Scalable Calculation of Reach Sets and Tubes for Nonlinear Systems with Terminal Integrators

A Mixed Implicit Explicit Formulation

Ian Mitchell

Department of Computer Science

The University of British Columbia

research supported by

National Science and Engineering Research Council of Canada

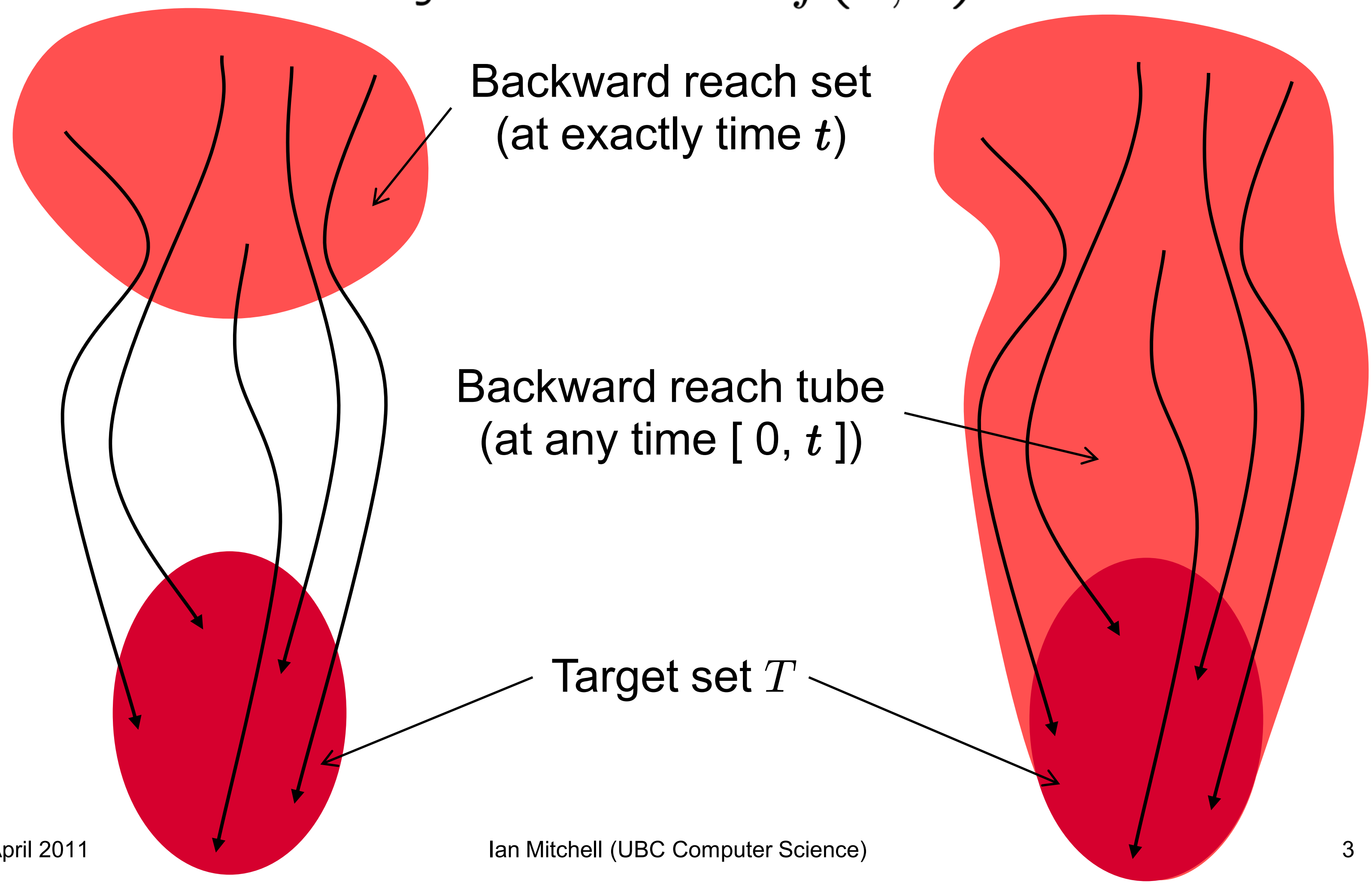
Outline

- Reachability
  - Backwards reach sets and tubes with inputs
  - Implicit surface representation of sets
  - Game of two identical vehicles
  - Formulation as a time-dependent HJ PDE
- The mixed implicit explicit (MIE) formulation of reach sets and tubes
  - Dynamics with terminal integrators
  - Hamilton-Jacobi / optimal control formulation of terminal integrator's reach set / tube
  - Examples: various double integrators and the pursuit of the oblivious evader

Backward Reachability

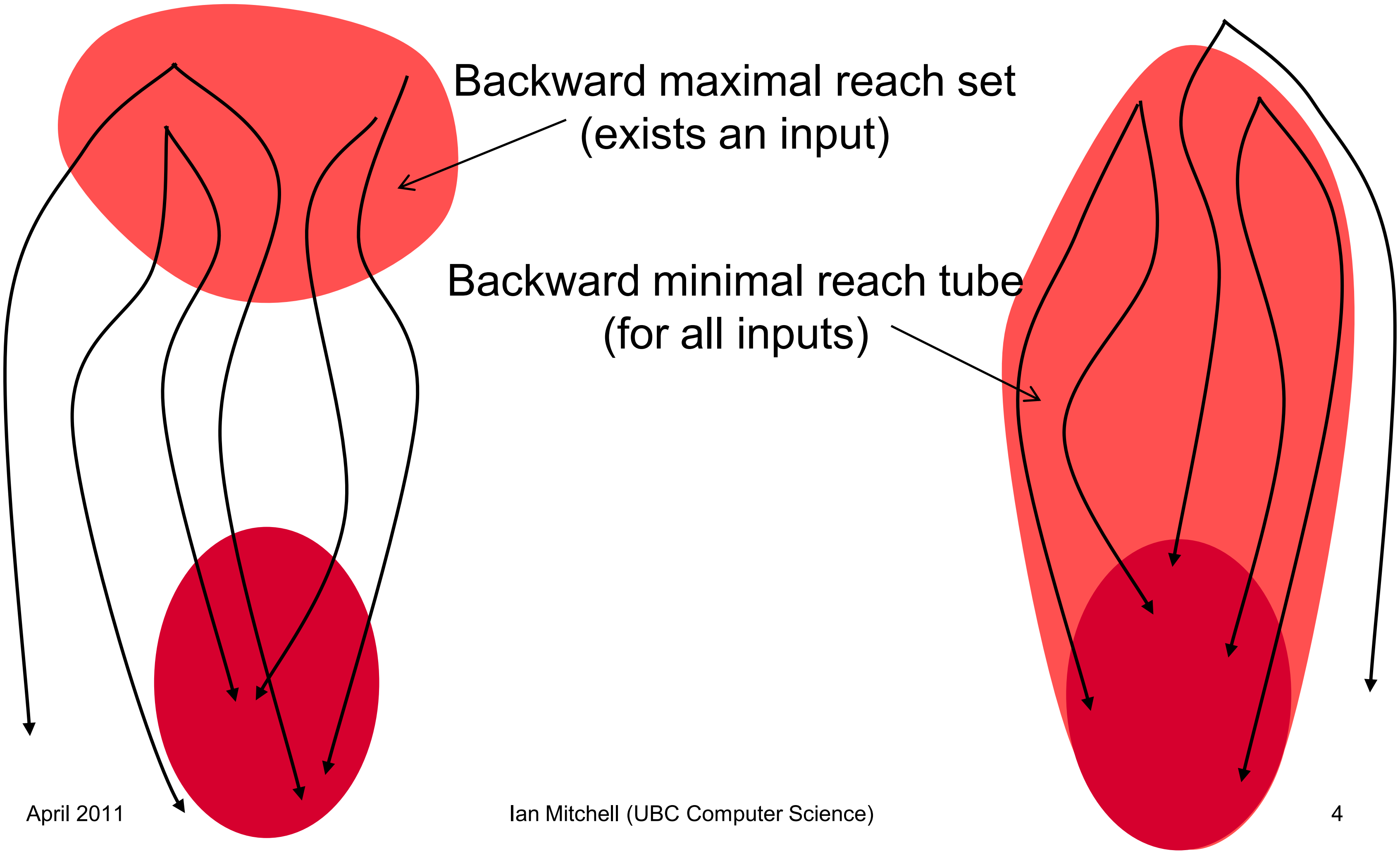
- Find trajectories leading to a target set

