24: Model Checking & Reachability Analysis

15-424: Foundations of Cyber-Physical Systems

Goran Frehse André Platzer

Verimag

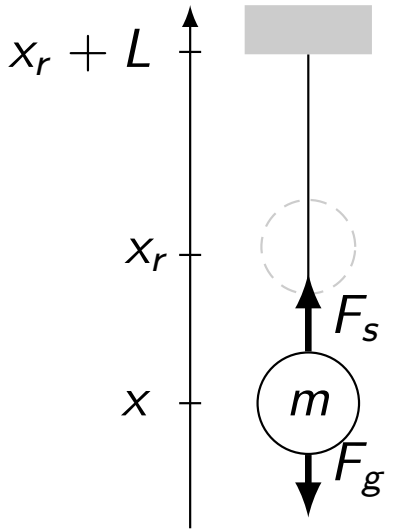
[1] Laurent Doyen, Goran Frehse, George J. Pappas, André Platzer. Verification of Hybrid Systems. In Edmund M. Clarke, Thomas A. Henzinger and Helmut Veith, editors, *Handbook of Model Checking*. Springer, 2017.

Outline

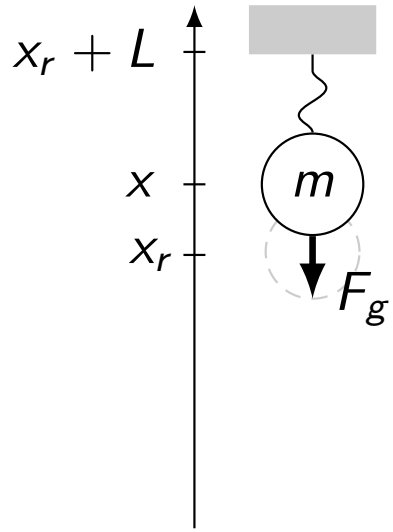
Outline

Outline

Example: Ball on String

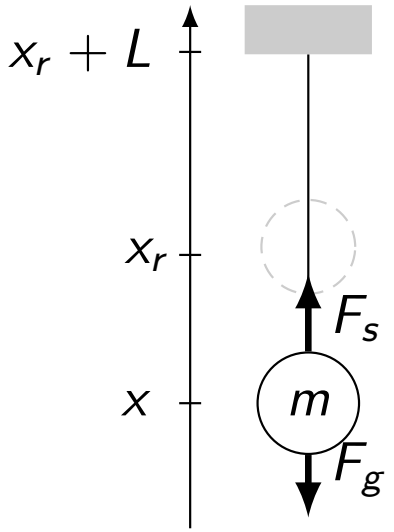


(a) *extension*

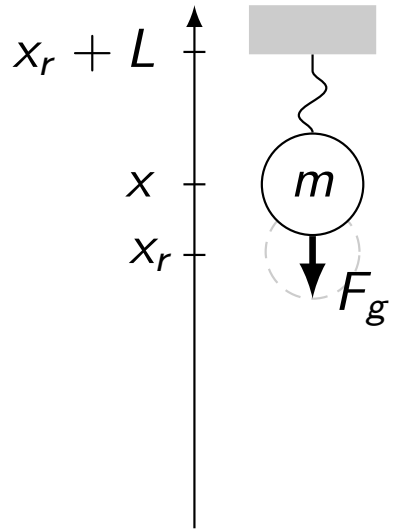


(b) *freefall*

Example: Ball on String



(c) *extension*



(d) *freefall*

dynamics in *freefall* when $x \geq x_r$, with mass m

$$m\ddot{x} = F_g = -mg$$

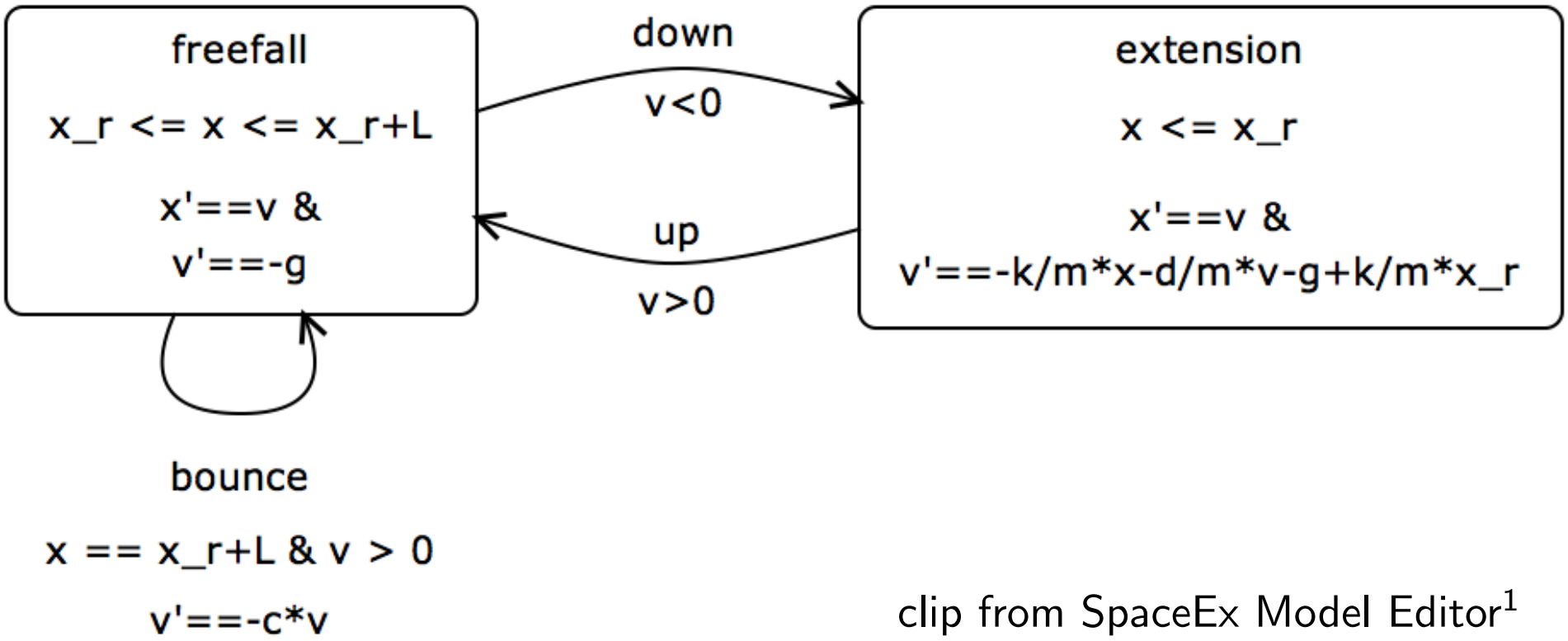
dynamics in *extension* when $x \leq x_r$, spring constant k, damping d

$$m\ddot{x} = F_g + F_s = -mg + kx_r - kx - d\dot{x}$$

transition when $x = x_r + L$, collision factor $c \in [0, 1]$ $\quad \dot{x}^+ = -c\dot{x}$

