Reach Sets and the Hamilton-Jacobi Equation

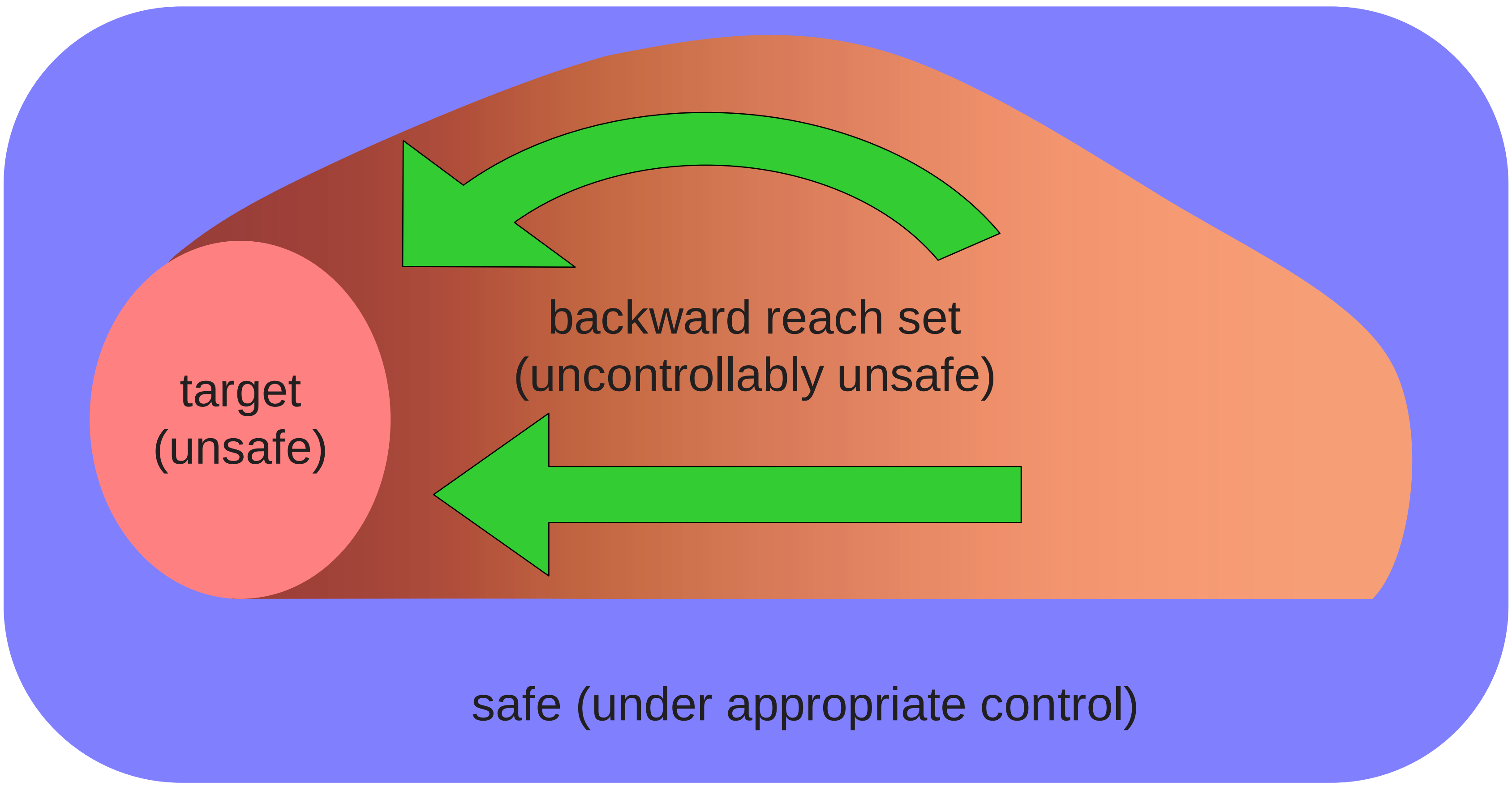
Ian Mitchell
Department of Computer Science
The University of British Columbia

Joint work with
Alex Bayen, Meeko Oishi & Claire Tomlin (Stanford)

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

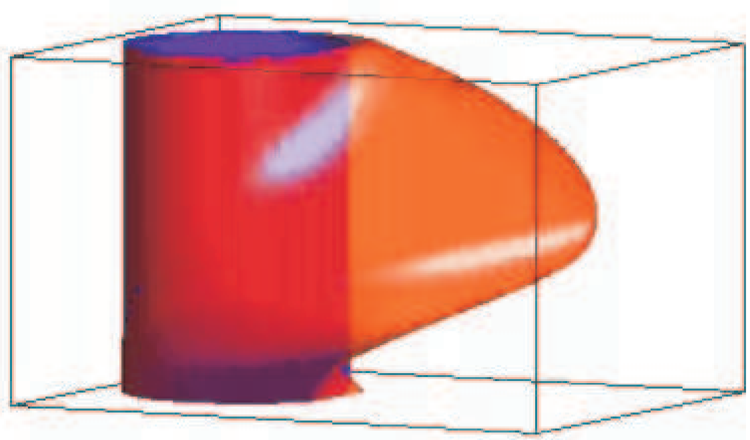
Reachable Sets: What and Why?

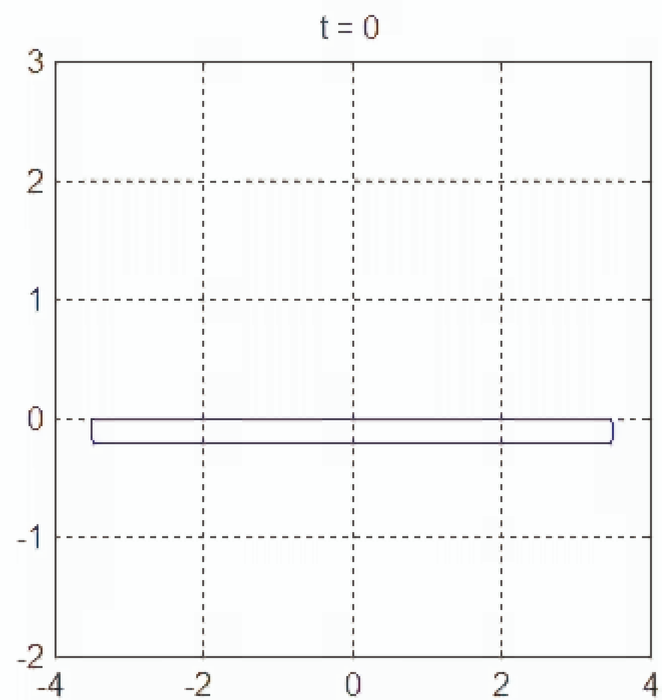
- One application: safety analysis
  - What states are doomed to become unsafe?
  - What states are safe given an appropriate control strategy?



Calculating Reach Sets

- 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 $dx/dt = f(x)$?



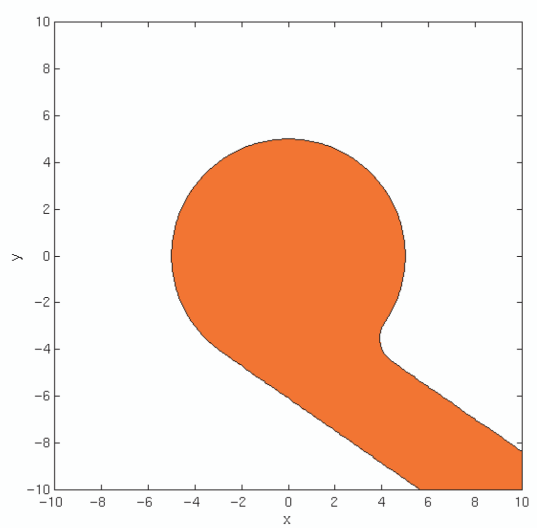


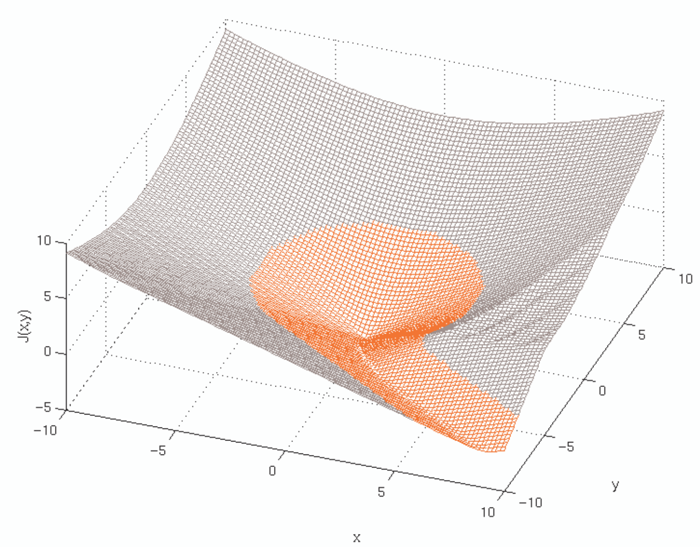
Implicit Surface Functions

- 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

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

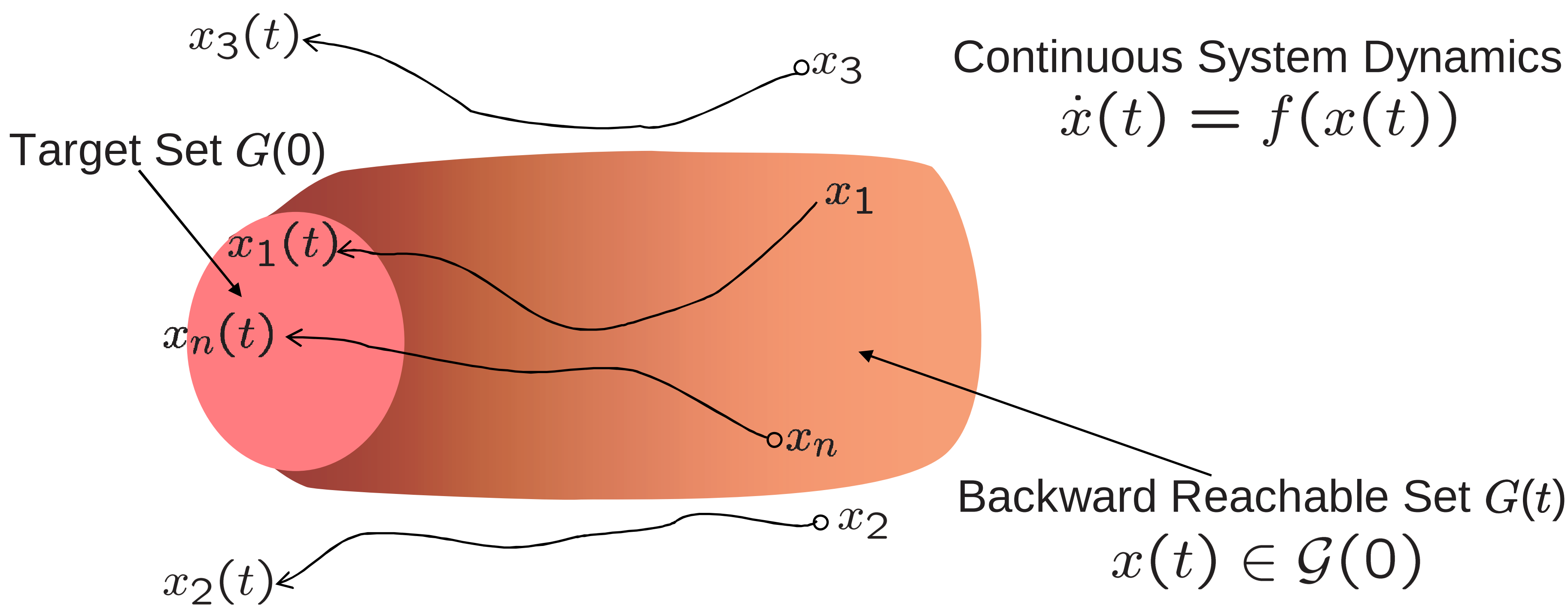
$$\mathcal{G}(t) = \{x \in \mathbb{R}^n \mid \phi(x,t) \leq 0\}$$





Continuous Backward Reachable Sets

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



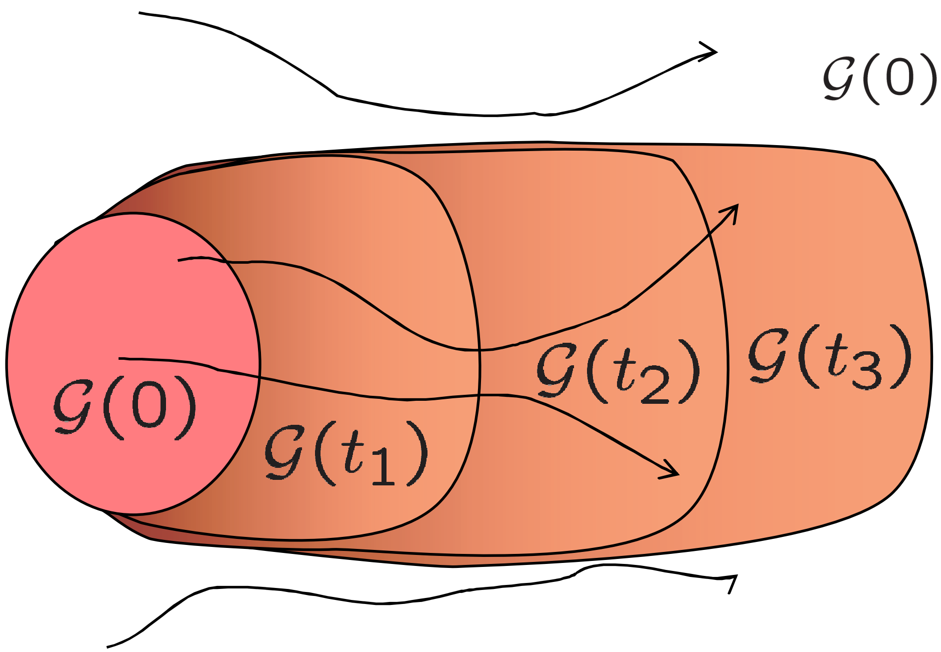
Why “Backward” Reachable Sets?

- To distinguish from forward reachable set
- To compute, run dynamics backwards in time from target set

$$\dot{x}(t) = -f(x(t))$$

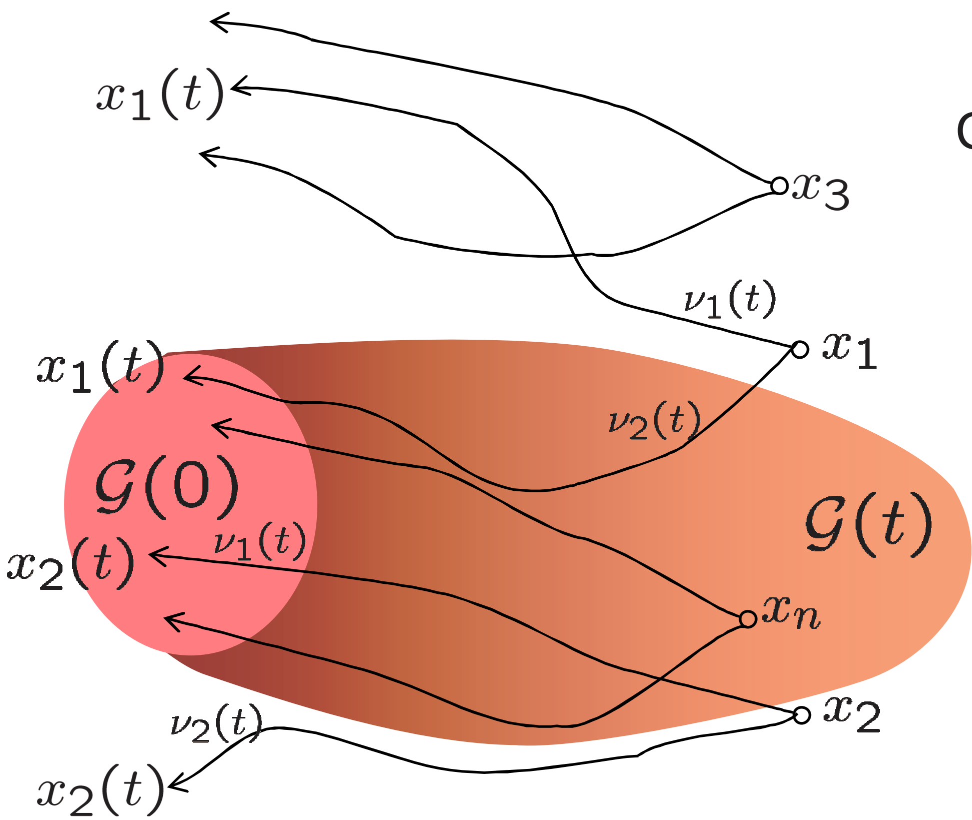
$$0 < t_1 < t_2 < t_3$$

$$\mathcal{G}(0) \subseteq \mathcal{G}(t_1) \subseteq \mathcal{G}(t_2) \subseteq \mathcal{G}(t_3)$$