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



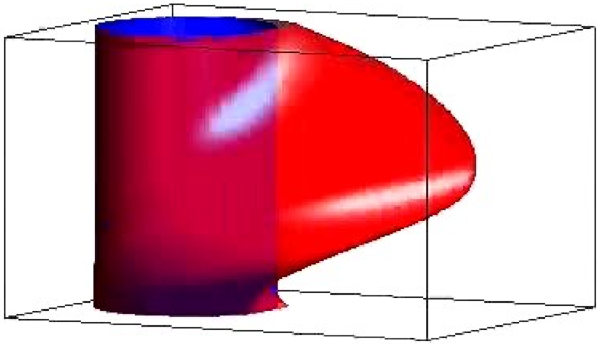
Nonlinear, Nondeterministic Dynamics

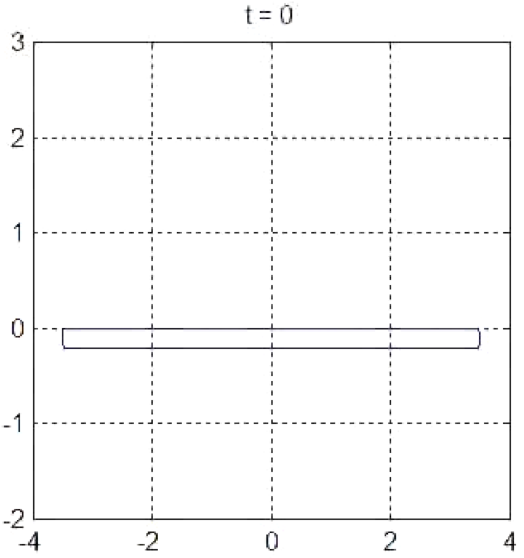
- Input $u \in U$ can maximize or minimize the set or tube
  - Interpretation as best or worst case depends on context



