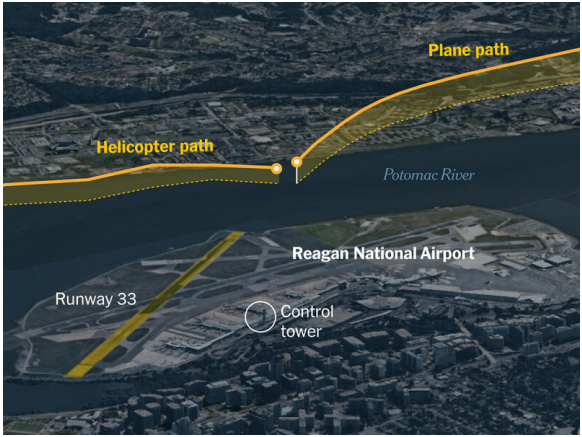
# Successive Control Certificates for Safe Autonomy

Rameez Wajid
Advisor: Sriram Sankaranarayanan

Proposal presentation, May 22, 2025

# Motivation

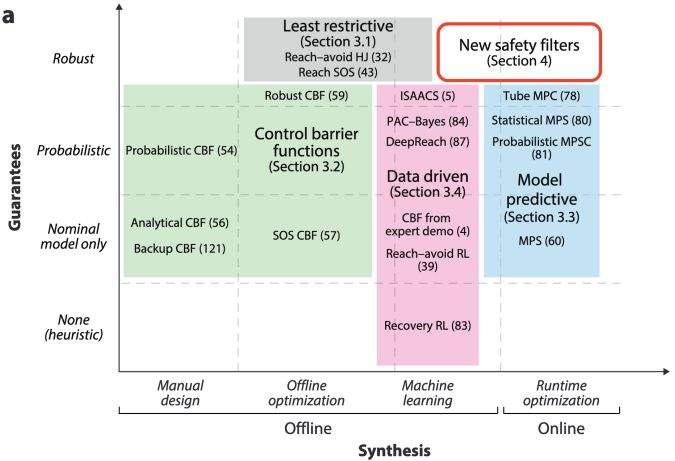

Washington plane crash
BBC NEWS


Plane path
Helicopter path
Potomac River
Reagan National Airport
Runway 33
Control tower

# Safety Filters

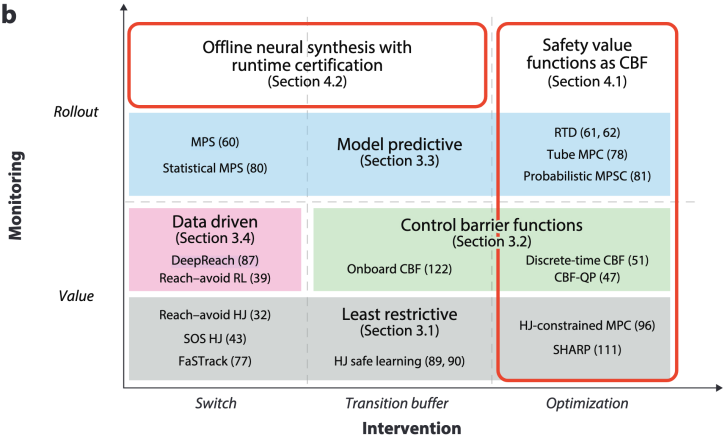
"A safety filter is an automatic process that monitors the operation of an autonomous system at runtime and intervenes, when deemed necessary, ... "

# Safety Filters




[0]Hsu, Kai-Chieh, Haimin Hu, and Jaime F. Fisac. "The safety filter: A unified view of safety-critical control in autonomous systems." Annual Review of Control, Robotics, and Autonomous Systems 7 (2023)

# Safety Filters




[0]Hsu, Kai-Chieh, Haimin Hu, and Jaime F. Fisac. "The safety filter: A unified view of safety-critical control in autonomous systems." Annual Review of Control, Robotics, and Autonomous Systems 7 (2023)

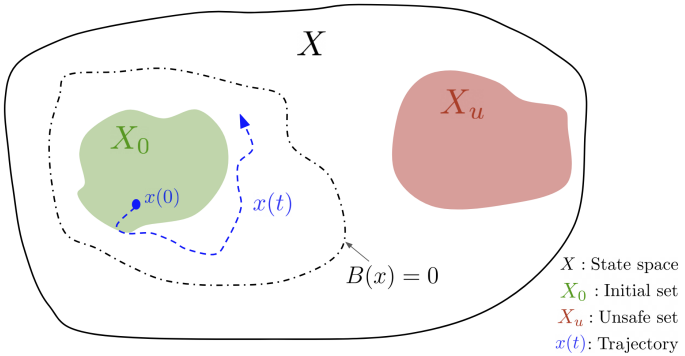
# Proposal

Successive Control Certificates:

- Easier control certificate synthesis
- Less conservative controlled invariant sets
- Reduce the time for the safety filter override
- More tractable certification
- Interesting nonlinear dynamics

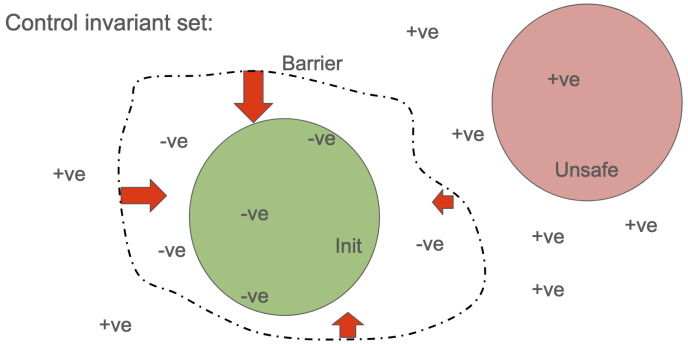
# Successive Control Barrier Functions

# Barrier Functions - [Prajna et al.]




- $B(\vec{x}) > 0$ for all $\vec{x} \in X_u$ (B is **positive** when unsafe)
- $B(\vec{x}) < 0$ for all $\vec{x} \in X_i$ (B is **negative** when init)
- $B(\vec{x}) = 0$ implies $\nabla B(\vec{x}) \cdot f(\vec{x}, \vec{u}) \leq 0$

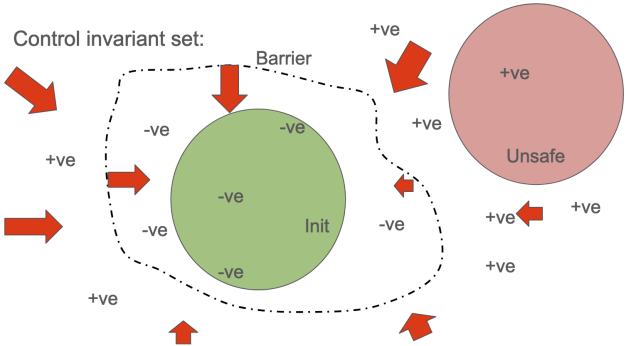
# Control Barrier Functions - [Ames et al.]




- $B(\vec{x}) > 0$ for all $\vec{x} \in X_u$ (B is **positive** when unsafe)
- $B(\vec{x}) < 0$ for all $\vec{x} \in X_i$ (B is **negative** when init)
- $B(\vec{x}) = 0$ implies there **exists a control input** $\vec{u} \in U$ such that $\nabla B(\vec{x}) \cdot f(\vec{x}, \vec{u}) < 0$

# Control Barrier Functions - Exponential [Kong et al.]

- State: $\vec{x} \in \mathbb{R}^n$
- Control inputs: $\vec{u} \in \mathbb{R}^m$
- $\dot{\vec{x}} = f(\vec{x}, \vec{u})$, $X \subseteq \mathbb{R}^n$,




- $B(\vec{x}) > 0$ for all $\vec{x} \in X_u$ (B is **positive** when unsafe)
- $B(\vec{x}) < 0$ for all $\vec{x} \in X_i$ (B is **negative** when init)
- for all $\vec{x} \in \mathbb{R}^n$ there **exists a control input** $\vec{u} \in U$ s.t. $\nabla B(\vec{x}) \cdot f(\vec{x}, \vec{u}) \leq -\lambda B(\vec{x})$

# Control Barrier Functions - Exponential [Kong et al.]

- State: $\vec{x} \in \mathbb{R}^n$
- Control inputs: $\vec{u} \in \mathbb{R}^m$
- $\dot{\vec{x}} = f(\vec{x}, \vec{u})$, $X \subseteq \mathbb{R}^n$,

It's a hard problem:

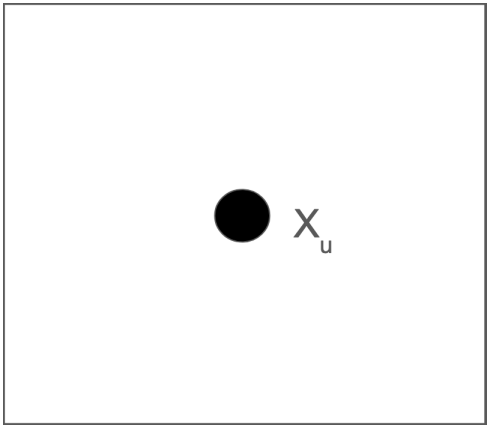
- Trying to synthesize *Barrier* and *Control* simultaneously
- Bilinearity

- $B(\vec{x}) > 0$ for all $\vec{x} \in X_u$ (B is **positive** when unsafe)
- $B(\vec{x}) \leq 0$ for all $\vec{x} \in X_i$ (B is **negative** when init)
- for all $\vec{x} \in \mathbb{R}^n$ there **exists a control input** $\vec{u} \in U$ s.t. $\nabla B(\vec{x}) \cdot f(\vec{x}, \vec{u}) \leq -\lambda B(\vec{x})$

# One Barrier is not enough

Computing Barrier:

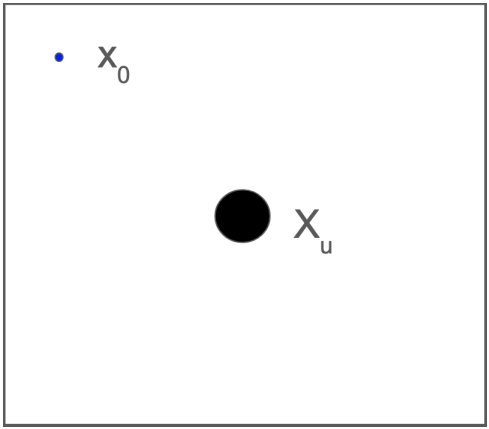
- Fix $u = u_i \in U$
- Barrier function: $B_i(\vec{x})$
- $\dot{\vec{x}} = f(\vec{x}, u_i)$



# One Barrier is not enough

Computing Barrier:

