Polynomial Loops: Beyond Termination

23rd Conference on Logic for Programming, Artificial Intelligence and Reasoning

Marcel Hark Florian Frohn Jürgen Giesl

Motivation

Setting the stage

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  $\rightarrow$ Witnesses for non-termination indicate implementation bugs.

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  - $\to$ Witnesses for non-termination indicate implementation bugs.
  - $\to$ Runtime bounds indicate efficiency of a program.

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  - $\rightarrow$ Witnesses for non-termination indicate implementation bugs.
  - $\rightarrow$ Runtime bounds indicate efficiency of a program.
- Drawback:

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  - $\rightarrow$ Witnesses for non-termination indicate implementation bugs.
  - $\rightarrow$ Runtime bounds indicate efficiency of a program.
- Drawback: Termination on a given input of general programs (Halting Problem) is undecidable [Turing 1937].

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  - $\rightarrow$ Witnesses for non-termination indicate implementation bugs.
  - $\rightarrow$ Runtime bounds indicate efficiency of a program.
- Drawback: Termination on a given input of general programs (Halting Problem) is undecidable [Turing 1937].
  - $\rightarrow$ No technique which can (dis-)prove Halting Problem for all programs.

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  - $\rightarrow$ Witnesses for non-termination indicate implementation bugs.
  - $\rightarrow$ Runtime bounds indicate efficiency of a program.
- Drawback: Termination on a given input of general programs (Halting Problem) is undecidable [Turing 1937].
  - $\rightarrow$ No technique which can (dis-)prove Halting Problem for all programs.
- Hope: Find sub-classes of programs where Halting Problem is decidable and runtime bounds can be computed.

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  $\rightarrow$ Witnesses for non-termination indicate implementation bugs.
  $\rightarrow$ Runtime bounds indicate efficiency of a program.
- Drawback: Termination on a given input of general programs (Halting Problem) is undecidable [Turing 1937].
  $\rightarrow$ No technique which can (dis-)prove Halting Problem for all programs.
- Hope: Find sub-classes of programs where Halting Problem is decidable and runtime bounds can be computed.
  $\rightarrow$ For Halting Problem: Linear loops [Li '17, Kincaid et al. '19]

$$\texttt{while}(\varphi)\{\quad \vec{x} \leftarrow A \cdot \vec{x} \quad\}$$

A has entries in $\mathbb{R}_{\mathbb{A}}$, φ a linear formula.

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  - $\rightarrow$ Witnesses for non-termination indicate implementation bugs.
  - $\rightarrow$ Runtime bounds indicate efficiency of a program.
- Drawback: Termination on a given input of general programs (Halting Problem) is undecidable [Turing 1937].
  - $\rightarrow$ No technique which can (dis-)prove Halting Problem for all programs.
- Hope: Find sub-classes of programs where Halting Problem is decidable and runtime bounds can be computed.
  - $\rightarrow$ For Halting Problem: Linear loops [Li '17, Kincaid et al. '19]

$$\texttt{while}\,(\varphi)\,\{\quad \vec{x} \leftarrow A \cdot \vec{x} \quad \}$$

A has entries in $\mathbb{R}_{\mathbb{A}}$, φ a linear formula.

- What about non-linear behavior in condition/update?

Motivation

Setting the stage

- Termination on a given input and Runtime Complexity are among the most important program properties.
  - $\rightarrow$ Witnesses for non-termination indicate implementation bugs.
  - $\rightarrow$ Runtime bounds indicate efficiency of a program.
- Drawback: Termination on a given input of general programs (Halting Problem) is undecidable [Turing 1937].
  - $\rightarrow$ No technique which can (dis-)prove Halting Problem for all programs.
- Hope: Find sub-classes of programs where Halting Problem is decidable and runtime bounds can be computed.
  - $\rightarrow$ For Halting Problem: Linear loops [Li '17, Kincaid et al. '19]

$$\texttt{while}(\varphi)\{ \quad \vec{x} \leftarrow A \cdot \vec{x} \quad \}$$

A has entries in $\mathbb{R}_{\mathbb{A}}$, φ a linear formula.

- What about non-linear behavior in condition/update?
- What about computability of runtime bounds?

Triangular Weakly Non-Linear Loops

Motivation

Triangular Weakly Non-Linear Loops

$$
\begin{array}{l}
\texttt{while}\,(\,\varphi\,)\,\{\\
\quad \vec{x} \leftarrow \vec{a}(\vec{x})\\
\}
\end{array}
$$

Motivation

Triangular Weakly Non-Linear Loops

```
while ( φ ) {
    x⃗ ← a⃗(x⃗)
}
```

- Crucial: Computability of closed form for the iterated update.

