AA 598B Special Topics

Decision-Making & Control for Safe Interactive Autonomy

Instructor: Prof. Karen Leung

Autumn 2024

https://faculty.washington.edu/kymleung/aa598/

Announcements

- Proposal feedback – can revise and resubmit by Friday.
  - Leave a comment on Canvas
- Homework 2 due (recommended)
- Start your project!

Last time

- Quick intro to game theory

$$\forall i \in [N] \begin{cases} \min_{X^i, U^i} & J^i(\mathbf{X}, U^i; \theta^i) \\ \text{s.t.} & x_{t+1}^i = f^i(x_t^i, u_t^i), \forall t \in [T-1] \\ & x_1^i = \hat{x}_1^i \\ & {}^p g^i(X^i, U^i) \geq 0 \\ & {}^s g(\mathbf{X}, \mathbf{U}) \geq 0. \end{cases}$$

$$\nabla_x \mathcal{L}(x^\star, \mu^\star, \lambda^\star) = 0$$
$$\nabla_\mu \mathcal{L}(x^\star, \mu^\star, \lambda^\star) = 0$$
$$\nabla_\lambda \mathcal{L}(x^\star, \mu^\star, \lambda^\star) = 0$$
$$\mu^\star \geq 0$$

KKT conditions

- Nash equilibrium: $J_i(u_i^*, u_{-i}^*) \leq J_i(u_i, u_{-1}^*) \quad \forall\, u_i \in U_i$

$$\mathbf{G}(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}) := \begin{bmatrix} \left.\begin{matrix} \nabla_{\mathbf{x}} J^i + \boldsymbol{\lambda}^{i\top} \nabla_{\mathbf{x}} \mathbf{F}(\mathbf{x}, \mathbf{u}) \\ \nabla_{\mathbf{u}^i} J^i + \boldsymbol{\lambda}^{i\top} \nabla_{\mathbf{u}^i} \mathbf{F}(\mathbf{x}, \mathbf{u}) \end{matrix}\right\} \forall i \in [N] \\ \mathbf{F}(\mathbf{x}, \mathbf{u}) \end{bmatrix} = \mathbf{0}. \tag{5}$$

$$\max_{\theta, \mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}} \quad p(\mathbf{y} \mid \mathbf{x}, \mathbf{u}) \qquad \text{s.t.} \quad \mathbf{G}(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}; \theta) = \mathbf{0}.$$

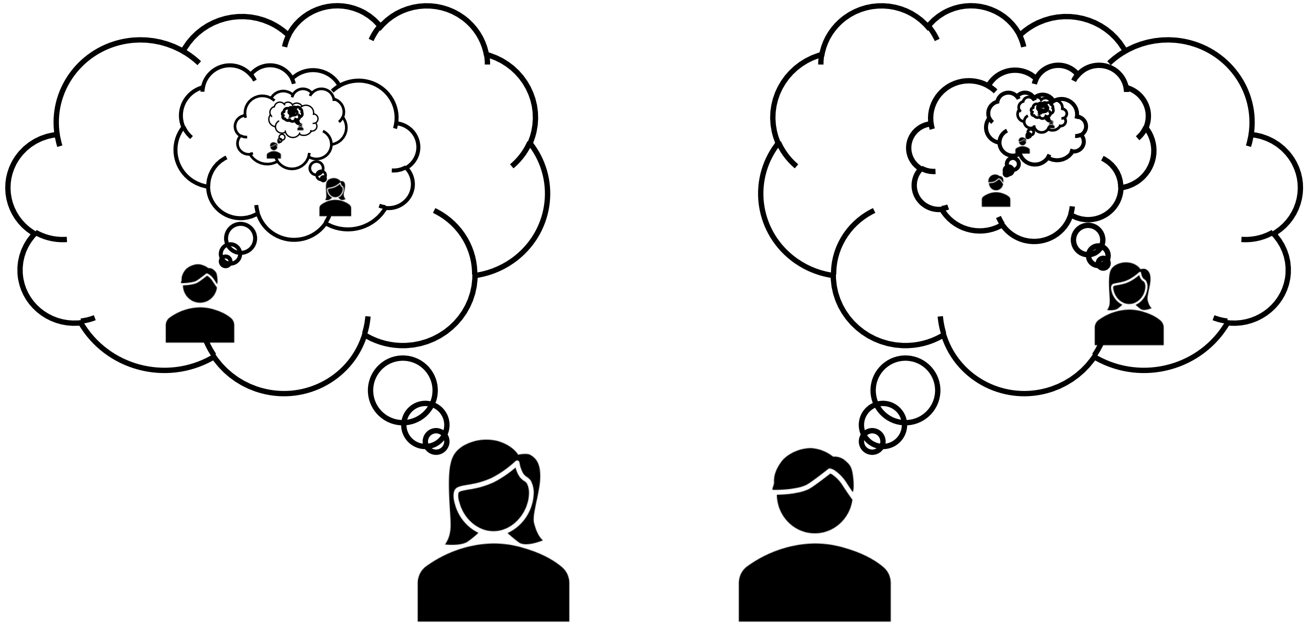
Today

- Wrap up game theory
- Sampling-based methods for planning
- Start Module #3: Controls

Finding Nash equilibria

Generally difficult to find

- Iterated best response



Wang et al 2021 https://msl.stanford.edu/papers/wang_game-theoretic_2021.pdf

Stackelberg games

Leader-follower structure

- Two players & one acts first and the second, after observing the the first player's action, acts afterwards.

For any function $f : Z \times W \to \mathbb{R}$,

$$\underset{\substack{\textbf{First}\\ \text{player}}}{\sup_{z \in Z}} \; \underset{\substack{\textbf{Second}\\ \text{player}}}{\inf_{w \in W}} f(z, w) \leq \underset{\substack{\textbf{First}\\ \text{player}}}{\inf_{w \in W}} \; \underset{\substack{\textbf{Second}\\ \text{player}}}{\sup_{z \in Z}} f(z, w)$$

What if we can’t compute gradients easily?

- So far, the methods relied on some sort of gradient descent
  - SQP assumes differentiability
  - KKT conditions
- Don’t have a differentiable model for human prediction, cost, constraints, etc.
- We can consider searching over the space via a sampling-based approach
  - Leverage computation!

Robot conditioned human trajectory predictor

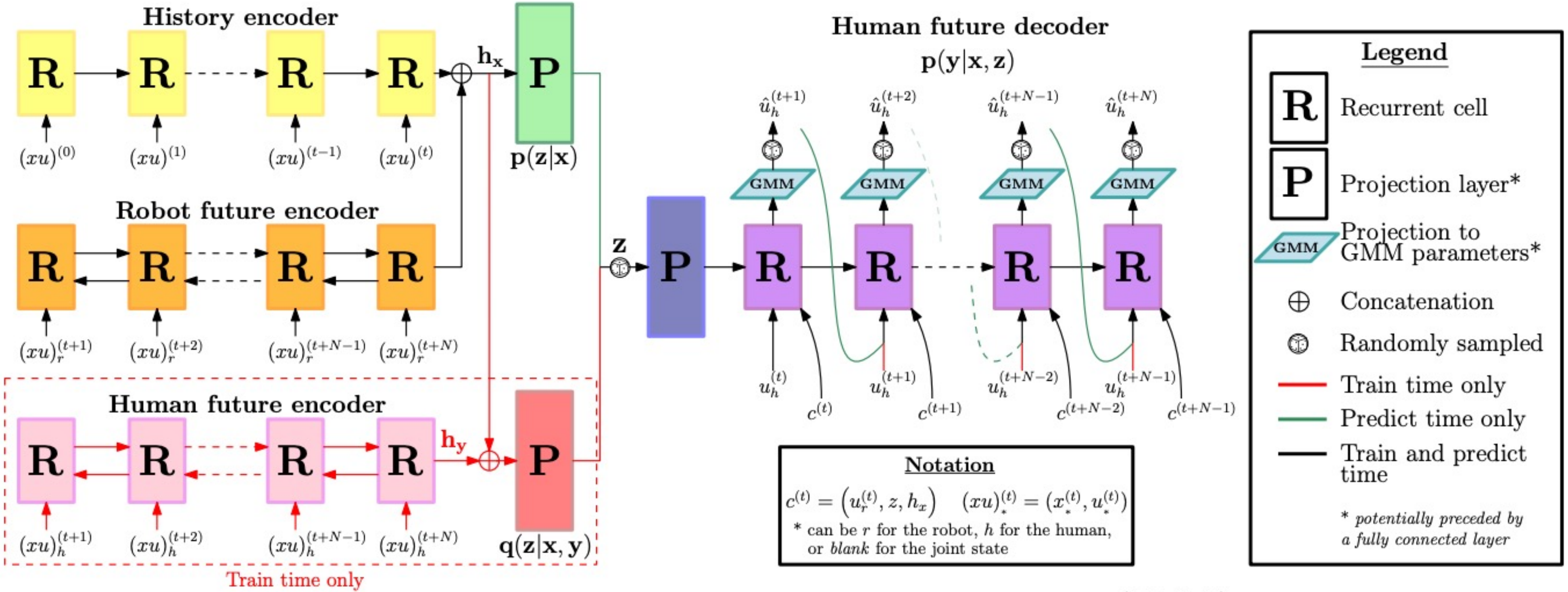
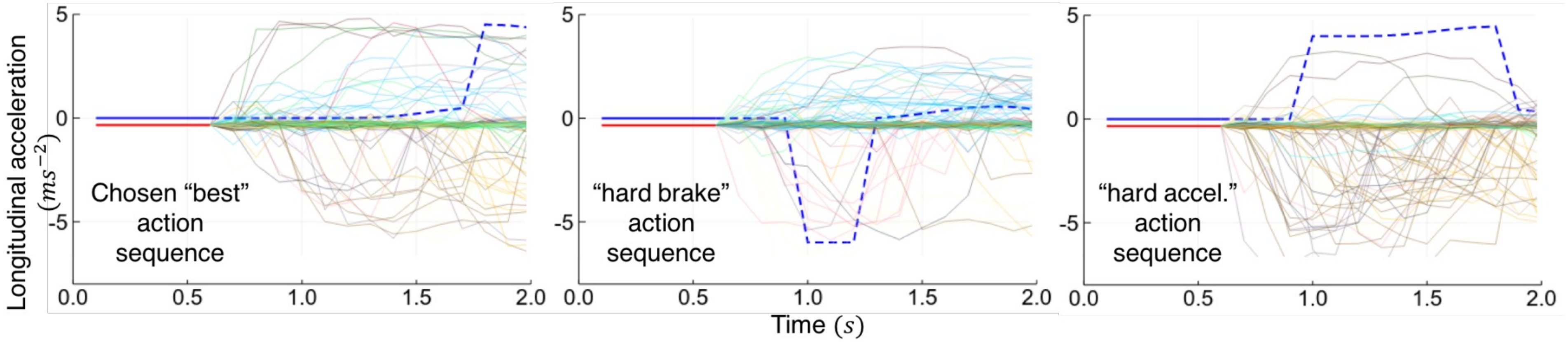


Fig. 2. CVAE architecture for sequence-to-sequence generative modeling of future human actions $\mathbf{y} = u_h^{(t+1:t+N)}$ conditioned on joint interaction history $(x^{(0:t)}, u^{(0:t)})$ and candidate robot future actions $u_r^{(t+1:t+N)}$ (together, $\mathbf{x}$). The random variable $\mathbf{z}$ is a latent mixture component index.

