Efficient Solution a Class of Universally Quantified Constraints

Stefan Ratschan[1]

[1]Academy of Sciences of the Czech Republic

Motivation: Termination Proof

Given: program of the form

while $P(x)$ **do**
  $x \leftarrow f(x)$

Motivation: Termination Proof

Given: program of the form

while $P(x)$ **do**
    $x \leftarrow f(x)$

where

- $x \in \mathbb{R}^n$ (cf. $\mathbb{F}^n$, $\mathbb{Z}^n$)

Motivation: Termination Proof

Given: program of the form

while $P(x)$ **do**
$\quad x \leftarrow f(x)$

where

- $x \in \mathbb{R}^n$ (cf. $\mathbb{F}^n$, $\mathbb{Z}^n$)
- f is given by an expression (e.g., $(x, y) \mapsto (x^2 + \sin y, x + 2y)$)

Motivation: Termination Proof

Given: program of the form

while $P(x)$ **do**
    $x \leftarrow f(x)$

where

- $x \in \mathbb{R}^n$ (cf. $\mathbb{F}^n$, $\mathbb{Z}^n$)
- f is given by an expression (e.g., $(x, y) \mapsto (x^2 + \sin y, x + 2y)$)
- $\{x \mid P(x)\}$ is compact (i.e., closed, bounded)

Motivation: Termination Proof

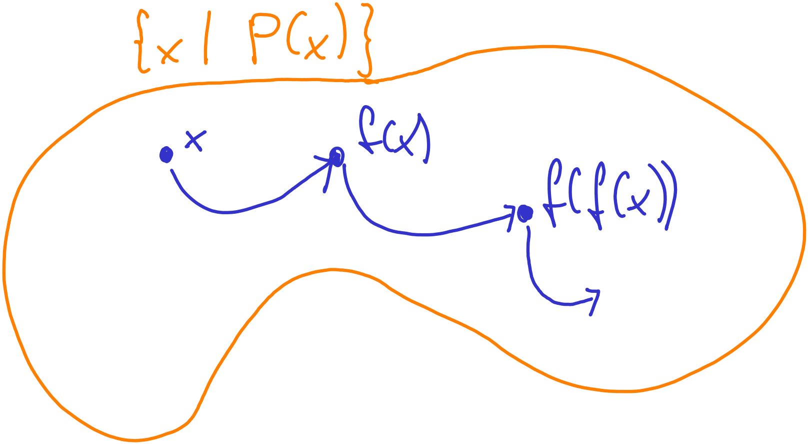
Given: program of the form

while $P(x)$ **do**
  $x \leftarrow f(x)$

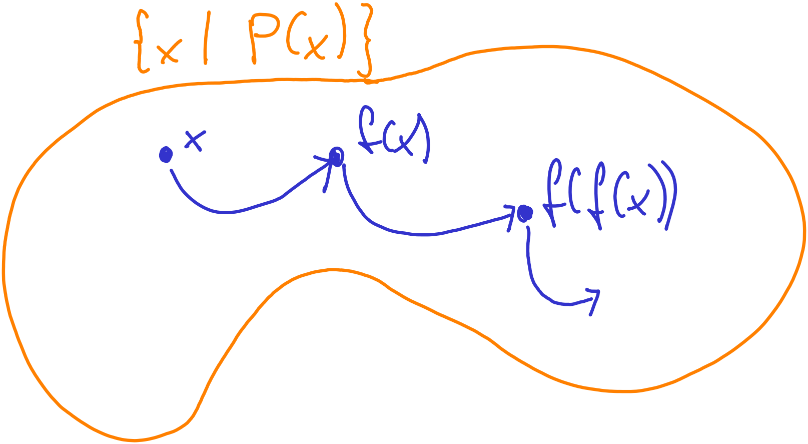
where

- $x \in \mathbb{R}^n$ (cf. $\mathbb{F}^n$, $\mathbb{Z}^n$)
- f is given by an expression (e.g., $(x, y) \mapsto (x^2 + \sin y, x + 2y)$)
- $\{x \mid P(x)\}$ is compact (i.e., closed, bounded)

