On The Complexity of Bounded Time Reachability for Piecewise Affine Systems*

H. Bazille[3] O. Bournez[1] W. Gomaa[2,4] A. Pouly[1]

[1]École Polytechnique, LIX, 91128 Palaiseau Cedex, France

[2]Egypt Japan University of Science and Technology, CSE, Alexandria, Egypt

[3]ENS Cachan/Bretagne et Université Rennes 1, France

[4]Faculty of Engineering, Alexandria University, Alexandria, Egypt

September 23, 2014

*This work was partially supported by DGA Project CALCULS.

Outline

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i = \text{convex polyhedron} = \{x \mid M_i x \leqslant v_i\}$ $\qquad M_i \in \mathbb{Q}^{d \times d}, v_i \in \mathbb{Q}^d$

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i = \text{convex polyhedron} = \{x \mid M_i x \leqslant v_i\}$ $\qquad M_i \in \mathbb{Q}^{d \times d}, v_i \in \mathbb{Q}^d$
- piecewise affine function $f : \mathcal{H} \to \mathcal{H}$

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i =$ convex polyhedron $= \{x \mid M_i x \leqslant v_i\}$ $\quad M_i \in \mathbb{Q}^{d \times d}, v_i \in \mathbb{Q}^d$
- piecewise affine function $f : \mathcal{H} \to \mathcal{H}$
  $f(x) = A_i x + b_i \quad$ for $x \in \mathcal{H}_i$ $\quad A_i \in \mathbb{Q}^{d \times d}, b_i \in \mathbb{Q}^d$

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i =$ convex polyhedron $= \{x \mid M_i x \leqslant v_i\}$ $\qquad M_i \in \mathbb{Q}^{d\times d}, v_i \in \mathbb{Q}^d$
- piecewise affine function $f : \mathcal{H} \to \mathcal{H}$
  $f(x) = A_i x + b_i \quad$ for $x \in \mathcal{H}_i$ $\qquad A_i \in \mathbb{Q}^{d\times d}, b_i \in \mathbb{Q}^d$
- trajectory: $x, f(x), f^{[2]}(x), \ldots, f^{[i]}(x), \ldots$

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i =$ convex polyhedron $= \{x \mid M_i x \leqslant v_i\}$ $\quad M_i \in \mathbb{Q}^{d \times d}, v_i \in \mathbb{Q}^d$
- piecewise affine function $f : \mathcal{H} \to \mathcal{H}$
  $f(x) = A_i x + b_i \quad$ for $x \in \mathcal{H}_i$ $\quad A_i \in \mathbb{Q}^{d \times d}, b_i \in \mathbb{Q}^d$
- trajectory: $x, f(x), f^{[2]}(x), \ldots, f^{[i]}(x), \ldots$

$\Rightarrow$ Discrete time dynamical system

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i =$ convex polyhedron $= \{x \mid M_i x \leqslant v_i\}$ $\qquad M_i \in \mathbb{Q}^{d \times d}, v_i \in \mathbb{Q}^d$
- piecewise affine function $f : \mathcal{H} \to \mathcal{H}$
  $f(x) = A_i x + b_i \quad$ for $x \in \mathcal{H}_i$ $\qquad A_i \in \mathbb{Q}^{d \times d}, b_i \in \mathbb{Q}^d$
- trajectory: $x, f(x), f^{[2]}(x), \ldots, f^{[i]}(x), \ldots$

$\Rightarrow$ Discrete time dynamical system

Three cases:

- $\mathbb{K} = \mathbb{N}$: integer case
- $\mathbb{K} = [0, 1]$: continuous bounded case
- $\mathbb{K} = \mathbb{R}$: continuous unbounded case

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i$ = convex polyhedron = $\{x \mid M_i x \leqslant v_i\}$ $\qquad M_i \in \mathbb{Q}^{d\times d}, v_i \in \mathbb{Q}^d$
- piecewise affine function $f : \mathcal{H} \to \mathcal{H}$
  $f(x) = A_i x + b_i$ for $x \in \mathcal{H}_i$ $\qquad A_i \in \mathbb{Q}^{d\times d}, b_i \in \mathbb{Q}^d$
- trajectory: $x, f(x), f^{[2]}(x), \ldots, f^{[i]}(x), \ldots$

$\Rightarrow$ Discrete time dynamical system

Three cases:

- $\mathbb{K} = \mathbb{N}$: integer case $\rightarrow$ Very different from $[0,1]$ and $\mathbb{R}$
- $\mathbb{K} = [0,1]$: continuous bounded case
- $\mathbb{K} = \mathbb{R}$: continuous unbounded case

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i$ = convex polyhedron = $\{x \mid M_i x \leqslant v_i\}$ $\quad M_i \in \mathbb{Q}^{d \times d}, v_i \in \mathbb{Q}^d$
- piecewise affine function $f : \mathcal{H} \to \mathcal{H}$
  $f(x) = A_i x + b_i \quad$ for $x \in \mathcal{H}_i$ $\quad A_i \in \mathbb{Q}^{d \times d}, b_i \in \mathbb{Q}^d$
- trajectory: $x, f(x), f^{[2]}(x), \dots, f^{[i]}(x), \dots$

$\Rightarrow$ Discrete time dynamical system

Three cases:

- $\mathbb{K} = \mathbb{N}$: integer case $\to$ Very different from $[0, 1]$ and $\mathbb{R}$
- $\mathbb{K} = [0, 1]$: continuous bounded case $\to$ Our case
- $\mathbb{K} = \mathbb{R}$: continuous unbounded case

Piecewise Affine System (1)

General Model

- vector space: $\mathcal{H} = \mathbb{K}^d$
- partition of the space: $\mathcal{H} = \cup_{i=1}^{m} \mathcal{H}_i$
  $\mathcal{H}_i$ = convex polyhedron = $\{x \mid M_i x \leqslant v_i\}$ $\qquad M_i \in \mathbb{Q}^{d \times d}, v_i \in \mathbb{Q}^d$
- piecewise affine function $f : \mathcal{H} \to \mathcal{H}$
  $f(x) = A_i x + b_i$ for $x \in \mathcal{H}_i$ $\qquad A_i \in \mathbb{Q}^{d \times d}, b_i \in \mathbb{Q}^d$
- trajectory: $x, f(x), f^{[2]}(x), \ldots, f^{[i]}(x), \ldots$

$\Rightarrow$ Discrete time dynamical system

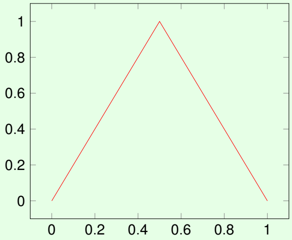
Three cases:

- $\mathbb{K} = \mathbb{N}$: integer case $\to$ Very different from $[0, 1]$ and $\mathbb{R}$
- $\mathbb{K} = [0, 1]$: continuous bounded case $\to$ Our case
- $\mathbb{K} = \mathbb{R}$: continuous unbounded case $\to$ Similar*ish* to $[0, 1]$?