Motivation

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\,\varphi\,)\,\{\\ \quad \vec{x} \leftarrow \vec{a}(\vec{x})\\ \}$$

- Crucial: Computability of closed form for the iterated update.
  $\rightarrow$ Restrict ourselves to triangular weakly non-linear (twn) loops.

Motivation

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\varphi)\,\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

- Crucial: Computability of closed form for the iterated update.
  $\rightarrow$ Restrict ourselves to triangular weakly non-linear (twn) loops.

Motivation

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\,\varphi\,)\,\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

Motivation

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\varphi)\,\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

Motivation

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\varphi)\,\{ \quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$

Motivation

Triangular Weakly Non-Linear Loops

while (φ) {

$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$

}

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$

- Variable value depends at most linearly on its previous value.

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\varphi)\,\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$

- Variable value depends at most linearly on its previous value.
  - $\rightarrow$ Prohibits super-exponential growth as in $x_1 \leftarrow x_1^2$.

Motivation

Triangular Weakly Non-Linear Loops

```
while ( φ ) {
```

$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$

```
}
```

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$

- Variable value depends at most linearly on its previous value.
  - $\rightarrow$ Prohibits super-exponential growth as in $x_1 \leftarrow x_1^2$.

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\varphi)\,\{\quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$

- Variable value depends at most linearly on its previous value.
  - $\rightarrow$ Prohibits super-exponential growth as in $x_1 \leftarrow x_1^2$.
- Variable value depends polynomially on previous values of variables with higher index.

Motivation

Triangular Weakly Non-Linear Loops

$$\texttt{while}(\varphi)\{ \quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

φ built from $\land, \lor, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$

- Variable value depends at most linearly on its previous value.
  - $\rightarrow$ Prohibits super-exponential growth as in $x_1 \leftarrow x_1^2$.
- Variable value depends polynomially on previous values of variables with higher index.
  - $\rightarrow$ Prohibits circular dependencies.

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\varphi)\,\{\quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$

$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

- Variable value depends at most linearly on its previous value.
  $\rightarrow$ Prohibits super-exponential growth as in $x_1 \leftarrow x_1^2$.
- Variable value depends polynomially on previous values of variables with higher index.
  $\rightarrow$ Prohibits circular dependencies.

Motivation

Triangular Weakly Non-Linear Loops

$$\texttt{while}\,(\varphi)\,\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

φ built from $\wedge, \vee, (\neg)$ and
polynomial inequations over $\mathcal{S}$
$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$
$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

- Variable value depends at most linearly on its previous value.
  $\rightarrow$ Prohibits super-exponential growth as in $x_1 \leftarrow x_1^2$.
- Variable value depends polynomially on previous values of variables with higher index.
  $\rightarrow$ Prohibits circular dependencies.
- Coefficients from ring $\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$.

Motivation

Triangular Weakly Non-Linear Loops

```
while ( φ ) {
```

$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$

```
}
```

φ built from $\wedge, \vee, (\neg)$ and
polynomial inequations over $\mathcal{S}$
$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$
$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

- Variable value depends at most linearly on its previous value.
  $\rightarrow$ Prohibits super-exponential growth as in $x_1 \leftarrow x_1^2$.
- Variable value depends polynomially on previous values of variables with higher index.
  $\rightarrow$ Prohibits circular dependencies.
- Coefficients from ring $\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$.
- Variables range over $\mathcal{S}^d$.

Triangular Weakly Non-Linear Loops

```
while ( φ ) {
```
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
```
}
```

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \dots, x_d]$

$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

```
while ( x1 + x2^2 > 0 ) {
```
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} 1 \cdot x_1 + x_2^2 \cdot x_3 \\ 1 \cdot x_2 - 2 \cdot x_3^2 \\ 1 \cdot x_3 \end{bmatrix}$$
```
}
```

- Variable value depends at most linearly on its previous value.
  $\rightarrow$ Prohibits super-exponential growth as in $x_1 \leftarrow x_1^2$.
- Variable value depends polynomially on previous values of variables with higher index.
  $\rightarrow$ Prohibits circular dependencies.
- Coefficients from ring $\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$.
- Variables range over $\mathcal{S}^d$.