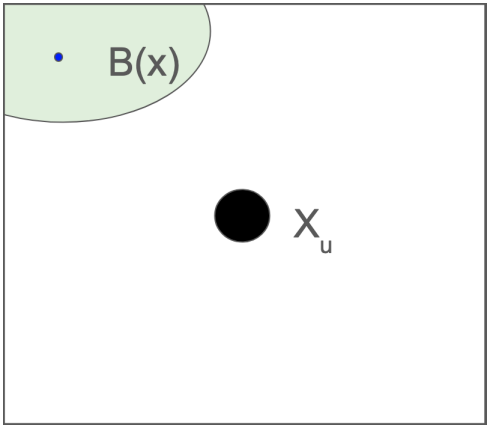
- Fix $u = u_i \in U$
- Barrier function: $B_i(\vec{x})$
- $\dot{\vec{x}} = f(\vec{x}, u_i)$



# One Barrier is not enough

Computing Barrier:

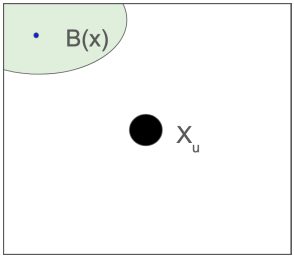
- Fix $u = u_i \in U$
- Barrier function: $B_i(\vec{x})$
- $\dot{\vec{x}} = f(\vec{x}, u_i)$



# One Barrier is not enough

Computing Barrier:

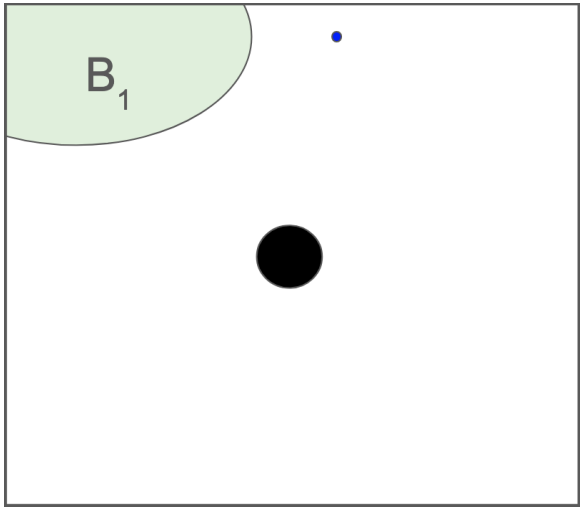
- Fix $u = u_i \in U$
- Barrier function: $B_i(\vec{x})$
- $\dot{\vec{x}} = f(\vec{x}, u_i)$



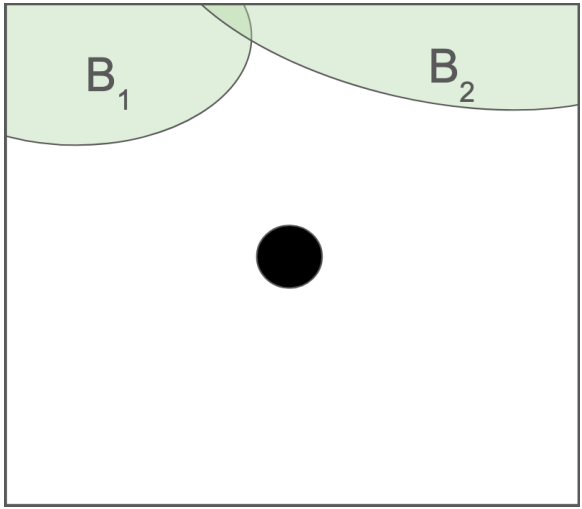

**BF $\rightarrow$ CBF ($u = u_i$)**

- **Easier computation [SOS]**
- **Very conservative**

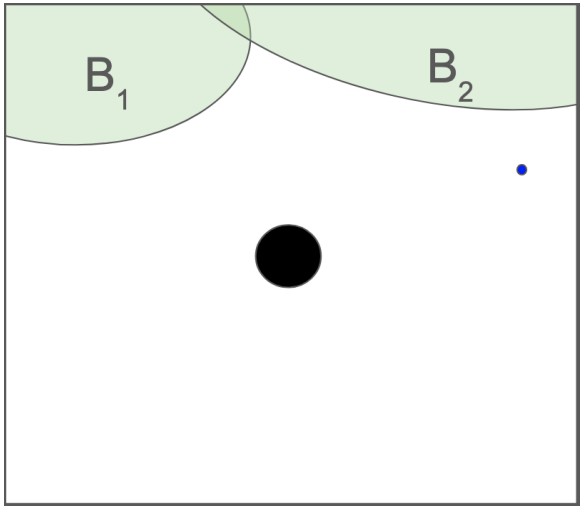
# Multiple Barriers



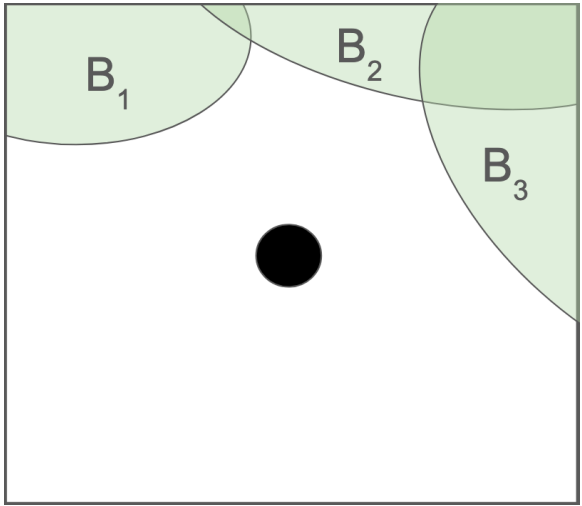
# Multiple Barriers



# Multiple Barriers



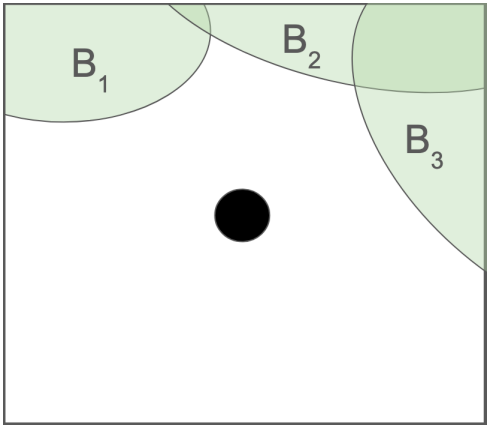
# Multiple Barriers



# Multiple Barriers

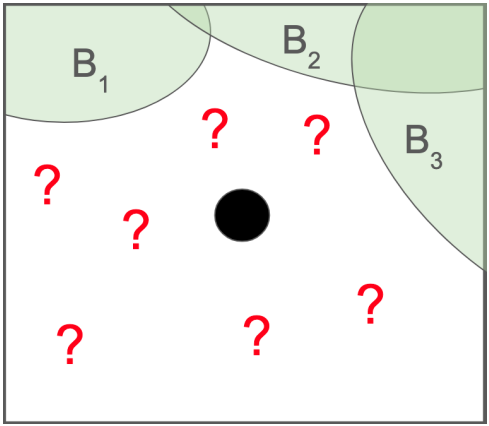
Combining Multiple Barriers:

- $\min(B_1, B_2, B_3) = B^{(1)}$
- Boolean Combination $\rightarrow$ Nonmsooth Analysis [Egerstedt]



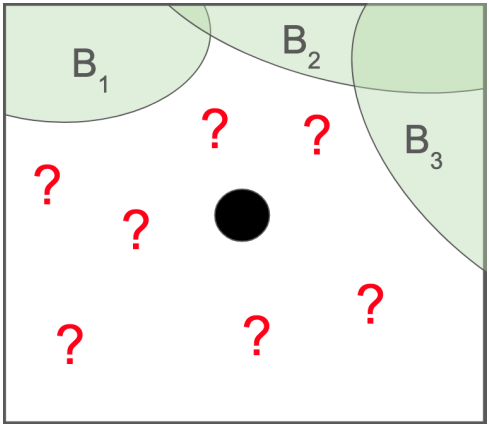
# Can we do better?

- We have multiple barriers,
- We have a controlled invariant (CI) region,
- Can we add more states to the CI region?



# Can we do better?

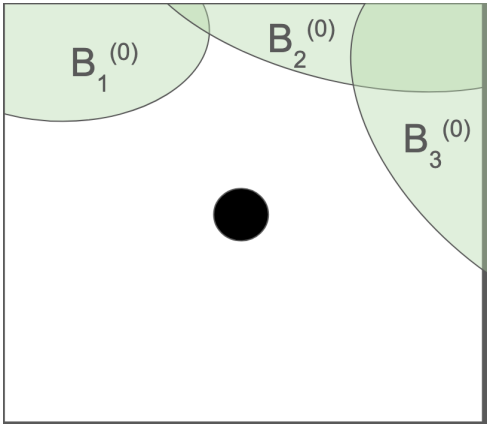
- We have multiple barriers,
- We have a controlled invariant (CI) region,
- Can we add more states to the CI region?




*Successive* barrier functions

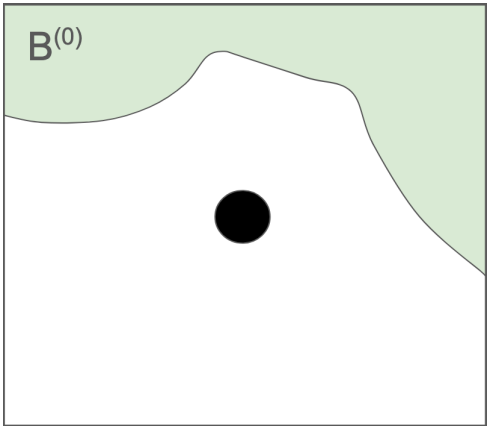
# *Successive* barrier functions

- $\nabla B_i \cdot f(\vec{x}, \vec{u_i}) \leq -\lambda B_i(\vec{x})$,
- Holds $\forall \vec{x} \in \mathbb{R}^n$



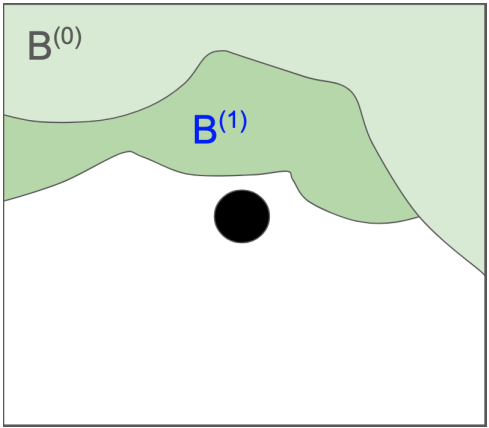
# *Successive* barrier functions

- $\nabla B_i \cdot f(\vec{x}, \vec{u}_i) \leq -\lambda B_i(\vec{x})$,
- Holds $\forall \vec{x} \in \mathbb{R}^n$



# *Successive* barrier functions

- $\nabla B_i \cdot f(\vec{x}, \vec{u}_i) \leq -\lambda B_i(\vec{x})$,
- ~~Holds $\forall \vec{x} \in \mathbb{R}^n$~~
- only when $B^{(0)}(\vec{x}) \geq 0$



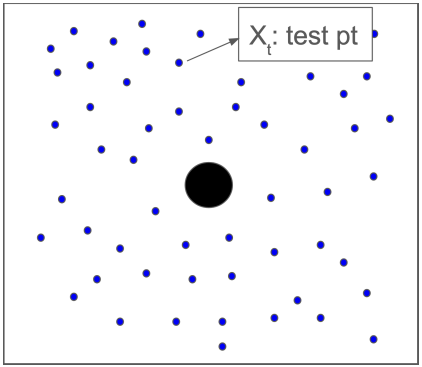
# Synthesis of Multiple and Successive CBFs

# Barrier Synthesis using SOS

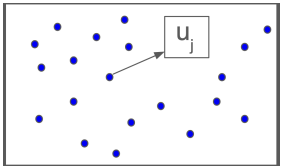
$$\text{Find} \quad B(\vec{x}) \; s.t. \\ \left.\begin{aligned} \forall \vec{x} \in X_u, B(\vec{x}) > 0 \\ \forall \vec{x} \in X_o, B(\vec{x}) < 0 \\ \forall \vec{x}, \nabla B(\vec{x}) \cdot f(\vec{x}) \leq -\lambda B(\vec{x}) \end{aligned}\right\}$$

Enforced using SOS
+
Putinar's Positivstellensatz
[Parillo et al.]