Piecewise Affine System (2)

f continuous

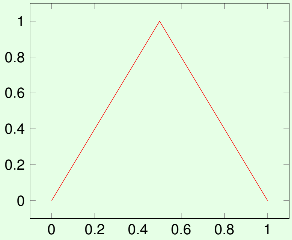
$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Piecewise Affine System (2)

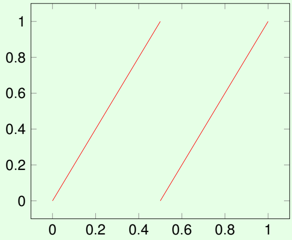
f continuous

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



f discontinuous

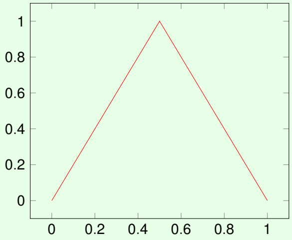
$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}[\\ 2x - 1 & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Piecewise Affine System (2)

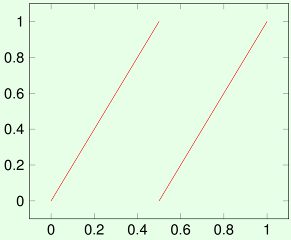
f continuous

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



f discontinuous

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}[\\ 2x - 1 & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$

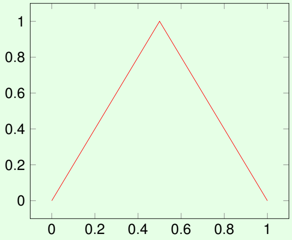


$\rightarrow$ Our case

Piecewise Affine System (2)

f continuous

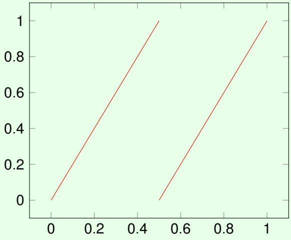
$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



→ Our case

f discontinuous

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}[\\ 2x - 1 & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$

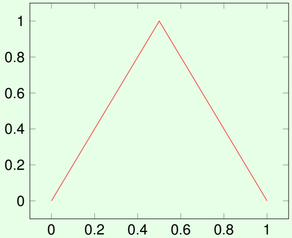


→ Quite different

Example

Function

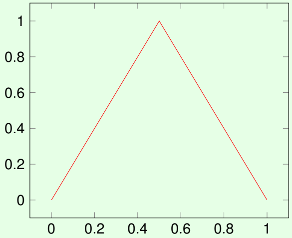
$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



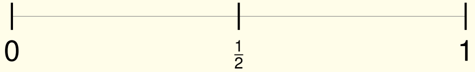
Example

Function

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



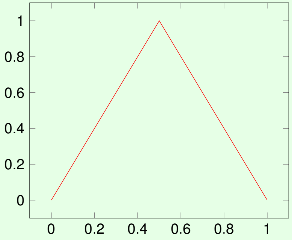
Trajectory



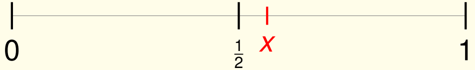
Example

Function

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Trajectory

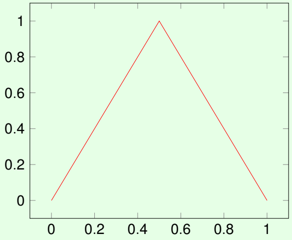


$x = 0.5625$

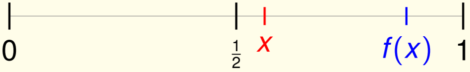
Example

Function

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Trajectory

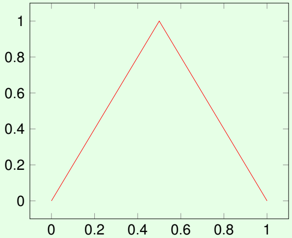


$$x = 0.5625$$
$$f(x) = 0.875$$

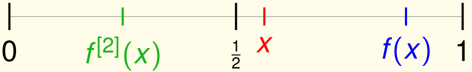
Example

Function

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Trajectory



$$x = 0.5625$$

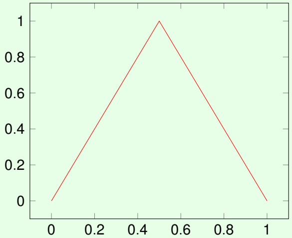
$$f(x) = 0.875$$

$$f^{[2]}(x) = 0.25$$

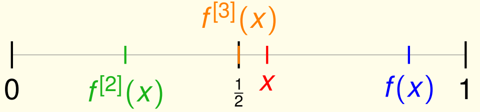
Example

Function

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Trajectory



$$x = 0.5625$$

$$f(x) = 0.875$$

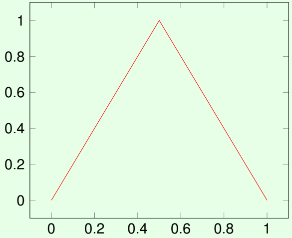
$$f^{[2]}(x) = 0.25$$

$$f^{[3]}(x) = 0.5$$

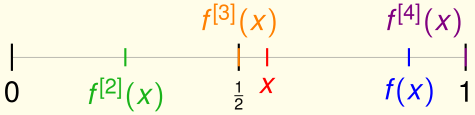
Example

Function

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Trajectory



$$x = 0.5625$$

$$f(x) = 0.875$$

$$f^{[2]}(x) = 0.25$$

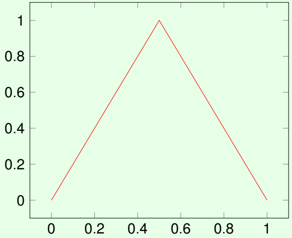
$$f^{[3]}(x) = 0.5$$

$$f^{[4]}(x) = 1$$

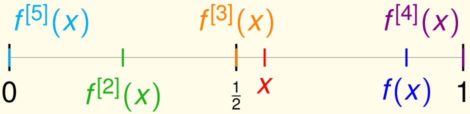
Example

Function

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Trajectory

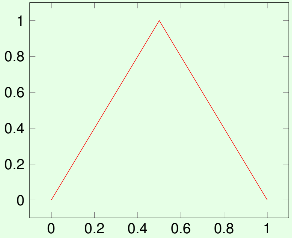


$$x = 0.5625$$
$$f(x) = 0.875$$
$$f^{[2]}(x) = 0.25$$
$$f^{[3]}(x) = 0.5$$
$$f^{[4]}(x) = 1$$
$$f^{[5]}(x) = 0$$

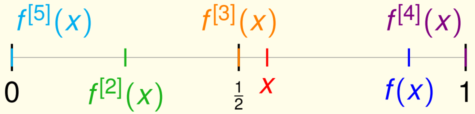
Example

Function

$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Trajectory

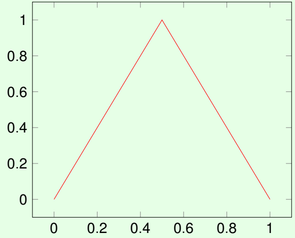


$$x = 0.5625$$
$$f(x) = 0.875$$
$$f^{[2]}(x) = 0.25$$
$$f^{[3]}(x) = 0.5$$
$$f^{[4]}(x) = 1$$
$$f^{[n]}(x) = 0 \qquad n \geqslant 5$$

Example

Function

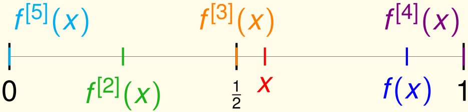
$$f(x) = \begin{cases} 2x & \text{if } x \in [0, \frac{1}{2}] \\ 2 - 2x & \text{if } x \in [\frac{1}{2}, 1] \end{cases}$$



Remark

Trajectory depends on the binary expansion of x

Trajectory



$$x = 0.5625$$
$$f(x) = 0.875$$
$$f^{[2]}(x) = 0.25$$
$$f^{[3]}(x) = 0.5$$
$$f^{[4]}(x) = 1$$
$$f^{[n]}(x) = 0 \qquad n \geqslant 5$$

Existings Results

Problem: REACH-REGION

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine

Existings Results

Problem: `REACH-REGION`

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$

Existings Results

Problem: `REACH-REGION`

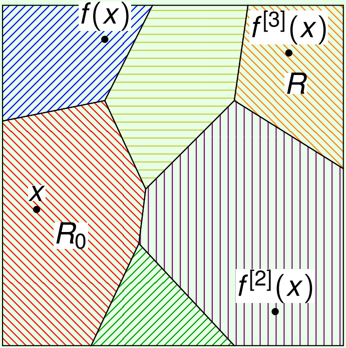
- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$
- **Question:** $\exists x \in R_0, \exists t \in \mathbb{N}, f^{[t]}(x) \in R$?