Motivation

Contribution

$$\texttt{while}\,(\varphi)\,\{\quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad\}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S},\ p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$

$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

Motivation

Contribution

$$\texttt{while}\,(\varphi)\,\{ \quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$

$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

- Halting Problem is decidable for $\mathcal{S} = \mathbb{R}_{\mathbb{A}}$.

Motivation

Contribution

$$\texttt{while}(\varphi)\{\quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$

$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

- Halting Problem is decidable for $\mathcal{S} = \mathbb{R}_{\mathbb{A}}$.
  $\rightarrow$ Halting problem is decidable for $\mathcal{S} \in \{\mathbb{Z}, \mathbb{Q}, \mathbb{R}_{\mathbb{A}}\}$.

Motivation

Contribution

$$\texttt{while}\,(\varphi)\,\{\quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

φ built from $\wedge, \vee, (\neg)$ and polynomial inequations over $\mathcal{S}$

$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$

$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

- Halting Problem is decidable for $\mathcal{S} = \mathbb{R}_{\mathbb{A}}$.
- Upper runtime bounds in size of input can always be computed if $\mathcal{S} = \mathbb{Z}$.

Motivation

Contribution

$$
\texttt{while}\,(\varphi)\,\{\quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}
$$

φ built from $\wedge, \vee, (\neg)$ and
polynomial inequations over $\mathcal{S}$
$c_i \in \mathcal{S}$, $p_i \in \mathcal{S}[x_{i+1}, \ldots, x_d]$
$\mathbb{Z} \leq \mathcal{S} \leq \mathbb{R}_{\mathbb{A}}$

- Halting Problem is decidable for $\mathcal{S} = \mathbb{R}_{\mathbb{A}}$.
- Upper runtime bounds in size of input can always be computed if $\mathcal{S} = \mathbb{Z}$.
  $\rightarrow$ Concept of size unclear over $\mathcal{S} \in \{\mathbb{Q}, \mathbb{R}_{\mathbb{A}}\}$ (see paper).

Outline

Computing Closed Forms

General Idea

$$\texttt{while}\,(\varphi)\,\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

Computing Closed Forms

General Idea

$$\texttt{while}\,(\varphi)\,\{ \quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

- Closed form $\vec{q}(\vec{e}, n)$ of the update: Values of $\vec{x}$ after n iterations expressed in initial values $\vec{e}$.

Computing Closed Forms

General Idea

$$\texttt{while}(\varphi)\{\quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

- Closed form $\vec{q}(\vec{e}, n)$ of the update: Values of $\vec{x}$ after n iterations expressed in initial values $\vec{e}$.
- Replace variables by closed form in condition

Computing Closed Forms

General Idea

$$\texttt{while}\,(\varphi)\,\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

- Closed form $\vec{q}(\vec{e}, n)$ of the update: Values of $\vec{x}$ after n iterations expressed in initial values $\vec{e}$.
- Replace variables by closed form in condition

$$\rightarrow \forall n \in \mathbb{N}.\ \varphi[\vec{x}/\vec{q}(\vec{e}, n)]$$

Computing Closed Forms

General Idea

$$\texttt{while}\,(\varphi)\,\{\quad \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix} \quad \}$$

- Closed form $\vec{q}(\vec{e}, n)$ of the update: Values of $\vec{x}$ after n iterations expressed in initial values $\vec{e}$.
- Replace variables by closed form in condition

$\to \forall n \in \mathbb{N}.\ \varphi[\vec{x}/\vec{q}(\vec{e}, n)] \iff \vec{e}$ witnesses non-termination.

Computing Closed Forms

General Idea

```
while(φ){
```

$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$

```
}
```

- Closed form $\vec{q}(\vec{e}, n)$ of the update: Values of $\vec{x}$ after n iterations expressed in initial values $\vec{e}$.
- Replace variables by closed form in condition

$\rightarrow \forall n \in \mathbb{N}.\ \varphi[\vec{x}/\vec{q}(\vec{e}, n)] \iff \vec{e}$ witnesses non-termination.

$\rightarrow M = \min_{n \in \mathbb{N}} \neg\varphi[\vec{x}/\vec{q}(\vec{e}, n)]$

Computing Closed Forms

General Idea

$$\texttt{while}(\varphi)\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

- Closed form $\vec{q}(\vec{e}, n)$ of the update: Values of $\vec{x}$ after n iterations expressed in initial values $\vec{e}$.
- Replace variables by closed form in condition

$\rightarrow \forall n \in \mathbb{N}.\ \varphi[\vec{x}/\vec{q}(\vec{e}, n)] \iff \vec{e}$ witnesses non-termination.