Schmerling et al 2018

Planning with ego-conditioned prediction

Multimodal Probabilistic Model-Based Planning for Human-Robot Interaction



Schmerling et al 2018

Search over a pre-computed trajectory tree

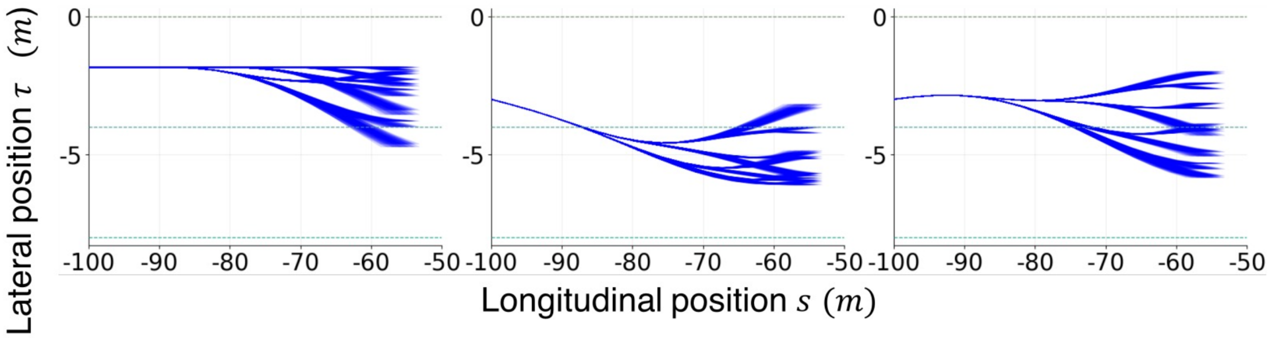


Fig. 5. 4096 candidate robot action sequences scored each planning loop.

Sampling to estimate gradient

Model Predictive Path Integral (MPPI)

(Homework 2)

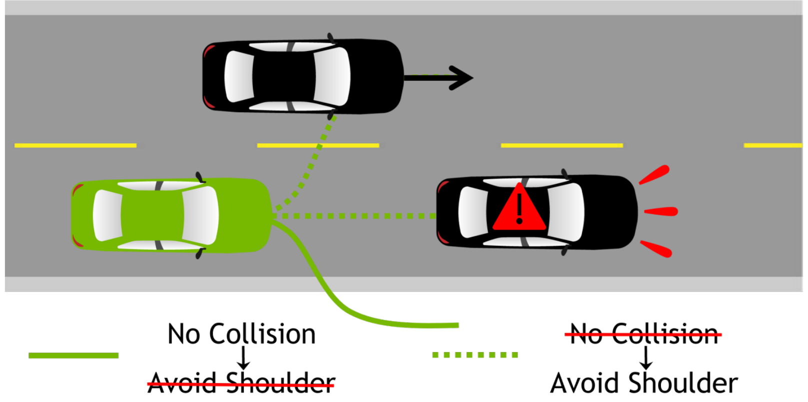
1. Start with nominal trajectory
2. Add noise to it to generate many trajectories
3. Evaluate cost of each trajectory
4. Compute weight for each trajectory
5. Compute weighted sum over controls to compute control

https://sites.gatech.edu/acds/mppi/

Potential drawbacks? Advantages?

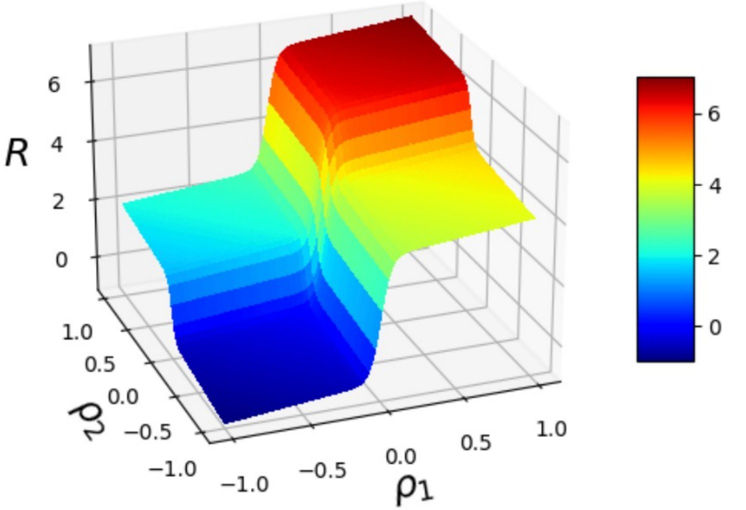
Planning with rules

Receding Horizon Planning with Rule Hierarchies for Autonomous Vehicles



Rank	Satisfied Rules
1	$\phi_1,\ \phi_2,\ \phi_3$
2	$\phi_1,\ \phi_2$
3	$\phi_1,\ \phi_3$
4	ϕ_1
5	$\phi_2,\ \phi_3$
6	ϕ_2
7	ϕ_3
8	$\emptyset$

TABLE I: Illustration of trajectory ranks for three rules.

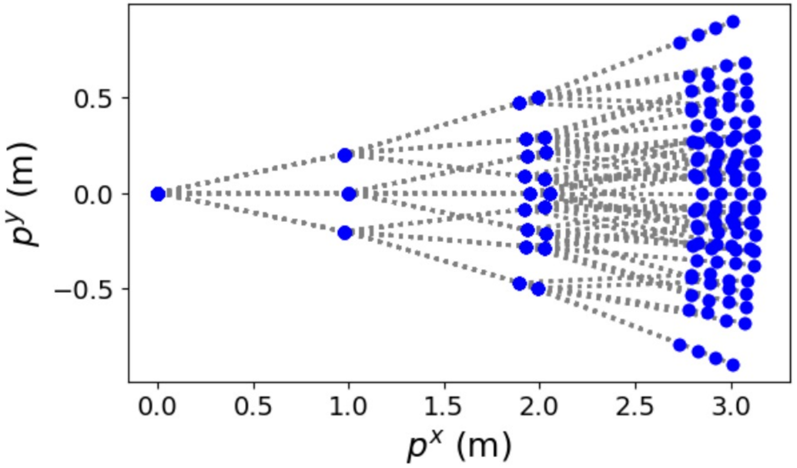


$$R(\rho) := \sum_{i=1}^{N} \left(a^{N-i+1} \mathrm{sigmoid}(c\rho_i) + \frac{1}{N}\rho_i \right)$$

Veer et al 2023

Planning with rules

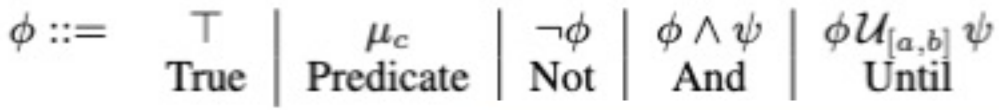
Two-step optimization: trajectory tree + local refinement



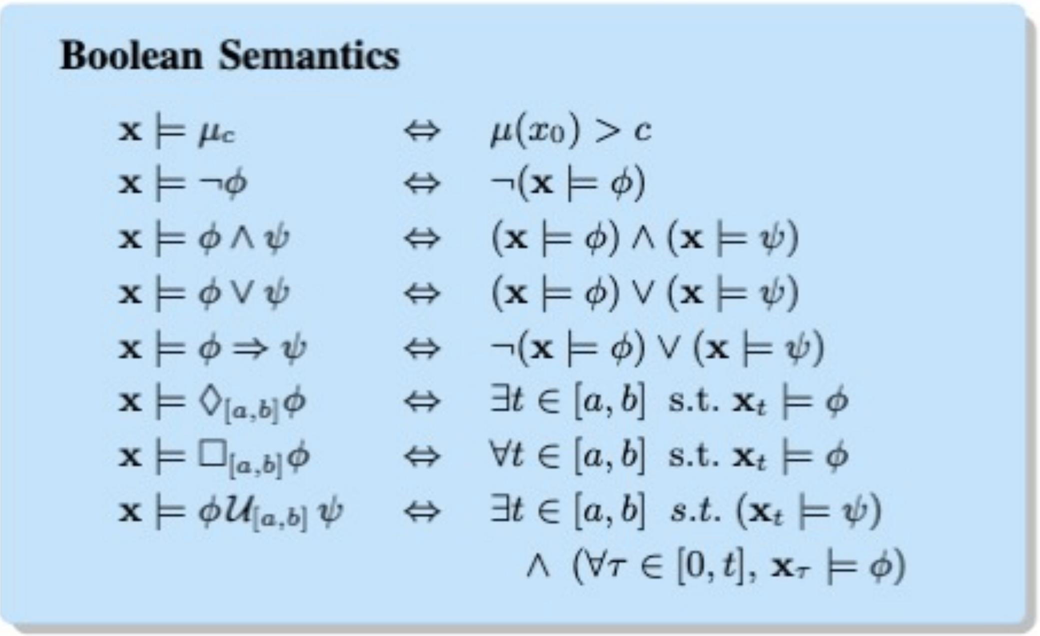
$$
\begin{aligned}
\min_{u_{0:T}} \quad & -R \circ \hat{\rho}(x_{0:T}, w) \\
\text{s.t.} \quad & x_{t+1} = f(x_t, u_t), \ \text{for } t = 1, \cdots, T
\end{aligned}
$$

https://github.com/UW-CTRL/stljax/

Signal temporal logic

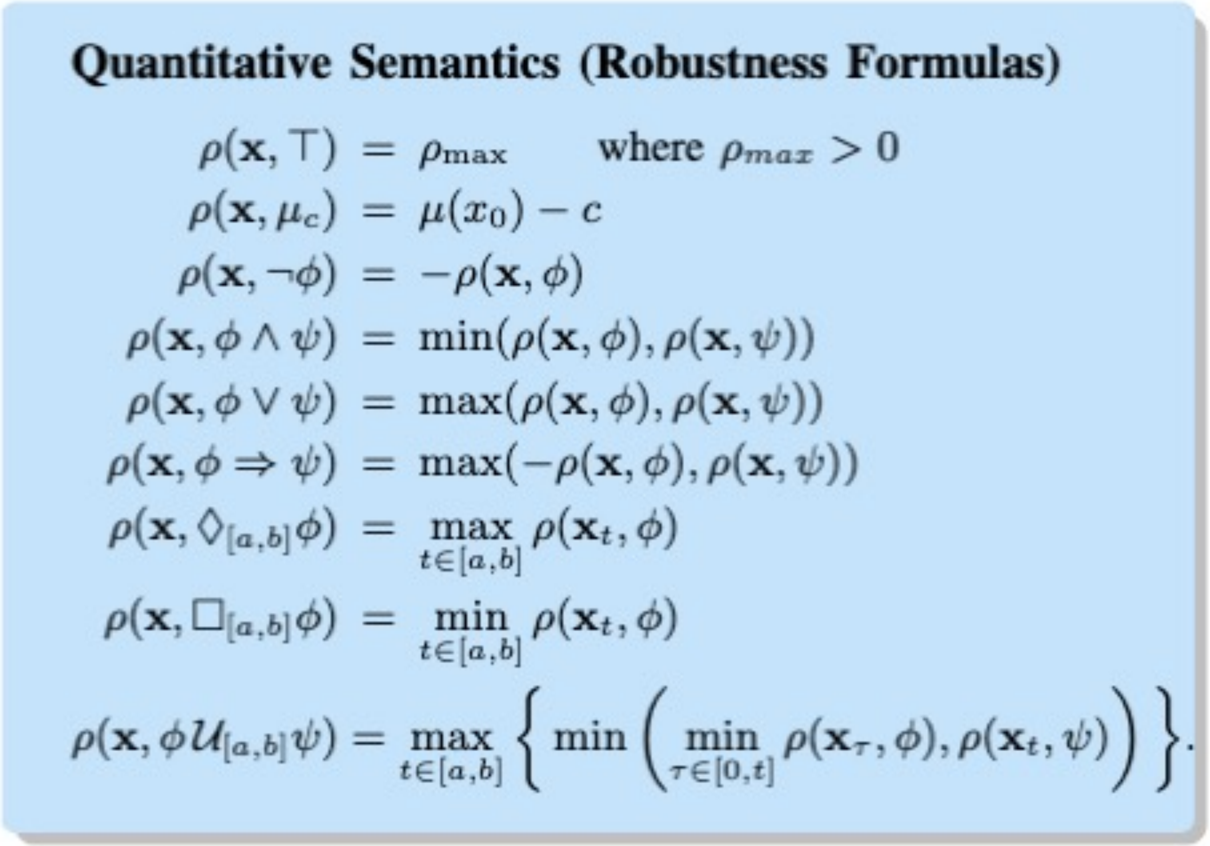


$$\begin{array}{lc|c|c|c|c} \phi ::= & \top & \mu_c & \neg\phi & \phi \wedge \psi & \phi\,\mathcal{U}_{[a,b]}\,\psi \\ & \text{True} & \text{Predicate} & \text{Not} & \text{And} & \text{Until} \end{array}$$