Determining Reach Sets

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



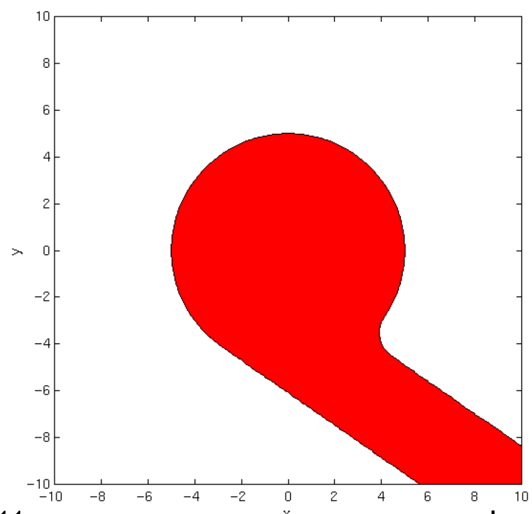


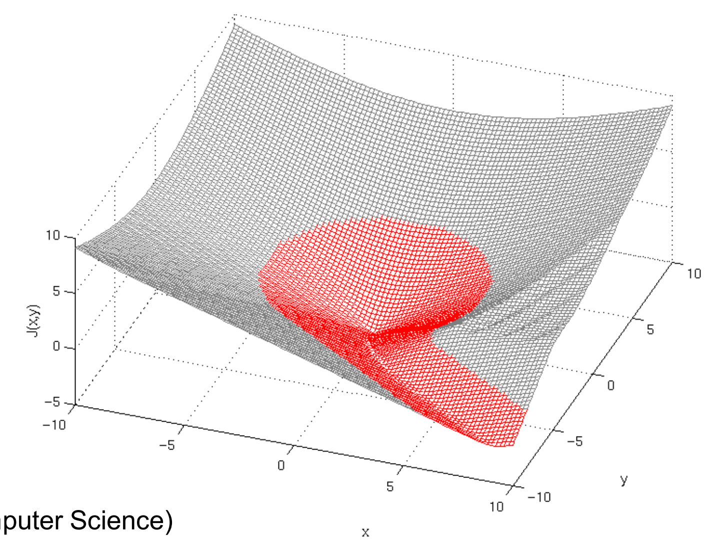
Implicit Surface Functions

- Set $S(t)$ 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

$$\phi : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$$

$$S(t) = \{x \in \mathbb{R}^n \mid \phi(x,t) \leq 0\}$$





Backward Reachability Algorithms

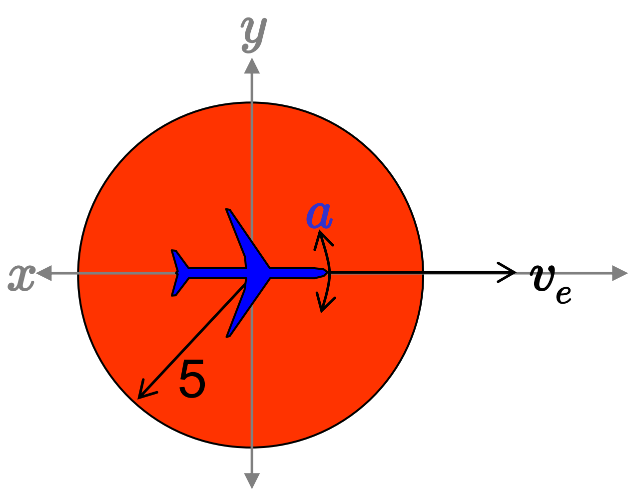
- Implicit / non-parametric representation typical of Eulerian algorithms
  - Representation is not moving (although it may adapt)
  - No prespecified class of sets
  - Generally handle nonlinear dynamics and multiple inputs
  - No examples beyond four dimensions?
- Examples
  - [Broucke, Benedetto, Gennaro & Sangiovanni-Vincentelli, HSCC 2001]
  - [Saint-Pierre, HSCC 2002]
  - [Sethian & Vladimirsky, HSCC 2002]
  - [Mitchell, Bayen & Tomlin, IEEE TAC 2005]
  - [Kitsios & Lygeros, CDC 2005]
  - [Djeridane & Lygeros, CDC 2006]
  - [Gao, Lygeros & Quincampoix, HSCC 2006]

Game of Two Identical Vehicles

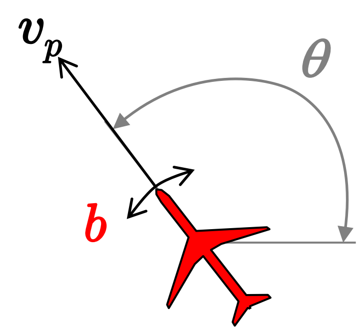
- Classical collision avoidance example
  - Collision if vehicles get within five units of one another
  - Evader chooses turn rate $|a| \leq 1$ to avoid collision
  - Pursuer chooses turn rate $|b| \leq 1$ to cause collision
  - Fixed equal velocity $v_e = v_p = 5$

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)

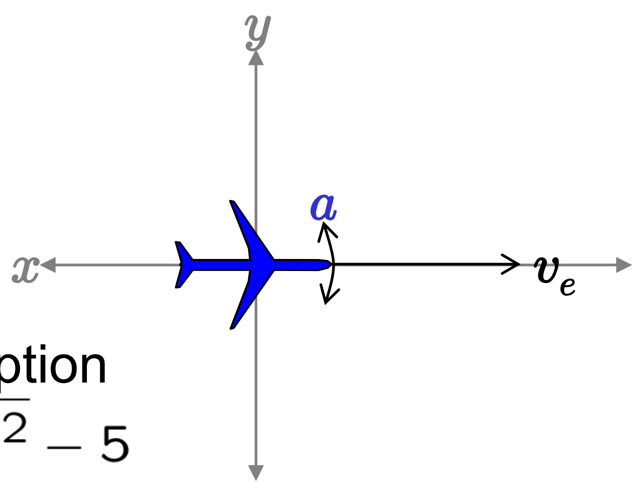
Collision Avoidance Computation

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

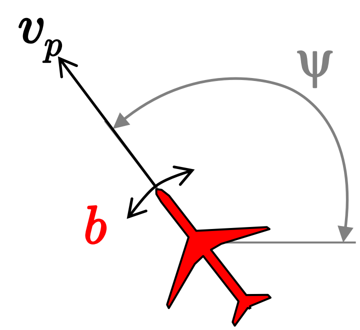
$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ \psi \end{bmatrix} = \begin{bmatrix} -v_e + v_p \cos\psi + ay \\ v_p \sin\psi - ax \\ b - a \end{bmatrix}$$

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
  - [Lygeros, Tomlin & Sastry, *Automatica* 1999], [Lygeros, *Automatica* 2004], [Mitchell, Bayen & Tomlin, *IEEE TAC* 2005]

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

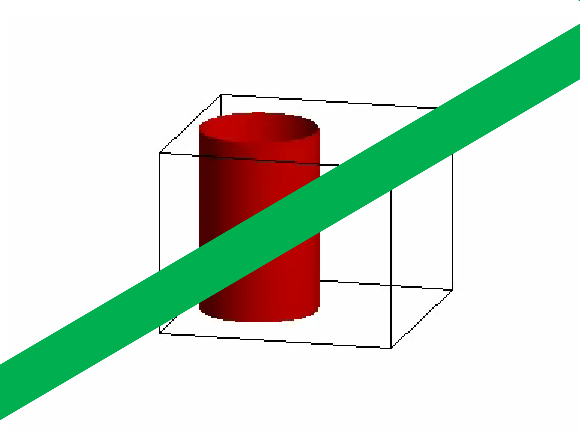
ith Hamiltonian : $H(x,p) = \max_{a\in\mathcal{A}} \min \quad , a, b) \cdot p$

and termina onditions : $\phi(x,0) = h($

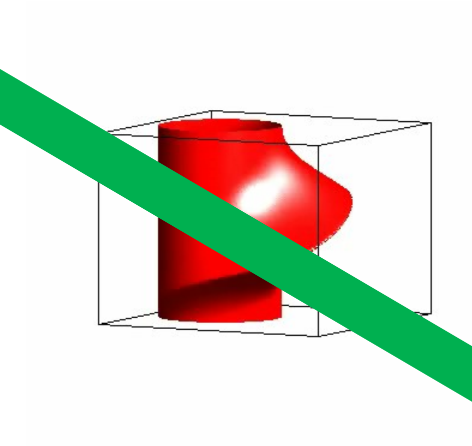
here $G(0)$ $\{x \in \mathbb{R}^n \mid h(x) \leq 0\}$

an $x = f(x,a,b)$

wth of reachable set



final reachable



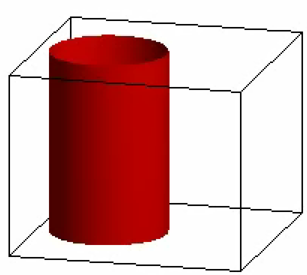
Hey! Where's the Good Old Blobby?

