Reachability: An Application of the Time-Dependent Hamilton-Jacobi Equation

Ian Mitchell
Department of Computer Science
The University of British Columbia

research supported by
National Science and Engineering Research Council of Canada
DARPA Software Enabled Control Project

Hamilton-Jacobi Flavours

- Stationary (static/time-independent) Hamilton-Jacobi used for target based cost to go and time to reach problems

$$H(x, D_x \vartheta(x)) = 0 \qquad \|\nabla \vartheta(x)\| = c(x)$$

  - PDE coupled to boundary conditions
  - Solution may be discontinuous
- Time-dependent Hamilton-Jacobi used for dynamic implicit surfaces and finite horizon optimal control / differential games

$$D_t \phi(x,t) + H(x, D_x \phi(x,t)) = 0$$

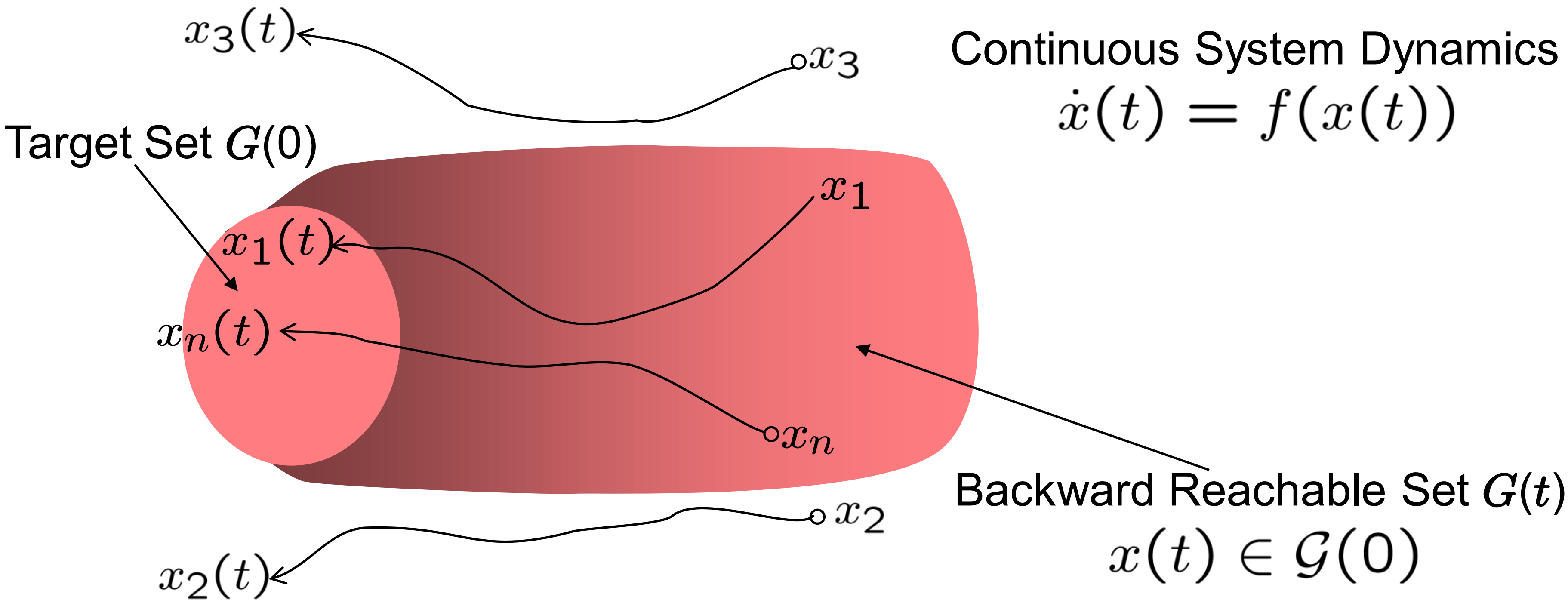
  - PDE coupled to initial/terminal and possibly boundary conditions
  - Solution continuous but not necessarily differentiable
- Other versions exist
  - Discounted and/or infinite horizon

Contents (not strictly ordered)

- Backward reach sets & tubes
  - Treatment of inputs
- Formulation as finite horizon optimal control
  - Implicit surface functions
  - Modification for optimal stopping
- Game of two identical vehicles
  - HJ PDE calculation
  - Analytic solution (almost)
  - Synthesis of safe controls
- Reducing the dimension
  - Systems with terminal integrators
  - Mixed implicit explicit representation
  - Target application: safety for the quadrotor flip

Continuous Backward Reach Tubes

- Set of all states from which trajectories can reach some given target state
  - For example, what states can reach $G(0)$?

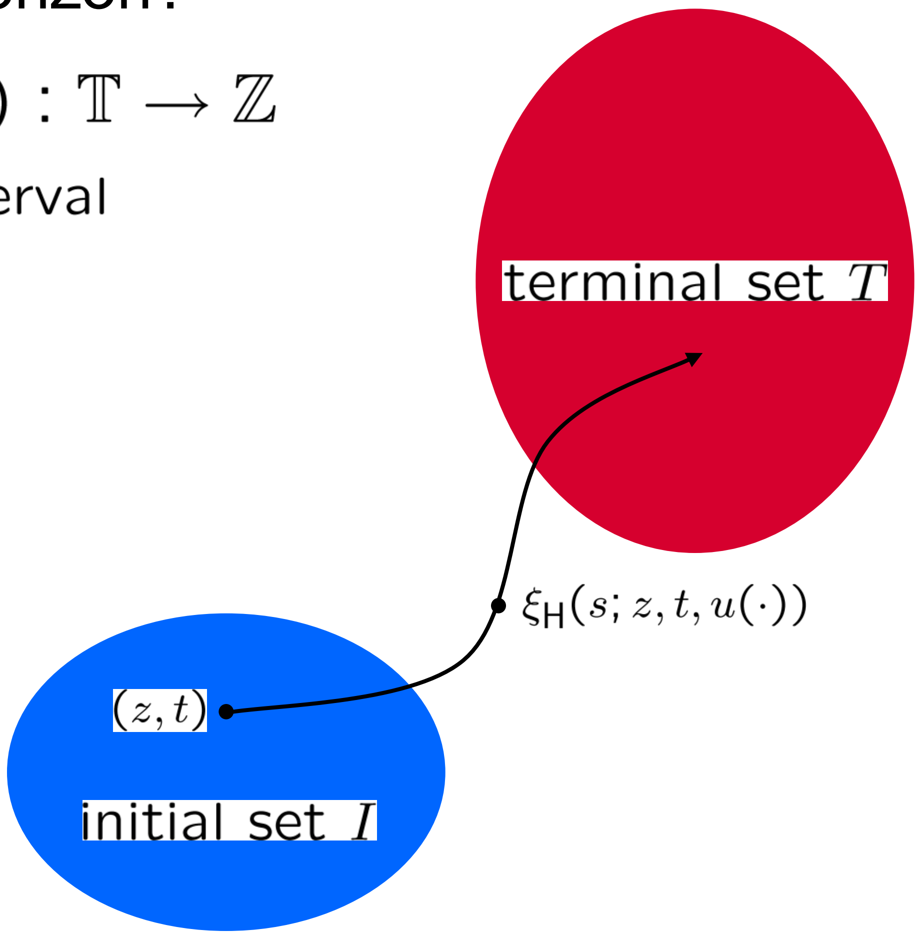


Verification: Safety Analysis

- Does there exist a trajectory of system H leading from a state in initial set I to a state in terminal set T during some finite time horizon?

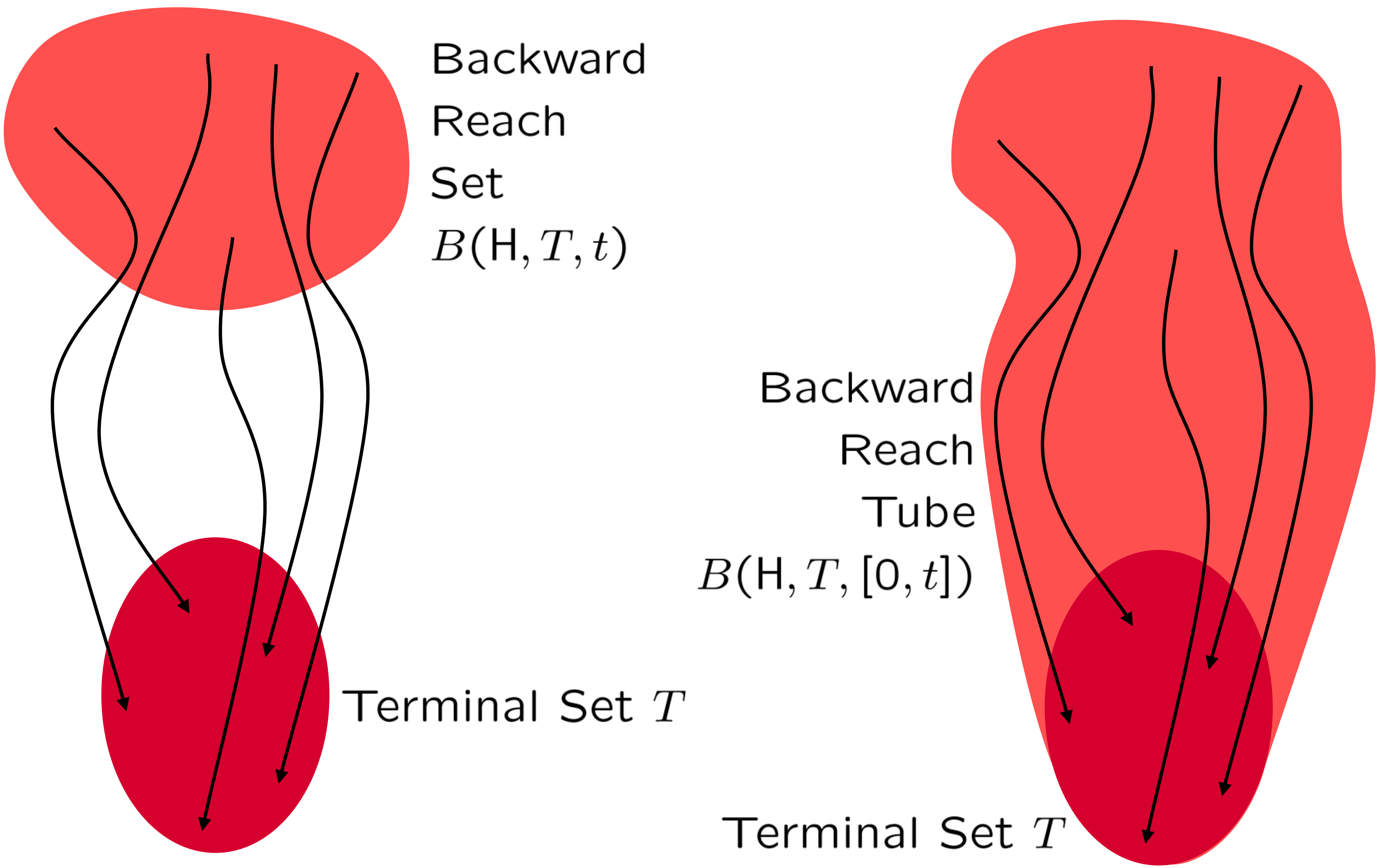
Trajectory $\xi_{\mathsf{H}}(s; z, t, u(\cdot)) : \mathbb{T} \to \mathbb{Z}$