Boolean Semantics

$$\begin{array}{lcl} \mathbf{x} \models \mu_c & \Leftrightarrow & \mu(x_0) > c \\ \mathbf{x} \models \neg\phi & \Leftrightarrow & \neg(\mathbf{x} \models \phi) \\ \mathbf{x} \models \phi \wedge \psi & \Leftrightarrow & (\mathbf{x} \models \phi) \wedge (\mathbf{x} \models \psi) \\ \mathbf{x} \models \phi \vee \psi & \Leftrightarrow & (\mathbf{x} \models \phi) \vee (\mathbf{x} \models \psi) \\ \mathbf{x} \models \phi \Rightarrow \psi & \Leftrightarrow & \neg(\mathbf{x} \models \phi) \vee (\mathbf{x} \models \psi) \\ \mathbf{x} \models \Diamond_{[a,b]}\phi & \Leftrightarrow & \exists t \in [a,b] \text{ s.t. } \mathbf{x}_t \models \phi \\ \mathbf{x} \models \Box_{[a,b]}\phi & \Leftrightarrow & \forall t \in [a,b] \text{ s.t. } \mathbf{x}_t \models \phi \\ \mathbf{x} \models \phi\,\mathcal{U}_{[a,b]}\,\psi & \Leftrightarrow & \exists t \in [a,b] \;\; s.t. \; (\mathbf{x}_t \models \psi) \\ & & \quad \wedge \; (\forall \tau \in [0,t], \, \mathbf{x}_\tau \models \phi) \end{array}$$



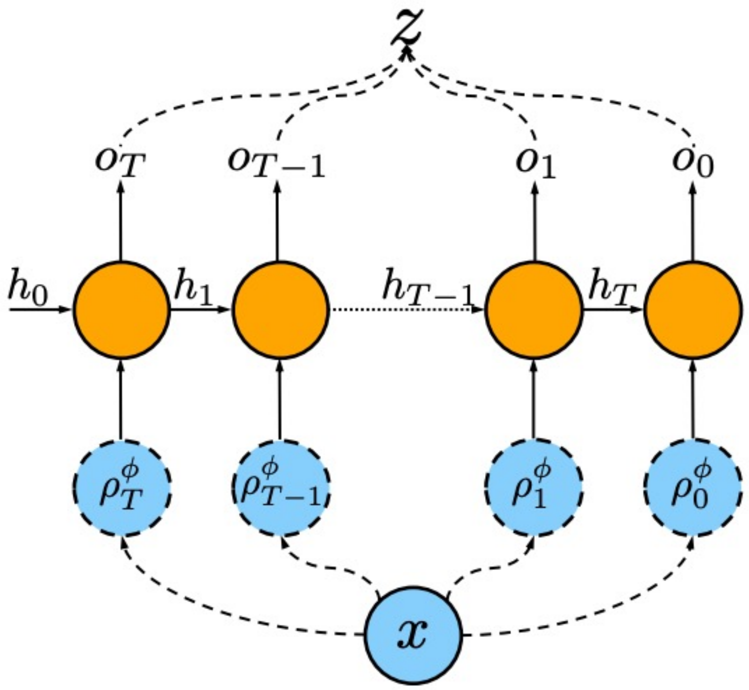
Quantitative Semantics (Robustness Formulas)

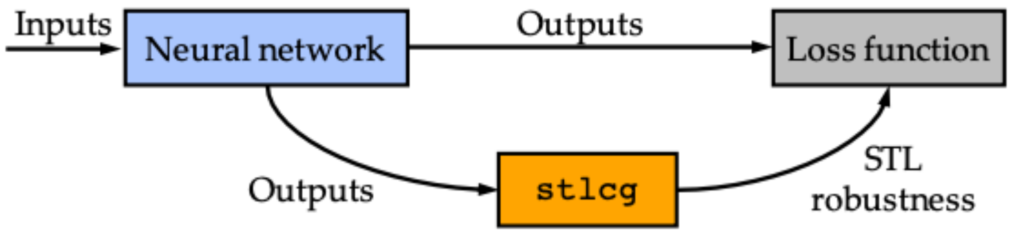
$$\begin{aligned} \rho(\mathbf{x}, \top) &= \rho_{\max} \qquad \text{where } \rho_{max} > 0 \\ \rho(\mathbf{x}, \mu_c) &= \mu(x_0) - c \\ \rho(\mathbf{x}, \neg\phi) &= -\rho(\mathbf{x}, \phi) \\ \rho(\mathbf{x}, \phi \wedge \psi) &= \min(\rho(\mathbf{x}, \phi), \rho(\mathbf{x}, \psi)) \\ \rho(\mathbf{x}, \phi \vee \psi) &= \max(\rho(\mathbf{x}, \phi), \rho(\mathbf{x}, \psi)) \\ \rho(\mathbf{x}, \phi \Rightarrow \psi) &= \max(-\rho(\mathbf{x}, \phi), \rho(\mathbf{x}, \psi)) \\ \rho(\mathbf{x}, \Diamond_{[a,b]}\phi) &= \max_{t \in [a,b]} \rho(\mathbf{x}_t, \phi) \\ \rho(\mathbf{x}, \Box_{[a,b]}\phi) &= \min_{t \in [a,b]} \rho(\mathbf{x}_t, \phi) \end{aligned}$$

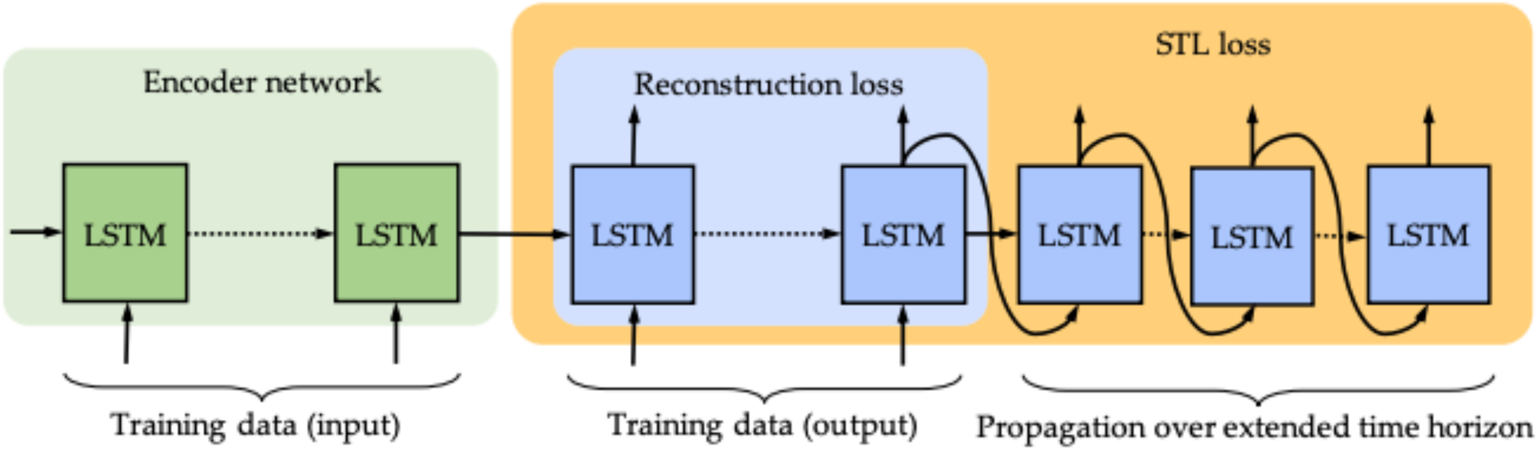
$$\rho(\mathbf{x}, \phi\,\mathcal{U}_{[a,b]}\psi) = \max_{t \in [a,b]} \left\{ \min\left(\min_{\tau \in [0,t]} \rho(\mathbf{x}_\tau, \phi), \rho(\mathbf{x}_t, \psi) \right) \right\}.$$

https://arxiv.org/abs/2008.00097

Differentiable STL



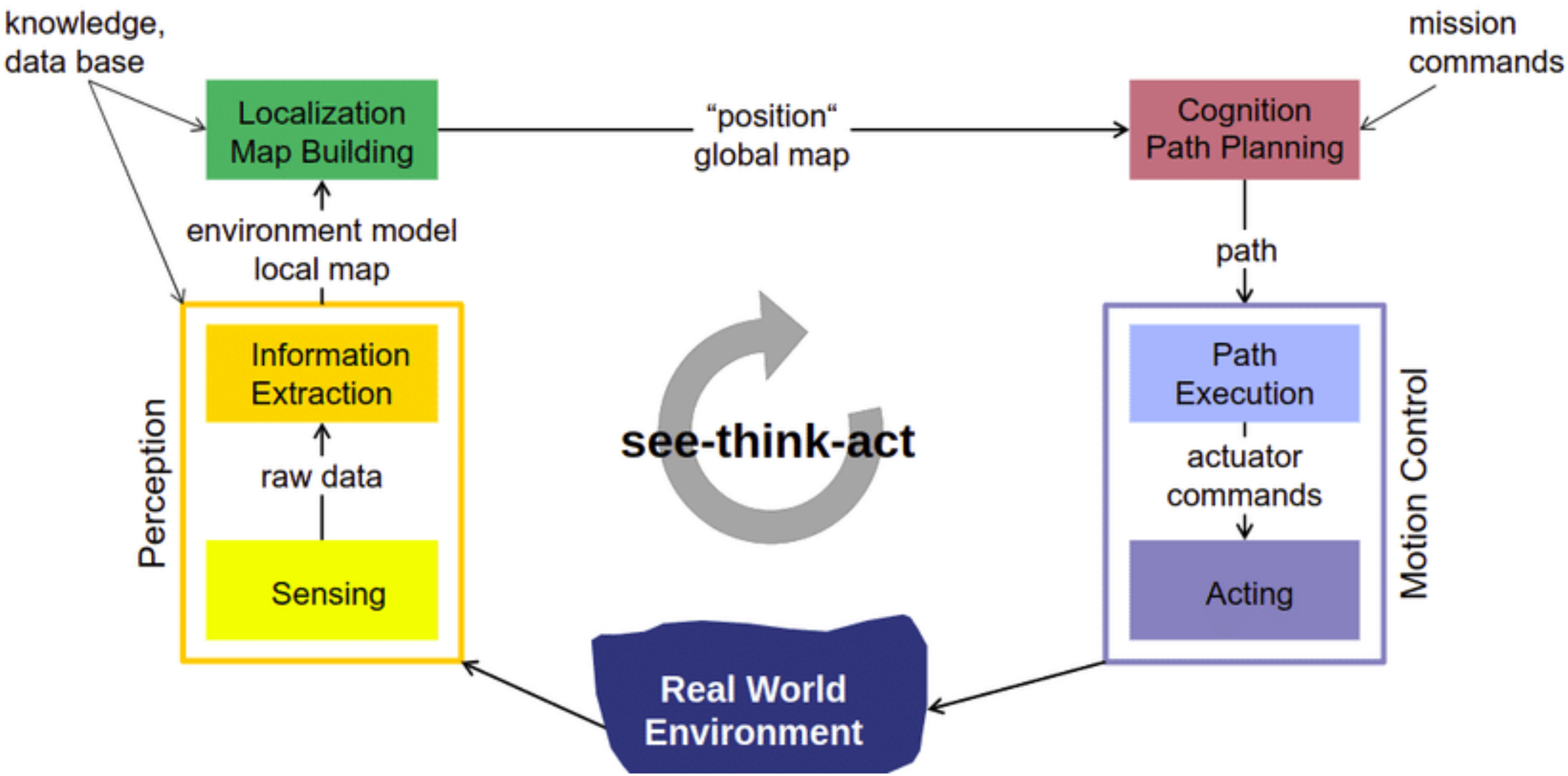




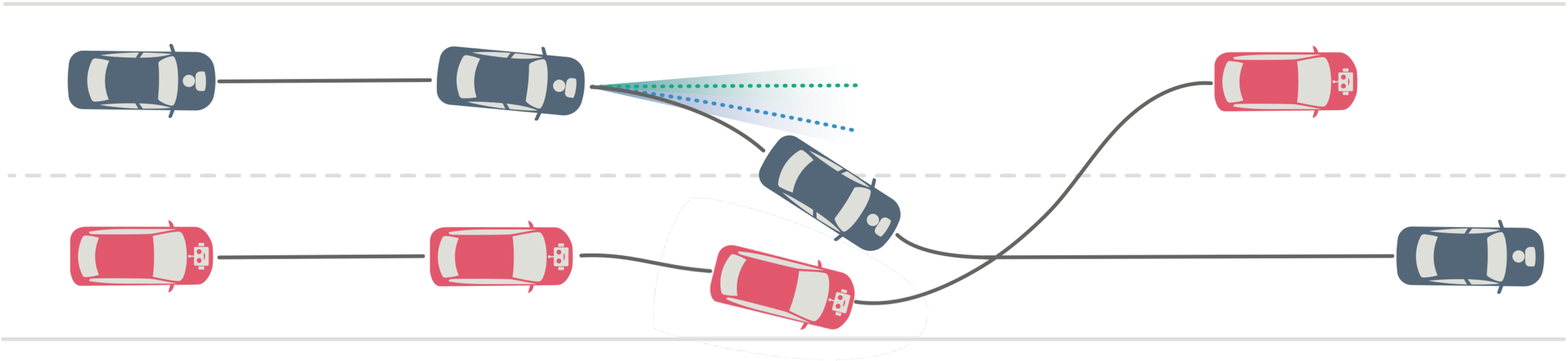
https://github.com/UW-CTRL/stljax/tree/main

