

Generating Loop Invariants via Polynomial Interpolation

Marc Moreno Maza
Joint work with
Rong Xiao

University of Western Ontario, Canada

ASCM 2012 - Beijing
Key Laboratory of Mathematics Mechanization
October 27, 2012

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: ProgramAnalysis

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: ProgramAnalysis

Loop model under study

```
while  $C_0$  do  
  if  $C_1$   
  then  
     $X := A_1(X)$ ;  
  elif  $C_2$   
  then  
     $X := A_2(X)$ ;  
  ...  
  elif  $C_m$   
  then  
     $X := A_m(X)$ ;  
  end if  
end while
```

- 1 Loop variables: $X = x_1, \dots, x_s$, rational value scalar
- 2 Conditions: each C_i is a quantifier free formula in X over \mathbb{Q} .
- 3 Assignments: $A_i \in \mathbb{Q}[X]$ inducing a polynomial map $M_i : \mathbb{R}^s \mapsto \mathbb{R}^s$
- 4 Initial condition: X -values defined by a semi-algebraic system.

Basic notions

```
x := a;  
y := b;  
while x < 10 do  
    x := x + y5;  
    y := y + 1;  
end do;
```

- x, y, a, b are **loop variables** since they are updated in the loop or used to update other loop variables.
- The set of the **initial values** of the loop is
$$\{(x, y, a, b) \mid x = a, y = b, (a, b) \in \mathbb{R}^2\}.$$
- The **loop trajectory** of the above loop starting at $(x, y, a, b) = (1, 0, 1, 0)$ is the sequence:

$$(1, 0, 1, 0), (1, 1, 1, 0), (2, 2, 1, 0), (34, 3, 1, 0).$$

- The **reachable set** $R(L)$ of a loop L consists of all tuples of all trajectories of L .
- If x_1, \dots, x_s are the loop variables of L , then a polynomial $P \in \mathbb{Q}[x_1, \dots, x_s]$ is a **(plain) loop invariant** of L whenever $R(L) \subseteq V(P)$ holds.

More notions

- The **inductive reachable set** $R_{\text{ind}}(L)$ of a loop L is the reachable set of the loop obtained from L by replacing the **guard condition** with **true**.
- The **absolute reachable set** $R_{\text{abs}}(L)$ of a loop L is the reachable set of the loop obtained from L by replacing the **guard condition** with **true**, **ignoring the branch conditions** and, at each iteration executing a branch action selected randomly.
- We clearly have

$$R(L) \subseteq R_{\text{ind}} \subseteq R_{\text{abs}}$$

More notions

- The **inductive reachable set** $R_{\text{ind}}(L)$ of a loop L is the reachable set of the loop obtained from L by replacing the **guard condition** with **true**.
- The **absolute reachable set** $R_{\text{abs}}(L)$ of a loop L is the reachable set of the loop obtained from L by replacing the **guard condition** with **true**, **ignoring the branch conditions** and, at each iteration executing a branch action selected randomly.
- We clearly have

$$R(L) \subseteq R_{\text{ind}} \subseteq R_{\text{abs}}$$

- If x_1, \dots, x_s are the loop variables of L , then a polynomial $P \in \mathbb{Q}[x_1, \dots, x_s]$ is an **inductive (resp. absolute) loop invariant** of L whenever $R_{\text{ind}}(L) \subseteq V(P)$ (resp. $R_{\text{abs}}(L) \subseteq V(P)$) holds.
- We denote by $\mathcal{I}(L)$ (resp. $\mathcal{I}_{\text{ind}}(L), \mathcal{I}_{\text{abs}}(L)$) the set of the polynomials that are plain (resp. inductive, absolute) loop invariants of L .
- These are radical ideals such that

$$\mathcal{I}_{\text{abs}}(L) \subseteq \mathcal{I}_{\text{ind}}(L) \subseteq \mathcal{I}(L)$$

Absolute invariants might be trivial

```
 $y_1 := 0;$   
 $y_2 := 0;$   
 $y_3 := x_1;$   
while  $y_3 \neq 0$  do  
  if  $y_2 + 1 = x_2$   
  then  
     $y_1 := y_1 + 1;$   
     $y_2 := 0;$   
     $y_3 := y_3 - 1;$   
  else  
     $y_2 := y_2 + 1;$   
     $y_3 := y_3 - 1;$   
  end if  
end do
```

