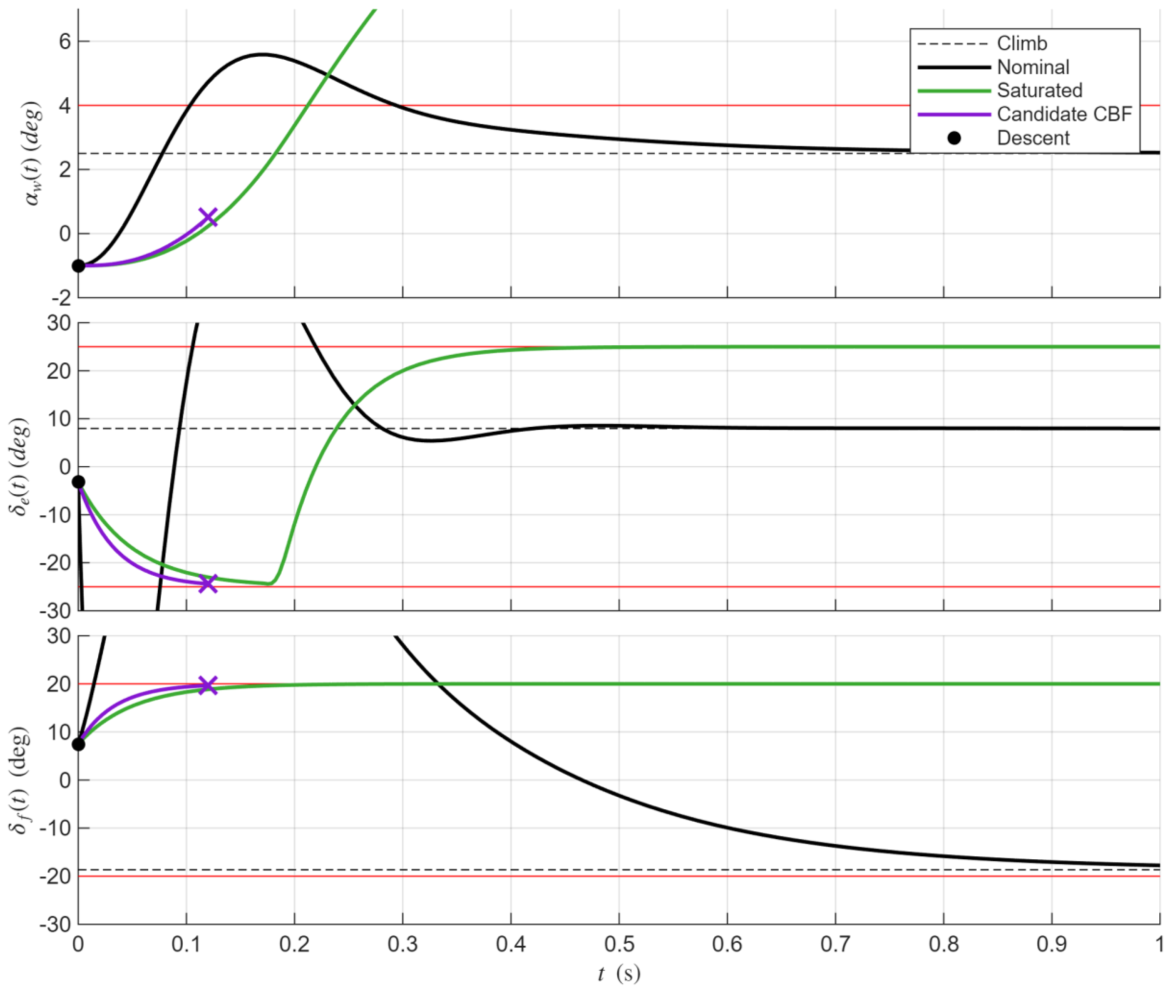
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\begin{aligned} \min_{\boldsymbol{u}\in\mathbb{R}^2} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & 20(\delta_e - \delta_{e,c}) \geq -30(25 - \delta_e) \\ & 20(\delta_f - \delta_{f,c}) \geq -30(20 - \delta_f) \\ & \dot{h}_1(\boldsymbol{x},\boldsymbol{u}) \geq -\alpha_1(h_1(\boldsymbol{x})) \end{aligned}$$

Infeasible

This procedure yields ***candidate CBFs***.
They provide **no safety guarantees**.