Safe control

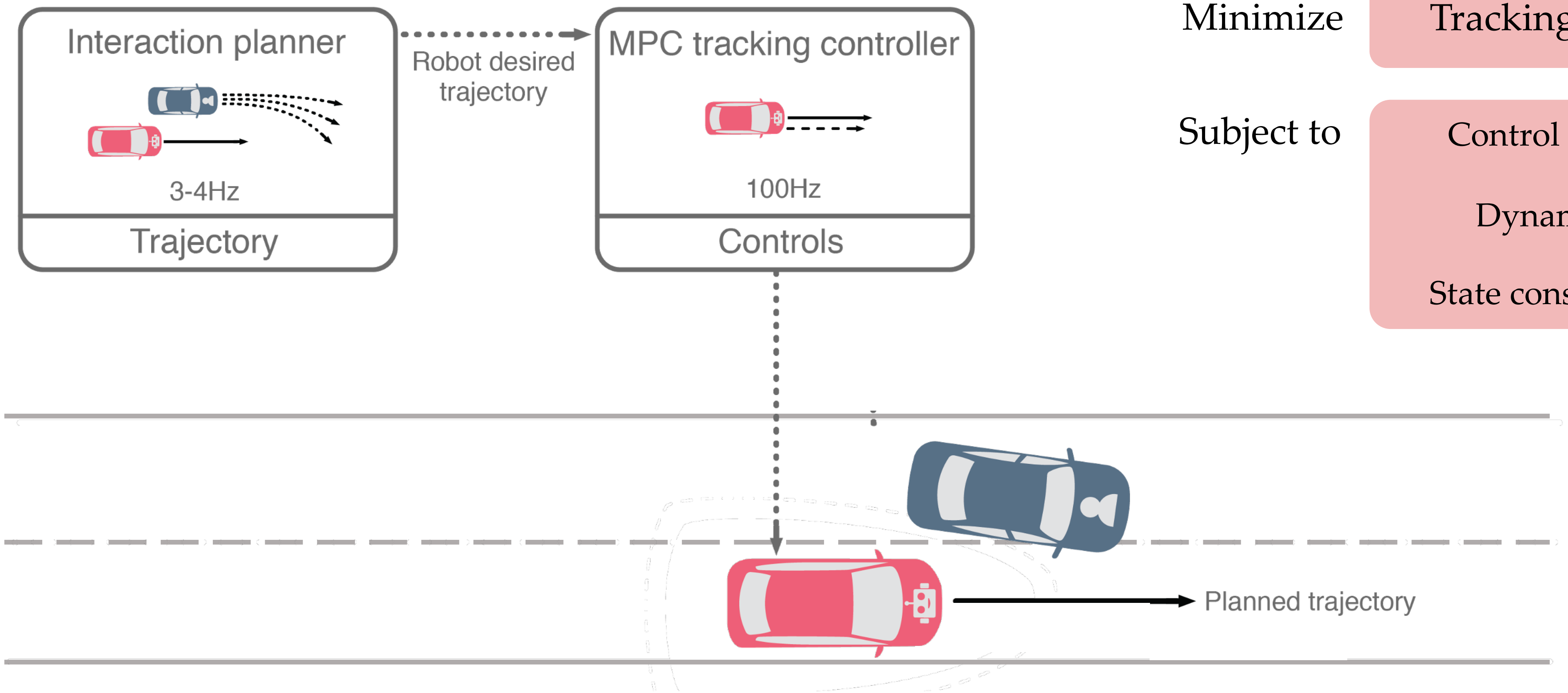
Module #3


knowledge,
data base
Localization
Map Building
"position"
global map
Cognition
Path Planning
mission
commands
environment model
local map
path
Information
Extraction
raw data
Sensing
Perception
see-think-act
Path
Execution
actuator
commands
Acting
Motion Control
Real World
Environment

Motivating example: Traffic-weaving



Planning and control stack



Interaction planner

3-4Hz

Trajectory

Robot desired trajectory

MPC tracking controller

100Hz

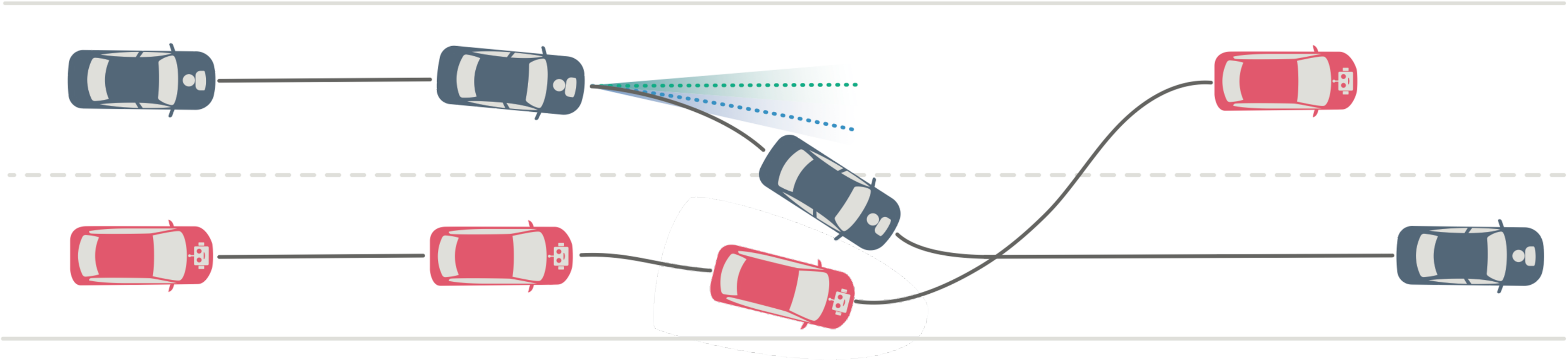
Controls

Planned trajectory

Minimize Tracking error

Subject to
Control limits
Dynamics
State constraints

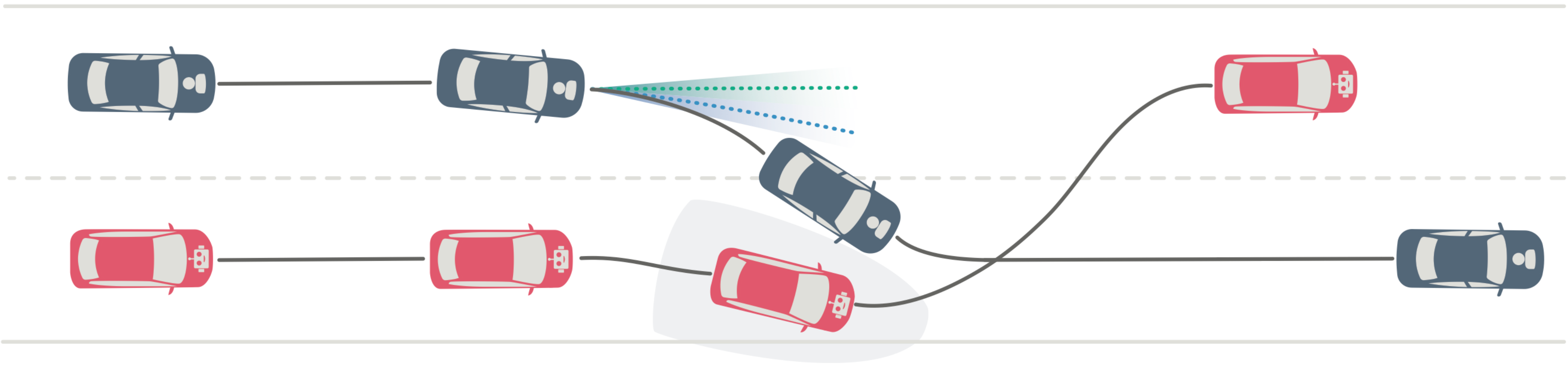
What if the human does something unexpected? What if the prediction model makes a “mistake”? Do we trust our prediction model + planner?



We need a *safety filter*

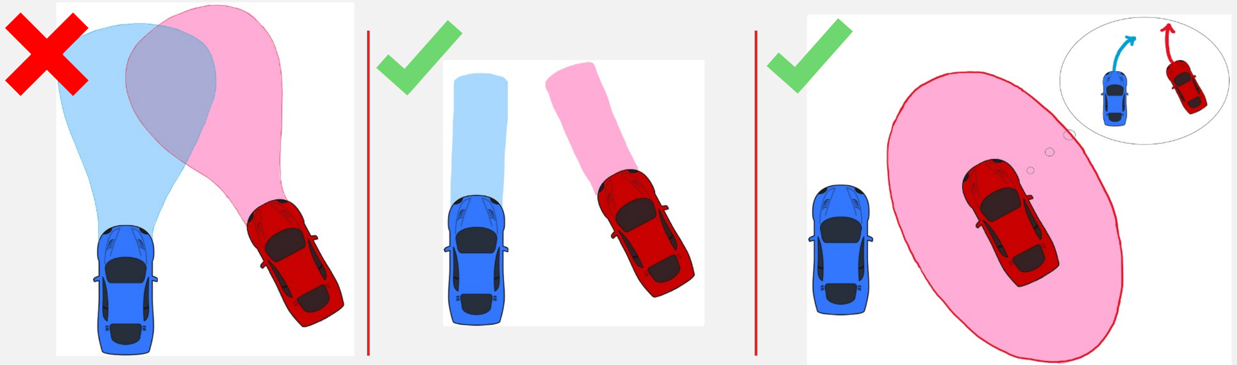
- **Safety filter**: A module that checks the computed controls and adjusts the control as needed.

How to define safety?
What to adjust it to?



To brake or not to brake?

Well, it depends…

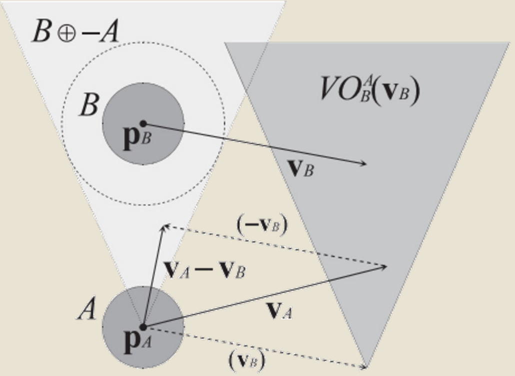

All possible futures Hard braking All possible reactions

What constitutes as *reasonable* behavior?