- Consider $y_1x_2 + y_2 + y_3 = x_1$ (E).
- If $x_1 = 0$ then the equation (E) holds initially and the loop is not entered.
- If $x_1 \neq 0$ and $x_2 = 1$ then (E) and $y_2 + 1 = x_2$ hold before each iteration.
- If $x_1 \neq 0$ and $x_2 \neq 1$ then the second action preserves (E).
- Therefore $y_1x_2 + y_2 + y_3 - x_1 \in \mathcal{I}(L)$ and $y_1x_2 + y_2 + y_3 - x_1 \in \mathcal{I}_{\text{ind}}(L)$ both hold.

Absolute invariants might be trivial

```

 $y_1 := 0;$ 
 $y_2 := 0;$ 
 $y_3 := x_1;$ 
while  $y_3 \neq 0$  do
  if  $y_2 + 1 = x_2$ 
  then
     $y_1 := y_1 + 1;$ 
     $y_2 := 0;$ 
     $y_3 := y_3 - 1;$ 
  else
     $y_2 := y_2 + 1;$ 
     $y_3 := y_3 - 1;$ 
  end if
end do

```

- Consider $y_1x_2 + y_2 + y_3 = x_1$ (E).
- If $x_1 = 0$ then the equation (E) holds initially and the loop is not entered.
- If $x_1 \neq 0$ and $x_2 = 1$ then (E) and $y_2 + 1 = x_2$ hold before each iteration.
- If $x_1 \neq 0$ and $x_2 \neq 1$ then the second action preserves (E).
- Therefore $y_1x_2 + y_2 + y_3 - x_1 \in \mathcal{I}(L)$ and $y_1x_2 + y_2 + y_3 - x_1 \in \mathcal{I}_{\text{ind}}(L)$ both hold.
- If conditions are ignored, $(x_1, x_2) = (0, 1)$ and execute the first branch once, then we obtain

$$y_1x_2 = 1 \text{ and } y_2 + y_3 = x_1.$$
- Then (E) is violated and we have

$$\mathcal{I}_{\text{abs}}(L) = \langle 0 \rangle.$$

Inductive invariants might not be plain invariants

```
 $x := 1;$   
while  $x \neq 1$  do  
     $x := x + 1;$   
end do
```

- $x - 1 = 0$ is an invariant but not an inductive of the following loop.
- Thus $\mathcal{I}_{\text{ind}}(L)$ is strictly smaller than $\mathcal{I}(L)$

Computing inductive invariants via elimination ideals

```

y := 1;
x := 0;
while true do
  z := x;
  x := y;
  y := z + y;
end while
  
```

- Solving for (x, y) as a 2-variable recurrence

$$x(n+1) = y(n), y(n+1) = x(n) + y(n), \text{ with } x(0) = 0, y(0) = 1.$$

- We obtain

$$x(n) = \frac{(\frac{\sqrt{5}+1}{2})^n}{\sqrt{5}} - \frac{(-\frac{\sqrt{5}+1}{2})^n}{\sqrt{5}},$$

$$y(n) = \frac{\sqrt{5}+1}{2} \frac{(\frac{\sqrt{5}+1}{2})^n}{\sqrt{5}} - \frac{-\sqrt{5}+1}{2} \frac{(-\frac{\sqrt{5}+1}{2})^n}{\sqrt{5}}.$$

- Let $u = (\frac{\sqrt{5}+1}{2})^n$, $v = (-\frac{\sqrt{5}+1}{2})^n$, $a = \sqrt{5}$
- Taking the dependencies $u^2 v^2 = 1, a^2 = 5$ into account, we want

$$\langle x - \frac{au}{5} + \frac{av}{5}, y - a\frac{a+1}{2}\frac{u}{5} + a\frac{-a+1}{2}\frac{v}{5}, a^2 - 5, u^2 v^2 - 1 \rangle \cap \mathbb{Q}[x, y],$$

- which is

$$\langle 1 - y^4 + 2xy^3 + x^2y^2 - 2x^3y - x^4 \rangle.$$

Summary and notes

- Computing $\mathcal{I}_{\text{ind}}(L)$ is a better approximation of $\mathcal{I}(L)$ than $\mathcal{I}_{\text{abs}}(L)$.
- The loop invariant generation methods of (E. Rodriguez-Carbonell & D. Kapur, ISSAC04) and (L. Kovács, TACAS08) focus on $\mathcal{I}_{\text{abs}}(L)$.

