AA203
Optimal and Learning-based Control

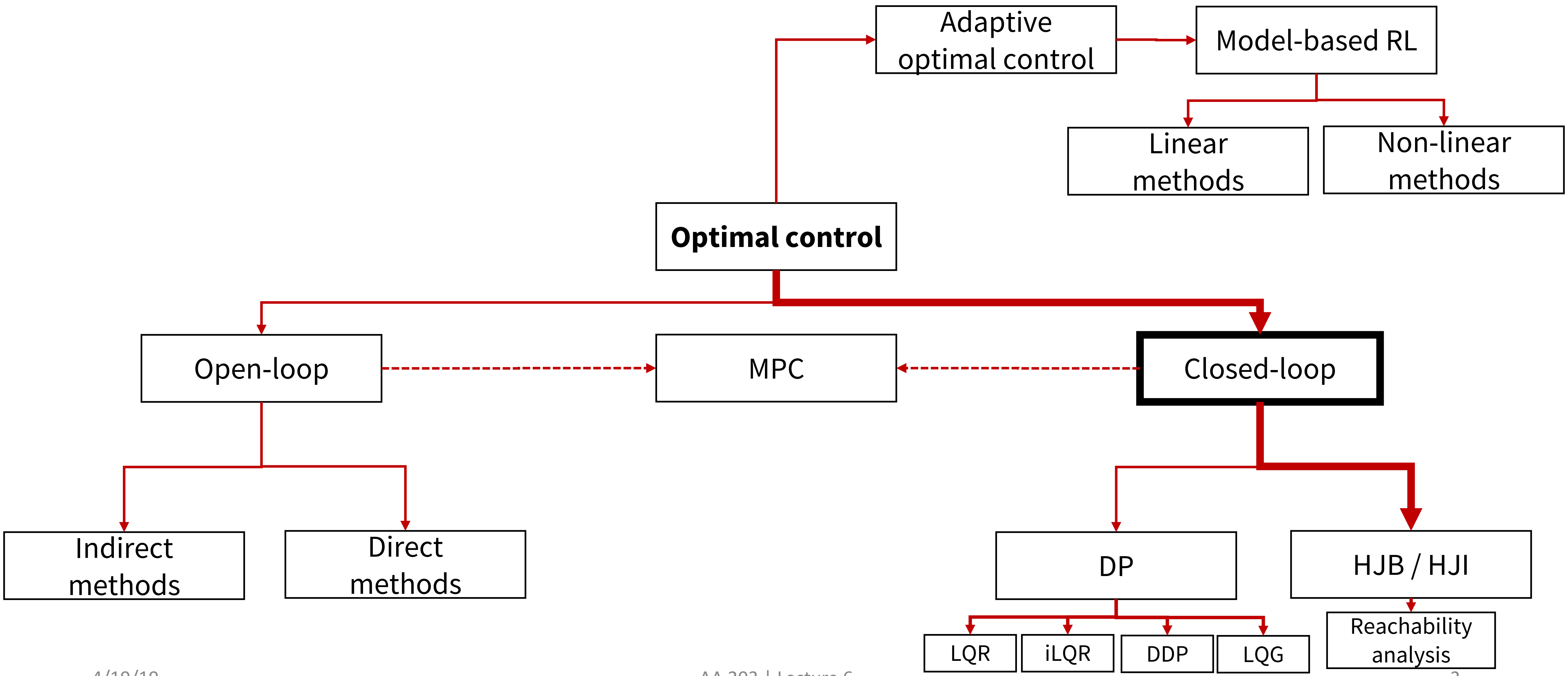
HJI Equation and reachability analysis*

Stanford University

*Slides based on material provided by Mo Chen, mochen@cs.sfu.ca, https://sfumars.com/mo-chen/

Autonomous Systems Lab

Roadmap


Adaptive optimal control
Model-based RL
Linear methods
Non-linear methods
Optimal control
Open-loop
MPC
Closed-loop
Indirect methods
Direct methods
DP
HJB / HJI
LQR
iLQR
DDP
LQG
Reachability analysis

Two-person, zero-sum differential games

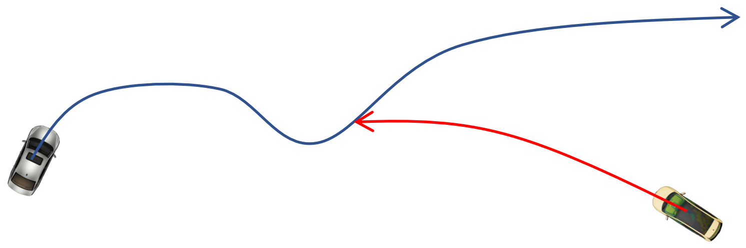
- What if there is another player (e.g., nature) that interferes with the fulfillment of our objective?

Two person differential game:

- Model: $\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{d}(t))$ (*joint* system dynamics),
- Cost: $J(\mathbf{x}(t)) = h(\mathbf{x}(0)) + \int_t^0 g(\mathbf{x}(\tau), \mathbf{u}(\tau), \mathbf{d}(\tau))\, d\tau$
- Player 1, with control $\mathbf{u}(\tau)$, will attempt to maximize J, while Player 2, with control $\mathbf{d}(t)$, will aim to minimize J, subject to the *joint* system dynamics
- $\mathbf{x}(\tau)$ is the *joint* system state

Information patter

- To fully specify the game, we need to specify the *information pattern*
- "Open-loop" strategies
  - Player 1, with control $\mathbf{u}(\tau)$, declares entire plan
  - Player 2, with control $\mathbf{d}(\tau)$, responds optimally
  - Conservative, unrealistic, but computationally cheap



- "Non-anticipative" strategies
  - Other robot acts based on state and control trajectory up to current time
  - Notation: $\mathbf{d}(\cdot) = \Gamma[\mathbf{u}](\cdot)$
  - Disturbance still has the advantage: it gets to react to the control!