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

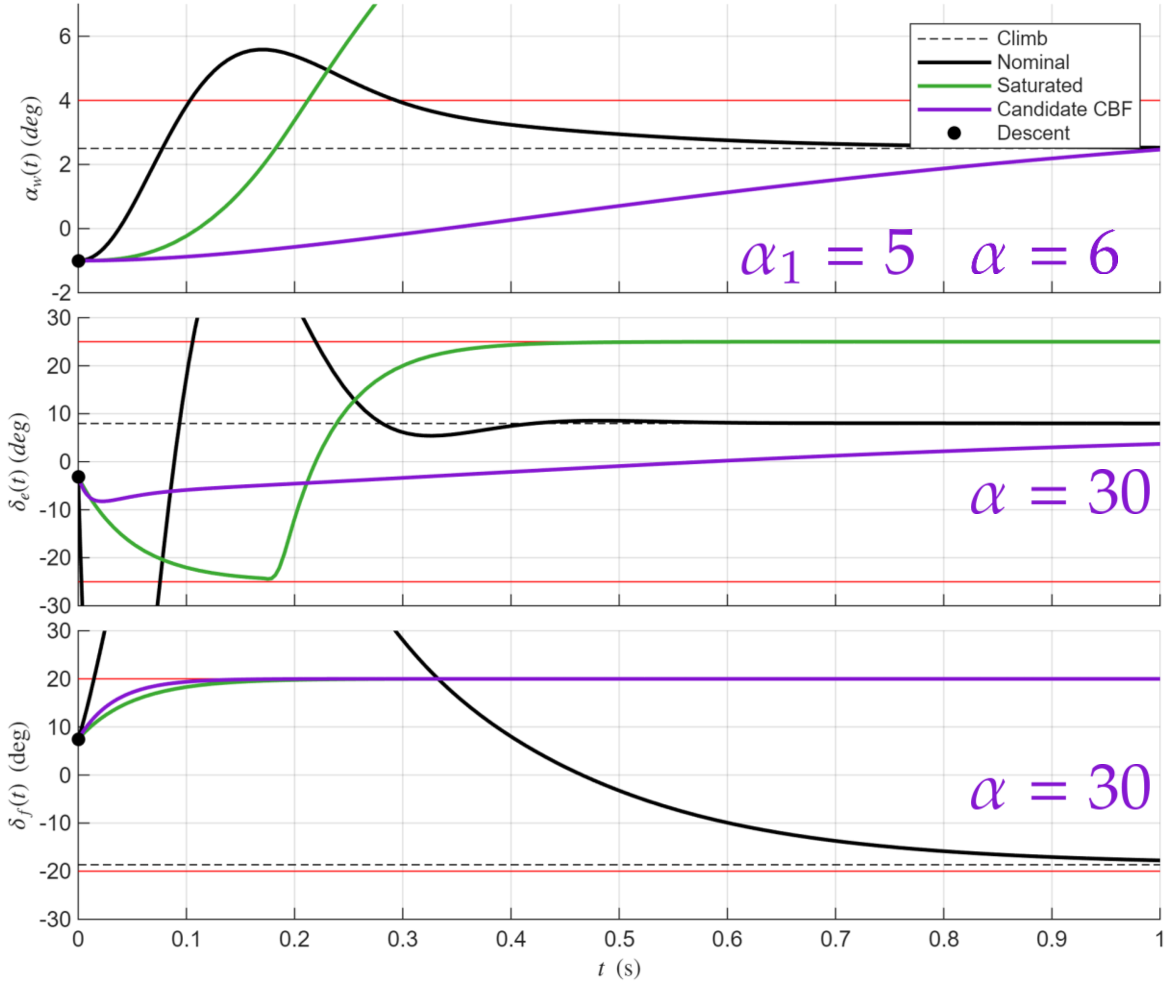
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\begin{aligned} \min_{\boldsymbol{u} \in \mathbb{R}^2} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & 20(\delta_e - \delta_{e,c}) \geq -30(25 - \delta_e) \\ & 20(\delta_f - \delta_{f,c}) \geq -30(20 - \delta_f) \\ & \dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha_1(h_1(\boldsymbol{x})) \end{aligned}$$

Infeasible

Can be fixed by tuning α

Example: Safety Filter for F-16



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\begin{aligned} \min_{\boldsymbol{u} \in \mathbb{R}^2} \quad & \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2 \\ \text{s.t.} \quad & 20(\delta_e - \delta_{e,c}) \geq -30(25 - \delta_e) \\ & 20(\delta_f - \delta_{f,c}) \geq -30(20 - \delta_f) \\ & \dot{h}_1(\boldsymbol{x}, \boldsymbol{u}) \geq -\alpha_1(h_1(\boldsymbol{x})) \end{aligned}$$

Infeasible

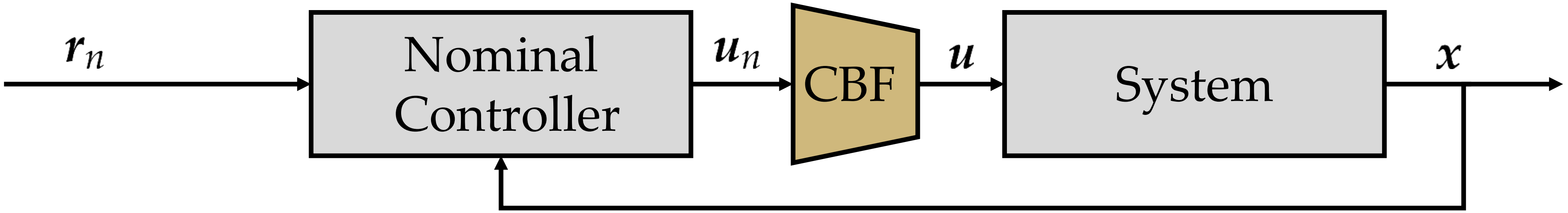
Can be fixed by tuning α, bad performance

Index

Safety Filters

Family of constrained controllers that use a two-step approach:

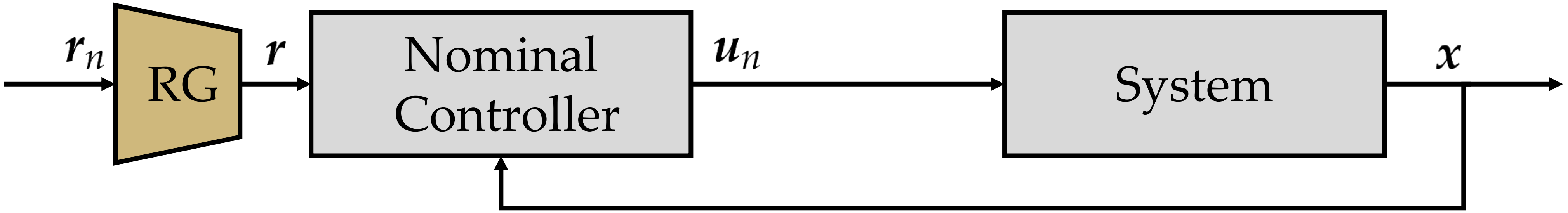
1. Design a control law for the **Unconstrained** system,
2. Introduce an **Add-on** unit for constraint enforcement.
   - **Control Barrier Function:** Filter nominal controller