Existings Results

Problem: REACH-REGION

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$
- **Question:** $\exists x \in R_0, \exists t \in \mathbb{N}, f^{[t]}(x) \in R$?

Example

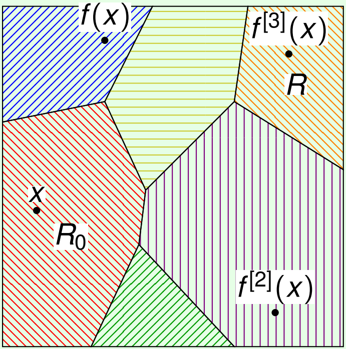


Existings Results

Problem: REACH−REGION

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$
- **Question:** $\exists x \in R_0, \exists t \in \mathbb{N}, f^{[t]}(x) \in R$?

Example



Theorem (Koiran, Cosnard, Garzon)

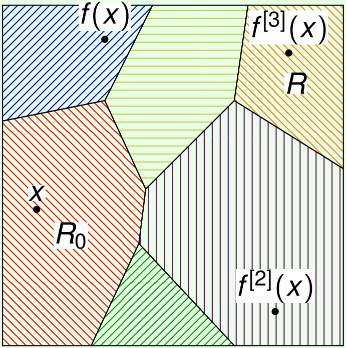
REACH−REGION is undecidable for $d \geqslant 2$

Existings Results

Problem: REACH−REGION

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$
- **Question:** $\exists x \in R_0, \exists t \in \mathbb{N}, f^{[t]}(x) \in R$?

Example



Theorem (Koiran, Cosnard, Garzon)

REACH−REGION is undecidable for $d \geqslant 2$

Proof (Idea)

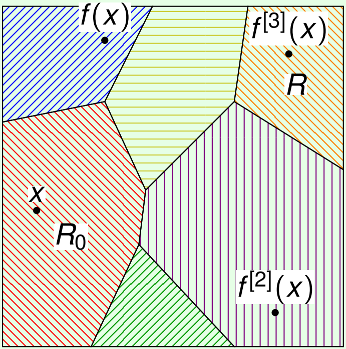
Simulate a Turing Machine and reduce from halting problem.

Existings Results

Problem: `REACH-REGION`

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$
- **Question:** $\exists x \in R_0, \exists t \in \mathbb{N}, f^{[t]}(x) \in R$?

Example



Theorem (Koiran, Cosnard, Garzon)

`REACH-REGION` is undecidable for $d \geqslant 2$

Proof (Idea)

Simulate a Turing Machine and reduce from halting problem.

Open Problem

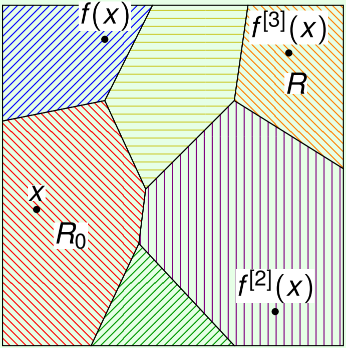
Decidability for $d = 1$.

Existings Results

Problem: CONTROL−REGION

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$
- **Question:** $\forall x \in R_0, \exists t \in \mathbb{N}, f^{[t]}(x) \in R$?

Example



Theorem (Blondel, Bournez, Koiran, Tsitsiklis)

CONTROL−REGION is undecidable for $d \geqslant 2$

Proof (Idea)

Harder simulation of a Turing Machine

Open Problem

Decidability for $d = 1$.

Our Results

Problem: `REACH-REGION-TIME`

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine

Our Results

Problem: `REACH-REGION-TIME`

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$, $T \in \mathbb{N}$ in unary

Our Results

Problem: `REACH-REGION-TIME`

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$, $T \in \mathbb{N}$ in unary
- **Question:** $\exists x \in R_0, \exists t \leqslant T, f^{[t]}(x) \in R$?

Our Results

Problem: REACH-REGION-TIME

- **Input:** $f : [0, 1]^d \to [0, 1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0, 1]^d$, $T \in \mathbb{N}$ in unary
- **Question:** $\exists x \in R_0, \exists t \leqslant T, f^{[t]}(x) \in R$?

Theorem

REACH-REGION-TIME is NP-complete for $d \geqslant 2$

Our Results

Problem: REACH-REGION-TIME

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$, $T \in \mathbb{N}$ in unary
- **Question:** $\exists x \in R_0, \exists t \leqslant T, f^{[t]}(x) \in R$?

Theorem

REACH-REGION-TIME is NP-complete for $d \geqslant 2$

Open Problem

Complexity for $d = 1$.

Our Results

Problem: CONTROL-REGION-TIME

- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$, $T \in \mathbb{N}$ in unary
- **Question:** $\forall x \in R_0, \exists t \leqslant T, f^{[t]}(x) \in R$?

Theorem

CONTROL-REGION-TIME is coNP-complete for $d \geqslant 2$

Open Problem

Complexity for $d = 1$.

Statement

Problem: REACH-REGION-TIME

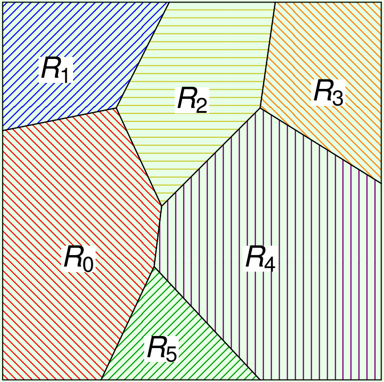
- **Input:** $f : [0,1]^d \to [0,1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0,1]^d$, $T \in \mathbb{N}$ in unary
- **Question:** $\exists x \in R_0, \exists t \leqslant T, f^{[t]}(x) \in R$?

Theorem

REACH-REGION-TIME is in NP.