HJI equation

Key idea: apply principle of optimality

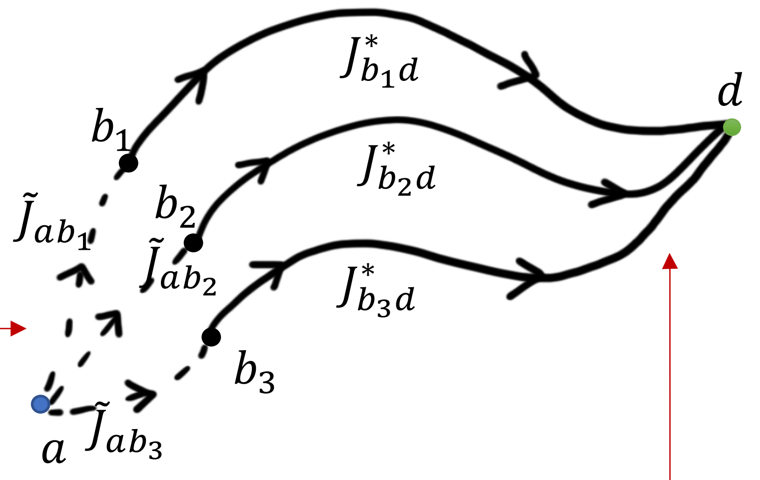
The "truncated" (lower-value) problem is

$$J(\mathbf{x}(t), t) = \min_{\Gamma[\mathbf{u}](\cdot)} \max_{\mathbf{u}(\cdot)} \left[\int_t^0 g\big(\mathbf{x}(\tau), \mathbf{u}(\tau), \mathbf{d}(\tau)\big) d\tau + h\big(\mathbf{x}(0)\big) \right]$$

Worst-case disturbance -- does the opposite of the control

HJI equation

- Dynamic programming principle:



$$J(\mathbf{x}(t), t) = \min_{\Gamma[\mathbf{u}](\cdot)} \max_{\mathbf{u}(\cdot)} \left[\int_t^{t+\Delta t} g\big(\mathbf{x}(\tau), \mathbf{u}(\tau), \mathbf{d}(\tau)\big) d\tau + J(\mathbf{x}(t+\Delta t), t+\Delta t) \right]$$

- Approximate integral and Taylor expand $J(\mathbf{x}(t+\Delta t), t+\Delta t)$
- Derive Hamilton-Jacobi-Isaacs partial differential equation (HJI PDE)

HJI equation

- Approximations for small Δt:

$$J(\mathbf{x}(t),t) = \min_{\Gamma[\mathbf{u}](\cdot)} \max_{\mathbf{u}(\cdot)} \left[\underbrace{\int_t^{t+\Delta t} g\big(\mathbf{x}(\tau),\mathbf{u}(\tau),\mathbf{d}(\tau)\big)d\tau}_{g\left(\mathbf{x}(t),\mathbf{u}(t),\mathbf{d}(t)\right)\Delta t} + \underbrace{J(\overbrace{\mathbf{x}(t+\Delta t)}^{\mathbf{x}(t)+\Delta t f(\mathbf{x},\mathbf{u},\mathbf{d})}, t+\Delta t)}_{J(\mathbf{x}(t),t) + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f\left(\mathbf{x}(t),\mathbf{u}(t),\mathbf{d}(t)\right) + \frac{\partial V}{\partial t}\Delta t} \right]$$

  - Omit t dependence…

$$J(\mathbf{x},t) = \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x},\mathbf{u},\mathbf{d})\Delta t + J(\mathbf{x},t) + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f(\mathbf{x},\mathbf{u},\mathbf{d}) + \frac{\partial J}{\partial t}\Delta t \right]$$

    - Assume constant u and d → Optimization over vectors, not functions!
    - Order of max and min reverse: disturbance has the advantage

  - $J(\mathbf{x},t)$ does not depend on $\mathbf{u}$ or $\mathbf{d}$

$$J(\mathbf{x},t) = J(\mathbf{x},t) + \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x},\mathbf{u},\mathbf{d})\Delta t + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f(\mathbf{x},\mathbf{u},\mathbf{d}) + \frac{\partial J}{\partial t}\Delta t \right]$$

HJI equation

- Approximations for small Δt:

$$J(\mathbf{x}(t),t) = \min_{\Gamma[\mathbf{u}](\cdot)} \max_{\mathbf{u}(\cdot)} \left[\underbrace{\int_t^{t+\Delta t} g\big(\mathbf{x}(\tau),\mathbf{u}(\tau),\mathbf{d}(\tau)\big) d\tau}_{g\left(\mathbf{x}(t),\mathbf{u}(t),\mathbf{d}(t)\right)\Delta t} + \underbrace{J(\overbrace{\mathbf{x}(t+\Delta t)}^{\mathbf{x}(t)+\Delta t f(\mathbf{x},\mathbf{u},\mathbf{d})}, t+\Delta t)}_{J(\mathbf{x}(t),t) + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f\left(\mathbf{x}(t),\mathbf{u}(t),\mathbf{d}(t)\right) + \frac{\partial V}{\partial t}\Delta t} \right]$$

  - Omit t dependence…

$$J(\mathbf{x},t) = \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x},\mathbf{u},\mathbf{d})\Delta t + J(\mathbf{x},t) + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f(\mathbf{x},\mathbf{u},\mathbf{d}) + \frac{\partial J}{\partial t}\Delta t \right]$$

    - Assume constant u and d → Optimization over vectors, not functions!
    - Order of max and min reverse: disturbance has the advantage

  - $J(\mathbf{x},t)$ does not depend on $\mathbf{u}$ or $\mathbf{d}$

$$\cancel{J(\mathbf{x},t)} = \cancel{J(\mathbf{x},t) +} \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x},\mathbf{u},\mathbf{d})\Delta t + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f(\mathbf{x},\mathbf{u},\mathbf{d}) + \frac{\partial J}{\partial t}\Delta t \right]$$