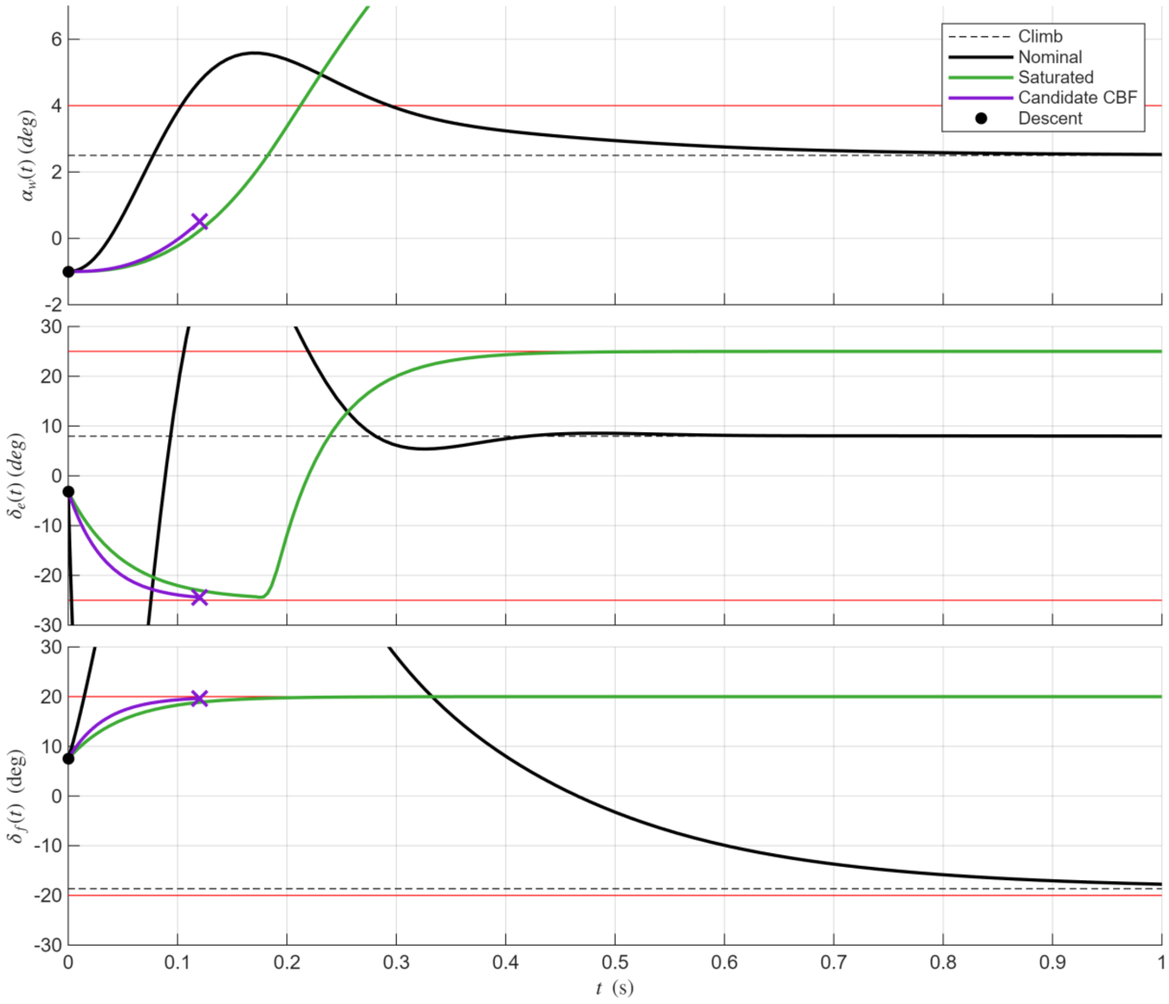
Safety Filters

Family of constrained controllers that use a two-step approach:

1. Design a control law for the **Unconstrained** system,
2. Introduce an **Add-on** unit for constraint enforcement.
   - **Control Barrier Function:** Filter nominal controller
   - **Reference Governor:** Filter nominal reference



