AA/EE/ME 548: Linear Multivariable Control

Lecture 10

4/30/2025

Announcements

- Homework 2 out
  - Start early! Or at least look at it early. Some parts may be helpful for your project!
  - Small update to dynamaxsys – please reinstall

- Project proposal due tomorrow!
  - Everyone must submit a proposal, even for group projects
  - Instructions: https://uw-ctrl.github.io/lmc-book/project/project.html

- Please sign release form

- Week 4 folks, worked examples due today!

- HW1 folks, solutions due Friday

Continuous time setting

$$V^*(x,t) = \min_{u \in \mathcal{U}(x)} \left(\epsilon J(x,u,t) + V^*(x(t+\epsilon), t+\epsilon) \right)$$

$$0 = \min_{u \in \mathcal{U}(x)} \left(\epsilon J(x,u,t) + V^*(x(t+\epsilon), t+\epsilon) - V^*(x,t) \right)$$

$$0 = \lim_{\epsilon \to 0} \min_{u \in \mathcal{U}(x)} \left(J(x,u,t) + \frac{V^*(x(t+\epsilon), t+\epsilon) - V^*(x,t)}{\epsilon} \right)$$

$$0 = \min_{u \in \mathcal{U}(x)} \left(J(x,u,t) + \frac{\partial V}{\partial t}(x,t) + \nabla V(x,t)^T f(x,u,t) \right)$$

The key equations so far…

- Bellman equation

$$V^*(x,t) = \min_{u \in \mathcal{U}(x)} \left(J(x,u,t) + V^*(f(x,u,t), t+1) \right)$$

- Infinite horizon case

$$V^{(k+1)}(x) \leftarrow \min_{u \in \mathcal{U}(x)} \left(J(x,u,t) + \gamma V^{(k)}(f(x,u,t)) \right)$$

- Stochastic case

$$V^*(x_t,t) = \min_{u_t \in \mathcal{U}(x_t)} \mathbb{E}_{x_{t+1} \sim p(x_{t+1}|x_t,u_t,t)} \left[J(x_t,u_t,t) + V^*(x_{t+1}, t+1) \right]$$

- Hamilton-Jacobi-Bellman equation

$$\frac{\partial V}{\partial t}(x,t) + \min_{u \in \mathcal{U}(x)} \left(J(x,u,t) + \nabla V(x,t)^T f(x,u,t) \right) = 0$$

- Hamilton-Jacobi-Isaacs equation

$$\frac{\partial V}{\partial t}(x,t) + \min_{u \in \mathcal{U}(x)} \max_{d \in \mathcal{D}(x)} \left(J(x,u,d,t) + \nabla V(x,t)^T f(x,u,d,t) \right) = 0$$

Disturbance acts 2nd!

Reachability analysis

- The study of the set of states a dynamical system can (and can't) reach over time given:

- **Forward reachability:** The set of states a system can reach within some time using any control input

- **Backward reachability**: Given a *target* set, what are all the states that could reach (or avoid) that set in the future?
  - Solve using dynamic programming
  - BRS: reach/avoid target set at time t
  - BRT: reach/avoid target set any time up until t

With disturbances

Reach case

$$\mathcal{R}(\mathcal{T}, t) = \{x_0 \in \mathcal{X} \mid \forall d(\cdot) \in \mathbb{D}[0, t], \exists u(\cdot) \in \mathbb{U}[0, t] \text{ s.t. } \xi_{x_0, 0}^{u(\cdot)}(t) \in \mathcal{T}\}$$

$$\widetilde{\mathcal{R}}(\mathcal{T}, t) = \{x_0 \in \mathcal{X} \mid \forall d(\cdot) \in \mathbb{D}[0, t], \exists s \in [0, t], \exists u(\cdot) \in \mathbb{U}[0, s] \text{ s.t. } \xi_{x_0, 0}^{u(\cdot)}(s) \in \mathcal{T}\}$$

Avoid case

$$\mathcal{A}(\mathcal{T}, t) = \{x_0 \in \mathcal{X} \mid \exists d(\cdot) \in \mathbb{D}[0, t], \forall u(\cdot) \in \mathbb{U}[0, t] \text{ s.t. } \xi_{x_0, 0}^{u(\cdot)}(t) \in \mathcal{T}\}$$

$$\widetilde{\mathcal{A}}(\mathcal{T}, t) = \{x_0 \in \mathcal{X} \mid \exists d(\cdot) \in \mathbb{D}[0, t], \exists s \in [0, t], \forall u(\cdot) \in \mathbb{U}[0, s] \text{ s.t. } \xi_{x_0, 0}^{u(\cdot)}(s) \in \mathcal{T}\}$$

How to compute the sets?