# Synthesis of Multiple Barriers

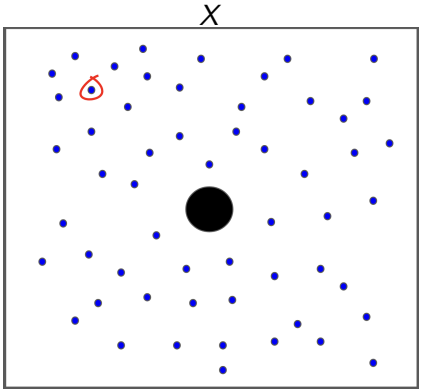


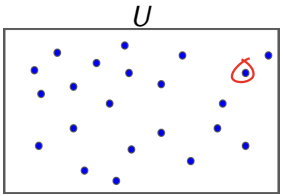

$x_t \in X$




$u_j \in U_{fin}$

# Synthesis of Multiple Barriers

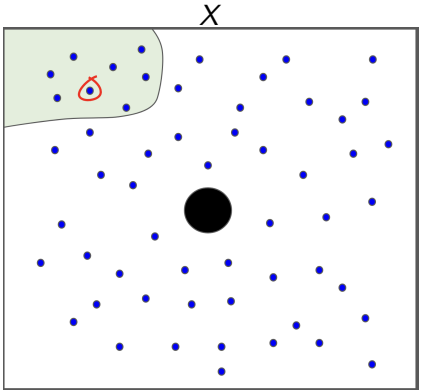


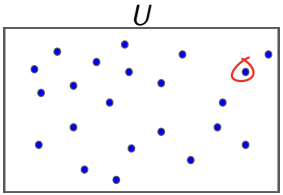




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers

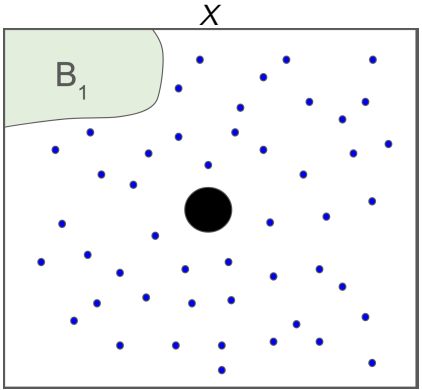


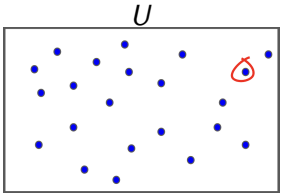




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers

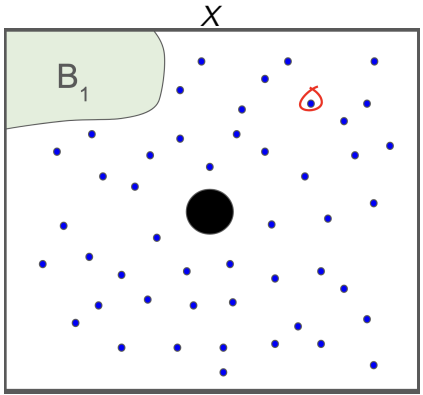


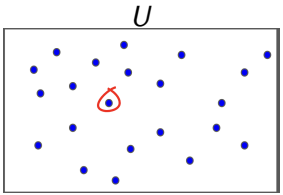




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers

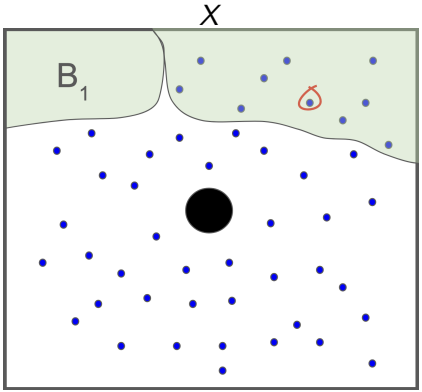


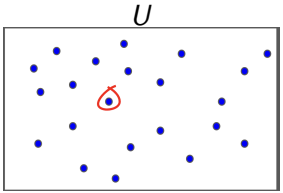




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers

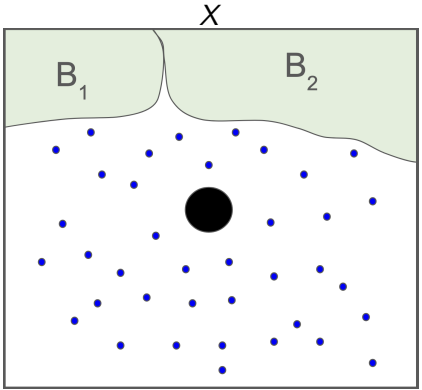


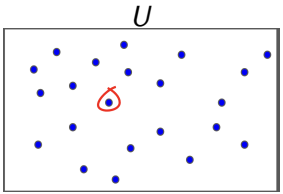




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers

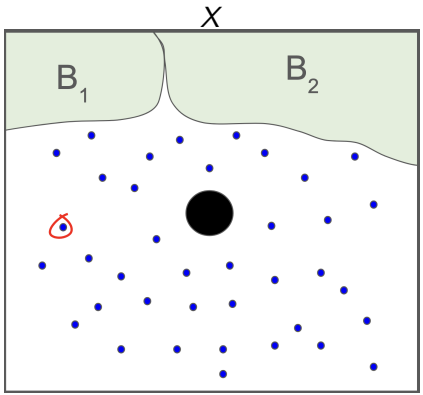


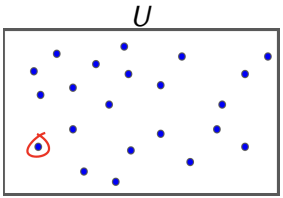




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers

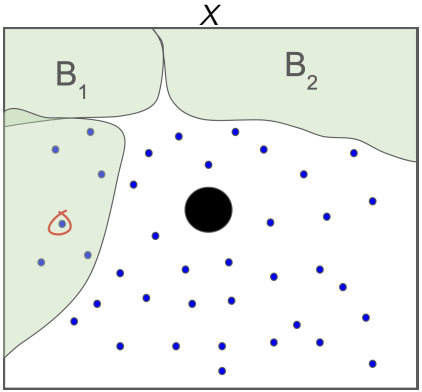


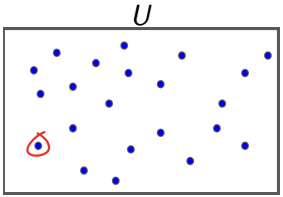




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers

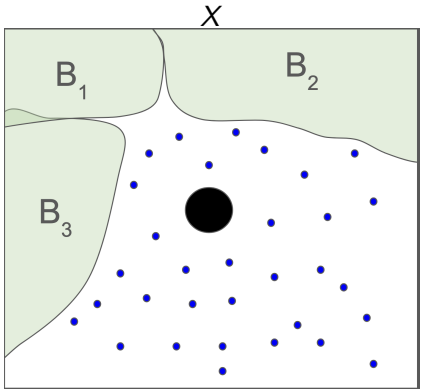


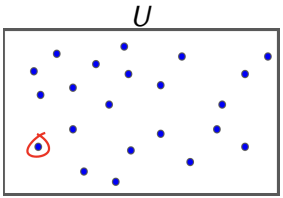




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers

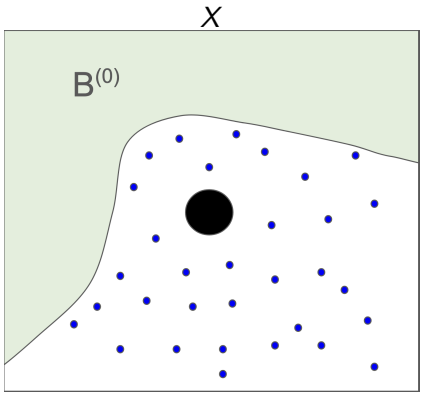


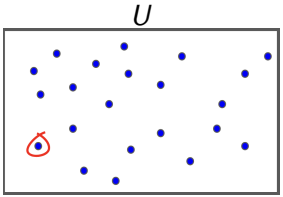




Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

# Synthesis of Multiple Barriers



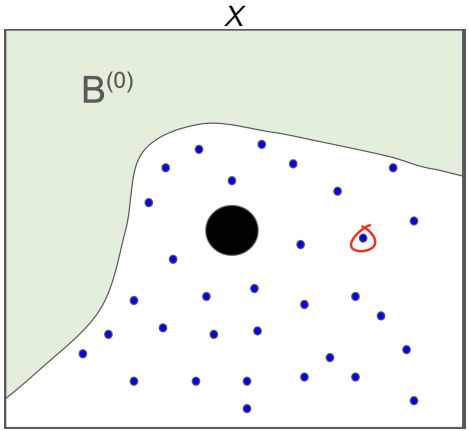



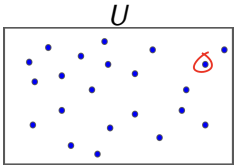

Synthesize a Barrier $B_j$:

- Fix $u = u_j \in U_{fin}$,
- $x_0 = x_t$

$$B^{(0)} = \min(B_1, \ldots, B_k)$$

# Synthesis of Successive Barriers



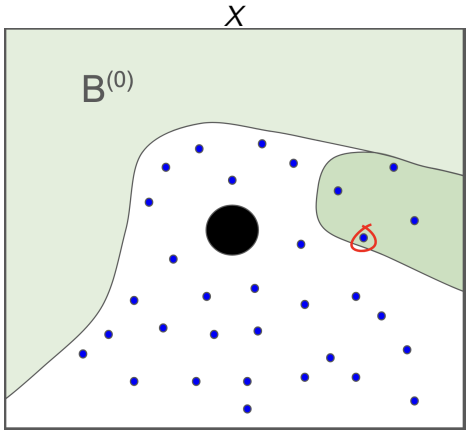



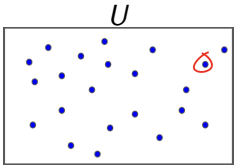

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $B^{(0)}(\vec{x}) \geq 0 \implies$
  $\nabla B_j \cdot f(\vec{x}, \vec{u}_j) \leq -\lambda B_j(\vec{x})$,

$$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)})$$

# Synthesis of Successive Barriers



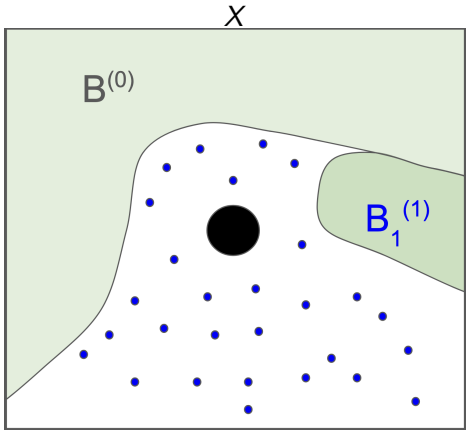



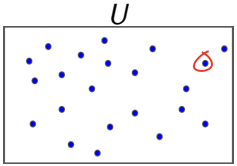

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $B^{(0)}(\vec{x}) \geq 0 \implies$
  $\nabla B_j \cdot f(\vec{x}, \vec{u}_j) \leq -\lambda B_j(\vec{x})$,

$$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)})$$

# Synthesis of Successive Barriers



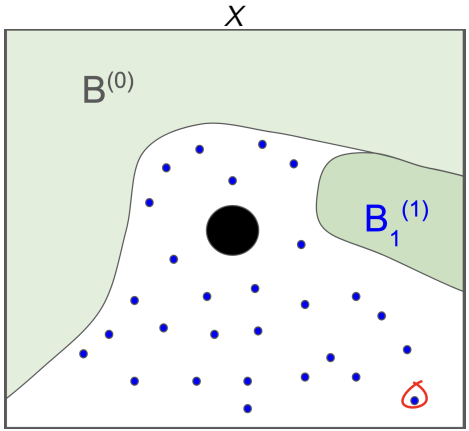



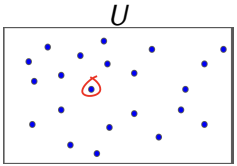

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $B^{(0)}(\vec{x}) \geq 0 \implies$
  $\nabla B_j \cdot f(\vec{x}, \vec{u}_j) \leq -\lambda B_j(\vec{x})$,

$$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)})$$

# Synthesis of Successive Barriers



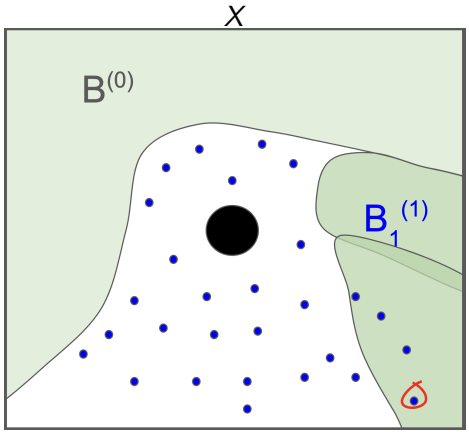



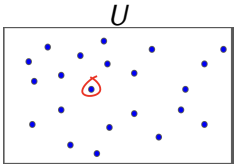

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $B^{(0)}(\vec{x}) \geq 0 \implies$
  $\nabla B_j \cdot f(\vec{x}, \vec{u}_j) \leq -\lambda B_j(\vec{x})$,

$$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)})$$

# Synthesis of Successive Barriers



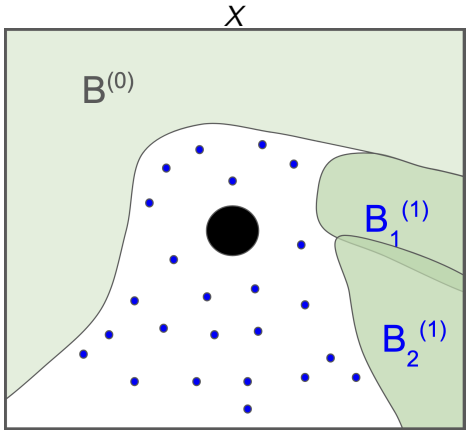



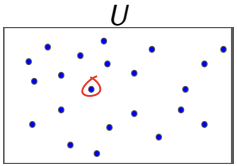

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $B^{(0)}(\vec{x}) \geq 0 \implies$
  $\nabla B_j \cdot f(\vec{x}, \vec{u}_j) \leq -\lambda B_j(\vec{x})$,

$$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)})$$

# Synthesis of Successive Barriers



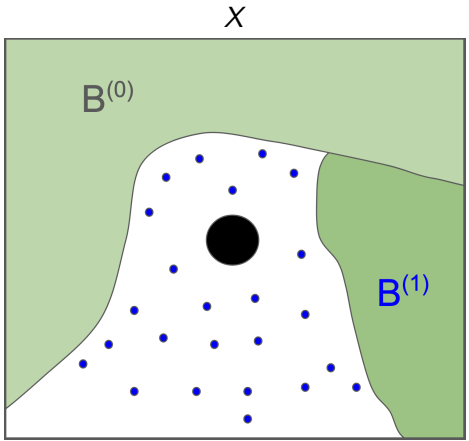



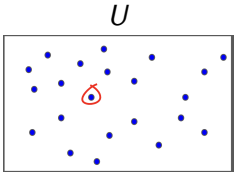

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $B^{(0)}(\vec{x}) \geq 0 \implies$
  $\nabla B_j \cdot f(\vec{x}, \vec{u}_j) \leq -\lambda B_j(\vec{x})$,

$$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)})$$

# Synthesis of Successive Barriers



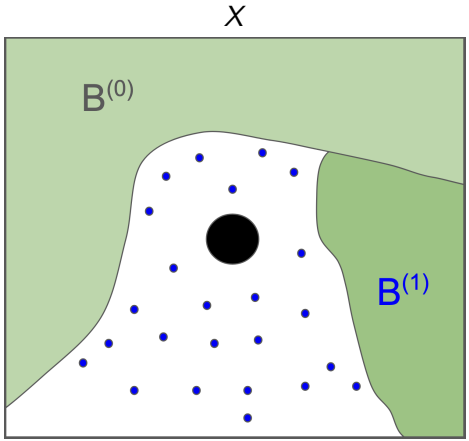



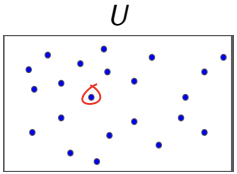

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $\begin{cases} B^{(0)}(\vec{x}) \geq 0 \implies \\ \quad \nabla B_j \cdot f(\vec{x}, \vec{u}_j) \leq -\lambda B_j(\vec{x}) \end{cases}$

$$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)}), B^{(1)} = \min(B_1^{(1)}, \ldots, B_k^{(1)})$$

# Synthesis of Successive Barriers



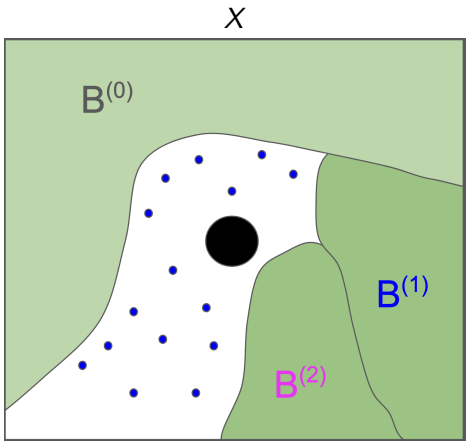



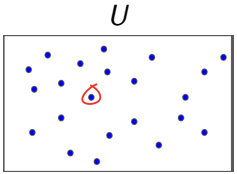

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $\begin{cases} B^{(0)}(\vec{x}) \geq 0 \wedge B^{(1)}(\vec{x}) \geq 0 \implies \\ \nabla B_j \cdot f(\vec{x}, \vec{u}_j) \leq -\lambda B_j(\vec{x}) \end{cases}$

$$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)}), B^{(1)} = \min(B_1^{(1)}, \ldots, B_k^{(1)})$$

# Synthesis of Successive Barriers



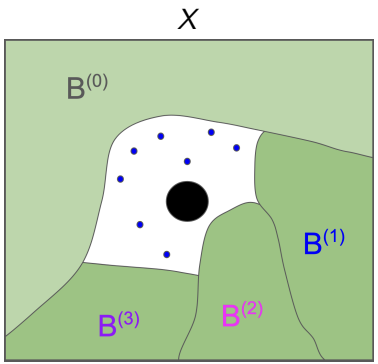



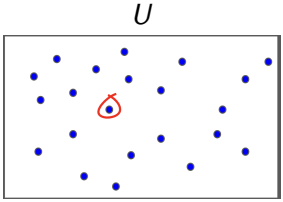

Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $\begin{cases} B^{(0)}(\vec{x}) \geq 0 \wedge B^{(1)}(\vec{x}) \geq 0 \implies \\ \nabla B_j \cdot f(\vec{x}, \vec{u_j}) \leq -\lambda B_j(\vec{x}) \end{cases}$

$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)}), B^{(1)} = \min(B_1^{(1)}, \ldots, B_k^{(1)}), B^{(2)} = \min(B_1^{(2)}, \ldots, B_k^{(2)})$

# Synthesis of Successive Barriers




Synthesize a Successive Barrier $B_j^k$:

- Fix $u = u_j \in U_{fin}$, $x_0 = x_t$
- $\begin{cases} B^{(0)}(\vec{x}) \geq 0 \wedge B^{(1)}(\vec{x}) \geq 0 \wedge B^{(2)}(\vec{x}) \geq 0 \implies \\ \nabla B_j \cdot f(\vec{x}, \vec{u_j}) \leq -\lambda B_j(\vec{x}) \end{cases}$

$B^{(0)} = \min(B_1^{(0)}, \ldots, B_k^{(0)}), B^{(1)} = \min(B_1^{(1)}, \ldots, B_k^{(1)}), B^{(2)} = \min(B_1^{(2)}, \ldots, B_k^{(2)})$

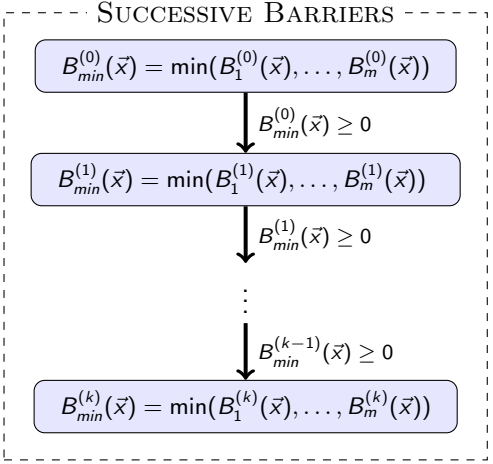
# Monitoring

# Monitoring

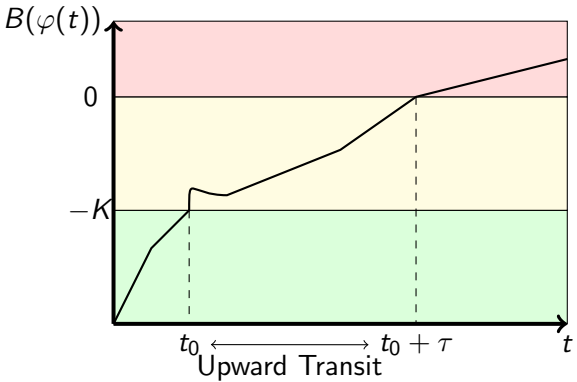
- Monitoring using QP [Ames et al.]
- QP is solved in "continuous time"
- Dwell time bounds [Breeden et al.]