My Work: DSMs are CBFs



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

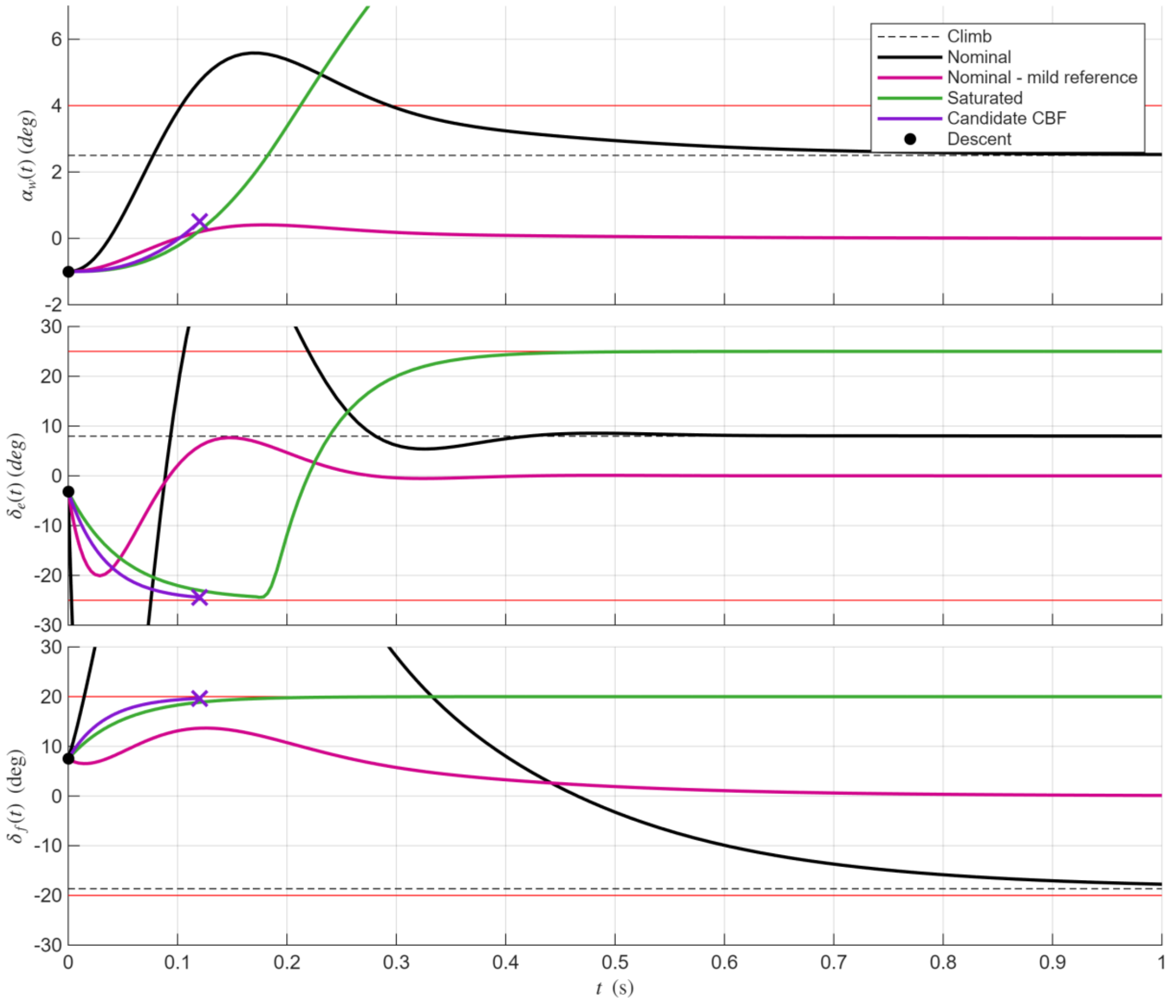
$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Descent: $\theta = -3,\ \gamma = -2$

↓ Goal

Climb: $\theta = 9,\ \gamma = 6.5$

My Work: DSMs are CBFs



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

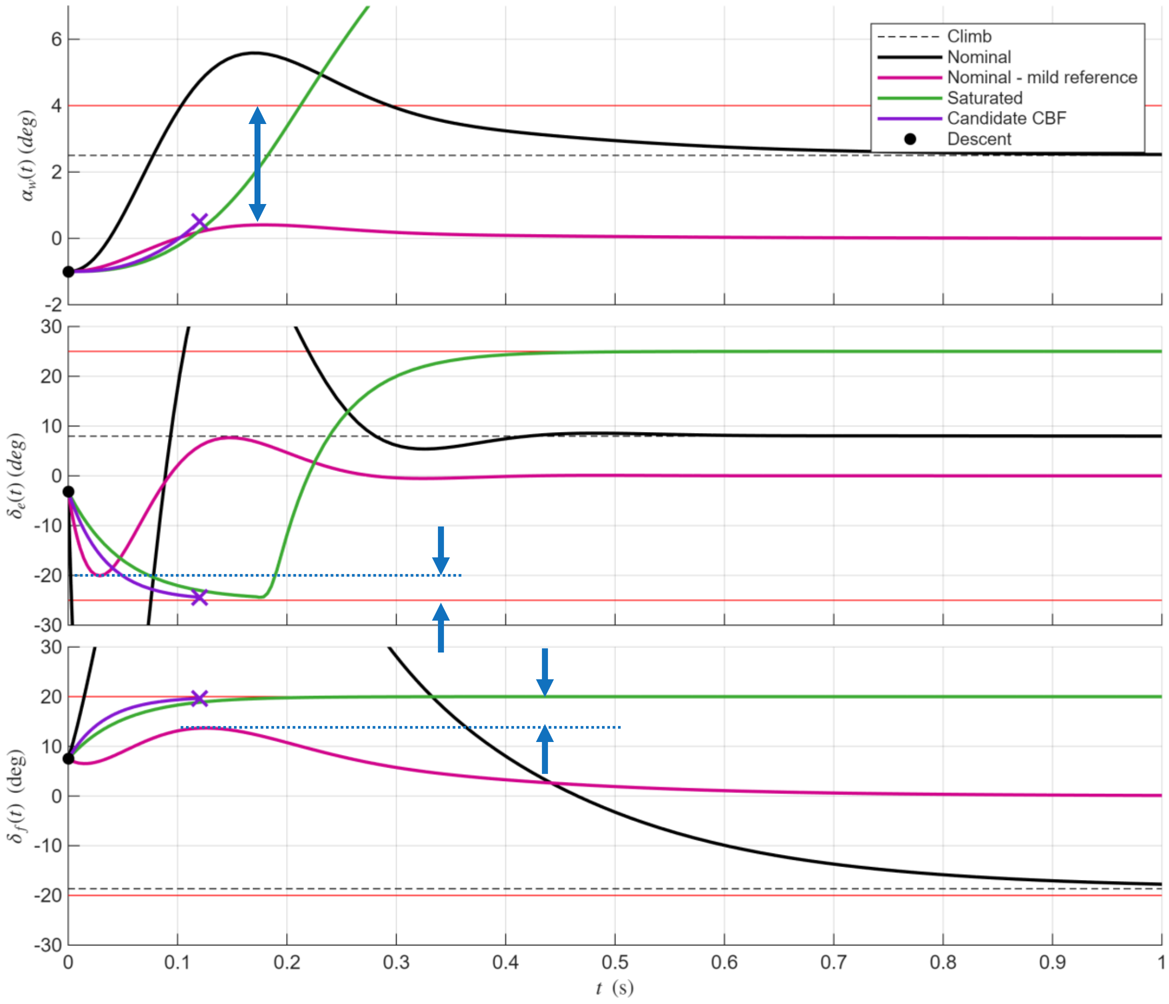
Descent: $\theta = -3,\ \gamma = -2$

Intermediary Goal

Milder Descent: $\theta = -1,\ \gamma = -1$

LQR Controller $\kappa(\boldsymbol{x}) = \bar{u}(\boldsymbol{v}) - K(\boldsymbol{x} - \bar{x}(\boldsymbol{v}))$