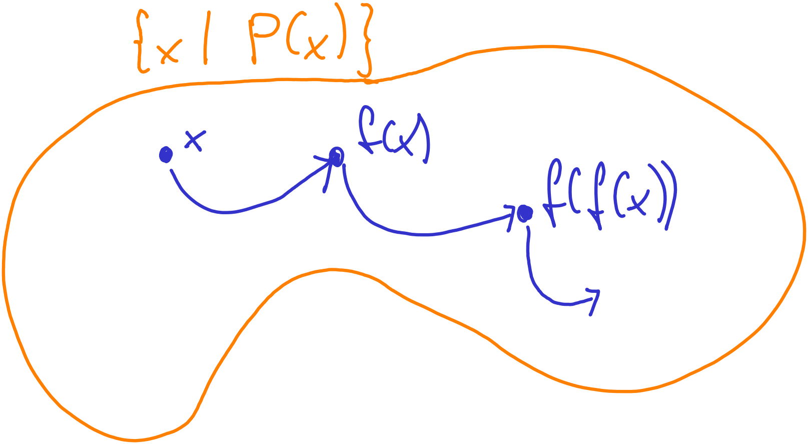
Prove: terminates always.



$\{x \mid P(x)\}$
x
$f(x)$
$f(f(x))$



- There is no infinite sequence $x_1, \ldots,$ s.t.
  for all i, $x_{i+1} = f(x_i)$, $P(x_i)$



- There is no infinite sequence $x_1, \ldots,$ s.t.
  for all i, $x_{i+1} = f(x_i)$, $P(x_i)$ or, equivalently
- for every infinity sequence $x_1, \ldots,$ s.t. for all i, $x_{i+1} = f(x_i)$,
  there is j s.t. $\neg P(x_j)$

Further Motivation: ODE

Instead of program, ODE $\dot{x} = f(x)$

Further Motivation: ODE

Instead of program, ODE $\dot{x} = f(x)$

Prove that it cannot stay forever in set $\{x \mid P(x)\}$

Further Motivation: ODE

Instead of program, ODE $\dot{x} = f(x)$

Prove that it cannot stay forever in set $\{x \mid P(x)\}$

In other words:

Prove that it eventually always reaches set $\{x \mid \neg P(x)\}$.

Further Motivation: ODE

Instead of program, ODE $\dot{x} = f(x)$

Prove that it cannot stay forever in set $\{x \mid P(x)\}$

In other words:

Prove that it eventually always reaches set $\{x \mid \neg P(x)\}$.

see also Luc's talk

Further Motivation: ODE

Instead of program, ODE $\dot{x} = f(x)$

Prove that it cannot stay forever in set $\{x \mid P(x)\}$

In other words:
Prove that it eventually always reaches set $\{x \mid \neg P(x)\}$.

see also Luc's talk

Rest of talk: program termination, for ODE's only slight changes.

Method

Find continuous function $V(x)$ s.t.

$$\forall x \;\; [P(x) \;\Rightarrow\; V(f(x)) \leq V(x) - \varepsilon],$$

for some $\varepsilon > 0$

Method

Find continuous function $V(x)$ s.t.

$$\forall x \ \ [P(x) \ \Rightarrow \ V(f(x)) \leq V(x) - \varepsilon],$$

for some $\varepsilon > 0$

Then:

- If loop would not terminate,
- then $V(x)$ would go to $-\infty$,
- which cannot happen since due to compactness of $\{x \mid P(x)\}$, $\{V(x) \mid P(x)\}$ is bounded from below

Method

Find continuous function $V(x)$ s.t.

$$\forall x \ [P(x) \ \Rightarrow \ V(f(x)) \leq V(x) - \varepsilon],$$

for some $\varepsilon > 0$

Then:

- If loop would not terminate,
- then $V(x)$ would go to $-\infty$,
- which cannot happen since due to compactness of $\{x \mid P(x)\}$, $\{V(x) \mid P(x)\}$ is bounded from below

How to find such a $V(x)$?

Method

Find continuous function $V(x)$ s.t.

$$\forall x \ [P(x) \ \Rightarrow \ V(f(x)) \leq V(x) - \varepsilon],$$

for some $\varepsilon > 0$

Then:

- If loop would not terminate,
- then $V(x)$ would go to $-\infty$,
- which cannot happen since due to compactness of $\{x \mid P(x)\}$, $\{V(x) \mid P(x)\}$ is bounded from below

How to find such a $V(x)$?