- Two identical vehicle collision avoidance
- Nonlinear dynamics with cooperative inputs

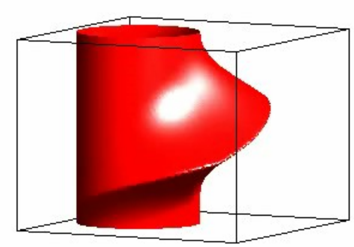
$$D_t\phi(x,t) + \min\left[0, H(x, D_x\phi(x,t))\right] = 0$$

$$\begin{aligned}
\text{with Hamiltonian}: \quad & H(x,p) = \max_{a\in\mathcal{A}} \overset{\max}{\cancel{\min_{b\in\mathcal{B}}}} f(x,a,b)\cdot p \\
\text{and terminal conditions}: \quad & \phi(x,0) = h(x) \\
\text{where} \quad & G(0) = \{x \in \mathbb{R}^n \mid h(x) \le 0\} \\
\text{and} \quad & \dot{x} = f(x,a,b)
\end{aligned}$$

growth of reachable set



final reachable set



Outline

- Reachability
  - Backwards reach sets and tubes with inputs
  - Implicit surface representation of sets
  - Game of two identical vehicles
  - Formulation as a time-dependent HJ PDE
- The mixed implicit explicit (MIE) formulation of reach sets and tubes
  - Dynamics with terminal integrators
  - Hamilton-Jacobi / optimal control formulation of terminal integrator's reach set / tube
  - Examples: various double integrators and the pursuit of the oblivious evader

Systems with Terminal Integrators

- Common form of system dynamics

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

- 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 $\mathcal{O}\left(2d_x n^{d_y}\right)$

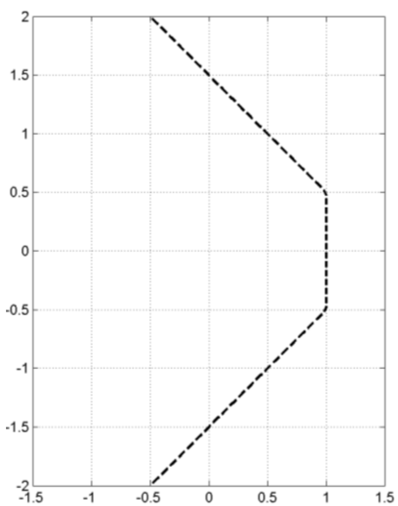
Mixed Implicit Explicit Formulation

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

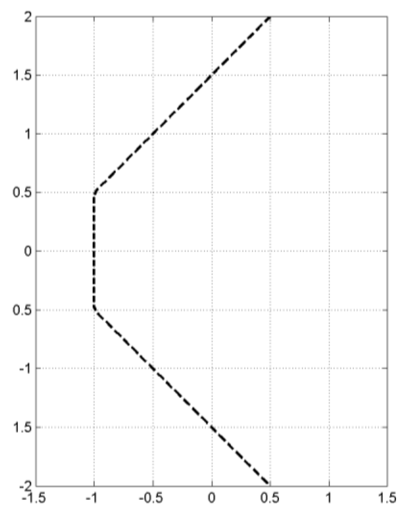
$$S = \{(x, y) \mid \psi(x, y) \leq 0\}$$

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

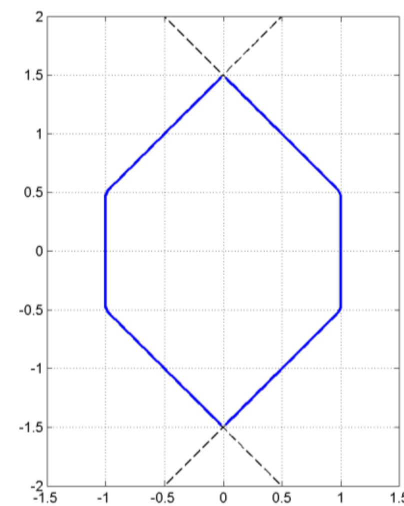
$$S = \left\{(x, y) \,\middle|\, \underline{\psi}_i(y) \leq x_i \leq \overline{\psi}_i(y)\right\}$$



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

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

- 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

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

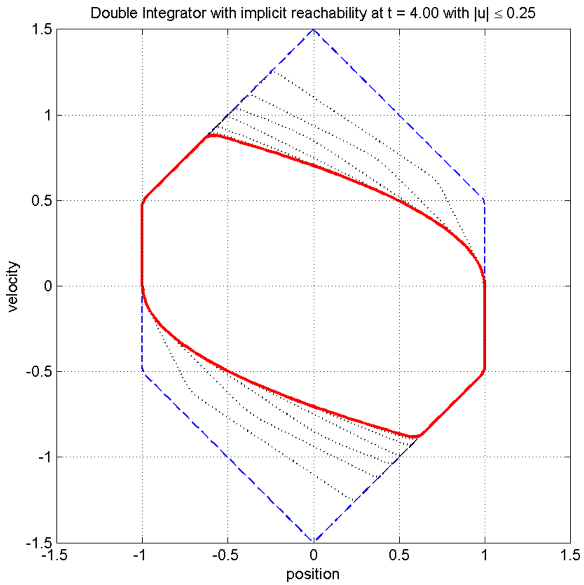
Double Integrator

- Dynamics

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

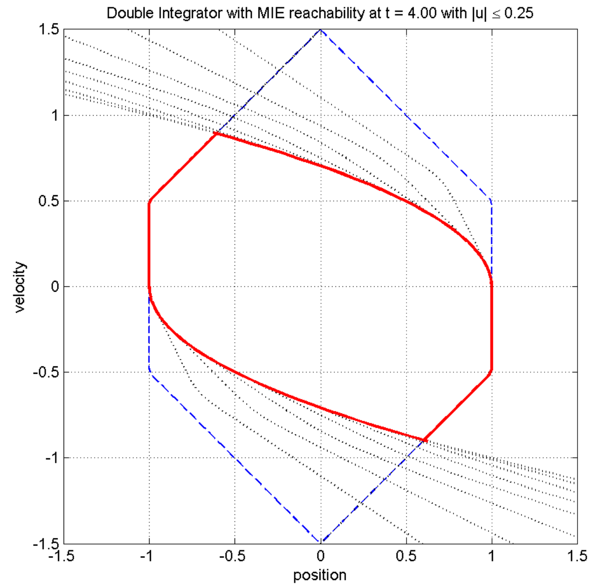
yields terminal integrator Hamiltonian (for upper bound)

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



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

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

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

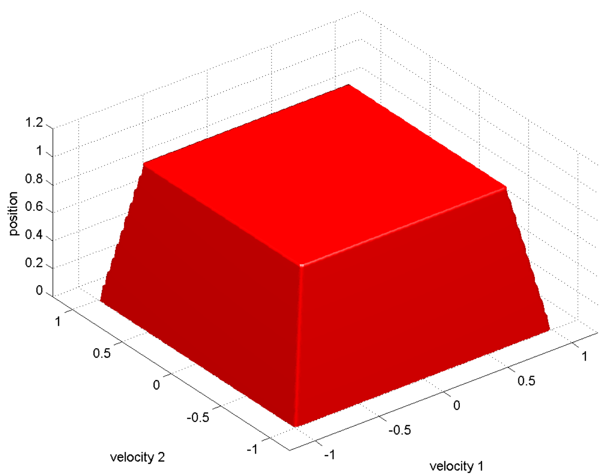
- Solution $\overline{\psi}(t, y)$ provides smallest $x(t, y(t))$ giving rise to a trajectory which reaches the upper boundary $x(0, y(0)) = \overline{\psi}_0(y(0))$ of the target set at $t = 0$