Summary and notes

- Computing $\mathcal{I}_{\text{ind}}(L)$ is a better approximation of $\mathcal{I}(L)$ than $\mathcal{I}_{\text{abs}}(L)$.
- The loop invariant generation methods of (E. Rodriguez-Carbonell & D. Kapur, ISSAC04) and (L. Kovács, TACAS08) focus on $\mathcal{I}_{\text{abs}}(L)$.
- In this talk, we target $\mathcal{I}_{\text{ind}}(L)$ (easier to compute than $\mathcal{I}(L)$) and call it the **Invariant Ideal** of the loop L . Same goal as in (Bin Wu, Liyong Shen, Min Wu, Zhengfeng Yang & Zhenbing Zeng, 2011).

Summary and notes

- Computing $\mathcal{I}_{\text{ind}}(L)$ is a better approximation of $\mathcal{I}(L)$ than $\mathcal{I}_{\text{abs}}(L)$.
- The loop invariant generation methods of (E. Rodriguez-Carbonell & D. Kapur, ISSAC04) and (L. Kovács, TACAS08) focus on $\mathcal{I}_{\text{abs}}(L)$.
- In this talk, we target $\mathcal{I}_{\text{ind}}(L)$ (easier to compute than $\mathcal{I}(L)$) and call it the **Invariant Ideal** of the loop L . Same goal as in (Bin Wu, Liyong Shen, Min Wu, Zhengfeng Yang & Zhenbing Zeng, 2011).
- We also want to avoid computing closed forms of loop variables, while
 - not making any assumptions on the shape of the polynomial invariants,
 - and avoiding an intensive use of expensive algebraic computations other than linear algebra, for which costs are predictable.

Summary and notes

- Computing $\mathcal{I}_{\text{ind}}(L)$ is a better approximation of $\mathcal{I}(L)$ than $\mathcal{I}_{\text{abs}}(L)$.
- The loop invariant generation methods of (E. Rodriguez-Carbonell & D. Kapur, ISSAC04) and (L. Kovács, TACAS08) focus on $\mathcal{I}_{\text{abs}}(L)$.
- In this talk, we target $\mathcal{I}_{\text{ind}}(L)$ (easier to compute than $\mathcal{I}(L)$) and call it the **Invariant Ideal** of the loop L . Same goal as in (Bin Wu, Liyong Shen, Min Wu, Zhengfeng Yang & Zhenbing Zeng, 2011).
- We also want to avoid computing closed forms of loop variables, while
 - not making any assumptions on the shape of the polynomial invariants,
 - and avoiding an intensive use of expensive algebraic computations other than linear algebra, for which costs are predictable.
- In (Sankaranarayanan, Sipma & Manna, SIGPLAN 2004) (Y. Chen, B. Xia, L. Yang, & N. Zhan, FMHRTS 2007) (D. Kapur Deduction and Applications 2005) template polynomials are used. Moreover, the latter two use real QE.

Summary and notes

- Computing $\mathcal{I}_{\text{ind}}(L)$ is a better approximation of $\mathcal{I}(L)$ than $\mathcal{I}_{\text{abs}}(L)$.
- The loop invariant generation methods of (E. Rodriguez-Carbonell & D. Kapur, ISSAC04) and (L. Kovács, TACAS08) focus on $\mathcal{I}_{\text{abs}}(L)$.
- In this talk, we target $\mathcal{I}_{\text{ind}}(L)$ (easier to compute than $\mathcal{I}(L)$) and call it the **Invariant Ideal** of the loop L . Same goal as in (Bin Wu, Liyong Shen, Min Wu, Zhengfeng Yang & Zhenbing Zeng, 2011).
- We also want to avoid computing closed forms of loop variables, while
 - not making any assumptions on the shape of the polynomial invariants,
 - and avoiding an intensive use of expensive algebraic computations other than linear algebra, for which costs are predictable.
- In (Sankaranarayanan, Sipma & Manna, SIGPLAN 2004) (Y. Chen, B. Xia, L. Yang, & N. Zhan, FMHRTS 2007) (D. Kapur Deduction and Applications 2005) template polynomials are used. Moreover, the latter two use real QE.
- The "abstract interpretation" method (E. Rodriguez-Carbonell & D. Kapur, Science of Computer Programming 2007) does not use templates but uses of Gröbner bases heavily.