Pattern polynomial, for example:

$$V(a_1, a_2, a_3, x_1, x_2) = a_1 x_1^3 x_2 + a_2 x_1^2 + a_3 x_2^2$$

Method

Find continuous function $V(x)$ s.t.

$$\forall x \ \ [P(x) \ \Rightarrow \ V(f(x)) \leq V(x) - \varepsilon],$$

for some $\varepsilon > 0$

Then:

- If loop would not terminate,
- then $V(x)$ would go to $-\infty$,
- which cannot happen since due to compactness of $\{x \mid P(x)\}$, $\{V(x) \mid P(x)\}$ is bounded from below

How to find such a $V(x)$?

Pattern polynomial, for example:

$$V(a_1, a_2, a_3, x_1, x_2) = a_1 x_1^3 x_2 + a_2 x_1^2 + a_3 x_2^2$$

Find a (i.e., for example, a_1, a_2, a_3) s.t.

$$\forall x \ [P(x) \ \Rightarrow \ V(a, f(x)) \leq V(a, x) - \varepsilon]$$

How to Solve Quantified Problem

$$\forall x \;\; [P(x) \;\Rightarrow\; V(a, f(x)) \leq V(a, x) - \varepsilon]$$

How to Solve Quantified Problem

$$\forall x \;\; [P(x) \;\Rightarrow\; V(a, f(x)) \leq V(a, x) - \varepsilon]$$

A. Tarski (30ies): always possible in polynomial case

How to Solve Quantified Problem

$$\forall x \ \ [P(x) \ \Rightarrow \ V(a, f(x)) \leq V(a, x) - \varepsilon]$$

A. Tarski (30ies): always possible in polynomial case (in theory)

How to Solve Quantified Problem

$$\forall x \ \ [P(x) \ \Rightarrow \ V(a, f(x)) \leq V(a, x) - \varepsilon]$$

A. Tarski (30ies): always possible in polynomial case (in theory)

But: related algorithms (e.g., quantifier elimination by cylindrical algebraic computation) not efficient enough

How to Solve Quantified Problem

$$\forall x \;\; [P(x) \;\Rightarrow\; V(a, f(x)) \leq V(a, x) - \varepsilon]$$

A. Tarski (30ies): always possible in polynomial case (in theory)

But: related algorithms (e.g., quantifier elimination by cylindrical algebraic computation) not efficient enough

Interval branch-and-bound techniques
(`http://rsolver.sourceforge.net`)

How to Solve Quantified Problem

$$\forall x \;\; [P(x) \;\Rightarrow\; V(a, f(x)) \leq V(a, x) - \varepsilon]$$

A. Tarski (30ies): always possible in polynomial case (in theory)

But: related algorithms (e.g., quantifier elimination by cylindrical algebraic computation) not efficient enough

Interval branch-and-bound techniques
(`http://rsolver.sourceforge.net`)

Goal: special method that exploits problem structure?

How to Solve Quantified Problem

$$\forall x \;\; [P(x) \;\Rightarrow\; V(a, f(x)) \leq V(a, x) - \varepsilon]$$

A. Tarski (30ies): always possible in polynomial case (in theory)

But: related algorithms (e.g., quantifier elimination by cylindrical algebraic computation) not efficient enough

Interval branch-and-bound techniques
(`http://rsolver.sourceforge.net`)

Goal: special method that exploits problem structure?

Which one?

How to Solve Quantified Problem

$$\forall x \;\; [P(x) \;\Rightarrow\; V(a, f(x)) \le V(a, x) - \varepsilon]$$

A. Tarski (30ies): always possible in polynomial case (in theory)

But: related algorithms (e.g., quantifier elimination by cylindrical algebraic computation) not efficient enough

Interval branch-and-bound techniques
(`http://rsolver.sourceforge.net`)

Goal: special method that exploits problem structure?