- $\mathbb{T} = [-\mathcal{T}, +\mathcal{T}]$ is time interval
- $\mathbb{Z}$ is state space of H
- $s \in \mathbb{T}$ is current time
- $z \in \mathbb{Z}$ is initial state
- $t \in \mathbb{T}$ is initial time
- $u(\cdot) \in \mathbb{U}$ is input signal



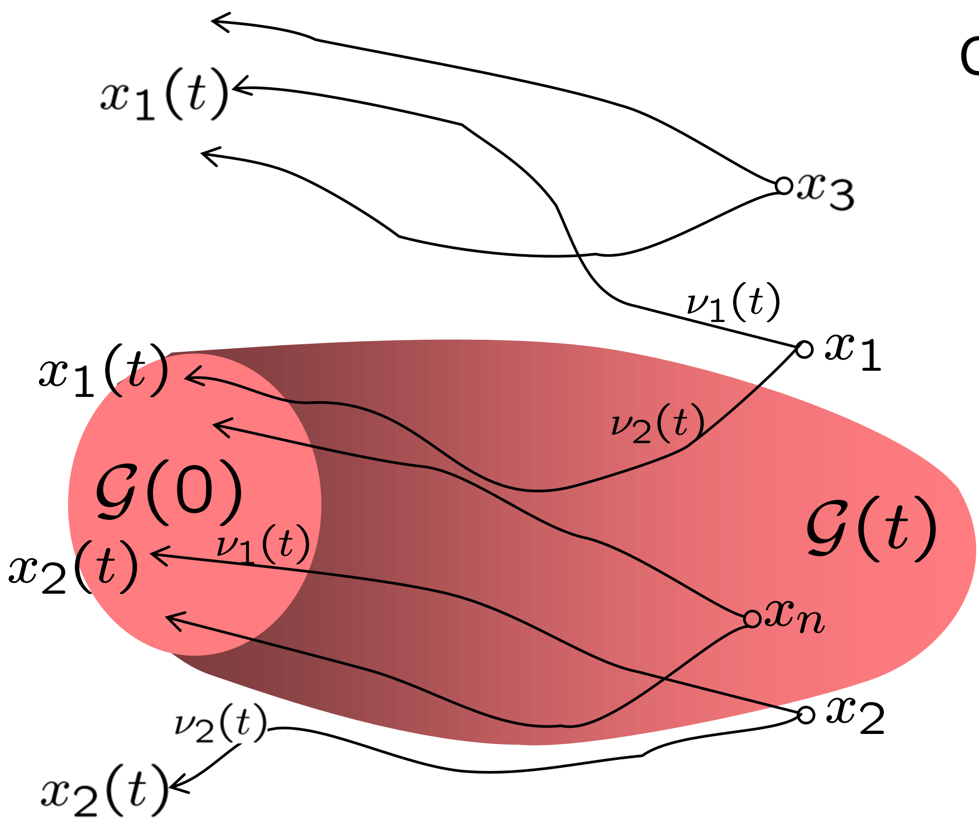
Reach Sets vs Reach Tubes

- Start at terminal set and compute backwards



Reach Tubes (controlled input)

- For most of our examples, target set is unsafe
- If we can control the input, choose it to avoid the target set
- Backward reachable set is unsafe no matter what we do
- "Minimal" backward reach tube



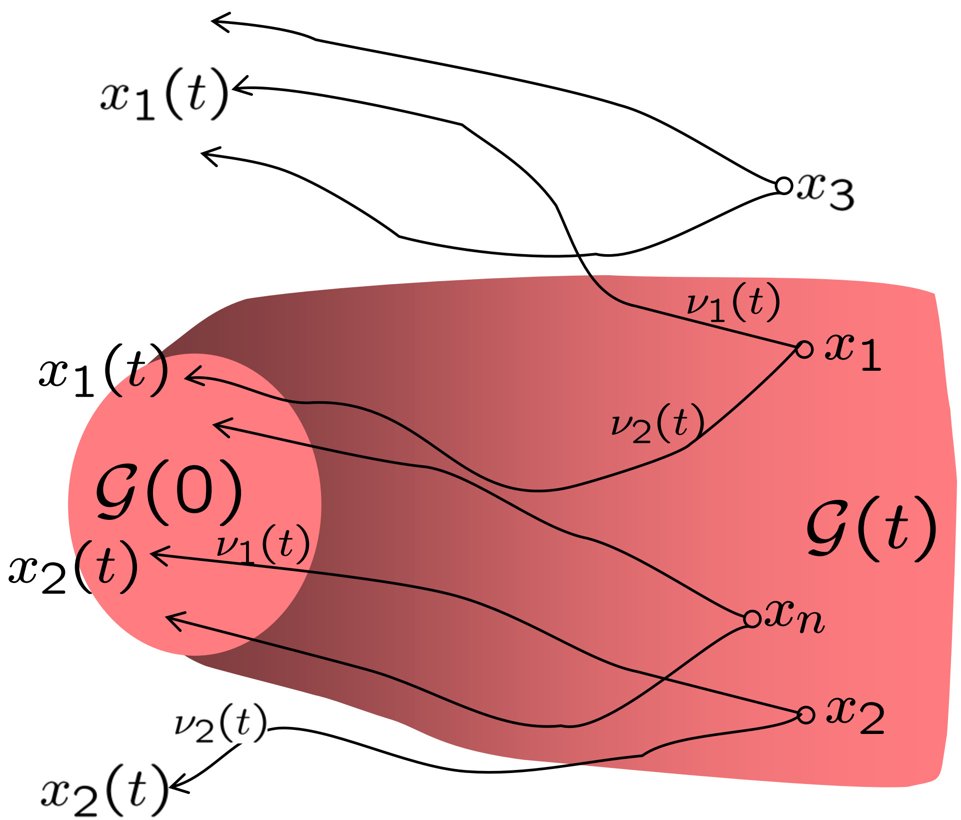
Continuous System Dynamics

$$\dot{x}(t) = f(x(t), \nu(t))$$

$$\forall \nu(\cdot), x(t) \in \mathcal{G}(0)$$

Reach Tubes (uncontrolled input)

- Sometimes we have no control over input signal
  - noise, actions of other agents, unknown system parameters
- It is safest to assume the worst case
- "Maximal" backward reach tube



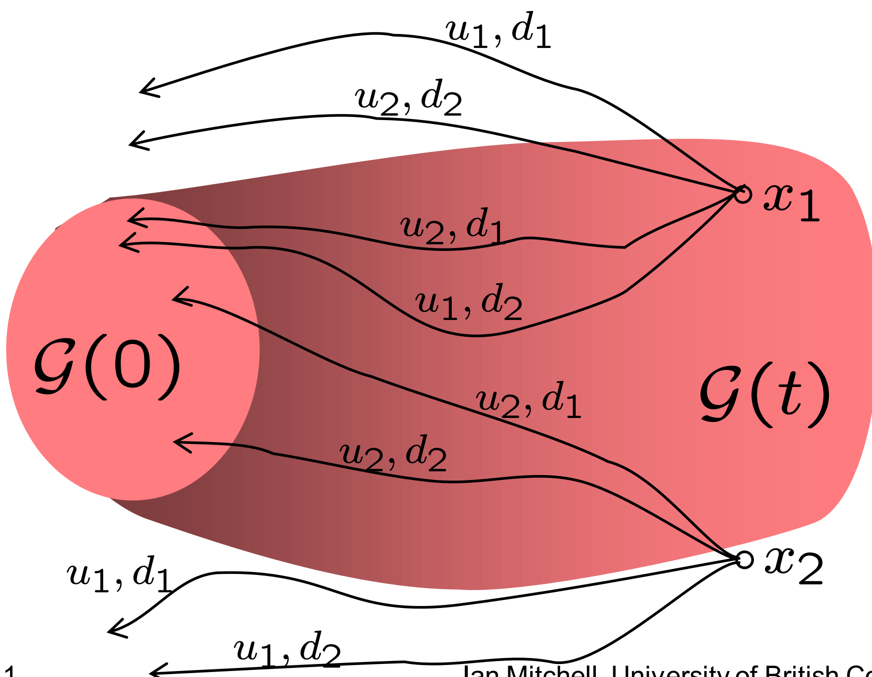
Continuous System Dynamics

$$\dot{x}(t) = f(x(t), \nu(t))$$

$$\exists \nu(\cdot), x(t) \in \mathcal{G}(0)$$

Two Competing Inputs

- For some systems there are two classes of inputs $\boldsymbol{\nu}$ = (u,v)
  - Controllable inputs $u \in U$
  - Uncontrollable (disturbance) inputs $v \in V$
- Equivalent to a zero sum differential game formulation
  - If there is an advantage to input ordering, give it to disturbances



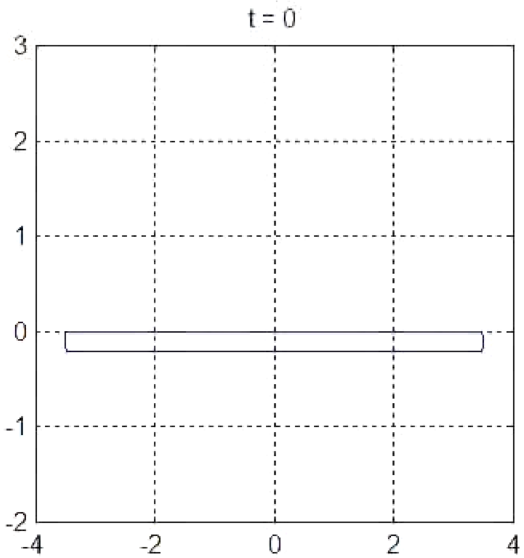
Continuous System Dynamics

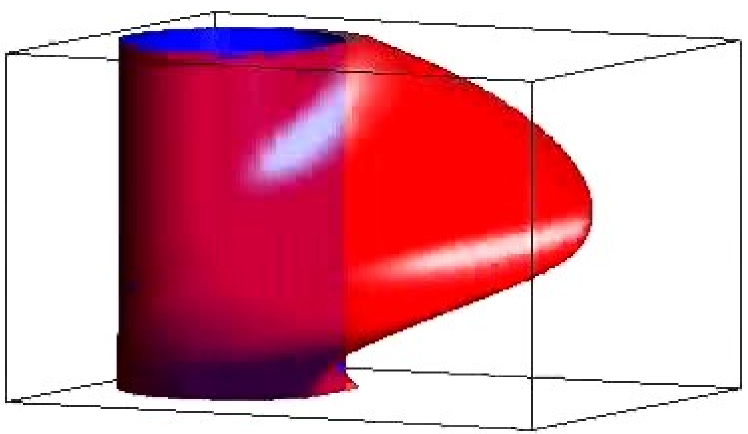
$$\dot{x}(t) = f(x(t), u(t), v(t))$$

$$\forall u(\cdot), \exists v(\cdot), x(t) \in \mathcal{G}(0)$$

Calculating Reach Sets & Tubes

- Two primary challenges
  - How to represent set of reachable states
  - How to evolve set according to dynamics
- Discrete systems $x_{k+1} = \delta(x_k)$
  - Enumerate trajectories and states
  - Efficient representations: Binary Decision Diagrams
- Continuous systems $dx/dt = f(x)$?