- Frame as optimal control problem!
- Key idea: *Only care whether the system reaches the target set.*

$$\min_{u(\cdot)\in\mathbb{U}[0,T]} \quad \cancel{\int_0^T J(x,u,t)dt} + J_T(x(T))$$

$$\text{subject to} \quad \dot{x} = f(x,u,t)$$

$$x_0 = x(0)$$

$$u \in \mathcal{U}(x), x \in \mathcal{X}$$

Define target set & value function interpretation

$$\mathcal{T} = \{x \in \mathcal{X} \mid J_T(x) \leq 0\}$$

$$V^*(x,t) = \min_{u(\cdot) \in \mathbb{U}[t,T]} \int_t^T J(x,u,\tau)d\tau + J_T(x(T))$$

$$V^*(x,t) = \min_{u(\cdot) \in \mathbb{U}[t,T]} J_T(x(T))$$

Reach case

"regular"

$$\frac{\partial V}{\partial t}(x,t) + \min_{u \in \mathcal{U}(x)} \nabla V(x,t)^T f(x,u,t) = 0$$

"w/ disturbance"

$$\frac{\partial V}{\partial t}(x,t) + \min_{u \in \mathcal{U}(x)} \max_{d \in \mathcal{D}(x)} \nabla V(x,t)^T f(x,u,d,t) = 0$$

"tube"

$$\frac{\partial V}{\partial t}(x,t) + \min\left(0, \min_{u \in \mathcal{U}(x)} \max_{d \in \mathcal{D}(x)} \nabla V(x,t)^T f(x,u,d,t) \right) = 0$$

$$V(x,T) = J_T(x)$$

Avoid case

"regular"
$$\frac{\partial V}{\partial t}(x,t) + \max_{u \in \mathcal{U}(x)} \nabla V(x,t)^T f(x,u,t) = 0$$

"w/ disturbance"
$$\frac{\partial V}{\partial t}(x,t) + \max_{u \in \mathcal{U}(x)} \min_{d \in \mathcal{D}(x)} \nabla V(x,t)^T f(x,u,d,t) = 0$$

"tube"
$$\frac{\partial V}{\partial t}(x,t) + \min\left(0, \max_{u \in \mathcal{U}(x)} \min_{d \in \mathcal{D}(x)} \nabla V(x,t)^T f(x,u,d,t)\right) = 0$$