Which one?

$$V(a_1, a_2, a_3, x_1, x_2) = a_1 x_1^3 x_2 + a_2 x_1^2 + a_3 x_2^2$$

How to Solve Quantified Problem

$$\forall x \ \ [P(x) \ \Rightarrow \ V(a, f(x)) \leq V(a, x) - \varepsilon]$$

A. Tarski (30ies): always possible in polynomial case (in theory)

But: related algorithms (e.g., quantifier elimination by cylindrical algebraic computation) not efficient enough

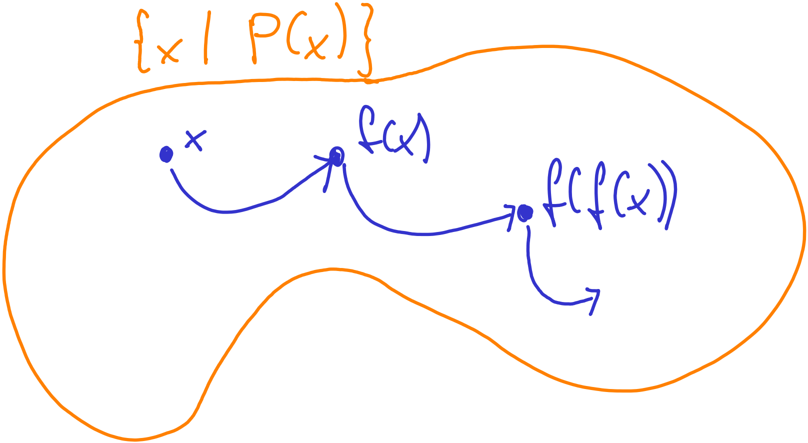
Interval branch-and-bound techniques (`http://rsolver.sourceforge.net`)

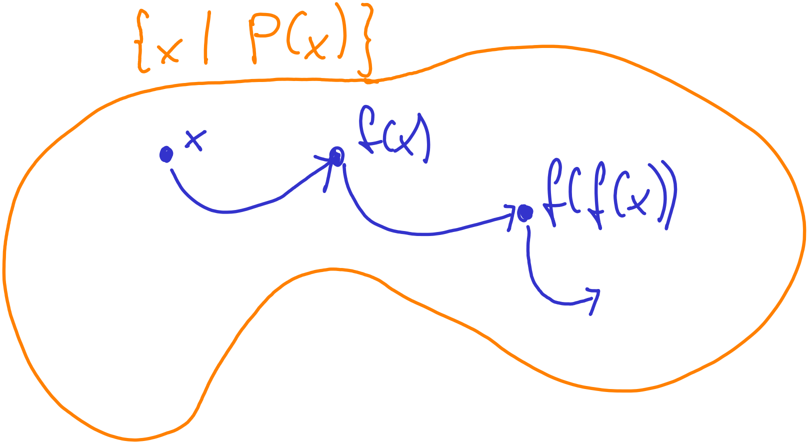
Goal: special method that exploits problem structure?

Which one?