Signature

Example



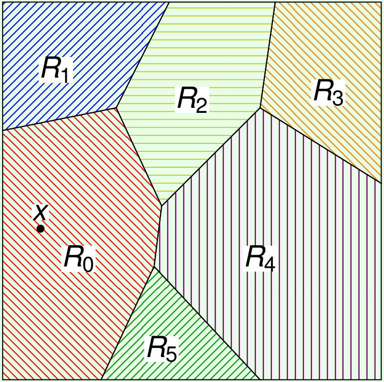
Definition

The *signature* $\sigma(x) \in \{0, \dots, n\}^{\mathbb{N}}$ of x is defined by:

$$\sigma_i(x) = j \quad \Leftrightarrow \quad f^{[i]}(x) \in R_j$$

Signature

Example



$$\sigma(x) = (0, \ldots)$$

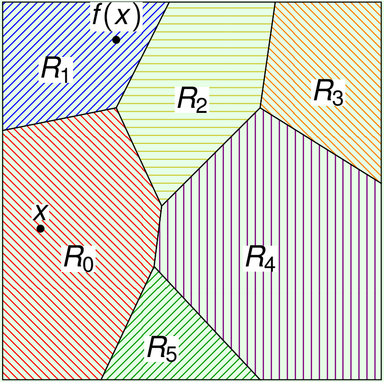
Definition

The *signature* $\sigma(x) \in \{0, \ldots, n\}^{\mathbb{N}}$ of x is defined by:

$$\sigma_i(x) = j \quad \Leftrightarrow \quad f^{[i]}(x) \in R_j$$

Signature

Example



$$\sigma(x) = (0, 1, \ldots)$$

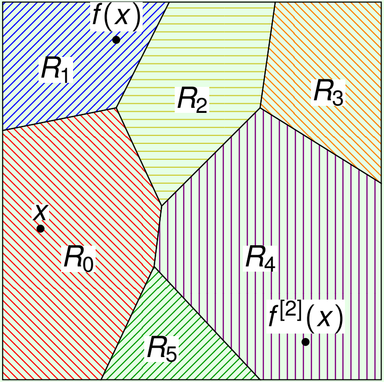
Definition

The *signature* $\sigma(x) \in \{0, \ldots, n\}^{\mathbb{N}}$ of x is defined by:

$$\sigma_i(x) = j \quad \Leftrightarrow \quad f^{[i]}(x) \in R_j$$

Signature

Example



$$\sigma(x) = (0, 1, 4, \ldots)$$

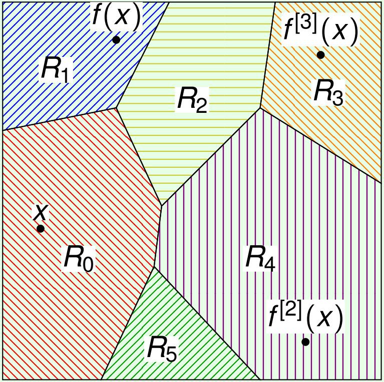
Definition

The *signature* $\sigma(x) \in \{0, \ldots, n\}^{\mathbb{N}}$ of x is defined by:

$$\sigma_i(x) = j \quad \Leftrightarrow \quad f^{[i]}(x) \in R_j$$

Signature

Example



$$\sigma(x) = (0, 1, 4, 3, \ldots)$$

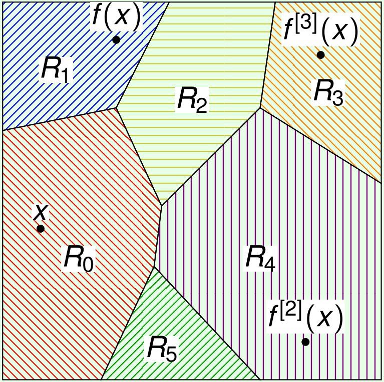
Definition

The *signature* $\sigma(x) \in \{0, \ldots, n\}^{\mathbb{N}}$ of x is defined by:

$$\sigma_i(x) = j \quad \Leftrightarrow \quad f^{[i]}(x) \in R_j$$

Signature

Example



$$\sigma(x) = (0, 1, 4, 3, \ldots)$$

Definition

The *signature* $\sigma(x) \in \{0, \ldots, n\}^{\mathbb{N}}$ of x is defined by:

$$\sigma_i(x) = j \quad \Leftrightarrow \quad f^{[i]}(x) \in R_j$$

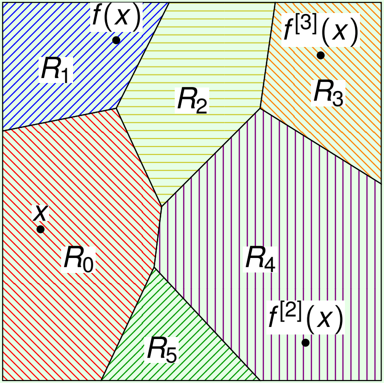
Lemma

If $\sigma(x) = (r_1, r_2, \ldots, r_t, \ldots)$ then

$$f^{[t]}(x) = A_{r_t}(\cdots(A_{r_1}x + b_{r_1})\cdots) + b_{r_t}$$
$$= C_\sigma + d_\sigma$$

Signature

Example



$$\sigma(x) = (0, 1, 4, 3, \ldots)$$

Definition

The *signature* $\sigma(x) \in \{0, \ldots, n\}^{\mathbb{N}}$ of x is defined by:

$$\sigma_i(x) = j \quad \Leftrightarrow \quad f^{[i]}(x) \in R_j$$

Lemma

If $\sigma(x) = (r_1, r_2, \ldots, r_t, \ldots)$ then

$$f^{[t]}(x) = A_{r_t}(\cdots(A_{r_1}x + b_{r_1})\cdots) + b_{r_t}$$
$$= C_\sigma + d_\sigma$$

Furthermore ($s(X)$ =coeff size):

$$s(C_\sigma, d_\sigma) = \text{poly}(s(A), s(b), t)$$

Algorithm

Given f, R_0, $R = R_n$ and T:

Algorithm

Given f, R_0, $R = R_n$ and T:

- Guess $t \leqslant T$ $\leftarrow$ Nondeterministic polynomial

Algorithm

Given f, R_0, $R = R_n$ and T:

- Guess $t \leqslant T$ $\leftarrow$ Nondeterministic polynomial
- Guess signature $r_1, \ldots, r_{t-1}$ $\leftarrow$ Nondeterministic polynomial

Algorithm

Given f, R_0, $R = R_n$ and T:

- Guess $t \leqslant T$ $\leftarrow$ Nondeterministic polynomial
- Guess signature $r_1, \ldots, r_{t-1}$ $\leftarrow$ Nondeterministic polynomial
- Guess $x \in \mathbb{Q}^d$ of polynomial size $\leftarrow$ Nondeterministic polynomial

Algorithm

Given f, R_0, $R = R_n$ and T:

- Guess $t \leqslant T$ $\leftarrow$ Nondeterministic polynomial
- Guess signature $r_1, \ldots, r_{t-1}$ $\leftarrow$ Nondeterministic polynomial
- Guess $x \in \mathbb{Q}^d$ of polynomial size $\leftarrow$ Nondeterministic polynomial
- Check that $f^{[i]}(x) \in R_{r_i}$ for all $i \in \{0, \ldots, t\}$:

Algorithm

Given f, R_0, $R = R_n$ and T:

- Guess $t \leqslant T$ $\leftarrow$ Nondeterministic polynomial
- Guess signature $r_1, \dots, r_{t-1}$ $\leftarrow$ Nondeterministic polynomial
- Guess $x \in \mathbb{Q}^d$ of polynomial size $\leftarrow$ Nondeterministic polynomial
- Check that $f^{[i]}(x) \in R_{r_i}$ for all $i \in \{0, \dots, t\}$:
  $f^{[i]}(x) \in R_{r_i} \Leftrightarrow M_{r_i}(C_i x + d_i) \leqslant v_i$ $\leftarrow$ Polynomial size

Algorithm

Given f, R_0, $R = R_n$ and T:

- Guess $t \leqslant T$ $\leftarrow$ Nondeterministic polynomial
- Guess signature $r_1, \ldots, r_{t-1}$ $\leftarrow$ Nondeterministic polynomial
- Guess $x \in \mathbb{Q}^d$ of polynomial size $\leftarrow$ Nondeterministic polynomial
- Check that $f^{[i]}(x) \in R_{r_i}$ for all $i \in \{0, \ldots, t\}$:
  $f^{[i]}(x) \in R_{r_i} \Leftrightarrow M_{r_i}(C_i x + d_i) \leqslant v_i$ $\leftarrow$ Polynomial size
- Accept if all systems are satisfied

Algorithm

Given f, R_0, $R = R_n$ and T:

- Guess $t \leqslant T$ $\leftarrow$ Nondeterministic polynomial
- Guess signature $r_1, \ldots, r_{t-1}$ $\leftarrow$ Nondeterministic polynomial
- Guess $x \in \mathbb{Q}^d$ of polynomial size $\leftarrow$ Nondeterministic polynomial
- Check that $f^{[i]}(x) \in R_{r_i}$ for all $i \in \{0, \ldots, t\}$:
  $f^{[i]}(x) \in R_{r_i} \Leftrightarrow M_{r_i}(C_i x + d_i) \leqslant v_i$ $\leftarrow$ Polynomial size
- Accept if all systems are satisfied

Theorem (Koiran)

Every satisfiable rational linear system $Ax \leqslant b$ has a rational solution of polynomial size.

Statement

Problem: `REACH-REGION-TIME`

- **Input:** $f : [0, 1]^d \to [0, 1]^d$ continuous, piecewise affine
- **Input:** R_0, R: convex regions of $[0, 1]^d$, $T \in \mathbb{N}$ in unary
- **Question:** $\exists x \in R_0, \exists t \leqslant T, f^{[t]}(x) \in R$?

Theorem

`REACH-REGION-TIME` is NP-hard for $d \geqslant 2$.

General idea

General idea

- Consider $\mathcal{L}$ a NP-hard problem

General idea

- Consider $\mathcal{L}$ a NP-hard problem
- Consider $\mathcal{L}'$ in P such that:

$$\mathcal{L} = \left\{x \mid \exists y, |y| \leqslant \text{poly}(|x|) \text{ and } (x, y) \in \mathcal{L}'\right\}$$

General idea

- Consider $\mathcal{L}$ a NP-hard problem
- Consider $\mathcal{L}'$ in P such that:

$$\mathcal{L} = \left\{ x \mid \exists y, |y| \leqslant \text{poly}(|x|) \text{ and } (x, y) \in \mathcal{L}' \right\}$$

- Define f a piecewise affine function which simulates $\mathcal{L}'$:

General idea

- Consider $\mathcal{L}$ a NP-hard problem

$\psi =$ encoding function

- Consider $\mathcal{L}'$ in P such that:

$$\mathcal{L} = \left\{ x \mid \exists y, |y| \leqslant \text{poly}(|x|) \text{ and } (x, y) \in \mathcal{L}' \right\}$$

- Define f a piecewise affine function which simulates $\mathcal{L}'$:

$$(x, y) \in \mathcal{L}' \Leftrightarrow \exists t \leqslant \text{poly}(|x|, |y|), f^{[t]}(\psi(x, y)) \in R$$

General idea

- Consider $\mathcal{L}$ a NP-hard problem

$\psi =$ encoding function

- Consider $\mathcal{L}'$ in P such that:

$$\mathcal{L} = \left\{x \mid \exists y, |y| \leqslant \text{poly}(|x|) \text{ and } (x, y) \in \mathcal{L}'\right\}$$

- Define f a piecewise affine function which simulates $\mathcal{L}'$:

$$(x, y) \in \mathcal{L}' \Leftrightarrow \exists t \leqslant \text{poly}(|x|, |y|), f^{[t]}(\psi(x, y)) \in R$$

- Define region $R_x = \left\{\psi(x, y) \mid |y| \leqslant \text{poly}(|x|)\right\}$

General idea

- Consider $\mathcal{L}$ a NP-hard problem $\psi =$ encoding function
- Consider $\mathcal{L}'$ in P such that:

$$\mathcal{L} = \left\{x \mid \exists y, |y| \leqslant \text{poly}(|x|) \text{ and } (x,y) \in \mathcal{L}'\right\}$$

- Define f a piecewise affine function which simulates $\mathcal{L}'$:

$$(x,y) \in \mathcal{L}' \Leftrightarrow \exists t \leqslant \text{poly}(|x|,|y|), f^{[t]}(\psi(x,y)) \in R$$

- Define region $R_x = \left\{\psi(x,y) \mid |y| \leqslant \text{poly}(|x|)\right\}$
- Reduce $\mathcal{L}$ to `REACH-REGION-TIME`:

$$x \in \mathcal{L} \Leftrightarrow \exists t \leqslant \text{poly}(|x|), \exists u \in R_x, f^{[t]}(u) \in R$$

General idea

- Consider $\mathcal{L}$ a NP-hard problem ψ = encoding function
- Consider $\mathcal{L}'$ in P such that:

$$\mathcal{L} = \left\{x \mid \exists y, |y| \leqslant \text{poly}(|x|) \text{ and } (x, y) \in \mathcal{L}'\right\}$$

- Define f a piecewise affine function which simulates $\mathcal{L}'$:

$$(x, y) \in \mathcal{L}' \Leftrightarrow \exists t \leqslant \text{poly}(|x|, |y|), f^{[t]}(\psi(x, y)) \in R$$

- Define region $R_x = \left\{\psi(x, y) \mid |y| \leqslant \text{poly}(|x|)\right\}$
- Reduce $\mathcal{L}$ to `REACH-REGION-TIME`:

$$x \in \mathcal{L} \Leftrightarrow \exists t \leqslant \text{poly}(|x|), \exists u \in \tilde{R}_x, f^{[t]}(u) \in R$$

Tricky points

- R_x is not a convex polyhedron: replace it with its convex hull $\tilde{R}_x$

General idea

- Consider $\mathcal{L}$ a NP-hard problem $\psi =$ encoding function
- Consider $\mathcal{L}'$ in P such that:

$$\mathcal{L} = \left\{x \,|\, \exists y, |y| \leqslant \text{poly}(|x|) \text{ and } (x, y) \in \mathcal{L}'\right\}$$

- Define f a piecewise affine function which simulates $\mathcal{L}'$:

$$(x, y) \in \mathcal{L}' \Leftrightarrow \exists t \leqslant \text{poly}(|x|, |y|), f^{[t]}(\psi(x, y)) \in R$$

- Define region $R_x = \left\{\psi(x, y) \,|\, |y| \leqslant \text{poly}(|x|)\right\}$
- Reduce $\mathcal{L}$ to `REACH-REGION-TIME`:

$$x \in \mathcal{L} \Leftrightarrow \exists t \leqslant \text{poly}(|x|), \exists u \in \tilde{R}_x, f^{[t]}(u) \in R$$

Tricky points

- R_x is not a convex polyhedron: replace it with its convex hull $\tilde{R}_x$
- Choice of $\mathcal{L}$?