HJI equation

- Approximations for small Δt:

$$J(\mathbf{x}(t), t) = \min_{\Gamma[\mathbf{u}](\cdot)} \max_{\mathbf{u}(\cdot)} \left[\underbrace{\int_t^{t+\Delta t} g\big(\mathbf{x}(\tau), \mathbf{u}(\tau), \mathbf{d}(\tau)\big) d\tau}_{g\left(\mathbf{x}(t), \mathbf{u}(t), \mathbf{d}(t)\right)\Delta t} + \underbrace{J(\overbrace{\mathbf{x}(t+\Delta t)}^{\mathbf{x}(t) + \Delta t f(\mathbf{x}, \mathbf{u}, \mathbf{d})}, t+\Delta t)}_{J(\mathbf{x}(t), t) + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f\left(\mathbf{x}(t), \mathbf{u}(t), \mathbf{d}(t)\right) + \frac{\partial V}{\partial t}\Delta t} \right]$$

  - Omit t dependence…

$$J(\mathbf{x}, t) = \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x}, \mathbf{u}, \mathbf{d})\Delta t + J(\mathbf{x}, t) + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f(\mathbf{x}, \mathbf{u}, \mathbf{d}) + \frac{\partial J}{\partial t}\Delta t \right]$$

    - Assume constant u and d → Optimization over vectors, not functions!
    - Order of max and min reverse: disturbance has the advantage

  - $J(\mathbf{x}, t)$ does not depend on $\mathbf{u}$ or $\mathbf{d}$

$$0 = \frac{\partial J}{\partial t}\cancel{\Delta t} + \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x}, \mathbf{u}, \mathbf{d})\cancel{\Delta t} + \frac{\partial J}{\partial \mathbf{x}} \cdot \cancel{\Delta t} f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right]$$

HJI equation

- Approximations for small Δt:

$$J(\mathbf{x}(t),t) = \min_{\Gamma[\mathbf{u}](\cdot)} \max_{\mathbf{u}(\cdot)} \left[\underbrace{\int_t^{t+\Delta t} g\big(\mathbf{x}(\tau),\mathbf{u}(\tau),\mathbf{d}(\tau)\big)d\tau}_{g\left(\mathbf{x}(t),\mathbf{u}(t),\mathbf{d}(t)\right)\Delta t} + \underbrace{J(\overbrace{\mathbf{x}(t+\Delta t)}^{\mathbf{x}(t)+\Delta t f(\mathbf{x},\mathbf{u},\mathbf{d})}, t+\Delta t)}_{J(\mathbf{x}(t),t) + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f\left(\mathbf{x}(t),\mathbf{u}(t),\mathbf{d}(t)\right) + \frac{\partial V}{\partial t}\Delta t} \right]$$

  - Omit t dependence…

$$J(\mathbf{x},t) = \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x},\mathbf{u},\mathbf{d})\Delta t + J(\mathbf{x},t) + \frac{\partial J}{\partial \mathbf{x}} \cdot \Delta t f(\mathbf{x},\mathbf{u},\mathbf{d}) + \frac{\partial J}{\partial t}\Delta t \right]$$

    - Assume constant u and d → Optimization over vectors, not functions!
    - Order of max and min reverse: disturbance has the advantage

  - $J(\mathbf{x},t)$ does not depend on $\mathbf{u}$ or $\mathbf{d}$

$$0 = \frac{\partial J}{\partial t} + \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x},\mathbf{u},\mathbf{d}) + \frac{\partial J}{\partial \mathbf{x}} \cdot f(\mathbf{x},\mathbf{u},\mathbf{d}) \right]$$

HJI equation

The end result is the Hamilton-Jacobi-Isaacs (HJI) equation

$$\boxed{0 = \frac{\partial J}{\partial t} + \max_{\mathbf{u}} \min_{\mathbf{d}} \left[g(\mathbf{x}, \mathbf{u}, \mathbf{d}) + \frac{\partial J}{\partial \mathbf{x}} \cdot f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right]}$$

with boundary condition

$$\boxed{J(\mathbf{x}, 0) = h(\mathbf{x})}$$

- Given the cost-to-go function, the optimal control for Player 1 is

$$\mathbf{u}^*(\mathbf{x}, t) = \arg\max_{\mathbf{u}} \min_{\mathbf{d}} g(\mathbf{x}, \mathbf{u}, \mathbf{d}) + \frac{\partial J}{\partial \mathbf{x}} \cdot f(\mathbf{x}, \mathbf{u}, \mathbf{d})$$

Applications of differential games

- Pursuit-evasion games
  - homicidal chauffeur problem
  - the lady in the lake

- Reachability analysis

- And many more (e.g., in economics)

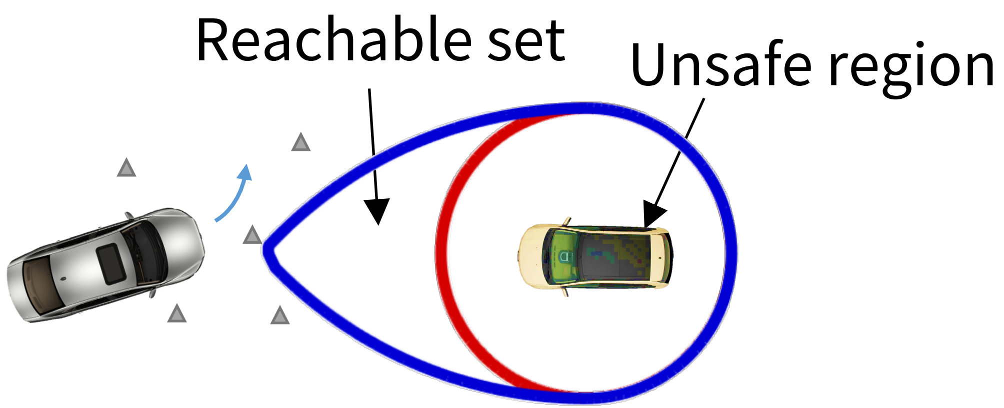
Applications of differential games

- Pursuit-evasion games
  - homicidal chauffeur problem
  - the lady in the lake

- Reachability analysis

- And many more (e.g., in economics)