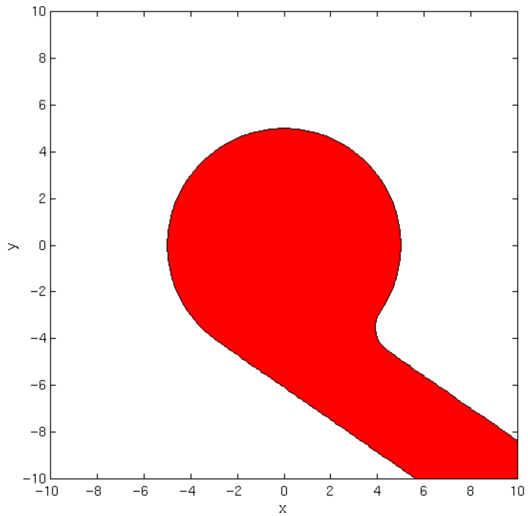


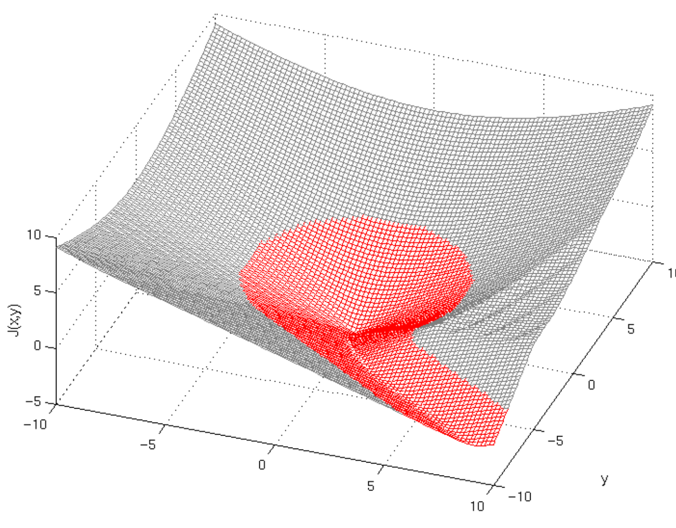


Implicit Surface Functions

- Set $G(t)$ is defined implicitly by an isosurface of a scalar function $\phi(x,t)$, with several benefits
  - State space dimension does not matter conceptually
  - Surfaces automatically merge and/or separate
  - Geometric quantities are easy to calculate
- Set must have an interior
  - Examples (and counter-examples) shown on board

$$\phi : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R} \qquad G(t) = \{x \in \mathbb{R}^n \mid \phi(x,t) \leq 0\}$$





Reach Set as Optimal Control

- Represent the target set as an implicit surface function

$$T = G(0) = \{x \mid h(x) \leq 0\}$$

- Solve an optimal control problem with target set implicit surface function as the terminal cost, and zero running cost

$$\begin{aligned}
\text{initial conditions:} \quad & \xi_f\left(t; x, t, a(\cdot), b(\cdot)\right) = x \\
\text{dynamics:} \quad & \dot{\xi}_f\left(s; x, t, a(\cdot), b(\cdot)\right) = f\left(x, a(s), b(s)\right) \\
\text{running cost:} \quad & g(x, a, b) = 0 \\
\text{terminal cost:} \quad & g_f(x) = h(x)
\end{aligned}$$

- Resulting value function is an implicit surface function for the backward reach set

$$V(x,t) = \phi(x,t) = \inf_{\gamma[a(\cdot)](\cdot)} \sup_{a(\cdot)} \left(h\left[\xi_f\left(0; x, t, a(\cdot), \gamma[a(\cdot)](\cdot)\right) \right] \right)$$

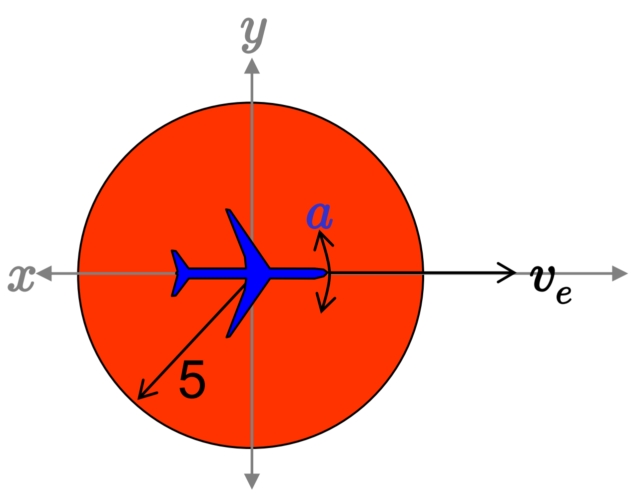
$$G(t) = \{x \mid \phi(x,t) \leq 0\}$$

Game of Two Identical Vehicles

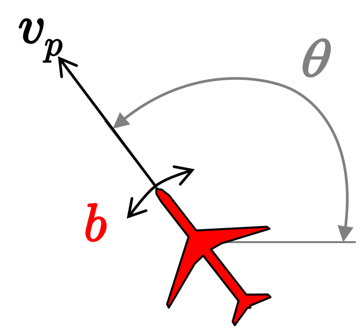
- Classical collision avoidance example
  - Collision occurs if vehicles get within five units of one another
  - Evader chooses turn rate $|a| \leq 1$ to avoid collision
  - Pursuer chooses turn rate $|b| \leq 1$ to cause collision
  - Fixed equal velocity $v_e = v_p = 5$

dynamics (pursuer)

$$\frac{d}{dt}\begin{bmatrix} x_p \\ y_p \\ \theta_p \end{bmatrix} = \begin{bmatrix} v_p \cos\theta_p \\ v_p \sin\theta_p \\ b \end{bmatrix}$$



evader aircraft (control)

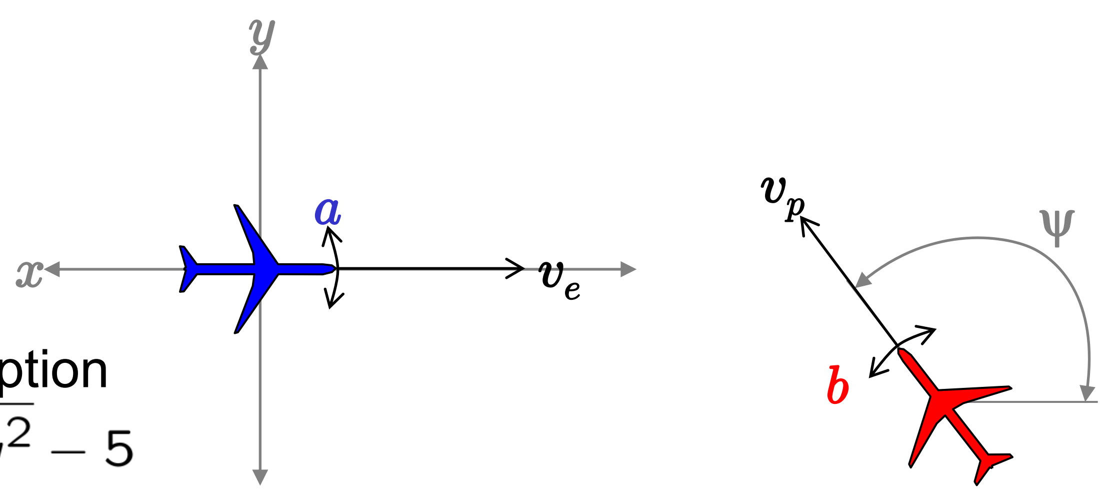


pursuer aircraft (disturbance)

Collision Avoidance Computation

- Work in relative coordinates with evader fixed at origin
  - State variables are now relative planar location (x,y) and relative heading ψ

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ \psi \end{bmatrix} = \begin{bmatrix} -v_e + v_p \cos\psi - ay \\ v_p \sin\psi - ax \\ b - a \end{bmatrix} = f(z, a, b)$$



target set description

$$h(x) = \sqrt{x^2 + y^2} - 5$$

evader aircraft (control)

pursuer aircraft (disturbance)

Evolving Reachable Sets

- Modified Hamilton-Jacobi partial differential equation

$$D_t\phi(z,t) + \min\left[0, H(z, D_z\phi(z,t))\right] = 0$$

$$\text{with Hamiltonian}: \quad H(z,p) = \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}} f(z,a,b)\cdot p$$

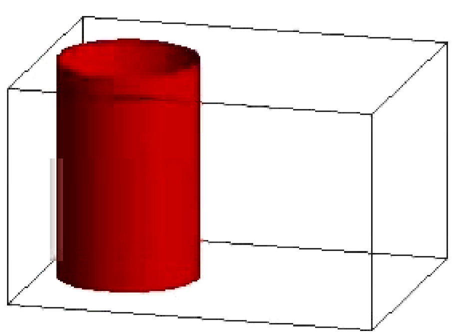
$$\text{and terminal conditions}: \quad \phi(z,0) = h(z)$$

$$\text{where} \quad G(0) = \{z \in \mathbb{R}^n \mid h(z) \le 0\}$$

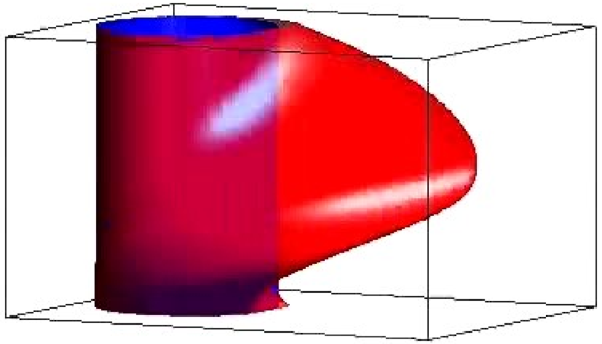
$$\text{and} \quad \dot{z} = f(z,a,b)$$

$$\text{yields} \quad G(t) = \{z \in \mathbb{R}^n \mid \phi(z,-t) \le 0\}$$

growth of reachable set



final reachable set



Solving a Differential Game

- Terminal cost differential game for trajectories $\xi_f(\cdot;\, x, t, a(\cdot), b(\cdot))$