Reachable Sets (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



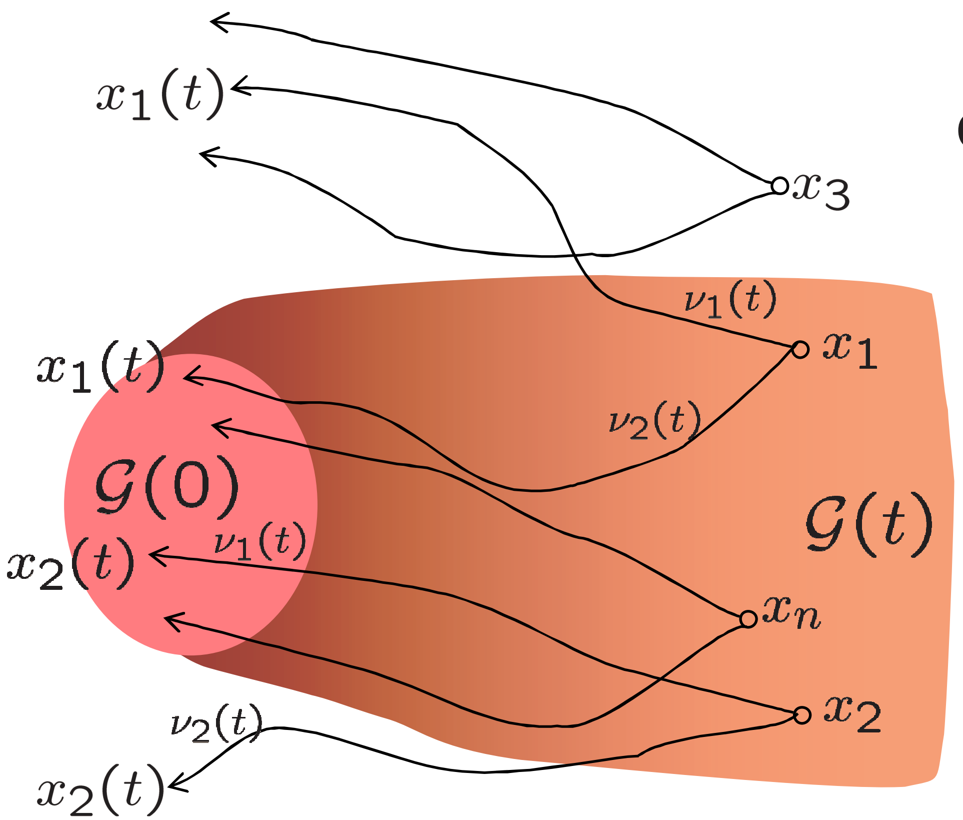
Continuous System Dynamics

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

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

Reachable Sets (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



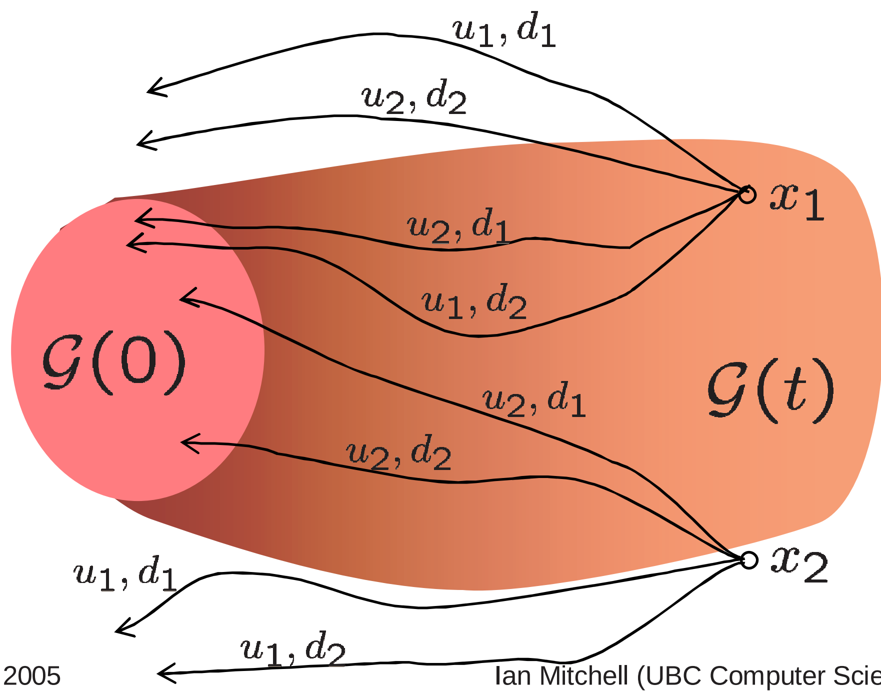
Continuous System Dynamics

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

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

Two Competing Inputs

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



Continuous System Dynamics

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

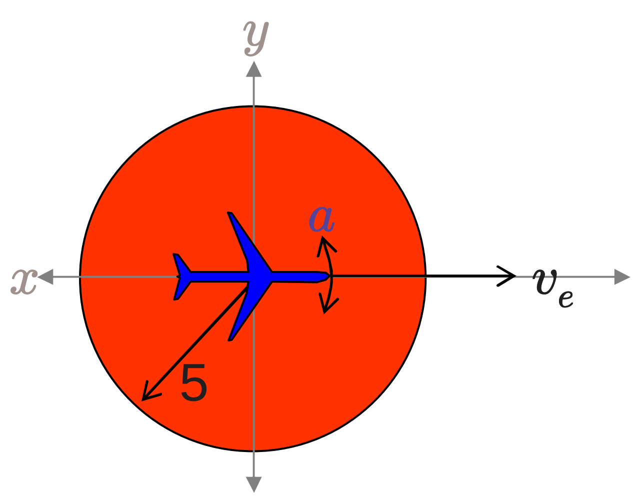
$$\forall u(\cdot), \exists d(\cdot), x(t) \in \mathcal{G}(0)$$

Game of Two Identical Vehicles

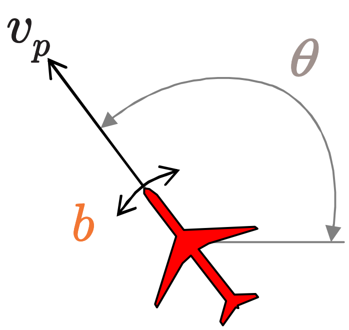
- Classical collision avoidance example
  - Collision occurs if vehicles get within five units of one another
  - Evader chooses turn rate $|a| \leq 1$ to avoid collision
  - Pursuer chooses turn rate $|b| \leq 1$ to cause collision
  - Fixed equal velocity $v_e = v_p = 5$

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)

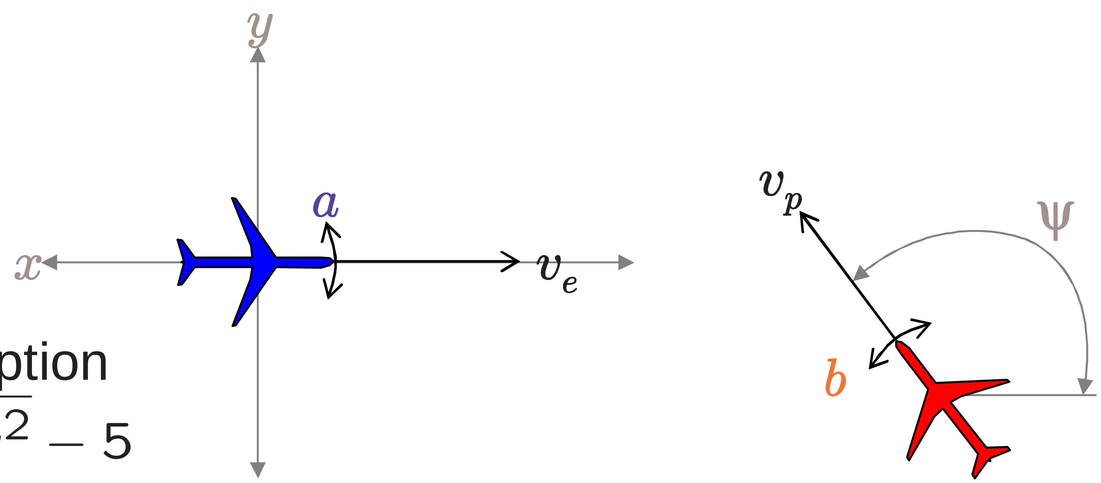


pursuer aircraft (disturbance)

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

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



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

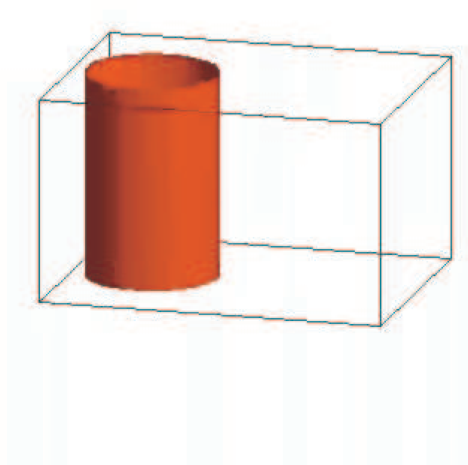
$$D_t\phi(x,t) + \min\left[0, H(x, D_x\phi(x,t))\right] = 0$$

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

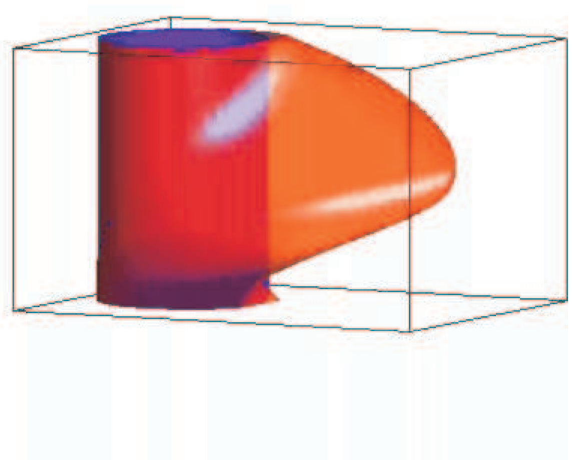
$$\text{and terminal conditions}: \quad \phi(x,0) = h(x)$$

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

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



growth of reachable set



final reachable set

Time-Dependent Hamilton-Jacobi Eq'n

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

- First order hyperbolic PDE
  - Solution can form kinks (discontinuous derivatives)
  - For the backwards reachable set, find the "viscosity" solution [Crandall, Evans, Lions, …]
- Level set methods
  - Convergent numerical algorithms to compute the viscosity solution [Osher, Sethian, …]
  - Non-oscillatory, high accuracy spatial derivative approximation
  - Stable, consistent numerical Hamiltonian
  - Variation diminishing, high order, explicit time integration

Solving a Differential Game

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

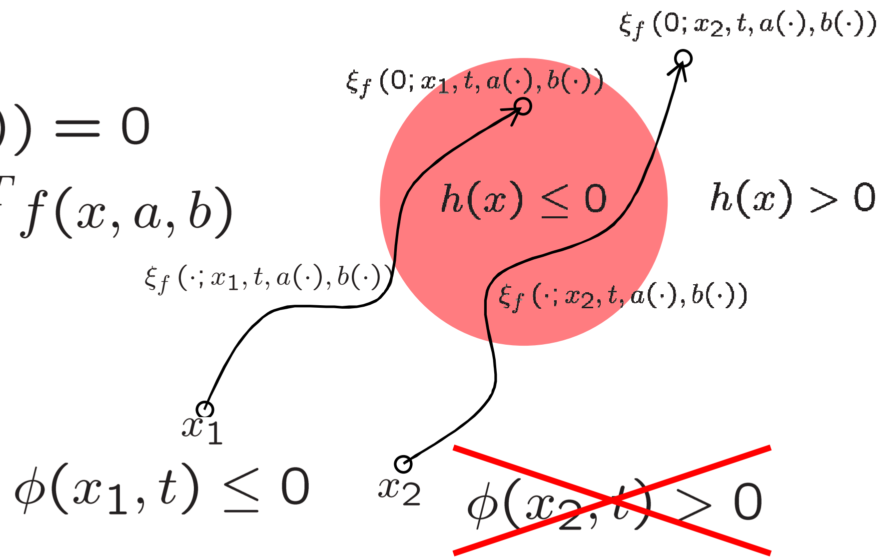
$$\phi(x,t) = \sup_{a(\cdot)} \inf_{b(\cdot)} h\left[\xi_f\,(0; x, t, a(\cdot), b(\cdot))\right]$$

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

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

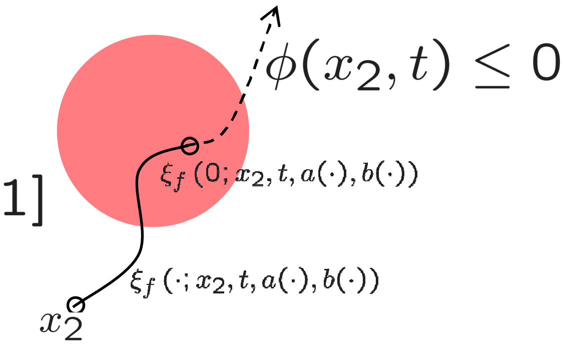
$$D_t\phi(x,t) + H\,(x, D_x\phi(x,t)) = 0$$

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



Modification for Optimal Stopping Time

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



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

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

  - 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

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

Alternative Eulerian Approaches

- Static Hamilton-Jacobi (Falcone, Sethian, …)
  - Minimum time to reach
  - (Dis)continuous implicit representation
  - Solution provides information on optimal input choices
- Viability kernels (Aubin, Saint-Pierre, …)
  - Based on set valued analysis for very general dynamics
  - Discrete implicit representation
  - Overapproximation guarantee
- Time-dependent Hamilton-Jacobi (this method)
  - Continuous solution
  - Information on optimal input choices available throughout entire state space
  - High order accurate approximations
- All three are theoretically equivalent

Forward Reachability

- Forwards reachable set is computed by following trajectories
- Examples:
  - Timed automata: Uppaal [Larsen, Pettersson…], Kronos [Yovine,…], …
  - Rectangular differential inclusions: Hytech, Hypertech [Henzinger, Ho, Horowitz, Wong-Toi, …]
  - Polyhedra and linear dynamics: Checkmate [Chutinan & Krogh], d/dt [Bournez, Dang, Maler, Pnueli, …], others [Bemporad, Morari, Torrisi, …], [Greenstreet & Mitchell], …
  - Ellipsoids and linear dynamics [Botchkarev, Kurzhanski, Tripakis, Varaiya, …]
  - Discretization (predicate abstraction) on grid [Kurshan & McMillan] or by cylindrical algebraic decomposition [Tiwari & Khanna]
- Advantages: Compact representation of sets, overapproximation guarantees
- Disadvantages: Linear dynamics, reliance on trajectory optimization, restrictive set representation, potentially large error

State Space Decomposition

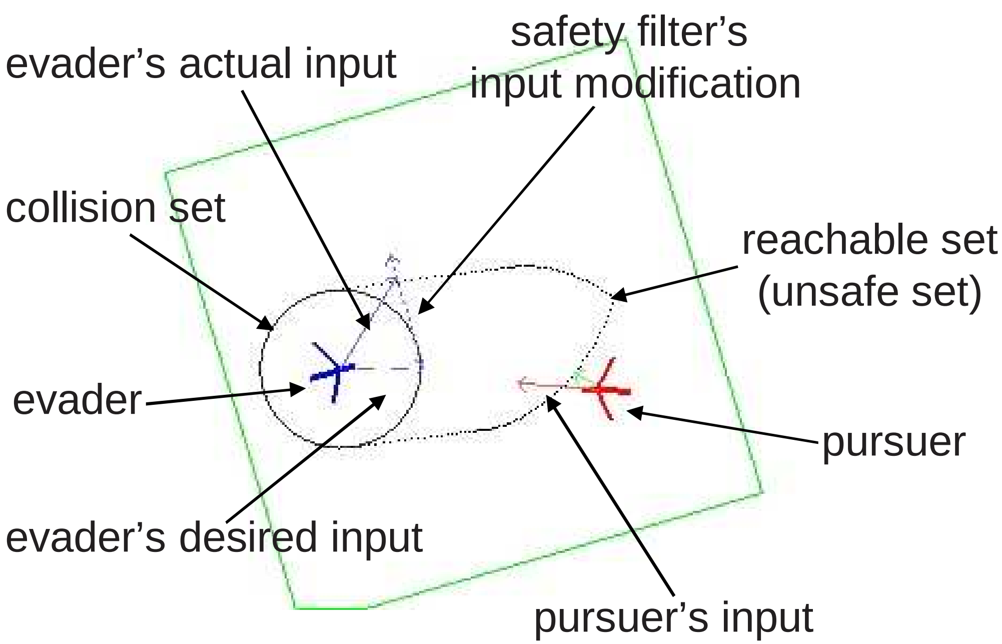
- Partition state space and compute reachability over partition
- Examples
  - Uniform grids: Kurshan & MacMillan and many others
  - Timed Automata “Region Graph”: Alur & Dill
  - Cylindrical Algebraic Decomposition: Tiwari & Khanna
- Advantages: No need to integrate dynamics, direct control over size of representation
- Disadvantages: Restricted classes of dynamics, “wrapping” problem (discrete system has transitions that do not exist in continuous system)