$$V(a_1, a_2, a_3, x_1, x_2) = a_1 x_1^3 x_2 + a_2 x_1^2 + a_3 x_2^2$$

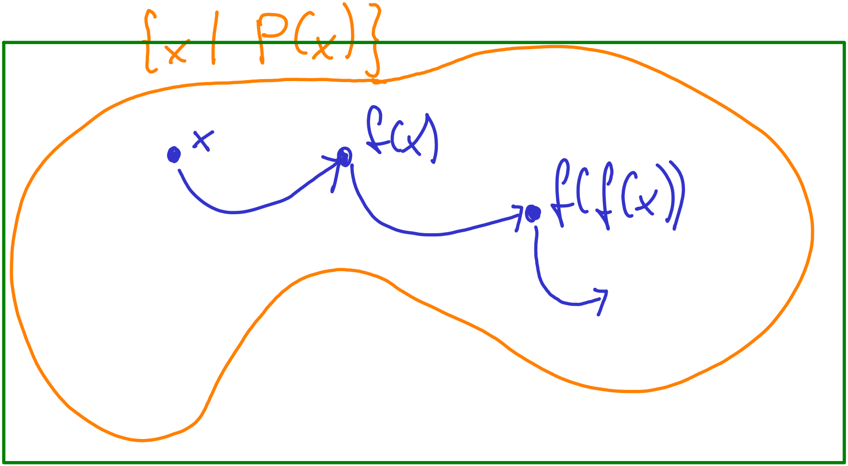
linear in parameters a_1, a_2, a_3


$\{x \mid P(x)\}$
x
$f(x)$
$f(f(x))$



instead of

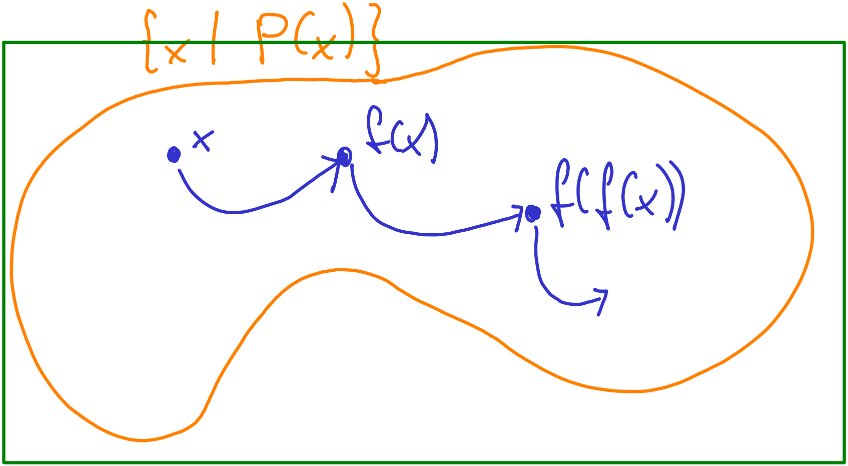
$$\forall x\, [P(a,x) \;\Rightarrow\; V(a,f(x)) \le V(a,x) - \varepsilon]$$



instead of

$$\forall x \, [P(a,x) \;\Rightarrow\; V(a,f(x)) \leq V(a,x) - \varepsilon]$$

$$\forall x \, [x \in B \;\Rightarrow\; V(a,f(x)) \leq V(a,x) - \varepsilon]$$



instead of

$$\forall x \, [P(a,x) \;\Rightarrow\; V(a,f(x)) \leq V(a,x) - \varepsilon]$$

$$\forall x \in B \;\; [V(a,f(x)) \leq V(a,x) - \varepsilon]$$

Reduction to Linear Programming

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Reduction to Linear Programming

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Then: substitute intervals given by B for x

So from

$$a_1 x_1^{\alpha_1} + \cdots + a_n x_n^{\alpha_n} \leq \varepsilon$$

Reduction to Linear Programming

$$\forall x \in B \ \ [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Then: substitute intervals given by B for x

So from

$$a_1 x_1^{\alpha_1} + \cdots + a_n x_n^{\alpha_n} \leq \varepsilon$$

to

$$a_1 I_1 + \cdots + a_n I_n \leq \varepsilon$$

Reduction to Linear Programming

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Then: substitute intervals given by B for x

So from

$$a_1 x_1^{\alpha_1} + \cdots + a_n x_n^{\alpha_n} \leq \varepsilon$$

to

$$I_1 a_1 + \cdots + I_n a_n \leq \varepsilon$$

Reduction to Linear Programming

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Then: substitute intervals given by B for x

So from

$$a_1 x_1^{\alpha_1} + \cdots + a_n x_n^{\alpha_n} \leq \varepsilon$$

to

$$I_1 a_1 + \cdots + I_n a_n \leq \varepsilon$$

(system of) linear inequalities with interval coefficients

Reduction to Linear Programming

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Then: substitute intervals given by B for x

So from

$$a_1 x_1^{\alpha_1} + \cdots + a_n x_n^{\alpha_n} \leq \varepsilon$$

to

$$I_1 a_1 + \cdots + I_n a_n \leq \varepsilon$$

(system of) linear inequalities with interval coefficients

Task: find $a_1, \ldots, a_n$ s.t.
for all elements of intervals, inequality holds.

Reduction to Linear Programming

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Then: substitute intervals given by B for x

So from

$$a_1 x_1^{\alpha_1} + \cdots + a_n x_n^{\alpha_n} \leq \varepsilon$$

to

$$I_1 a_1 + \cdots + I_n a_n \leq \varepsilon$$

(system of) linear inequalities with interval coefficients

Task: find $a_1, \ldots, a_n$ s.t.
for all elements of intervals, inequality holds.

Lossless reduction to linear progr. [Rohn and Kreslová, 1994].