# Monitoring

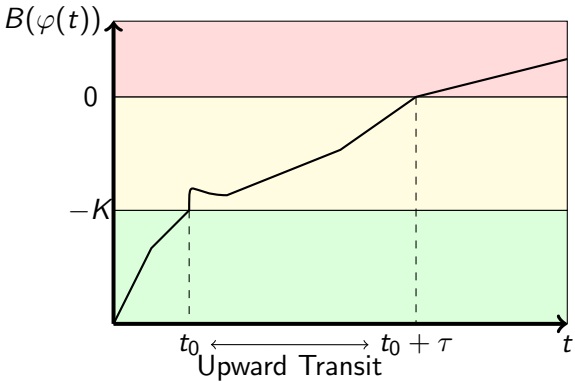
- Monitoring using QP [Ames et al.]
- QP is solved in "continuous time"
- Dwell time bounds [Breeden et al.]



# Monitor Synthesis

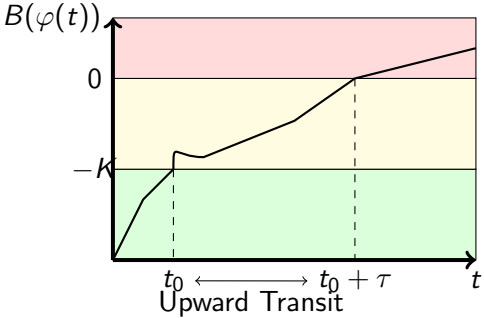


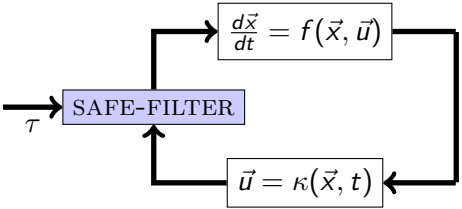
# Monitor Synthesis



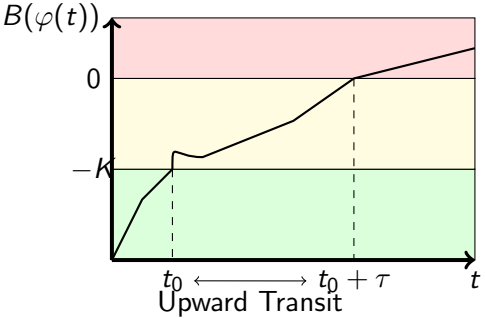

- unsafe region; $B(\vec{x}) \geq 0$
- transit region; $B(\vec{x}) \in [-K, 0)$
- safe region; $B(\vec{x}) < -K$

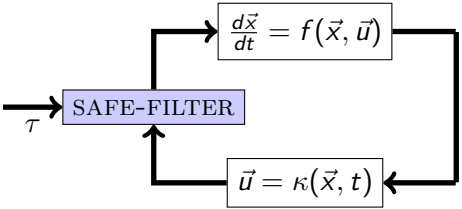
# Monitor Synthesis



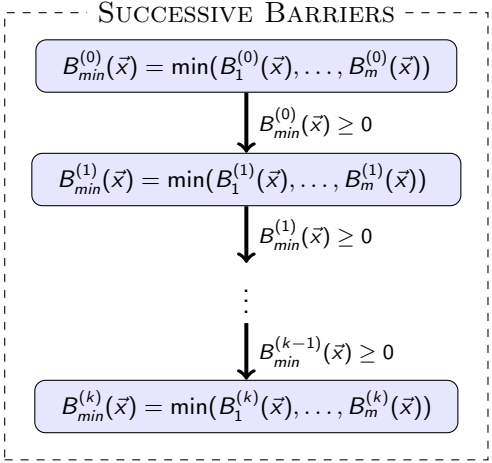
# Monitor Synthesis



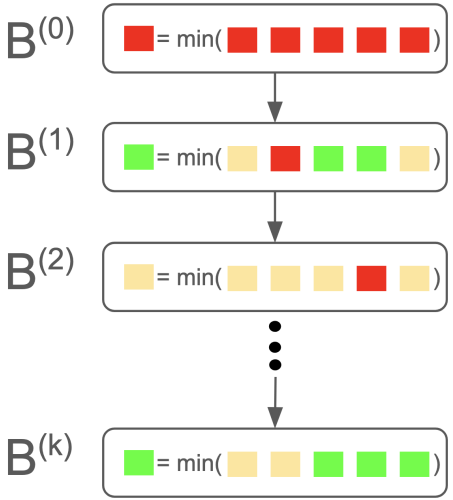




$$\text{SAFE-FILTER}(\vec{x}; B_{min}) = \begin{cases} \text{PASS} & \text{if } B_{min}(\vec{x}) < -K \\ \text{OVERRIDE}(\vec{u}_{min}(\vec{x})) & \text{otherwise} \end{cases}$$

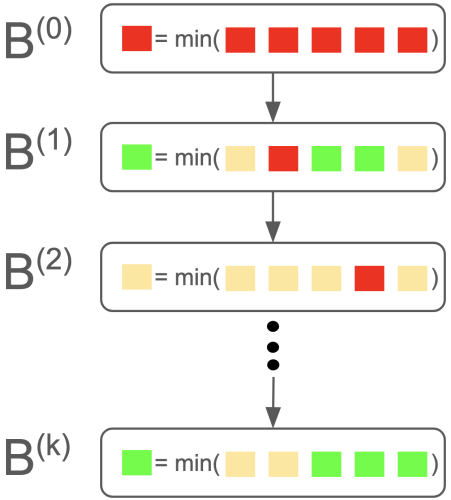
# Monitor Synthesis



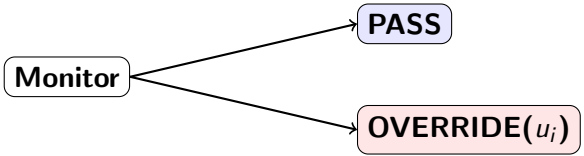
# Monitor Synthesis



# Monitor Synthesis




$\rightarrow$ Synthesizing a finite state machine:



*Evaluations*

# Evaluations - 2D

2D Nonlinear Dynamics with 2 control inputs $u_1, u_2 \in [-0.1, 0.1]^2$;

$$\dot{x}_1 = \frac{1}{2}x_1 - \frac{1}{5}x_2 - \frac{1}{100}x_1x_2 - \frac{1}{2}u_1 + \frac{1}{2}u_2,$$
$$\dot{x}_2 = x_1 - \frac{2}{5}x_2 - \frac{1}{20}x_2^2 - \frac{7}{10}u_2 + \frac{1}{10}u_1,$$

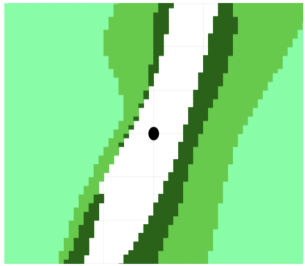


Figure: Improvement in CI region with three iterations of successive barrier functions

# Evaluations - 4D

4D Nonlinear Dynamics with 2 control inputs $u_1, u_2 \in [-0.1, 0.1]^2$;

$$
\begin{aligned}
\dot{x}_1 &= x_2, \\
\dot{x}_2 &= \frac{1}{2}x_1 - \frac{1}{5}x_2 + \frac{1}{20}x_3 x_1 - \frac{1}{100}x_1 x_2 - \frac{1}{2}u_1, \\
\dot{x}_3 &= x_4, \\
\dot{x}_4 &= -\frac{2}{5}x_4 + \frac{1}{5}x_1 - \frac{1}{20}x_3^2 - \frac{7}{10}u_2,
\end{aligned}
$$

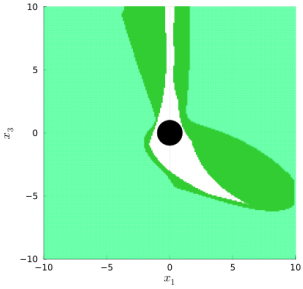



Figure: Improvement in CI region with two iterations of successive barrier functions

# Evaluations - 5D

5D coordinated turn model with 2 control inputs $u_1, u_2 \in [-5, 5]^2$;

$$\begin{aligned}
\dot{x}_1 &= x_3 \cos x_4, \\
\dot{x}_2 &= x_3 \sin x_4, \\
\dot{x}_3 &= u_1, \\
\dot{x}_4 &= x_5, \\
\dot{x}_5 &= u_2,
\end{aligned}$$

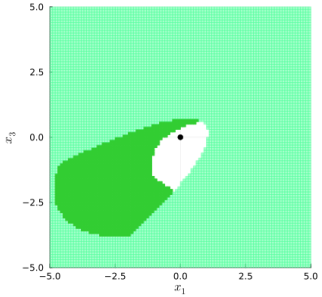
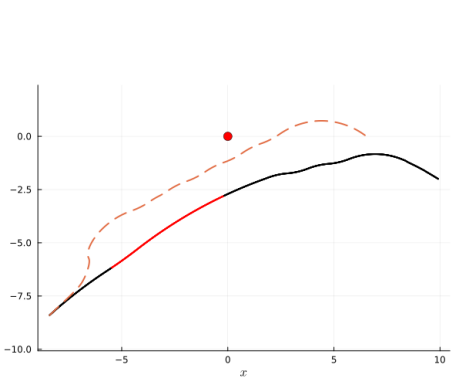
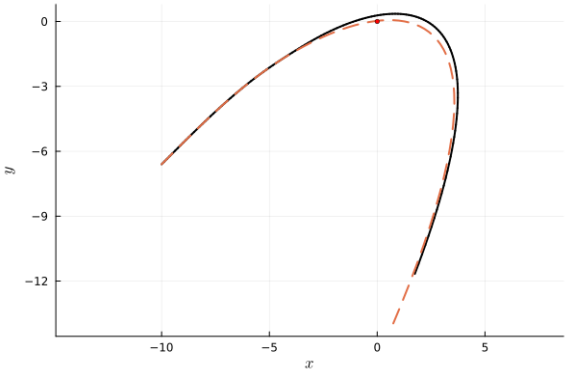



Figure: Improvement in CI region with two iterations of successive barrier functions

# Evaluations - Trajectories




2D model




6D model(planar quadrotor)

Figure: Effect of implementing the safety filter. Dashed lines show the original behavior.