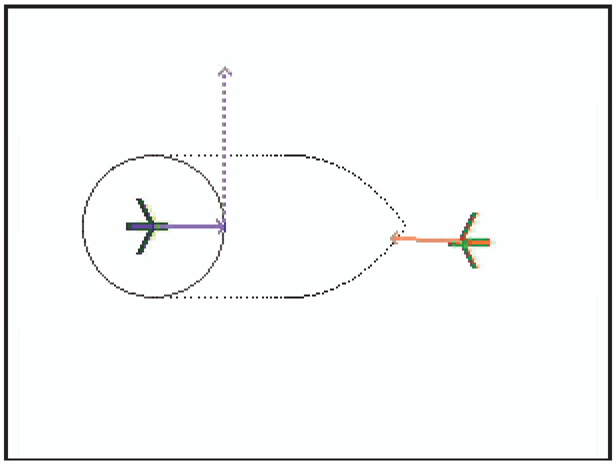
Lyapunov-like Methods

- Invariant sets are isosurfaces of Lyapunov-like functions
- Examples:
  - Convex optimization: Boyd, Hindi, Hassibi
  - Sum of Squares: Prajna, Papachristodoulou, Parrilo
- Advantages: Short certificate proves analytic invariance, no need to integrate dynamics
- Disadvantages: Restricted class of dynamics, difficult to extract counterexamples

Application: Softwalls for Aircraft Safety

- 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

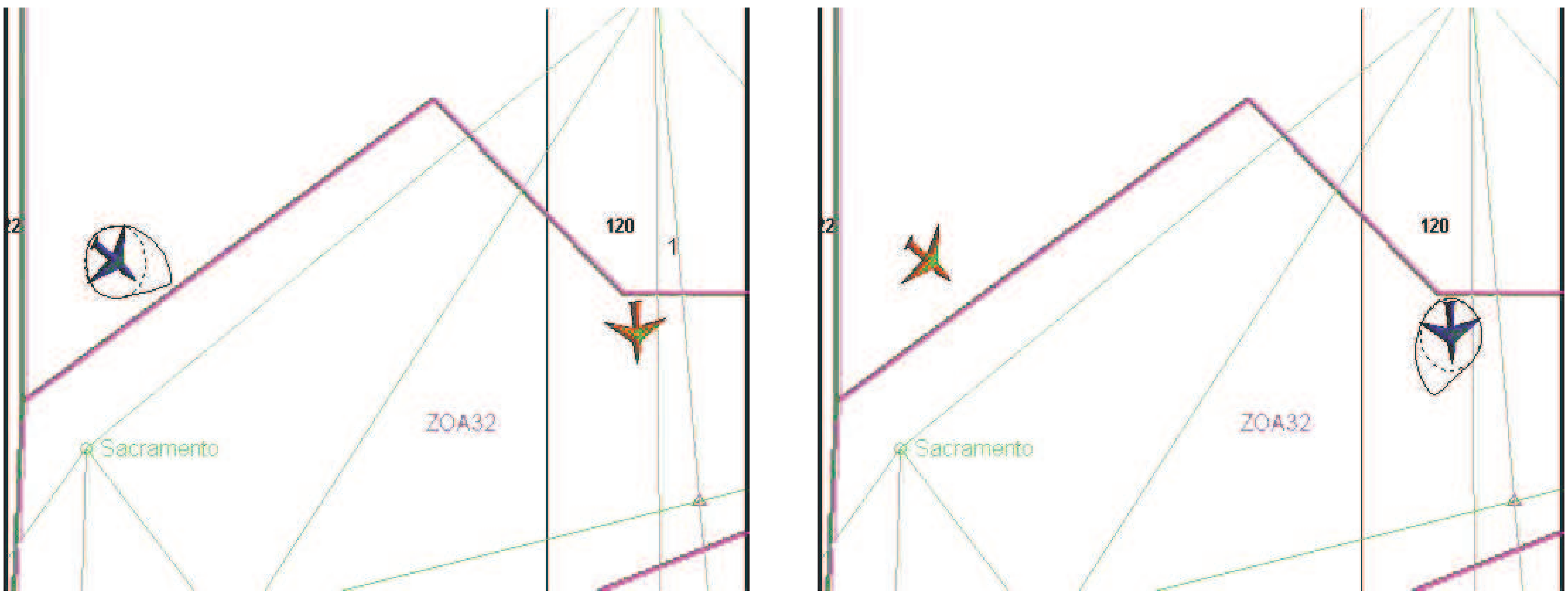




joint work with Edward Lee & Adam Cataldo

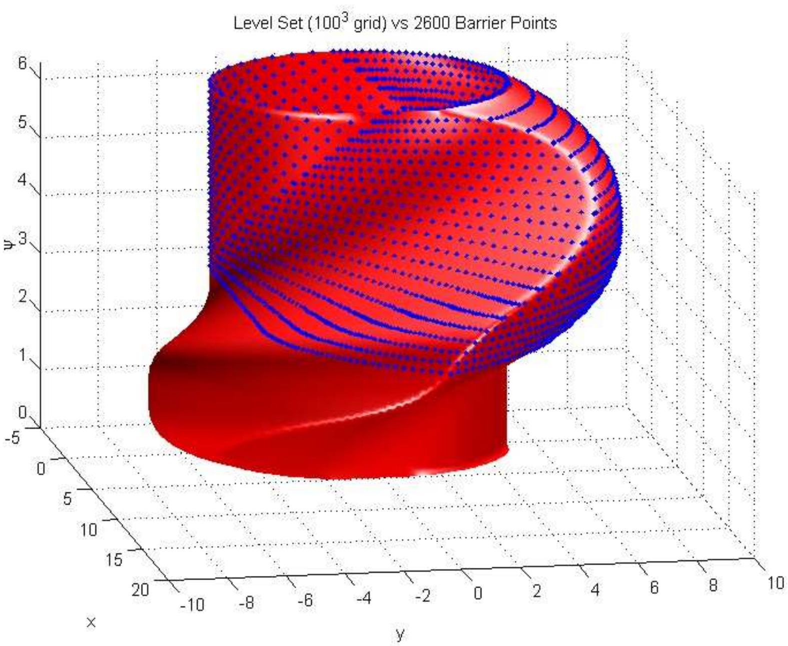
Application: Collision Alert for ATC

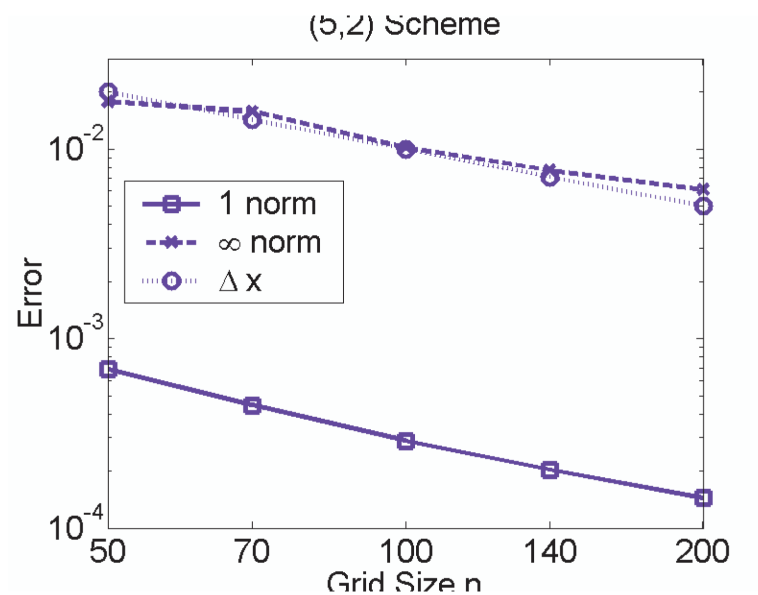
- Use reachable set to detect potential collisions and warn Air Traffic Control (ATC)
  - Find aircraft pairs in ETMS database whose flight plans intersect
  - Check whether either aircraft is in the other's collision region
  - If so, examine ETMS data to see if aircraft path is deviated
  - One hour sample in Oakland center's airspace—
    - 1590 pairs, 1555 no conflict, 25 detected conflicts, 2 false alerts



Validating the Numerical Algorithm

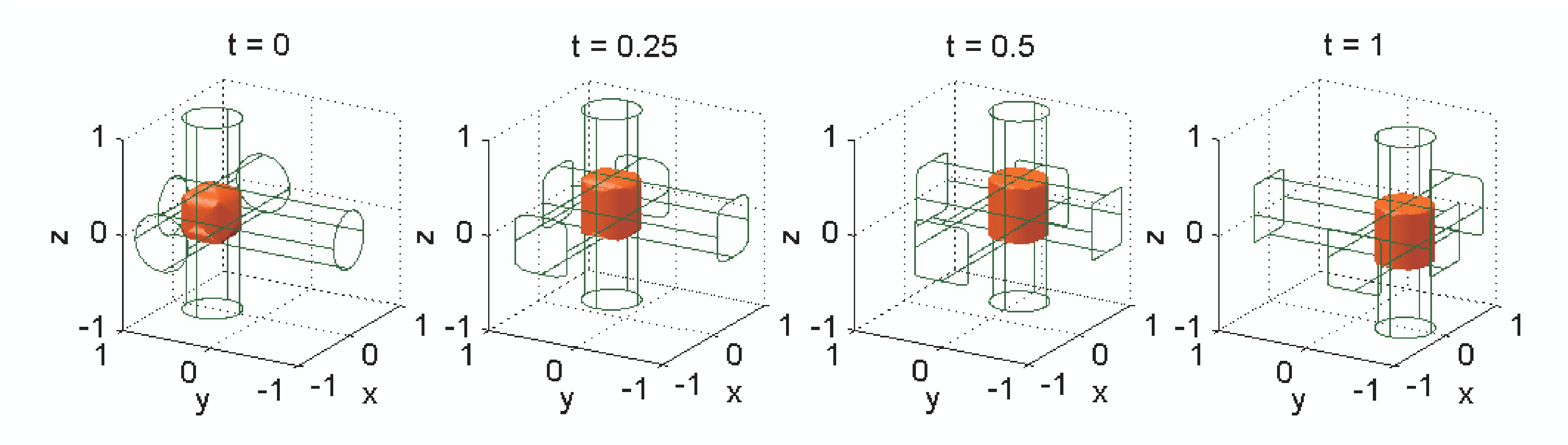
- Analytic solution for reachable set can be found [Merz, 1972]
  - Applies only to identical pursuer and evader dynamics
  - Merz's solution placed pursuer at the origin, game is not symmetric
  - Analytic solution can be used to validate numerical solution
  - [Mitchell, 2001]





Projective Overapproximation

- Overapproximate reachable set of high dimensional system as the intersection of reachable sets for lower dimensional projections
  - [Mitchell & Tomlin, JSC 2003]
  - Example: rotation of “sphere” about z-axis



Hamilton-Jacobi in the Projection

- Consider x–z projection represented by level set $\phi_{xz}(x,z,t)$
  - Back projection into 3D yields a cylinder $\phi_{xz}(x,y,z,t)$
- Simple HJ PDE for this cylinder

$$D_t \phi_{xz}(x,y,z,t) + \sum_{i=1}^{3} p_i f_i(x,y,z) = 0 \quad \text{where} \begin{cases} p_1 = D_x \phi_{xz}(x,y,z,t) \\ p_2 = D_y \phi_{xz}(x,y,z,t) \\ p_z = D_z \phi_{xz}(x,y,z,t) \end{cases}$$

  - But for cylinder parallel to y-axis, $p_2 = 0$

$$D_t \phi_{xz}(x,y,z,t) + p_1 f_1(x,y,z) + p_3 f_3(x,y,z) = 0$$

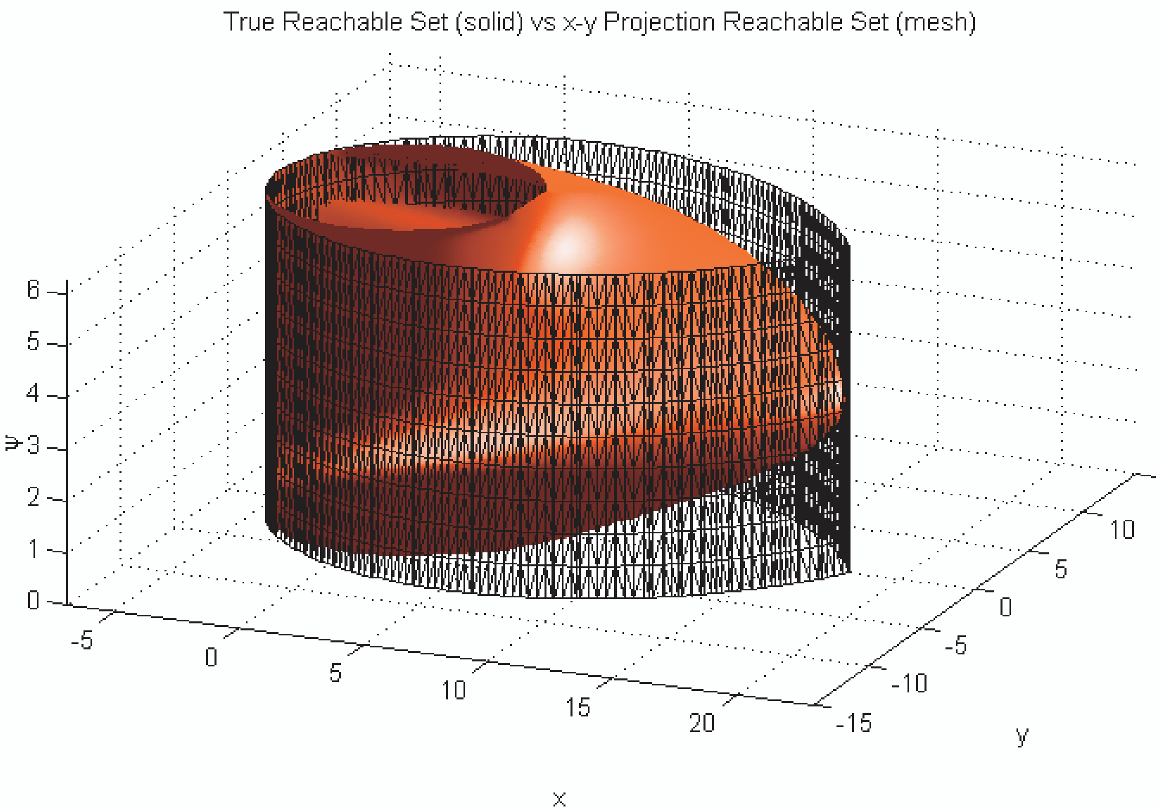
- What value to give free variable y in $f_i(x,y,z)$?
  - Treat it as a disturbance, bounded by the other projections

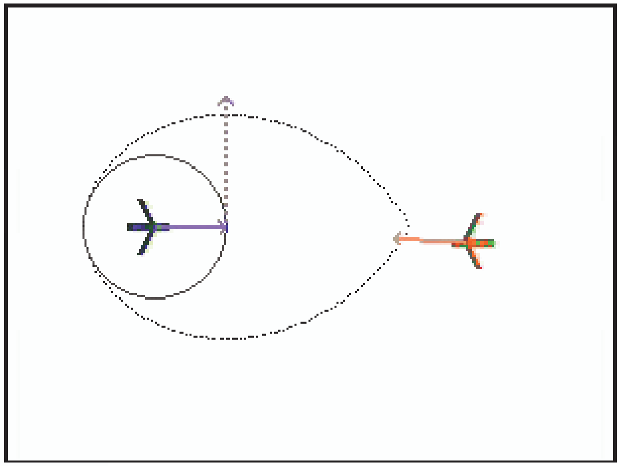
$$D_t \phi_{xz}(x,y,z,t) + \min_y \left[p_1 f_1(x,y,z) + p_3 f_3(x,y,z) \right] = 0$$

- Hamiltonian no longer depends on y, so computation can be done entirely in x–z space on $\phi_{xz}(x,z,t)$

Projective Collision Avoidance

- Work strictly in relative x–y plane
  - Treat relative heading $\psi \in [0, 2\pi]$ as a disturbance input
  - Compute time: 40 seconds in 2D vs 20 minutes in 3D
  - Compare overapproximative prism (mesh) to true set (solid)