$\rightarrow M = \min_{n\in\mathbb{N}} \neg\varphi[\vec{x}/\vec{q}(\vec{e}, n)] \iff$ Loop does M iterations on $\vec{e}$.

Computing Closed Forms

Poly-Exponential Expressions

$$\texttt{while}\,(\,\varphi\,)\,\{$$

$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$

$$\}$$

Computing Closed Forms

Poly-Exponential Expressions

$$
\texttt{while}\,(\,\varphi\,)\,\{
$$

$$
\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}
$$

$$
\}
$$

- Closed forms for twn-loops: Poly-Exponential Expressions.

Computing Closed Forms

Poly-Exponential Expressions

$$\texttt{while}\,(\varphi)\,\{$$
$$\begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix} \leftarrow \begin{bmatrix} c_1 \cdot x_1 + p_1 \\ \vdots \\ c_d \cdot x_d + p_d \end{bmatrix}$$
$$\}$$

- Closed forms for twn-loops: Poly-Exponential Expressions.
- Closed form is computable.

Computing Closed Forms

Poly-Exponential Expressions

$$\texttt{while}\ (x_1 + x_2^2 > 0)\ \{$$
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$
$$\}$$

- Closed forms for twn-loops: Poly-Exponential Expressions.
- Closed form is computable.

Computing Closed Forms

Poly-Exponential Expressions

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{$$
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$
$$\}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

- Closed forms for twn-loops: Poly-Exponential Expressions.
- Closed form is computable.

Computing Closed Forms

Poly-Exponential Expressions

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{$$
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$
$$\}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

- Closed forms for twn-loops: Poly-Exponential Expressions.
- Closed form is computable.
- $x_4 \leftarrow 2 \cdot x_4$ has closed form $2^n \cdot e_4$.

Computing Closed Forms

Poly-Exponential Expressions

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$

$$\}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

Computing Closed Forms

Poly-Exponential Expressions

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$

$$\}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

Poly-Exponential Expressions

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{ \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix} \quad \}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

$$\left(\frac{4}{3} \cdot e_3^5\right) \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3 + 4 \cdot e_3^4\right) \cdot n^2$$
$$+ \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3 - 4 \cdot e_2 \cdot e_3^2\right) \cdot n + \left(e_1 + e_2^2\right)$$

Computing Closed Forms

Poly-Exponential Expressions

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{ \quad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix} \quad \}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

$$\left(\tfrac{4}{3} \cdot e_3^5\right) \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3 + 4 \cdot e_3^4\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \tfrac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3 - 4 \cdot e_2 \cdot e_3^2\right) \cdot n + \left(e_1 + e_2^2\right)$$

$\rightarrow$ Truth-value of condition after n iterations on initial values $\vec{e}$.

Dominant term

$$
\begin{aligned}
&\left(\tfrac{4}{3} \cdot e_3^5\right) \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3 + 4 \cdot e_3^4\right) \cdot n^2 \\
+ &\left(e_2^2 \cdot e_3 + \tfrac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3 - 4 \cdot e_2 \cdot e_3^2\right) \cdot n + \left(e_1 + e_2^2\right)
\end{aligned}
$$

Dominant term

$$\left(\tfrac{4}{3} \cdot e_3^5\right) \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3 + 4 \cdot e_3^4\right) \cdot n^2$$
$$+ \left(e_2^2 \cdot e_3 + \tfrac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3 - 4 \cdot e_2 \cdot e_3^2\right) \cdot n + \left(e_1 + e_2^2\right)$$

- Decide halting problem for $\vec{e} = (-4, 2, 1)$.

The Halting Problem

Dominant term

$$\frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3}$$

- Decide halting problem for $\vec{e} = (-4, 2, 1)$.

Dominant term

$$\frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3}$$

- Decide halting problem for $\vec{e} = (-4, 2, 1)$.
- $\frac{4}{3} \cdot n^3$ is dominant and positive.

The Halting Problem

Dominant term

$$\frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3}$$

- Decide halting problem for $\vec{e} = (-4, 2, 1)$.
- $\frac{4}{3} \cdot n^3$ is dominant and positive.
  $\rightarrow$ Expression is eventually positive $(n \rightarrow \infty)$.

Dominant term

$$\frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3}$$

- Decide halting problem for $\vec{e} = (-4, 2, 1)$.
- $\frac{4}{3} \cdot n^3$ is dominant and positive.

  $\rightarrow$ Expression is eventually positive ($n \rightarrow \infty$).