Rotating Double Integrator

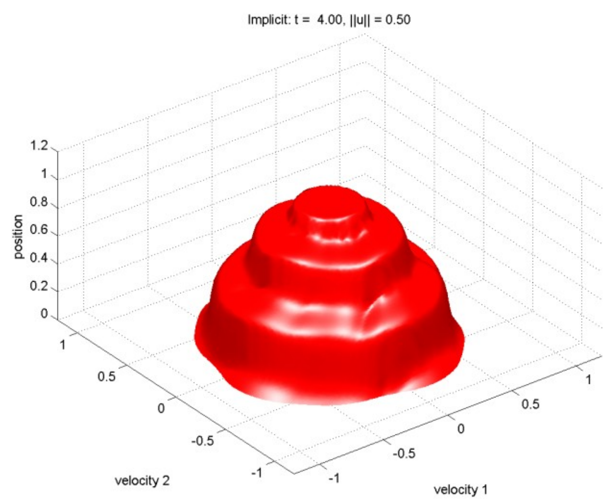
- Let $u \in U = \{u \in \mathbb{R}^2 \mid \|u\|_2 \leq u_{\max}\}$ and

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

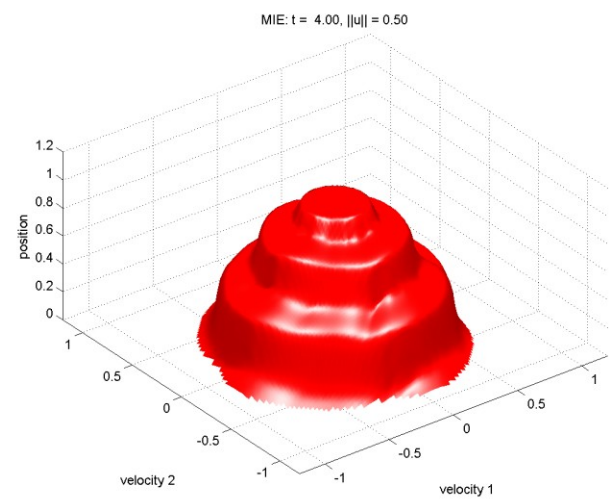
- Behaves radially like first quadrant of traditional double integrator for $\mu(\alpha) \equiv 1$
- For this experiment, $\mu(\alpha) = 2\sin(4\pi\alpha)$



Target Set



Implicit



MIE

Terminal Integrators with Linear Self-Coupling

- Generalization of terminal integrator's dynamics

$$\dot{x} = a(y)x + b(y)$$

- Leads to an HJ PDE

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

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

  - HJ PDEs of this form arise in discounted finite horizon optimal control
  - In this interpretation, $a(y)$ is the discount factor
- Existing viscosity solution theory supports positive constant $a(y) \equiv a > 0$
  - Stable systems will have $a(y) < 0$ (for backwards reachability)
  - It is probably possible to extend the theory

This Ain't No Double Integrator

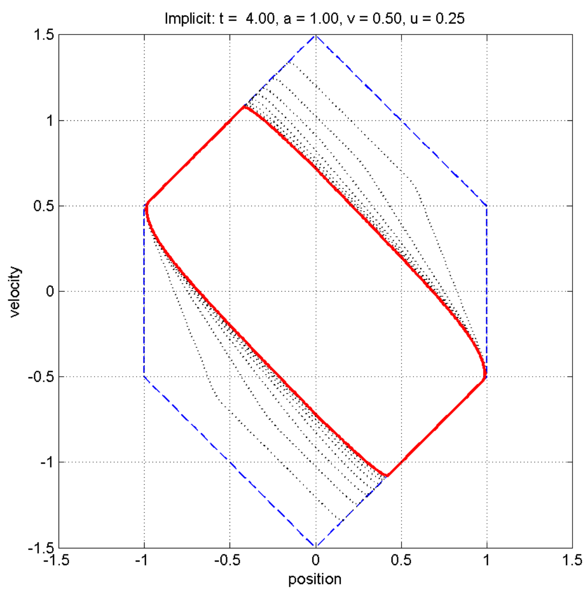
- Theoretically supported but unstable dynamics

$$\dot{x} = ax + y + v \qquad \dot{y} = u$$

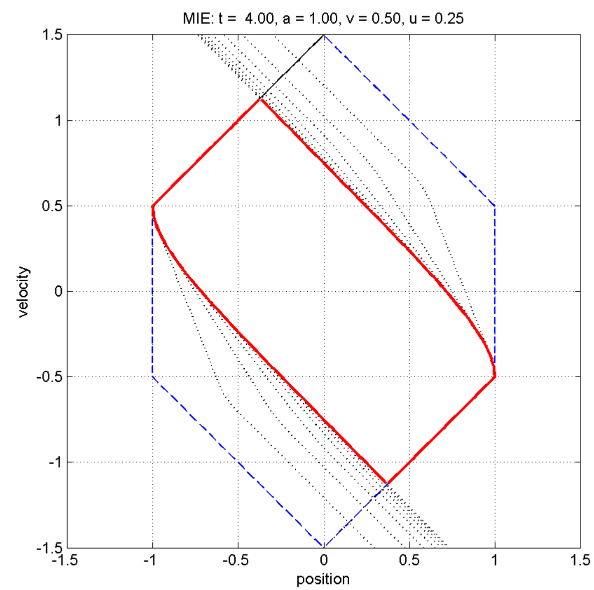
with $a = 1 > 0, \quad |u| \leq u_{\max} = 0.25, \quad |v| \leq v_{\max} = 0.5$

yields Hamiltonian

$$H(t, y, q, p) = \max_{u,v}(p \cdot u - aq - y - v) = (|p|u_{\max} - aq - y + v_{\max})$$



Regular implicit surface formulation
One HJ PDE in 2D

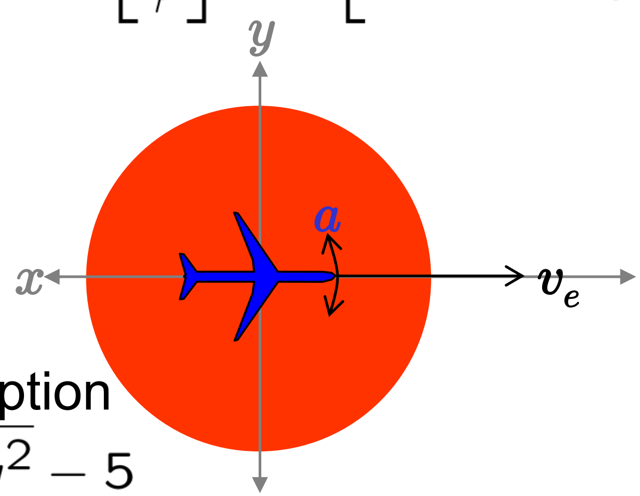


Terminal integrator formulation
Two HJ PDEs in 1D

MIE for the Collision Avoidance Example?