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: ProgramAnalysis

Poly-geometrical expression

Notations

Let $\alpha_1, \dots, \alpha_k$ be k elements of $\overline{\mathbb{Q}}^* \setminus \{1\}$. Let n be a variable taking non-negative integer values. We regard $n, \alpha_1^n, \dots, \alpha_k^n$ as independent variables and we call $\alpha_1^n, \dots, \alpha_k^n$ **n -exponential variables**.

Definition

Any $f \in \overline{\mathbb{Q}}[n, \alpha_1^n, \dots, \alpha_k^n]$ is called a **poly-geometrical expression in n over $\overline{\mathbb{Q}}$ w.r.t. $\alpha_1, \dots, \alpha_k$** . For such an f , we denote by $f|_{n=i}$ the **evaluation** of f at i . For such f, g we write $f = g$ whenever $f|_{n=i} = g|_{n=i}$ holds for all i .

Examples of poly-geometrical expressions

Example

The closed form $f := \frac{(n+1)^2 n^2}{4}$ of $\sum_{i=0}^n i^3$ is a poly-geometrical expression in n over $\overline{\mathbb{Q}}$ without n -exponential variables.

Example

The expression $g := n^2 2^{(n+1)} - n 2^n 3^{\frac{n}{2}}$ is a poly-geometrical in n over $\overline{\mathbb{Q}}$ w.r.t. $2, \sqrt{3}$.

Example

The sum $\sum_{i=1}^{n-1} i^k$ has $n - 1$ terms while its closed form below

$$\sum_{i=1}^k \left\{ \begin{matrix} k \\ i \end{matrix} \right\} \frac{n^{i+1}}{i+1},$$

where $\left\{ \begin{matrix} k \\ i \end{matrix} \right\}$ the number of ways to partition k into i non-zero summands, has a fixed number of terms and thus is poly-geometrical in n over $\overline{\mathbb{Q}}$.

Multiplicative relation ideal: example

Definition

Let $A := (\alpha_1, \dots, \alpha_k)$ be a sequence of k elements of $\overline{\mathbb{Q}}$. Assume w.l.o.g. that for some ℓ , with $1 \leq \ell \leq k$, we have $\alpha_1 \neq 0, \dots, \alpha_\ell \neq 0$, $\alpha_{\ell+1} = \dots = \alpha_k = 0$. We associate each α_i with a “new” variable y_i . The binomial ideal $\text{MRI}(A; y_1, \dots, y_k)$ of $\mathbb{Q}[y_1, y_2, \dots, y_k]$ generated by

$$\left\{ \prod_{j \in \{1, \dots, \ell\}, v_j > 0} y_j^{v_j} - \prod_{i \in \{1, \dots, \ell\}, v_i < 0} y_i^{-v_i} \mid (v_1, \dots, v_\ell) \in Z \right\},$$

and $\{y_{\ell+1}, \dots, y_k\}$, where Z is the multiplicative relation lattice.

Example

Consider $A = (1/2, 1/3, -1/6, 0)$. The multiplicative relation lattice of $(1/2, 1/3, -1/6)$ is generated by $(2, 2, -2)$. Thus the MRI of A associated with y_1, y_2, y_3, y_4 is

$$\langle y_1^2 y_2^2 - y_3^2, y_4 \rangle.$$

Degree estimates for x satisfying $x(n+1) = \lambda x(n) + h(n)$

Lemma

Let $\alpha_1, \dots, \alpha_k \in \overline{\mathbb{Q}} \setminus \{0, 1\}$. Let $\lambda \in \overline{\mathbb{Q}} \setminus \{0\}$. Let $h(n) \in \overline{\mathbb{Q}}[n, \alpha_1^n, \dots, \alpha_k^n]$. Consider the following single-variable recurrence relation R :

$$x(n+1) = \lambda x(n) + h(n).$$

Then, there exists $s(n) \in \overline{\mathbb{Q}}[n, \alpha_1^n, \dots, \alpha_k^n]$ such that we have

$$\deg(s(n), \alpha_i^n) \leq \deg(h(n), \alpha_i^n) \quad \text{and} \quad \deg(s(n), n) \leq \deg(h(n), n) + 1,$$

and such that

- if $\lambda = 1$ holds, then $s(n)$ solves R ,
- if $\lambda \neq 1$ holds, then there exists a constant c depending on $x(0)$ (that is, the initial value of x) such that $c\lambda^n + s(n)$ solves R .

