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?

![Diagram showing target (unsafe), backward reach set (uncontrollably unsafe), and safe (under appropriate control) areas.](image-url)
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 $\frac{dx}{dt} = f(x)$?
Approaches to Continuous Reach Sets

• Lagrangian approaches
  – Forward reach sets
  – Restricted class of dynamics
  – Restricted class of sets with compact representation
  – Guarantees of overapproximation
  – Examples: HyTech (Henzinger), Checkmate (Krogh), $d/dt$ (Dang), ellipsoidal (Kurzhanski)

• Eulerian approaches
  – Backward reach sets
  – General dynamics including competitive inputs
  – General set shapes represented implicitly
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

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

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

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

$$\begin{align*}
\dot{x}(t) &= f(x(t)) \\
x(t) &\in G(0)
\end{align*}$$
Why “Backward” Reachable Sets?

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

\[ \dot{x}(t) = -f(x(t)) \]

\[ 0 < t_1 < t_2 < t_3 \]
\[ g(0) \subseteq g(t_1) \subseteq g(t_2) \subseteq g(t_3) \]
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
\[
\dot{x}(t) = f(x(t), \nu(t))
\]

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

\[ x_1(t) \]
\[ x_2(t) \]
\[ x_3 \]
\[ x_1 \]
\[ x_2 \]
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

\[
\begin{align*}
\dot{x}(t) &= f(x(t), \nu(t)) \\
\exists \nu(\cdot), x(t) &\in \mathcal{G}(0)
\end{align*}
\]
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

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

$\forall u(\cdot), \exists d(\cdot), x(t) \in G(0)$
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 $v_e = v_p = 5$

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

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 \(\psi\)

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

**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
  \[ D_t \phi(x, t) + \min [0, H(x, D_x \phi(x, t))] = 0 \]
  with Hamiltonian: \[ H(x, p) = \max_{a \in A} \min_{b \in B} f(x, a, b) \cdot p \]
  and terminal conditions: \[ \phi(x, 0) = h(x) \]
  where \[ G(0) = \{ x \in \mathbb{R}^n \mid h(x) \leq 0 \} \]
  and \[ \dot{x} = f(x, a, b) \]
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 $\xi_f(\cdot; x, t, a(\cdot), b(\cdot))$

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

where

$$\begin{cases} 
\xi_f(t; x, t, a(\cdot), b(\cdot)) = x \\
\dot{\xi}_f((s; x, t, a(\cdot), b(\cdot)) = f(x, a(s), b(s))
\end{cases}$$

- Value function solution $\phi(x, t)$ given by viscosity solution to basic Hamilton-Jacobi equation

  - [Evans & Souganidis, 1984]

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

where

$$\begin{cases} 
H(x, p) = \max_{a \in A} \min_{b \in B} p^T f(x, a, b) \\
\phi(x, 0) = h(x)
\end{cases}$$
Modification for Optimal Stopping Time

• How to keep trajectories from passing through \( G(0) \)?
  
  – [Mitchell, Bayen & Tomlin 2004]
  – Augment disturbance input
    
    \[
    \begin{bmatrix}
    \tilde{b} = \begin{bmatrix} b & b \end{bmatrix}
    \end{bmatrix}
    \text{ where } b : [t, 0] \rightarrow [0, 1]
    \]
    
    \[
    \tilde{f}(x, a, \tilde{b}) = b f(x, a, b)
    \]
  
  – Augmented Hamilton-Jacobi equation solves for reachable set
    
    \[
    D_t \phi(x, t) + \tilde{H} (x, D_x \phi(x, t)) = 0 \quad \text{where} \quad \begin{cases} 
    \tilde{H}(x, p) = \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} p^T \tilde{f}(x, a, \tilde{b}) \\
    \phi(x, 0) = h(x)
    \end{cases}
    \]

  – Augmented Hamiltonian is equivalent to modified Hamiltonian
    
    \[
    \tilde{H}(x, p) = \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} p^T \tilde{f}(x, a, \tilde{b})
    \]
    
    \[
    = \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} \min_{b \in [0,1]} bp^T f(x, a, b)
    \]
    
    \[
    = \min \left[ 0, \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} p^T f(x, a, b) \right] = \min \left[ 0, H(x, p) \right]
    \]
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
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]