  $\rightarrow$ From certain N onwards dominant term dominates remaining expression.

Dominant term

$$\frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3}$$

- Decide halting problem for $\vec{e} = (-4, 2, 1)$.
- $\frac{4}{3} \cdot n^3$ is dominant and positive.

  $\rightarrow$ Expression is eventually positive ($n \rightarrow \infty$).

  $\rightarrow$ From certain N onwards dominant term dominates remaining expression.

  $\rightarrow$ Upper bound on N is always computable (see paper).

Dominant term

$$\frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3}$$

- Decide halting problem for $\vec{e} = (-4, 2, 1)$.
- $\frac{4}{3} \cdot n^3$ is dominant and positive.
  - $\rightarrow$ Expression is eventually positive ($n \rightarrow \infty$).
  - $\rightarrow$ From certain N onwards dominant term dominates remaining expression.
  - $\rightarrow$ Upper bound on N is always computable (see paper).
  - $\rightarrow$ Here $N \leq 4$.

The Halting Problem

Dominant term

$$\frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3}$$

- Decide halting problem for $\vec{e} = (-4, 2, 1)$.
- $\frac{4}{3} \cdot n^3$ is dominant and positive.
  - $\rightarrow$ Expression is eventually positive ($n \rightarrow \infty$).
  - $\rightarrow$ From certain N onwards dominant term dominates remaining expression.
  - $\rightarrow$ Upper bound on N is always computable (see paper).
  - $\rightarrow$ Here $N \leq 4$.

$$\Longrightarrow \ \forall n \geq 4. \ \frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3} > 0$$

Deciding the Halting Problem

$$\forall n \geq 4.\ \frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3} > 0$$

Deciding the Halting Problem

$$\forall n \geq 4.\ \tfrac{4}{3} \cdot n^3 - 2 \cdot n^2 + \tfrac{2}{3} > 0$$

- Truth value for $n \geq 4$ is known.

Deciding the Halting Problem

$$\forall n \geq 4.\ \frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3} > 0$$

- Truth value for $n \geq 4$ is known.
  $\rightarrow$ Only finitely many cases remain.

Deciding the Halting Problem

$$\forall n \geq 4.\ \frac{4}{3} \cdot n^3 - 2 \cdot n^2 + \frac{2}{3} > 0$$

- Truth value for $n \geq 4$ is known.
  - $\rightarrow$ Only finitely many cases remain.
  - $\rightarrow$ Here: $n = 0, 1, 2, 3$.

Deciding the Halting Problem

$$\forall n \geq 4.\ \tfrac{4}{3} \cdot n^3 - 2 \cdot n^2 + \tfrac{2}{3} > 0$$

- Truth value for $n \geq 4$ is known.
  - $\rightarrow$ Only finitely many cases remain.
  - $\rightarrow$ Here: $n = 0, 1, 2, 3$.

$$\tfrac{4}{3} \cdot 0^3 - 2 \cdot 0^2 + \tfrac{2}{3} > 0 \qquad \tfrac{4}{3} \cdot 1^3 - 2 \cdot 1^2 + \tfrac{2}{3} < 0$$

$$\tfrac{4}{3} \cdot 2^3 - 2 \cdot 2^2 + \tfrac{2}{3} > 0 \qquad \tfrac{4}{3} \cdot 3^3 - 2 \cdot 3^2 + \tfrac{2}{3} > 0$$

Deciding the Halting Problem

$$\forall n \geq 4.\ \tfrac{4}{3} \cdot n^3 - 2 \cdot n^2 + \tfrac{2}{3} > 0$$

- Truth value for $n \geq 4$ is known.
  - $\rightarrow$ Only finitely many cases remain.
  - $\rightarrow$ Here: $n = 0, 1, 2, 3$.

$$\begin{array}{ll} \tfrac{4}{3} \cdot 0^3 - 2 \cdot 0^2 + \tfrac{2}{3} > 0 & \tfrac{4}{3} \cdot 1^3 - 2 \cdot 1^2 + \tfrac{2}{3} < 0 \\ \tfrac{4}{3} \cdot 2^3 - 2 \cdot 2^2 + \tfrac{2}{3} > 0 & \tfrac{4}{3} \cdot 3^3 - 2 \cdot 3^2 + \tfrac{2}{3} > 0 \end{array}$$

$\rightarrow$ Loop halts on $\vec{e} = (-4, 2, 1)$.

Deciding the Halting Problem

- Truth value for $n \geq N$ is known.
  $\rightarrow$ Only finitely many cases remain.