Hybrid Automaton Model: Ball on String

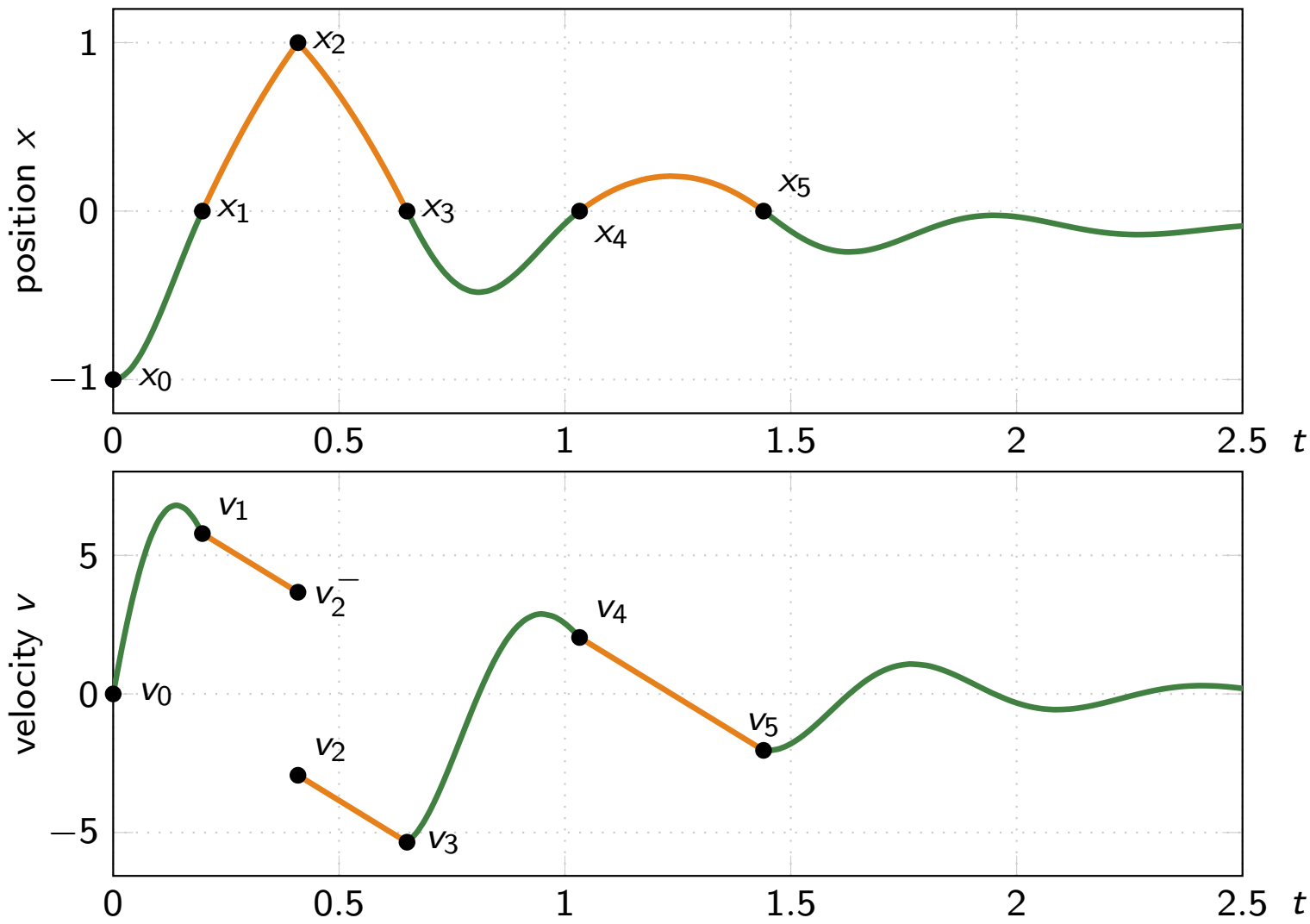
auxiliary variable $v = \dot{x}$, so $\dot{v} = \ddot{x}$



clip from SpaceEx Model Editor[1]

[1] G. Frehse, C. L. Guernic, A. Donzé, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, and O. Maler, "Spaceex: Scalable verification of hybrid systems," in *CAV'11*, ser. LNCS, Springer, 2011.

Simulation


position x
1
0
−1
x_0
x_1
x_2
x_3
x_4
x_5
0
0.5
1
1.5
2
2.5
t
velocity v
5
0
−5
v_0
v_1
v_2^-
v_2
v_3
v_4
v_5
0
0.5
1
1.5
2
2.5
t

Outline

Hybrid Automata

(Alur, Henzinger, '95)[2][3]

- locations $\text{Loc} = \{\ell_1, \ldots, \ell_m\}$ and variables $X = \{x_1, \ldots, x_n\}$ define the state space $\text{Loc} \times \mathbb{R}^X$
- transitions $\text{Edg} \subseteq \text{Loc} \times \text{Lab} \times \text{Loc}$ define location changes with synchronization labels Lab
- evolution domain alias invariant $\text{Inv} \subseteq \text{Loc} \times \mathbb{R}^X$,
- flow relation Flow, where $\text{Flow}(\ell) \subseteq \mathbb{R}^{\dot{X}} \times \mathbb{R}^X$, e.g.,

$$\dot{\mathbf{x}} = f(\mathbf{x});$$

- jump relation Jump, where $\text{Jump}(e) \subseteq \mathbb{R}^X \times \mathbb{R}^{X^+}$, e.g.,

$$\text{Jump}(e) = \{(\mathbf{x}, \mathbf{x}^+) \mid \mathbf{x} \in \mathcal{G} \wedge \mathbf{x}^+ = r(\mathbf{x})\}$$

- initial states $\text{Init} \subseteq \text{Inv}$

Hybrid Automata (Alur, Henzinger, '95)[2][3]

- locations $\text{Loc} = \{\ell_1, \ldots, \ell_m\}$ and variables $X = \{x_1, \ldots, x_n\}$ define the state space $\text{Loc} \times \mathbb{R}^X$
- transitions $\text{Edg} \subseteq \text{Loc} \times \text{Lab} \times \text{Loc}$ define location changes with synchronization labels Lab
- evolution domain alias invariant $\text{Inv} \subseteq \text{Loc} \times \mathbb{R}^X$,
- flow relation Flow, where $\text{Flow}(\ell) \subseteq \mathbb{R}^{\dot{X}} \times \mathbb{R}^X$, e.g.,

$$\dot{\mathbf{x}} = f(\mathbf{x});$$

- jump relation Jump, where $\text{Jump}(e) \subseteq \mathbb{R}^X \times \mathbb{R}^{X^+}$, e.g.,

$$\text{Jump}(e) = \{(\mathbf{x}, \mathbf{x}^+) \mid \mathbf{x} \in \mathcal{G} \wedge \mathbf{x}^+ = r(\mathbf{x})\}$$

- initial states $\text{Init} \subseteq \text{Inv}$
- Except: all described by semialgebraic sets (often polyhedra)

Run Semantics

$$(\ell_0, \mathbf{x}_0) \xrightarrow{\delta_0, \xi_0} (\ell_0, \xi_0(\delta_0)) \xrightarrow{\alpha_0} (\ell_1, \mathbf{x}_1) \xrightarrow{\delta_1, \xi_1} (\ell_1, \xi_1(\delta_1)) \ldots$$

with $(\ell_0, \mathbf{x}_0) \in \mathsf{Init}$, $\alpha_i \in \mathsf{Lab} \cup \{\tau\}$, and for $i = 0, 1, \ldots$:

1. Trajectories: $(\dot{\xi}(t), \xi(t)) \in \mathsf{Flow}(\ell)$ and $\xi_i(t) \in \mathsf{Inv}(\ell_i)$ for all $t \in [0, \delta_i]$.
2. Jumps: $(\xi_i(\delta_i), \mathbf{x}_{i+1}) \in \mathsf{Jump}(e_i)$, $e_i = (\ell_i, \alpha_i, \ell_{i+1}) \in \mathsf{Edg}$, and $\mathbf{x}_{i+1} \in \mathsf{Inv}(\ell_{i+1})$.

A state $(\ell, \mathbf{x})$ is reachable if there exists a run with $(\ell_i, \mathbf{x}_i) = (\ell, \mathbf{x})$ for some i.

Run Semantics

$$(\ell_0, \mathbf{x}_0) \xrightarrow{\delta_0,\xi_0} (\ell_0, \xi_0(\delta_0)) \xrightarrow{\alpha_0} (\ell_1, \mathbf{x}_1) \xrightarrow{\delta_1,\xi_1} (\ell_1, \xi_1(\delta_1)) \dots$$