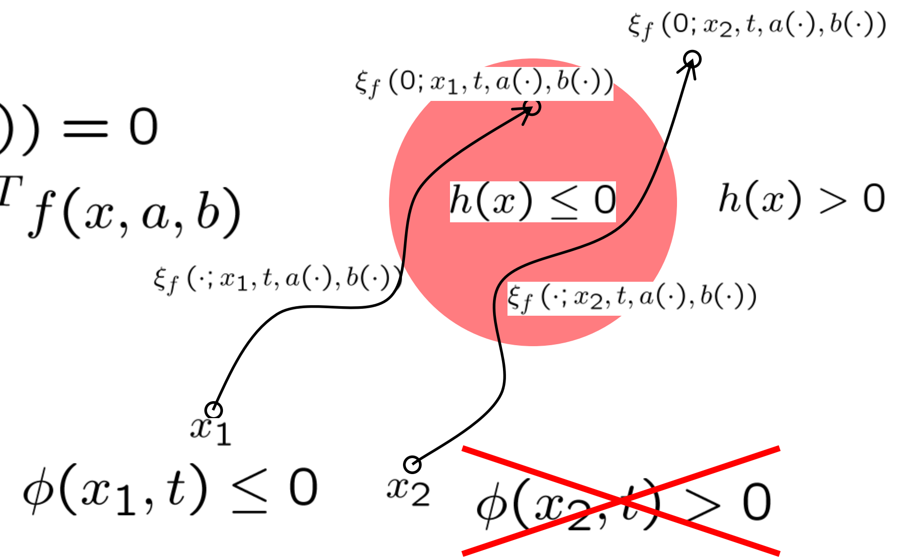
$$\phi(x,t) = \inf_{\gamma[a(\cdot)](\cdot)} \sup_{a(\cdot)} h\left[\xi_f\left(0; x, t, a(\cdot), \gamma[a(\cdot)](\cdot)\right)\right]$$

$$\text{where} \begin{cases} \xi_f\left(t; x, t, a(\cdot), b(\cdot)\right) = x \\ \dot{\xi}_f((s; x, t, a(\cdot), b(\cdot)) = f\left(x, a(s), b(s)\right) \\ \text{terminal payoff function } h(x) \end{cases}$$

- Value function solution $\phi(x,t)$ given by viscosity solution to basic Hamilton-Jacobi equation
  - [Evans & Souganidis, 1984]

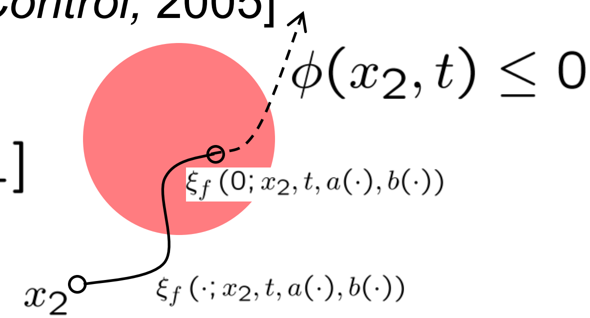
$$D_t\phi(x,t) + H\left(x, D_x\phi(x,t)\right) = 0$$

$$\text{where} \begin{cases} H(x,p) = \max_{a\in\mathcal{A}} \min_{b\in\mathcal{B}} p^T f(x,a,b) \\ \phi(x,0) = h(x) \end{cases}$$



Modification for Optimal Stopping Time

- How to keep trajectories from passing through $G(0)$?
  - [Mitchell, Bayen & Tomlin, *IEEE T. Automatic Control,* 2005]
  - Augment disturbance input



$$\tilde{b} = \begin{bmatrix} b & \underline{b} \end{bmatrix} \text{ where } \underline{b} : [t, 0] \rightarrow [0, 1]$$

$$\tilde{f}(x, a, \tilde{b}) = \underline{b} f(x, a, b)$$

  - Augmented Hamilton-Jacobi equation solves for reachable set

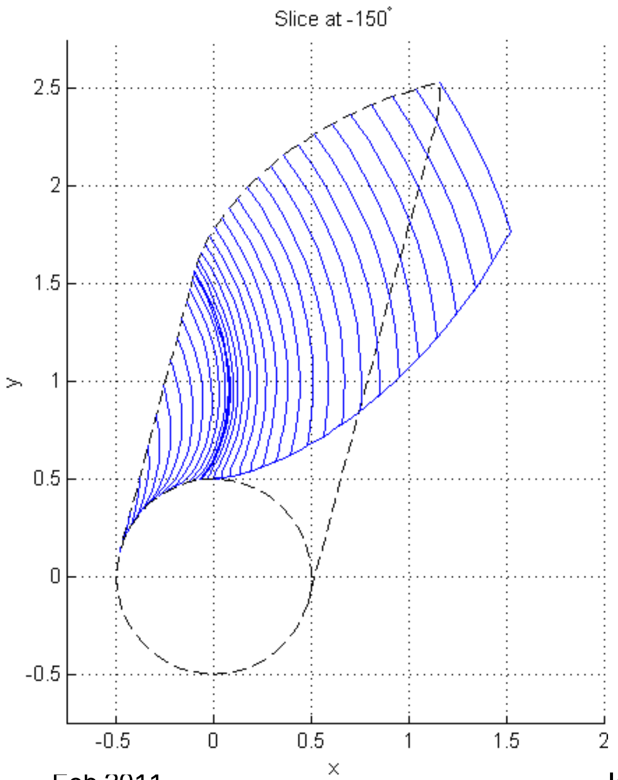
$$D_t \phi(x, t) + \tilde{H}\left(x, D_x \phi(x, t)\right) = 0 \text{ where } \begin{cases} \tilde{H}(x, p) = \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} p^T \tilde{f}(x, a, \tilde{b}) \\ \phi(x, 0) = h(x) \end{cases}$$

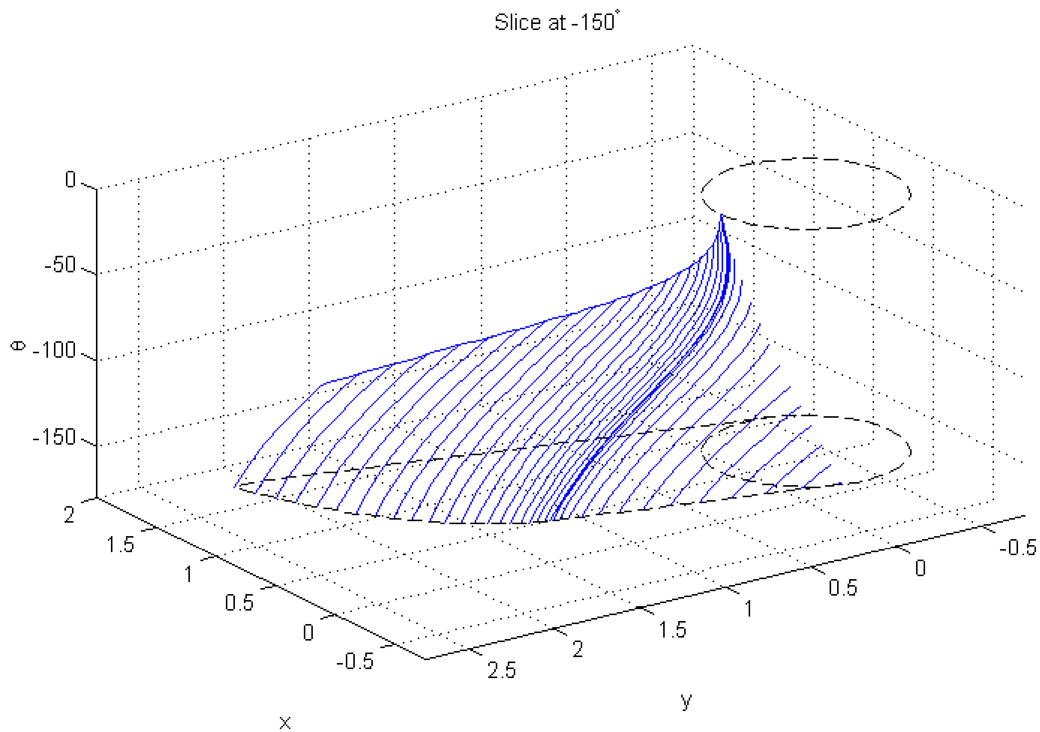
  - Augmented Hamiltonian is equivalent to modified Hamiltonian

$$\begin{aligned} \tilde{H}(x, p) &= \max_{a \in \mathcal{A}} \min_{\tilde{b} \in \tilde{\mathcal{B}}} p^T \tilde{f}(x, a, \tilde{b}) \\ &= \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} \min_{\underline{b} \in [0,1]} \underline{b} p^T f(x, a, b) \\ &= \min \left[0, \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} p^T f(x, a, b) \right] = \min \left[0, H(x, p) \right] \end{aligned}$$

Analytic Solution

- A. W. Merz (1971) solved differential game of two identical cars
  - Optimal inputs and trajectories
  - Slight modification yield analytic solution to collision avoidance
- Result: analytic collection of points lying on the interface
  - Mitchell, “Games of Two Identical Vehicles”, SUDAAR 740, 2001