More on tricky points

$R_x = \{\text{initial configuration}\}$ $\tilde{R}_x = \text{convex hull of } R_x$

More on tricky points

$$R_x = \{\text{initial configuration}\} \quad \tilde{R}_x = \text{convex hull of } R_x$$

Problem

$\tilde{R}_x \setminus R_x$ contains *bizarre* points

More on tricky points

$$R_x = \{\text{initial configuration}\} \quad \tilde{R}_x = \text{convex hull of } R_x$$

Problem

$\tilde{R}_x \setminus R_x$ contains *bizarre* points

Example

- Take $u \in \tilde{R}_x \setminus R_x$, assume $x \notin \mathcal{L}$

More on tricky points

$$R_x = \{\text{initial configuration}\} \quad \tilde{R}_x = \text{convex hull of } R_x$$

Problem

$\tilde{R}_x \setminus R_x$ contains *bizarre* points

Example

- Take $u \in \tilde{R}_x \setminus R_x$, assume $x \notin \mathcal{L}$
- $u \neq \psi(x, y)$ for all $x, y \rightarrow$ point normally inacessible

More on tricky points

$$R_x = \{\text{initial configuration}\} \quad \tilde{R}_x = \text{convex hull of } R_x$$

Problem

$\tilde{R}_x \setminus R_x$ contains *bizarre* points

Example

- Take $u \in \tilde{R}_x \setminus R_x$, assume $x \notin \mathcal{L}$
- $u \neq \psi(x, y)$ for all $x, y \rightarrow$ point normally inacessible
- $f(u)$ may be uncontrolled

More on tricky points

$$R_x = \{\text{initial configuration}\} \quad \tilde{R}_x = \text{convex hull of } R_x$$

Problem

$\tilde{R}_x \setminus R_x$ contains *bizarre* points

Example

- Take $u \in \tilde{R}_x \setminus R_x$, assume $x \notin \mathcal{L}$
- $u \neq \psi(x, y)$ for all $x, y \rightarrow$ point normally inacessible
- $f(u)$ may be uncontrolled
- if $\exists t, f^{[t]}(u) \in R$, system wrongly accepts x

More on tricky points

$$R_x = \{\text{initial configuration}\} \quad \tilde{R}_x = \text{convex hull of } R_x$$

Problem

$\tilde{R}_x \setminus R_x$ contains *bizarre* points

Example

- Take $u \in \tilde{R}_x \setminus R_x$, assume $x \notin \mathcal{L}$
- $u \neq \psi(x, y)$ for all $x, y \to$ point normally inacessible
- $f(u)$ may be uncontrolled
- if $\exists t, f^{[t]}(u) \in R$, system wrongly accepts x

So what ?

- The simulation of $\mathcal{L}'$ has to be studied for *bizarre* points too

More on tricky points

$$R_x = \{\text{initial configuration}\} \quad \tilde{R}_x = \text{convex hull of } R_x$$

Problem

$\tilde{R}_x \setminus R_x$ contains *bizarre* points

Example

- Take $u \in \tilde{R}_x \setminus R_x$, assume $x \notin \mathcal{L}$
- $u \neq \psi(x, y)$ for all $x, y \rightarrow$ point normally inacessible
- $f(u)$ may be uncontrolled
- if $\exists t, f^{[t]}(u) \in R$, system wrongly accepts x

So what ?

- The simulation of $\mathcal{L}'$ has to be studied for *bizarre* points too
- This is difficult for most languages

And the winner is...

And the winner is...

Problem SUBSEM−SUM

- **Input:** a goal $B \in \mathbb{N}$ and integers $A_1, \ldots, A_n \in \mathbb{N}$

And the winner is...

Problem SUBSEM−SUM

- **Input:** a goal $B \in \mathbb{N}$ and integers $A_1, \ldots, A_n \in \mathbb{N}$
- **Question:** $\exists I \subseteq \{1, \ldots, n\}, \sum_{i \in I} A_i = B$?

And the winner is...

Problem SUBSEM−SUM

- **Input:** a goal $B \in \mathbb{N}$ and integers $A_1, \ldots, A_n \in \mathbb{N}$
- **Question:** $\exists I \subseteq \{1, \ldots, n\}, \sum_{i \in I} A_i = B$?

Simulation (1)

- **Configuration:** $(i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$ $\qquad i \in \{1, \ldots, n+1\}, \varepsilon_i \in \{0, 1\}$

And the winner is...

Problem SUBSEM−SUM

- **Input:** a goal $B \in \mathbb{N}$ and integers $A_1, \dots, A_n \in \mathbb{N}$
- **Question:** $\exists I \subseteq \{1, \dots, n\}, \sum_{i \in I} A_i = B$?

Simulation (1)

- **Configuration:** $(i, \sigma, \varepsilon_i, \dots, \varepsilon_n)$ $\quad i \in \{1, \dots, n+1\}, \varepsilon_i \in \{0, 1\}$

$i =$ current number $\quad \sigma =$ current sum $\quad \varepsilon_i =$ pick A_i ?

And the winner is...

Problem SUBSEM−SUM

- **Input:** a goal $B \in \mathbb{N}$ and integers $A_1, \ldots, A_n \in \mathbb{N}$
- **Question:** $\exists I \subseteq \{1, \ldots, n\}, \sum_{i \in I} A_i = B$?

Simulation (1)

- **Configuration:** $(i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$ $\quad i \in \{1, \ldots, n+1\}, \varepsilon_i \in \{0, 1\}$

  $i =$ current number $\quad \sigma =$ current sum $\quad \varepsilon_i =$ pick A_i ?

- **Transition:**

$$(i, \sigma, \varepsilon_1, \ldots, \varepsilon_n) \rightsquigarrow (i+1, \sigma + \varepsilon_i A_i, \varepsilon_{i+1}, \ldots, \varepsilon_n)$$

And the winner is...

Problem SUBSEM−SUM

- **Input:** a goal $B \in \mathbb{N}$ and integers $A_1, \ldots, A_n \in \mathbb{N}$
- **Question:** $\exists I \subseteq \{1, \ldots, n\}, \sum_{i \in I} A_i = B$?

Simulation (1)

- **Configuration:** $(i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$ $\qquad i \in \{1, \ldots, n+1\}, \varepsilon_i \in \{0, 1\}$

  $i =$ current number $\qquad \sigma =$ current sum $\qquad \varepsilon_i =$ pick A_i ?

- **Transition:**

$$(i, \sigma, \varepsilon_1, \ldots, \varepsilon_n) \rightsquigarrow (i+1, \sigma + \varepsilon_i A_i, \varepsilon_{i+1}, \ldots, \varepsilon_n)$$

Simulation lemma (1)

Instance is satisfiable $\Leftrightarrow \exists \varepsilon_1, \ldots \varepsilon_n \in \{0, 1\}$ such that

$$(1, 0, \varepsilon_1, \ldots, \varepsilon_n) \rightsquigarrow^n (n+1, B)$$

Tell me more...

Why SUBSET−SUM ?

Tell me more...

Why SUBSET−SUM ?

- Configuration encoding: $c = (i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$

Tell me more...

Why SUBSET−SUM ?

- Configuration encoding: $c = (i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$

$$\psi(c) = \begin{pmatrix} 0.\; i \;\; \sigma \\ 0.\; 0 \; \cdots \; \varepsilon_i \; \cdots \; \varepsilon_n \end{pmatrix}$$

Tell me more...

Why SUBSET-SUM ?

- Configuration encoding: $c = (i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$

$$\psi(c) = \begin{pmatrix} 0.\ \boxed{\quad i \quad}\boxed{\quad \sigma \quad} \\ 0.\ \boxed{0}\boxed{\cdots}\boxed{\varepsilon_i}\boxed{\cdots}\boxed{\varepsilon_n} \end{pmatrix} = \begin{pmatrix} i2^{-p} + \sigma 2^{-q} \\ \varepsilon_i 2^{-1} + \varepsilon_{i+1} 2^{-2} + \cdots \end{pmatrix}$$

Tell me more...