Reduction to Linear Programming

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Then: substitute intervals given by B for x

So from

$$a_1 x_1^{\alpha_1} + \cdots + a_n x_n^{\alpha_n} \leq \varepsilon$$

to

$$I_1 a_1 + \cdots + I_n a_n \leq \varepsilon$$

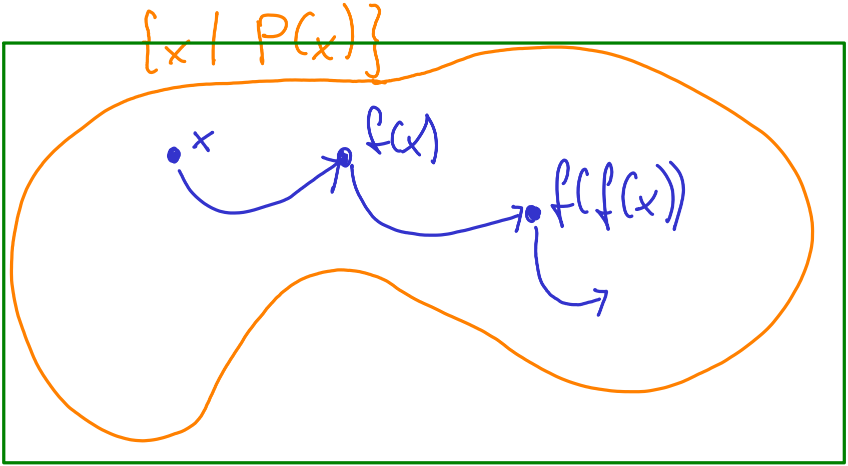
(system of) linear inequalities with interval coefficients

Task: find $a_1, \ldots, a_n$ s.t.
for all elements of intervals, inequality holds.

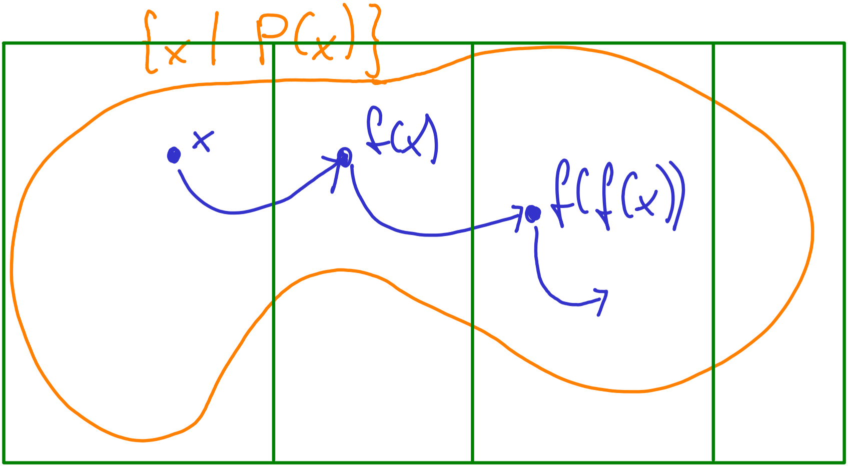
Lossless reduction to linear progr. [Rohn and Kreslová, 1994].

May lose solvability (over-approximation in interval substitution)

Regaining Solvability/Dependence



Regaining Solvability/Dependence



Regaining Solvability/Dependence

Split B into B_1, B_2 and

Regaining Solvability/Dependence

Split B into B_1, B_2 and rewrite

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

to

$$\forall x \in B_1 \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon] \wedge$$
$$\forall x \in B_2 \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

Regaining Solvability/Dependence

Split B into B_1, B_2 and rewrite

$$\forall x \in B \; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

to

$$\begin{aligned} &\forall x \in B_1 \; [V(a, f(x)) \leq V(a, x) - \varepsilon] \wedge \\ &\forall x \in B_2 \; [V(a, f(x)) \leq V(a, x) - \varepsilon] \end{aligned}$$

Each box: interval linear inequality

So: system of interval linear inequalities

Regaining Solvability/Dependence

Split B into B_1, B_2 and rewrite

$$\forall x \in B \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon]$$

to

$$\begin{aligned} &\forall x \in B_1 \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon] \wedge \\ &\forall x \in B_2 \;\; [V(a, f(x)) \leq V(a, x) - \varepsilon] \end{aligned}$$