- No terminal integrators: all variables are coupled for evader input $a \neq 0$
  - Also, target set has coupling of x and y

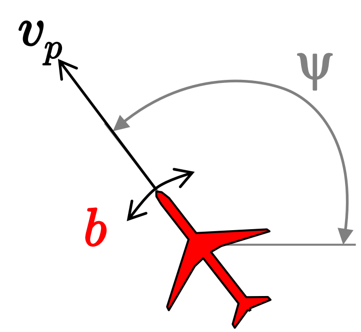
$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ \psi \end{bmatrix} = \begin{bmatrix} -v_e + v_p \cos\psi + ay \\ v_p \sin\psi - ax \\ b - a \end{bmatrix}$$



target set description

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

evader aircraft (control)



pursuer aircraft (disturbance)

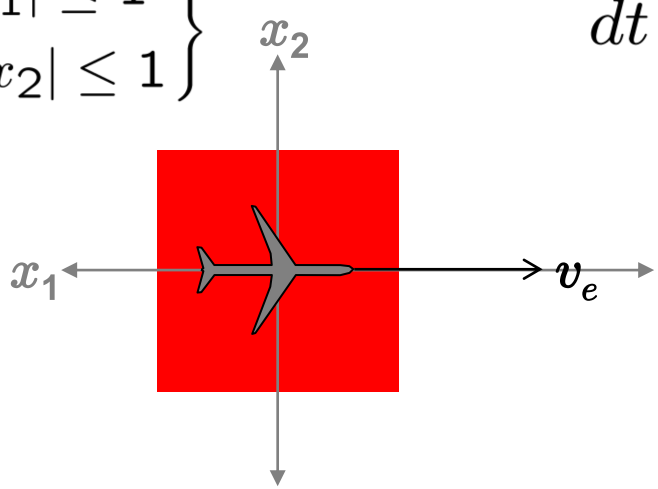
Pursuit of an Oblivious Evader

- The oblivious evader has no input
- Relative position variables are terminal integrators
  - Decouple target set by using a box
- Allow pursuer to adjust both heading and speed
  - $\omega_p \in \Omega_p$ and $a_p \in A_p$

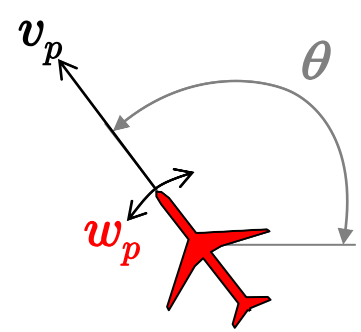
target set description

$$\left\{ (x_1, x_2) \;\middle|\; \begin{array}{c} |x_1| \leq 1 \\ \wedge |x_2| \leq 1 \end{array} \right\}$$

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



evader aircraft (no input)



pursuer aircraft (disturbance)

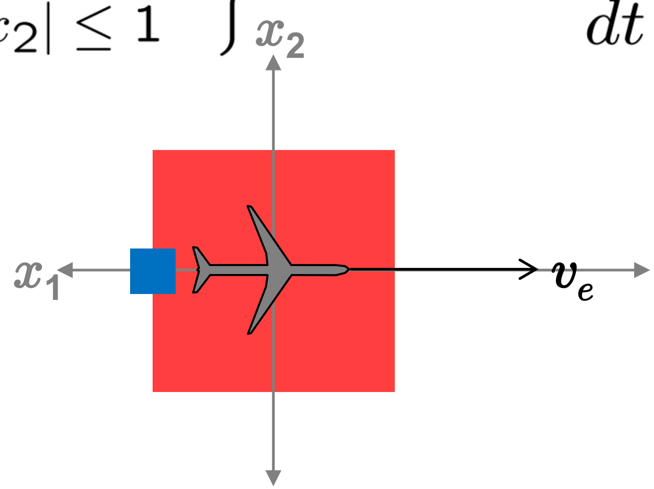
Related Application: Aerial Refueling

- [Ding & Tomlin, CDC 2010]
  - Tanker flies straight at constant speed
  - UAV attempts to reach small rectangular refueling zone without crashing into the tanker
  - Not yet clear how to handle reach-avoid

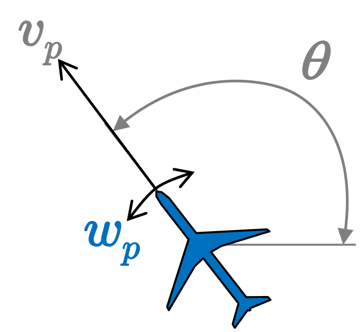
target set description

$$\left\{ (x_1, x_2) \;\middle|\; \begin{array}{c} |x_1 - \hat{x}_1| \le 1 \\ \wedge |x_2| \le 1 \end{array} \right\}$$

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



tanker aircraft (no input)



UAV aircraft (control)

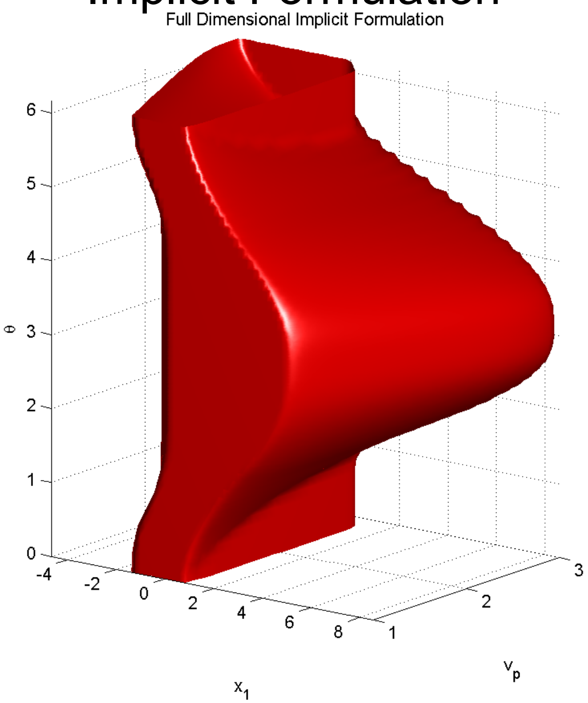
Multiple Terminal Integrators

- Three approaches
  - Solve one HJ PDE in d_y + d_x dimensions (y, x_1, ..., x_{d_x}) to get a full implicit representation of the reach set / tube
  - Solve one HJ PDE in each of d_x separate subspaces (y, x_i) of dimension d_y + 1 to get implicit representations of the projections of the reach set / tube [Mitchell & Tomlin, JSC 2003]
  - Solve two HJ PDEs for each of d_x separate dimensions in the subspace (y) of dimension d_y to get MIE representation of the projections of the reach set / tube
- Projection-based representations require decoupling of the inputs
  - Independent choice of input in each projection could be pessimistic but sound, or could introduce leaky corners