$$V(x,T) = J_T(x)$$

Computing HJ reachable sets

- https://github.com/StanfordASL/hj_reachability
- https://github.com/StanfordASL/hj_reachability/blob/main/examples/quickstart.ipynb
- Solving the HJI PDE explicitly. I.e., discretizing state space
  - Suffers from curse of dimensionality!
  - Doable with ~5 states (depends on your computer), more/faster with GPU ☺

```python
dynamics = hj.systems.Air3d()

grid = hj.Grid.from_lattice_parameters_and_boundary_conditions(hj.sets.Box(np.array([-6., -10., 0.]),
np.array([20., 10., 2 * np.pi])), (51, 40, 50), periodic_dims=2)

values = jnp.linalg.norm(grid.states[..., :2], axis=-1) - 5

solver_settings = hj.SolverSettings.with_accuracy("very_high",
hamiltonian_postprocessor=hj.solver.backwards_reachable_tube)
```
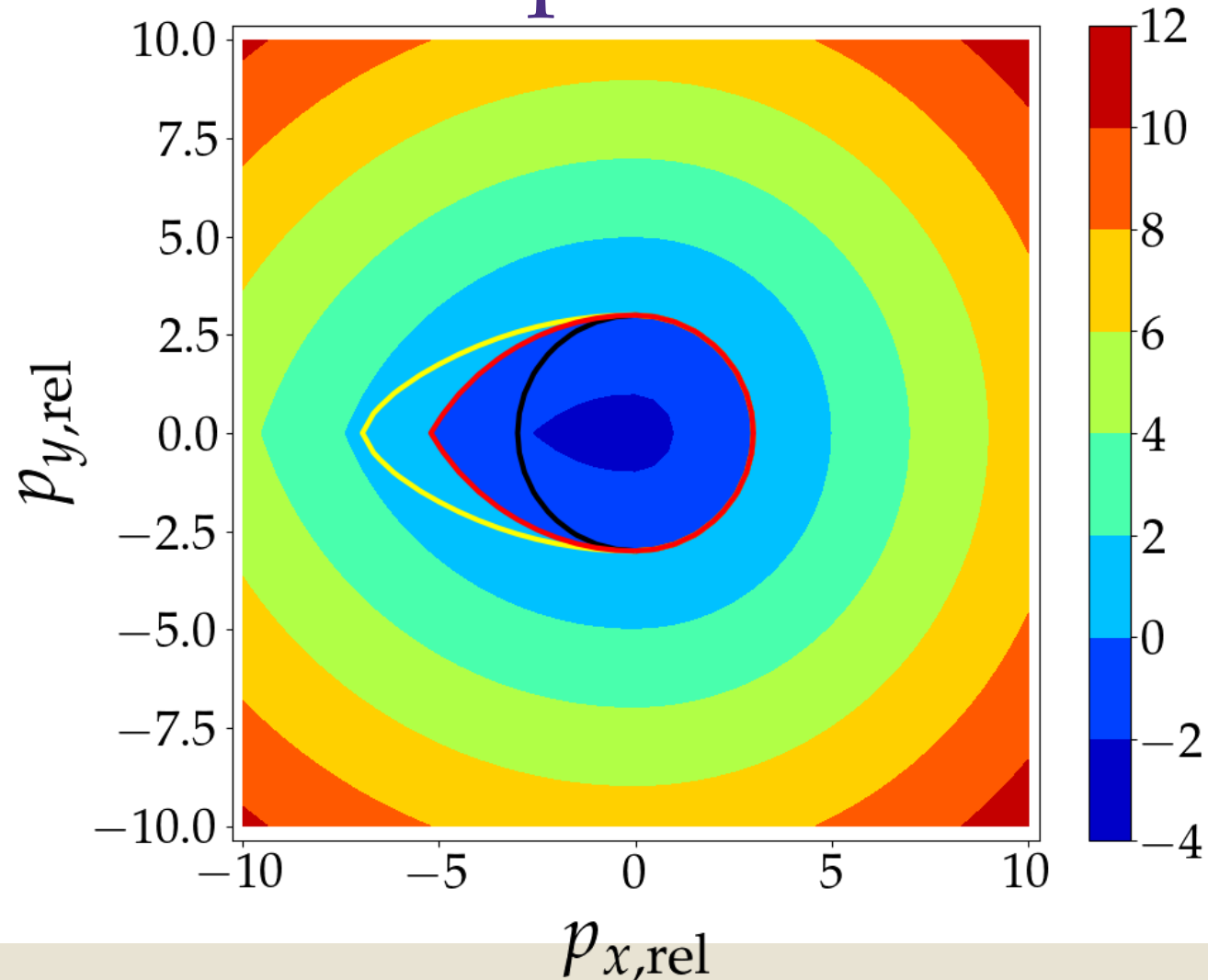
Dubins' car example (avoid case)