My Work: DSMs are CBFs



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

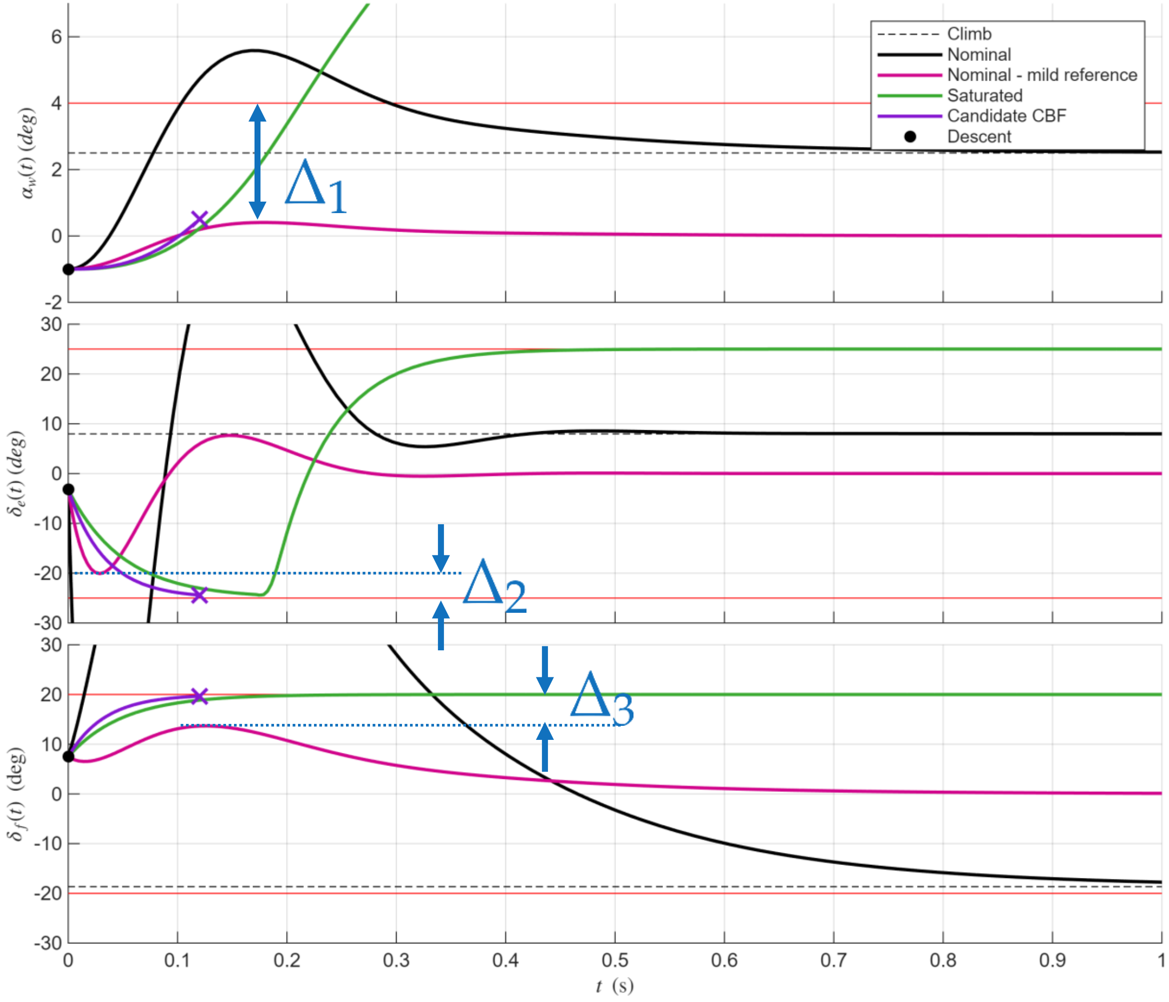
Descent: $\theta = -3,\ \gamma = -2$

Intermediary Goal

Milder Descent: $\theta = -1,\ \gamma = -1$

LQR Controller $\kappa(\boldsymbol{x}) = \bar{u}(\boldsymbol{v}) - K(\boldsymbol{x} - \bar{x}(\boldsymbol{v}))$

My Work: DSMs are CBFs



F-16 Pitch Dynamics

$$\boldsymbol{x} = [\theta \quad \dot{\theta} \quad \alpha_w \quad \delta_e \quad \delta_f]$$