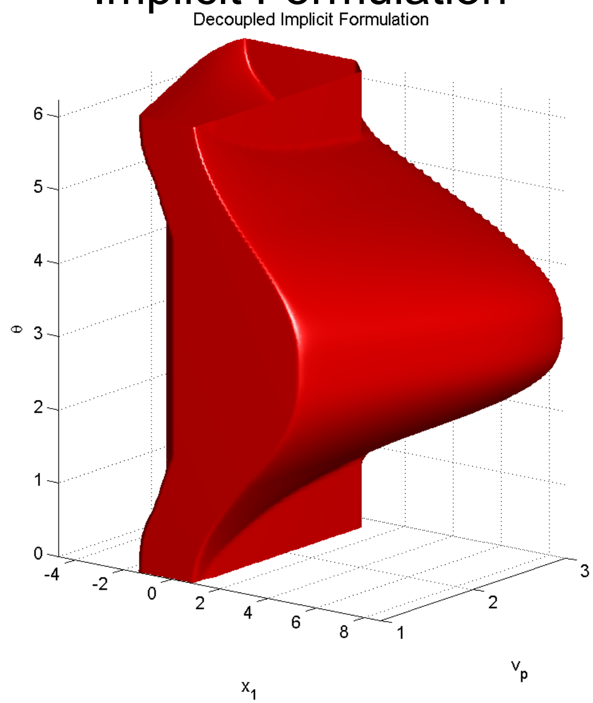
Pursuit of the Oblivious Evader

- Parameters for this run were v_e = 1, A_p = [-0.2,+0.2], Ω_p = [-0.2, +0.2], $v_p \in$ [1.0, 3.0], $t_{\max}$ = 2.0
- Projections into (x_1, v_p, θ)

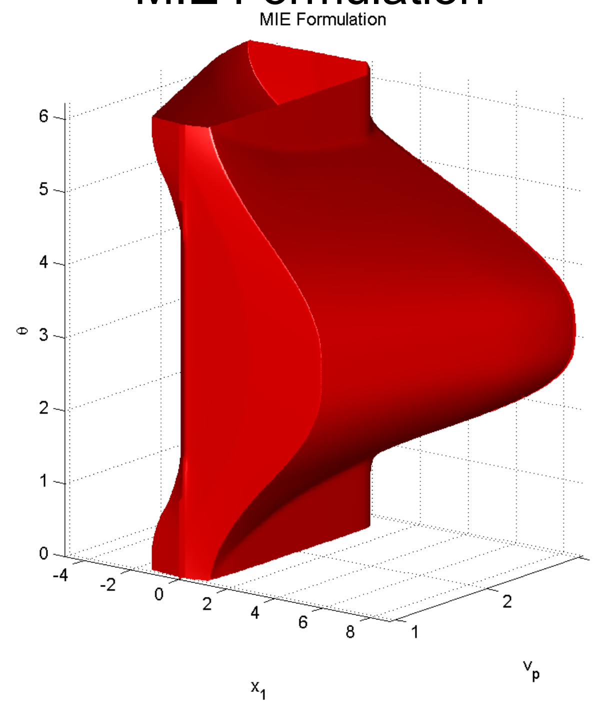
Full Dimensional Implicit Formulation



Decoupled Implicit Formulation



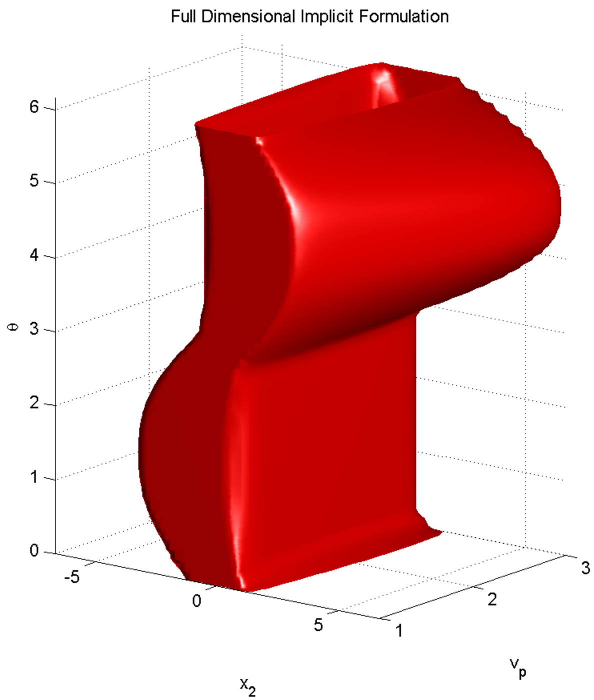
Decoupled MIE Formulation



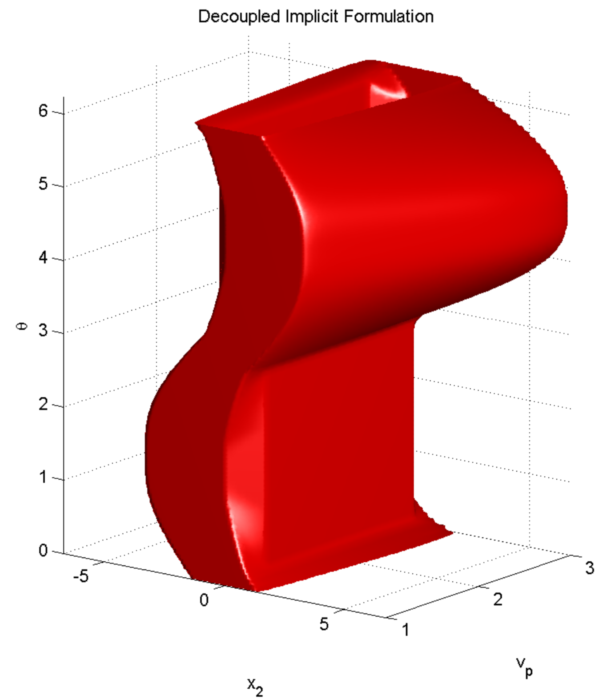
Pursuit of the Oblivious Evader

- Parameters for this run were v_e = 1, A_p = [-0.2,+0.2], Ω_p = [-0.2, +0.2], $v_p \in$ [1.0, 3.0], $t_{\max}$ = 2.0
- Projections into (x_2, v_p, θ)

