On the Unsolvability of Loop Analysis

Anton Varonka

Joint work with Laura Kovács

TU Wien Informatics*

Reachability Problems 2022

*Funded by the ERC CoG ARTIST 101002685 and the Marie Skłodowska-Curie Network LogiCS@TU Wien

Positivity: a hard problem

A **linear recurrence sequence** (LRS) over $\mathbb{Z}$ is an infinite sequence $\boldsymbol{u} = \langle u_0, u_1, u_2, \ldots \rangle$ of integers defined by

$$u_{n+d} = c_1 u_{n+d-1} + \cdots + c_d u_n$$

for all $n \geq 0$ and the initial values $u_0, \ldots, u_{d-1} \in \mathbb{Z}$.

Positivity Problem

Are all terms of given $\boldsymbol{u}$ non-negative?

This talk is **not** about LRS.

Positivity as Halting

$\boldsymbol{x} := \boldsymbol{c}$
while $x_1 \geq 0$ **do**
  $\boldsymbol{x} := A \cdot \boldsymbol{x}$

Does this **loop** run forever?

Positivity as Halting

$\boldsymbol{x} := \boldsymbol{c}$
while $x_1 \geq 0$ **do**
  $\boldsymbol{x} := A \cdot \boldsymbol{x}$

Does this **loop** run forever?
In general, we don't know.

Positivity is hard

decidability with 6 and more variables $\Rightarrow$ (hard) open problems in Diophantine approximation solved...

Positivity as Halting

$\boldsymbol{x} := \boldsymbol{c}$
while $x_1 \geq 0$ **do**
  $\boldsymbol{x} := A \cdot \boldsymbol{x}$

Does this **loop** run forever?
In general, we don't know.

Positivity is hard

decidability with 6 and more variables $\Rightarrow$ (hard) open problems in Diophantine approximation solved...

What about **universal termination**?

Termination Problem

$$\begin{array}{l} \boldsymbol{x} := \boldsymbol{c} \\ \textbf{while } x_1 \geq 0 \textbf{ do} \\ \quad \boldsymbol{x} := A \cdot \boldsymbol{x} \end{array}$$

Termination of Affine Loops over the Integers

Mehran Hosseini[1], Joël Ouaknine[1,2], James Worrell[1]

RP
12 September, 2019

Over all inputs:

Termination of single-path linear loops over $\mathbb{Z}$ is decidable.

- universal termination
- updates: linear (affine)
- guards: linear inequalities

- universal termination
- updates: linear (affine)
- guards: linear inequalities
- single-path?

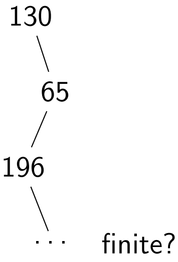
Mark Braverman
(2022 IMU Abacus Medal)

Q: “How much non-determinism can be introduced in a linear loop [. . .] before termination becomes undecidable?”

Not what Braverman wanted

Problems shown undecidable:

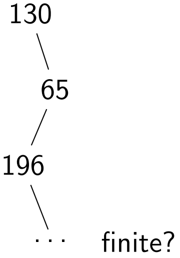
1. Termination of piecewise affine loops
2. Generalized Collatz Conjecture



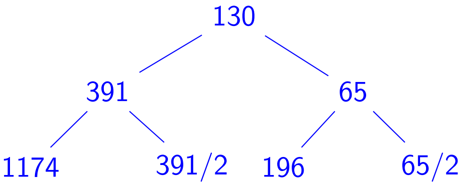
Not what Braverman wanted

Problems shown undecidable:

1. Termination of piecewise affine loops
2. Generalized Collatz Conjecture



Non-determistic termination tree:



Programs we consider

Non-deterministic.

Sample loop

while **do**

σ_1 :

or

σ_2 :

or

σ_3 :

Programs we consider

Non-deterministic. Arbitrary dimension.