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ u+d \end{bmatrix}$$
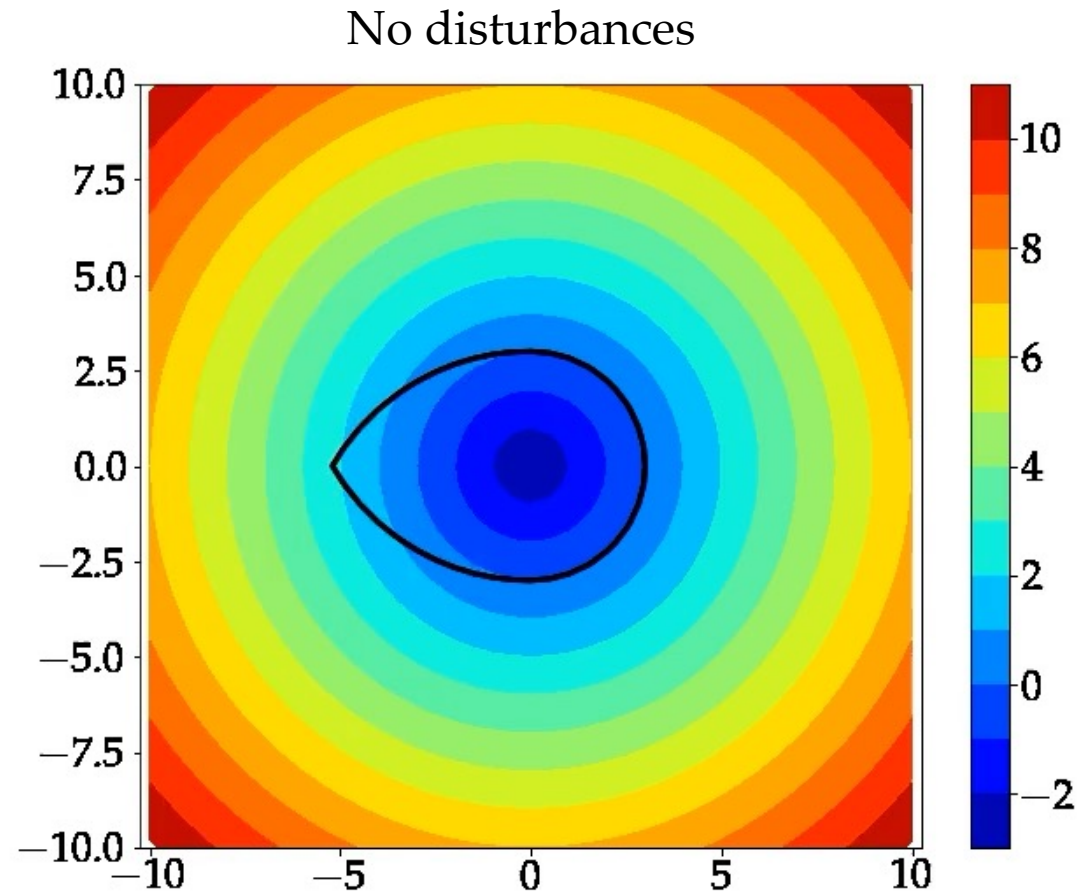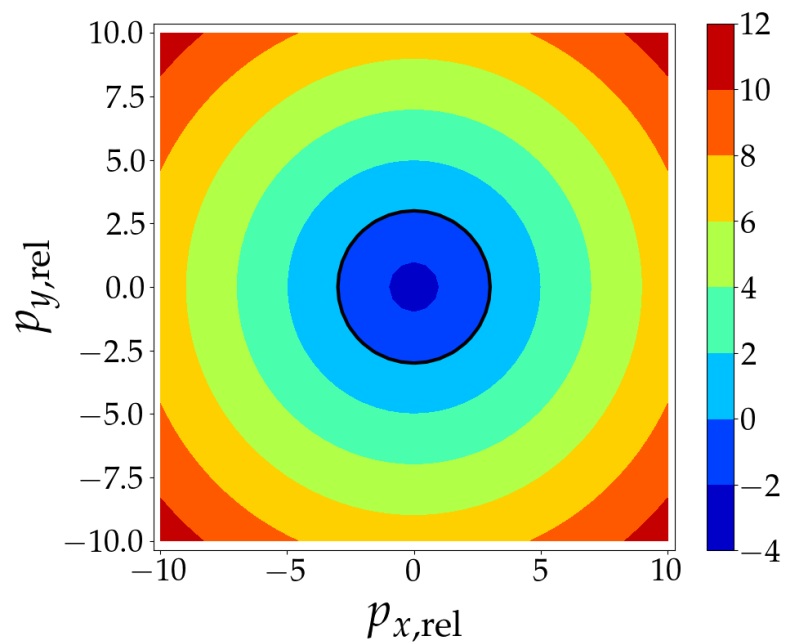
No disturbances