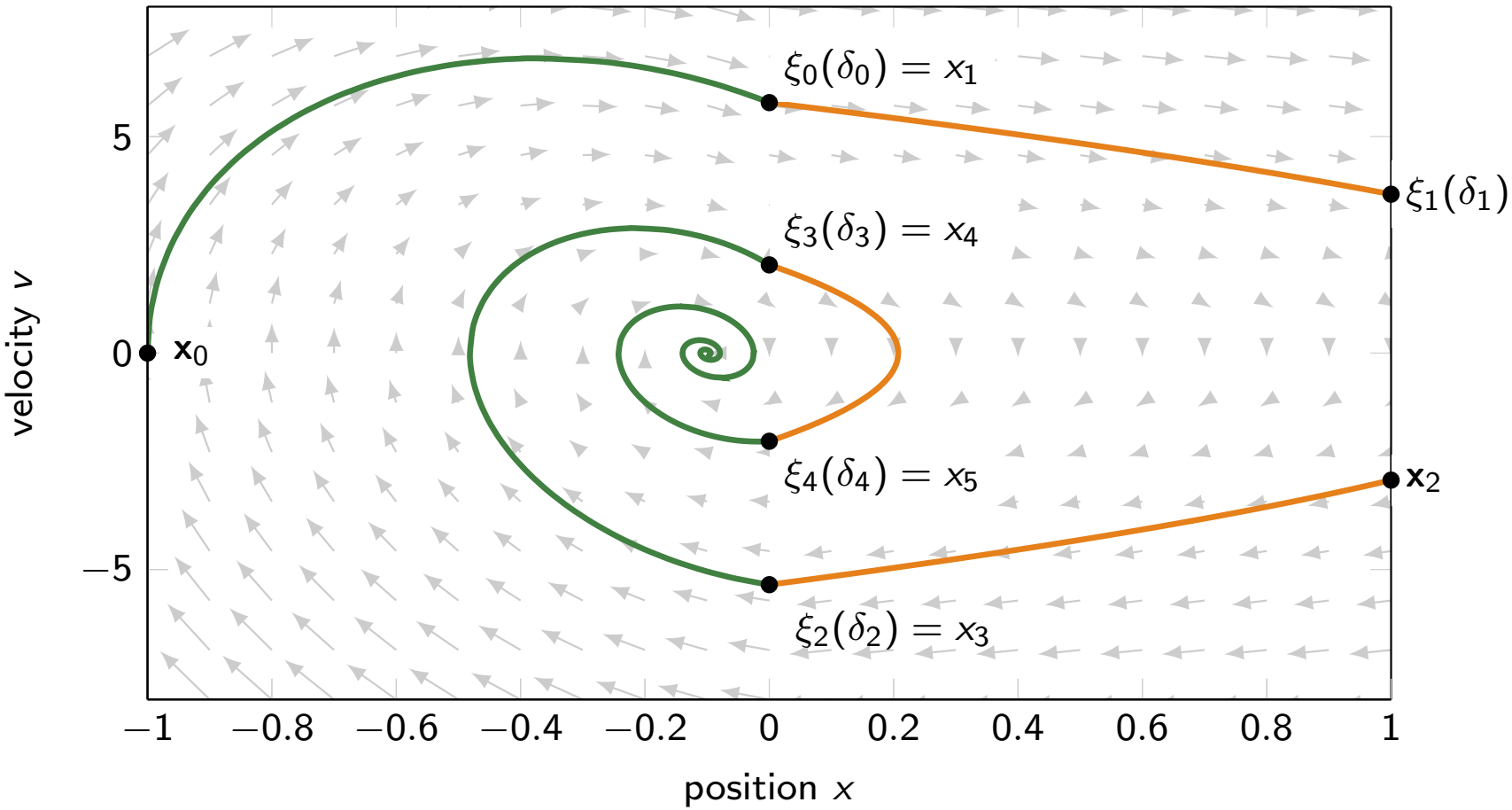
with $(\ell_0, \mathbf{x}_0) \in \mathsf{Init}$, $\alpha_i \in \mathsf{Lab} \cup \{\tau\}$, and for $i = 0, 1, \dots$:

1. Trajectories: $(\dot{\xi}(t), \xi(t)) \in \mathsf{Flow}(\ell)$ and $\xi_i(t) \in \mathsf{Inv}(\ell_i)$ for all $t \in [0, \delta_i]$.
2. Jumps: $(\xi_i(\delta_i), \mathbf{x}_{i+1}) \in \mathsf{Jump}(e_i)$, $e_i = (\ell_i, \alpha_i, \ell_{i+1}) \in \mathsf{Edg}$, and $\mathbf{x}_{i+1} \in \mathsf{Inv}(\ell_{i+1})$.

A state $(\ell, \mathbf{x})$ is reachable if there exists a run with $(\ell_i, \mathbf{x}_i) = (\ell, \mathbf{x})$ for some i.

Usually: $\mathsf{Flow}(\ell)$ is ODE and trajectory $\xi(t)$ its solution

Example: Ball on String



Outline

Set-Based Reachability

Extending numerical simulation from numbers to sets

- account for nondeterminism
- exhaustive
- infinite time horizon

Downsides:

- only approximate for complex dynamics
- generally not scalable in number of variables
- trade-off between runtime and accuracy
- may not terminate or degenerate to "system could be anywhere"

Reachability Algorithm

One-step successors by time elapse from set of states S,

$$\mathsf{Post}_C(S) = \left\{(\ell, \xi(\delta)) \;\middle|\; \exists (\ell, x) \in S : (\ell, \mathbf{x}) \xrightarrow{\delta,\xi} (\ell, \xi(\delta))\right\}$$

One-step successors by jump from set of states S,

$$\mathsf{Post}_D(S) = \left\{(\ell', \mathbf{x}') \;\middle|\; \exists (\ell', \mathbf{x}') \in S, \exists \alpha \in \mathsf{Lab} \cup \{\tau\} : (\ell, \mathbf{x}) \xrightarrow{\alpha} (\ell', \mathbf{x}')\right\}$$

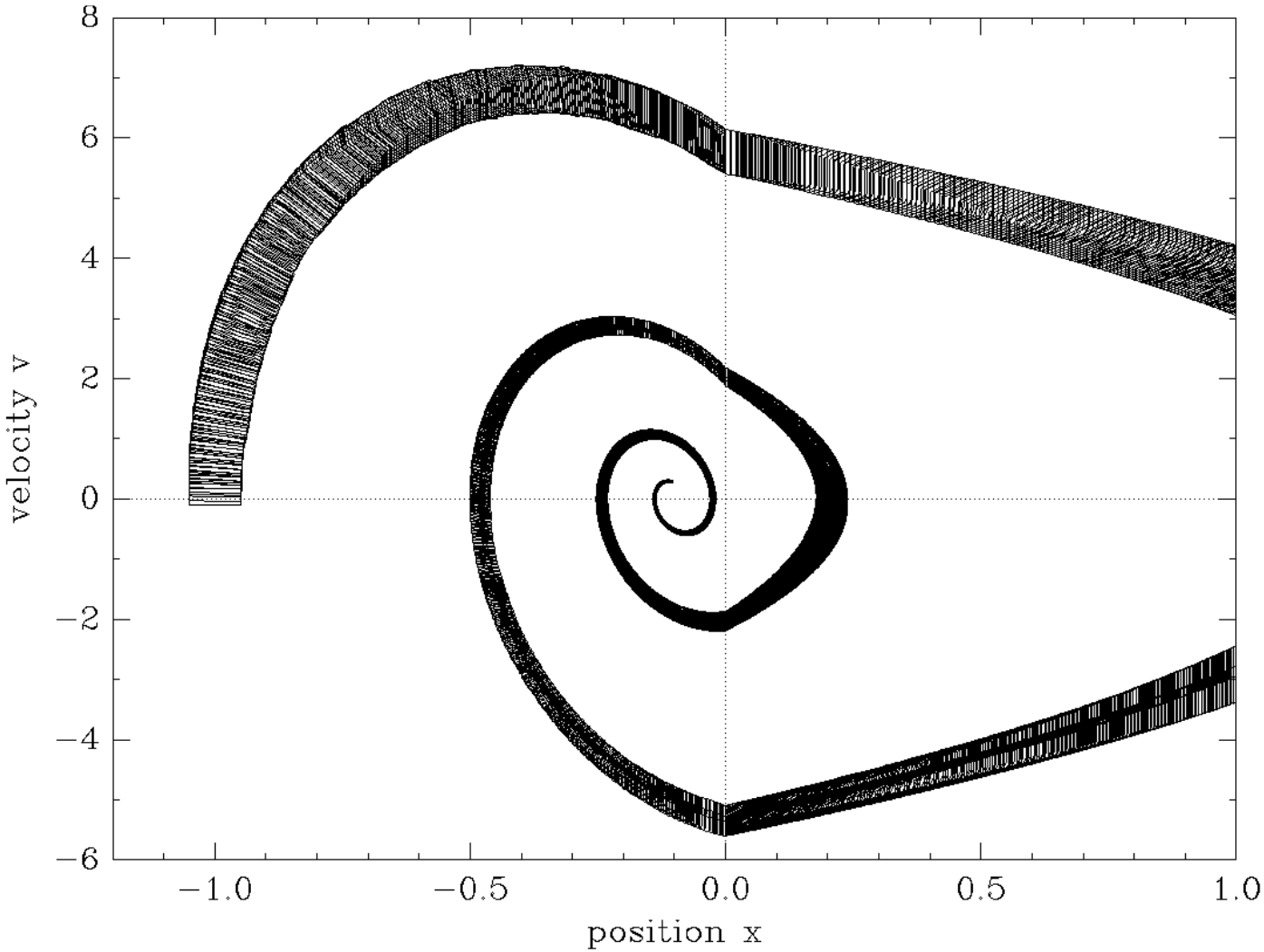
$$\begin{aligned} R_0 &:= \mathsf{Post}_C(\mathsf{Init}), \\ R_{i+1} &:= R_i \cup \mathsf{Post}_C(\mathsf{Post}_D(R_i)). \end{aligned}$$

If $R_{i+1} = R_i$, then $R_i =$ reachable states.