Sample loop (in 3D)

$(x, y, z) := (-1, -1, 2)$
while **do**
$\boldsymbol{\sigma_1}$: $(x, y, z) := (x - y + 1, y - 2z, 2z - x - 1)$
or
$\boldsymbol{\sigma_2}$: $(x, y, z) := (-\frac{3}{2}, x + y + \frac{1}{2}, -x - y + 1)$
or
$\boldsymbol{\sigma_3}$: $(x, y, z) := (2x, y + z, -x)$

Programs we consider

Non-deterministic. Arbitrary dimension. Linear inequality conditions.

Sample loop (in 3D)

$(x, y, z) := (-1, -1, 2)$

while $x + 2y + 3z > 0 \land x \leq 10$ **do**

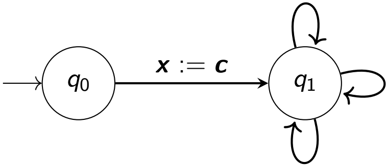
$\boldsymbol{\sigma_1}$: $(x, y, z) := (x - y + 1, y - 2z, 2z - x - 1)$

or

$\boldsymbol{\sigma_2}$: $(x, y, z) := (-\frac{3}{2}, x + y + \frac{1}{2}, -x - y + 1)$

or

$\boldsymbol{\sigma_3}$: $(x, y, z) := (2x, y + z, -x)$



Program correctness:

- Termination on all branches
- Finding good invariants

Termination Result

Theorem

Termination of multi-path affine loops with linear inequality conditions **is undecidable**.

Proof by reduction from the Post's Correspondence Problem (its complement). A loop terminates iff an instance of PCP has no solution.

Remains undecidable with:

- just 4 variables, or
- just 2 linear updates.

Termination undecidable

PCP input: $\{\frac{011}{0}\}, \{\frac{1}{11}\}$.

$$\frac{1}{1} \to \frac{1011}{10} \to \frac{10111}{1011} \to \frac{101111}{101111} \qquad \mapsto \qquad \frac{1}{1} \xrightarrow{\sigma_1} \frac{11}{2} \xrightarrow{\sigma_2} \frac{23}{11} \xrightarrow{\sigma_2} \frac{47}{47}$$

$(x, y, z) := (1, 1, 1)$
while $c \geq 0 \wedge z \geq 0 \wedge z \leq 1$ **do**
  $\boldsymbol{\sigma_1}$ or $\boldsymbol{\sigma_2}$ or $\boldsymbol{\sigma_3}$

Updates σ_1 and σ_2 guarantee: a fixpoint $\begin{pmatrix}47 & 47 & 0 & 0\end{pmatrix}$ of σ_3 is reached
$\sigma_1\sigma_2\sigma_2(\sigma_3)^\omega$ is a non-terminating execution
other executions: forced to apply σ_1 or σ_2 until $x = y$, otherwise termination in at most 2 steps

Invariants

$(x, y, z) := (-1, -1, 2)$
while true **do**
  $(x, y, z) := (x - y + 1, y - z, 2z + y - 1)$
  or
  $(x, y, z) := (-\frac{3}{2}, x + y + \frac{1}{2}, z + 1)$
  or
  $(x, y, z) := (2x, y + z, -x)$

Inductive invariant is a relation between variables of a loop $\mathcal{L}$ which is *preserved under any update of* $\mathcal{L}$.

$$f(x, y, z) = 0$$

Invariants

$(x, y, z) := (-1, -1, 2)$
while true **do**
$(x, y, z) := (x - y + 1, y - z, 2z + y - 1)$
or
$(x, y, z) := (-\frac{3}{2}, x + y + \frac{1}{2}, z + 1)$
or
$(x, y, z) := (2x, y + z, -x)$

Inductive invariant is a relation between variables of a loop $\mathcal{L}$ which is *preserved under any update of* $\mathcal{L}$.