Different stakeholders make different assumptions

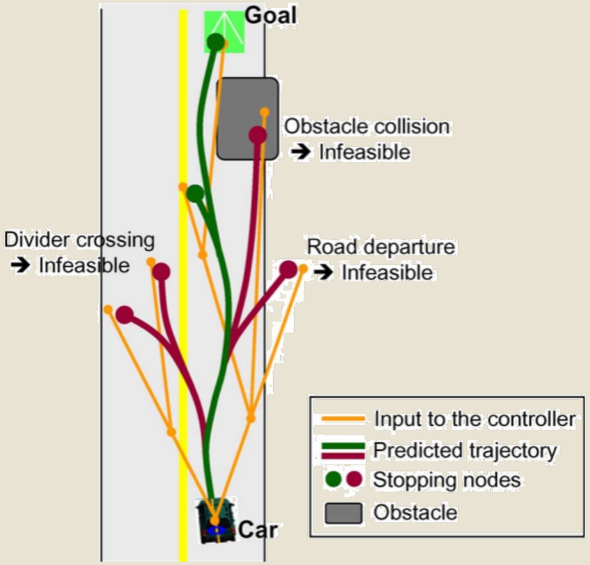
Examples of safe control strategies

Velocity obstacles



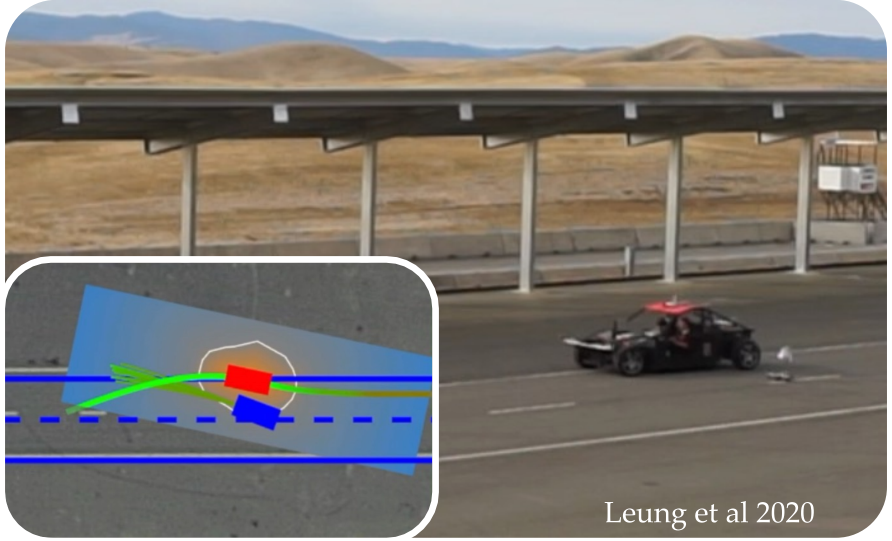
Van der Berg et al 2008

Contingency planning



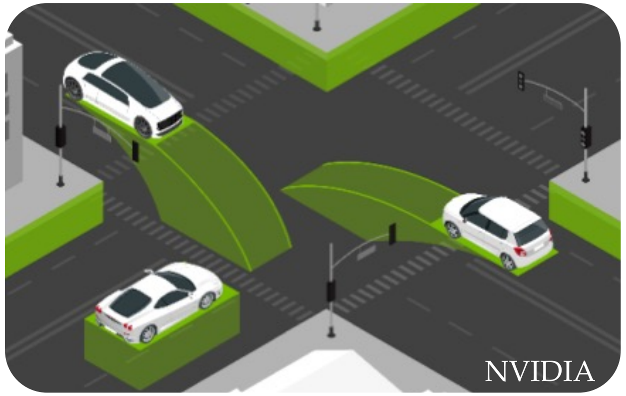
Kuwata et al 2009

Hamilton-Jacobi reachability



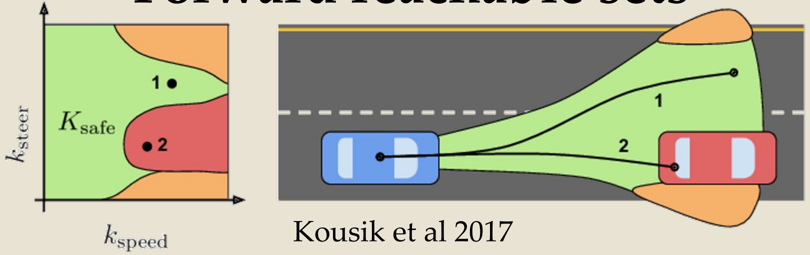
Leung et al 2020

Safety force field (SFF)



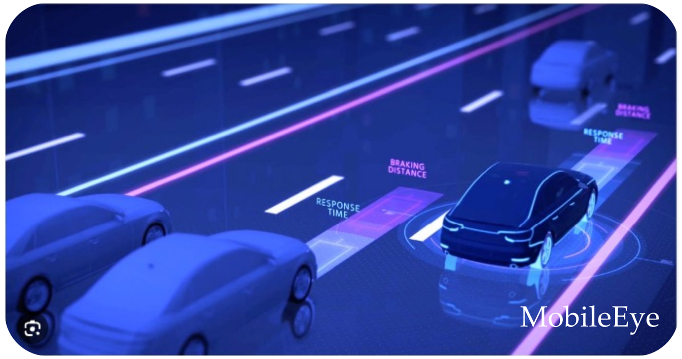
NVIDIA

Forward reachable sets



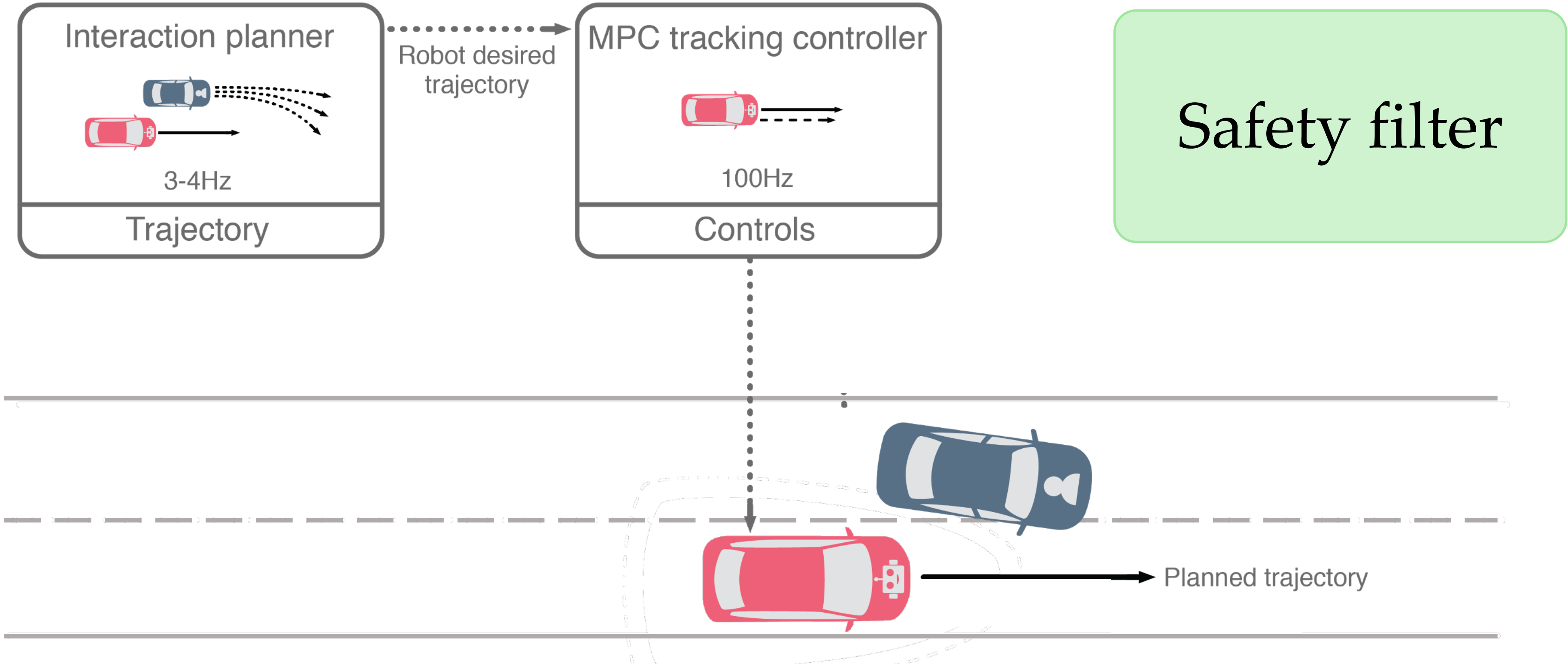
Kousik et al 2017

Responsibility sensitive safety



MobileEye

Planning and control stack



Interaction planner

3-4Hz

Trajectory

Robot desired trajectory

MPC tracking controller

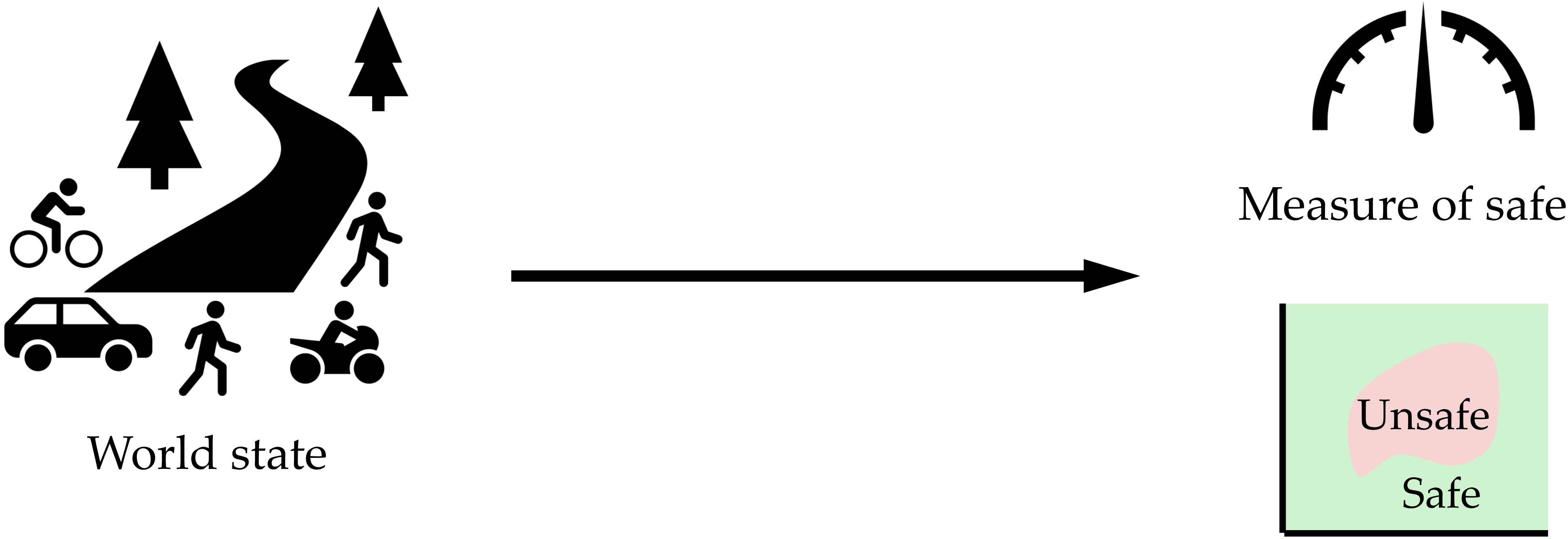
100Hz

Controls

Safety filter

Planned trajectory

A **safety concept** helps determine (1) how safe and (2) what are allowable safe controls to take