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: ProgramAnalysis

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: `ProgramAnalysis`

The multivariate case: setting

Let n_1, \dots, n_k be positive integers and define $s := n_1 + \dots + n_k$. Let M be a block-diagonal square matrix over \mathbb{Q} of order s , with shape:

$$M := \begin{pmatrix} \mathbf{M}_{n_1 \times n_1} & \mathbf{0}_{n_1 \times n_2} & \cdots & \mathbf{0}_{n_1 \times n_k} \\ \mathbf{0}_{n_2 \times n_1} & \mathbf{M}_{n_2 \times n_2} & \cdots & \mathbf{0}_{n_2 \times n_k} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{0}_{n_k \times n_1} & \mathbf{0}_{n_k \times n_2} & \cdots & \mathbf{M}_{n_k \times n_k} \end{pmatrix}.$$

Consider an s -variable recurrence relation R in x_1, x_2, \dots, x_s , with shape:

$$\begin{pmatrix} x_1(n+1) \\ x_2(n+1) \\ \vdots \\ x_s(n+1) \end{pmatrix} = M \times \begin{pmatrix} x_1(n) \\ x_2(n) \\ \vdots \\ x_s(n) \end{pmatrix} + \begin{pmatrix} \mathbf{f}_{1n_1 \times 1} \\ \mathbf{f}_{2n_2 \times 1} \\ \vdots \\ \mathbf{f}_{kn_k \times 1} \end{pmatrix},$$

where \mathbf{f}_1 is a vector of length n_1 with coordinates in \mathbb{Q} and where \mathbf{f}_i is a tuple of length n_i with coordinates in the polynomial ring $\mathbb{Q}[x_1, \dots, x_{n_1 + \dots + n_{i-1}}]$, for $i = 2, \dots, k$.

The multivariate case: definition

Setting (recall)

$$\begin{pmatrix} x_1(n+1) \\ x_2(n+1) \\ \vdots \\ x_s(n+1) \end{pmatrix} = M \times \begin{pmatrix} x_1(n) \\ x_2(n) \\ \vdots \\ x_s(n) \end{pmatrix} + \begin{pmatrix} \mathbf{f}_{1n_1 \times 1} \\ \mathbf{f}_{2n_2 \times 1} \\ \vdots \\ \mathbf{f}_{kn_k \times 1} \end{pmatrix},$$

where \mathbf{f}_1 is a vector over \mathbb{Q} of length n_1 and where \mathbf{f}_i is a tuple of length n_i with coordinates in $\mathbb{Q}[x_1, \dots, x_{n_1+\dots+n_{i-1}}]$, for $i = 2, \dots, k$.

Definition

Then, the recurrence relation R is called P -solvable over \mathbb{Q} and the matrix M is called the **coefficient matrix** of R .

The notion of P -solvable recurrence is equivalent to that of *solvable mapping* in (E. Rodriguez-Carbonell & D. Kapur, ISSAC04) or that of *solvable loop* (L. Kovacs TACAS08) in the respective contexts.

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: ProgramAnalysis

Degree estimates for solutions of P -solvable recurrences: theorem

Assume M is in a Jordan normal form. Assume the eigenvalues $\lambda_1, \dots, \lambda_s$ of M (counted with multiplicities) are different from 0, 1, with λ_i being the i -th diagonal element of M . Assume for each block j the total degree of any polynomial in \mathbf{f}_j (for $i = 2 \dots k$) is upper bounded by d_j . For each i , we denote by $b(i)$ the block number of the index i , that is,

$$\sum_{j=1}^{b(i)-1} n_j < i \leq \sum_{j=1}^{b(i)} n_j.$$

Let $D_1 := n_1$ and for all $j \in \{2, \dots, k\}$ let $D_j := d_j D_{j-1} + n_j$. Then, there exists a solution (y_1, y_2, \dots, y_s) for R of the following form:

$$y_i := c_i \lambda_i^n + g_i, \quad i = 1 \dots s \quad \text{where}$$

- (a) c_i is a constant depending only on the initial value of the recurrence;
- (b) g_i is a poly-geometrical expression P in n w.r.t. $\lambda_1, \dots, \lambda_{i-1}$, such that

$$\deg(g_i) \leq D_{b(i)}.$$

Degree estimates for solutions of P -solvable recurrences: example

Consider the recurrence:

$$\begin{pmatrix} x(n+1) \\ y(n+1) \\ z(n+1) \end{pmatrix} := \begin{pmatrix} 2 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{pmatrix} \times \begin{pmatrix} x(n) \\ y(n) \\ z(n) \end{pmatrix} + \begin{pmatrix} 0 \\ x(n)^2 \\ x(n)^3 \end{pmatrix}$$

Viewing the recurrence as two blocks (x) and (y, z) , the degree upper bounds are

$$D_1 := n_1 = 1 \quad \text{and} \quad D_2 := d_2 D_1 + n_2 = 3 \times 1 + 2.$$

If we decouple the (y, z) block to the following two recurrences

$$y(n+1) = 3y(n) + x(n)^2 \quad \text{and} \quad z(n+1) = 3z(n) + x(n)^3,$$

then we deduce that the degree of the poly-geometrical expression for y and z are upper bounded by 2 and 3 respectively.

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: `ProgramAnalysis`

Degree estimates for the invariant ideal: theorem

- Let R be a P -solvable recurrence relation with variables (x_1, x_2, \dots, x_s) .
- Suppose R has a k -block configuration as $(n_1, 1), \dots, (n_k, d_k)$.
- Let $D_1 := n_1$; and for all $j \in \{2, \dots, k\}$, let $D_j := d_j D_{j-1} + n_j$.
- Let $A = \lambda_1, \lambda_2, \dots, \lambda_s$ be the eigenvalues (counted with multiplicities) of the coefficient matrix of R .
- Let \mathcal{M} be the multiplicative relation ideal of A associated with variables y_1, \dots, y_k . Let $r := \dim(\mathcal{M})$.
- Let $\mathcal{I} \subset \mathbb{Q}[x_1, x_2, \dots, x_s]$ be the invariant ideal of R .

Then, we have

$$\deg(\mathcal{I}) \leq \deg(\mathcal{M}) D_k^{r+1}.$$

Degree estimates for the invariant ideal: example

Consider again solving for (x, y) as a 2-variable recurrence

$$x(n+1) = y(n), y(n+1) = x(n) + y(n), \text{ with } x(0) = 0, y(0) = 1.$$

Recall that we obtained

$$\begin{aligned} x(n) &= \frac{\left(\frac{\sqrt{5}+1}{2}\right)^n}{\sqrt{5}} - \frac{\left(\frac{-\sqrt{5}+1}{2}\right)^n}{\sqrt{5}}, \\ y(n) &= \frac{\sqrt{5}+1}{2} \frac{\left(\frac{\sqrt{5}+1}{2}\right)^n}{\sqrt{5}} - \frac{-\sqrt{5}+1}{2} \frac{\left(\frac{-\sqrt{5}+1}{2}\right)^n}{\sqrt{5}}. \end{aligned}$$

Observe that $A := \frac{-\sqrt{5}+1}{2}, \frac{\sqrt{5}+1}{2}$ is weakly multiplicatively independent. The multiplicative relation ideal of A associated with variables u, v is generated by $u^2v^2 - 1$ and thus has degree 4 and dimension 1 in $\mathbb{Q}[u, v]$. Therefore, the previous theorem implies that the degree of invariant ideal bounded by 4×1^1 . This is sharp since this ideal is

$$\langle 1 - y^4 + 2xy^3 + x^2y^2 - 2x^3y - x^4 \rangle.$$

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: `ProgramAnalysis`

Dimension estimates for the invariant ideal: theorem

Theorem

Using the same notations as in the definition of P -solvable recurrences.

- Let $\lambda_1, \lambda_2, \dots, \lambda_s$ be the eigenvalues of M counted with multiplicities.
- Let \mathcal{M} be the multiplicative relation ideal of $\lambda_1, \lambda_2, \dots, \lambda_s$.
- Let r be the dimension of \mathcal{M} . Let \mathcal{I} be the invariant ideal of R .