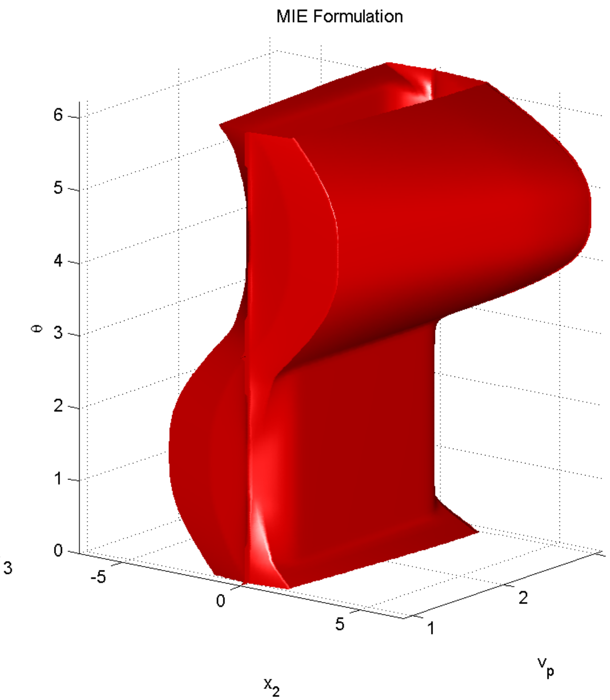
Full Dimensional
Implicit Formulation



Decoupled
Implicit Formulation

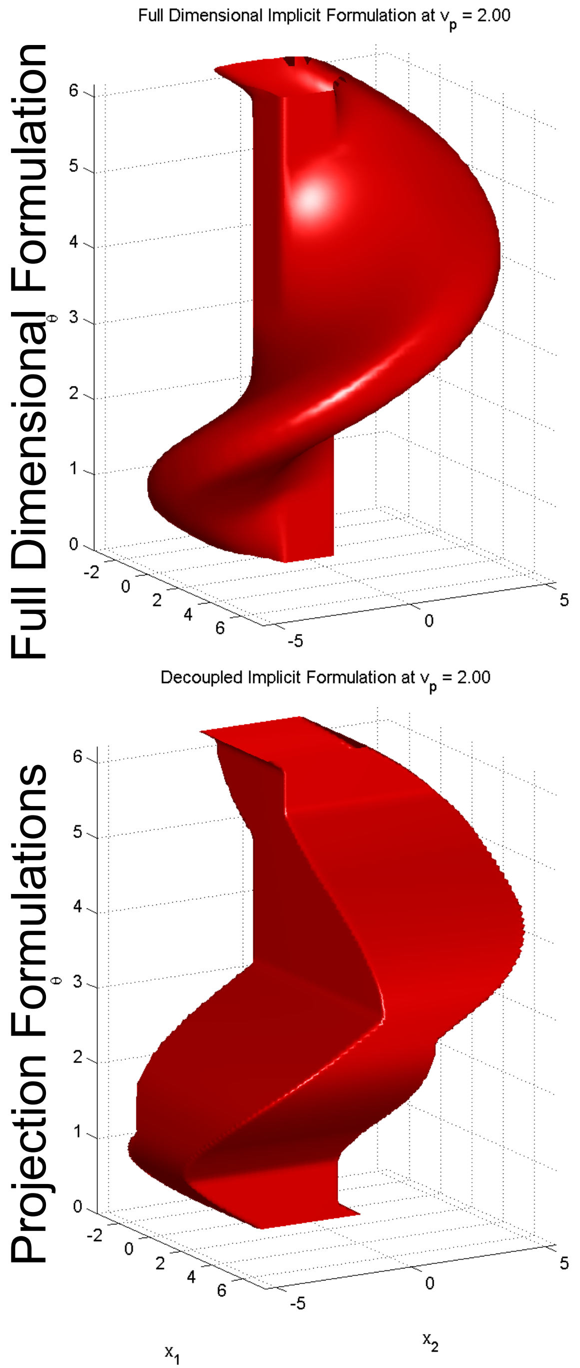


Decoupled
MIE Formulation



Pros & Cons

- Slices of reach tube at v_p = 2
  - Projection formulations cannot represent true reach tube, must overapproximate
- Inputs calculated separately in each projection
  - Results are pessimistic but sound
  - Not appropriate for refueling scenario
- Computational cost
  - MIE: 65×100 grid, 3.1 seconds
  - Decoupled implicit: $151 \times 65 \times 100$ grid, 541 seconds
  - Full dimensional implicit: $151^2 \times 65 \times 100$ grid estimated at 30 hours



Comparing Formulations

- Memory and computational cost
  - Full implicit: $\mathcal{O}(n^{(d_y+d_x)})$
  - Decoupled implicit: $\mathcal{O}(d_x n^{(d_y+1)})$
  - Decoupled MIE: $\mathcal{O}(2 d_x n^{d_y})$
- Benefits of implicit representations
  - Implicit surface function remains continuous: easier theory, numerics, application of constraints
  - Narrowband or local level set schemes reduce cost
  - Artificial boundary conditions at the edge of the computational domain are dealt with more easily
  - Optimal control may be extracted from gradient of solution
- ToolboxLS implementation provides no guarantee on the sign of the error, but the MIE approach can be used with other implementations

Reproducible Research

"[a]n article about computational science in a scientific publication is not the scholarship itself, it is merely advertising of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions which generated the figures."
[Jon Claerbout, as quoted by Buckheit & Donoho, 1995]

- Scientific results should be independently replicable (given appropriate resources)
  - Key challenges in computational science: appropriate licensing standards, disciplined software development processes, incentives for authors, publishers & referees
- More details:
  - [Stodden, *IEEE Computing in Science & Engineering* 2009] and [Stodden, *Int. J. Communications Law & Policy*, 2009]
  - Reproducible research workshop, July 13–16, 2011 at UBC in Vancouver (just before ICIAM 2011)
- Examples from this paper:

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

Conclusions

- The Hamilton-Jacobi PDE can be used to compute backwards reachability
  - Reachability formulated as terminal cost optimal control
  - Solution provides implicit representation of reach set / tube (plus costate and proximity information)
  - Solution can be approximated with level set methods
  - Matlab implementation available in ToolboxLS
  - Computational cost of approximating HJ PDE grows exponentially with dimension
- Nonlinear Systems with Terminal Integrators
  - Reachability for terminal integrator states can be decoupled and computed in separate projections
  - Mixed implicit explicit (MIE) representation: Reach set / tube can be represented explicitly as the <u>solution</u> of a terminal + running cost HJ PDE
  - Matlab / ToolboxLS implementation is available, but MIE formulation admits other implementations

Scalable Calculation of Reach Sets and Tubes for Nonlinear Systems with Terminal Integrators

A Mixed Implicit Explicit Formulation

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