# Comparison with FOSSIL

| system | dim | inputs | FOSSIL success | FOSSIL time(s) | Ours success | Ours time(s):$B^{(1)}$ | Ours # barriers |
|---|---|---|---|---|---|---|---|
| poly1 | 2 | 1 | ✓ | 3.2 | ✓ | **0.6** | 2 |
| poly2 | 2 | 1 | ✓ | 1.9 | ✓ | **1.0** | 2 |
| van der Pol | 2 | 1 | ✓ | 4.8 | ✓ | **2.0** | 2 |
| inv pendulum | 2 | 1 | ✓ | 3.2 | ✓ | **2.0** | 2 |
| poly3 | 2 | 2 | ✓ | **1.3** | ✓ | 7.6 | 4 |
| poly4 | 3 | 2 | ✓ | 74.8 | ✓ | **6.8** | 4 |
| poly5 | 4 | 2 | ✗ | - | ✓ | **36.4** | 4 |
| coord turn | 5 | 2 | ✗ | - | ✓ | **108.4** | 4 |
| planar multirotor | 6 | 2 | ✗ | - | ✓ | **131.6** | 4 |

# Comparison with FOSSIL

| system | dimensions | inputs | F | S | $\neg F \wedge S$ | $F \wedge \neg S$ |
|---|---|---|---|---|---|---|
| poly1 | 2 | 1 | 417 | **649** | **375** | 143 |
| poly2 | 2 | 1 | 716 | **922** | **215** | 9 |
| van der Pol | 2 | 1 | 507 | **649** | **322** | 180 |
| inv pendulum | 2 | 1 | 683 | **798** | **193** | 78 |
| poly3 | 2 | 2 | 556 | **977** | **421** | 0 |

# Comparison with FOSSIL

| system | dimensions | inputs | F | S | $\neg F \wedge S$ | $F \wedge \neg S$ |
|---|---|---|---|---|---|---|
| poly1 | 2 | 1 | 417 | **649** | **375** | 143 |
| poly2 | 2 | 1 | 716 | **922** | **215** | 9 |
| van der Pol | 2 | 1 | 507 | **649** | **322** | 180 |
| inv pendulum | 2 | 1 | 683 | **798** | **193** | 78 |
| poly3 | 2 | 2 | 556 | **977** | **421** | 0 |

# Comparison with FOSSIL

| system | dimensions | inputs | F | S | $\neg F \wedge S$ | $F \wedge \neg S$ |
|---|---|---|---|---|---|---|
| poly1 | 2 | 1 | 417 | **649** | **375** | 143 |
| poly2 | 2 | 1 | 716 | **922** | **215** | 9 |
| van der Pol | 2 | 1 | 507 | **649** | **322** | 180 |
| inv pendulum | 2 | 1 | 683 | **798** | **193** | 78 |
| poly3 | 2 | 2 | 556 | **977** | **421** | 0 |

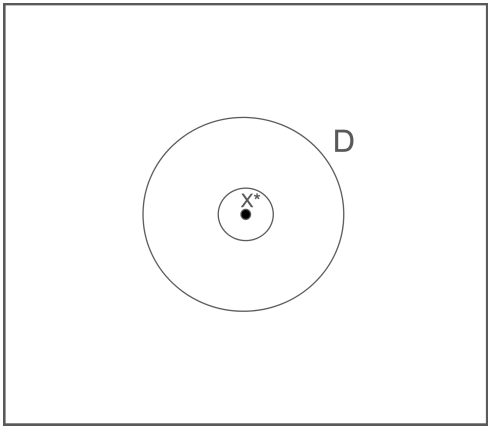
*Successive Control Lyapunov Functions*

# Control Lyapunov Functions

- State: $\vec{x} \in \mathbb{R}^n$
- Control inputs: $\vec{u} \in \mathbb{R}^m$
- $\dot{\vec{x}} = f(\vec{x}, \vec{u})$, $X \subseteq \mathbb{R}^n$,
- $\exists \vec{u}, \forall X, f(\vec{x^*}, \vec{u^*}) = 0$,
  - $V(\vec{x}) \geq 0$ for all $\vec{x} \in D$ (V is **positive definite** with $V(\vec{x^*}) = 0$)
  - for all $\vec{x} \in D \backslash \vec{x^*}$ there **exists a control input** $\vec{u} \in U$ s.t. $\nabla V(\vec{x}) \cdot f(\vec{x}, \vec{u}) < 0$

# Control Lyapunov Functions

- $\dot{\vec{x}} = f(\vec{x}, \vec{u})$, $X \subseteq \mathbb{R}^n$,
- $\exists \vec{u}, \forall X, f(\vec{x}^*, \vec{u}^*) = 0$,
- $u_i \in U_{fin} \subset U$

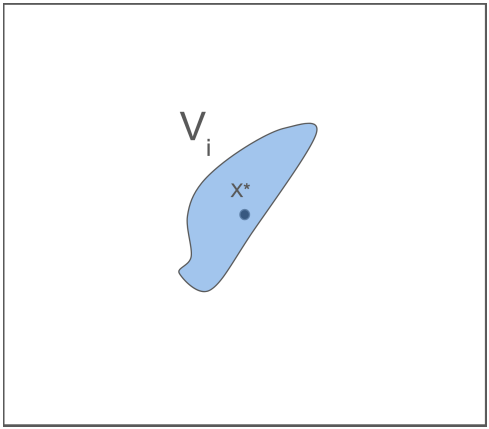



- $V(\vec{x}) \geq \varepsilon \|\vec{x}\|_2^2$ for all $\vec{x} \in D$ $(V(\vec{x}^*) = 0)$
- for all $\vec{x} \in D \backslash \vec{x}^*$ there **exists a control input** $\vec{u} \in U$ s.t. $\nabla V(\vec{x}) \cdot f(\vec{x}, \vec{u}) \leq -\varepsilon \|\vec{x}\|_2^2$

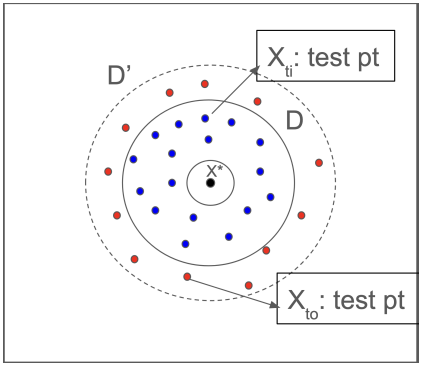
# One CLF is not enough

Computing CLF:

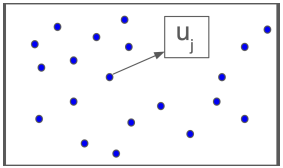
- Fix $u = u_i \in U$
- $\dot{\vec{x}} = f(\vec{x}, u_i)$
- Lyapunov function: $V_i(\vec{x})$



# Synthesis of Multiple CLFs

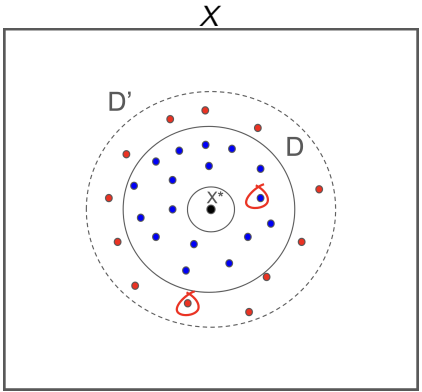


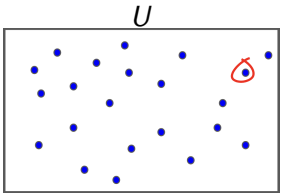

$x_{ti}, x_{to} \in X$




$u_j \in U_{fin}$

# Synthesis of Multiple CLFs

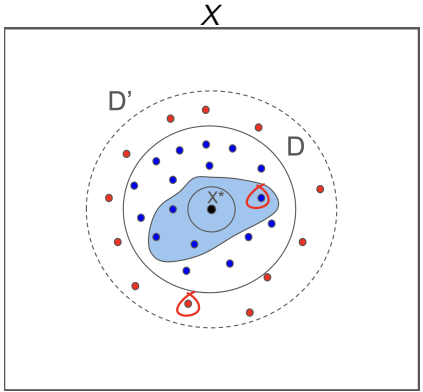


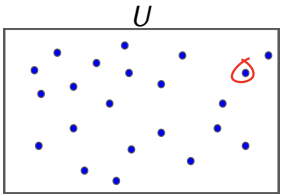




Synthesize a CLF $V_j$:

- Fix $u = u_j \in U_{fin}$,
- $V_j(x_{ti}) \leq \gamma_j$, $V_j(x_{to}) \geq \gamma_j$

# Synthesis of Multiple CLFs

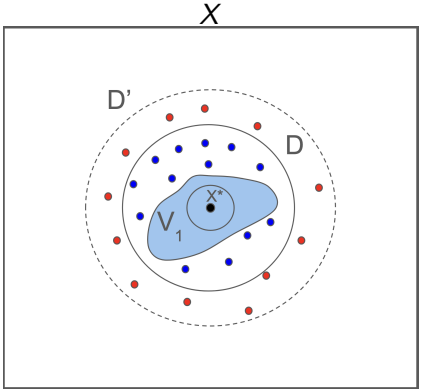


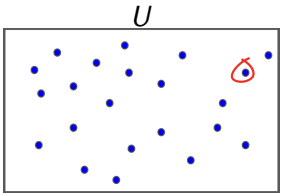




Synthesize a CLF $V_j$:

- Fix $u = u_j \in U_{fin}$,
- $V_j(x_{ti}) \leq \gamma_j$, $V_j(x_{to}) \geq \gamma_j$

# Synthesis of Multiple CLFs

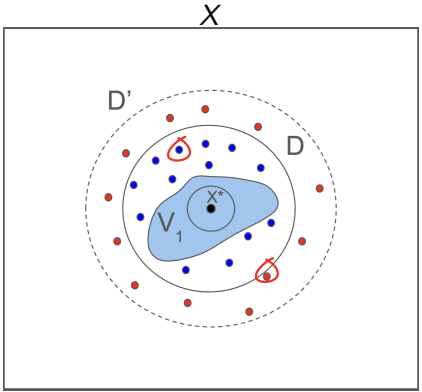


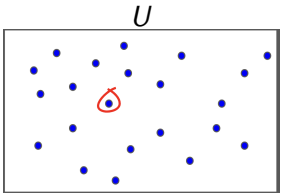




Synthesize a CLF $V_j$:

- Fix $u = u_j \in U_{fin}$,
- $V_j(x_{ti}) \leq \gamma_j$, $V_j(x_{to}) \geq \gamma_j$

# Synthesis of Multiple CLFs

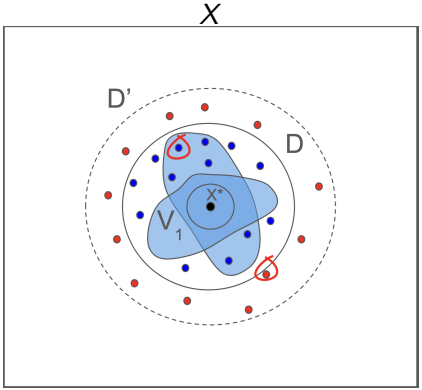


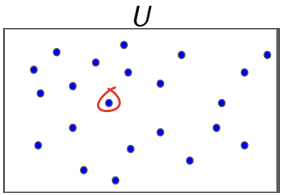




Synthesize a CLF $V_j$:

- Fix $u = u_j \in U_{fin}$,
- $V_j(x_{ti}) \leq \gamma_j$, $V_j(x_{to}) \geq \gamma_j$

# Synthesis of Multiple CLFs

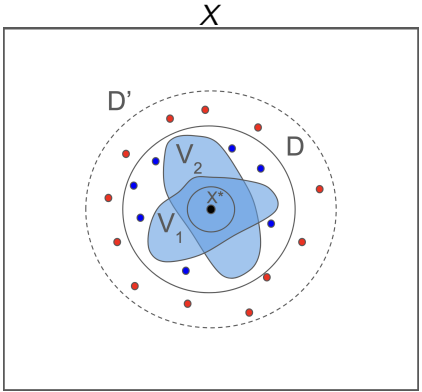


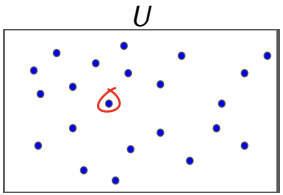




Synthesize a CLF $V_j$:

- Fix $u = u_j \in U_{fin}$,
- $V_j(x_{ti}) \leq \gamma_j$, $V_j(x_{to}) \geq \gamma_j$

# Synthesis of Multiple CLFs

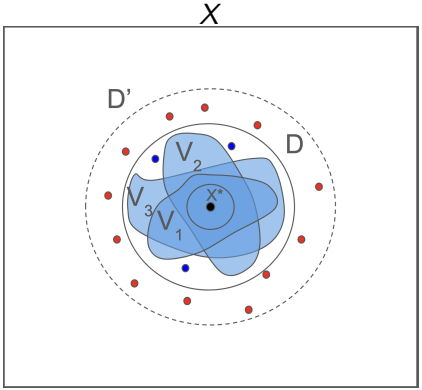


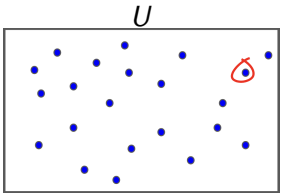




Synthesize a CLF $V_j$:

- Fix $u = u_j \in U_{fin}$,
- $V_j(x_{ti}) \leq \gamma_j$, $V_j(x_{to}) \geq \gamma_j$

# Synthesis of Multiple CLFs

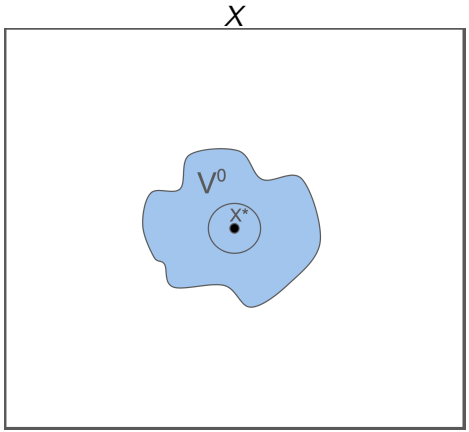


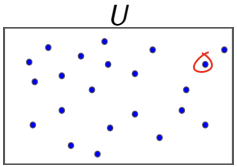




Synthesize a CLF $V_j$:

- Fix $u = u_j \in U_{fin}$,
- $V_j(x_{ti}) \leq \gamma_j$, $V_j(x_{to}) \geq \gamma_j$

# Synthesis of Successive CLFs

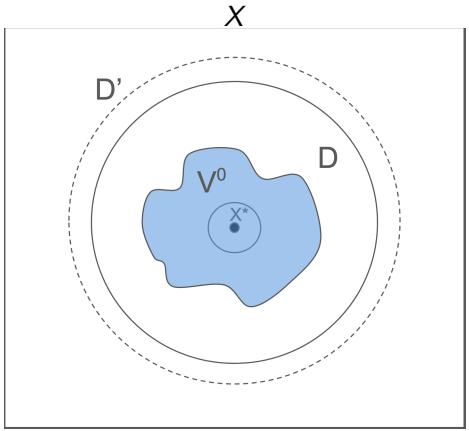


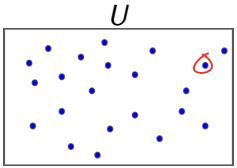




Synthesize a Successive CLF $B_j^k$:

- Fix $u = u_j \in U_{fin}$,
- $V^{(0)}(\vec{x}) \geq 1 \implies$
  $\nabla V_i(\vec{x}) \cdot f(\vec{x}, u_i) \leq -\varepsilon \|\vec{x}\|_2^2$,

# Synthesis of Successive CLFs

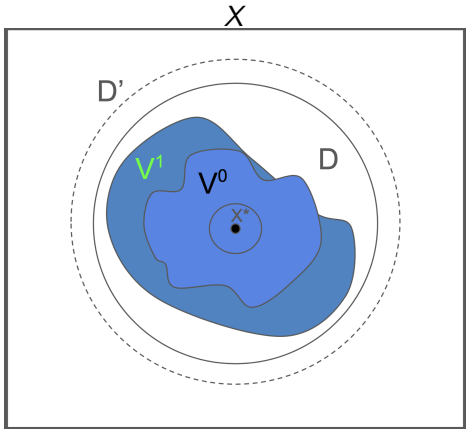


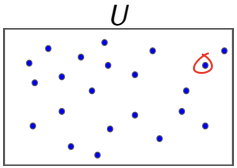




Synthesize a Successive CLF $B_j^k$:

- Fix $u = u_j \in U_{fin}$,
- $V^{(0)}(\vec{x}) \geq 1 \implies$
  $\nabla V_i(\vec{x}) \cdot f(\vec{x}, u_i) \leq -\varepsilon \|\vec{x}\|_2^2$,

# Synthesis of Successive CLFs

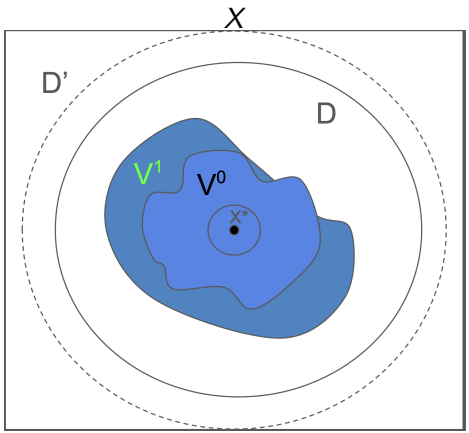


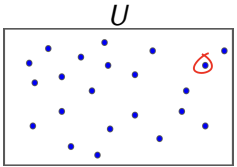




Synthesize a Successive CLF $B_j^k$:

- Fix $u = u_j \in U_{fin}$,
- $V^{(0)}(\vec{x}) \geq 1 \implies$
  $\nabla V_i(\vec{x}) \cdot f(\vec{x}, u_i) \leq -\varepsilon \|\vec{x}\|_2^2$,

# Synthesis of Successive CLFs

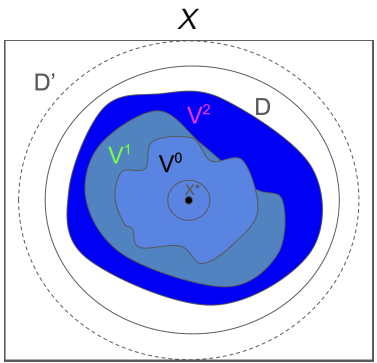


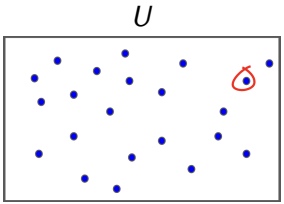




Synthesize a Successive CLF $B_j^k$:

- Fix $u = u_j \in U_{fin}$,
- $V^{(0)}(\vec{x}) \geq 1 \wedge V^{(1)}(\vec{x}) \geq 1 \implies \nabla V_i(\vec{x}) \cdot f(\vec{x}, u_i) \leq -\varepsilon \|\vec{x}\|_2^2$,

# Synthesis of Successive CLFs




Synthesize a Successive CLF $B_j^k$:

- Fix $u = u_j \in U_{fin}$,
- $\begin{cases} V^{(0)}(\vec{x}) \geq 1 \wedge V^{(1)}(\vec{x}) \geq 1 \wedge V^{(2)}(\vec{x}) \geq 1 \implies \\ \nabla V_i(\vec{x}) \cdot f(\vec{x}, u_i) \leq -\varepsilon \|\vec{x}\|_2^2 \end{cases}$

# Evaluations - ZubovDavidson

2D Nonlinear Dynamics with control input $u \in [-0.1, 0.1]$;

$$\dot{x}_1 = -x_1 + 2(x_1^2 x_2),$$
$$\dot{x}_2 = -x_2 + u,$$

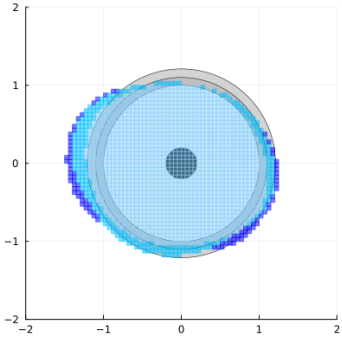



Figure: Improvement in RoA with three iterations of successive CLFs

# Evaluations - LotkaVoltera

2D Nonlinear Dynamics with control input $u \in [-0.1, 0.1]$;

$$\dot{x}_1 = -0.42x_1 - 1.05x_2 - 2.3x_1^2 - 0.5x_1x_2 - x_1^3,$$
$$\dot{x}_2 = 1.98x_1 + x_1x_2 + u,$$

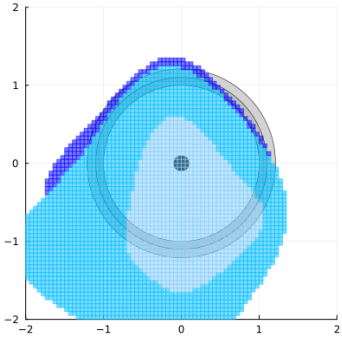



Figure: Improvement in RoA with three iterations of successive CLFs

# Evaluations - 3D

3D Nonlinear Dynamics with control input $u \in [-1, 1]$;

$$\dot{x}_1 = 1 - \frac{x_3^2}{2},$$
$$\dot{x}_2 = x_3 - \frac{x_3^3}{6},$$
$$\dot{x}_3 = u$$

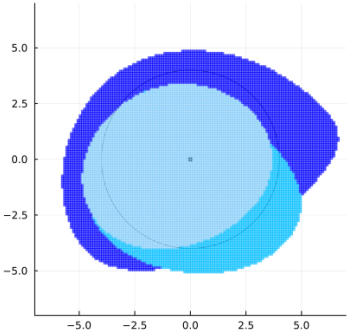



Figure: Improvement in RoA with three iterations of successive CLFs

# Evaluations - WheeledPathFollower

3D Nonlinear Dynamics with control input $u \in [-1, 1]$;

$$\dot{x}_1 = v\frac{1 - x_3^2/2}{1 - x_2},$$
$$\dot{x}_2 = v(x_3 - x_3^3/6),$$
$$\dot{x}_3 = v\frac{u + u^3/3}{L} - v\frac{1 - x_3^2/2}{1 - x_2}$$

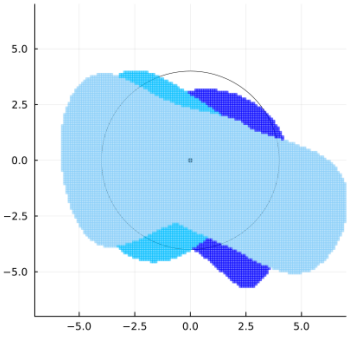



Figure: Improvement in RoA with three iterations of successive CLFs

*Certifying Sum of Squares*

# Certifying SOS Programs

Verify that numerical issues do not invalidate the SOS programming results.