Deciding the Halting Problem

- Truth value for $n \geq N$ is known.
  $\rightarrow$ Only finitely many cases remain.

  $\rightarrow$ The Halting Problem for twn-loops is decidable.

The Halting Problem

Deciding the Halting Problem

- Truth value for $n \geq N$ is known.

  $\rightarrow$ Only finitely many cases remain.

  $\rightarrow$ The Halting Problem for twn-loops is decidable.

  $\rightarrow$ Witnesses for Non-Termination are enumerable.

Outline

Runtime Bounds

Reminder

Runtime Bounds

Reminder

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{$$
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$
$$\}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

$$\left(\frac{4}{3} \cdot e_3^5\right) \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3 + 4 \cdot e_3^4\right) \cdot n^2$$
$$+ \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3 - 4 \cdot e_2 \cdot e_3^2\right) \cdot n + \left(e_1 + e_2^2\right)$$

Runtime Bounds

Reminder

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{$$
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$
$$\}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

$$\left(\frac{4}{3} \cdot e_3^5\right) \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3 + 4 \cdot e_3^4\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3 - 4 \cdot e_2 \cdot e_3^2\right) \cdot n + \left(e_1 + e_2^2\right)$$

$\rightarrow$ Truth-value of condition after n iterations on initial values $\vec{e}$.

Runtime Bounds

Reminder

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$

$$\}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

$$\left(\tfrac{4}{3} \cdot e_3^5\right) \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3 + 4 \cdot e_3^4\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \tfrac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3 - 4 \cdot e_2 \cdot e_3^2\right) \cdot n + \left(e_1 + e_2^2\right)$$

$\rightarrow$ Truth-value of condition after n iterations on initial values $\vec{e}$.

$\rightarrow$ Termination reached whenever expression ≤ 0.

Runtime Bounds

Reminder

$$\texttt{while}\left(x_1 + x_2^2 > 0\right)\{$$
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$
$$\}$$

$$\begin{bmatrix} e_1 + \frac{4}{3} \cdot e_3^5 \cdot n^3 + \left(-2 \cdot e_3^5 - 2 \cdot e_2 \cdot e_3^3\right) \cdot n^2 + \left(e_2^2 \cdot e_3 + \frac{2}{3} \cdot e_3^5 + 2 \cdot e_2 \cdot e_3^3\right) \cdot n \\ e_2 - 2 \cdot e_3^2 \cdot n \\ e_3 \end{bmatrix}$$

$$\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2$$
$$+ \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

→ Truth-value of condition after n iterations on initial values $\vec{e}$.

→ Termination reached whenever expression ≤ 0.

→ Rescale for simplicity.

Dominant Term

$$\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2$$
$$+ \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

Dominant Term

$$(4 \cdot e_3^5) \cdot n^3 + (-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4) \cdot n^2$$
$$+ (3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2) \cdot n + 3 \cdot (e_1 + e_2^2)$$

- If $e_3 \neq 0 \implies (4 \cdot e_3^5) \cdot n^3$ is dominant.

Dominant Term

$$(4 \cdot e_3^5) \cdot n^3 + (-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4) \cdot n^2$$
$$+ (3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2) \cdot n + 3 \cdot (e_1 + e_2^2)$$

- If $e_3 \neq 0 \implies (4 \cdot e_3^5) \cdot n^3$ is dominant.

  $\rightarrow$ From certain $N(\vec{e})$ onwards dominant term dominates remaining expression.

Runtime Bounds

Dominant Term

$$(4 \cdot e_3^5) \cdot n^3 + (-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4) \cdot n^2$$
$$+ (3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2) \cdot n + 3 \cdot (e_1 + e_2^2)$$

- If $e_3 \neq 0 \implies (4 \cdot e_3^5) \cdot n^3$ is dominant.

  $\to$ From certain $N(\vec{e})$ onwards dominant term dominates remaining expression.

  $\to$ Upper bound on $N(\vec{e})$ depending only on $||\vec{e}||$ for arbitrary $\vec{e}$ is always computable.

Dominant Term

$$\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2$$
$$+ \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

- If $e_3 \neq 0 \implies \left(4 \cdot e_3^5\right) \cdot n^3$ is dominant.

  $\to$ From certain $N(\vec{e})$ onwards dominant term dominates remaining expression.
  $\to$ Upper bound on $N(\vec{e})$ depending only on $\|\vec{e}\|$ for arbitrary $\vec{e}$ is always computable.
  $\to$ Yields upper bound on runtime.

