Hybrid Systems

Prof. Dr. Andreas Podelski

Chair of Software Engineering
University of Freiburg

SS 2012

General forward reachability computation

Input: Set Init of initial states.

Algorithm:

$$
\begin{array}{l}
R^{\mathsf{new}} := \mathsf{Init}; \\
R := \emptyset; \\
\text{while } (R^{\mathsf{new}} \neq \emptyset)\{ \\
\qquad R \qquad := R \cup R^{\mathsf{new}}; \\
\qquad R^{\mathsf{new}} \quad := \mathsf{Reach}(R^{\mathsf{new}}) \backslash R; \\
\}
\end{array}
$$

Output: Set R of reachable states.

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.
- Let us have a look at (2) in more details.

Computing reachability

We need to solve two problems:

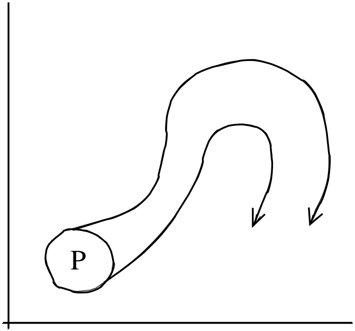
Continuous dynamics

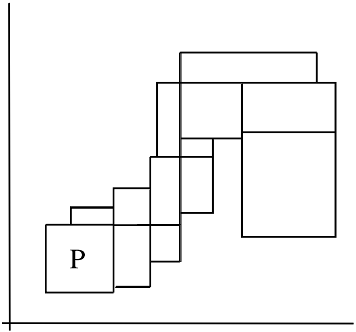
Given a dynamical system defined by $\dot{x} = f(x)$, where x takes values from $\mathbb{R}^d$, and given $P \subseteq \mathbb{R}^d$, calculate (or over-approximate) the set of points in $\mathbb{R}^d$ reached by trajectories (solutions) starting in P.

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 the 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

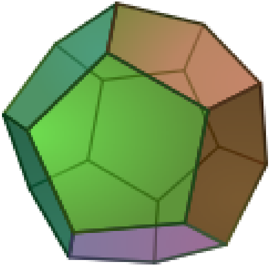
Approaches:

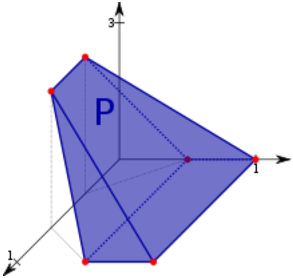
- Convex polyhedra
- Orthogonal polyhedra
- Oriented rectangular hulls
- Zonotopes, ellipsoids, support functions,...

1 Convex polyhedra

2 Operations on convex polyhedra

Polyhedra





Convex polyhedra

Definition

A (convex) polyhedron in $\mathbb{R}^d$ is the solution set to a finite number of linear inequalities with real coefficients in d real variables. A bounded polyhedron is called polytope.

Depending on the form of the representation, we distinguish between

- $\mathcal{H}$-polytopes and
- $\mathcal{V}$-polytopes.

$\mathcal{H}$-polytopes

Definition (Closed halfspace)

A d-dimensional closed halfspace is a set $\mathcal{H} = \{x \in \mathbb{R}^d \mid c \cdot x \leq z\}$ for some $c \in \mathbb{R}^d$, called the normal of the halfspace, and a $z \in \mathbb{R}$.

Definition (Closed halfspace)

A d-dimensional closed halfspace is a set $\mathcal{H} = \{x \in \mathbb{R}^d \mid c \cdot x \leq z\}$ for some $c \in \mathbb{R}^d$, called the normal of the halfspace, and a $z \in \mathbb{R}$.

Definition ($\mathcal{H}$-polyhedron, $\mathcal{H}$-polytope)

A d-dimensional $\mathcal{H}$-polyhedron $P = \bigcap_{i=1}^{n} \mathcal{H}_i$ is the intersection of finitely many closed halfspaces. A bounded $\mathcal{H}$-polyhedron is called an $\mathcal{H}$-polytope.

The facets of a d-dimensional $\mathcal{H}$-polytope are $d-1$-dimensional $\mathcal{H}$-polytopes.

$\mathcal{H}$-polytopes

An $\mathcal{H}$-polytope

$$P \quad = \quad \bigcap_{i=1}^{n} \mathcal{H}_i \quad = \quad \bigcap_{i=1}^{n} \{x \in \mathbb{R}^d \mid c_i \cdot x \leq z_i\}$$

can also be written in the form

$$P = \{x \in \mathbb{R}^d \mid Cx \leq z\}.$$

We call (C, z) the $\mathcal{H}$-representation of the polytope.

$\mathcal{H}$-polytopes

An $\mathcal{H}$-polytope

$$P \quad = \quad \bigcap_{i=1}^{n} \mathcal{H}_i \quad = \quad \bigcap_{i=1}^{n} \{x \in \mathbb{R}^d \mid c_i \cdot x \leq z_i\}$$

can also be written in the form

$$P = \{x \in \mathbb{R}^d \mid Cx \leq z\}.$$

We call (C, z) the $\mathcal{H}$-representation of the polytope.