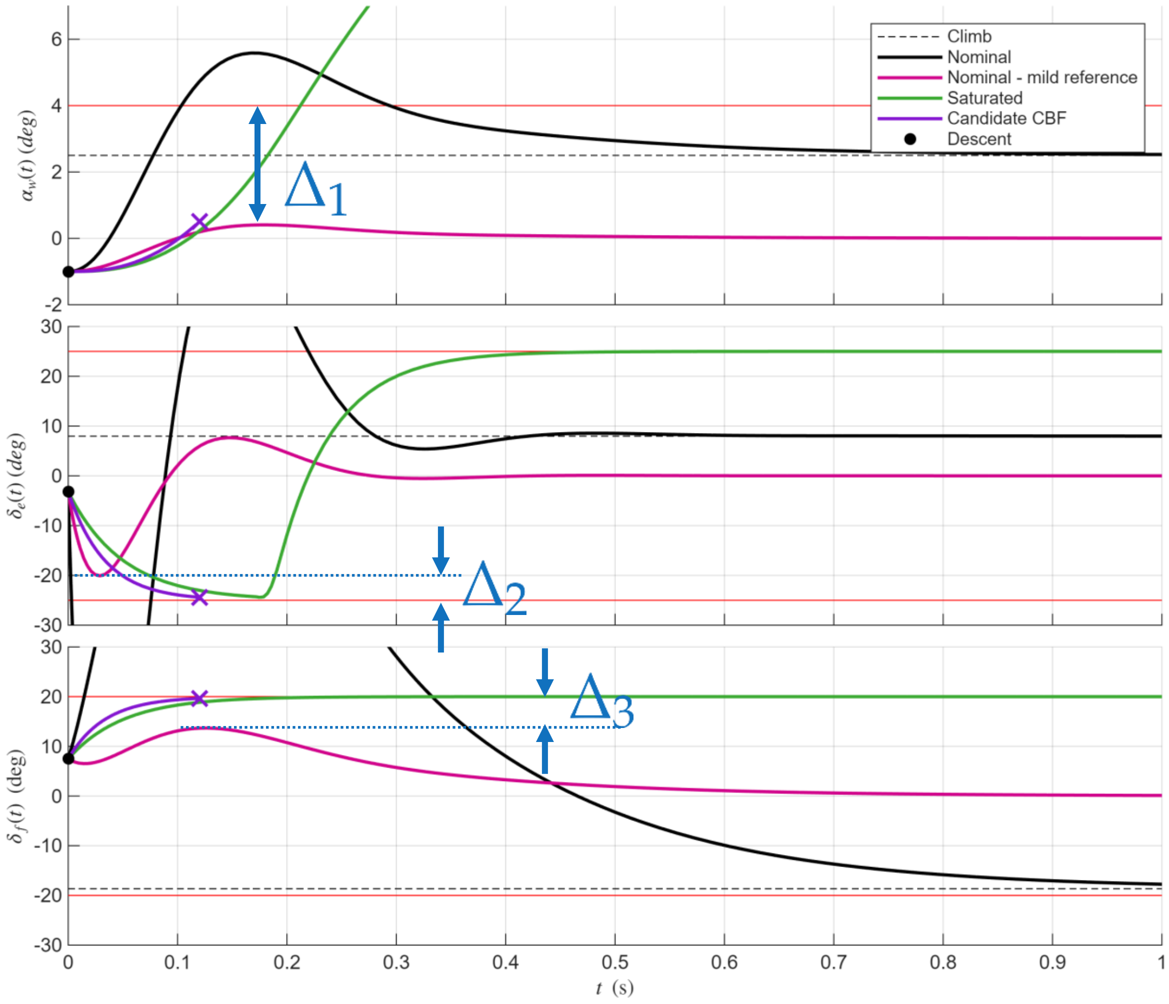
$$\boldsymbol{u} = [\delta_{e,c} \quad \delta_{f,c}]$$

Dynamic Safety Margin (DSM)

$$\Delta(\boldsymbol{x}, v) = \inf_{\tau \in [0,\infty)} h\big(\Phi(\tau, v)\big)$$

LQR Controller $\kappa(\boldsymbol{x}) = \bar{u}(v) - K(\boldsymbol{x} - \bar{x}(v))$

My Work: DSMs are CBFs



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

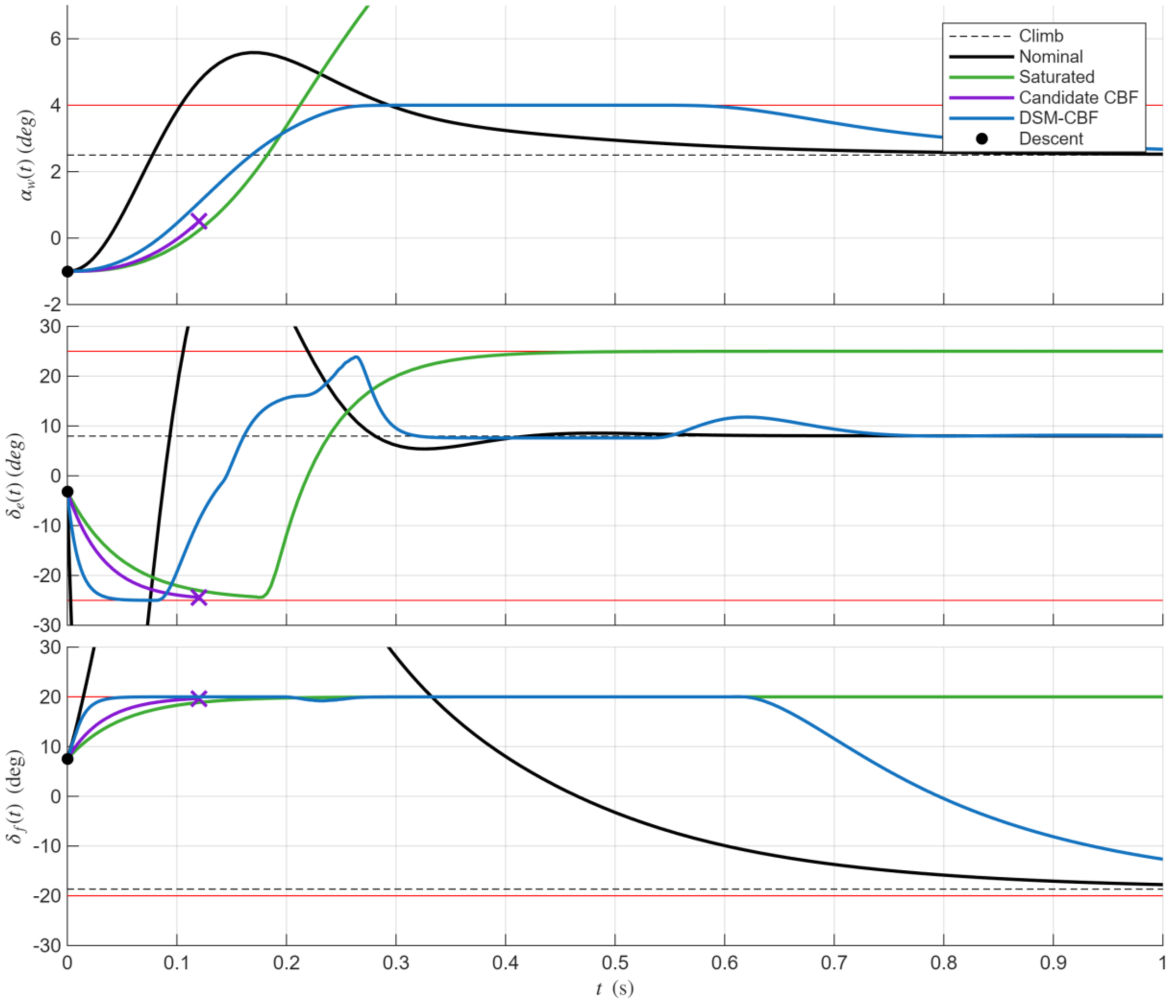
Dynamic Safety Margin (DSM)