Each box: interval linear inequality

So: system of interval linear inequalities

Iterate splitting until solved

Example

$$\begin{cases} \dot{x}_1 = -x_2 \\ \dot{x}_2 = -x_3 \\ \dot{x}_3 = -x_1 - 2x_2 - x_3 + x_1^3 \end{cases}$$

$$V(x_1, x_2, x_3) = ax_1^2 + bx_2^2 + cx_3^2 + dx_1x_2 + ex_1x_3 + fx_2x_3,$$

$$B = [-0.2, 0.2] \times [-0.2, 0.2] \times [-0.2, 0.2] \backslash \\ (-0.1, 0.1) \times (-0.1, 0.1) \times (-0.1, 0.1)$$

$$V(x_1, x_2, x_3) = x_1^2 + 0.494353826851x_2^2 + 0.505646173149x_3^2 + \\ -1.0112923463x_1x_3 + 0.0225846925972x_2x_3.$$

Intermediate Summary

Intermediate Summary

- loop termination/ODE leaves region

Intermediate Summary

- loop termination/ODE leaves region
- find function $V(x)$

Intermediate Summary

- loop termination/ODE leaves region
- find function $V(x)$
- choose pattern polynomial $V(a, x)$, find a

Intermediate Summary

- loop termination/ODE leaves region
- find function $V(x)$
- choose pattern polynomial $V(a, x)$, find a
- system of interval linear inequalities

Intermediate Summary

- loop termination/ODE leaves region
- find function $V(x)$
- choose pattern polynomial $V(a, x)$, find a
- system of interval linear inequalities
- split/iterate

Intermediate Summary

- loop termination/ODE leaves region
- find function $V(x)$
- choose pattern polynomial $V(a, x)$, find a
- system of interval linear inequalities
- split/iterate

We could also iteration on pattern polynomial (increase degree)

Splitting Heuristics (joint work with Milan Hladík)

Problem: blind equi-distant splitting

Splitting Heuristics (joint work with Milan Hladík)

Problem: blind equi-distant splitting

Goal: make system solvable with only a few splits.

Splitting Heuristics (joint work with Milan Hladík)

Problem: blind equi-distant splitting

Goal: make system solvable with only a few splits.

That is: system of linear interval inequalities $M^I a \leq q$

Splitting Heuristics (joint work with Milan Hladík)

Problem: blind equi-distant splitting

Goal: make system solvable with only a few splits.

That is: system of linear interval inequalities $M^I a \leq q$

Not solvable, i.e., no a such that for all $M \in M^I$, $Ma \leq q$

Splitting Heuristics (joint work with Milan Hladík)

Problem: blind equi-distant splitting

Goal: make system solvable with only a few splits.

That is: system of linear interval inequalities $M^I a \leq q$

Not solvable, i.e., no a such that for all $M \in M^I$, $Ma \leq q$

Splits shrink intervals in M^I, which one to shrink?

Which Interval in M^I to Shrink?

Which Interval in M^I to Shrink?

Rewrite (well known):

$$\forall M \in M^I \quad Ma \leq q$$

Which Interval in M^I to Shrink?

Rewrite (well known):

$$\forall M \in M^I \;\; Ma \leq q$$

$$\forall M \in [M^c - M^{\triangle}, M^c + M^{\triangle}] \;\; Ma \leq q$$

Which Interval in M^I to Shrink?

Rewrite (well known):

$$\forall M \in M^I \;\; Ma \leq q$$

$$\forall M \in [M^c - M^{\triangle}, M^c + M^{\triangle}] \;\; Ma \leq q$$

$$\forall M \in [-M^{\triangle}, M^{\triangle}] \;\; Ma \leq q - M^c a$$

Which Interval in M^I to Shrink?

Rewrite (well known):

$$\forall M \in M^I \;\; Ma \leq q$$

$$\forall M \in [M^c - M^\triangle, M^c + M^\triangle] \;\; Ma \leq q$$

$$\forall M \in [-M^\triangle, M^\triangle] \;\; Ma \leq q - M^c a$$

$$M^\triangle |a| \leq q - M^c a$$

Which Interval in M^I to Shrink?

Rewrite (well known):

$$\forall M \in M^I \;\; Ma \leq q$$

$$\forall M \in [M^c - M^\triangle, M^c + M^\triangle] \;\; Ma \leq q$$

$$\forall M \in [-M^\triangle, M^\triangle] \;\; Ma \leq q - M^c a$$

$$M^\triangle |a| \leq q - M^c a$$

Choose an a close to expected solution

Which Interval in M^I to Shrink?