Then, we have

$$\dim(\mathcal{I}) \leq r + 1.$$

Moreover, for generic initial values,

- ① we have $r \leq \dim(\mathcal{I})$,
- ② if 0 is not an eigenvalue of M and $\lambda_1, \lambda_2, \dots, \lambda_s$ is weakly multiplicatively independent, then we have $r = \dim(\mathcal{I})$.

Corollaries

- ① If $r + 1 < s$ holds, then \mathcal{I} is not the zero ideal in $\mathbb{Q}[x_1, x_2, \dots, x_s]$.
- ② Assume that $x_1(0) := a_1, \dots, x_s(0) := a_s$ are independent indeterminates. If the eigenvalues of R are multiplicatively independent, then the inductive invariant ideal of the loop is the zero ideal in $\mathbb{Q}[a_1, \dots, a_s, x_1, x_2, \dots, x_s]$.

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: ProgramAnalysis

Loop model under study: recall

```
while  $C_0$  do  
  if  $C_1$   
    then  
       $X := A_1(X)$ ;  
    elif  $C_2$   
      then  
         $X := A_2(X)$ ;  
    ...  
    elif  $C_m$   
      then  
         $X := A_m(X)$ ;  
      end if  
  end while
```

- 1 Loop variables: $X = x_1, \dots, x_s$,
rational value scalar
- 2 Conditions: each C_i is a quantifier free
formula in X over \mathbb{Q} .
- 3 Assignments: $A_i \in \mathbb{Q}[X]$ inducing a
polynomial map $M_i : \mathbb{R}^s \mapsto \mathbb{R}^s$
- 4 Initial condition: X -values defined by a
semi-algebraic system.

A direct approach

Input

- (i) $M := m_1, m_2, \dots, m_c$ is a sequence of monomials in the loop variables X ,
- (ii) $S := s_1, s_2, \dots, s_r$ is a set of r points on the inductive trajectory of the loop,
- (iii) E is a polynomial system defining the loop initial values,
- (iv) B is the transitions $(C_1, A_1), \dots, (C_m, A_m)$ of the loop.

Algorithm

- 1 $L := \text{BuildLinSys}(M, S)$
- 2 $N := \text{LinSolve}(L)$ is full row rank and generates the null space of L .
- 3 $F := \emptyset$
- 4 For each row vector $\mathbf{v} \in N$ do
 - $F := F \cup \{\text{GenPoly}(M, \mathbf{v})\}$
- 6 If $Z(E) \not\subseteq Z(F)$ then return FAIL
- 6 For each branch $(C_i, A_i) \in B$ do
 - if $A_i(Z(F) \cap Z(C_i)) \not\subseteq Z(F)$ then return FAIL
- 7 Return F , a list of polynomial equation invariants for the target loop.

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: ProgramAnalysis

A small-prime approach: algorithm

Algorithm

- ① $p := \text{MaxMachinePrime}(); L_p := \text{BuildLinSysModp}(M, S, p);$
- ② $N_p := \text{LinSolveModp}(L_p, \mathbf{p})$
- ③ $d := \dim(N_p); \mathbf{N} := (N_p); \mathbf{P} := (p);$
- ④ While $p > 2$ do
 - ① If $d = 0$ then return FAIL
 - ② $N := \text{RatRecon}(\mathbf{N}, \mathbf{P})$
 - ③ If $N \neq \text{FAIL}$ then break;
 - ④ $p := \text{PrevPrime}(p); L_p := \text{BuildLinSysModp}(M, S, p);$
 $N_p := \text{LinSolveModp}(L_p, \mathbf{p})$
 - ⑤ If $d > \dim(N_p)$ then $d := \dim(N_p); \mathbf{N} := (N_p); \mathbf{P} := (p)$
 - ⑥ else $\mathbf{N} := \text{Append}(\mathbf{N}, N_p); \mathbf{P} := \text{Append}(\mathbf{P}, p)$
- ⑤ If $p = 2$ then return FAIL
- ⑥ $F := \emptyset$
- ⑦ For each row vector $\mathbf{v} \in N$ do

$$F := F \cup \{\text{GenPoly}(M, \mathbf{v})\}$$
- ⑧ If $Z(E) \not\subseteq Z(F)$ then return FAIL
- ⑨ For each branch $(C_i, A_i) \in B$ do

$$\text{if } A_i(Z(F) \cap Z(C_i)) \not\subseteq Z(F) \text{ then return FAIL}$$
- ⑩ Return F , a list of polynomial equation invariants for the target loop.