- Each row of C is the normal vector to the ith facet of the polytope.

$\mathcal{H}$-polytopes

An $\mathcal{H}$-polytope

$$P \quad = \quad \bigcap_{i=1}^{n} \mathcal{H}_i \quad = \quad \bigcap_{i=1}^{n} \{x \in \mathbb{R}^d \mid c_i \cdot x \leq z_i\}$$

can also be written in the form

$$P = \{x \in \mathbb{R}^d \mid Cx \leq z\}.$$

We call (C, z) the $\mathcal{H}$-representation of the polytope.

- Each row of C is the normal vector to the ith facet of the polytope.
- An $\mathcal{H}$-polytope P has a finite number of vertices $V(P)$.

$\mathcal{H}$-polytopes

An $\mathcal{H}$-polytope

$$P \quad = \quad \bigcap_{i=1}^{n} \mathcal{H}_i \quad = \quad \bigcap_{i=1}^{n} \{x \in \mathbb{R}^d \mid c_i \cdot x \leq z_i\}$$

can also be written in the form

$$P = \{x \in \mathbb{R}^d \mid Cx \leq z\}.$$

We call (C, z) the $\mathcal{H}$-representation of the polytope.

- Each row of C is the normal vector to the ith facet of the polytope.
- An $\mathcal{H}$-polytope P has a finite number of vertices $V(P)$.

Definition

A set S is called convex, if

$$\forall x, y \in S.\ \forall \lambda \in [0, 1] \subseteq \mathbb{R}.\ \lambda x + (1 - \lambda) y \in S.$$

$\mathcal{H}$-polyhedra are convex sets.

$\mathcal{V}$-polytopes

Definition (Convex hull)

Given a set $V \subseteq \mathbb{R}^d$, the convex hull $CH(V)$ of V is the smallest convex set that contains V.

$\mathcal{V}$-polytopes

Definition (Convex hull)

Given a set $V \subseteq \mathbb{R}^d$, the convex hull $CH(V)$ of V is the smallest convex set that contains V.

For a finite set $V = \{v_1, \ldots, v_n\}$, its convex hull can be computed by

$$CH(V) = \{x \in \mathbb{R}^d \mid \exists \lambda_1, \ldots, \lambda_n \in [0,1] \subseteq \mathbb{R}^d. \sum_{i=1}^{n} \lambda_i = 1 \wedge \sum_{i=1}^{n} \lambda_i v_i = x\}.$$

$\mathcal{V}$-polytopes

Definition (Convex hull)

Given a set $V \subseteq \mathbb{R}^d$, the convex hull $CH(V)$ of V is the smallest convex set that contains V.

For a finite set $V = \{v_1, \ldots, v_n\}$, its convex hull can be computed by

$$CH(V) = \{x \in \mathbb{R}^d \mid \exists \lambda_1, \ldots, \lambda_n \in [0,1] \subseteq \mathbb{R}^d.\ \sum_{i=1}^{n} \lambda_i = 1 \wedge \sum_{i=1}^{n} \lambda_i v_i = x\}.$$

Definition ($\mathcal{V}$-polytope)

A $\mathcal{V}$-polytope $P = CH(V)$ is the convex hull of a finite set $V \subset \mathbb{R}^d$. We call V the $\mathcal{V}$-representation of the polytope.

$\mathcal{V}$-polytopes

Definition (Convex hull)

Given a set $V \subseteq \mathbb{R}^d$, the convex hull $CH(V)$ of V is the smallest convex set that contains V.

For a finite set $V = \{v_1, \ldots, v_n\}$, its convex hull can be computed by

$$CH(V) = \{x \in \mathbb{R}^d \mid \exists \lambda_1, \ldots, \lambda_n \in [0,1] \subseteq \mathbb{R}^d.\ \sum_{i=1}^{n} \lambda_i = 1 \wedge \sum_{i=1}^{n} \lambda_i v_i = x\}.$$

Definition ($\mathcal{V}$-polytope)

A $\mathcal{V}$-polytope $P = CH(V)$ is the convex hull of a finite set $V \subset \mathbb{R}^d$. We call V the $\mathcal{V}$-representation of the polytope.

Note that all $\mathcal{V}$-polytopes are bounded.

Motzkin's theorem

- For each $\mathcal{H}$-polytope, the convex hull of its vertices defines the same set in the form of a $\mathcal{V}$-polytope, and vice versa,
- each set defined as a $\mathcal{V}$-polytope can be also given as an $\mathcal{H}$-polytope by computing the halfspaces defined by its facets.

The translations between the $\mathcal{H}$- and the $\mathcal{V}$-representations of polytopes can be very expensive.