Reachability analysis: avoidance

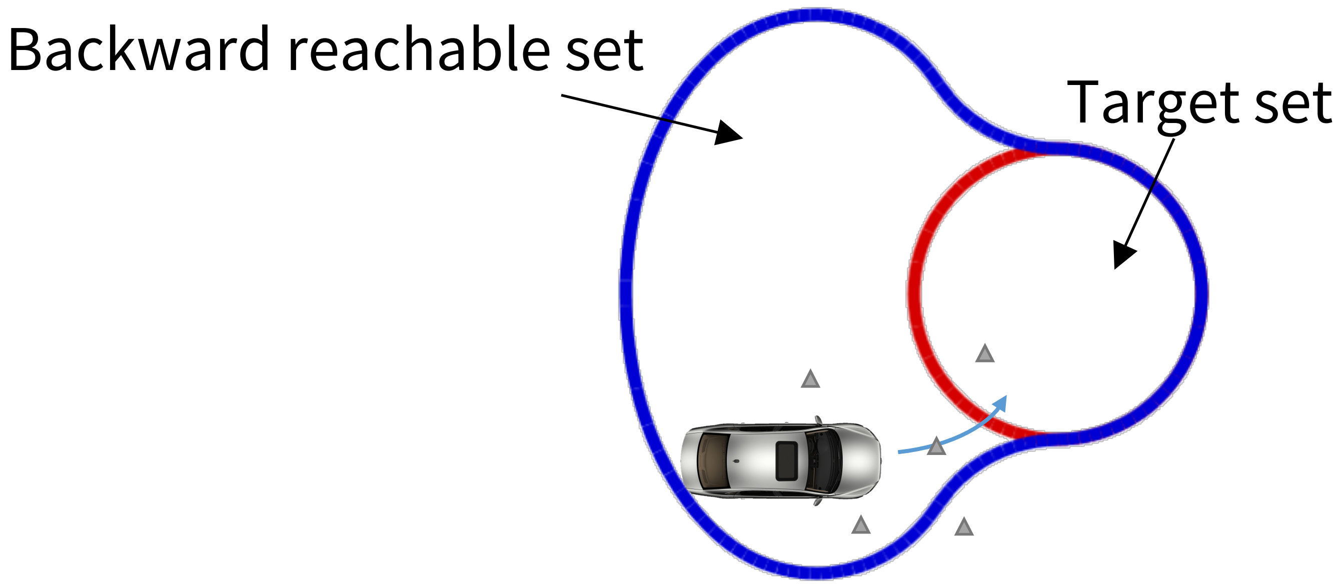


Assumptions:
- Model of robot
- Unsafe region: e.g., obstacle

Control policy

Backward reachable set
(States leading to danger)

Reachability analysis: goal reaching



- Model of robot
- Goal region

Control policy

Backward reachable set
(States leading to goal)

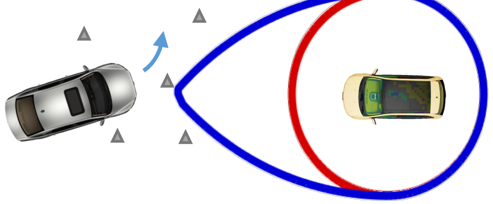
Reachability analysis

- $\mathcal{A}(t) = \{\bar{\mathbf{x}}\colon \exists\Gamma[\boldsymbol{u}](\cdot), \forall\mathbf{u}(\cdot), \dot{\mathbf{x}} = f(\mathbf{x},\mathbf{u},\mathbf{d}), \mathbf{x}(t) = \bar{\mathbf{x}}, \mathbf{x}(0) \in \mathcal{T}\}$

- Model of robot
- Unsafe region

Backward reachable set (states leading to danger)

Control policy



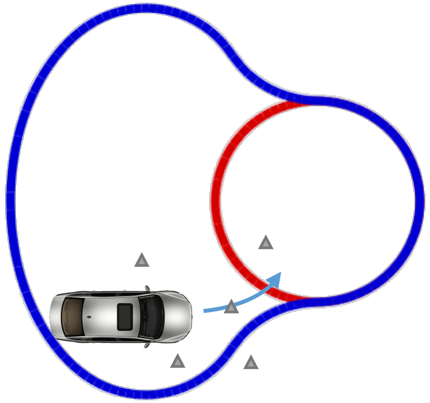
- $\dot{\mathbf{x}} = f(\mathbf{x},\mathbf{u},\mathbf{d})$
- $\mathcal{T}$

- $\mathbf{u}^*(\mathbf{x},t)$

- Model of robot
- Goal region

Control policy

Backward reachable set (states leading to goal)



- $\mathcal{R}(t) = \{\bar{x}\colon \forall\Gamma[\mathbf{u}](\cdot), \exists\mathbf{u}(\cdot), \dot{\mathbf{x}} = f(\mathbf{x},\mathbf{u},\mathbf{d}), \mathbf{x}(t) = \bar{\mathbf{x}}, \mathbf{x}(0) \in \mathcal{T}\}$

Reachability analysis

States at time t satisfying the following:

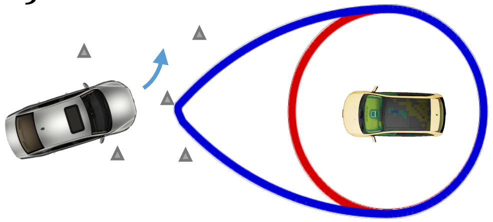
there exists a disturbance such that for all control, system enters target set at $t = 0$

- $\mathcal{A}(t) = \{\bar{\mathbf{x}}: \exists \Gamma[\mathbf{u}](\cdot), \forall \mathbf{u}(\cdot), \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{d}), \mathbf{x}(t) = \bar{\mathbf{x}}, \mathbf{x}(0) \in \mathcal{T}\}$

- Model of robot
- Unsafe region

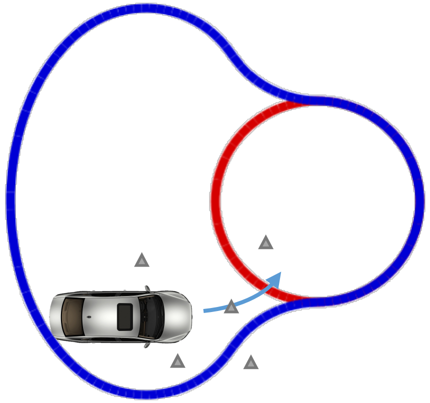
Backward reachable set (States leading to danger)