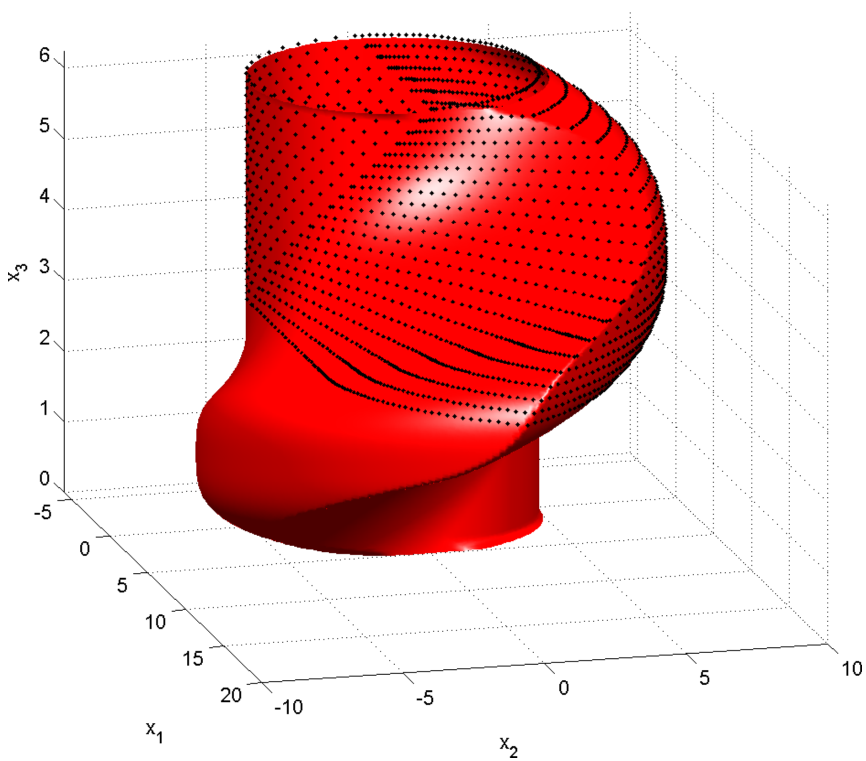


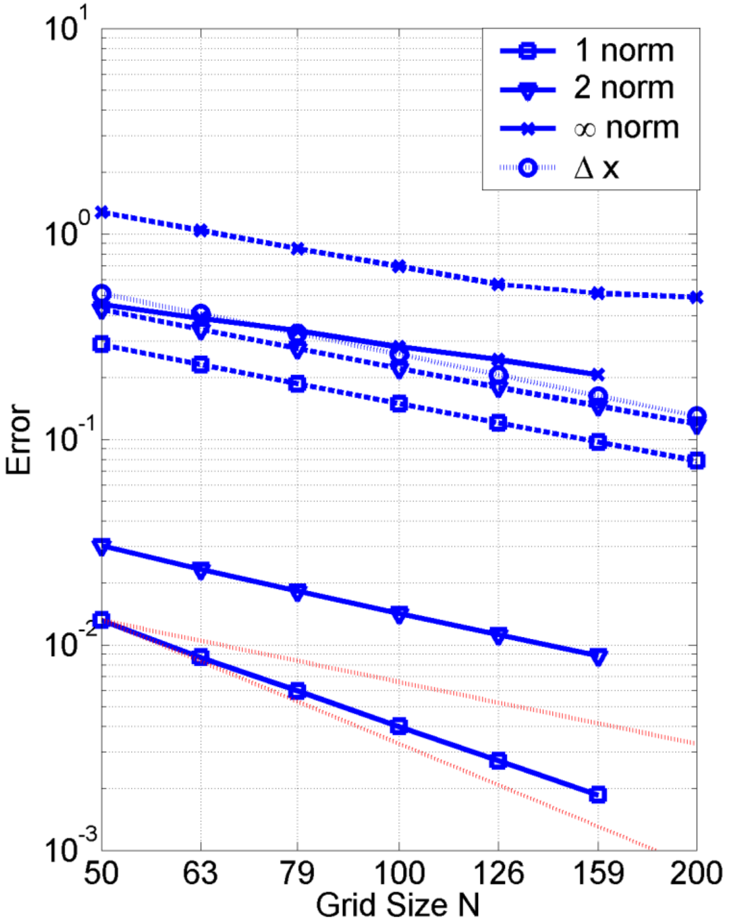


Validating the Numerical Algorithm

- Analytic solution $\{x_i\}$ validates interface location

error: $$\sum_{i=1}^{N} |\phi(x_i)|$$





Application: Synthesizing Safe Controllers

- By construction, on the boundary of the unsafe set there exists a control to keep trajectories safe
  - Filter potentially unsafe controls to ensure safety

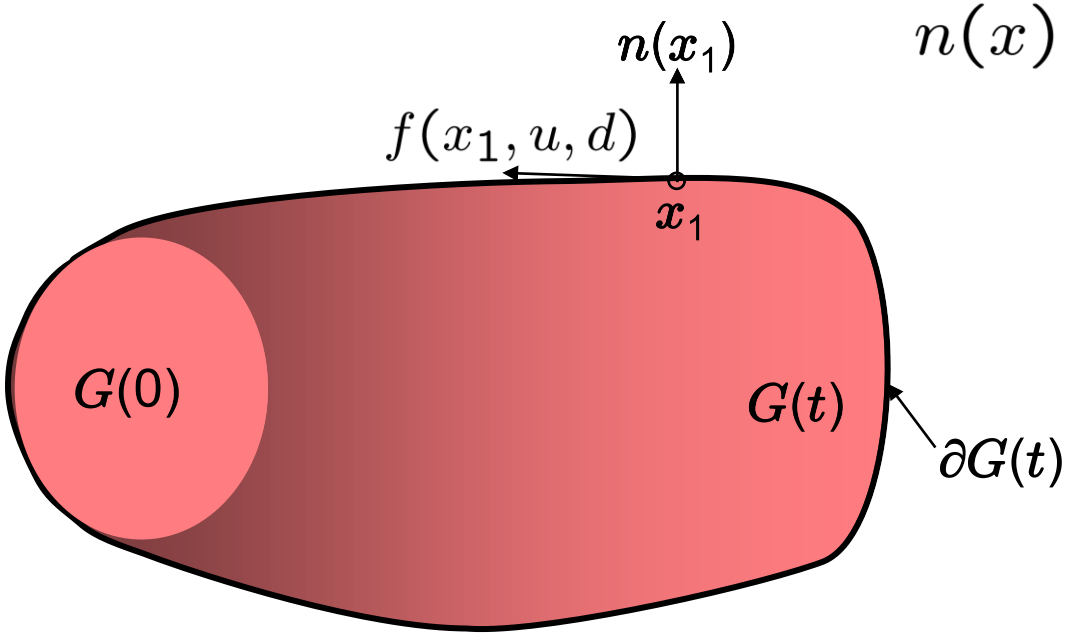
continuous system dynamics

$$\dot{x} = f(x, u, d)$$

by construction

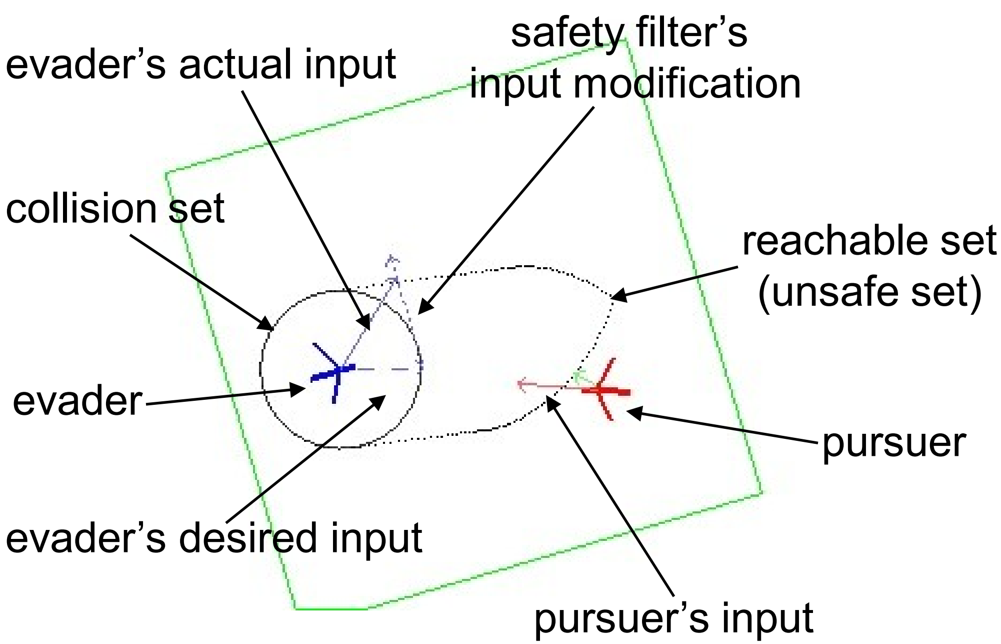
$$\forall x \in \partial G(t), \exists u \in \mathcal{U}, \forall v \in \mathcal{V}$$

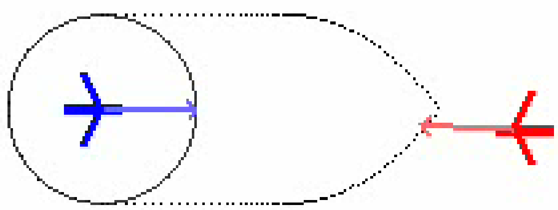
$$n(x) \cdot f(x, u, v) \geq 0$$



Synthesizing Safe Controls (No Safety)

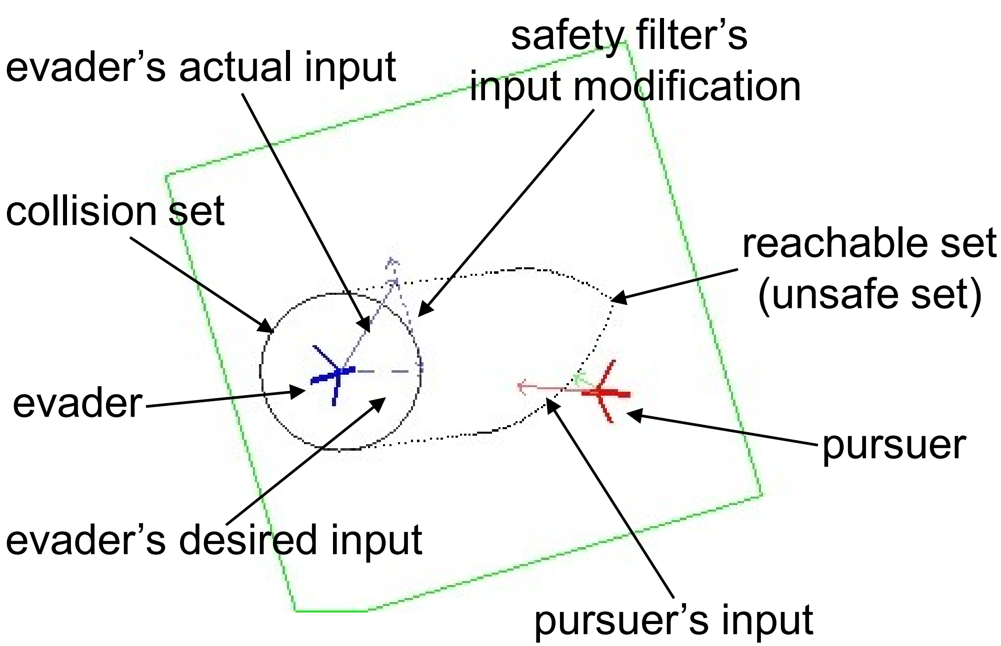
- Use reachable sets to guarantee safety
- Basic Rules
  - Pursuer: turn to head toward evader
  - Evader: turn to head east
- No filtering of evader input

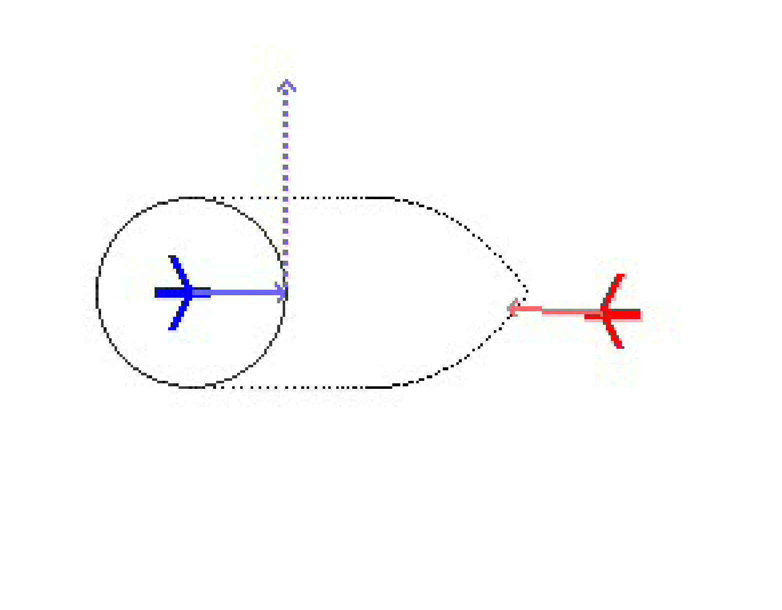




Synthesizing Safe Controls (Success)