- Each barrier has multiple entailment relations:

$$p_1(\vec{x}) \geq 0, \ldots, p_m(\vec{x}) \geq 0 \;\models\; p \geq 0\,,$$

- Certify via a Putinar positivstellensatz proof that states that

$$\exists\ \sigma_1, \ldots, \sigma_m \in \mathrm{SOS}_d[\vec{x}]\ p - \sigma_1 p_1 - \cdots - \sigma_m p_m \in \mathrm{SOS}_d[\vec{x}]\,,$$

($\mathrm{SOS}_d[\vec{x}]$ represents the set of all SOS polynomials over $\vec{x}$ of degree at most $d$)

# Certifying SOS Programs

$$\begin{cases} B_i(\vec{x}) > 0; \forall \vec{x} \in X_u \\ B_i(\vec{x}) \leq 0; \forall \vec{x} \in X_i \\ \nabla B_i(\vec{x}) \cdot f(\vec{x}, \vec{u}) \leq \lambda B_i(\vec{x}) \end{cases} \qquad \begin{array}{l} B_i(\vec{x}) \equiv \Sigma \alpha_i p_i + \alpha_0 \\ -B_i(\vec{x}) \equiv \Sigma \beta_i q_i + \beta_0 \\ \nabla B_i(\vec{x}) \cdot f(\vec{x}, \vec{u}) - \lambda B_i(\vec{x}) \equiv \Sigma \sigma_i r_i + \sigma_0 \end{array}$$

$$\alpha_i, \beta_i, \sigma_i, \ldots \implies m(\vec{x})^\top Q_i m(\vec{x})$$

$Q_i$ should be **positive semi-definite**

# Certifying SOS Programs

How to certify:

- output the polynomials $\sigma_1, \ldots, \sigma_m$
- compute the "residue" $p - \sigma_1 p_1 - \cdots - \sigma_m p_m$
- obtain a representation $\sigma_i = m(\vec{x})^\top Q_i m(\vec{x})$
- verify that $Q_i$ is positive semi-definite by computing its Cholesky decomposition

The C++ library *Eigen* was used to carry out the Cholesky decomposition using 512 bit floating point representation

# Robust Sum of Squares

$$\begin{cases} B_i(\vec{x}) > 0; \forall \vec{x} \in X_u \\ B_i(\vec{x}) \leq 0; \forall \vec{x} \in X_i \\ \nabla B_i(\vec{x}) \cdot f(\vec{x}, \vec{u}) \leq \lambda B_i(\vec{x}) \end{cases} \qquad \begin{array}{l} B_i(\vec{x}) \equiv \Sigma \alpha_i p_i + \alpha_0 \\ -B_i(\vec{x}) \equiv \Sigma \beta_i q_i + \beta_0 \\ \nabla B_i(\vec{x}) \cdot f(\vec{x}, \vec{u}) - \lambda B_i(\vec{x}) \equiv \Sigma \sigma_i r_i + \sigma_0 \end{array}$$

$$\alpha_i, \beta_i, \sigma_i, \ldots \implies m(\vec{x})^\top Q_i m(\vec{x})$$

$Q_i$ should be **positive semi-definite**

# Robust Sum of Squares

$$\begin{cases} B_i(\vec{x}) > 0; \forall \vec{x} \in X_u \\ B_i(\vec{x}) \leq 0; \forall \vec{x} \in X_i \\ \nabla B_i(\vec{x}) \cdot f(\vec{x}, \vec{u}) \leq \lambda B_i(\vec{x}) \end{cases} \qquad \begin{aligned} B_i(\vec{x}) &\equiv \Sigma \alpha_i p_i + \alpha_0 + \alpha_{DSOS} \\ -B_i(\vec{x}) &\equiv \Sigma \beta_i q_i + \beta_0 + \beta_{DSOS} \\ \nabla B_i(\vec{x}) \cdot f(\vec{x}, \vec{u}) - \lambda B_i(\vec{x}) &\equiv \Sigma \sigma_i r_i + \sigma_0 + \sigma_{DSOS} \end{aligned}$$

$$\alpha_i, \beta_i, \sigma_i, \ldots \implies m(\vec{x})^\top Q_i m(\vec{x})$$

$Q_i$ should be **positive semi-definite**

# Robust Sum of Squares

$$\begin{cases} B_i(\vec{x}) > 0; \forall \vec{x} \in X_u \\ B_i(\vec{x}) \leq 0; \forall \vec{x} \in X_i \\ \nabla B_i(\vec{x}) \cdot f(\vec{x}, \vec{u}) \leq \lambda B_i(\vec{x}) \end{cases} \qquad \begin{aligned} B_i(\vec{x}) &\equiv \Sigma \alpha_i p_i + \alpha_0 + \alpha_{DSOS} \\ -B_i(\vec{x}) &\equiv \Sigma \beta_i q_i + \beta_0 + \beta_{DSOS} \\ \nabla B_i(\vec{x}) \cdot f(\vec{x}, \vec{u}) - \lambda B_i(\vec{x}) &\equiv \Sigma \sigma_i r_i + \sigma_0 + \sigma_{DSOS} \end{aligned}$$

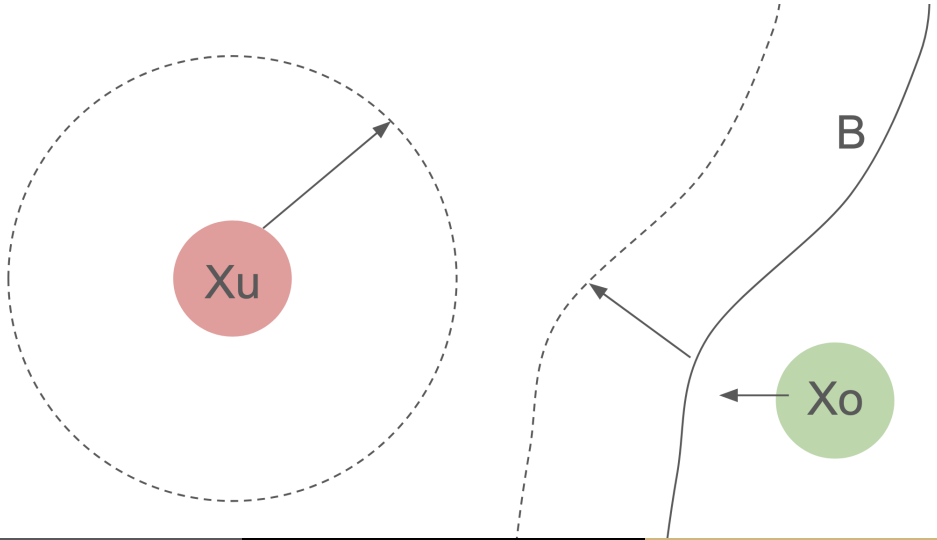
$$\alpha_i, \beta_i, \sigma_i, \ldots \implies m(\vec{x})^\top Q_i m(\vec{x})$$

$Q_i$ should be **positive semi-definite**

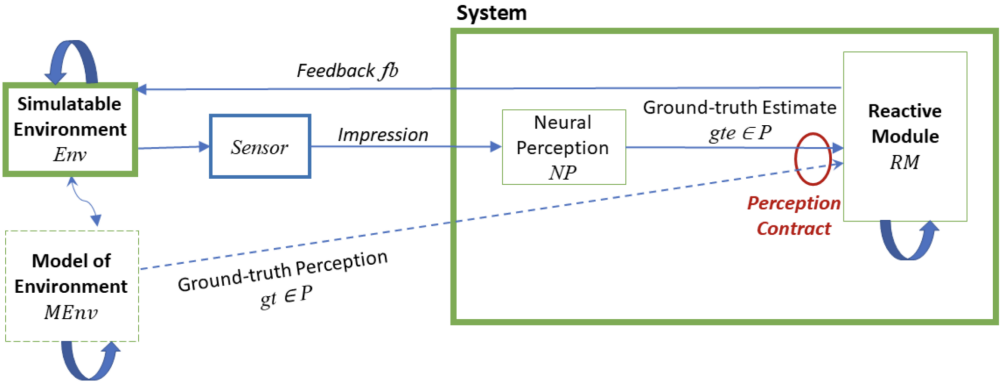
**Can we automate these proofs in LEAN?**

*Proposed Work*

# Future: Moving Obstacles



# Future: Demand-driven Perception Contracts




[0]Astorga, A., Hsieh, C., Madhusudan, P., & Mitra, S. (2023). Perception contracts for safety of ML-enabled systems. Proceedings of the ACM on Programming Languages, 7(OOPSLA2), 2196-2223

# Future: What is a good approximation?

Sum of Squares programs require polynomial representations:

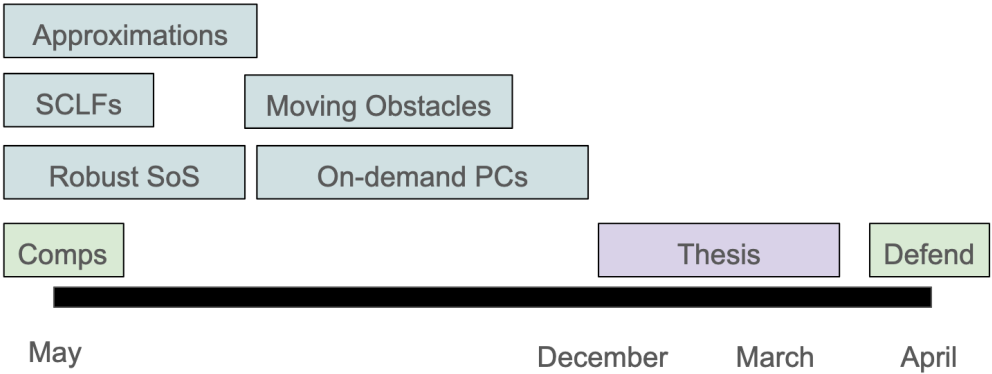
- Piece-wise affine
- Splines
- Taylor polynomials
- Rational functions

# Proposal

Successive Control Certificates:

- Easier synthesis
- Less conservative controlled invariant sets
- Reduce the time for the safety filter override
- More tractable certification
- Interesting nonlinear dynamics

# Timeline


Approximations
SCLFs
Moving Obstacles
Robust SoS
On-demand PCs
Comps
Thesis
Defend
May
December
March
April

*Thank You*