Why SUBSET-SUM ?

- Configuration encoding: $c = (i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$

$$\psi(c) = \begin{pmatrix} 0.\,[\;i\;]\,[\;\sigma\;] \\ 0.\,[\,0 \mid \cdots \mid \varepsilon_i \mid \cdots \mid \varepsilon_n\,] \end{pmatrix} = \begin{pmatrix} i2^{-p} + \sigma 2^{-q} \\ \varepsilon_i 2^{-1} + \varepsilon_{i+1} 2^{-2} + \cdots \end{pmatrix}$$

- Transitions: $\psi(c) \rightsquigarrow \psi(c')$

Tell me more...

Why SUBSET-SUM ?

- Configuration encoding: $c = (i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$

$$\psi(c) = \begin{pmatrix} 0.\,\boxed{\quad i \quad}\boxed{\quad \sigma \quad} \\ 0.\,\boxed{0}\boxed{\cdots}\boxed{\varepsilon_i}\boxed{\quad\cdots\quad}\boxed{\varepsilon_n} \end{pmatrix} = \begin{pmatrix} i2^{-p} + \sigma 2^{-q} \\ \varepsilon_i 2^{-1} + \varepsilon_{i+1} 2^{-2} + \cdots \end{pmatrix}$$

- Transitions: $\psi(c) \rightsquigarrow \psi(c')$

$$\bullet \varepsilon_i = 0 : \quad \begin{pmatrix} 0.\,\boxed{\quad i \quad}\boxed{\quad \sigma \quad} \\ 0.\,\boxed{0}\boxed{\cdots}\boxed{0}\boxed{\varepsilon_{i+1}}\boxed{\cdots}\boxed{\varepsilon_n} \end{pmatrix} \rightsquigarrow$$

Tell me more...

Why SUBSET−SUM ?

- Configuration encoding: $c = (i, \sigma, \varepsilon_i, \ldots, \varepsilon_n)$

$$\psi(c) = \begin{pmatrix} 0.\,\boxed{i}\,\boxed{\sigma} \\ 0.\,\boxed{0}\,\boxed{\cdots}\,\boxed{\varepsilon_i}\,\boxed{\cdots}\,\boxed{\varepsilon_n} \end{pmatrix} = \begin{pmatrix} i2^{-p} + \sigma 2^{-q} \\ \varepsilon_i 2^{-1} + \varepsilon_{i+1} 2^{-2} + \cdots \end{pmatrix}$$

- Transitions: $\psi(c) \rightsquigarrow \psi(c')$

$$\bullet \varepsilon_i = 0 : \quad \begin{pmatrix} 0.\,\boxed{i}\,\boxed{\sigma} \\ 0.\,\boxed{0}\,\boxed{\cdots}\,\boxed{0}\,\boxed{\varepsilon_{i+1}}\,\boxed{\cdots}\,\boxed{\varepsilon_n} \end{pmatrix} \rightsquigarrow \begin{pmatrix} 0.\,\boxed{i+1}\,\boxed{\sigma} \\ 0.\,\boxed{0}\,\boxed{\cdots}\,\boxed{0}\,\boxed{\varepsilon_{i+1}}\,\boxed{\cdots}\,\boxed{\varepsilon_n} \end{pmatrix}$$

Tell me more...

Why SUBSET−SUM ?

- Configuration encoding: $c = (i, \sigma, \varepsilon_i, \dots, \varepsilon_n)$

$$\psi(c) = \begin{pmatrix} 0.\ [\ i \mid \sigma\ ] \\ 0.\ [\ 0 \mid \cdots \mid \varepsilon_i \mid \cdots \mid \varepsilon_n\ ] \end{pmatrix} = \begin{pmatrix} i2^{-p} + \sigma 2^{-q} \\ \varepsilon_i 2^{-1} + \varepsilon_{i+1} 2^{-2} + \cdots \end{pmatrix}$$

- Transitions: $\psi(c) \rightsquigarrow \psi(c')$

$$\bullet \varepsilon_i = 0 : \quad \begin{pmatrix} 0.\ [\ i \mid \sigma\ ] \\ 0.\ [\ 0 \mid \cdots \mid 0 \mid \varepsilon_{i+1} \mid \cdots \mid \varepsilon_n\ ] \end{pmatrix} \rightsquigarrow \begin{pmatrix} 0.\ [\ i+1 \mid \sigma\ ] \\ 0.\ [\ 0 \mid \cdots \mid 0 \mid \varepsilon_{i+1} \mid \cdots \mid \varepsilon_n\ ] \end{pmatrix}$$

$$\bullet \varepsilon_i = 1 : \quad \begin{pmatrix} 0.\ [\ i \mid \sigma\ ] \\ 0.\ [\ 0 \mid \cdots \mid 1 \mid \varepsilon_{i+1} \mid \cdots \mid \varepsilon_n\ ] \end{pmatrix} \rightsquigarrow$$

Tell me more...

Why SUBSET−SUM ?

- Configuration encoding: $c = (i, \sigma, \varepsilon_i, \dots, \varepsilon_n)$

$$\psi(c) = \begin{pmatrix} 0.\;[\;i\;|\;\sigma\;] \\ 0.\;[\;0\;|\cdots|\;\varepsilon_i\;|\;\cdots\;|\;\varepsilon_n\;] \end{pmatrix} = \begin{pmatrix} i2^{-p} + \sigma 2^{-q} \\ \varepsilon_i 2^{-1} + \varepsilon_{i+1} 2^{-2} + \dots \end{pmatrix}$$

- Transitions: $\psi(c) \rightsquigarrow \psi(c')$

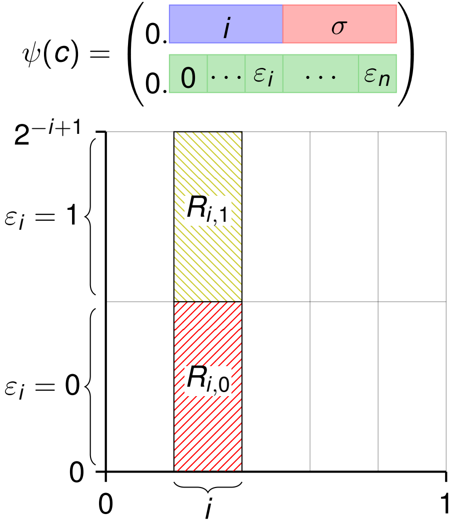
$$\bullet \varepsilon_i = 0 : \quad \begin{pmatrix} 0.\;[\;i\;|\;\sigma\;] \\ 0.\;[\;0\;|\cdots|\;0\;|\;\varepsilon_{i+1}\;|\cdots|\;\varepsilon_n\;] \end{pmatrix} \rightsquigarrow \begin{pmatrix} 0.\;[\;i+1\;|\;\sigma\;] \\ 0.\;[\;0\;|\cdots|\;0\;|\;\varepsilon_{i+1}\;|\cdots|\;\varepsilon_n\;] \end{pmatrix}$$

$$\bullet \varepsilon_i = 1 : \quad \begin{pmatrix} 0.\;[\;i\;|\;\sigma\;] \\ 0.\;[\;0\;|\cdots|\;1\;|\;\varepsilon_{i+1}\;|\cdots|\;\varepsilon_n\;] \end{pmatrix} \rightsquigarrow \begin{pmatrix} 0.\;[\;i+1\;|\;\sigma + A_i\;] \\ 0.\;[\;0\;|\cdots|\;0\;|\;\varepsilon_{i+1}\;|\cdots|\;\varepsilon_n\;] \end{pmatrix}$$