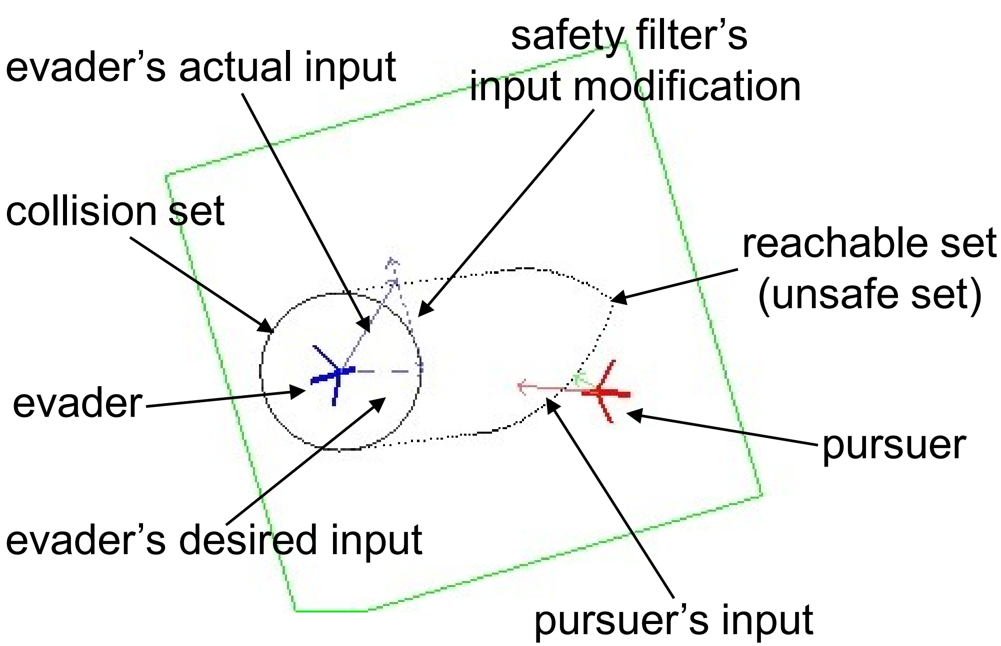
- Use reachable sets to guarantee safety
- Basic Rules
  - Pursuer: turn to head toward evader
  - Evader: turn to head east
- Evader's input is filtered to guarantee that pursuer does not enter the reachable set

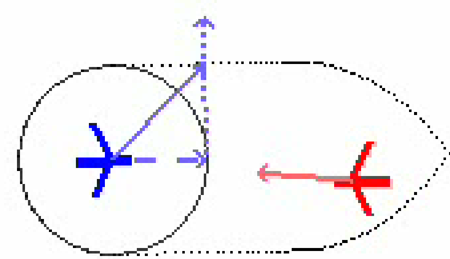




Synthesizing Safe Controls (Failure)

- Use reachable sets to guarantee safety
- Basic Rules
  - Pursuer: turn to head toward evader
  - Evader: turn to head east
- Evader's input is filtered, but pursuer is already inside reachable set, so collision cannot be avoided





Acoustic Capture

- Modified version of homicidal chauffeur from [Cardaliaguet, Quincampoix & Saint-Pierre, 1999]
  - Pursuer is faster with limited turn radius but fast rotation
  - Evader can move any direction, but speed is lowered near pursuer
- Also solved in relative coordinates

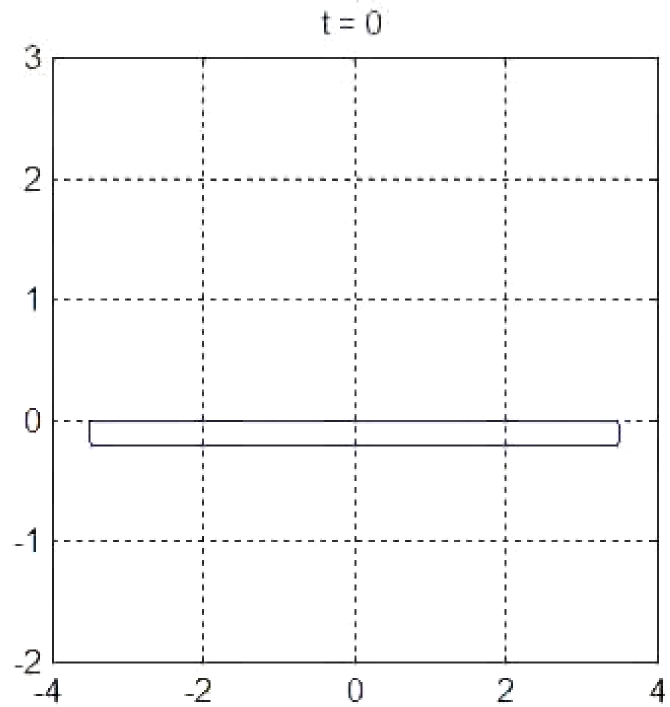
continuous system dynamics

$$\frac{d}{dt}\begin{bmatrix} x \\ y \end{bmatrix} = W_p \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \frac{W_p}{R}\begin{bmatrix} y \\ -x \end{bmatrix} b + 2W_e \min\left(\sqrt{x^2+y^2}, S\right) a$$

$$a \in \mathbb{R}^2, \|a\| \leq 1$$

$$b \in [-1, +1]$$

$$W_p, W_e, R, S \text{ constant}$$



Systems with Terminal Integrators

- Common form of system dynamics

$$\dot{y} = f(y,u) \quad \text{coupled states } y \in \mathbb{R}^{d_y},$$
$$\dot{x}_i = b(y) \quad \text{terminal integrator } x_i \in \mathbb{R}$$
$$\text{for } i = 1, \ldots, d_x$$

- Computational cost of reachability for full system with n grid points is $\mathcal{O}\left(n^{(d_y+d_x)}\right)$
- Instead
  - Run two modified HJ PDEs on $\mathbb{R}^{d_y}$ for each of the x_i variables
  - States are inside overall reach set only if inside every PDE's reach set
  - Computational cost $\mathcal{O}\left(2d_x n^{d_y}\right)$
- Mitchell, "Scalable Calculation of Reach Sets and Tubes for Partially Nonlinear Systems with Terminal Integrators," HSCC 2011

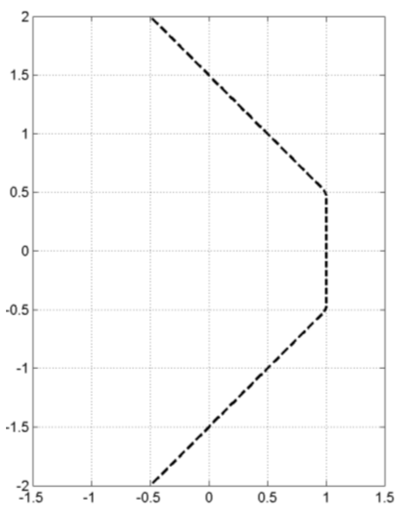
Mixed Implicit Explict Formulation

- Traditional *implicit* formulation represents sets with an implicit surface function

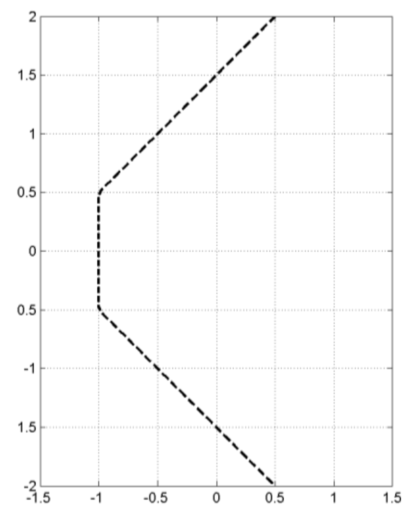
$$S = \{(x, y) \mid \psi(x, y) \leq 0\}$$

- New *mixed implicit explicit* (MIE) formulation represents sets as an interval in x_i for every i and y

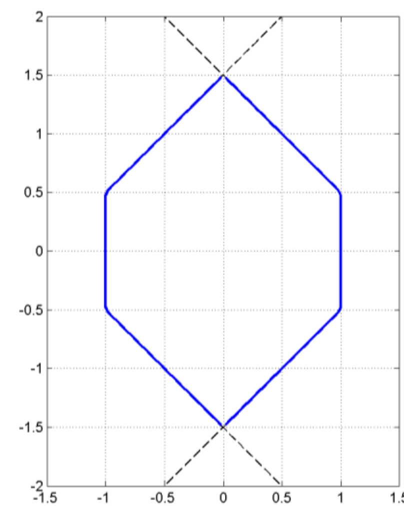
$$S = \left\{(x, y) \,\middle|\, \underline{\psi}_i(y) \leq x_i \leq \overline{\psi}_i(y)\right\}$$



upper bound $x \leq \overline{\psi}_0(y)$



lower bound $x \leq \underline{\psi}_0(y)$



overall target set

Terminal Integrator's HJ PDEs

- For scalar terminal integrator $d_x = 1$ define target set

$$S = \left\{ (x, y) \,\middle|\, \underline{\psi}_0(y) \leq x \leq \overline{\psi}_0(y) \right\}$$

- If $x(t, y) = \overline{\psi}(t, y)$ is the upper boundary of the reach set, then formally

$$b(y) = \tfrac{d}{dt} x(t, y) = \tfrac{d}{dt} \overline{\psi}(t, y) = D_t \overline{\psi}(t, y) + D_y \overline{\psi}(t, y) \cdot f(y, u)$$

- Rearrange to find terminal value HJ PDE

$$D_t \overline{\psi}(t, y) + H\left(t, y, D_x \overline{\psi}(t, y)\right) = 0 \qquad \overline{\psi}(0, y) = \overline{\psi}_0(y)$$

$$\text{with } H(t, y, p) = \max_{u \in U} \left(p \cdot f(y, u) - b(y)\right)$$

- Repeat with $x(t, y) = \underline{\psi}(t, y)$ for lower boundary (with adjustment of optimizations)
- Yields backwards reach set

$$B(S, t) = \left\{ (x, y) \,\middle|\, \underline{\psi}(t, y) \leq x \leq \overline{\psi}(t, y) \right\}$$