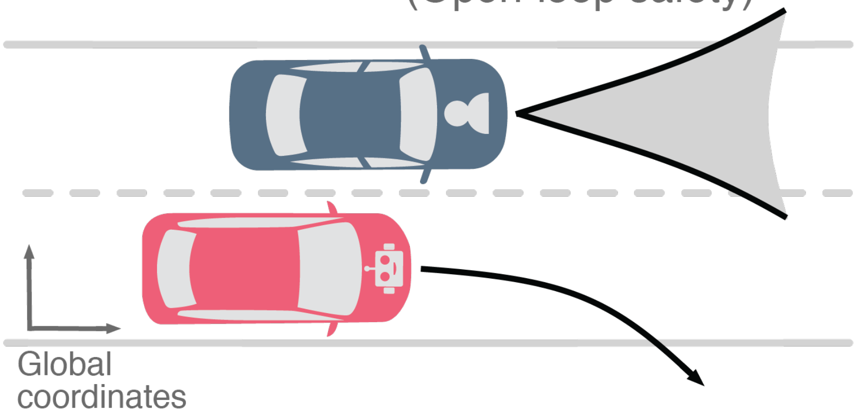
World state

Measure of safety

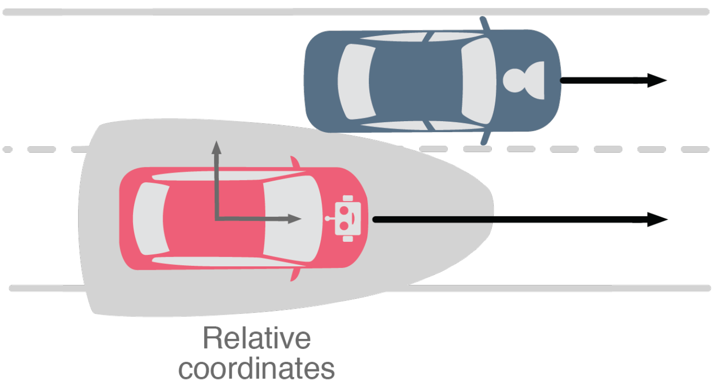
Set of allowable safe controls

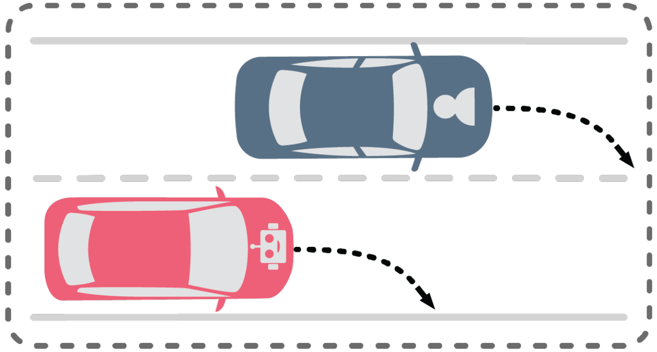
Hamilton-Jacobi reachability

Forward Reachable Set
(Open-loop safety)



Backward Reachable Set
(Closed-loop safety)





Robot’s reactive policy

Sequential decision-making
Dynamic programming

$$\begin{aligned} \min_u \ & \int_0^T c(x(t), u(t))dt + F(x(T)) \\ \text{s.t.} \ & \dot{x} = f(x, u) \\ & u \in \mathcal{U} \\ & x(0) = x_0 \end{aligned}$$

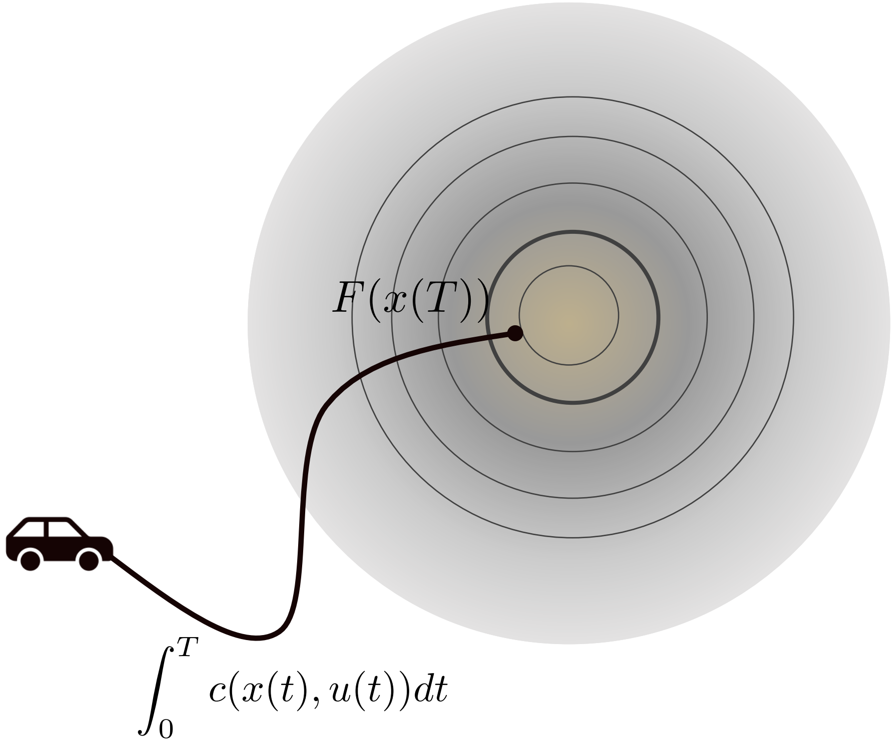
Hamilton-Jacobi-Bellman Equation

$$\frac{\partial V(x,t)}{\partial t} + \min_u \left\{ \frac{\partial V(x,t)}{\partial x} \cdot f(x,u) + c(x,u) \right\} = 0$$

$$V(x,0) = F(x)$$

But we only care if we reach the target set

$$\begin{aligned} \min_{u} \; & \int_0^T c(x(t), u(t))dt + F(x(T)) \\ \text{s.t. } & \dot{x} = f(x, u) \\ & u \in \mathcal{U} \\ & x(0) = x_0 \end{aligned}$$



Backward reachable set

Hamilton-Jacobi-Bellman Equation

$$\frac{\partial V(x,t)}{\partial t} + \min_u \left\{ \frac{\partial V(x,t)}{\partial x} \cdot f(x,u) + c(x,u) \right\} = 0$$

$$V(x,0) = F(x)$$

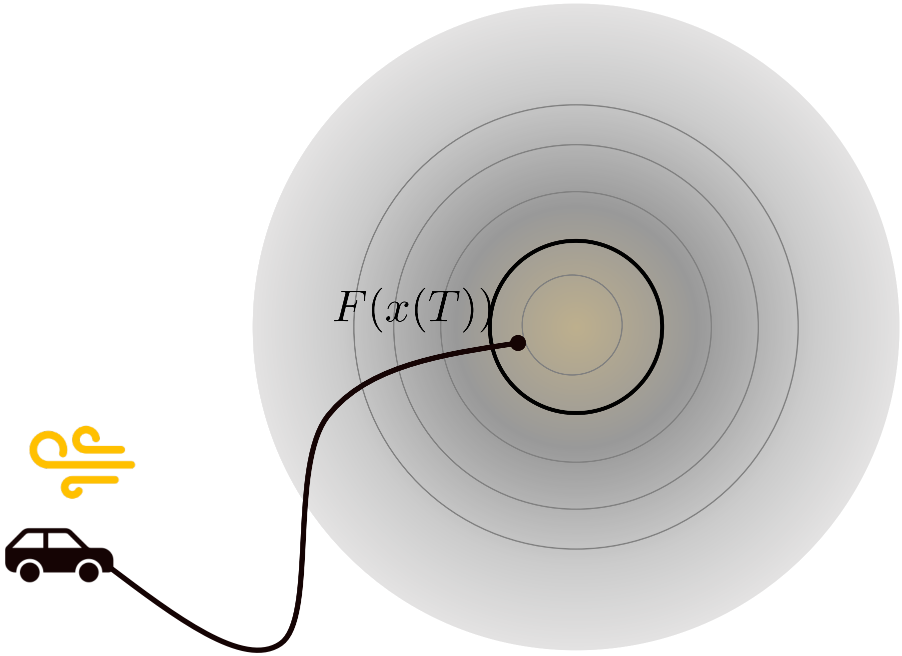
The value function represents the terminal cost of the system at the final time.

$$F(x) \leq 0 \Leftrightarrow x \in \mathcal{T}$$

$$F(x) > 0 \Leftrightarrow x \notin \mathcal{T}$$

$$V(x,-T) < 0 \Rightarrow \text{System will end up in } \mathcal{T}$$

Adding disturbances

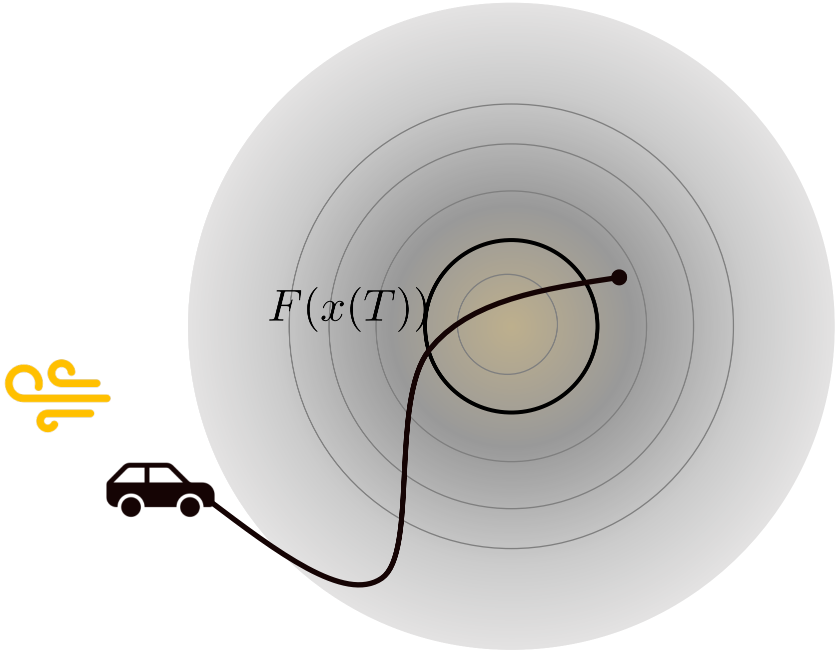


Hamilton-Jacobi-Isaacs Equation

$$\frac{\partial V(x,t)}{\partial t} + \min_u \max_d \left\{ \frac{\partial V(x,t)}{\partial x} \cdot f(x,u,d) \right\} = 0$$

$$V(x,0) = F(x)$$

Backward reachable tube



Hamilton-Jacobi-Isaacs Equation

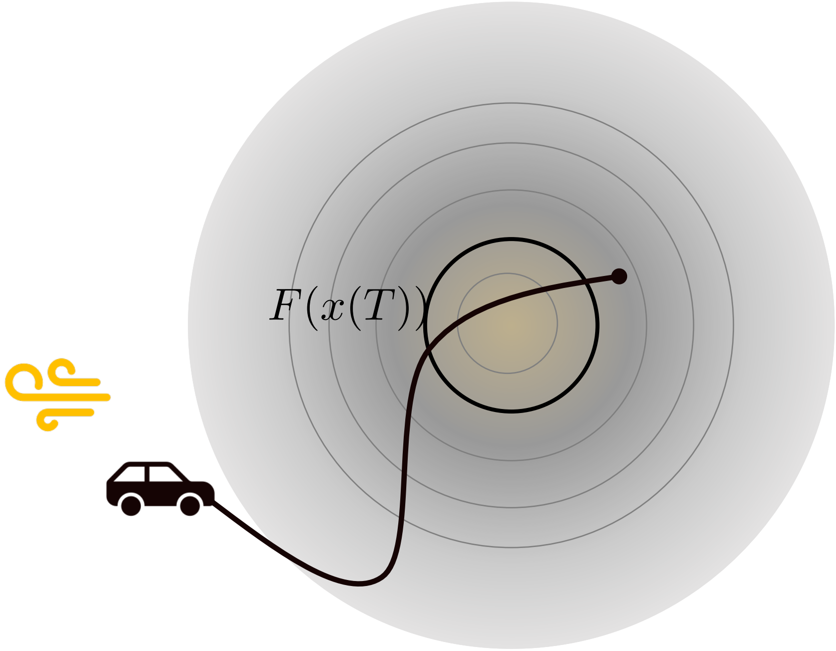
$$\frac{\partial V(x,t)}{\partial t} + \min(0, \mathcal{H}(x,t)) = 0$$

$$\mathcal{H}(x,t) = \min_u \max_d \left\{ \frac{\partial V(x,t)}{\partial x} \cdot f(x,u,d) \right\}$$

$$V(x,0) = F(x)$$

"Take the minimum value along the trajectory"