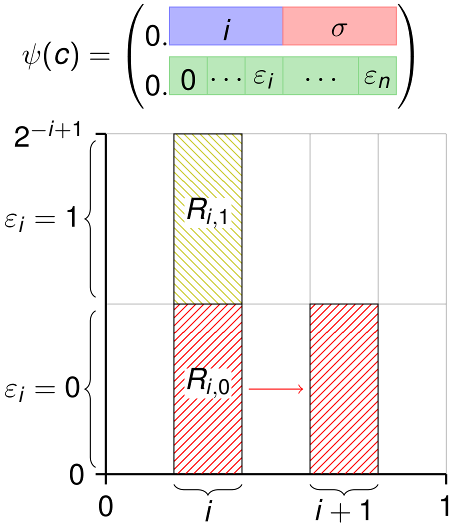
And then were the regions...

$$\psi(c) = \begin{pmatrix} 0.\boxed{\quad i \quad}\boxed{\quad \sigma \quad} \\ 0.\boxed{0}\boxed{\cdots}\boxed{\varepsilon_i}\boxed{\quad \cdots \quad}\boxed{\varepsilon_n} \end{pmatrix}$$

And then were the regions...



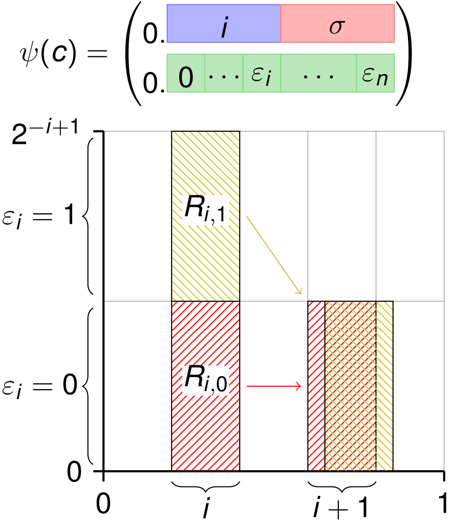
And then were the regions...



Transition on $R_{i,0}$

$$f\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}x+2^{-p}\\y\end{pmatrix}$$

And then were the regions...



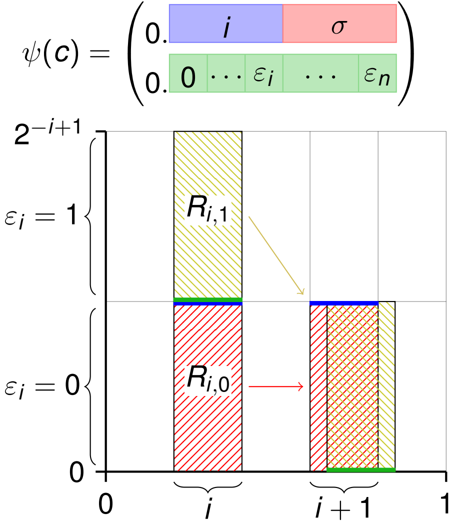
Transition on $R_{i,0}$

$$f\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}x+2^{-p}\\y\end{pmatrix}$$

Transition on $R_{i,1}$

$$f\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}x+2^{-p}+A_i2^{-q}\\y-2^{-i}\end{pmatrix}$$

And then were the regions...



Transition on $R_{i,0}$

$$f\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}x+2^{-p}\\y\end{pmatrix}$$

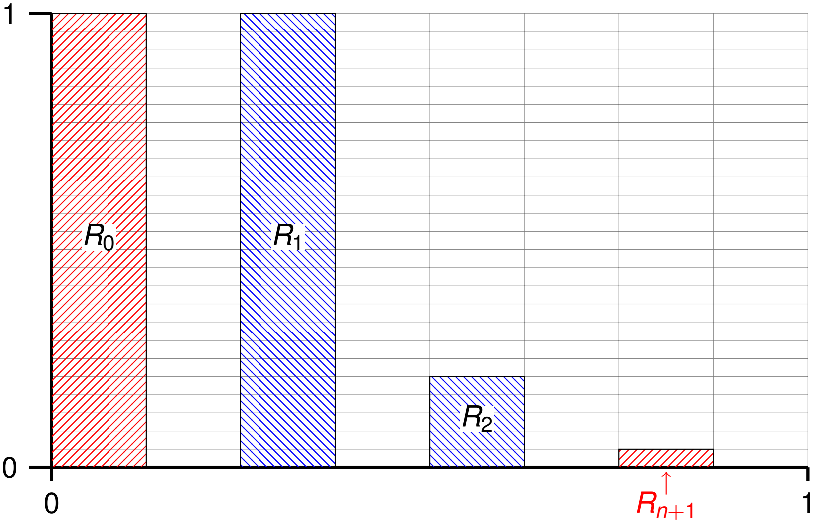
Transition on $R_{i,1}$

$$f\begin{pmatrix}x\\y\end{pmatrix}=\begin{pmatrix}x+2^{-p}+A_i2^{-q}\\y-2^{-i}\end{pmatrix}$$

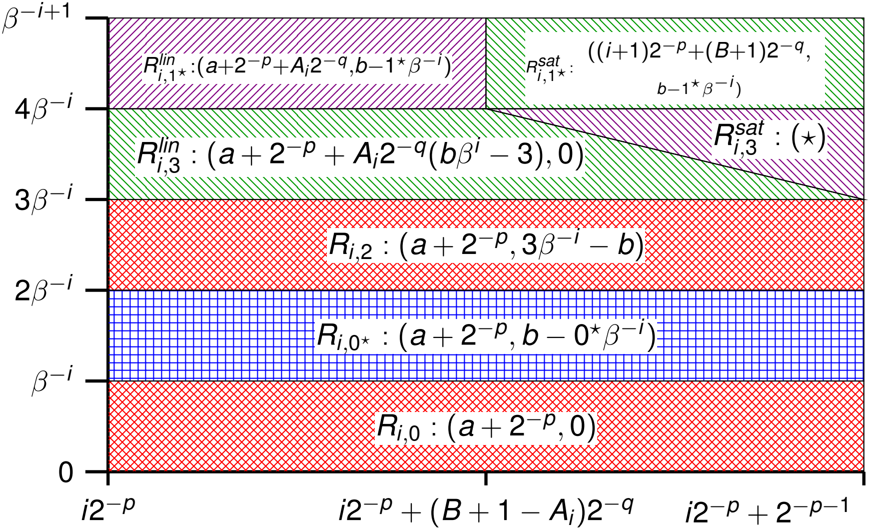
But this doesn't work, right ?

f is not continuous

Ok, the actual proof is slightly more complicated...



...horribly more complicated



$(\star):((i+1)2^{-p}+2^{-p-1}-(b\beta^i-3)(2^{-p-1}-(B+1)2^{-q}),0)$

Conclusion

Reachability in piecewise affine systems:

Conclusion

Reachability in piecewise affine systems:

- undecidable for $d \geqslant 2$

Conclusion

Reachability in piecewise affine systems:

- undecidable for $d \geqslant 2$
- NP-complete for $d \geqslant 2$ (bounded time variant)

Conclusion

Reachability in piecewise affine systems:

- undecidable for $d \geqslant 2$
- NP-complete for $d \geqslant 2$ (bounded time variant)
- open problem for $d = 1$

Questions ?

- Do you have any questions ?