Control policy



- $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{d})$
- $\mathcal{T}$

- $\mathbf{u}^*(\mathbf{x}, t)$



- Model of robot
- Goal region

Control policy

Backward reachable set (States leading to goal)

- $\mathcal{R}(t) = \{\bar{\mathbf{x}}: \forall \Gamma[\mathbf{u}](\cdot), \exists \mathbf{u}(\cdot), \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{d}), x(t) = \bar{\mathbf{x}}, \mathbf{x}(0) \in \mathcal{T}\}$

States at time t satisfying the following:

for all disturbances, there exists a control such that system enters target set at $t = 0$

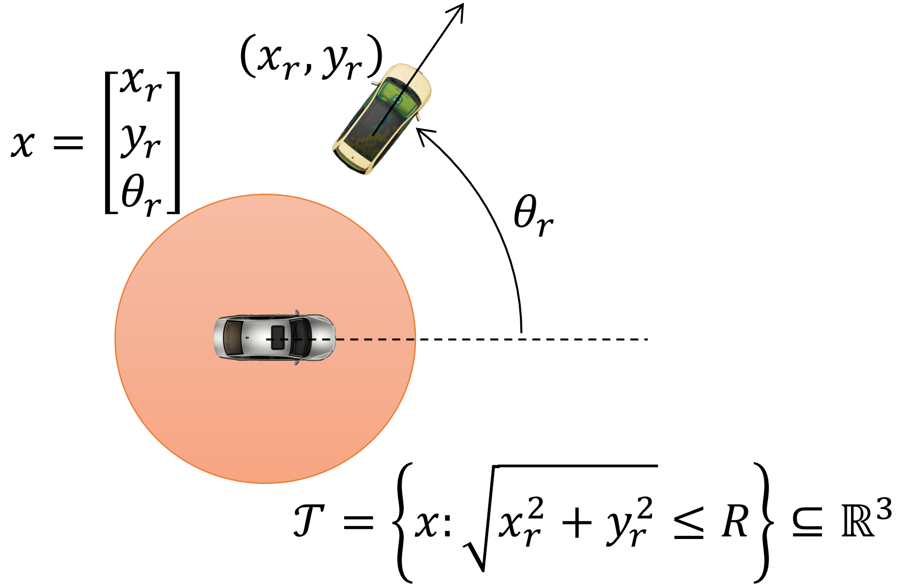
17

From HJI to reachability analysis

- Computation of the BRS entails solving a differential *game of kind*, where the outcome is Boolean (the system either reaches the target set or not)
- One can “encode” this Boolean outcome by (1) removing the running cost and (2) picking the final cost intelligently

From HJI to reachability analysis

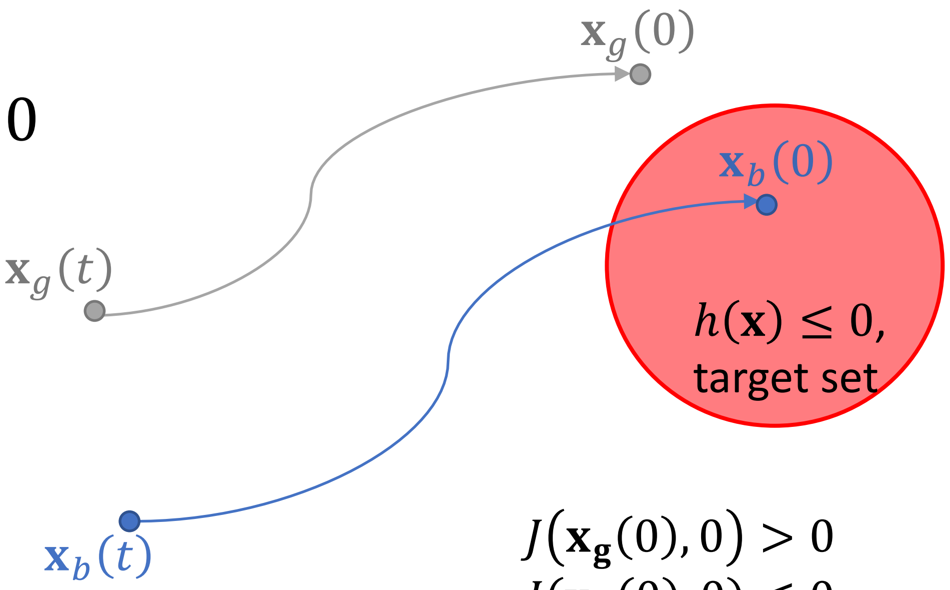
- Hamilton-Jacobi Equation
  - $0 = \frac{\partial J}{\partial t} + \max_{\mathbf{d}} \min_{\mathbf{u}} \left[g(\mathbf{x}, \mathbf{u}, \mathbf{d}) + \frac{\partial J}{\partial \mathbf{x}} \cdot f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right],\ J(\mathbf{x}, 0) = h(\mathbf{x})$
- Remove running cost
  - $0 = \frac{\partial J}{\partial t} + \max_{\mathbf{d}} \min_{\mathbf{u}} \left[\frac{\partial J}{\partial \mathbf{x}} \cdot f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right],\ J(\mathbf{x}, 0) = h(\mathbf{x})$
- Pick final cost such that
  - $\mathbf{x} \in \mathcal{T} \Leftrightarrow h(\mathbf{x}) \leq 0$
  - Example: If $\mathcal{T} = \left\{ \mathbf{x} : \sqrt{x_r^2 + y_r^2} \leq R \right\} \subseteq \mathbb{R}^3$, we can pick $h(x_r, y_r, \theta_r) = \sqrt{x_r^2 + y_r^2} - R$



Pick Final Cost

- Pick final cost such that
  - $x \in \mathcal{T} \Leftrightarrow h(\mathbf{x}) \leq 0$
  - If $\mathcal{T} = \left\{x: \sqrt{x_r^2 + y_r^2} \leq R\right\} \subseteq \mathbb{R}^3$, we can pick $h(x_r, y_r, \theta_r) = \sqrt{x_r^2 + y_r^2} - R$