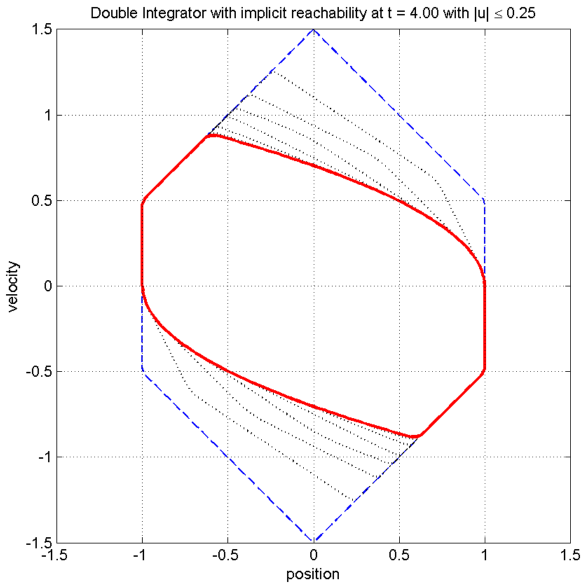
Double Integrator

- Dynamics

$$\dot{y} = f(y, u) = u \qquad \dot{x} = y \qquad |u| \leq u_{\text{max}}$$

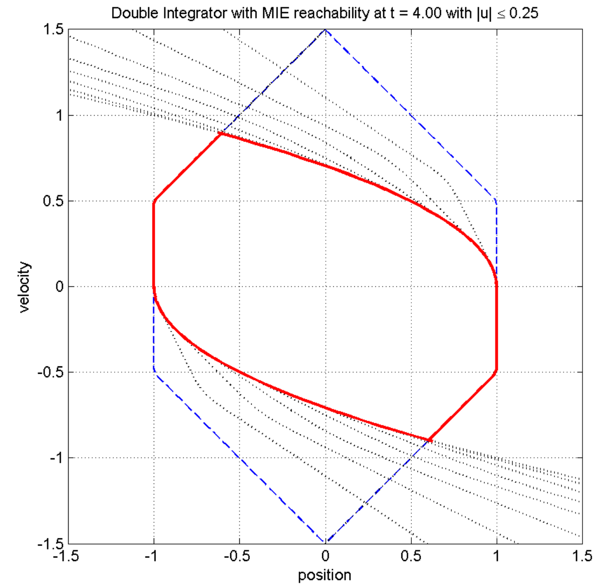
yields terminal integrator Hamiltonian (for upper bound)

$$H(t, y, r, p) = \max_{|u| \leq u_{\text{max}}} (p \cdot u - y) = (|p| u_{\text{max}} - y)$$



Regular implicit surface formulation

One HJ PDE in 2D



Terminal integrator formulation

Two HJ PDEs in 1D

Finite Horizon Optimal Control

- Terminal integrator's dynamics (for $t < 0$) are

$$x(0, y(0)) = x(t, y(t)) + \int_t^0 b(y(s))ds$$

or

$$x(t, y(t)) = \int_t^0 -b(y(s))ds + x(0, y(0))$$

- Can be interpreted as a finite horizon optimal control problem with associated HJ PDE

$$D_t\overline{\psi}(t, y) + H\left(t, y, D_x\overline{\psi}(t, y)\right) = 0 \qquad \overline{\psi}(0, y) = \overline{\psi}_0(y)$$

$$\text{with } H(t, y, p) = \max_{u \in U}\left(p \cdot f(y, u) - b(y)\right)$$

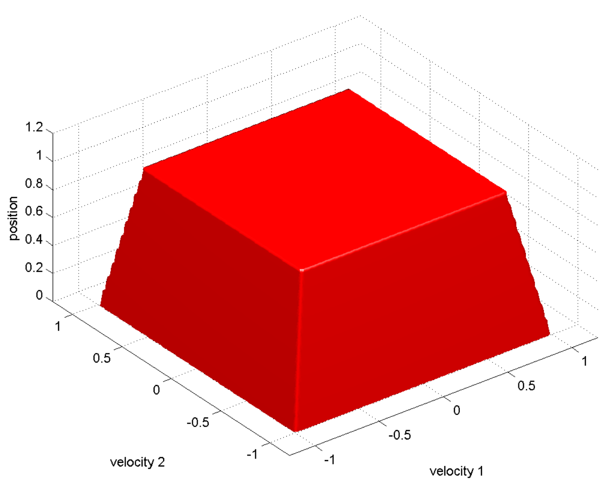
- Solution $\overline{\psi}(t, y)$ provides smallest $x(t, y(t))$ giving rise to a trajectory which reaches the upper boundary $x(0, y(0)) = \overline{\psi}_0(y(0))$ of the target set at $t = 0$

Rotating Double Integrator

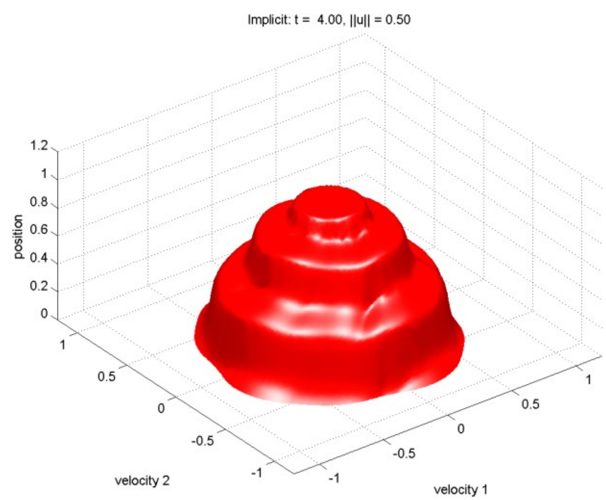
- Let $u \in U = \{u \in \mathbb{R}^2 \mid \|u\|_2 \leq u_{\max}\}$ and

$$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} -y_2 \\ +y_1 \end{bmatrix} + \mu(\|y\|_2) \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \qquad \dot{x} = \|y\|_2$$

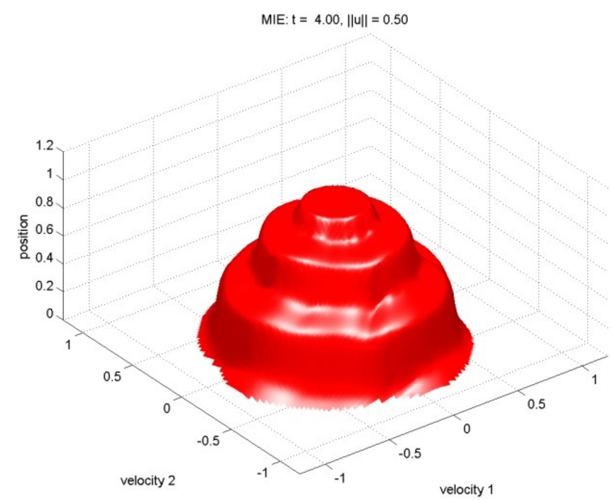
- Behaves radially like first quadrant of traditional double integrator for $\mu(\alpha) \equiv 1$
- For this experiment, $\mu(\alpha) = 2\sin(4\pi\alpha)$



Target Set



Implicit



MIE

Pursuit of an Oblivious Vehicle

- Modified game of two identical vehicles
  - Evader has fixed linear velocity and heading
  - Pursuer has linear acceleration and angular velocity as inputs
- Position variables treated as separate terminal integrators

parameters

$$a_p \in [-0.2, +0.2]$$

$$\omega_p \in [-0.2, +0.2]$$

$$v_p \in [1.0, 3.0]$$

$$v_e = 1.0$$

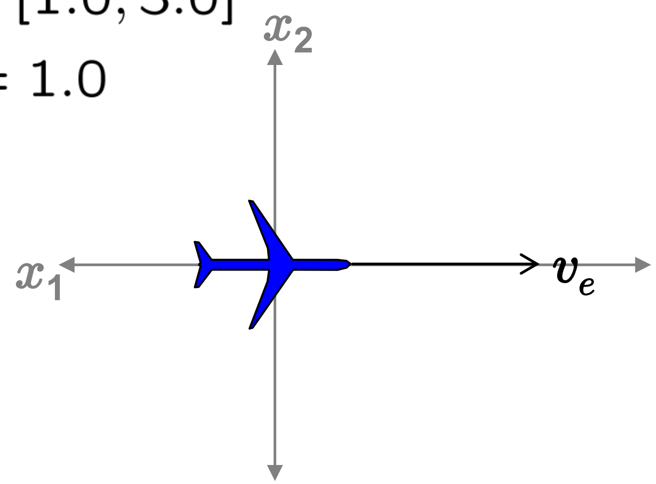
dynamics

$$\frac{d}{dt}\begin{bmatrix}\theta \\ v_p \\ x_1 \\ x_2\end{bmatrix} = \begin{bmatrix}\omega_p \\ a_p \\ -v_e + v_p \cos\theta \\ v_p \sin\theta\end{bmatrix}$$