$$x + y + z = 0$$

Strongest Algebraic Invariants

RESEARCH-ARTICLE OPEN ACCESS

Polynomial Invariants for Affine Programs

Authors: Ehud Hrushovski, Joël Ouaknine, Amaury Pouly, James Worrell Authors Info & Claims

LICS '18: Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science • July 2018 • Pages 530–539 • https://doi.org/10.1145/3209108.3209142

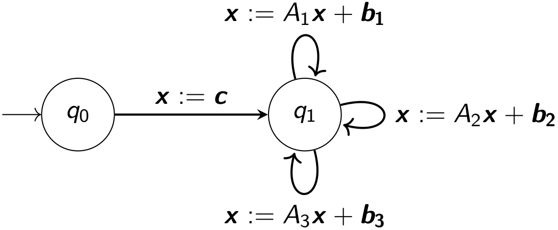
Online: 09 July 2018 Publication History

Algebraic invariants are those defined by polynomial equations. There exists the strongest algebraic invariant.

$$x^2 - y^3 = 0 \wedge y - 2z + 1 = 0$$

Multi-Path Affine Loops

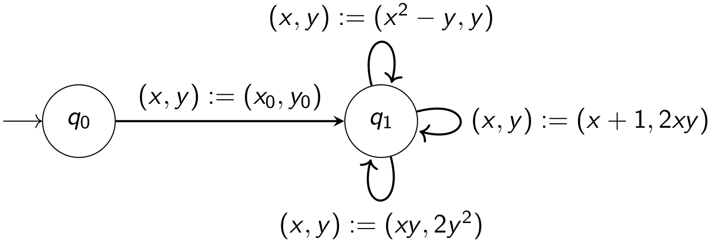
Hrushovski, Ouaknine, Pouly, Worrell: **an algorithm** to compute all polynomial equations among variables $\boldsymbol{x}$.



Invariants and Non-linearity

Proposition

Finding the strongest algebraic invariant of a multi-path loop with update degrees ≤ 2 is algorithmically **unsolvable**.

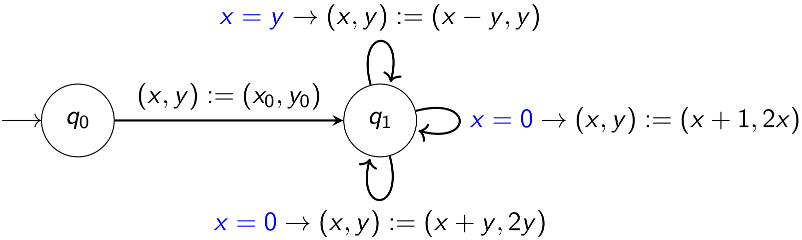


Reduction from Reset VASS Unboundedness.

What Else is Unsolvable?

Proposition

Finding the strongest algebraic invariant of a multi-path affine loop with guarded affine updates is algorithmically **unsolvable**.



Open questions

1. Halting for multi-path affine loops

Open questions

1. Halting for multi-path affine loops
2. Invariants for deterministic loops with non-affine updates

$$\textit{while}\ \text{true}\ \textit{do}\ (x, y) := (x^2 - y, xy)$$

Open questions

1. Halting for multi-path affine loops
2. Invariants for deterministic loops with non-affine updates

$$\textit{while} \;\; \text{true} \;\; \textit{do} \;\; (x, y) := (x^2 - y, xy)$$

3. Termination of linear-constraint loops

$$\textit{while} \;\; B\boldsymbol{x} \geq \boldsymbol{b} \;\; \textit{do} \;\; A \begin{bmatrix} \boldsymbol{x} & \boldsymbol{x'} \end{bmatrix}^T \leq \boldsymbol{c}$$

Take-away

Undecidable termination:

non-determinism +
affine updates +
linear inequality conditions

Unsolvable invariant generation:

non-determinism +
quadratic updates *or*
affine equality guards

Thank You!