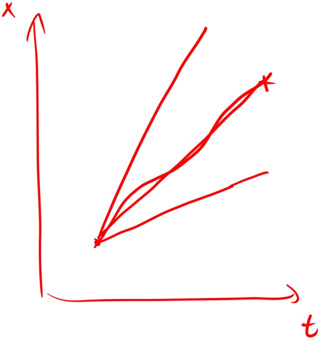
Modeling and Analysis of Hybrid Systems

Reachability analysis for hybrid automata

Prof. Dr. Erika Ábrahám

Informatik 2 - Theory of Hybrid Systems
RWTH Aachen University

SS 2012

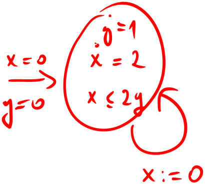


$\dot{x} \in [a, b]$

$\dot{x} = Ax + Bu$

General forward reachability computation

$\dot{x} = Ax + Bu$

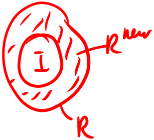


Input: Set Init of initial states.

Algorithm:

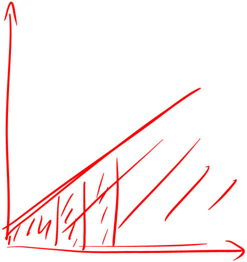
$R^{\text{new}} :=$ Init;
$R := \emptyset$;
while $(R^{\text{new}} \neq \emptyset)\{$
  $R := R \cup R^{\text{new}}$;
  $R^{\text{new}} :=$ Reach$(R^{\text{new}}) \backslash R$;
$\}$

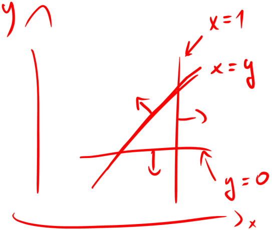
Output: Set R of reachable states.

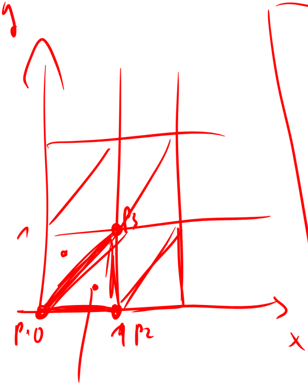


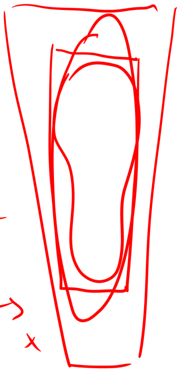
Reach$(x=0 \wedge y>0) = x \geq 0 \wedge y>0 \wedge x \leq 2y$

$R^{new} = x=0 \wedge y=0$
$R = \emptyset$
$\rightarrow$
Reach$(R^{new}) = x \geq 0 \wedge y \geq 0 \wedge x=2y$
$R = x=0 \wedge y=0$
$R^{new} = x>0 \wedge y>0 \wedge x=2y$
$\rightarrow$
Reach$(x>0 \wedge y>0 \wedge x=2y) = x=0 \wedge y>0$
$R = x \geq 0 \wedge y \geq 0 \wedge x=2y$
$R^{new} = (x=0 \wedge y>0 \wedge x \leq 2y) \backslash \{x \geq 0 \wedge y \geq 0 \wedge x=2y\}$
$= x=0 \wedge y>0 \wedge x \leq 2y \wedge x \neq 2y$





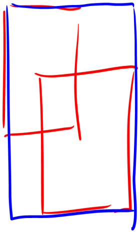




$0 < x < 1 \wedge 0 < y < 1 \wedge x > y$

$x < 1 \wedge y > 0 \wedge x > y$

$\{p_1, p_2, p_3\}$



Reachability computation

- When applied to hybrid automata, there is a problem with this procedure:
  How to compute Reach(P) for a set P?

Reachability computation

- When applied to hybrid automata, there is a problem with this procedure:
  How to compute Reach(P) for a set P?
- Generally there are two kinds of approaches:
  1. CEGAR (CounterExample-Guided Abstraction Refinement):
     - Build a finite abstraction of the state space.
     - Compute reachability for the abstract system.
     - Spurious counterexamples $\rightarrow$ abstraction refinement.
  2. Compute an over-approximation of Reach(P) in the above procedure.

Reachability computation

- When applied to hybrid automata, there is a problem with this procedure:
  How to compute Reach(P) for a set P?
- Generally there are two kinds of approaches:
  1. CEGAR (CounterExample-Guided Abstraction Refinement):
     - Build a finite abstraction of the state space.
     - Compute reachability for the abstract system.
     - Spurious counterexamples $\rightarrow$ abstraction refinement.
  2. Compute an over-approximation of Reach(P) in the above procedure.
- We have seen an example for (1) for timed automata.
- We have seen another example for (1) for minimization with on-the-fly refinement during the fixed-point computation.
- Let us now have a closer look at (2).

Computing reachability

We need to solve two problems:

$\dot{x} = Ax$

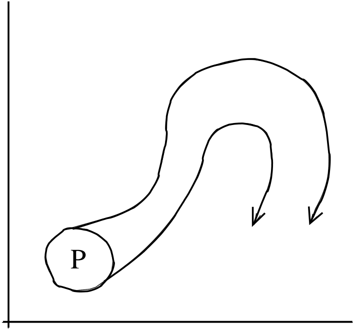
Continuous dynamics

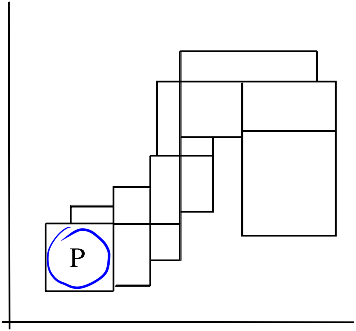
Given a dynamical system defined by $\dot{x} = f(x)$, where x takes values from $\mathbb{R}^d$, and given $P \subseteq \mathbb{R}^d$, calculate (or over-approximate) the set of points in $\mathbb{R}^d$ reached by trajectories (solutions of $\dot{x} = f(x)$) starting in P.

Discrete steps

Given a discrete transition of a hybrid system with state space $\mathbb{R}^d$, and given $P \subseteq \mathbb{R}^d$, calculate (or approximate) the set of points in $\mathbb{R}^d$ reachable by taking a discrete transition starting in P.

Reachability approximation for hybrid automata





State set representation

- The geometry chosen to represent reachable sets has a crucial effect on the efficiency of the whole procedure.
- Usually, the more complex the geometry,
  1. the more costly is the storage of the sets,
  2. the more difficult it is to perform operations like union and intersection, and
  3. the more elaborate is the computation of new reachable sets, but
  4. the better the approximation of the set of reachable states.
- Choosing the geometry has to be a compromise between these impacts.

Representation requirements

The geometry should allow efficient computation of the operations for

- membership relation,
- union,
- intersection,
- subtraction,
- test for emptiness.

State set representation

Approaches:

- Convex polyhedra
- Orthogonal polyhedra
- Oriented rectangular hulls
- Zonotopes
- Support functions
- Ellipsoids
- ...