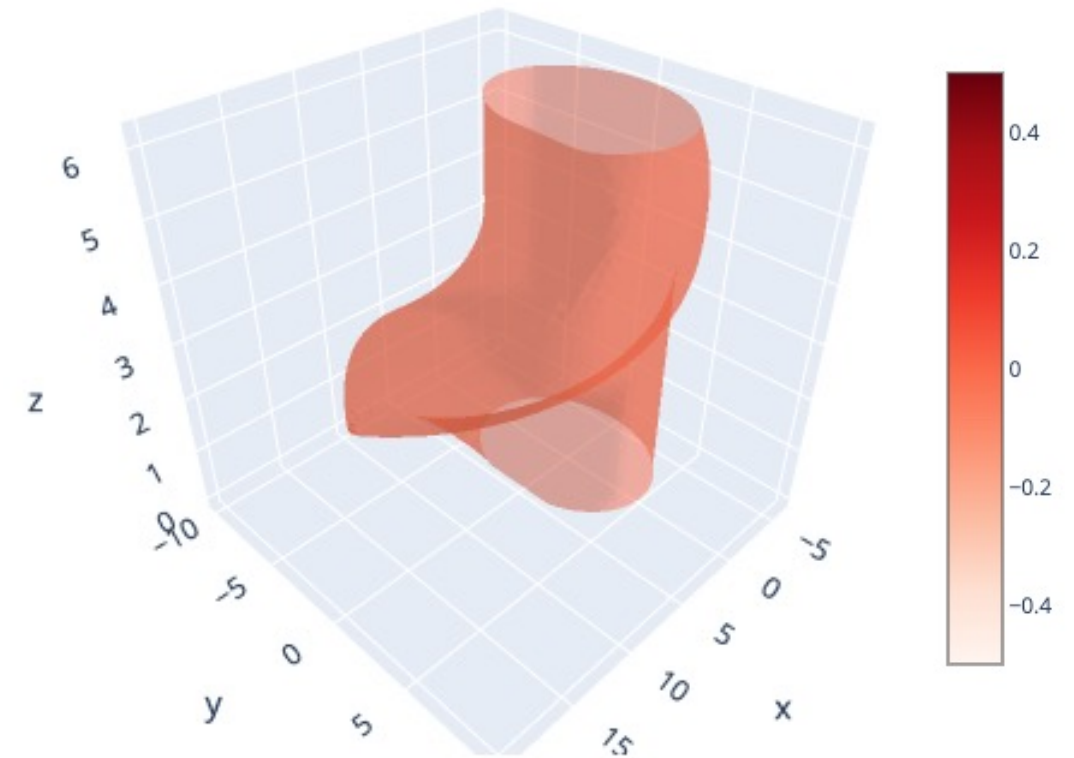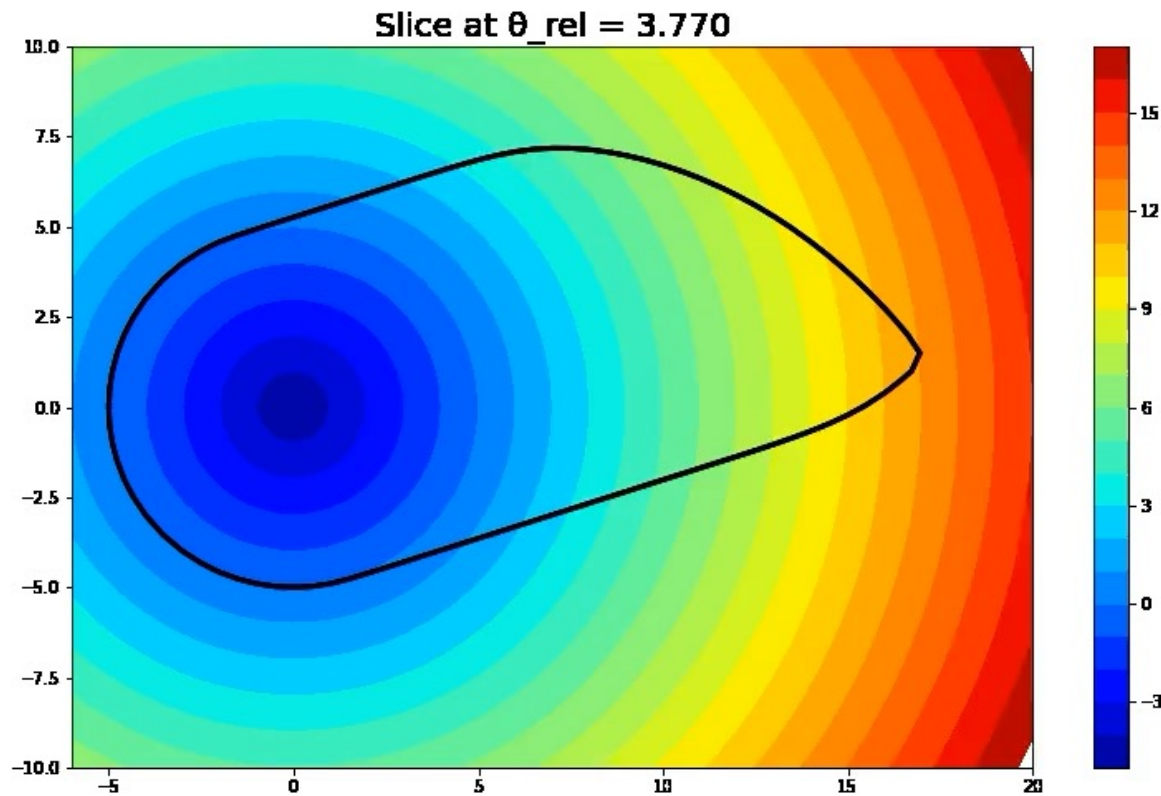