Backward reachable tube



Hamilton-Jacobi-Isaacs Equation

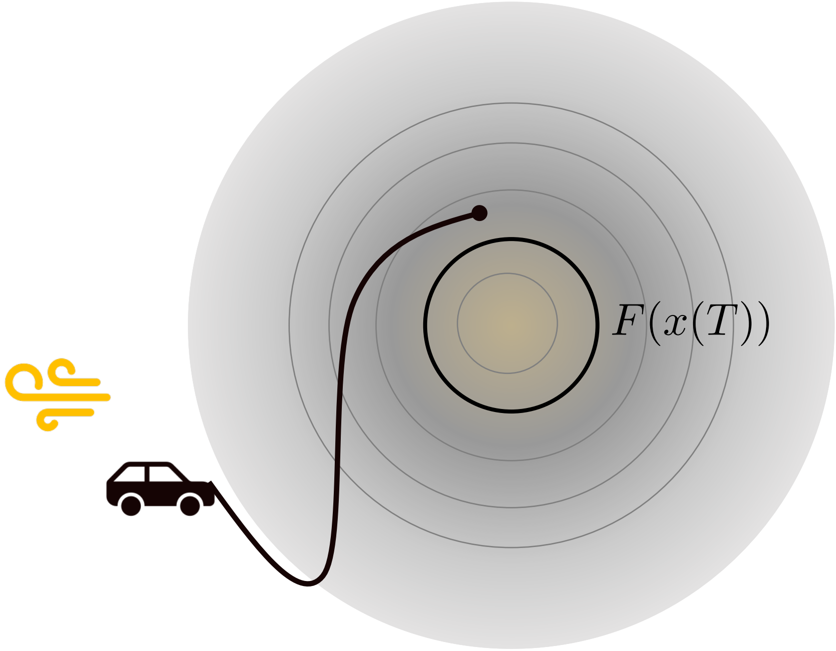
$$\frac{\partial V(x,t)}{\partial t} + \min(0, \mathcal{H}(x,t)) = 0$$

$$\mathcal{H}(x,t) = \min_{u} \max_{d} \left\{ \frac{\partial V(x,t)}{\partial x} \cdot f(x,u,d) \right\}$$

$$V(x,0) = F(x)$$

"Take the minimum value along the trajectory"

Also consider *avoiding* the target set



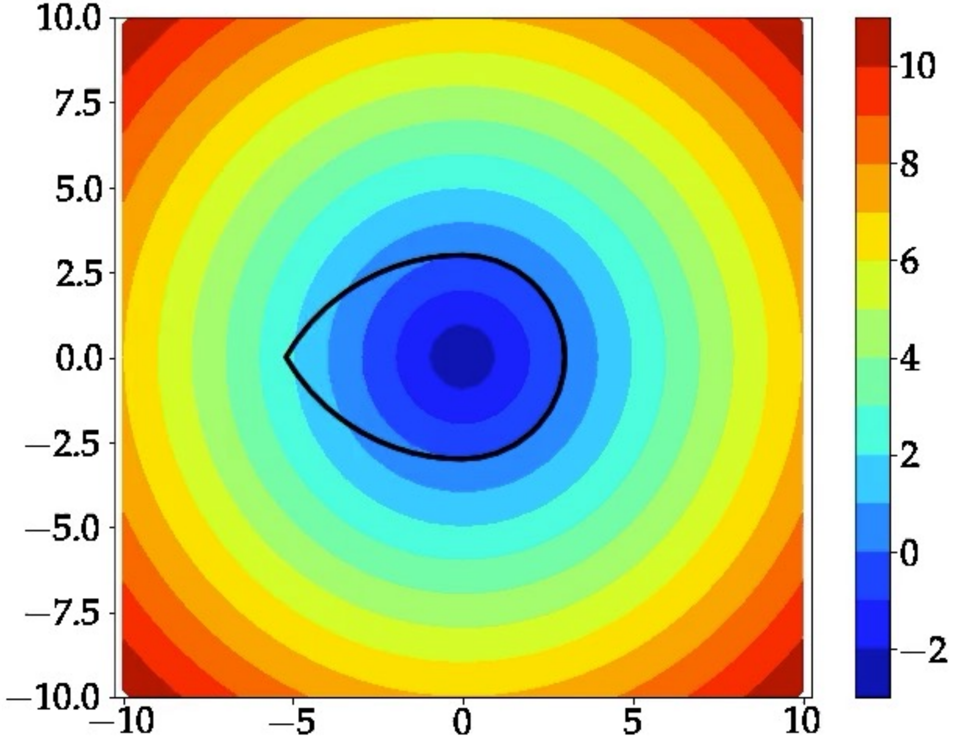
Hamilton-Jacobi-Isaacs Equation

$$\frac{\partial V(x,t)}{\partial t} + \max_u \min_d \left\{ \frac{\partial V(x,t)}{\partial x} \cdot f(x,u) \right\} = 0$$

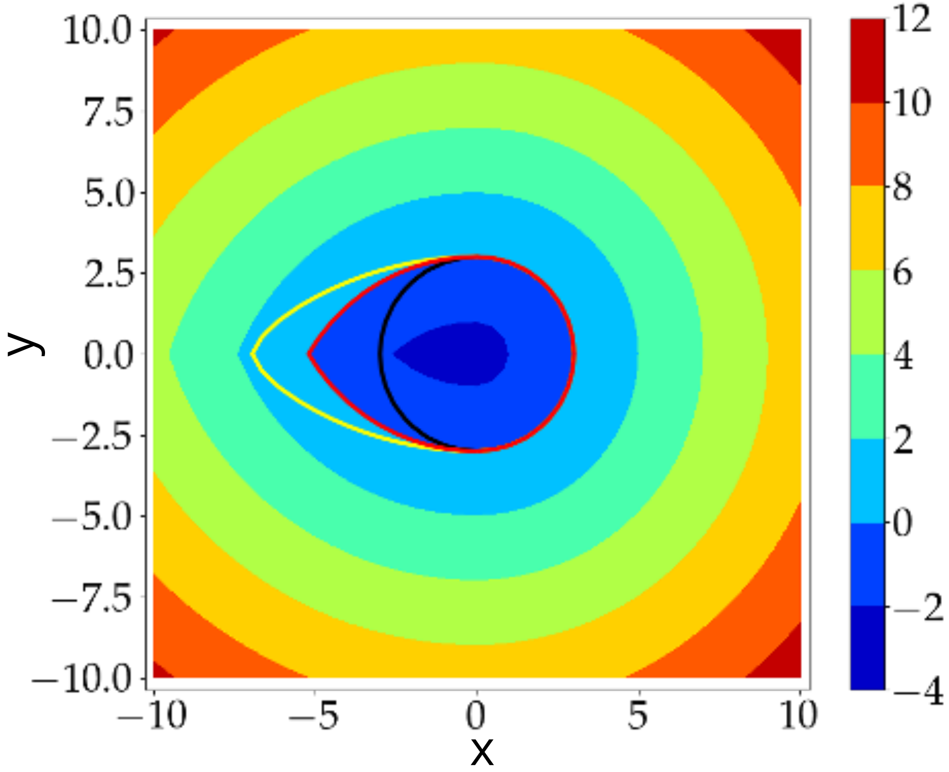
$$V(x,0) = F(x)$$

Dubins car example (avoid)

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix} = \begin{bmatrix}v\cos\theta\\ v\sin\theta\\ u+d\end{bmatrix}$$