- may not terminate if states unbounded (counter)
- problem undecidable in general[2]

[2] T. A. Henzinger, P. W. Kopke, A. Puri, and P. Varaiya, "What's decidable about hybrid automata?" *Journal of Computer and System Sciences*, vol. 57, pp. 94–124, 1998.

Ball on String: Reachable States



(clip from SpaceEx output)

Outline

HA of piecewise constant dynamics (PCDA,LHA)

- initial states and invariants are conjunctions of linear constraints

$$x_r \leq x \wedge x \leq x_r + L$$

- flows are conjunctions of linear constraints over derivatives $\dot{X}$

$$\dot{x} = 5 \wedge \dot{v} = -1 \wedge 2 \leq \dot{z} \wedge \dot{z} \leq 5$$

- jumps are conjunctive linear constraints over $X \cup X^+$, where X^+ denote the variables after the jump.

$$x = x_r + L \wedge v > 0 \wedge v^+ = -0.5 * v$$

One-step successors of PCDA can be computed exactly.
Often: assume all constraints are compact or at least closed.

Polyhedra in Constraint Form

Conjunctive linear constraints are polyhedra

$\mathcal{H}$-polyhedron (constraint form)

$$\mathcal{P} = \left\{ \mathbf{x} \;\middle|\; \bigwedge_{i=1}^{m} \mathbf{a}_i^{\mathsf{T}} \mathbf{x} \leq b_i \right\},$$

with facet normals $\mathbf{a}_i \in \mathbb{R}^n$ and inhomogeneous coefficients $b_i \in \mathbb{R}$.
vector-matrix notation:

$$\mathcal{P} = \left\{ \mathbf{x} \;\middle|\; A\mathbf{x} \leq \mathbf{b} \right\}, \text{ with } A = \begin{pmatrix} \mathbf{a}_1^{\mathsf{T}} \\ \vdots \\ \mathbf{a}_m^{\mathsf{T}} \end{pmatrix}, \mathbf{b} = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}.$$

Time Elapse with Polyhedra

For PCDA, it suffices to consider straight-line trajectories:[3]

Lemma (Constant Derivatives)

There is a trajectory $\xi(t)$ *from* $\mathbf{x} = \xi(0)$ *to* $\mathbf{x}' = \xi(\delta)$, $\delta > 0$, *iff* $\eta(t) = \mathbf{x} + \mathbf{q}t$ *with* $\mathbf{q} = (\mathbf{x}' - \mathbf{x})/\delta$ *is a trajectory from* $\mathbf{x}$ *to* $\mathbf{x}'$.

[3] P.-H. Ho, "Automatic analysis of hybrid systems," Technical Report CSD-TR95-1536, PhD thesis, Cornell University, Aug. 1995.

Time Elapse with Polyhedra

For PCDA, it suffices to consider straight-line trajectories:[3]

Lemma (Constant Derivatives)

There is a trajectory $\xi(t)$ *from* $\mathbf{x} = \xi(0)$ *to* $\mathbf{x}' = \xi(\delta)$, $\delta > 0$, *iff* $\eta(t) = \mathbf{x} + \mathbf{q}t$ *with* $\mathbf{q} = (\mathbf{x}' - \mathbf{x})/\delta$ *is a trajectory from* $\mathbf{x}$ *to* $\mathbf{x}'$.

Proof Idea: Average slope is enough by mean-value theorem

[3] P.-H. Ho, "Automatic analysis of hybrid systems," Technical Report CSD-TR95-1536, PhD thesis, Cornell University, Aug. 1995.

Time Elapse with Polyhedra

Given polyhedra $\mathcal{P} = \{\mathbf{x} \mid A\mathbf{x} \leq \mathbf{b}\}$, $\mathcal{Q} = \{\mathbf{q} \mid \bar{A}\mathbf{q} \leq \bar{\mathbf{b}}\}$
Time successors with constant slope $\in \mathcal{Q}$ (ignores evolution domain):

$$\mathcal{P} \nearrow \mathcal{Q} = \{\mathbf{x}' \mid \mathbf{x} \in \mathcal{P}, \mathbf{q} \in \mathcal{Q}, t \in \mathbb{R}^{\geq 0}, \mathbf{x}' = \mathbf{x} + \mathbf{q}t\}.$$

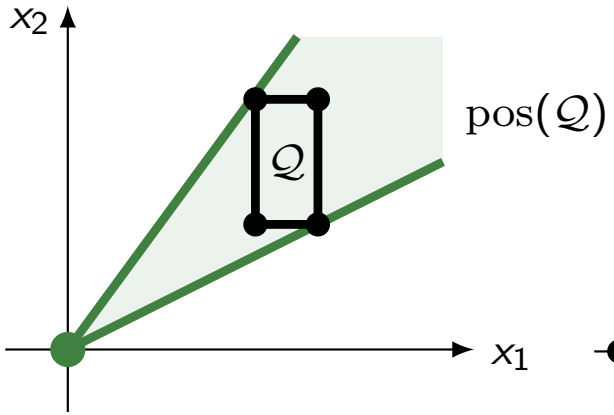
Eliminating $\mathbf{q} = \frac{\mathbf{x}' - \mathbf{x}}{t}$ for $t > 0$, plug into $\mathcal{Q}$, and multiplying with t:

$$\mathcal{P} \nearrow \mathcal{Q} = \left\{\mathbf{x}' \;\middle|\; A\mathbf{x} \leq \mathbf{b} \;\wedge\; \bar{A}(\mathbf{x}' - \mathbf{x}) \leq \bar{\mathbf{b}} \cdot t \;\wedge\; t \geq 0\right\}.$$

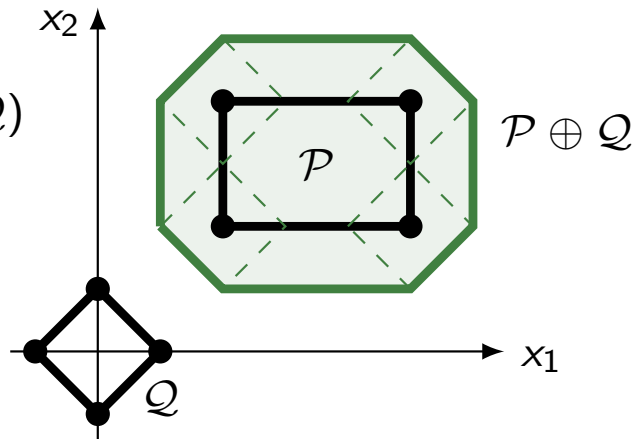
Quantifier elimination of t squares the number of constraints. FM
More general union of two polyhedra if $\mathcal{Q}$ not compact.
Subsequently intersect with evolution domain Inv:

$$\text{post}_C(\ell \times P) = \ell \times \big(P \nearrow \text{Flow}(\ell)\big) \cap \text{Inv}(\ell).$$

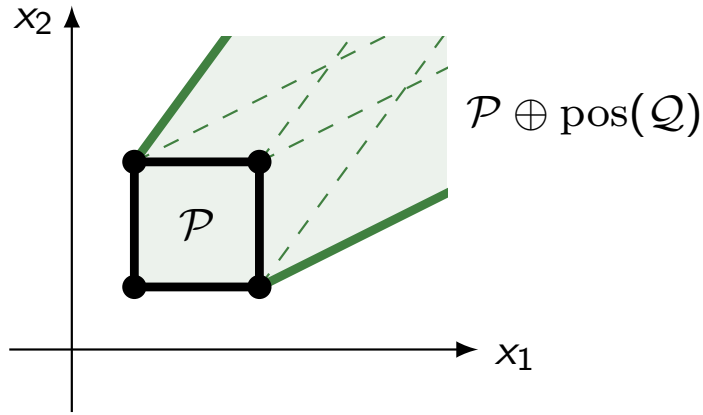
Time Elapse with Polyhedra – Geometric Version