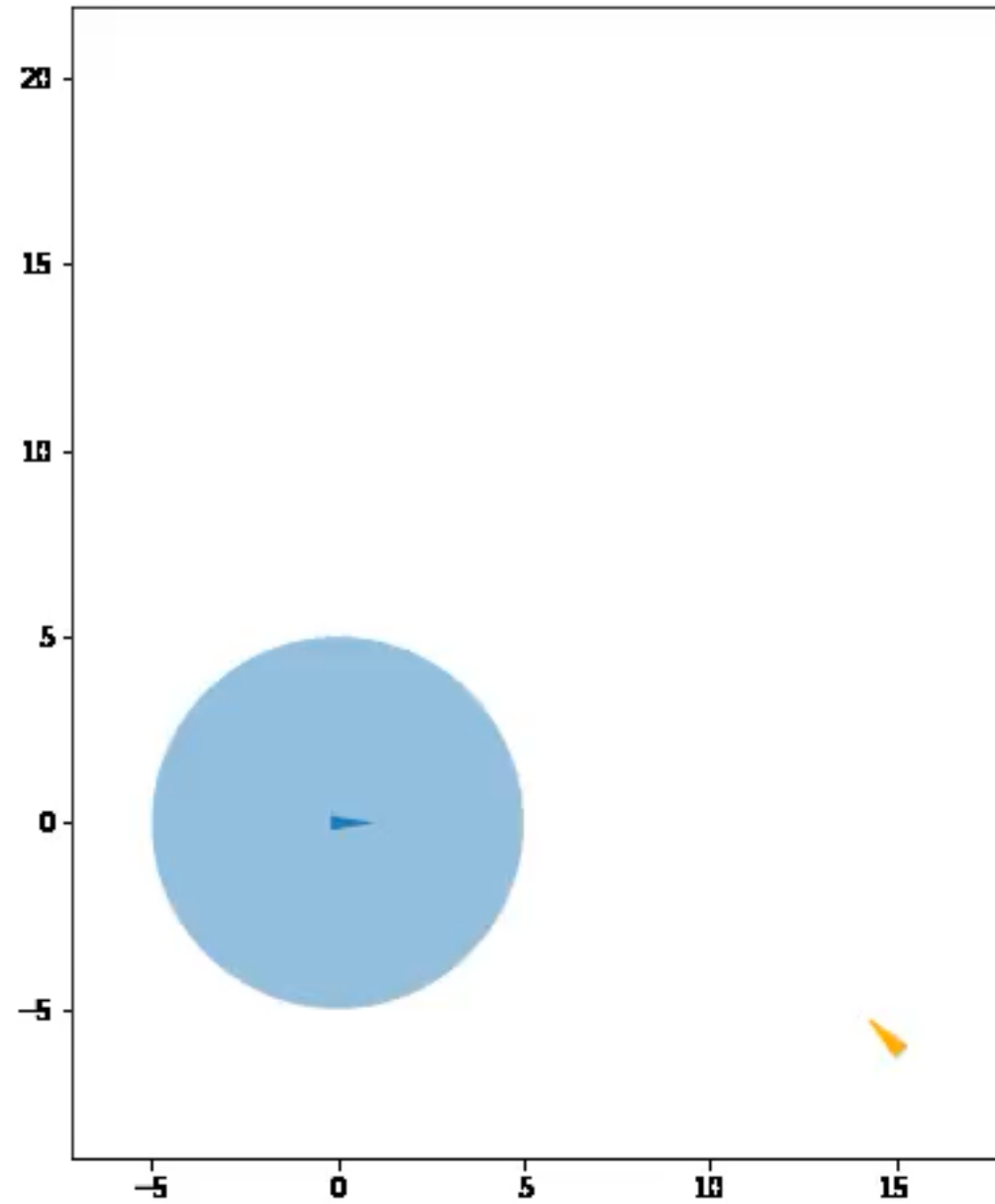# Dubins' car example with disturbances