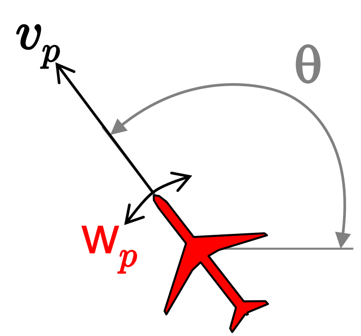
target set description

$$x_1 \in [-1, +1]$$

$$x_2 \in [-1, +1]$$



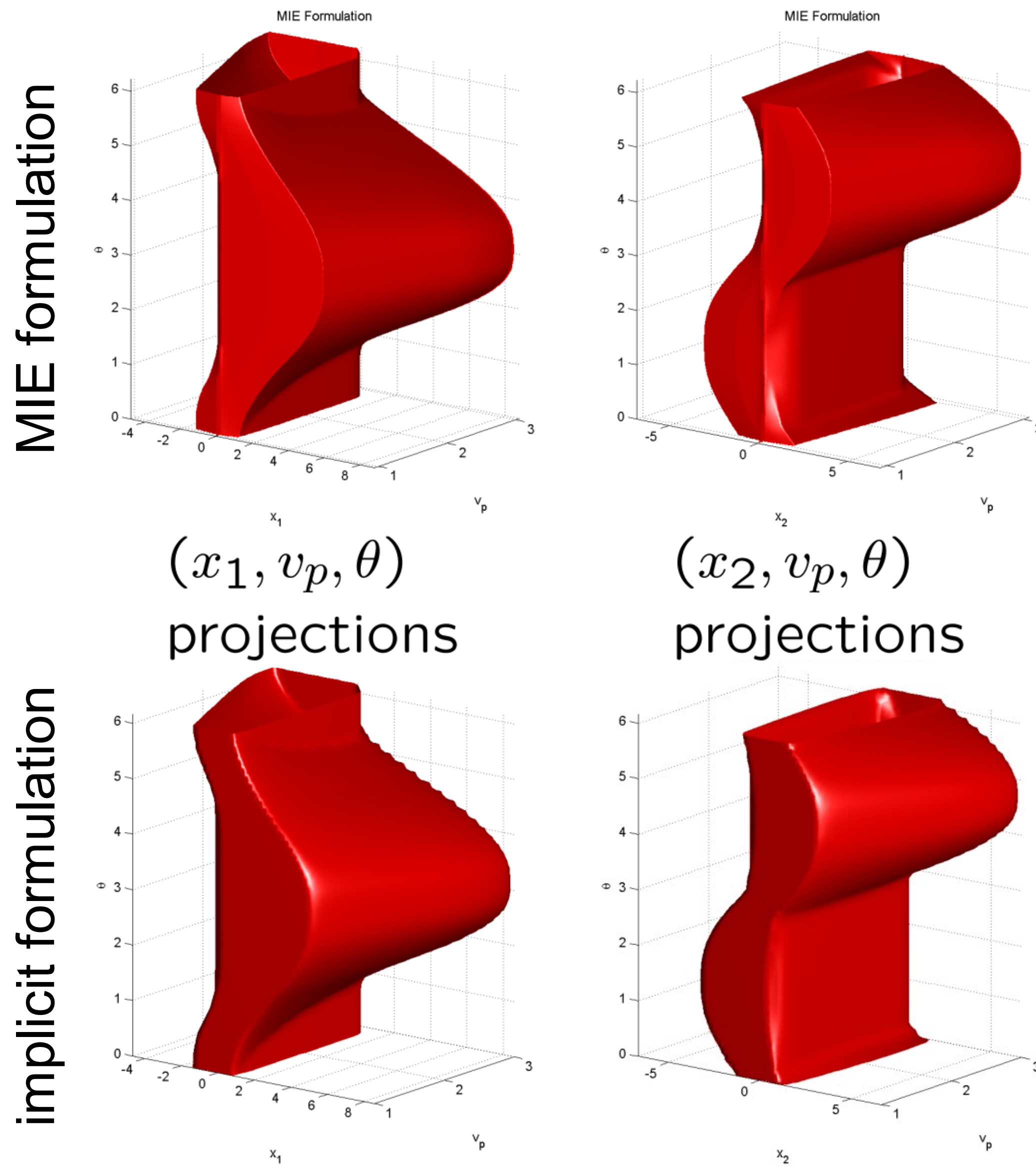
evader aircraft (oblivious)



pursuer aircraft (disturbance)

MIE versus Fully Implicit

- Difficult to visualize four dimensional reach tube
- Projections onto subspaces
  - Directly calculated by MIE formulation
  - Projected as a post-processing step in implicit formulation

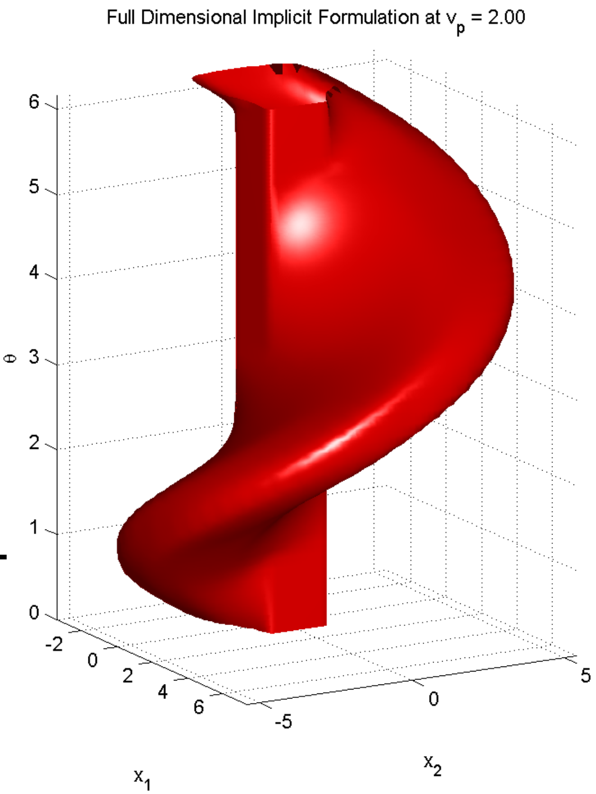


MIE Pros and Cons

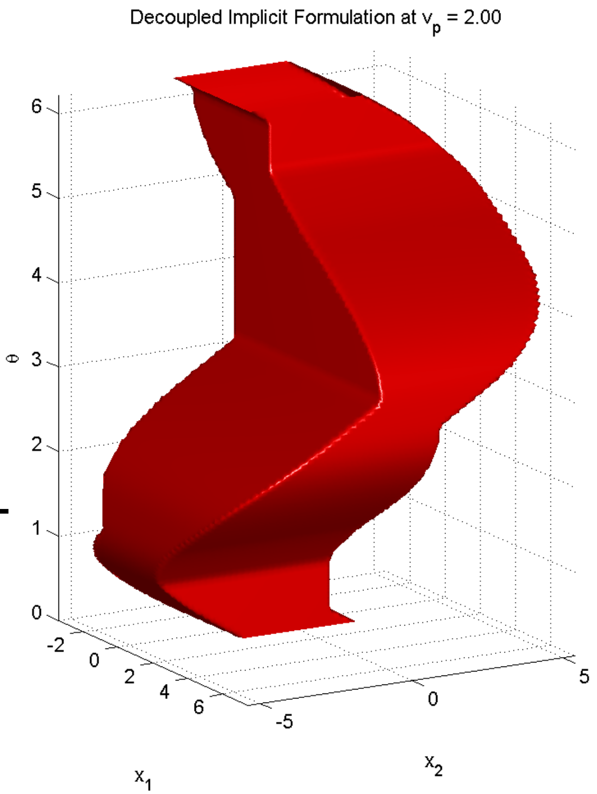
- MIE computation is much less costly
  - MIE: four HJ PDEs in two dimensions took ~3 seconds
  - Implicit: one HJ PDE in four dimensions took too much memory, but estimated at ~30 hours
- MIE computation works in state space projections
  - Overapproximation of reach tube is inevitable

slice of reach tube for v_p = 2.0 (using backprojection for decoupled formulation)

implicit formulation



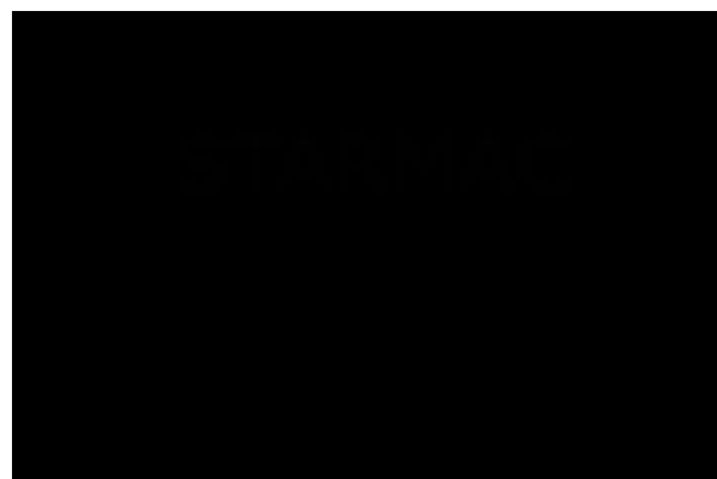
decoupled formulation



Safely Switching Control Modes

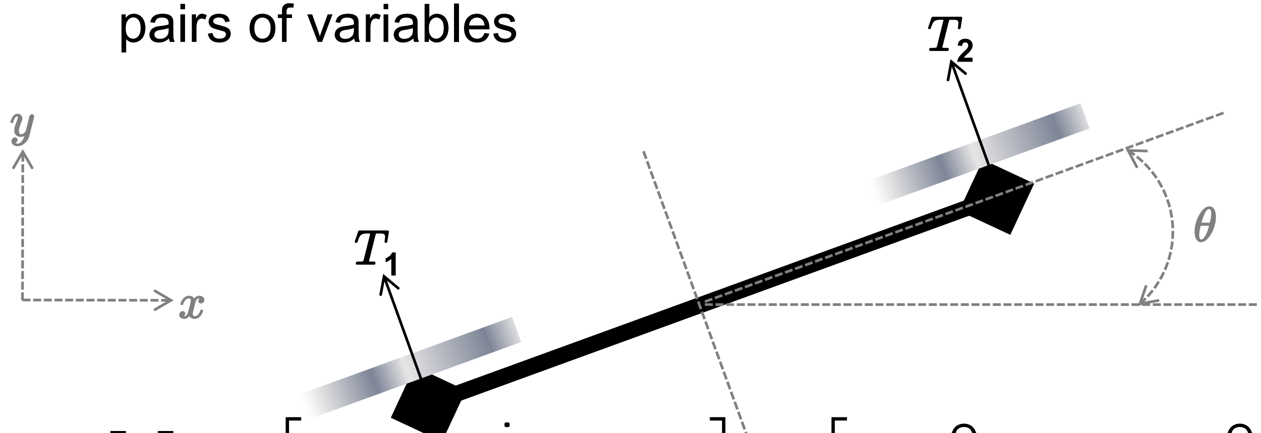
- One application of reach sets is to determine when it is safe to switch between distinct control modes
  - Final mode has region S_0 within which final mode's controller is known to be stable
  - Compute $B(S_0, [\ 0, t_0\ ])$ using dynamics for final mode's controller to determine region within which switch to final mode is safe
  - Pick $S_1 \subset B(S_0, [\ 0, t_0\ ])$ as target for second to last mode
  - Compute $B(S_1, [\ 0, t_1\ ])$ using dynamics for second to last mode's controller, and so on

Autonomous Helicopter Backflips
Jeremy H. Gillula
Haomiao Huang
Michael P. Vitus
Claire J. Tomlin
ICRA 2010



Three Dimensions is Child's Play

- Simplified longitudinal quadrotor dynamics are six dimensional
  - Assumes that out-of-plane dynamics can be stabilized
  - Analysis performed separately on three position / velocity pairs of variables



$$\frac{\partial}{\partial t}\begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ \theta \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{x} \\ -\frac{1}{m}C_D^v \dot{x} \\ \dot{y} \\ -\frac{1}{m}(mg + C_D^v \dot{y}) \\ \dot{\theta} \\ -\frac{1}{I_{yy}} C_D^\theta \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -\frac{1}{m}\sin\theta & -\frac{1}{m}\sin\theta \\ 0 & 0 \\ +\frac{1}{m}\cos\theta & +\frac{1}{m}\cos\theta \\ 0 & 0 \\ -\frac{1}{I_{yy}} & \frac{1}{I_{yy}} \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \end{bmatrix}$$

Contents (not strictly ordered)

- Backward reach sets & tubes
  - Treatment of inputs
- Formulation as finite horizon optimal control
  - Implicit surface functions
  - Modification for optimal stopping
- Game of two identical vehicles
  - HJ PDE calculation
  - Analytic solution (almost)
  - Synthesis of safe controls
- Reducing the dimension
  - Systems with terminal integrators
  - Mixed implicit explicit representation
  - Target application: safety for the quadrotor flip

Reachability: An Application of the Time-Dependent Hamilton-Jacobi Equation

For more information contact

Ian Mitchell

Department of Computer Science

The University of British Columbia

`mitchell@cs.ubc.ca`

`http://www.cs.ubc.ca/~mitchell`