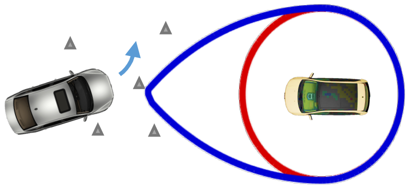
- Why is this correct?
  - Final state $\mathbf{x}(0)$ is in $\mathcal{T}$ if and only if $h\big(\mathbf{x}(0)\big) \leq 0$
  - To avoid $\mathcal{T}$, control should maximize $h\big(\mathbf{x}(0)\big)$
    - Worst-case disturbance would minimize
  - $J(\mathbf{x}, t) = \min_{\Gamma[\mathbf{u}]} \max_{\mathbf{u}} h\big(\mathbf{x}(0)\big)$



$J\big(\mathbf{x_g}(0), 0\big) > 0$
$J\big(\mathbf{x}_b(0), 0\big) \leq 0$

Reaching vs. Avoiding

- Avoiding danger



  - BRS definition

$$\mathcal{A}(t) = \{\bar{\mathbf{x}}: \exists \Gamma[\mathbf{u}](\cdot), \forall \mathbf{u}(\cdot), \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{d}), \mathbf{x}(t) = \bar{\mathbf{x}}, \mathbf{x}(0) \in \mathcal{T}\}$$

  - Value function

$$J(\mathbf{x}, t) = \min_{\Gamma[\mathbf{u}]} \max_{\mathbf{u}} h\big(\mathbf{x}(0)\big)$$

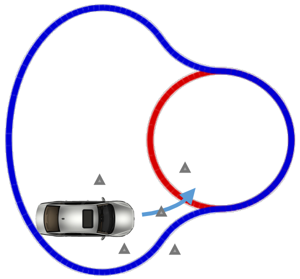
  - HJI

$$\frac{\partial J}{\partial t} + \max_{\mathbf{u}} \min_{\mathbf{d}} \left[\left(\frac{\partial J}{\partial \mathbf{x}} \right)' f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right] = 0$$

  - Optimal control

$$\mathbf{u}^* = \arg \max_{\mathbf{u}} \min_{\mathbf{d}} \left(\frac{\partial J}{\partial \mathbf{x}} \right)' f(\mathbf{x}, \mathbf{u}, \mathbf{d})$$

- Reaching a goal



  - BRS definition

$$\mathcal{R}(t) = \{\bar{\mathbf{x}}: \forall \Gamma[\mathbf{u}](\cdot), \exists \mathbf{u}(\cdot), \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{d}), \mathbf{x}(t) = \bar{\mathbf{x}}, \mathbf{x}(0) \in \mathcal{T}\}$$

  - Value function

$$J(\mathbf{x}, t) = \max_{\Gamma[\mathbf{u}]} \min_{\mathbf{u}} h\big(\mathbf{x}(0)\big)$$

  - HJI

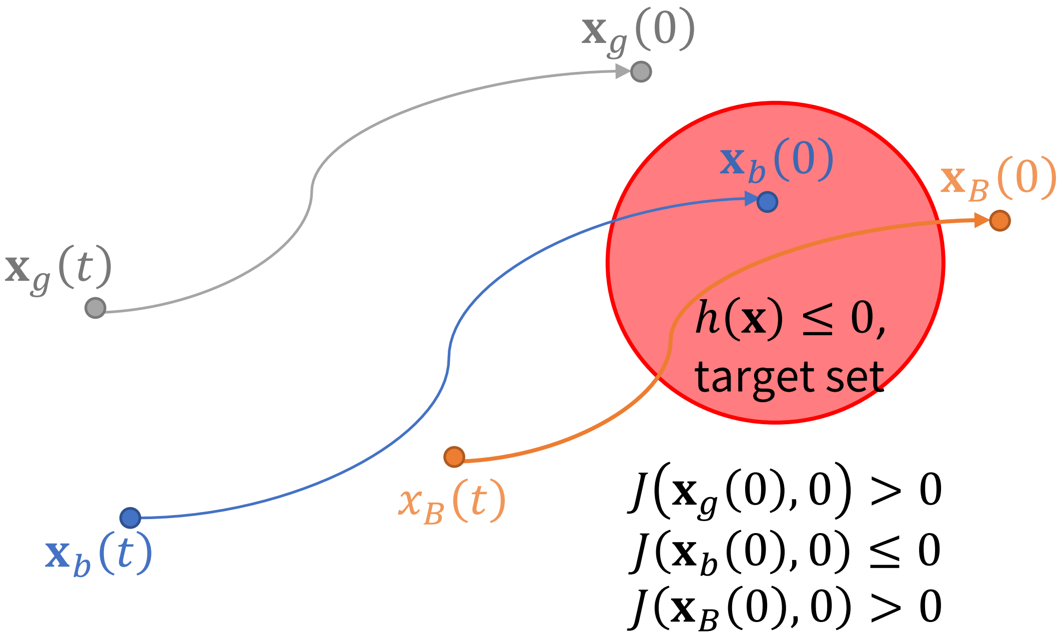
$$\frac{\partial J}{\partial t} + \min_{\mathbf{u}} \max_{\mathbf{d}} \left[\left(\frac{\partial J}{\partial \mathbf{x}} \right)' f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right] = 0$$