Runtime Bounds

Upper Bound

$$\begin{aligned}&\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 \\ +&\left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)\end{aligned}$$

Runtime Bounds

Upper Bound

$$\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2$$
$$+ \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

- Idea: compute bound such that n^3 dominates remaining expression.

Runtime Bounds

Upper Bound

$$n^3 > \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 + \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

- Idea: compute bound such that n^3 dominates remaining expression.

Upper Bound

$$n^3 > \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 + \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

- Idea: compute bound such that n^3 dominates remaining expression.
  $\rightarrow$ Compute upper bound on all coefficients first.

Upper Bound

$$n^3 > \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 + \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

- Idea: compute bound such that n^3 dominates remaining expression.
  $\rightarrow$ Compute upper bound on all coefficients first.

$$\begin{aligned}
-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4 &\leq 12 \cdot \textstyle\sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j \\
3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2 &\leq 12 \cdot \textstyle\sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j \\
3 \cdot \left(e_1 + e_2^2\right) &\leq 12 \cdot \textstyle\sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j
\end{aligned}$$

Runtime Bounds

Upper Bound

$$n^3 > \left(12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j\right) \cdot \left(n^2 + n + 1\right)$$

- Idea: compute bound such that n^3 dominates remaining expression.
  $\rightarrow$ Compute upper bound on all coefficients first.

$$\begin{aligned}
-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4 &\leq 12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j \\
3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2 &\leq 12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j \\
3 \cdot \left(e_1 + e_2^2\right) &\leq 12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j
\end{aligned}$$

Upper Bound

$$n^3 > \left(12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j\right) \cdot (n^2 + n + 1)$$

- Idea: compute bound such that n^3 dominates remaining expression.

Runtime Bounds

Upper Bound

$$n^3 > \left(12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j\right) \cdot (n^2 + n + 1)$$

- Idea: compute bound such that n^3 dominates remaining expression.
  - $\rightarrow$ Combine the terms of lower order.

Runtime Bounds

Upper Bound

$$n^3 > \left(12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j\right) \cdot (n^2 + n + 1)$$

- Idea: compute bound such that n^3 dominates remaining expression.
  - $\rightarrow$ Combine the terms of lower order.
  - $\rightarrow$ For $n > 1$ we have $n^2 > n + 1$

Runtime Bounds

Upper Bound

$$n^3 > \left(12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j\right) \cdot (n^2 + n^2)$$

- Idea: compute bound such that n^3 dominates remaining expression.

  $\rightarrow$ Combine the terms of lower order.

  $\rightarrow$ For $n > 1$ we have $n^2 > n + 1$

Runtime Bounds

Upper Bound

$$n^3 > \left(12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j\right) \cdot 2 \cdot n^2$$

- Idea: compute bound such that n^3 dominates remaining expression.

  $\to$ Combine the terms of lower order.

  $\to$ For $n > 1$ we have $n^2 > n + 1$

Runtime Bounds

Upper Bound

$$n > \left(12 \cdot \sum_{j=0}^{5}(|e_1| + |e_2| + |e_3|)^j\right) \cdot 2$$

- Idea: compute bound such that n^3 dominates remaining expression.
  - $\rightarrow$ Combine the terms of lower order.
  - $\rightarrow$ For $n > 1$ we have $n^2 > n + 1$

Runtime Bounds

Upper Bound

Runtime Bounds

Upper Bound

$$n^3 > \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 + \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

$$\Uparrow$$

$$n > \left(12 \cdot \sum_{j=0}^{5} (|e_1| + |e_2| + |e_3|)^j\right) \cdot 2$$

Runtime Bounds

Upper Bound

$$n^3 > \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 + \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

$$\Uparrow$$

$$n > \left(12 \cdot \sum_{j=0}^{5} (|e_1| + |e_2| + |e_3|)^j\right) \cdot 2$$

$$\begin{aligned} &\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 \\ &+ \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right) \end{aligned}$$

Upper Bound

$$n^3 > \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 + \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

$$\Uparrow$$

$$n > \left(12 \cdot \sum_{j=0}^{5} (|e_1| + |e_2| + |e_3|)^j\right) \cdot 2$$

$$\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2$$
$$+ \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

$\rightarrow$ At this point, the dominant term dominates the remaining expression.

Runtime Bounds

Upper Bound

$$n^3 > \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 + \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

$$\Uparrow$$

$$n > \left(12 \cdot \sum_{j=0}^{5} (|e_1| + |e_2| + |e_3|)^j\right) \cdot 2$$

$$\begin{aligned} &\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 \\ &+ \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right) \end{aligned}$$

$\rightarrow$ At this point, the dominant term dominates the remaining expression.