$$\Delta(\boldsymbol{x}, v) = \inf_{\tau \in [0,\infty)} h\big(\Phi(\tau, v)\big)$$

Theorem: *DSMs are CBFs*

If Δ is a DSM, then it is a CBF for an augmented system $\begin{bmatrix} \boldsymbol{x} & v \end{bmatrix}^\top$.

My Work: DSMs are CBFs



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

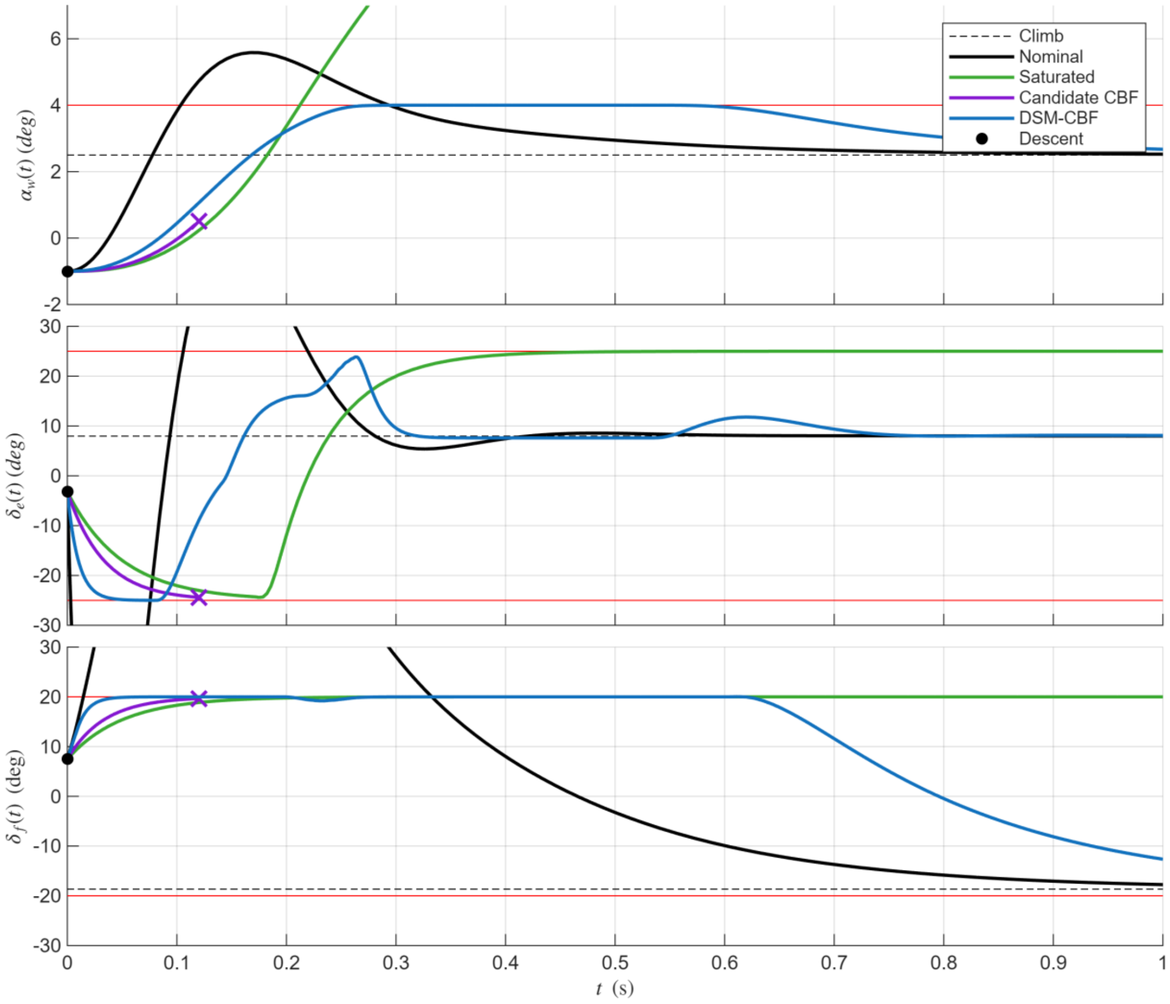
$$\min_{\boldsymbol{u}\in\mathbb{R}^2} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$

$$\text{s.t.} \quad \dot{\Delta}_1(\boldsymbol{x},\boldsymbol{v},\boldsymbol{u}) \geq -\alpha_1(\Delta_1(\boldsymbol{x},\boldsymbol{b}))$$

$$\dot{\Delta}_2(\boldsymbol{x},\boldsymbol{v},\boldsymbol{u}) \geq -\alpha_2(\Delta_2(\boldsymbol{x},\boldsymbol{b}))$$

$$\dot{\Delta}_3(\boldsymbol{x},\boldsymbol{v},\boldsymbol{u}) \geq -\alpha_3(\Delta_3(\boldsymbol{x},\boldsymbol{b}))$$