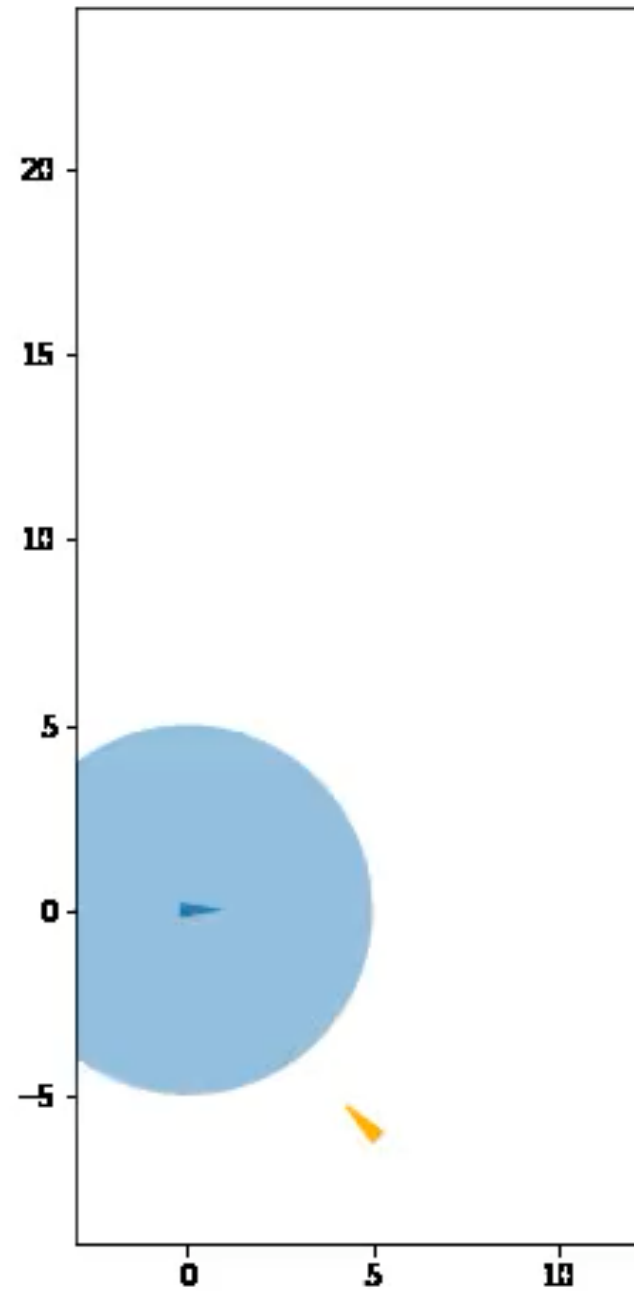
Pursuit-evasion example