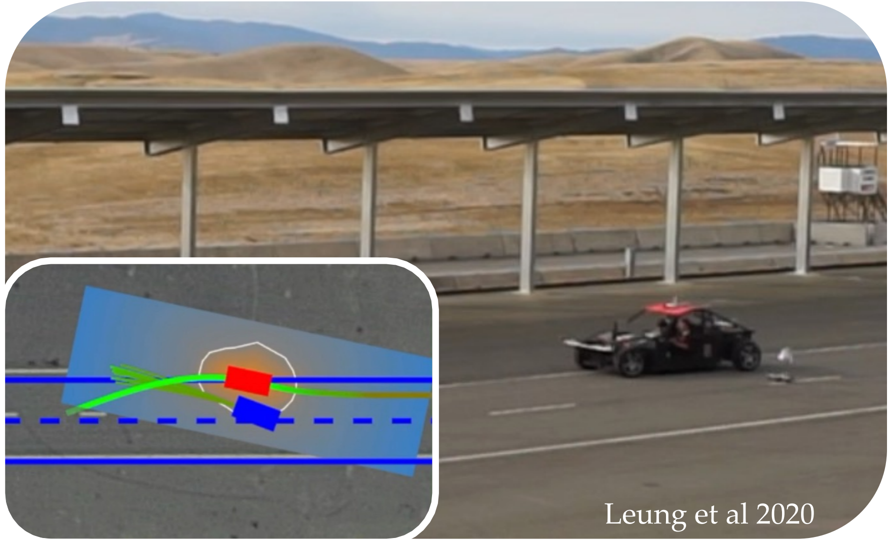


With (yellow) and without (red) disturbance



Can apply to *relative dynamics*

Hamilton-Jacobi reachability


Leung et al 2020

Hamilton-Jacobi-Isaacs partial differential equation

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

$$V(x,0) = v(x)$$

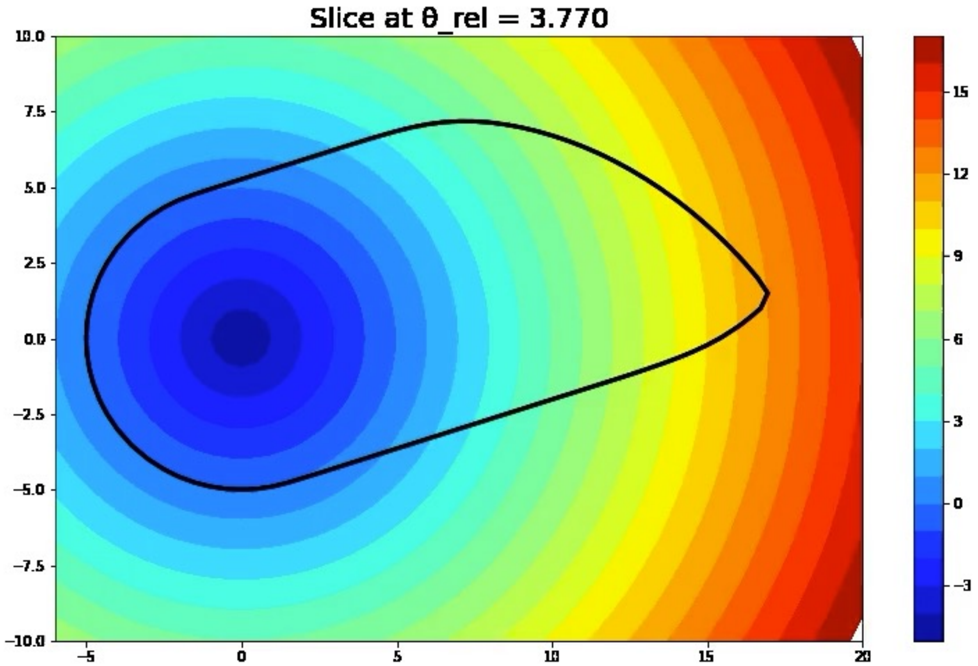
Measure of safety

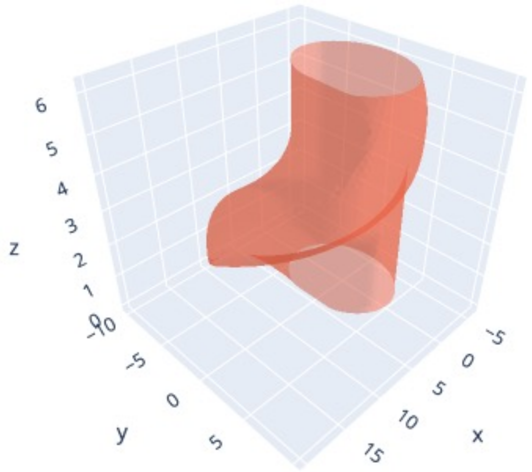
Robot avoids collision

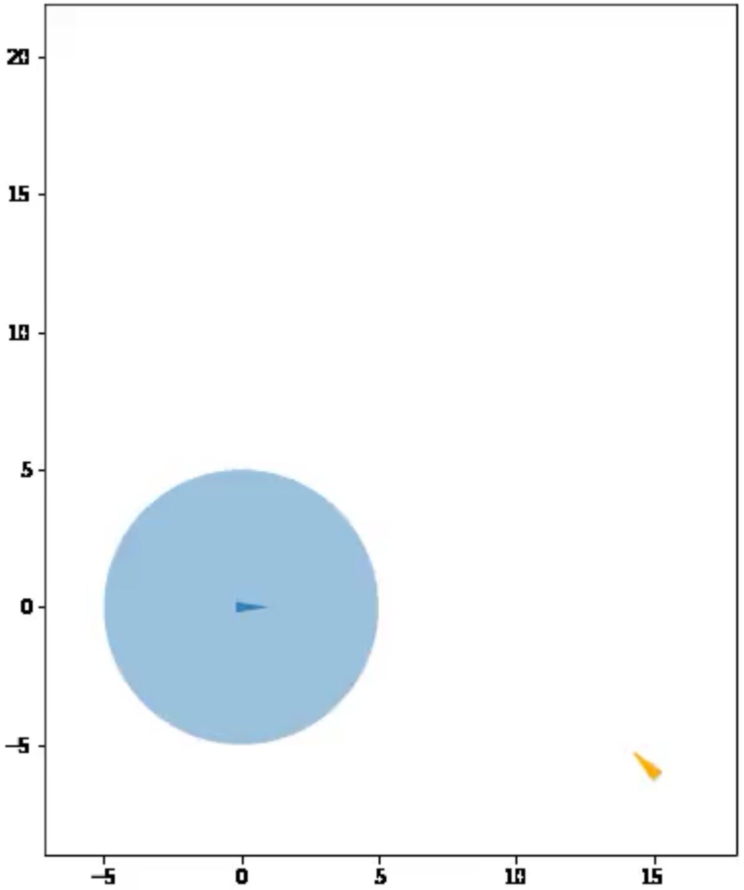
Human seeks collision

Aircraft collision avoidance

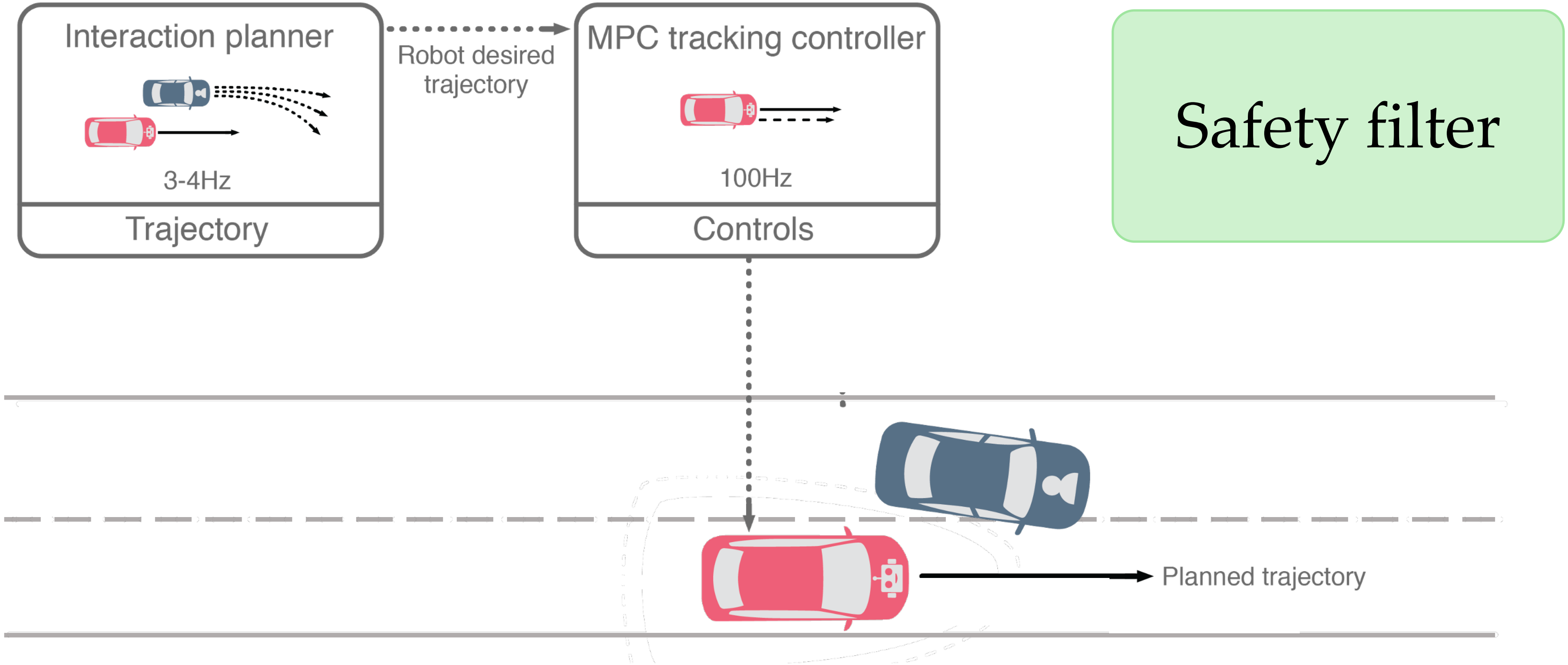
$$\begin{bmatrix} \dot{x}_{\mathrm{rel}} \\ \dot{y}_{\mathrm{rel}} \\ \dot{\theta}_{\mathrm{rel}} \end{bmatrix} = \begin{bmatrix} -v_{\mathrm{a}} + v_{\mathrm{b}} \cos \theta_{\mathrm{rel}} + y_{\mathrm{rel}} u_{\mathrm{a}} \\ v_{\mathrm{b}} \sin \theta_{\mathrm{rel}} - x_{\mathrm{rel}} u_{\mathrm{a}} \\ u_{\mathrm{b}} - u_{\mathrm{a}} \end{bmatrix}$$







Planning and control stack



Interaction planner

3-4Hz

Trajectory

Robot desired trajectory

MPC tracking controller

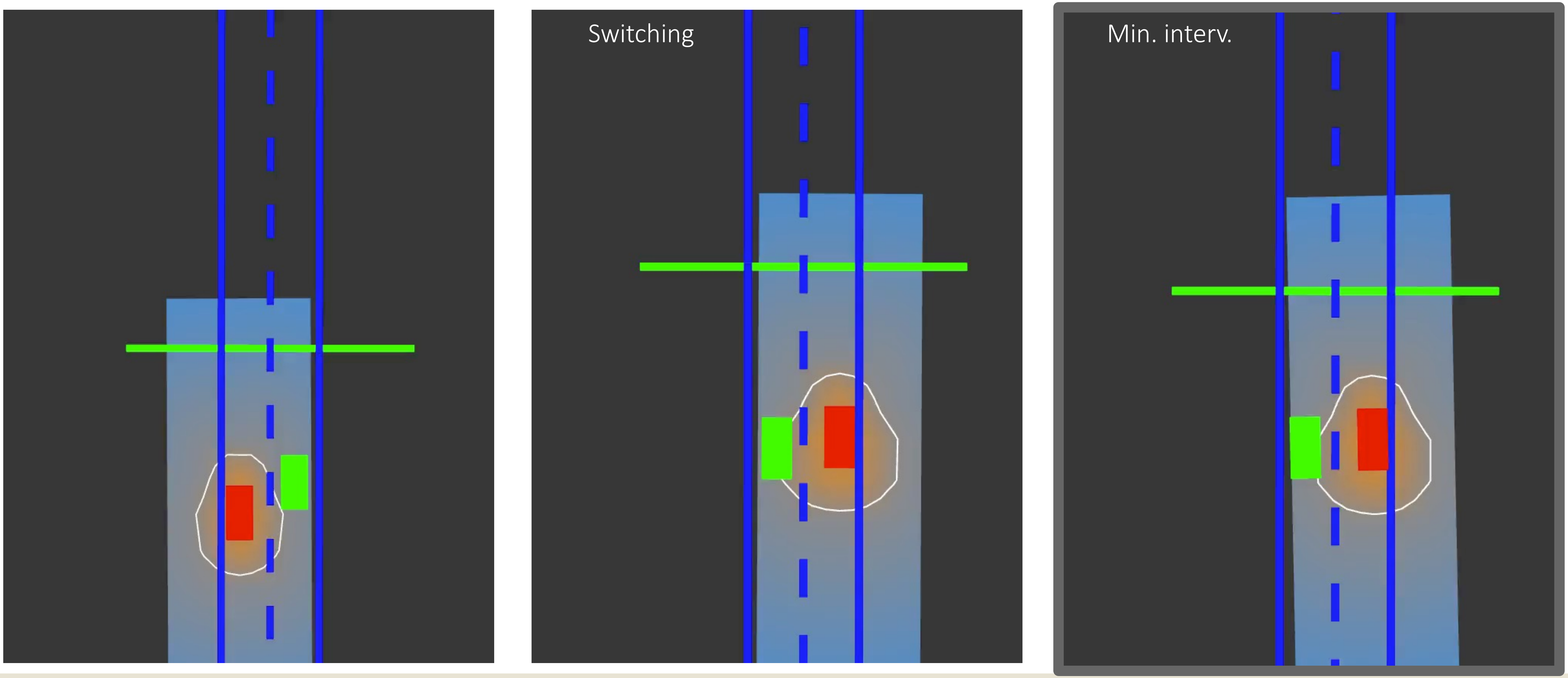
100Hz

Controls

Safety filter

Planned trajectory

Using HJ BRTs as safety filters



Unification of Safety Concepts via Optimal Control Theory

Depends on the assumptions you make about other agents when evaluating safety

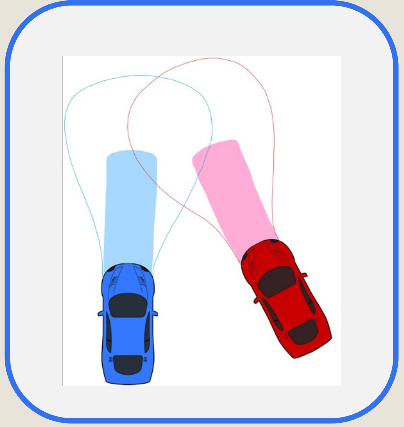
Hamilton-Jacobi Reachability

Open-loop
“non-reactive” policies

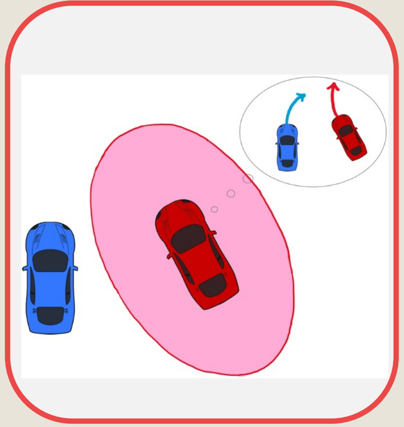
Closed-loop
“reactive” policies

Consider *all* possible behaviors
Full forward reachable set

. . .

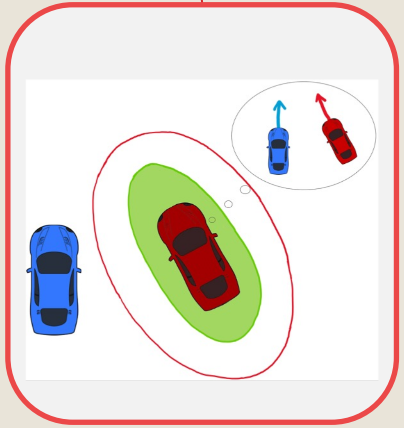


Consider only a *subset* of possible behaviors
*e.g., hard-braking **(SFF)***



Guard against *all* possible policies
Including worst-case outcomes

. . .



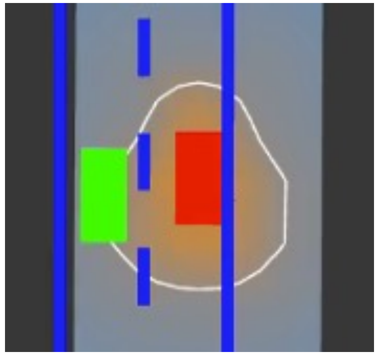
Guard against a *subset* of possible policies
Assumptions on other agent’s behaviors

What are “reasonable” assumptions about how other agents behave?

Hamilton-Jacobi-Isaacs partial differential equation

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

$$V(x,0) = v(x)$$



Overly-conservative assumptions lead to impractical safety concepts!

How to pick “reasonable” choices for U and D?