$\rightarrow$ At this point, the loop has terminated on $\vec{e}$ or never will.

Runtime Bounds

Upper Bound

$$n^3 > \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 + \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right)$$

$$\Uparrow$$

$$n > \left(12 \cdot \sum_{j=0}^{5} \|\vec{e}\|^j\right) \cdot 2$$

$$\begin{aligned} &\left(4 \cdot e_3^5\right) \cdot n^3 + \left(-6 \cdot e_3^5 - 6 \cdot e_2 \cdot e_3^3 + 12 \cdot e_3^4\right) \cdot n^2 \\ &+ \left(3 \cdot e_2^2 \cdot e_3 + 2 \cdot e_3^5 + 6 \cdot e_2 \cdot e_3^3 - 12 \cdot e_2 \cdot e_3^2\right) \cdot n + 3 \cdot \left(e_1 + e_2^2\right) \end{aligned}$$

$\rightarrow$ At this point, the dominant term dominates the remaining expression.

$\rightarrow$ At this point, the loop has terminated on $\vec{e}$ or never will.

Runtime Bounds

Upper Bound

Runtime Bounds

Upper Bound

$$\texttt{while}\left(x_1 + x_2^2 > 0\right) \{$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}$$

$$\}$$

Runtime Bounds

Upper Bound

$$
\begin{array}{l}
\texttt{while}\ (x_1 + x_2^2 > 0)\ \{ \\
\qquad \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix} \\
\}
\end{array}
$$

If loop terminates, termination within $\left(12 \cdot \sum_{j=0}^{5} \|\vec{e}\|^{j}\right) \cdot 2$ iterations.

Runtime Bounds

Upper Bound

$$
\texttt{while}\left(x_1 + x_2^2 > 0\right)\{
$$
$$
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}
$$
$$
\}
$$

If loop terminates, termination within $\left(12 \cdot \sum_{j=0}^{5} \|\vec{e}\|^j\right) \cdot 2$ iterations.

$\rightarrow$ Polynomial runtime bound for twn-loop over integers.

Runtime Bounds

Upper Bound

$$
\texttt{while}\ (x_1 + x_2^2 > 0)\ \{
$$

$$
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leftarrow \begin{bmatrix} x_1 + x_2^2 \cdot x_3 \\ x_2 - 2 \cdot x_3^2 \\ x_3 \end{bmatrix}
$$

$$
\}
$$

If loop terminates, termination within $\left(12 \cdot \sum_{j=0}^{5} \|\vec{e}\|^j\right) \cdot 2$ iterations.

$\rightarrow$ Polynomial runtime bound for twn-loop over integers.

$\rightarrow$ Linear twn-loops over integers have at most linear runtime (see paper).

Conclusion

Summary

Conclusion

Summary

- Halting problem for twn-loops is decidable.

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.
- Can handle loops out of reach for incomplete techniques using (lexicographic) ranking functions (see paper).

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.
- Can handle loops out of reach for incomplete techniques using (lexicographic) ranking functions (see paper).

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.
- Can handle loops out of reach for incomplete techniques using (lexicographic) ranking functions (see paper).

Future Work

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.
- Can handle loops out of reach for incomplete techniques using (lexicographic) ranking functions (see paper).

Future Work

- Handling of non-twn loops.

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.
- Can handle loops out of reach for incomplete techniques using (lexicographic) ranking functions (see paper).

Future Work

- Handling of non-twn loops.
- Investigate tightness of upper runtime bounds.

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.
- Can handle loops out of reach for incomplete techniques using (lexicographic) ranking functions (see paper).

Future Work

- Handling of non-twn loops.
- Investigate tightness of upper runtime bounds.
- Asymptotic bound for runtime directly from loop.

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.
- Can handle loops out of reach for incomplete techniques using (lexicographic) ranking functions (see paper).

Future Work

- Handling of non-twn loops.
- Investigate tightness of upper runtime bounds.
- Asymptotic bound for runtime directly from loop.

Conclusion

Summary

- Halting problem for twn-loops is decidable.
- Witnesses for non-termination are enumerable.
- Upper runtime bound for integer twn-loops is always computable.
- Can handle loops out of reach for incomplete techniques using (lexicographic) ranking functions (see paper).

Future Work

- Handling of non-twn loops.
- Investigate tightness of upper runtime bounds.
- Asymptotic bound for runtime directly from loop.

Thank you