(a) cone $\text{pos}(\mathcal{Q})$



(b) $\mathcal{P} \oplus \mathcal{Q}$



(c) $\mathcal{P} \nearrow \mathcal{Q} = \mathcal{P} \oplus \text{pos}(\mathcal{Q})$

cone around $\mathcal{Q}$

$$\text{pos}(\mathcal{Q}) = \{\mathbf{q} \cdot t \mid \mathbf{q} \in \mathcal{Q}, t \geq 0\}$$

Minkowski sum

$$\mathcal{P} \oplus \mathcal{Q} = \{\mathbf{p} + \mathbf{q} \mid \mathbf{p} \in \mathcal{P}, \mathbf{q} \in \mathcal{Q}\}$$

convex hull

$$\text{chull}(\mathcal{Q}) = \left\{ \sum_{\mathbf{q}_i \in \mathcal{Q}} \lambda_i \cdot \mathbf{q}_i \;\middle|\; \lambda_i \geq 0, \sum_i \lambda_i = 1 \right\}$$

Polyhedra in Generator Form

$\mathcal{H}$-polyhedron (constraint form)

$$\mathcal{P} = \left\{ \mathbf{x} \;\middle|\; A\mathbf{x} \leq \mathbf{b} \right\}, \text{ with } A = \begin{pmatrix} \mathbf{a}_1^\mathsf{T} \\ \vdots \\ \mathbf{a}_m^\mathsf{T} \end{pmatrix}, \mathbf{b} = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}.$$

$\mathcal{V}$-polyhedron (generator form)

$$\mathcal{P} = (V, R) = \text{chull}\,(V) \oplus \text{pos}(\text{chull}(R)).$$

with vertices $V \subseteq \mathbb{R}^n$ and rays $R \subseteq \mathbb{R}^n$
If $\mathcal{P} = (V, R)$ and $\mathcal{Q} = (V', R')$ in **generator form** and closed, then

$$\mathcal{P} \nearrow \mathcal{Q} = (V, R) \nearrow (V', R') = (V, R \cup V' \cup R')$$

But: conversion between $\mathcal{H}$- and $\mathcal{V}$-polyhedra is expensive
cube: $2n$ constraints, 2^n vertices
cross-polytope (diamond): $2n$ vertices, 2^n constraints

Discrete Successors

Edge $e = (\ell, \alpha, \ell')$ with guard $\mathbf{x} \in \mathcal{G}$ and nondeterministic assignment $\mathbf{x}^+ = C\mathbf{x} + \mathbf{w}$, $\mathbf{w} \in \mathcal{W}$,

$$\text{post}_D(\ell \times P) = \ell' \times \left(C(\mathcal{P} \cap \mathcal{G}) \oplus \mathcal{W}\right) \cap \text{Inv}(\ell').$$

If constant matrix C invertible and everything in **constraint form**

$$C\mathcal{P} = \{\mathbf{x} \mid AC^{-1}\mathbf{x} \leq b\} \qquad \text{where} \qquad \mathcal{P} = \left\{\mathbf{x} \;\middle|\; A\mathbf{x} \leq \mathbf{b}\right\}$$

Otherwise requires quantifier elimination

Computational Cost

operation		polyhedra	
		m constraints	k generators
cone	$\mathrm{pos}()$	m^2	k
Minkowski sum	$\oplus$	exp	k^2
linear map	$Ax+b$	m / exp	k
intersection	$\cap$	$2m$	exp

Generator form good for continuous successors $\mathcal{P} \nearrow \mathcal{Q} = \mathcal{P} \oplus \mathrm{pos}(\mathcal{Q})$
Constraint form is better for discrete successors if no uncertainty
Conversion is expensive

Outline

Piecewise Affine Dynamics

Hybrid automata with piecewise affine dynamics (PWA)

- initial states and invariants are conjunctive linear constraints over X alias polyhedra

$$x_r \leq x \wedge x \leq x_r + L$$

- flows are affine ODEs with compact convex polyhedra $\mathcal{U}$

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}, \qquad \mathbf{u} \in \mathcal{U},$$

- jumps have a guard set and *assignments*

$$\mathbf{x}^+ = C\mathbf{x} + \mathbf{w}, \qquad \mathbf{w} \in \mathcal{W}.$$

Continuous successors

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}, \qquad \mathbf{u} \in \mathcal{U},$$

trajectory $\xi(t)$ from $\xi(0) = \mathbf{x}_0$ for integrable input signal $\zeta(t) \in \mathcal{U}$:

$$\xi_{\mathbf{x}_0,\zeta}(t) = e^{At}\mathbf{x}_0 + \int_0^t e^{A(t-s)} B\zeta(s) ds$$

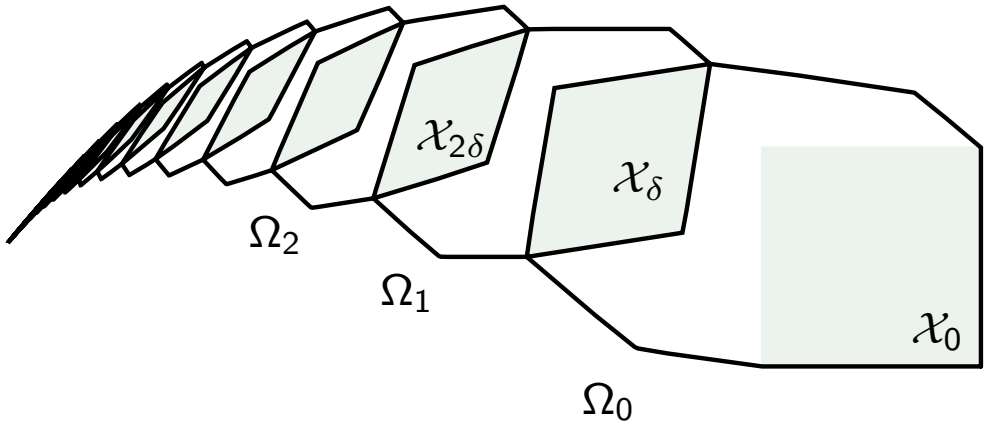
superposition with "backdated impact $e^{A(t-s)}$ of input $\zeta(s)$ at s"

reachable states from set $\mathcal{X}_0$ for any input signal:

$$\mathcal{X}_t = e^{At}\mathcal{X}_0 \oplus \mathcal{Y}_t$$

$$\mathcal{Y}_t = \int_0^t e^{As}\mathcal{U} ds = e^{At}\mathcal{X}_0 \oplus \lim_{\delta \to 0} \bigoplus_{k=0}^{\lfloor t/\delta \rfloor} e^{A\delta k}\delta\mathcal{U}$$

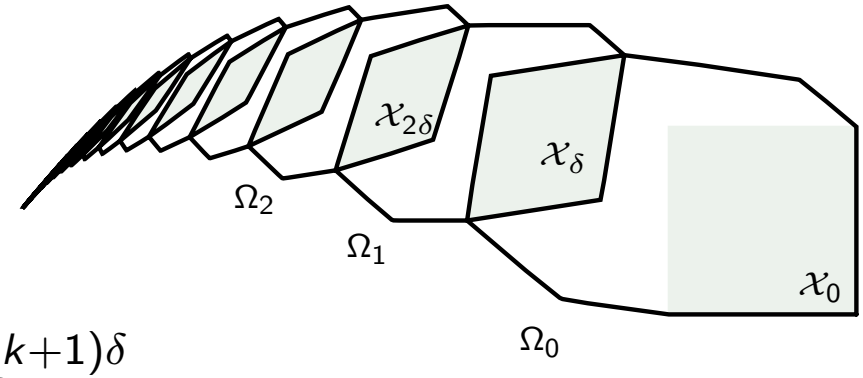
Computing a Convex Cover



Compute $\Omega_0, \Omega_1, \ldots$ for fixed time horizon T such that

$$\bigcup_{0 \leq t \leq T} \mathcal{X}_t \subseteq \Omega_0 \cup \Omega_1 \cup \ldots.$$

Time Discretization at Step-size δ



Semi-group property: $(\mathcal{X}_{k\delta})_{\delta} = \mathcal{X}_{(k+1)\delta}$