Rewrite (well known):

$$\forall M \in M^I \;\; Ma \leq q$$

$$\forall M \in [M^c - M^\triangle, M^c + M^\triangle] \;\; Ma \leq q$$

$$\forall M \in [-M^\triangle, M^\triangle] \;\; Ma \leq q - M^c a$$

$$M^\triangle |a| \leq q - M^c a$$

Choose an a close to expected solution

Evaluate both sides

Which Interval in M^I to Shrink?

Rewrite (well known):

$$\forall M \in M^I \;\; Ma \le q$$

$$\forall M \in [M^c - M^\triangle, M^c + M^\triangle] \;\; Ma \le q$$

$$\forall M \in [-M^\triangle, M^\triangle] \;\; Ma \le q - M^c a$$

$$M^\triangle |a| \le q - M^c a$$

Choose an a close to expected solution

Evaluate both sides

Choose split that improves worst violation the most

Implementation

Implementation

- Exists

Implementation

- Exists
- No systematic computational experiments of splitting heuristics, yet

Implementation

- Exists
- No systematic computational experiments of splitting heuristics, yet
- Heuristics improve run-time several times

Implementation

- Exists
- No systematic computational experiments of splitting heuristics, yet
- Heuristics improve run-time several times
- Largest examples: dimension 6

General Algorithm

Find $a_1, \ldots, a_r$ s.t.

$$\bigwedge_{i=1}^{n} \forall x_1, \ldots, x_s \in B_i \,.\, \phi_i(a_1, \ldots, a_r, x_1, \ldots x_s)$$

where

- each B_i is a box in $\mathbb{R}^s$
- each of the $\phi_1, \ldots, \phi_m$ is a Boolean combination of inequalities where
  - only one of those inequalities contains a variable $x_1, \ldots, x_s$ and
  - this one inequality contains those variables only linearly.

Conclusion: Vision

Conclusion: Vision

Completely automatize Luc :-)

Conclusion: Vision

Completely automatize Luc :-)

Automatic, verified, global analysis of dynamical system

Conclusion: Vision

Completely automatize Luc :-)

Automatic, verified, global analysis of dynamical system

computer programs	ODEs

Conclusion: Vision

Completely automatize Luc :-)

Automatic, verified, global analysis of dynamical system

Sub-problems:

computer programs	ODEs
termination	leaves region

Conclusion: Vision

Completely automatize Luc :-)

Automatic, verified, global analysis of dynamical system

Sub-problems:

computer programs	ODEs
termination	leaves region
invariant sets	invariant sets

Conclusion: Vision

Completely automatize Luc :-)

Automatic, verified, global analysis of dynamical system

Sub-problems:

computer programs	ODEs	objects
termination	leaves region	V(x)
invariant sets	invariant sets	barrier

Conclusion: Vision

Completely automatize Luc :-)

Automatic, verified, global analysis of dynamical system

Sub-problems:

computer programs	ODEs	objects
termination	leaves region	V(x)
invariant sets	invariant sets	barrier

Infrastructure: solver for quantified constraints

Literature I

Tomáš Dzetkulič and Stefan Ratschan. Incremental computation of succinct abstractions for hybrid systems. In *FORMATS 2011*, volume 6919 of *LNCS*, pages 271–285. Springer, Heidelberg (2011), 2011.

M. Fränzle, C. Herde, S. Ratschan, T. Schubert, and T. Teige. Efficient solving of large non-linear arithmetic constraint systems with complex boolean structure. *JSAT—Journal on Satisfiability, Boolean Modeling and Computation, Special Issue on SAT/CP Integration*, 1:209–236, 2007.

Stefan Ratschan and Zhikun She. Safety verification of hybrid systems by constraint propagation based abstraction refinement. *ACM Transactions in Embedded Computing Systems*, 6(1):1–23, 2007. article no. 8.

Literature II

Stefan Ratschan and Zhikun She. Providing a basin of attraction to a target region of polynomial systems by computation of Lyapunov-like functions. *SIAM Journal on Control and Optimization*, 48(7):4377–4394, 2010. doi: 10.1137/090749955. URL `http://link.aip.org/link/?SJC/48/4377/1`.

Jiří Rohn and Jana Kreslová. Linear interval inequalities. *Linear and Multilinear Algebra*, 38:79–82, 1994.