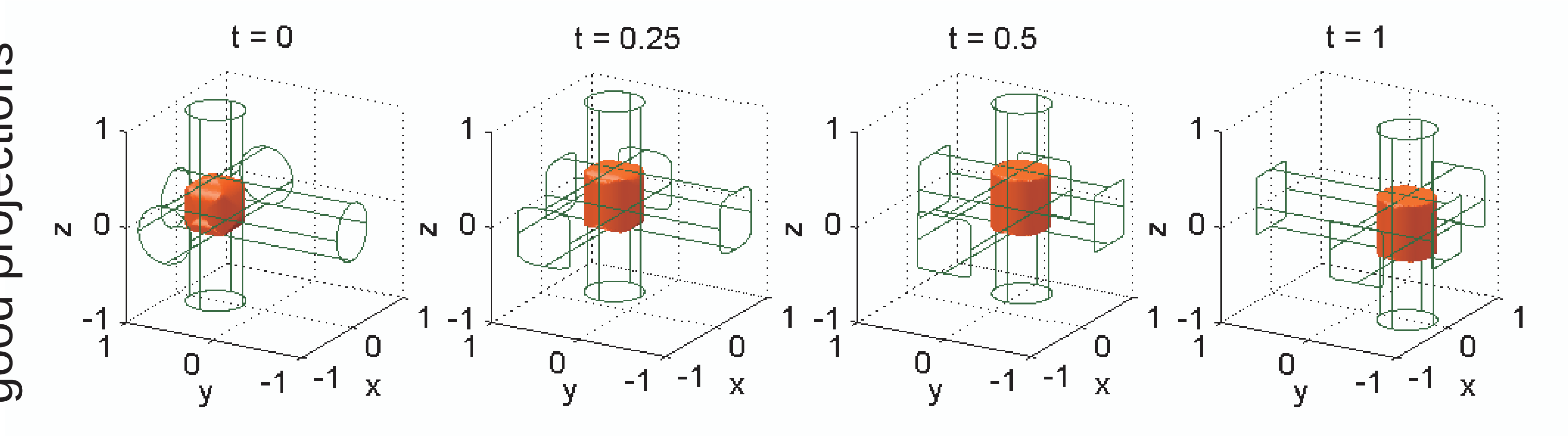




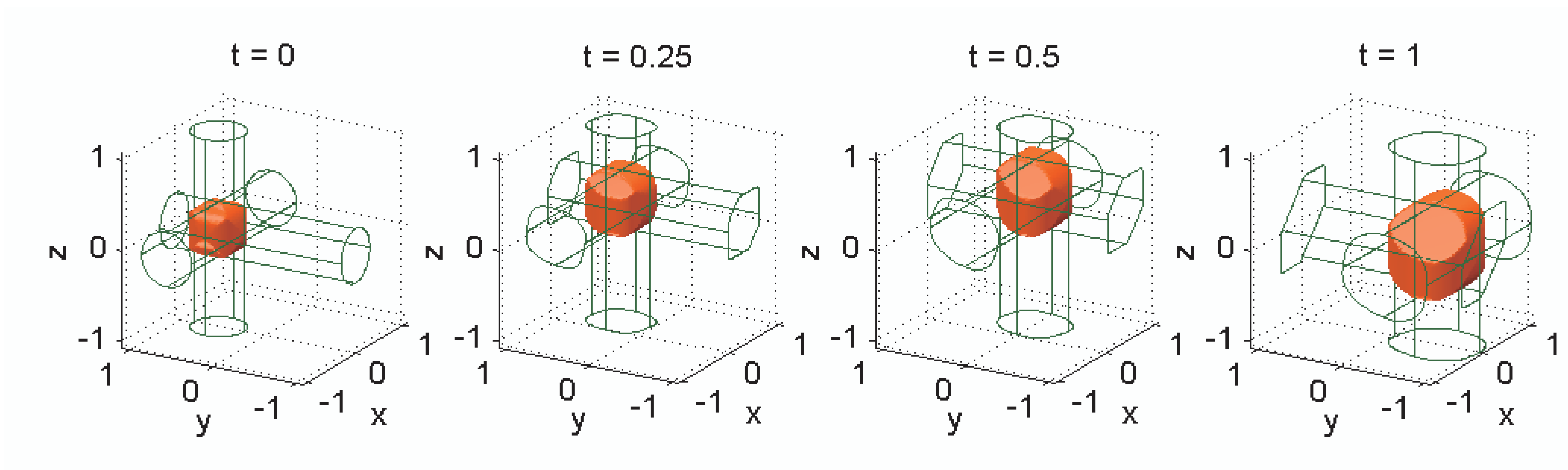
Projection Choices

- Poorly chosen projections may lead to large overapproximations
  - Projections need not be along coordinate axes
  - Number of projections is not constrained by number of dimensions

good projections



poor projections



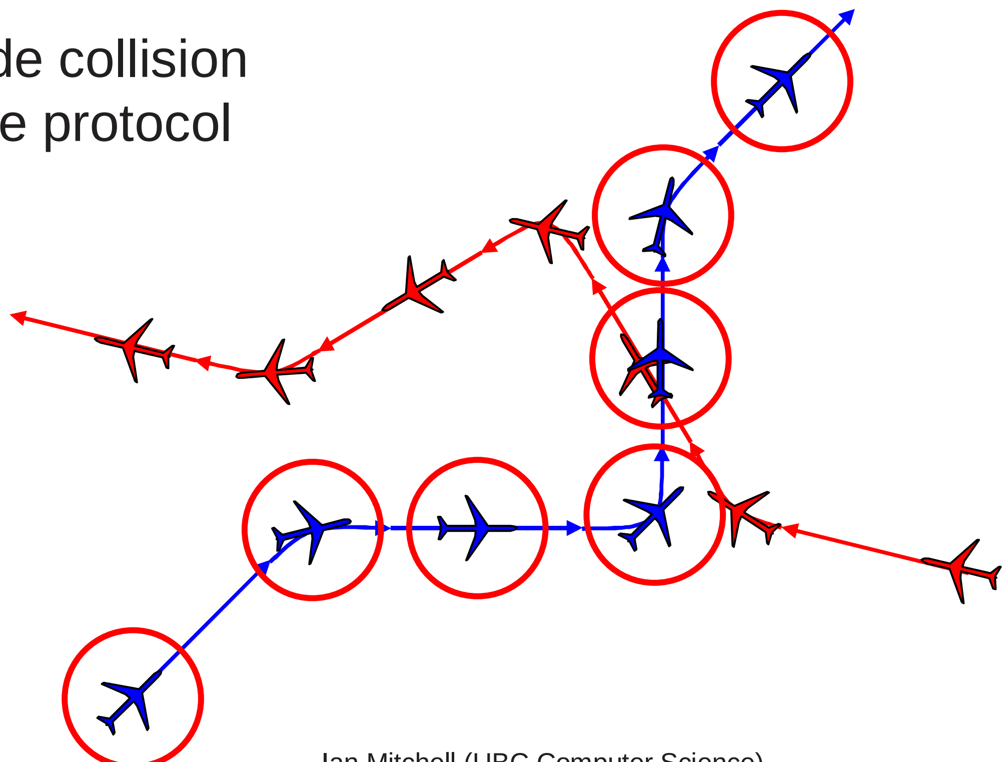
Hybrid System Reach Sets

Combining Continuous and Discrete
Evolution

Why Hybrid Systems?

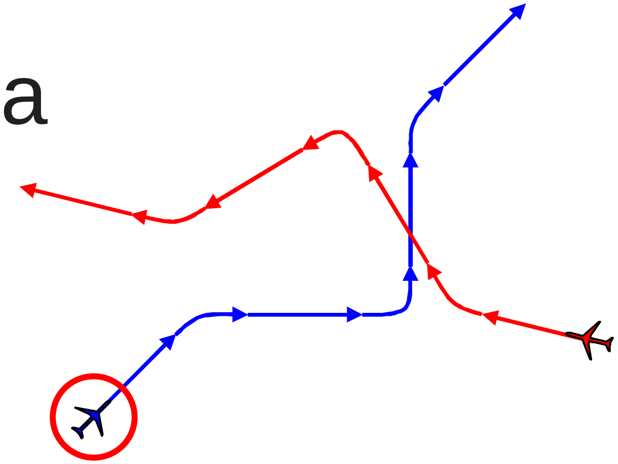
- Computers are increasingly interacting with external world
  - Flexibility of such combinations yields huge design space
  - Design methods and tools targeted (mostly) at either continuous or discrete systems
- Example: aircraft flight control systems

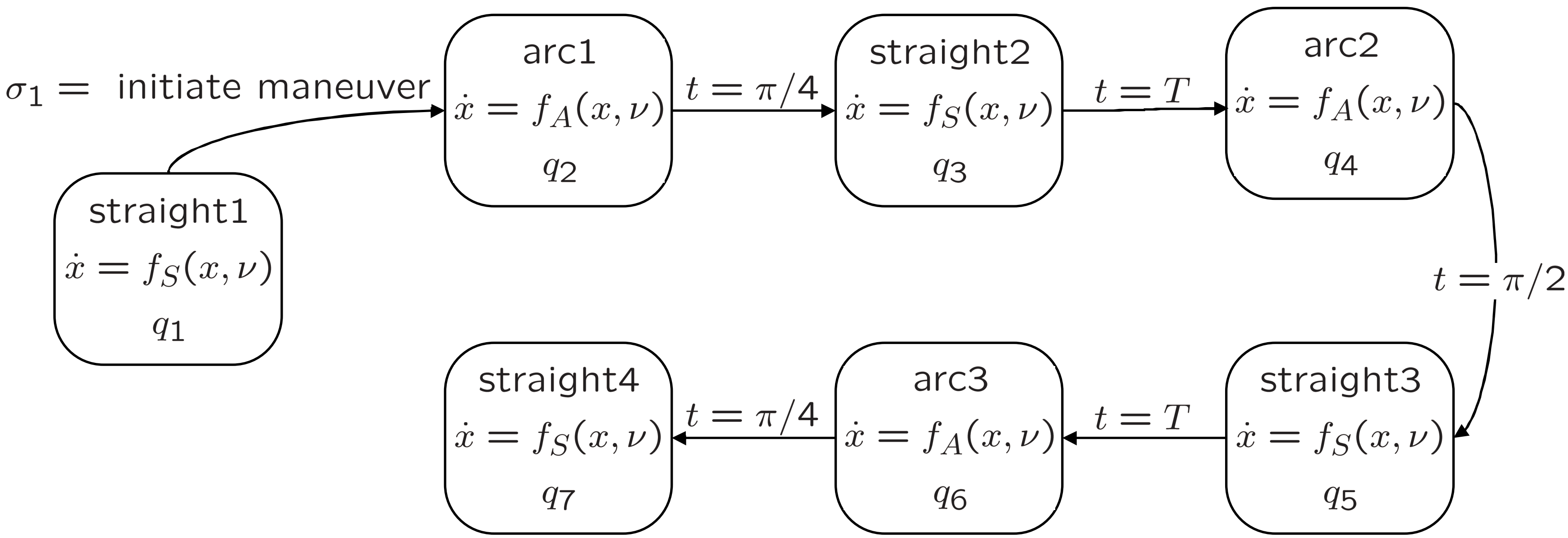
seven mode collision avoidance protocol



Hybrid Automata

- Discrete modes and transitions
- Continuous evolution within each mode





$$f_S \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -v + v\cos\psi \\ v\sin\psi \end{bmatrix}$$

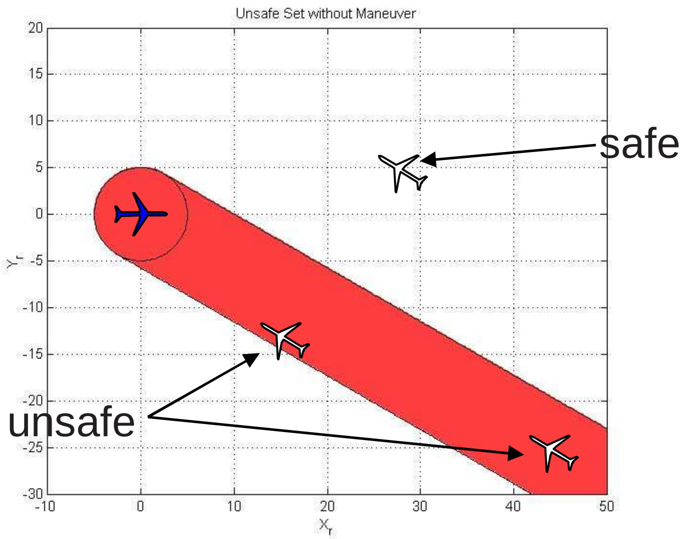
dynamics in straight modes

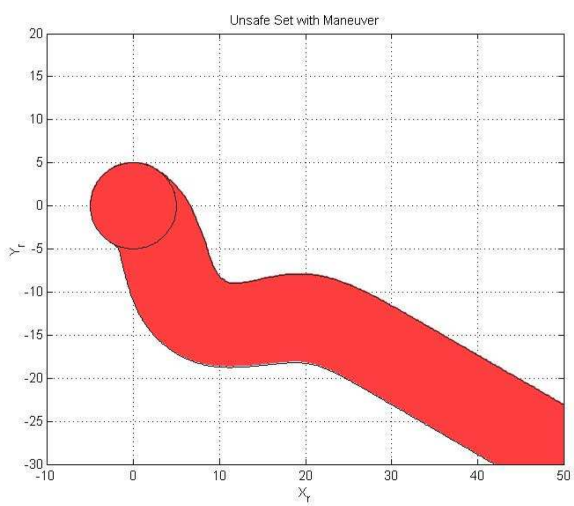
$$f_A \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -v + v\cos\psi - x_2 \\ v\sin\psi + x_1 \end{bmatrix}$$