My Work: DSMs are CBFs



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

$$\min_{\boldsymbol{u} \in \mathbb{R}^2} \quad \|\boldsymbol{u} - \kappa(\boldsymbol{x})\|^2$$

$$\text{s.t.} \quad \dot{\Delta}_1(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{u}) \geq -\alpha_1(\Delta_1(\boldsymbol{x}, \boldsymbol{b}))$$

$$\dot{\Delta}_2(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{u}) \geq -\alpha_2(\Delta_2(\boldsymbol{x}, \boldsymbol{b}))$$

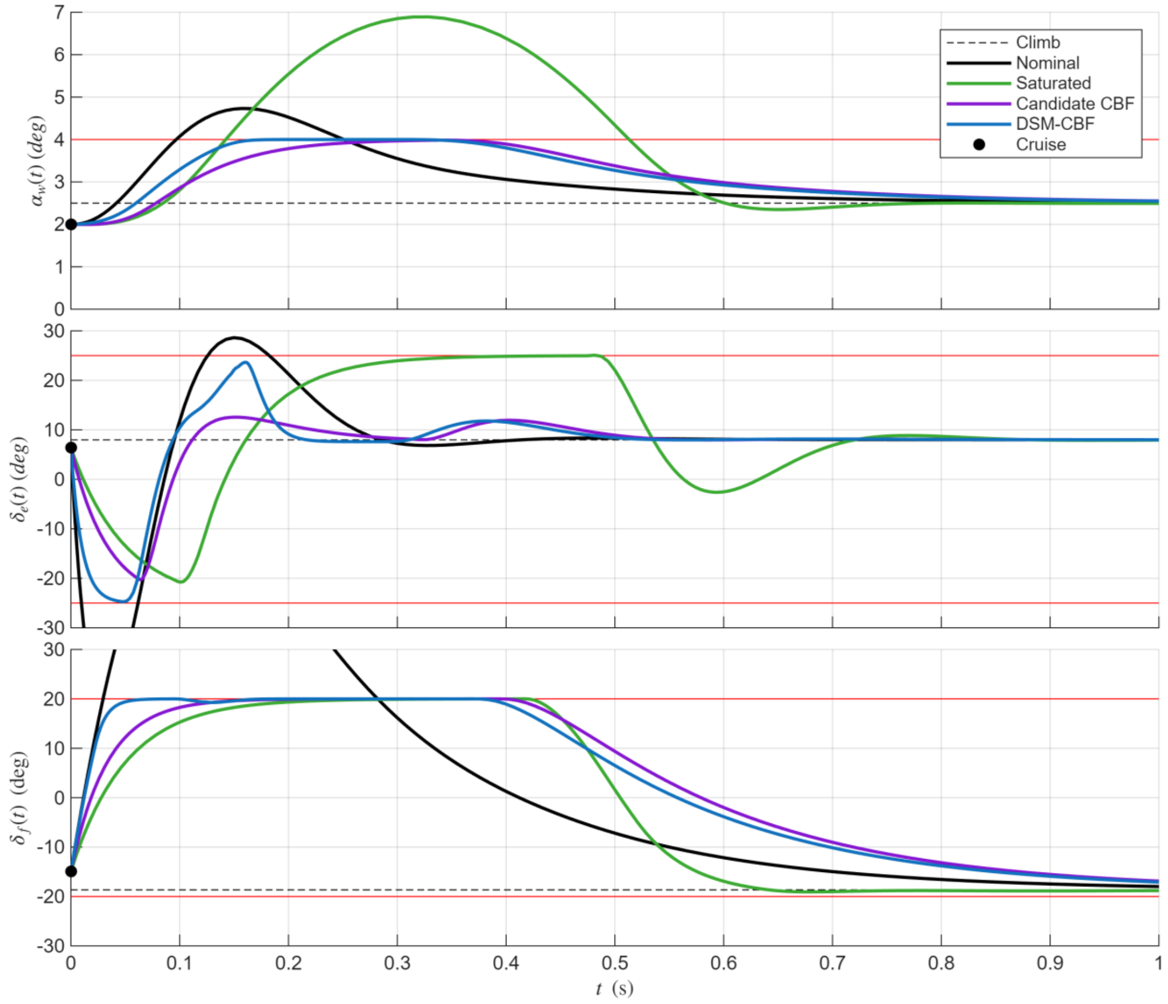
$$\dot{\Delta}_3(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{u}) \geq -\alpha_3(\Delta_3(\boldsymbol{x}, \boldsymbol{b}))$$

mosek

DSM-CBF: 16 ms solvetime

Candidate CBF: 3.5 ms solvetime

My Work: DSMs are CBFs



F-16 Pitch Dynamics

$$\boldsymbol{x} = \begin{bmatrix} \theta & \dot{\theta} & \alpha_w & \delta_e & \delta_f \end{bmatrix}$$

$$\boldsymbol{u} = \begin{bmatrix} \delta_{e,c} & \delta_{f,c} \end{bmatrix}$$

Cruise: $\theta = 2,\ \gamma = 0$

↓ Goal

Climb: $\theta = 9,\ \gamma = 6.5$

mosek

***DSM-CBF**: 16 ms solvetime*

***Candidate CBF**: 3.5 ms solvetime*

Acknowledgements