  - Optimal control

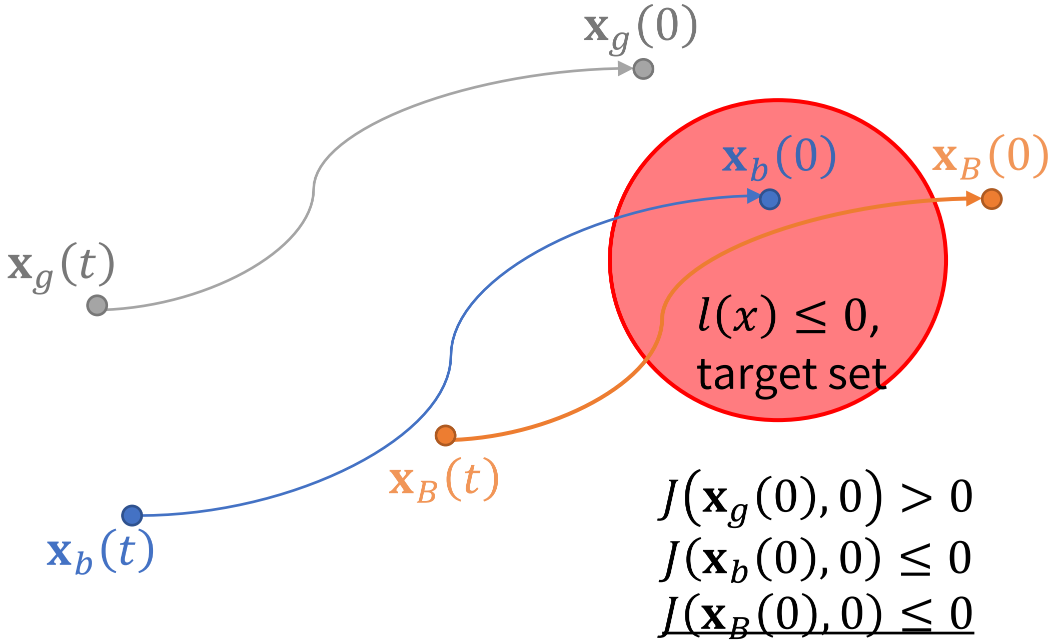
$$\mathbf{u}^* = \arg \min_{\mathbf{u}} \max_{\mathbf{d}} \left(\frac{\partial J}{\partial \mathbf{x}} \right)' f(\mathbf{x}, \mathbf{u}, \mathbf{d})$$

“Sets” vs. “Tubes”

- Backward reachable set (BRS)
  - Only final time matters
  - Initial states that passing through target are not necessarily in BRS
  - Not ideal for safety

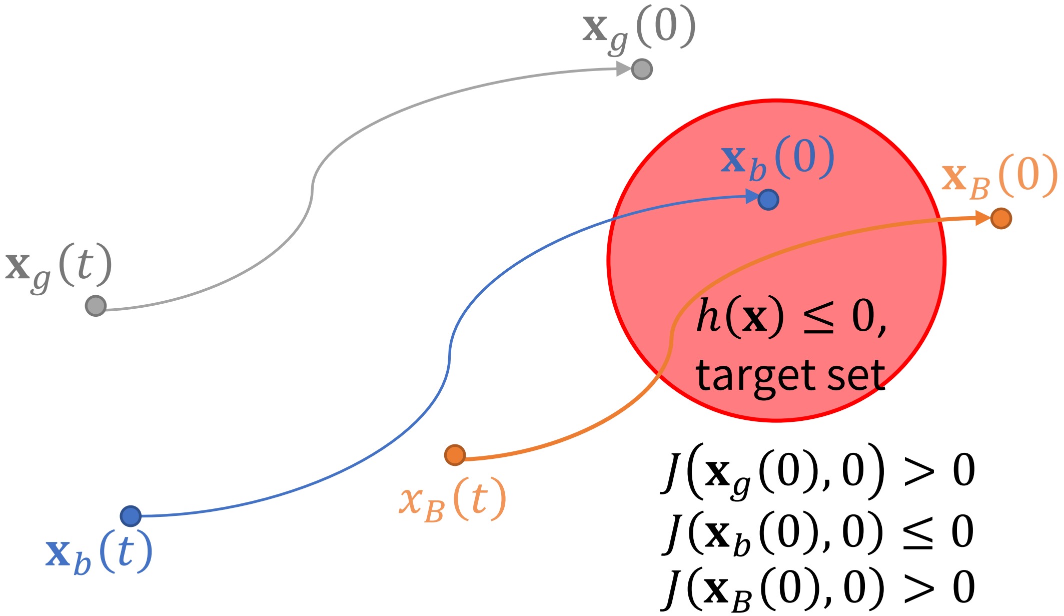


- Backward reachable tube (BRT)
  - Keep track of entire time duration
  - Initial states that pass through target are in BRT
  - Used to make safety guarantees



“Sets” vs. “Tubes”

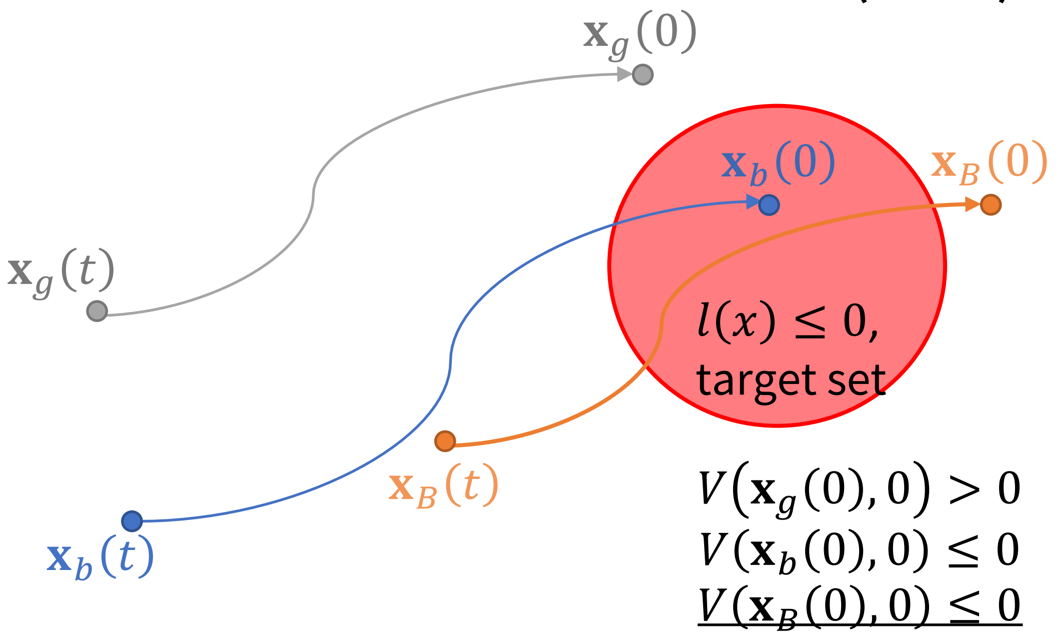
- Backward reachable set (BRS)