dynamics in arc modes

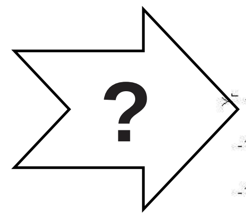
Seven Mode Safety Analysis

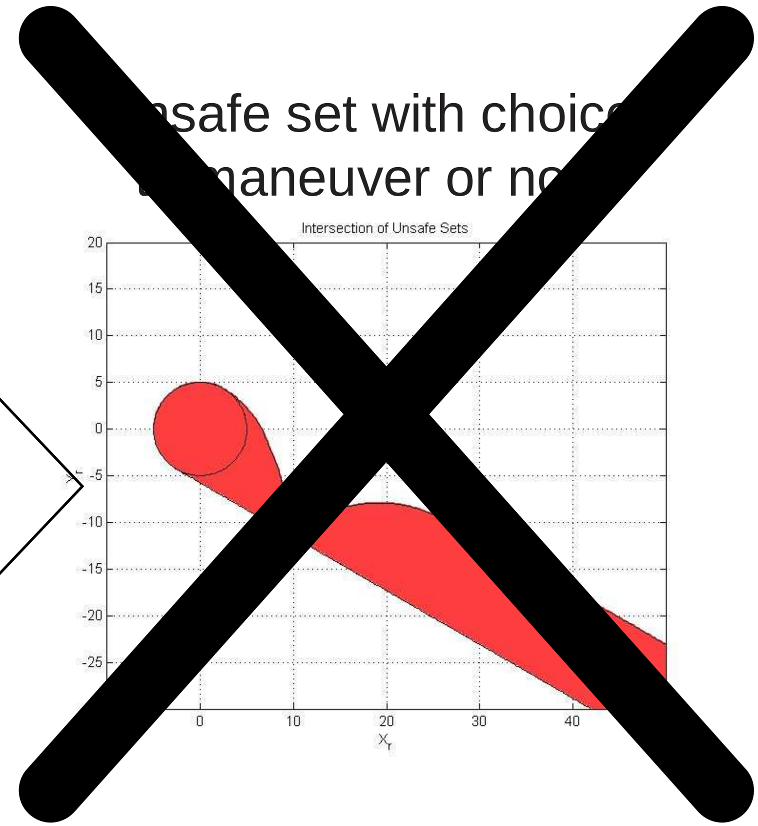
unsafe set without maneuver





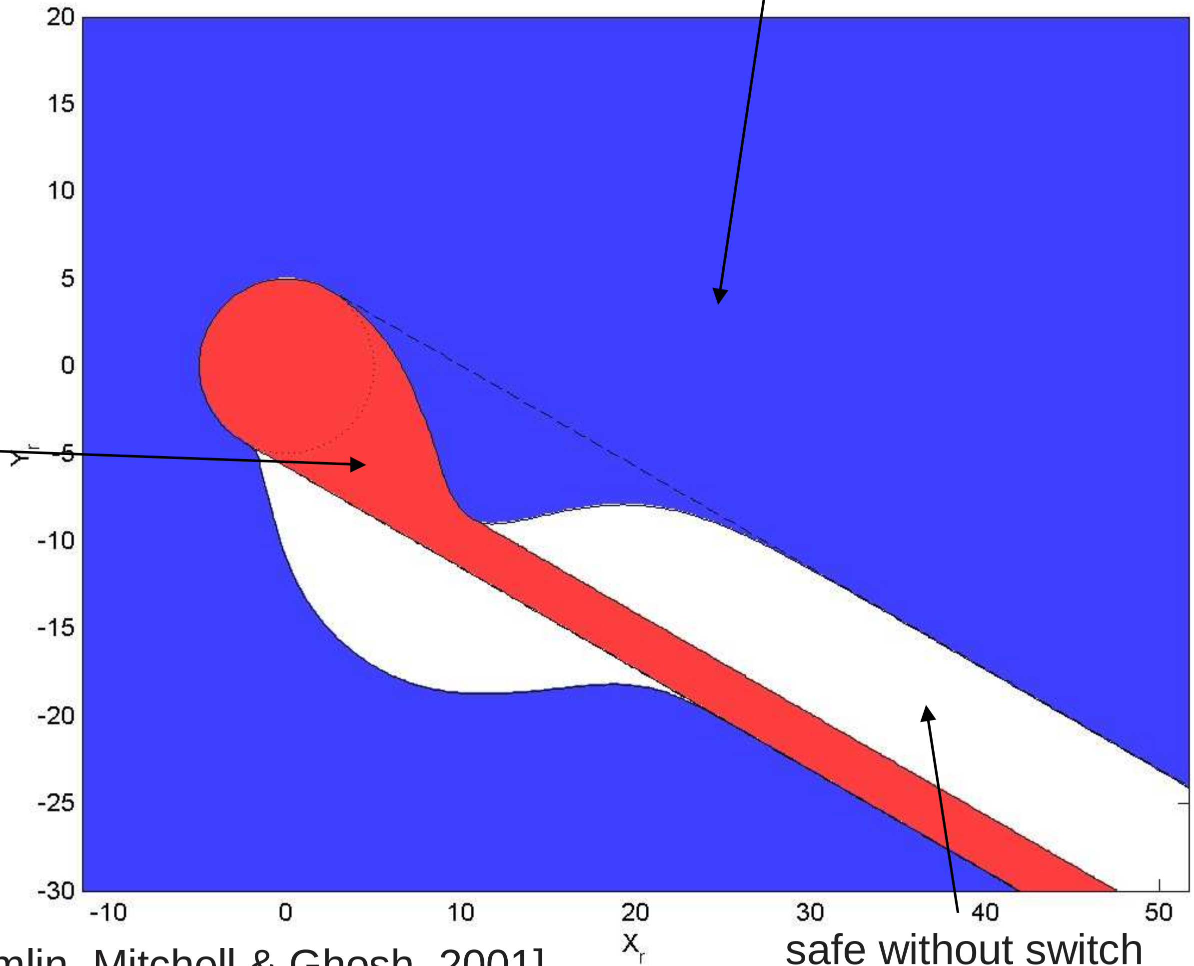
unsafe set with maneuver





Seven Mode Safety Analysis

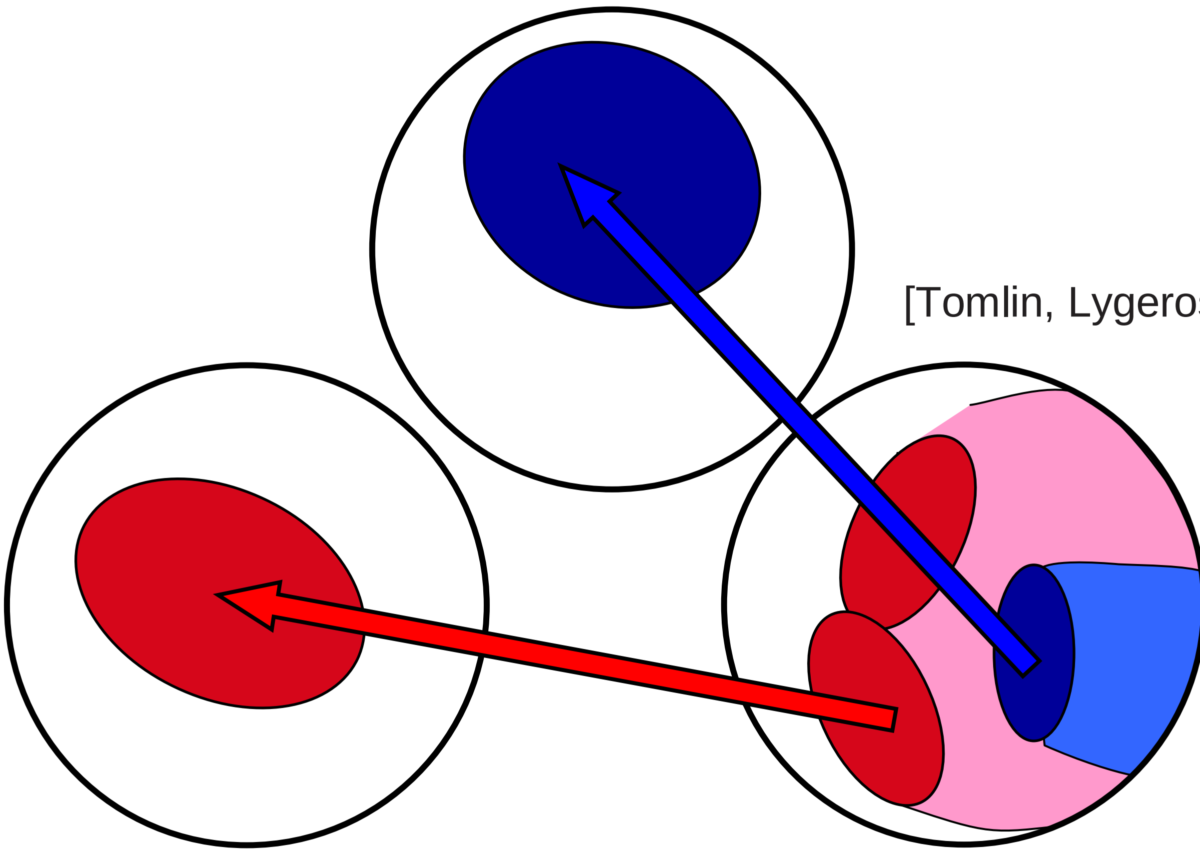
- Ability to choose maneuver start time further reduces unsafe set



[Tomlin, Mitchell & Ghosh, 2001]

Computing Hybrid Reachable Sets

- Compute continuous reachable set in each mode separately
  - Uncontrollable switches may introduce unsafe sets
  - Controllable switches may introduce safe sets
  - Forced switches introduce boundary conditions

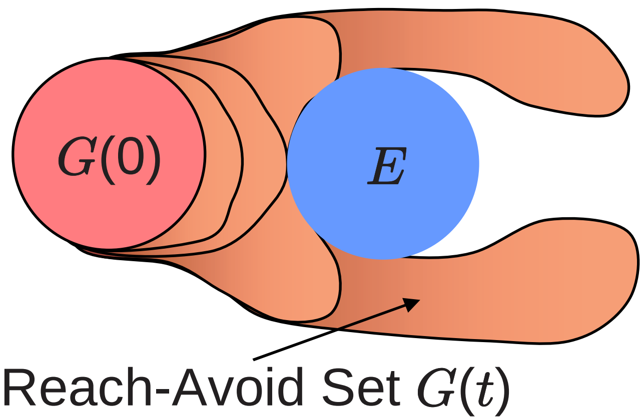


[Tomlin, Lygeros & Sastry, 2000]

Reach-Avoid Operator

- Compute set of states which reaches $G(0)$ without entering E

$$G(t) = \{x \in \mathbb{R}^n \mid \phi_G(x,t) \le 0\}$$
$$E = \{x \in \mathbb{R}^n \mid \phi_E(x) \le 0\}$$



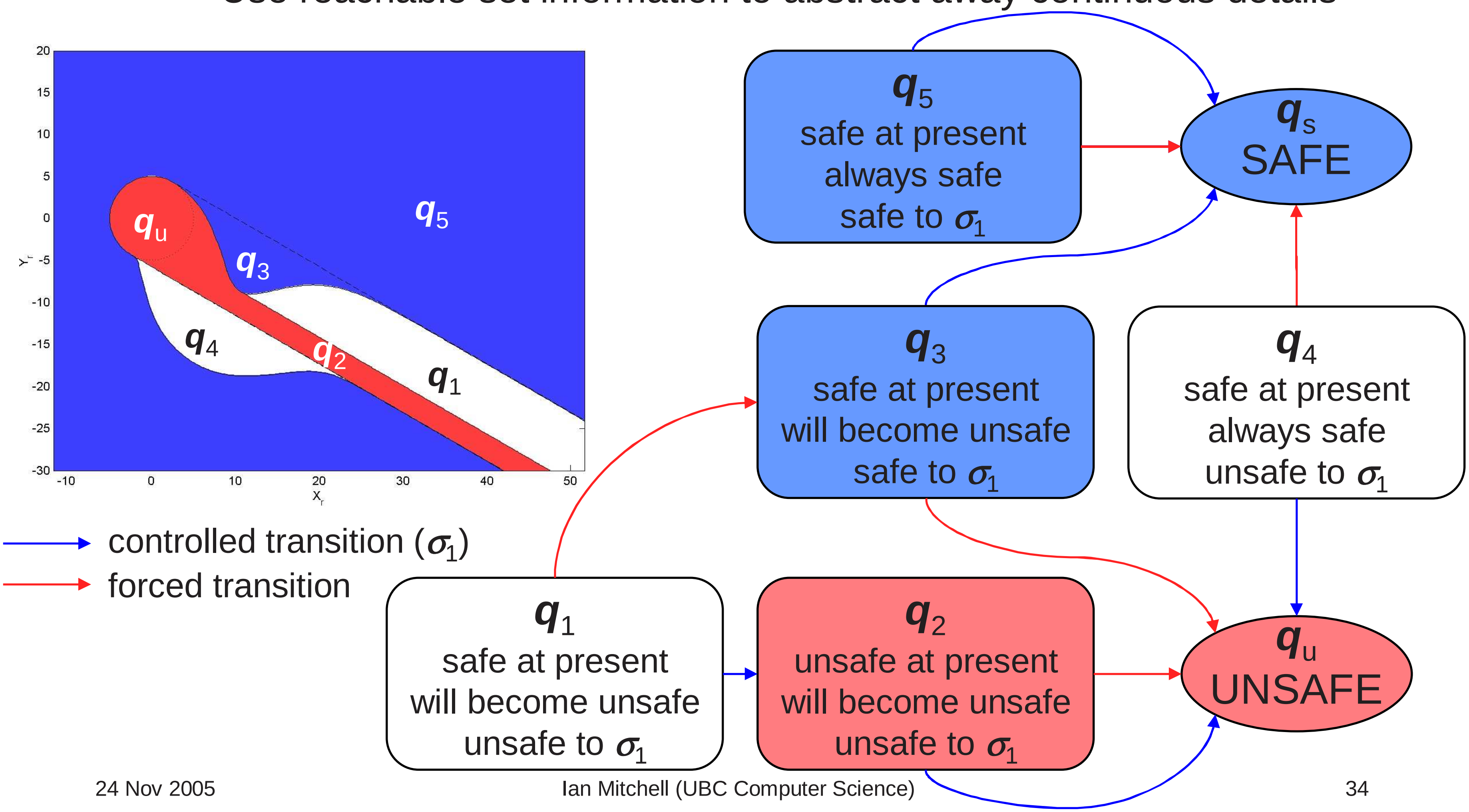
- Formulated as a constrained Hamilton-Jacobi equation or variational inequality
  - [Mitchell & Tomlin, 2000]

$$D_t\phi_G(x,t) + \min\left[0, H(x, D_x\phi_G(x,t))\right] = 0$$
$$\text{subject to: } \phi_G(x,t) \ge \phi_E(x)$$

- Level set can represent often odd shape of reach-avoid sets

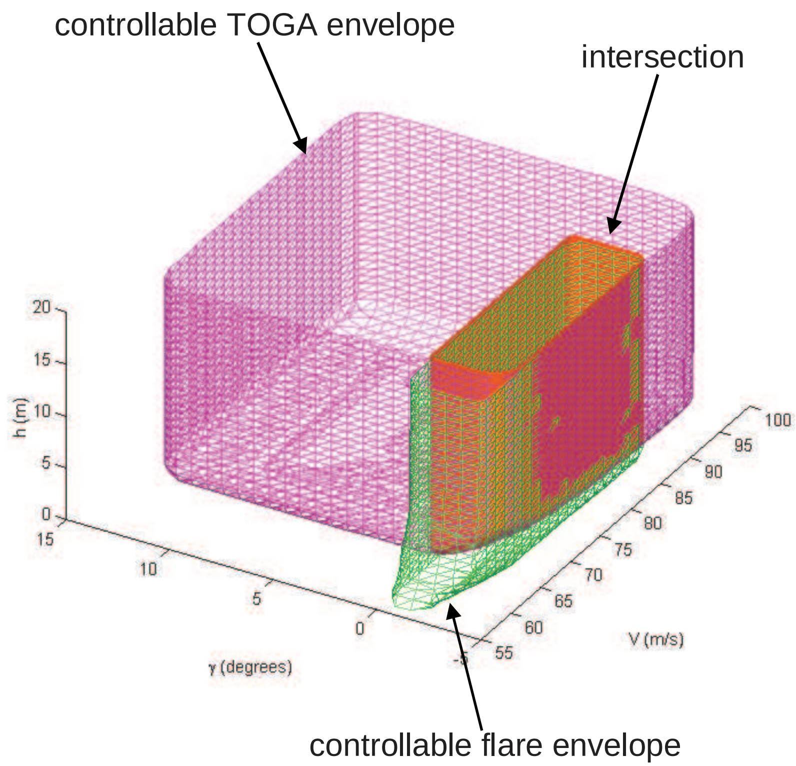
Application: Discrete Abstractions

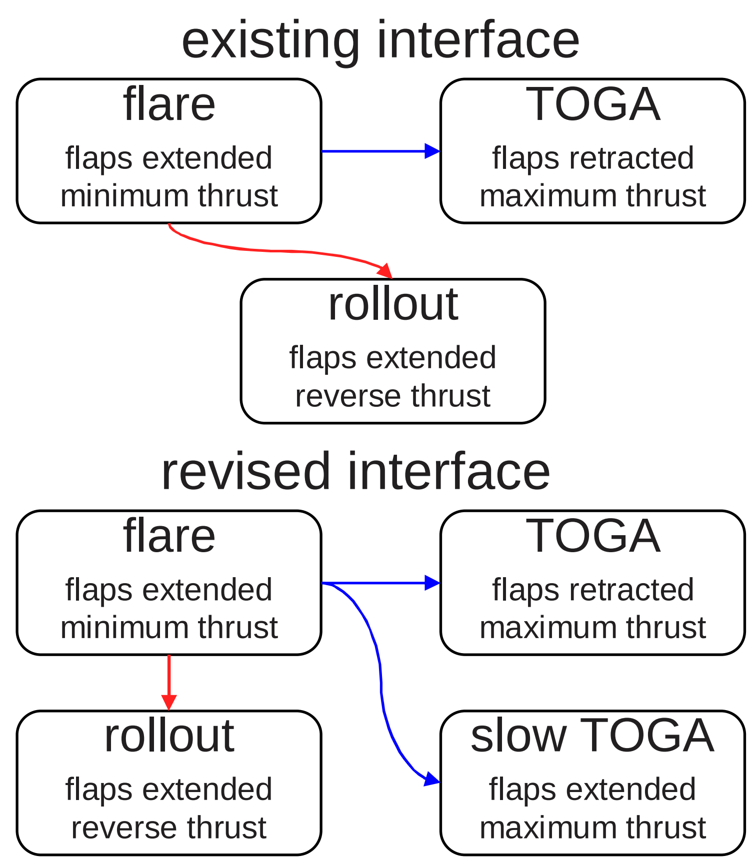
- It can be easier to analyze discrete automata than hybrid automata or continuous systems
  - Use reachable set information to abstract away continuous details



Application: Cockpit Display Analysis

- Controllable flight envelopes for landing and Take Off / Go Around (TOGA) maneuvers may not be the same
- Pilot's cockpit display may not contain sufficient information to distinguish whether TOGA can be initiated

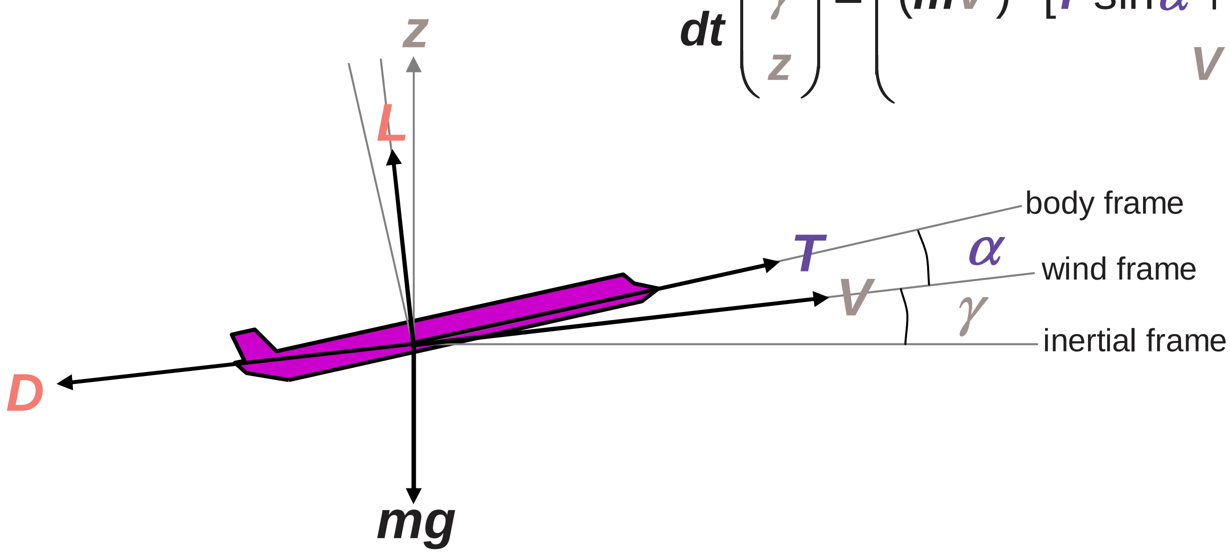




Application: Aircraft Autolander

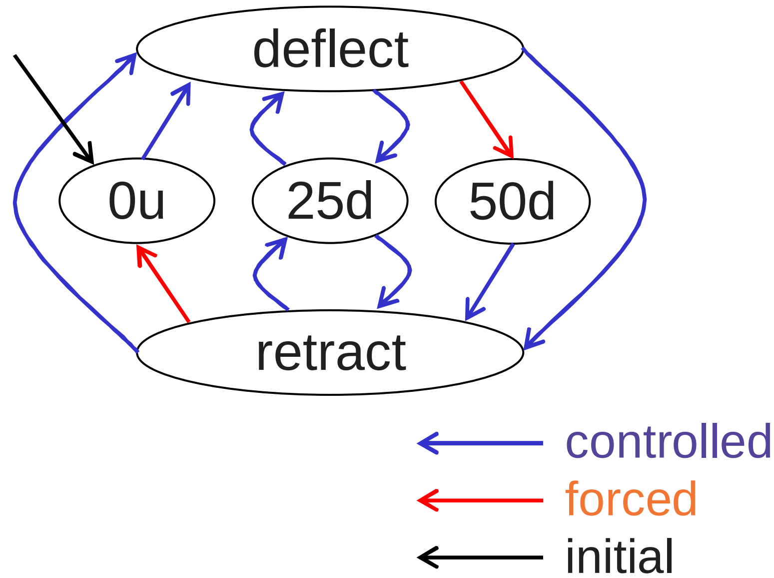
- Airplane must stay within safe flight envelope during landing
  - Bounds on velocity (V), flight path angle (γ), height (z)
  - Control over engine thrust (T), angle of attack (α), flap settings
  - Model flap settings as discrete modes of hybrid automata
  - Terms in continuous dynamics may depend on flap setting
  - [Mitchell, Bayen & Tomlin, 2001]

$$\frac{d}{dt}\begin{pmatrix} V \\ \gamma \\ z \end{pmatrix} = \begin{pmatrix} m^{-1}[T\cos\alpha - D(\alpha, V) - mg\sin\gamma] \\ (mV)^{-1}[T\sin\alpha + L(\alpha, V) - mg\cos\gamma] \\ V\sin\gamma \end{pmatrix}$$