Time discretization: $\mathcal{X}_{(k+1)\delta} = e^{A\delta}\mathcal{X}_{k\delta} \oplus \mathcal{Y}_{\delta}$

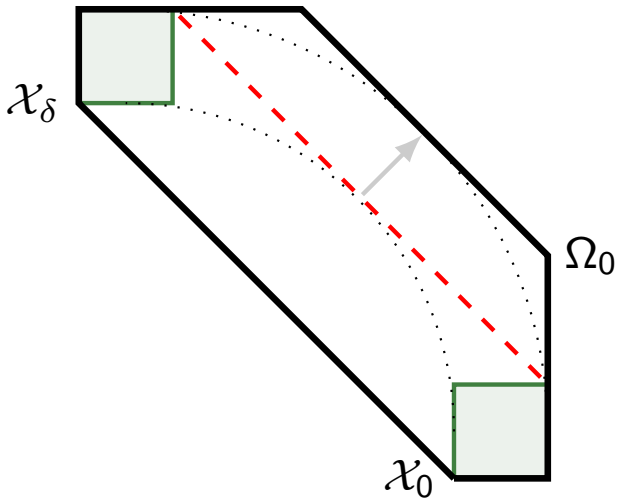
Given initial approximations Ω_0 and Ψ_{δ} for the first δ such that

$$\bigcup_{0 \le t \le \delta} \mathcal{X}_t \subseteq \Omega_0, \qquad \mathcal{Y}_{\delta} \subseteq \Psi_{\delta},$$

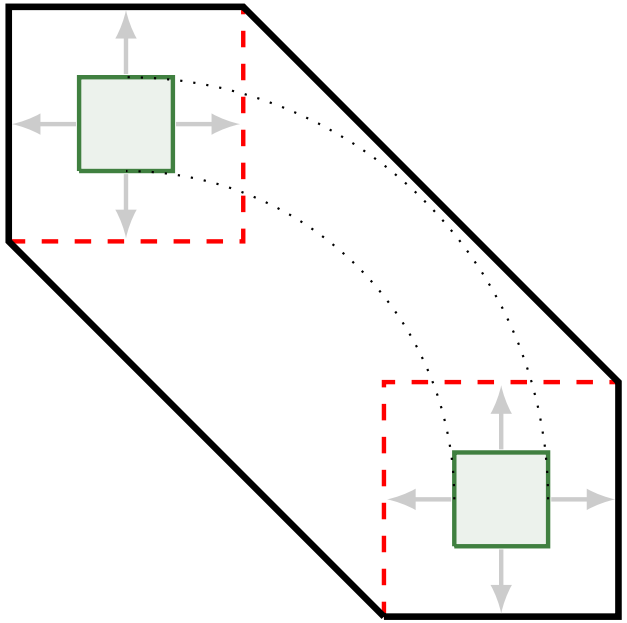
$\mathcal{X}_t$ is covered by the sequence

$$\Omega_{k+1} = e^{A\delta}\Omega_k \oplus \Psi_{\delta}.$$

Initial Approximations



(a) convex hull and pushing facets



(b) convex hull and bloating

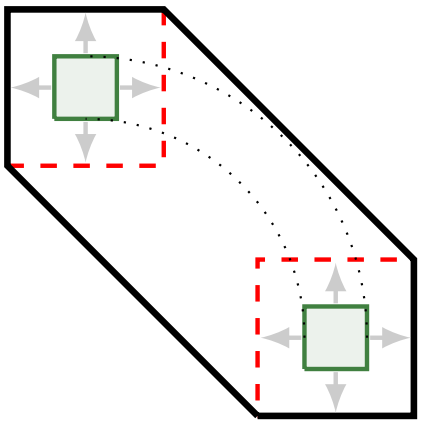
By Taylor approximation or by solving optimization problems

Initial Approximations – Forward Bloating

Bloating based on norms bounding Taylor approximation error:[4]

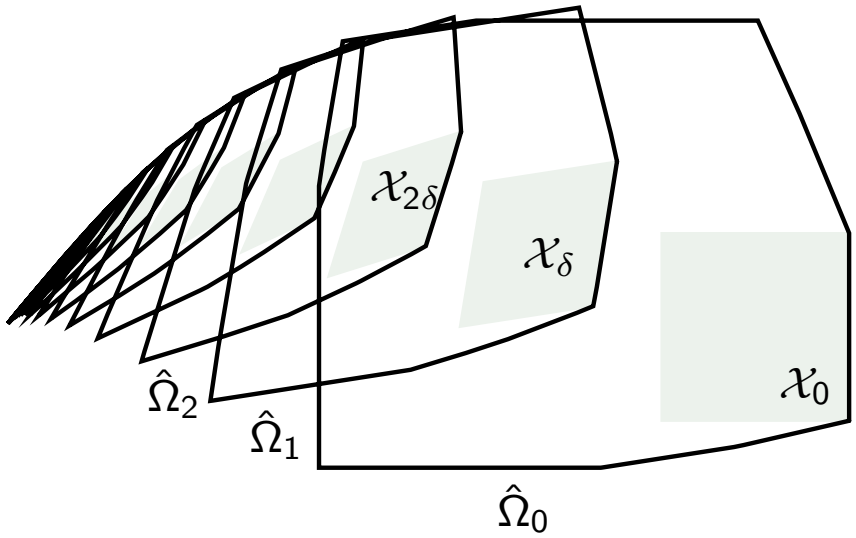
$$\begin{aligned}
\Omega_0 &= \mathrm{chull}(\mathcal{X}_0 \cup e^{A\delta}\mathcal{X}_0) \oplus (\alpha_\delta + \beta_\delta)\mathcal{B}, \\
\Psi_\delta &= \beta_\delta \mathcal{B}, \\
\alpha_\delta &= \mu(\mathcal{X}_0) \cdot (e^{\|A\|\delta} - 1 - \|A\|\delta), \\
\beta_\delta &= \tfrac{1}{\|A\|}\mu(B\mathcal{U}) \cdot (e^{\|A\|\delta} - 1),
\end{aligned}$$

with radius $\mu(\mathcal{X}) = \max_{x \in \mathcal{X}} \|x\|$ and unit ball $\mathcal{B}$.



[4] A. Girard, "Reachability of uncertain linear systems using zonotopes," in *HSCC*, 2005, pp. 291–305.

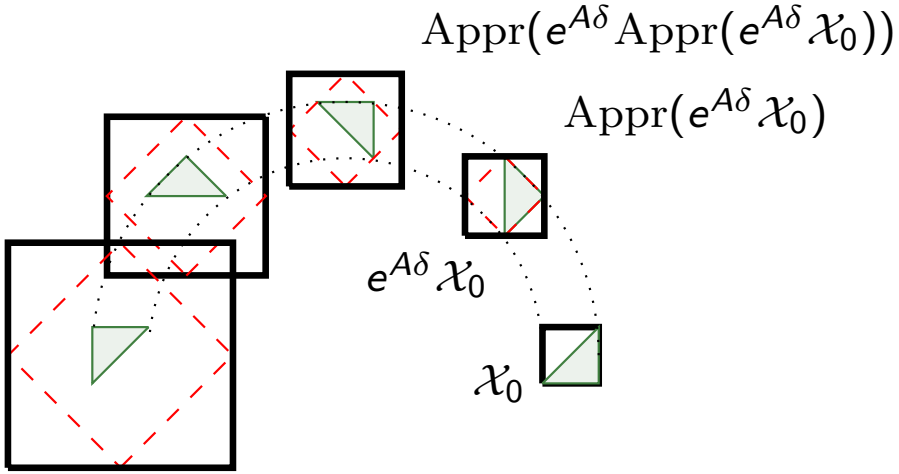
Initial Approximations – Forward Bloating



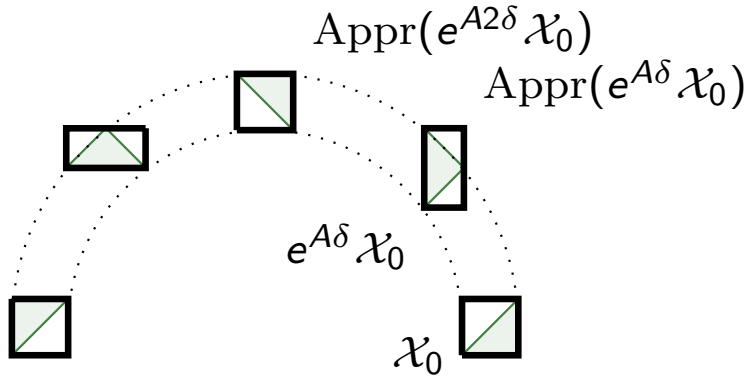
Forward bloating is tight on $\mathcal{X}_0$ and bloated on $\mathcal{X}_\delta$.
Improvements:

- intersect forward bloating with backward bloating
- bloat based on interpolation error

Wrapping Effect: Error Accumulation



(a) with wrapping effect



(b) with wrapping-free algorithm

Avoid increasing complexity through deliberate over-approximation

$$\hat{\Omega}_{k+1} = \mathrm{Appr}(e^{A\delta}\hat{\Omega}_k \oplus \Psi_\delta).$$

Example: bounding box over-approximation is fast

Wrapping Effect: Wrapping-free Cases

Idea: Over-approximate each image of initial approximation separately
Solution: Split sequence[5]

$$\begin{aligned} \hat{\Psi}_{k+1} &= \mathrm{Appr}(e^{Ak\delta}\Psi_\delta) \oplus \hat{\Psi}_k \quad \text{with } \hat{\Psi}_0 = \{0\}, \\ \hat{\Omega}_k &= \mathrm{Appr}(e^{Ak\delta}\Omega_0) \oplus \hat{\Psi}_k \quad \text{from initial } \Omega_0 \end{aligned}$$

satisfies $\hat{\Omega}_k = \mathrm{Appr}(\Omega_k)$ (wrapping-free) if

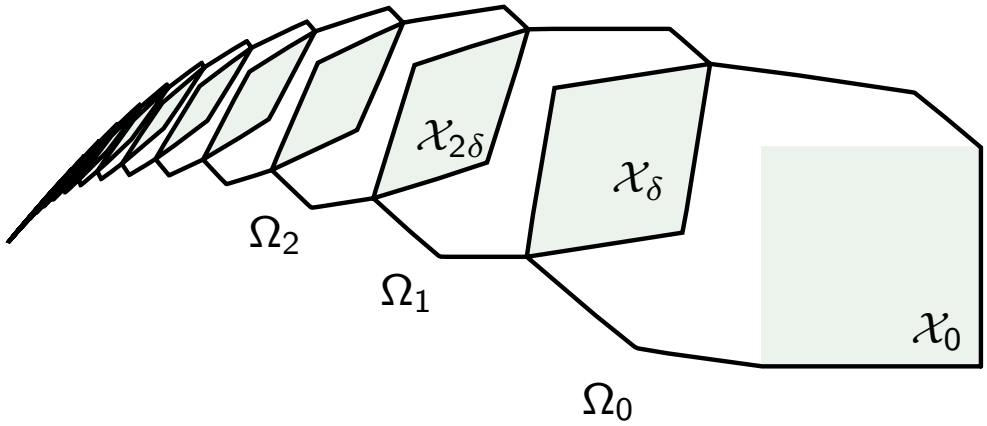
$$\mathrm{Appr}(\mathcal{P} \oplus \mathcal{Q}) = \mathrm{Appr}(\mathcal{P}) \oplus \mathrm{Appr}(\mathcal{Q}),$$

e.g., bounding box.

[5] A. Girard, C. L. Guernic, and O. Maler, "Efficient computation of reachable sets of linear time-invariant systems with inputs," in *HSCC*, 2006, pp. 257–271.

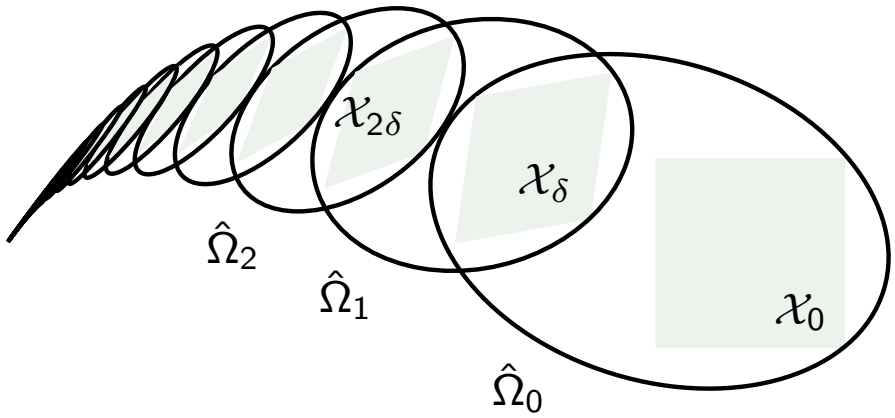
Outline

Polyhedra



operation		polyhedra m constr.	k gen.
convex hull	chull	exp	$2k$
Minkowski sum	$\oplus$	exp	k^2
linear map	$Ax+b$	m / exp	k
intersection	$\cap$	$2m$	exp

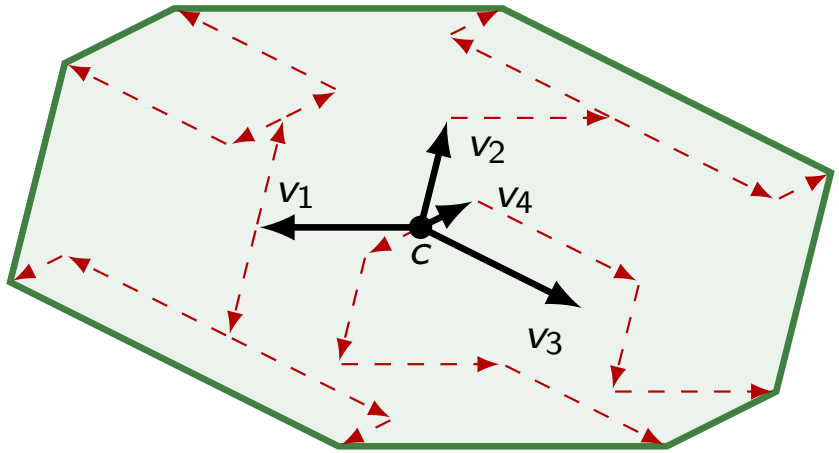
Ellipsoids[6]



operation		polyhedra m constr.	polyhedra k gen.	ellipsoids $n \times n$ matrix
convex hull	chull	exp	$2k$	approx
Minkowski sum	$\oplus$	exp	k^2	approx
linear map	$Ax+b$	m / exp	k	$\mathcal{O}(n^3)$
intersection	$\cap$	$2m$	exp	approx

[6] A. B. Kurzhanski and P. Varaiya, *Dynamics and Control of Trajectory Tubes*. Springer, 2014.

Zonotopes

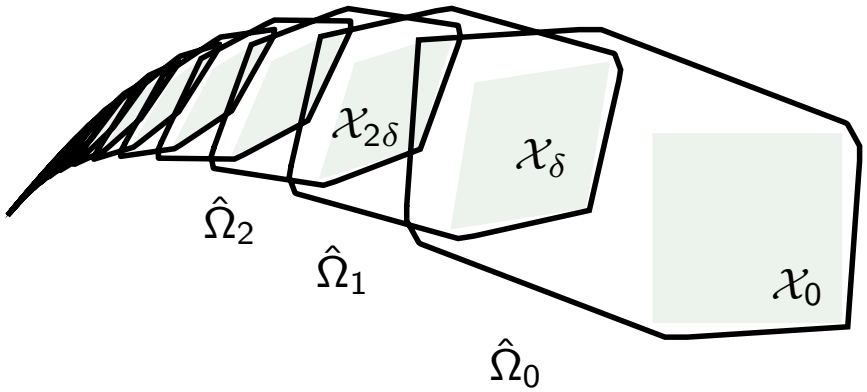


Zonotope with center $\mathbf{c} \in \mathbb{R}^n$ and generators $\mathbf{v}_1, \ldots, \mathbf{v}_k \in \mathbb{R}^n$

$$\mathcal{P} = \left\{ \mathbf{c} + \sum_{i=1}^{k} \alpha_i \mathbf{v}_i \;\middle|\; \alpha_i \in [-1, 1] \right\}.$$

linear map: $(A\mathbf{c}, \langle A\mathbf{v}_1, \ldots, A\mathbf{v}_k \rangle)$
Minkowski sum: $\mathcal{P} \oplus \mathcal{Q} = (\mathbf{c} + \mathbf{d}, \langle \mathbf{v}_1, \ldots, \mathbf{v}_k, \mathbf{w}_1, \ldots, \mathbf{w}_m \rangle)$
Not closed under chull or $\cap$ so need approximations