- Value function definition
  - $J(\mathbf{x}, t) = \min_{\Gamma[\mathbf{u}]} \max_{\mathbf{u}} h\big(\mathbf{x}(0)\big)$
- Value function obtained from

$$\frac{\partial J}{\partial t} + \max_{\mathbf{u}} \min_{\mathbf{d}} \left[\left(\frac{\partial J}{\partial \mathbf{x}} \right)' f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right] = 0$$

- Backward reachable tube (BRT)



- Value function definition
  - $J(\mathbf{x}, t) = \min_{\Gamma[\mathbf{u}]} \max_{\mathbf{u}} \min_{\tau \in [t, 0]} h\big(\mathbf{x}(\tau)\big)$
- Value function obtained from

$$\min \left\{ \frac{\partial J}{\partial t} + \max_{\mathbf{u}} \min_{\mathbf{d}} \left[\left(\frac{\partial J}{\partial \mathbf{x}} \right)^{\top} f(\mathbf{x}, \mathbf{u}, \mathbf{d}) \right], h(\mathbf{x}) - J(\mathbf{x}, t) \right\} = 0$$

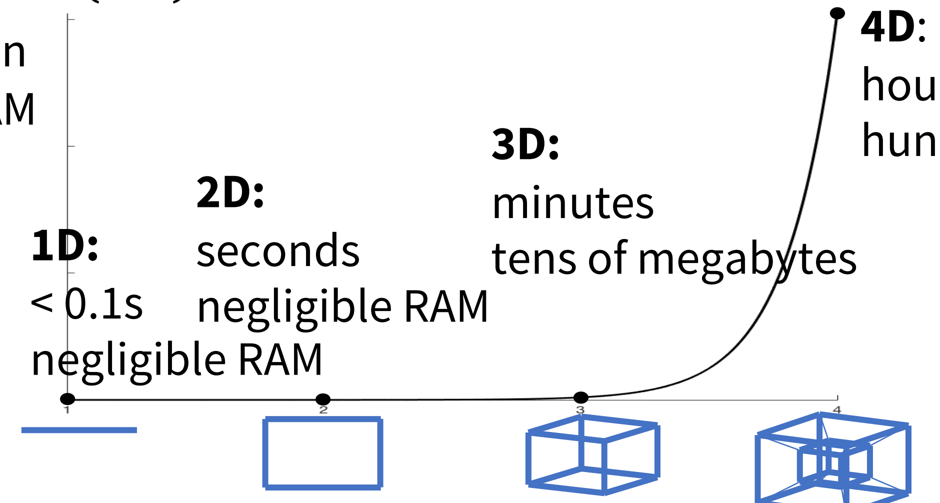
Computational aspects

- Computational complexity
  - $J(\mathbf{x}, t)$ is computed on an $(n+1)$-dimensional grid
  - Currently, $n \leq 5$ is possible. GPU acceleration under-way
  - Dimensionality reduction methods sometimes help

- Related approaches
  - Sacrifice global optimality
  - Give up guarantees

  - Sampling-based methods
  - Reinforcement learning

$O(N^d)$ time and space complexity!



Numerical toolboxes

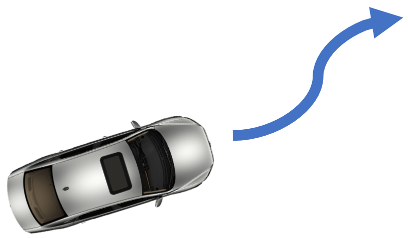
- helperOC Matlab toolbox
  - https://github.com/HJReachability/helperOC.git
  - Reachability wrapper around the level set toolbox
  - Requires level set toolbox
    - Hamilton-Jacobi PDE solver by Ian Mitchell, UBC
    - https://bitbucket.org/ian_mitchell/toolboxls

- C++ and CUDA version in development, beta also available
  - C++: 5+ times faster than Matlab
  - CUDA: Up to 100 times faster than Matlab
  - https://github.com/HJReachability/beacls

Example – waypoint reaching with Dubins Car

Dubins Car Model

$$\begin{cases} \dot{x} = v\cos\theta + \boldsymbol{d_x} \\ \dot{y} = v\sin\theta \\ \dot{\theta} = k\,\boldsymbol{u} \end{cases}$$



Control: $\boldsymbol{u}$

Disturbance: $\boldsymbol{d_x}$

- Target set:

$$\mathcal{T} = \{(x,y,\theta) \in \mathbb{R}^3 : h(x,y,\theta) := ((x,y,\theta) - (x_{max}, y_{max}, \theta_{max}), (x_{min}, y_{min}, \theta_{min}) - (x,y,\theta)) \leq 0\}$$

- HJI equation:

$$\frac{\partial J}{\partial t}(x,y,\theta,t) + \min_{|u|\leq u_{max}} \max_{|d_x|\leq d^x_{max}} \nabla J(x,y,\theta,t)' f(x,y,\theta,u,d) = 0$$

- Optimal quantities:

$$u^*(x,y,\theta,t) = \arg\min_{|u|\leq u_{max}} \max_{|d_x|\leq d^x_{max}} \nabla J(x,y,\theta,t)^\top f(x,y,\theta,u,d)$$

$$d^*(x,y,\theta,t) = \arg\max_{|d_x|\leq d^x_{max}} \nabla J(x,y,\theta,t)^\top f(x,y,\theta,u^*,d)$$

$$\Rightarrow$$

$$u = -u_{max}\,\mathrm{sign}\left(\frac{\partial J}{\partial \theta}\right)$$

$$d_x = d^x_{max}\,\mathrm{sign}\left(\frac{\partial J}{\partial x}\right)$$

Next time

- Calculus of variations