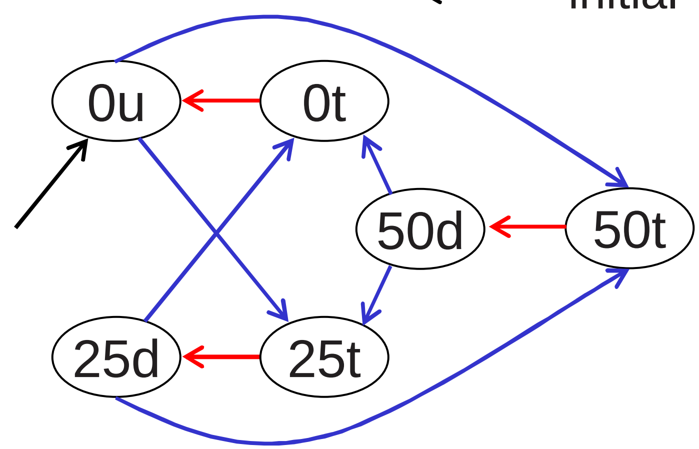


Landing Example: Discrete Model

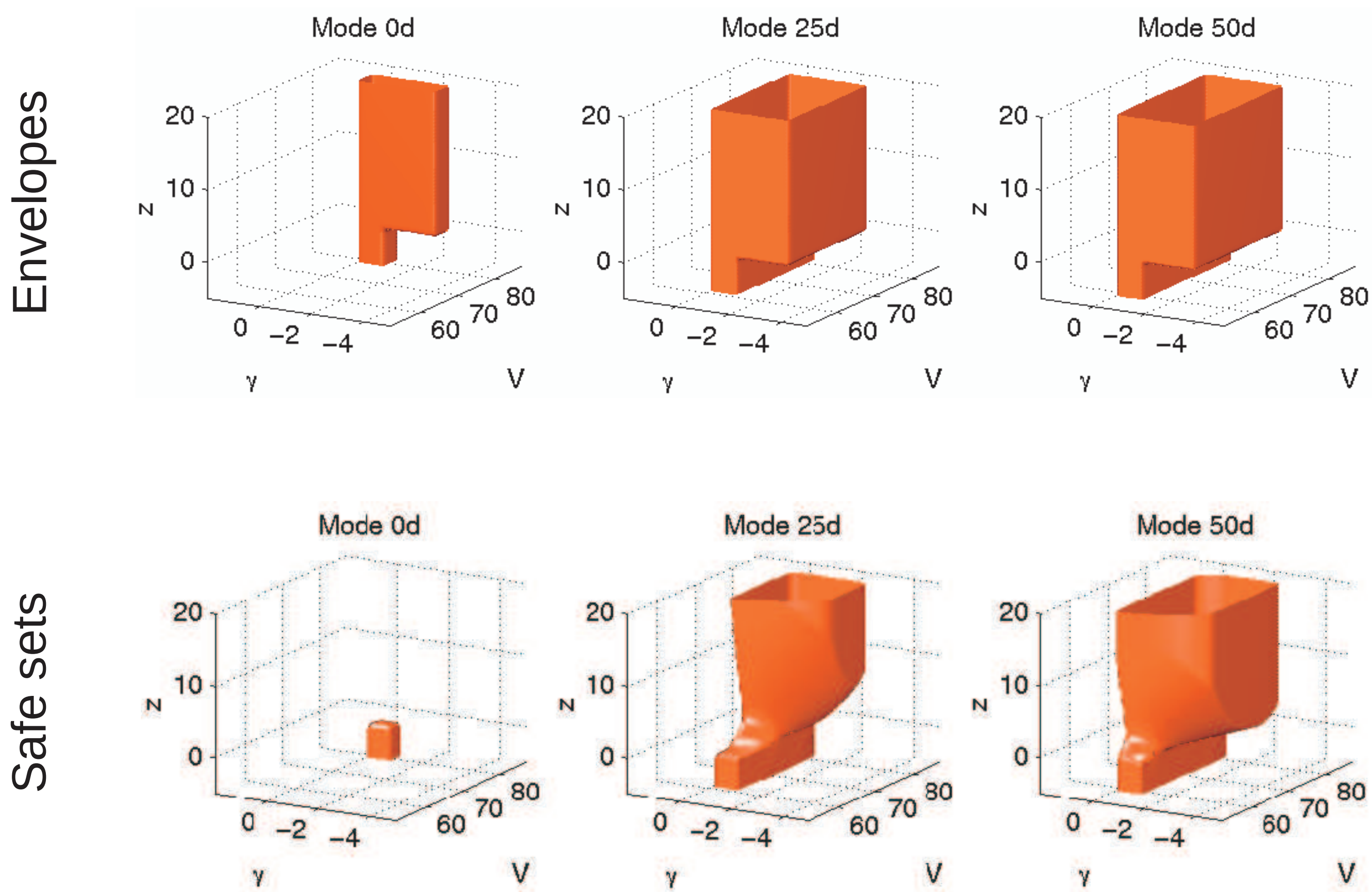
- Flap dynamics version
  - Pilot can choose one of three flap deflections
  - Thirty seconds for zero to full deflection



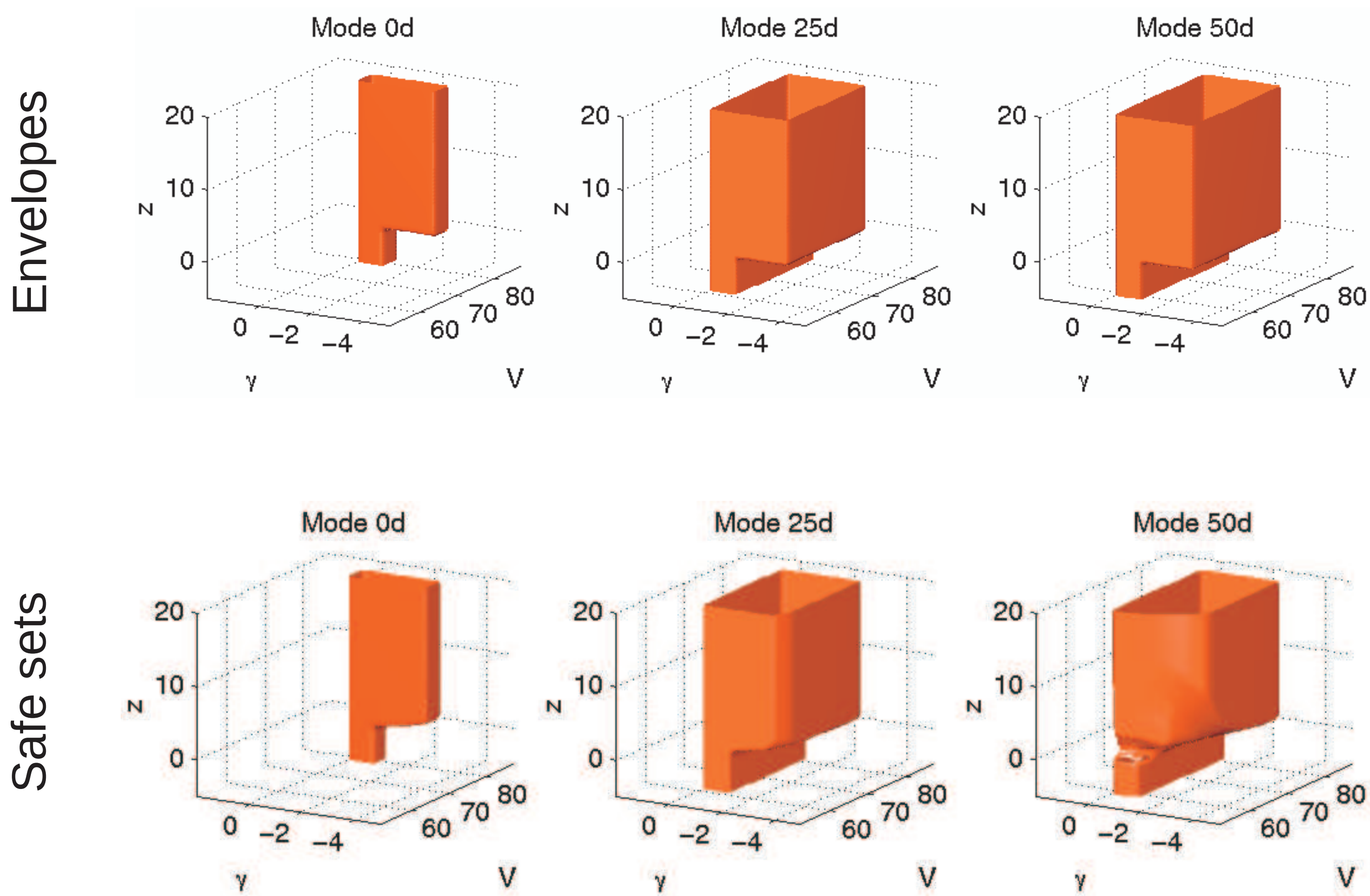
- Implemented version
  - Instant switches between fixed deflections
  - Additional timed modes to remove Zeno behavior



Landing Example: No Mode Switches

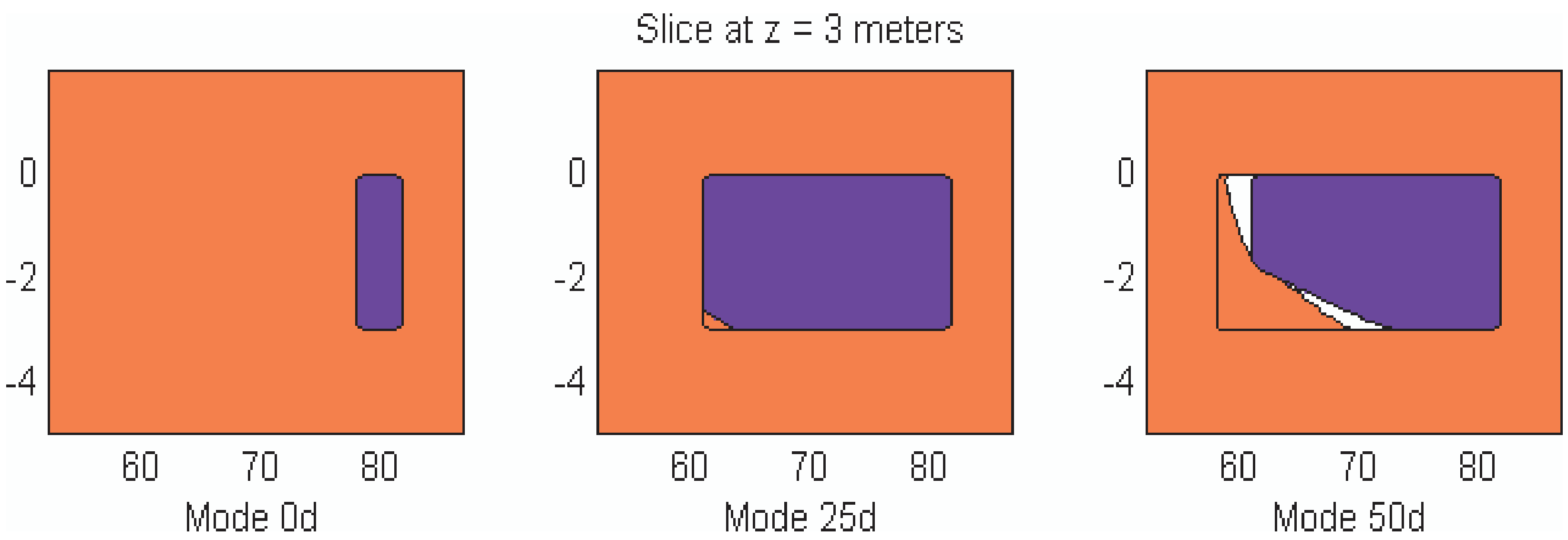


Landing Example: Mode Switches



Landing Example: Synthesizing Control

- For states at the boundary of the safe set, results of reach-avoid computation determine
  - What continuous inputs (if any) maintain safety
  - What discrete jumps (if any) are safe to perform
  - Level set values & gradients provide all relevant data



Viability Theory

An Alternative Approach Based on Set Valued Analysis

Differential Inclusions

- Dynamics defined by differential inclusion

$$\frac{dx}{dt} \in \mathcal{F}(x), \qquad \mathcal{F}(x) : \mathbb{R}^n \to \mathcal{P}(\mathbb{R}^n)$$

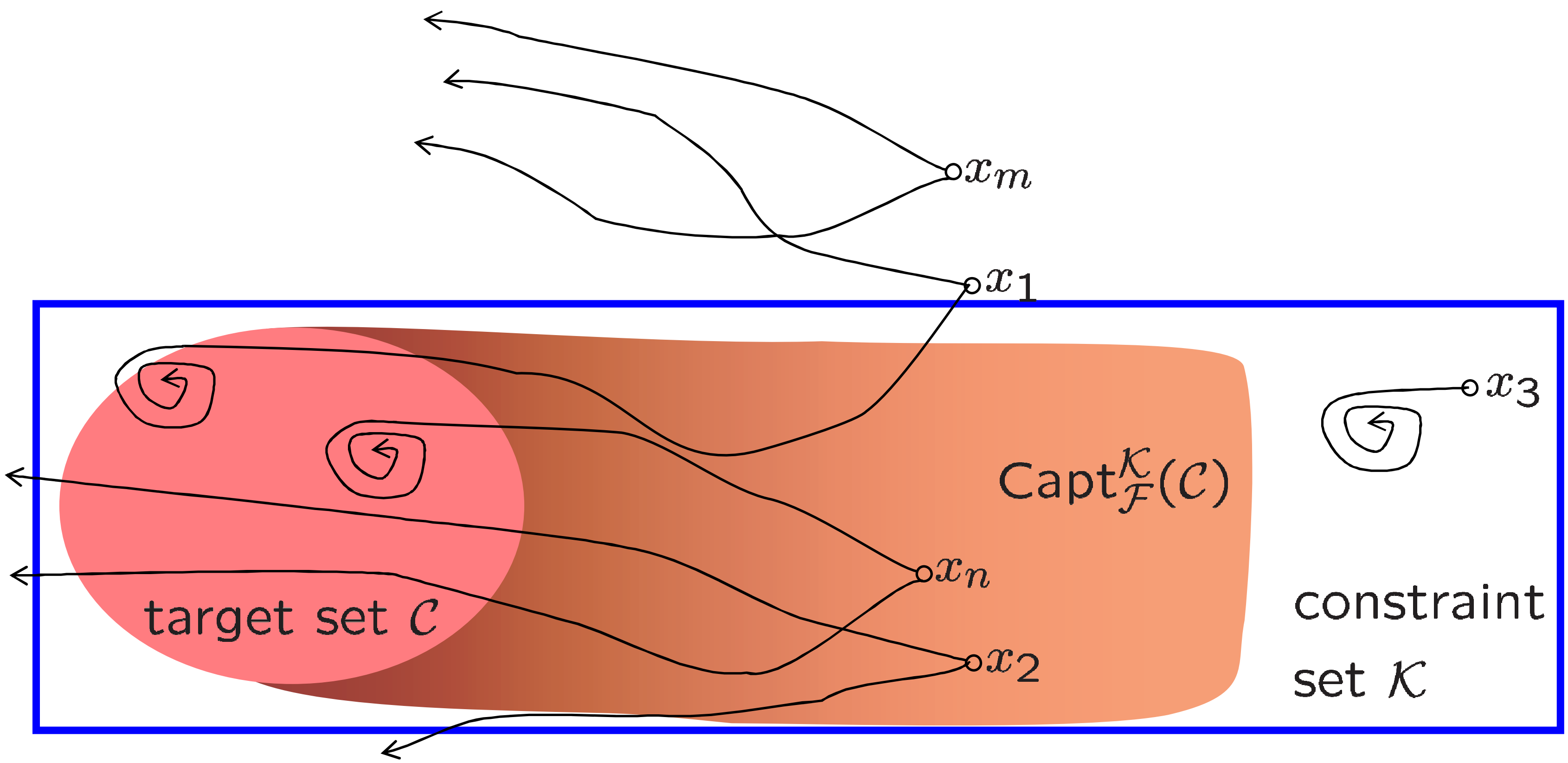
  – For example

$$\mathcal{F}(x) = \{y \in \mathbb{R}^n \mid \exists b \in \mathcal{B}, y = f(x, b)\}$$