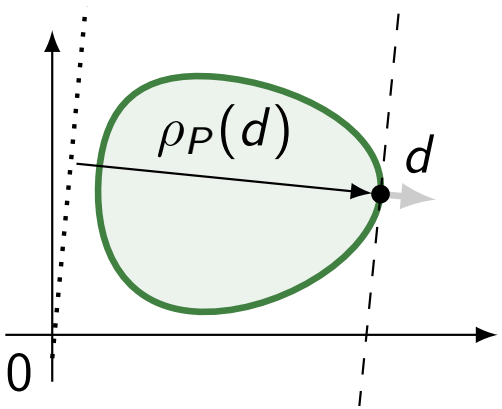
Zonotopes[7]



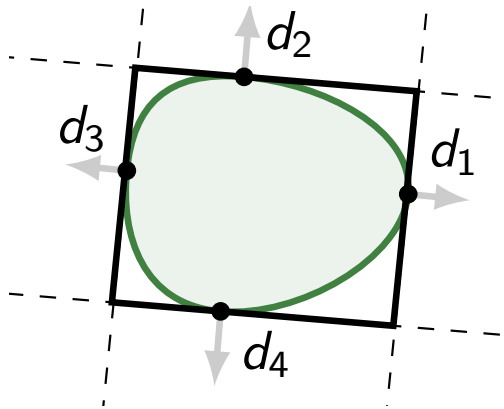
operation		polyhedra m constr.	polyhedra k gen.	ellipsoids $n \times n$ matrix	zonotopes k generators
convex hull	chull	exp	$2k$	approx	approx
Minkowski sum	$\oplus$	exp	k^2	approx	$2k$
linear map	$Ax+b$	m / exp	k	$\mathcal{O}(n^3)$	k
intersection	$\cap$	$2m$	exp	approx	approx

[7] A. Girard, "Reachability of uncertain linear systems using zonotopes," in *HSCC*, 2005, pp. 291–305.

Support Functions of Compact Convex Set $\mathcal{P}$



(a) support function in direction d



(b) outer approximation

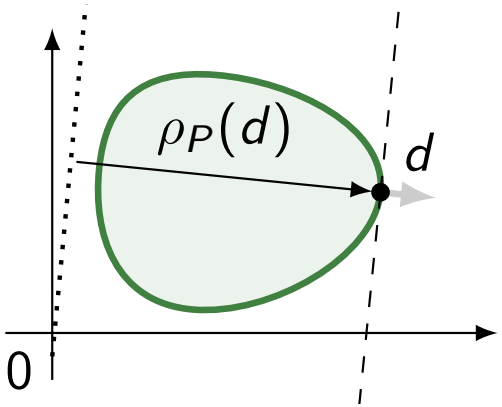
support function $\rho_{\mathcal{P}}(\cdot)$ by linear optimization (efficient!)

$$\rho_{\mathcal{P}}(\mathbf{d}) = \max\{\mathbf{d}^{\mathsf{T}}\mathbf{x} \mid \mathbf{x} \in \mathcal{P}\}$$

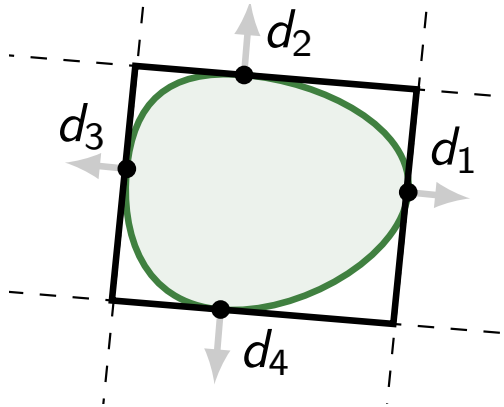
computed values define polyhedral outer approximation

$$\lceil \mathcal{P} \rceil_{\mathcal{D}} = \bigcap_{\mathbf{d} \in \mathcal{D}} \{\mathbf{d}^{\mathsf{T}} x \leq \rho_{\mathcal{P}}(\mathbf{d})\}$$

Support Functions



(a) support function in direction d



(b) outer approximation

linear map: $\rho_{A\mathcal{X}}(\ell) = \rho_{\mathcal{X}}(A^{\mathsf{T}}\ell)$ $\mathcal{O}(mn)$

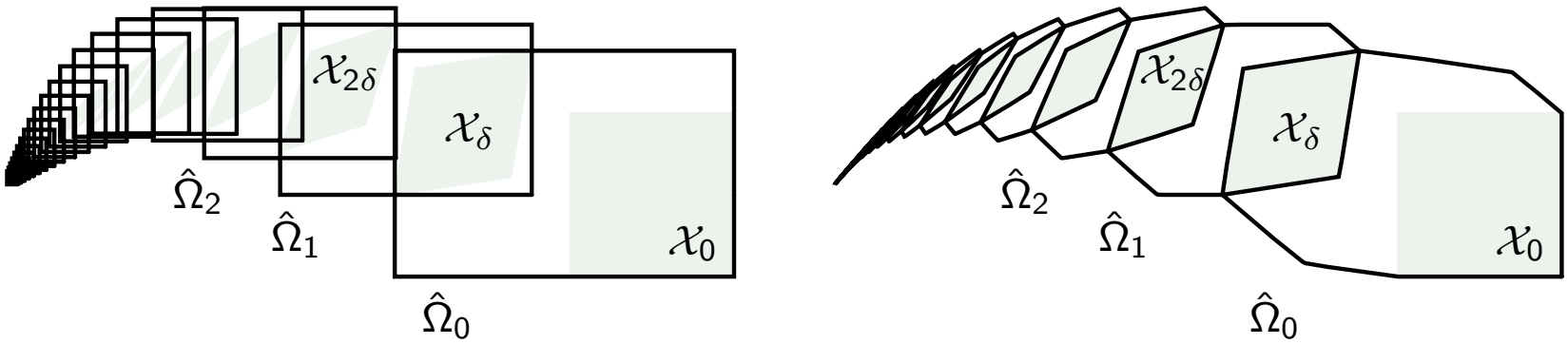
convex hull: $\rho_{\text{chull}(\mathcal{P}\cup\mathcal{Q})}(\ell) = \max\{\rho_{\mathcal{P}}(\ell), \rho_{\mathcal{Q}}(\ell)\}$ $\mathcal{O}(1)$

Minkowski sum: $\rho_{\mathcal{X}\oplus\mathcal{Y}}(\ell) = \rho_{\mathcal{X}}(\ell) + \rho_{\mathcal{Y}}(\ell)$ $\mathcal{O}(1)$

Intersection: expressible as optimization problem

Support Functions

(Le Guernic, Girard,'09)[9]



support functions: lazy approximation on demand

operation		polyhedra m constr.	polyhedra k gen.	ellipsoids $n \times n$ matrix	zonotopes k generators	support f. —
convex hull	chull	exp	$2k$	approx	approx	$\mathcal{O}(1)$
Minkowski sum	$\oplus$	exp	k^2	approx	$2k$	$\mathcal{O}(1)$
linear map	$Ax+b$	m / exp	k	$\mathcal{O}(n^3)$	k	$\mathcal{O}(n^2)$
intersection	$\cap$	$2m$	exp	approx	approx	opt. / approx

Outline

Conclusions

- Hybrid automata are challenging for model checking.
- Set-based reachability is exhaustive, sufficient for safety and bounded liveness.
  - expensive, scalable for piecewise affine dynamics
- Abstraction lifts reachability to more complex systems
  - progress with approximate simulation relations
- Verification by numerical simulation extends properties from traces to sets of states
  - sampling of initial states limited to low dimensional sets

References

[2] R. Alur, C. Courcoubetis, N. Halbwachs, T. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine, "The algorithmic analysis of hybrid systems," *Theoretical Computer Science*, vol. 138, pp. 3–34, 1995.

[3] T. A. Henzinger, "The theory of hybrid automata.," in *LICS*, Los Alamitos: IEEE Computer Society, 1996, pp. 278–292.

[9] C. Le Guernic and A. Girard, "Reachability analysis of linear systems using support functions," *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 2, pp. 250–262, 2010.