- Can consider a *relative* system between two agents & treat other agent as a disturbance

- Consider two Dubins' car agents, one is pursuing the other (i.e., cat and mouse)
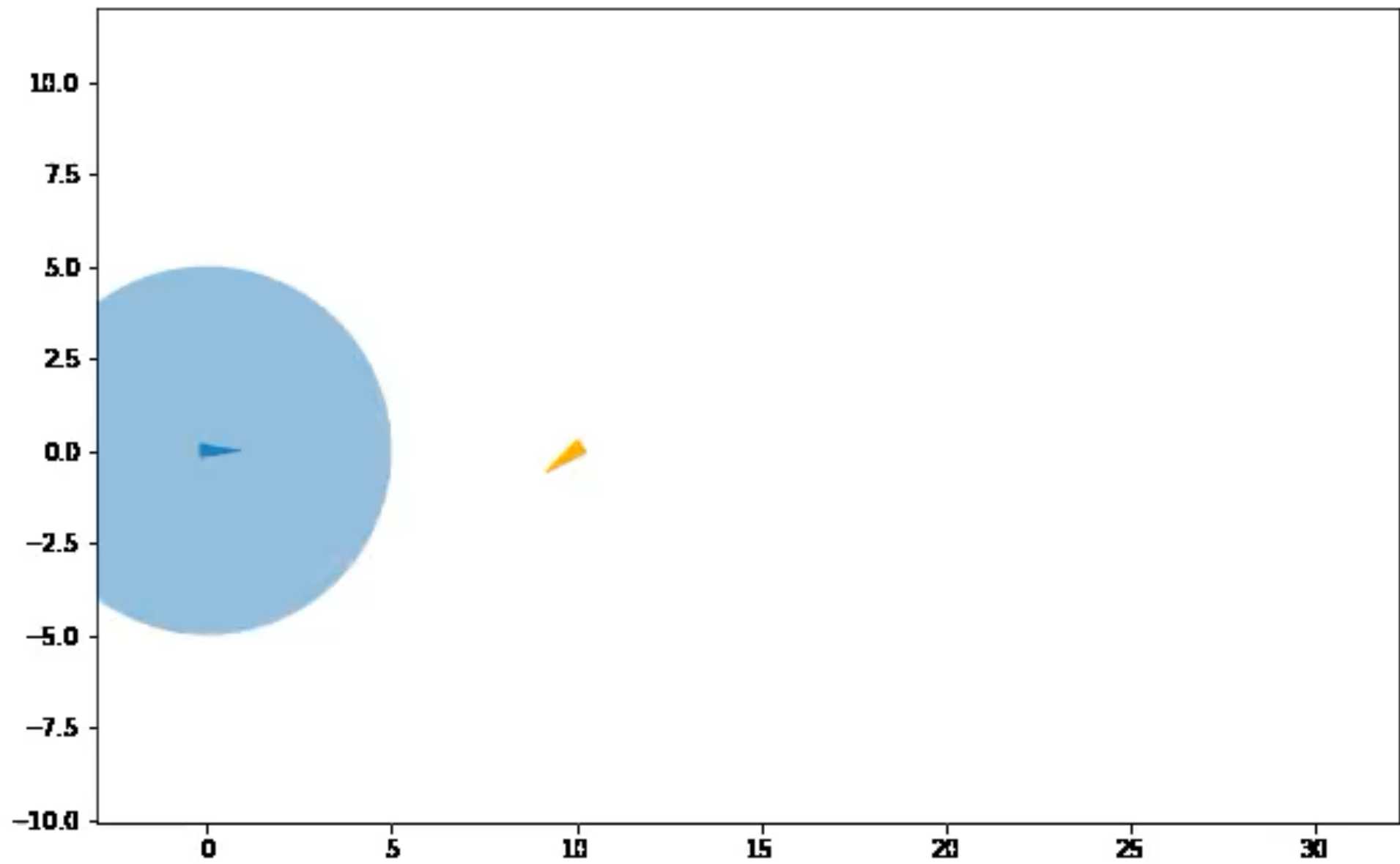
$$
\begin{bmatrix} \dot{x}_{\mathrm{rel}} \\ \dot{y}_{\mathrm{rel}} \\ \dot{\theta}_{\mathrm{rel}} \end{bmatrix} = \begin{bmatrix} -v_{\mathrm{a}} + v_{\mathrm{b}} \cos \theta_{\mathrm{rel}} + y_{\mathrm{rel}} u_{\mathrm{a}} \\ v_{\mathrm{b}} \sin \theta_{\mathrm{rel}} - x_{\mathrm{rel}} u_{\mathrm{a}} \\ u_{\mathrm{b}} - u_{\mathrm{a}} \end{bmatrix}
$$

Pursuit-evasion example

Slice at θ_rel = 3.770

What can I do with the HJ value function?

- HJ value functions are CBFs!
  - https://arxiv.org/abs/2104.02808
  - https://youtu.be/9SIO-oNT7Gs By Dr. Sylvia Herbert, UCSD

- Least-restrictive controller
  - When $V(x, t) \leq \epsilon$, switch to optimal HJ policy

- Minimally-interventional safe controller
  - When $V(x, t) \leq \epsilon$, add HJ control constraint in tracking controller
  - https://arxiv.org/abs/2012.03390

Minimally-interventional control

Minimize

Tracking error

Subject to

Control limits

Dynamics

State constraints

$+$

If $V(x, t) < \epsilon$
then activate HJ
safety-preserving
control set constraint

Tracking only

Switching

Min. interv.

Robot Human