- Set-valued map $\mathcal{F}(x)$ has Lipschitz-like but less restrictive conditions
  – For example, discontinuous $f(x,b)$ can be represented
- Extensions exist for differential game settings

Capture Basin

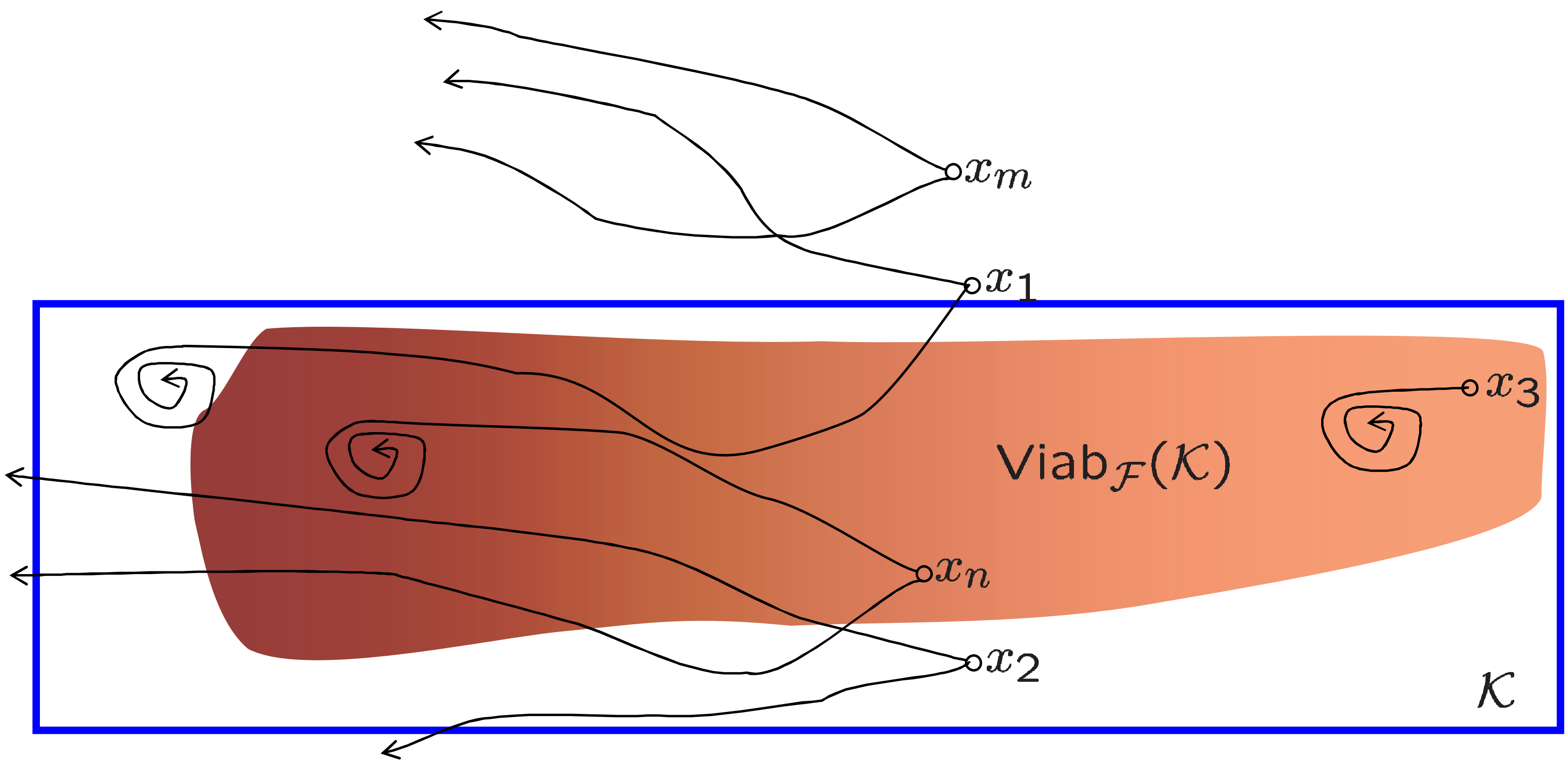
$$\text{Capt}_{\mathcal{F}}^{\mathcal{K}}(\mathcal{C}) = \{x_i \mid \exists t > 0, x(t) \in \mathcal{C} \wedge \forall s \in [0,t], x(s) \in \mathcal{K}\}$$
$$\text{where } \dot{x} \in \mathcal{F}(x) \text{ and } x(0) = x_i$$



Viability Kernel

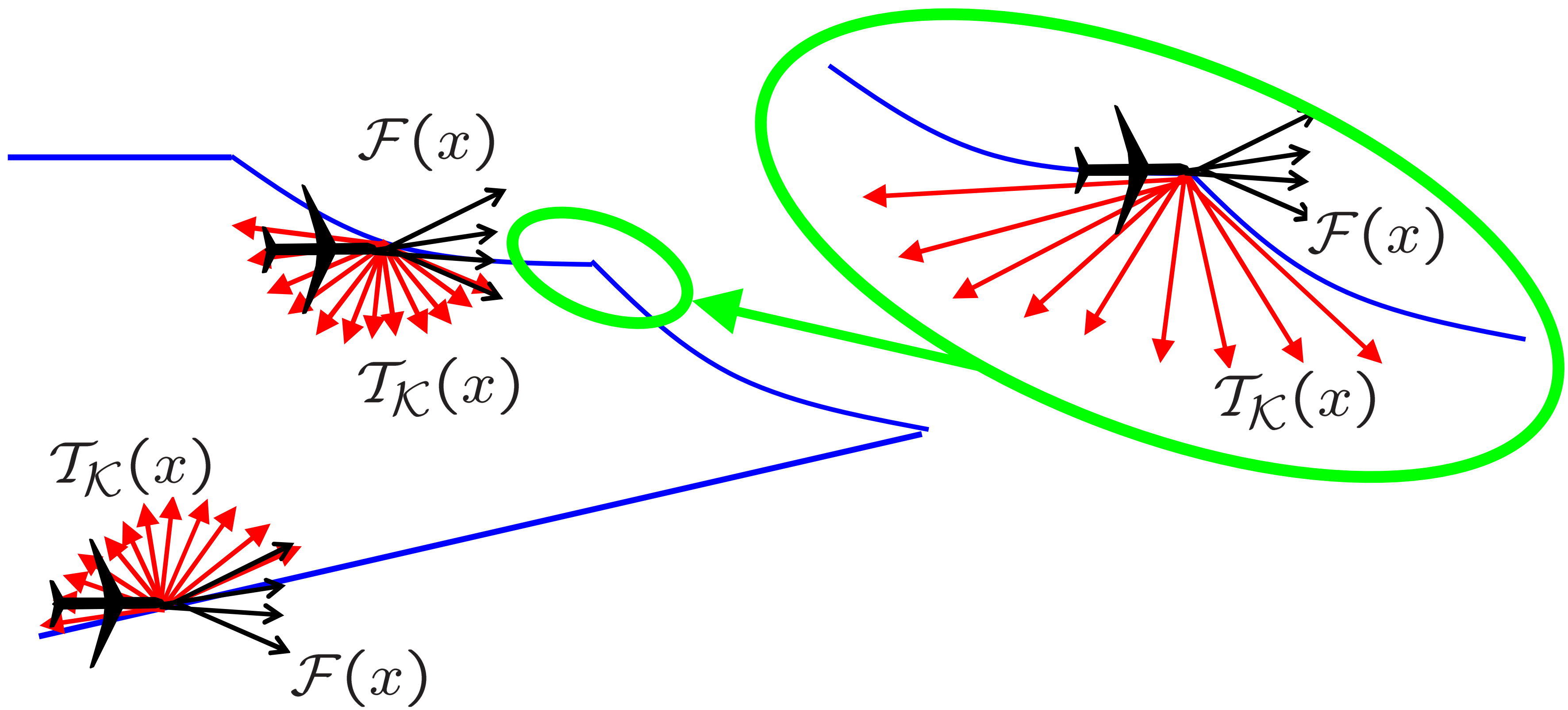
$$\text{Viab}_{\mathcal{F}}(\mathcal{K}) = \{x_i \mid \forall t > 0, x(t) \in \mathcal{K}\}$$

$$\text{where } \dot{x} \in \mathcal{F}(x) \text{ and } x(0) = x_i$$



The Contingent Cone

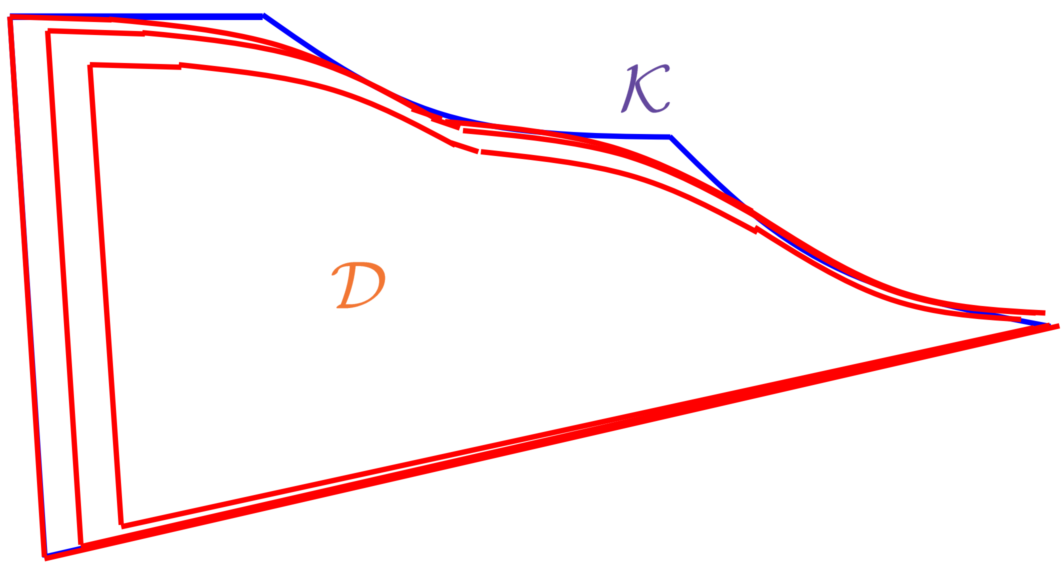
$$\mathcal{T}_{\mathcal{K}}(x) = \limsup_{h \to 0^+} \frac{\mathcal{K} - x}{h}$$



Defining the Viability Kernel

Assume $\mathcal{F}$ is Marchaud and $\mathcal{K}$ is closed.

Then $\text{Viab}_{\mathcal{F}}(\mathcal{K})$ is the largest closed $\mathcal{D}$ such that $\mathcal{F}(x) \cap \mathcal{T}_{\mathcal{D}}(x) \neq \emptyset$.

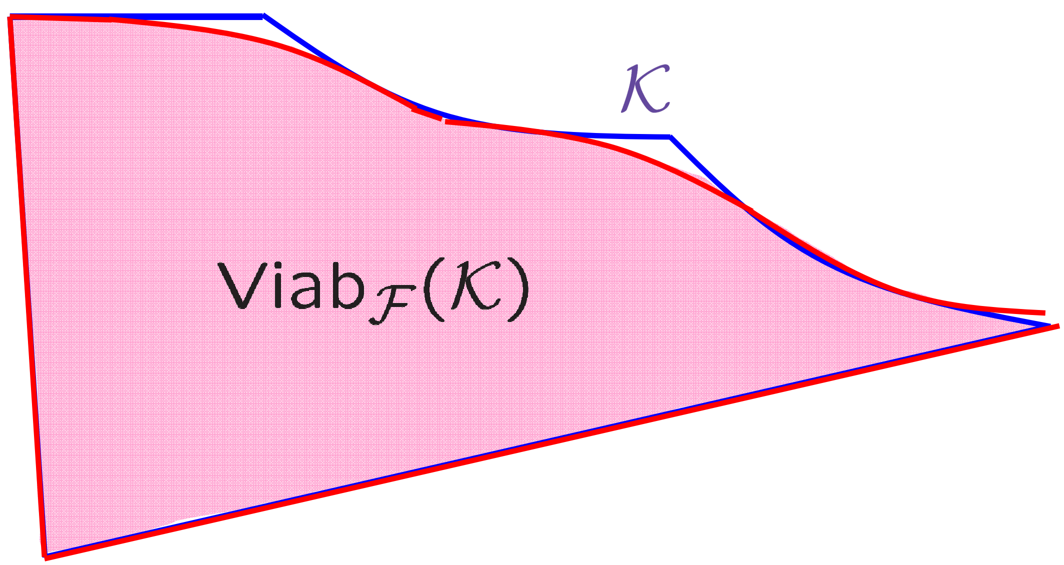


$$\mathcal{F} \text{ is Marchaud if } \begin{cases} \text{the graph and domain of } \mathcal{F} \text{ are nonempty and closed,} \\ \text{the values } \mathcal{F}(x) \text{ are convex,} \\ \text{the growth of } \mathcal{F} \text{ is linear: } \exists c > 0, \forall x, \sup_{v \in \mathcal{F}(x)} \|v\| \leq c(\|x\| + 1) \end{cases}$$

Defining the Viability Kernel

Assume $\mathcal{F}$ is Marchaud and $\mathcal{K}$ is closed.

Then $\text{Viab}_{\mathcal{F}}(\mathcal{K})$ is the largest closed $\mathcal{D}$ such that $\mathcal{F}(x) \cap \mathcal{T}_{\mathcal{D}}(x) \neq \emptyset$.



$$\mathcal{F} \text{ is Marchaud if } \begin{cases} \text{the graph and domain of } \mathcal{F} \text{ are nonempty and closed,} \\ \text{the values } \mathcal{F}(x) \text{ are convex,} \\ \text{the growth of } \mathcal{F} \text{ is linear: } \exists c > 0, \forall x, \sup\limits_{v \in \mathcal{F}(x)} \|v\| \leq c(\|x\| + 1) \end{cases}$$

Connection to Minimum Time to Reach

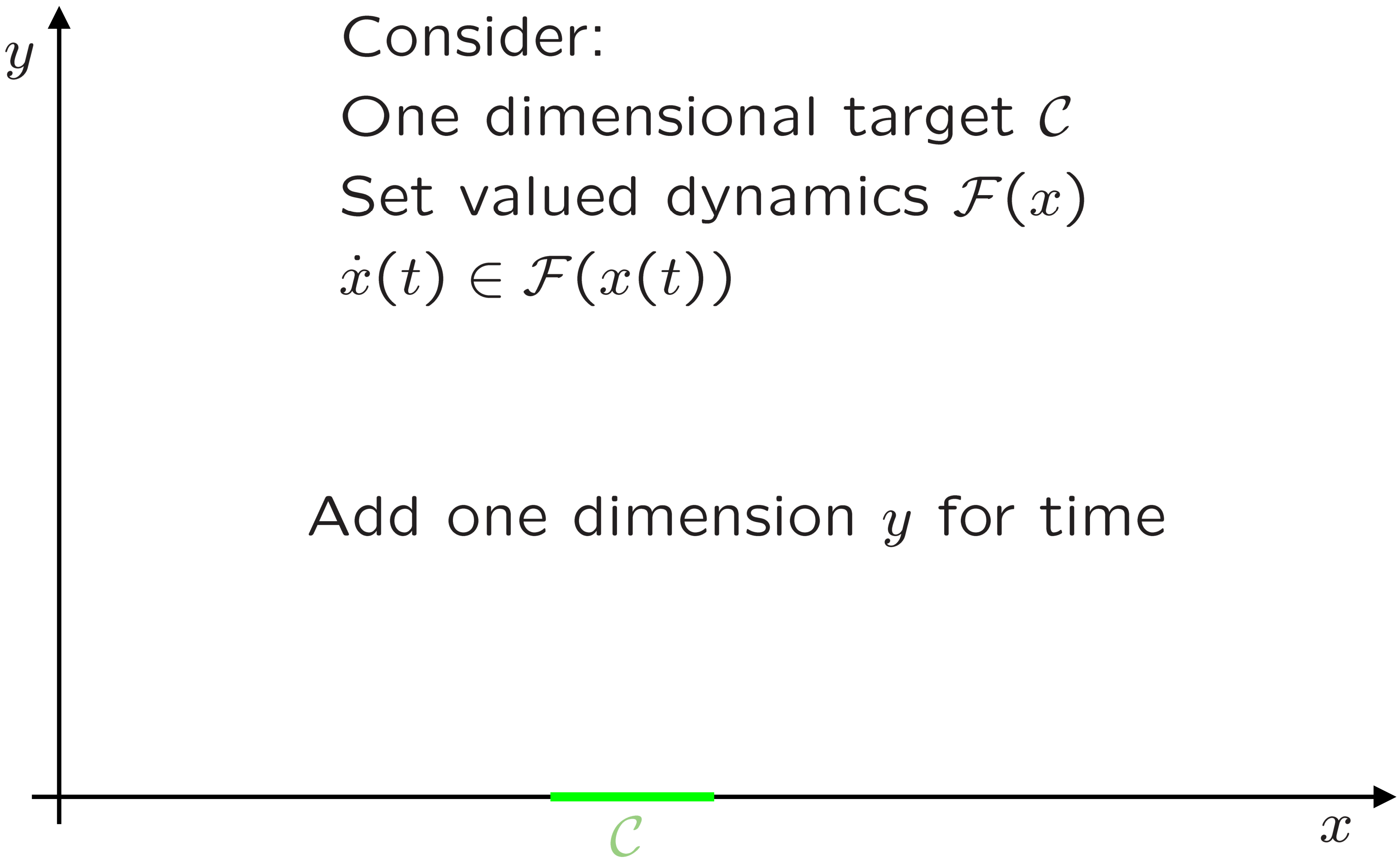
Consider:

One dimensional target $\mathcal{C}$

Set valued dynamics $\mathcal{F}(x)$

$\dot{x}(t) \in \mathcal{F}(x(t))$

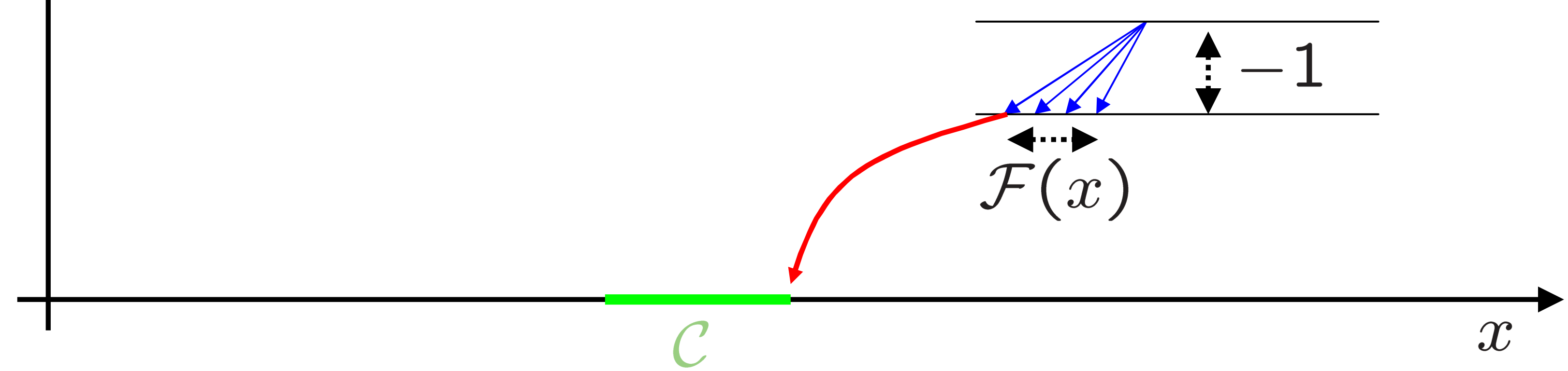
Add one dimension y for time



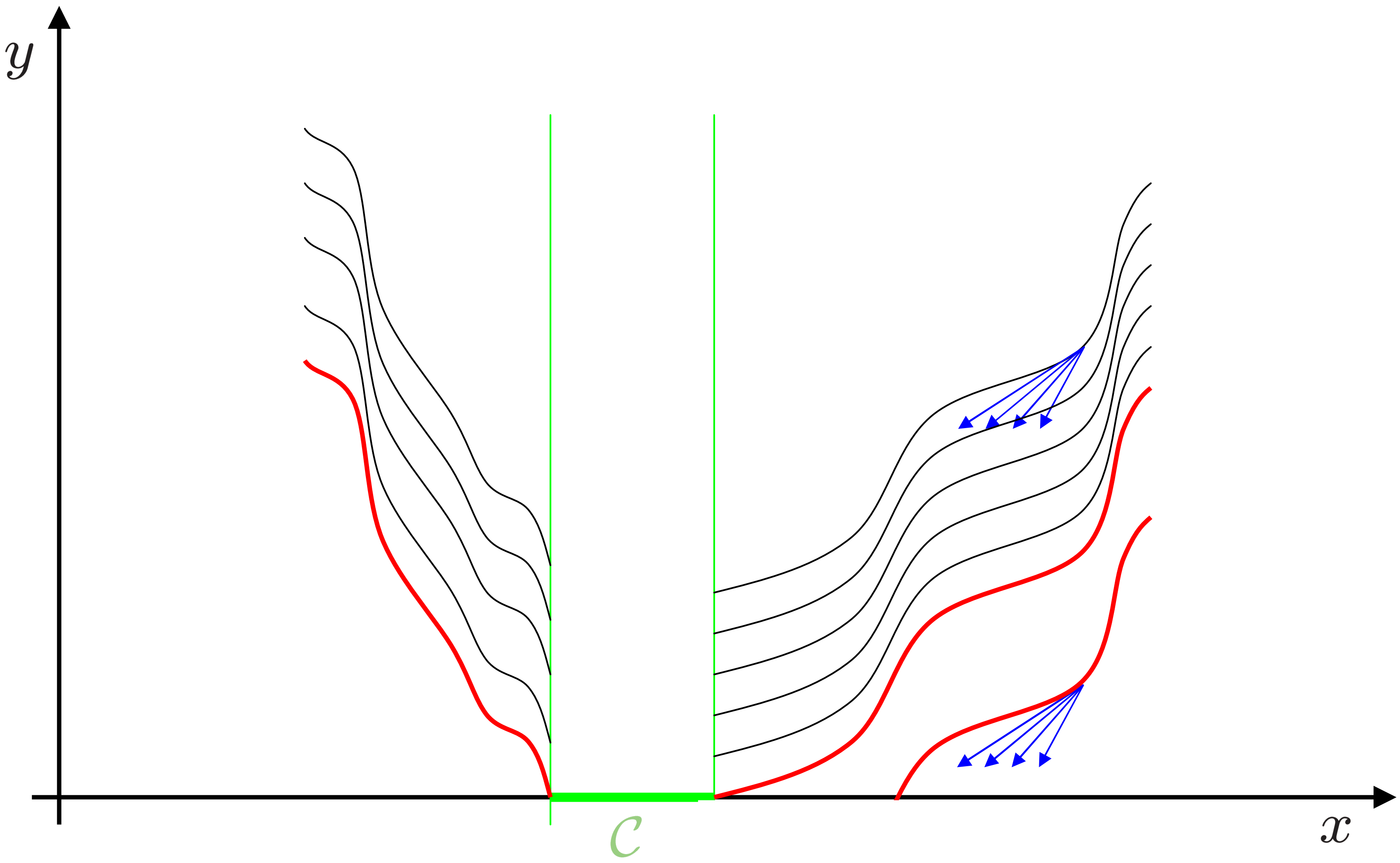
Connection to Minimum Time to Reach

Define dynamics along y axis
so that it is possible to stop in $\mathcal{C}$

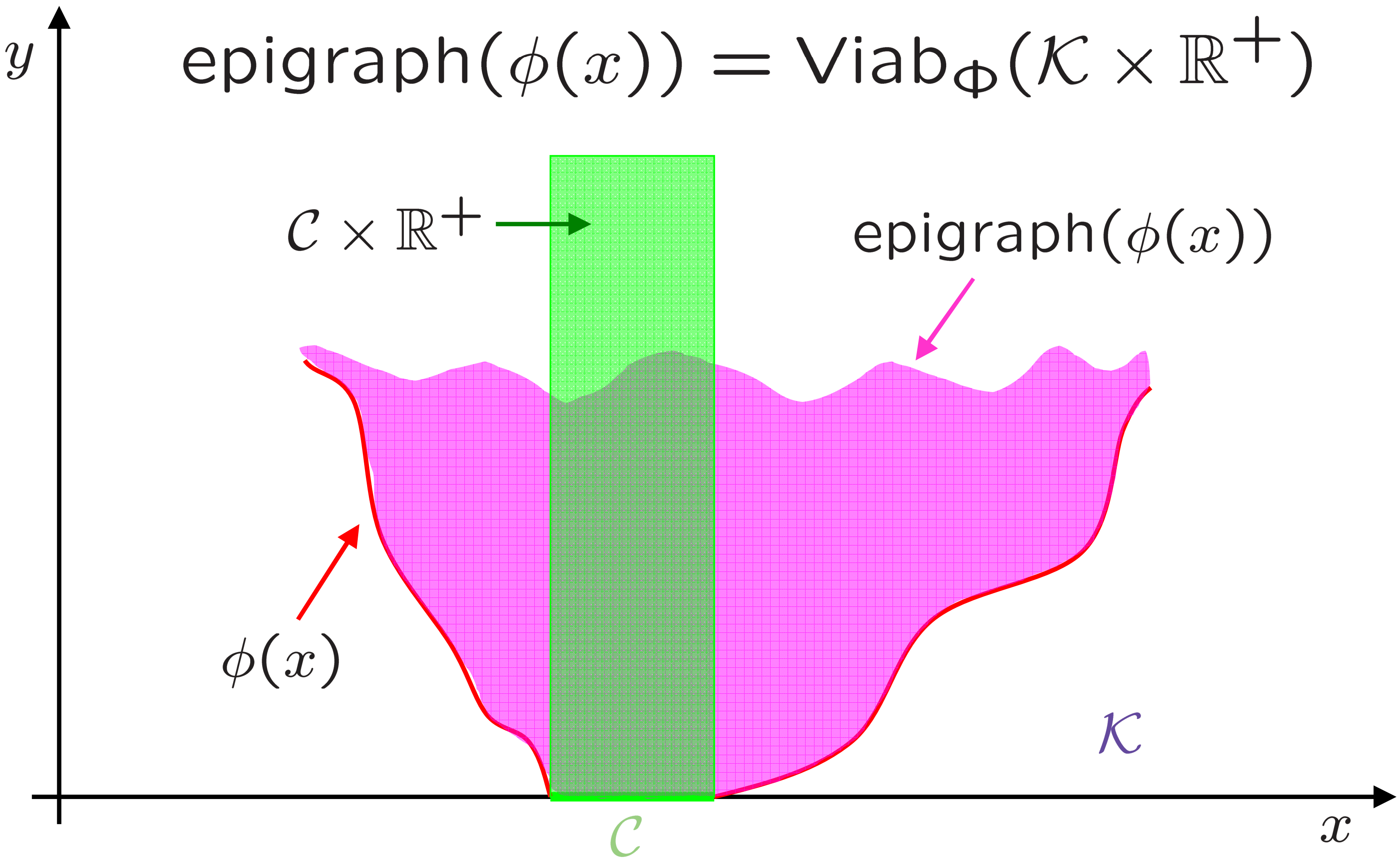
$$\Phi(x) = \begin{cases} \mathcal{F}(x) \times \{-1\}, & \text{if } x \notin \mathcal{C}; \\ \text{convex hull}\left(\mathcal{F}(x) \times \{-1\}, \{0,0\}\right), & \text{if } x \in \mathcal{C} \end{cases}$$



Connection to Minimum Time to Reach



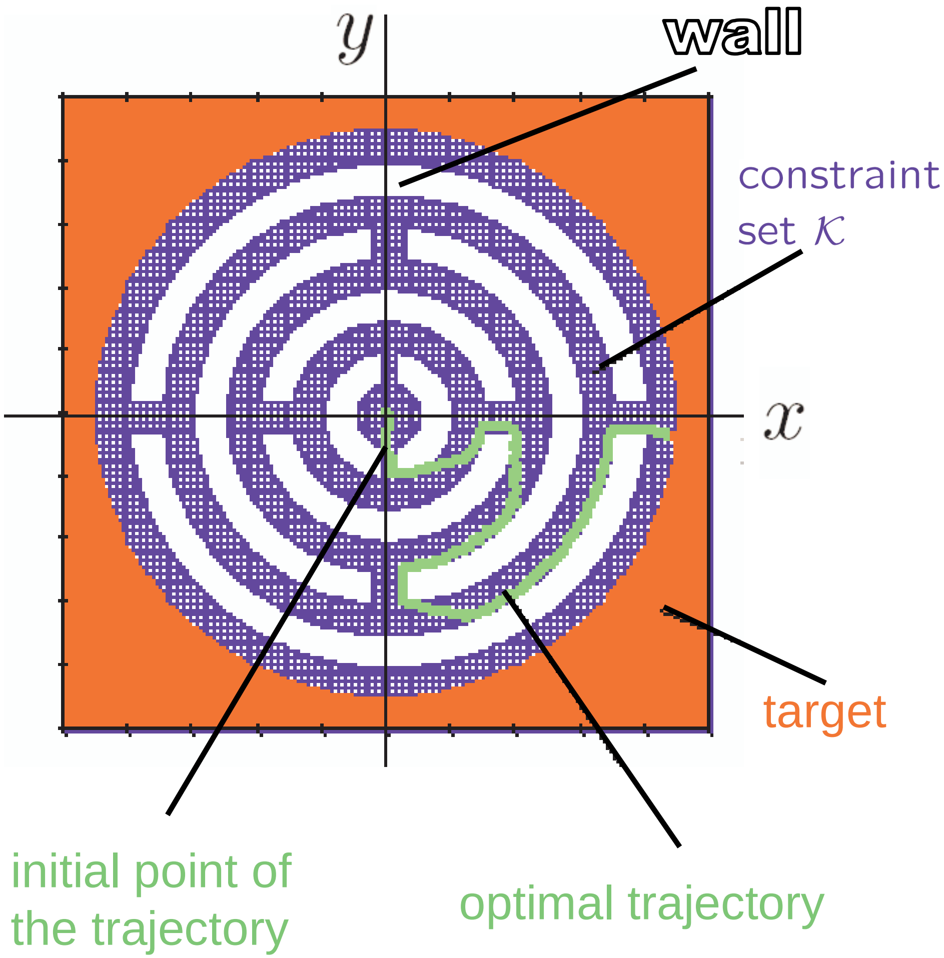
Connection to Minimum Time to Reach

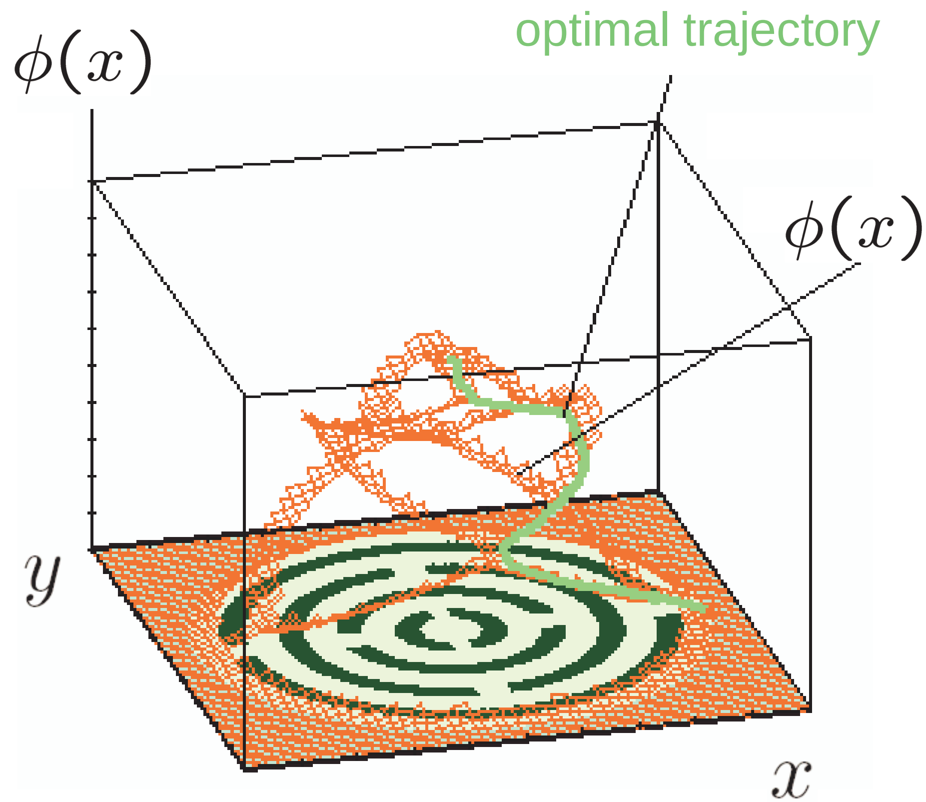


$$\text{epigraph}(\phi(x)) = \{(x, y) \in \mathcal{K} \times \mathbb{R}^{+} \mid y \geq \phi(x)\}$$

[Cardaliaguet, Quincampoix, Saint-Pierre, 1997]

Minimum Time to Reach Example





$$\dot{x} \in \{v \in \mathbb{R}^2 \mid \|v\| \leq 1\}$$

[Saint-Pierre, 2001]