State set representation

1 Convex polyhedra

2 Operations on convex polyhedra

Operations

If we represent reachable sets of hybrid automata by polytopes, we need some operations like

- membership computation,
- intersection, or the
- union of two polytopes.

Operations: Membership

Membership for $p \in \mathbb{R}^d$:

Operations: Membership

Membership for $p \in \mathbb{R}^d$:

- $\mathcal{H}$-polytope defined by $Cx \leq z$:

Operations: Membership

Membership for $p \in \mathbb{R}^d$:

- $\mathcal{H}$-polytope defined by $Cx \leq z$:
  just substitute p for x to check if the inequation holds.

Operations: Membership

Membership for $p \in \mathbb{R}^d$:

- $\mathcal{H}$-polytope defined by $Cx \leq z$:
  just substitute p for x to check if the inequation holds.
- $\mathcal{V}$-polytope defined by the vertex set V:

Operations: Membership

Membership for $p \in \mathbb{R}^d$:

- $\mathcal{H}$-polytope defined by $Cx \leq z$:
  just substitute p for x to check if the inequation holds.
- $\mathcal{V}$-polytope defined by the vertex set V:
  check satisfiability of

$$\exists \lambda_1, \ldots, \lambda_n \in [0,1] \subseteq \mathbb{R}^d.\ \sum_{i=1}^{n} \lambda_i = 1 \wedge \sum_{i=1}^{n} \lambda_i v_i = x\ .$$

Operations: Membership

Membership for $p \in \mathbb{R}^d$:

- $\mathcal{H}$-polytope defined by $Cx \leq z$:
  just substitute p for x to check if the inequation holds.
- $\mathcal{V}$-polytope defined by the vertex set V:
  check satisfiability of

$$\exists \lambda_1, \ldots, \lambda_n \in [0,1] \subseteq \mathbb{R}^d. \sum_{i=1}^{n} \lambda_i = 1 \wedge \sum_{i=1}^{n} \lambda_i v_i = x \ .$$

Alternatively:

Operations: Membership

Membership for $p \in \mathbb{R}^d$:

- $\mathcal{H}$-polytope defined by $Cx \leq z$:
  just substitute p for x to check if the inequation holds.
- $\mathcal{V}$-polytope defined by the vertex set V:
  check satisfiability of

$$\exists \lambda_1, \ldots, \lambda_n \in [0,1] \subseteq \mathbb{R}^d. \sum_{i=1}^{n} \lambda_i = 1 \wedge \sum_{i=1}^{n} \lambda_i v_i = x \; .$$

Alternatively: convert the $\mathcal{V}$-polytope into an $\mathcal{H}$-polytope by computing its facets.

Intersection

Intersection for two polytopes P_1 and P_2:

- $\mathcal{H}$-polytopes defined by $C_1 x \leq z_1$ and $C_2 x \leq z_2$:

Intersection for two polytopes P_1 and P_2:

- $\mathcal{H}$-polytopes defined by $C_1 x \leq z_1$ and $C_2 x \leq z_2$:
  the resulting $\mathcal{H}$-polytope is defined by $\begin{pmatrix} C_1 \\ C_2 \end{pmatrix} x \leq \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$.
- $\mathcal{V}$-polytopes defined by V_1 and V_2:

Intersection

Intersection for two polytopes P_1 and P_2:

- $\mathcal{H}$-polytopes defined by $C_1 x \leq z_1$ and $C_2 x \leq z_2$:
  the resulting $\mathcal{H}$-polytope is defined by $\begin{pmatrix} C_1 \\ C_2 \end{pmatrix} x \leq \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$.
- $\mathcal{V}$-polytopes defined by V_1 and V_2:
  Convert P_1 and P_2 to $\mathcal{H}$-polytopes and convert the result back to a $\mathcal{V}$-polytope.

Union

Note that the union of two convex polytopes is in general not a convex polytope.

Union

Note that the union of two convex polytopes is in general not a convex polytope.
$\rightarrow$ take the convex hull of the union.

Union

Note that the union of two convex polytopes is in general not a convex polytope.
$\rightarrow$ take the convex hull of the union.

- $\mathcal{V}$-polytopes defined by V_1 and V_2:

Union

Note that the union of two convex polytopes is in general not a convex polytope.
$\rightarrow$ take the convex hull of the union.

- $\mathcal{V}$-polytopes defined by V_1 and V_2:
  $\mathcal{V}$-representation $V_1 \cup V_2$.
- $\mathcal{H}$-polytopes defined by $C_1 x \leq z_1$ and $C_2 x \leq z_2$:

Union

Note that the union of two convex polytopes is in general not a convex polytope.

$\rightarrow$ take the convex hull of the union.

- $\mathcal{V}$-polytopes defined by V_1 and V_2:
  $\mathcal{V}$-representation $V_1 \cup V_2$.
- $\mathcal{H}$-polytopes defined by $C_1 x \leq z_1$ and $C_2 x \leq z_2$:
  convert to $\mathcal{V}$-polytopes and compute back the result.