A small-prime approach: complexity result

Proposition

Both algorithms run in singly exponential time w.r.t. number of loop variables.

Indeed

- the number of monomials of M is singly exponential w.r.t. number of loop variables.
- applying our criterion to certify the result can be reduced to an ideal membership problem, which is singly exponential w.r.t. number of loop variables.

A small-prime approach: example

Consider the following recurrence relation on (x, y, z) :

$$\begin{pmatrix} x(n+1) \\ y(n+1) \\ z(n+1) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & -3 \\ 0 & 1 & 3 \end{pmatrix} \begin{pmatrix} x(n) \\ y(n) \\ z(n) \end{pmatrix}$$

with initial value $(x(0), y(0), z(0)) = (1, 2, 3)$.

- Note that the characteristic polynomial of the coefficient matrix has 1 as a triple root and the mult. rel. ideal of the eigenvalues is 0-dimensional.
- So the invariant ideal of this recurrence has dimension either 0 or 1.
- On the other hand, we can show that for all $k \in \mathbb{N}$, we have $M^k \neq M$; so there are infinitely many points in the set $\{(x(k), y(k), z(k)) \mid k \in \mathbb{N}\}$, whenever $(x(0), y(0), z(0)) \neq (0, 0, 0)$.
- With our method, we compute the following invariant polynomials

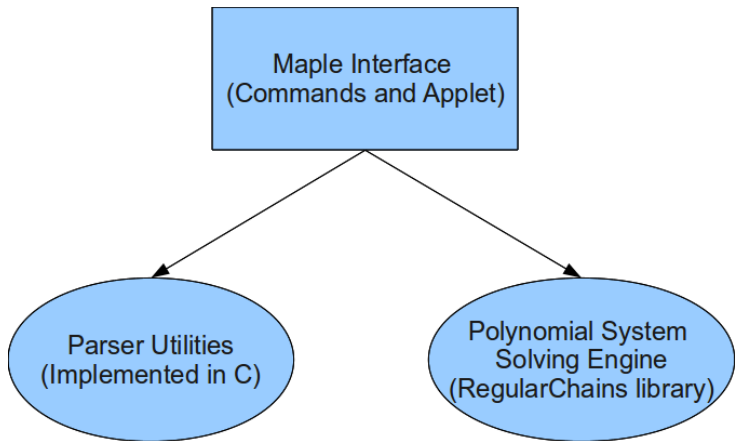
$$x + y + z - 6, y^2 + 4yz + 4z^2 - 6y - 24z + 20,$$

which generate a prime ideal of dimension 1, thus the invariant ideal of this recurrence.

Plan

- 1 Preliminaries
 - Notions on loop invariants
 - Poly-geometric summations
- 2 Invariant ideal of P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - P -solvable recurrences
 - Degree estimates for solutions of P -solvable recurrences
 - Degree estimates for their invariant ideal
 - Dimension estimates for their invariant ideal
- 3 Loop invariant generation via polynomial interpolation
 - A direct approach
 - A modular method
 - Maple Package: ProgramAnalysis

ProgramAnalysis: package architecture



Maple session: the input program in a file

```
wensley2 := proc(P, Q, E)
local a, b, d, y,
  a := 0;
  b := 1/2*Q;
  d := 1;
  y := 0;
  #PRE:  $Q > P$  and  $P \geq 0$  and  $E > 0$ 
  while  $E \leq d$  do
    if  $P < a + b$  then
      b := 1/2*b;
      d := 1/2*d
    else
      a := a + b;
      y := y + 1/2*d;
      b := 1/2*b;
      d := 1/2*d
    end if
  end do;
  #POST:  $P/Q \geq y$  and  $y > P/Q - E$ 
  return y
end proc
```

Maple session: the sample points

$$\begin{aligned}
 & \left[\left[0, \frac{5}{2}, 1, 0 \right], \left[\frac{5}{2}, \frac{5}{4}, \frac{1}{2}, \frac{1}{2} \right], \left[\frac{5}{2}, \frac{5}{8}, \frac{1}{4}, \frac{1}{2} \right], \left[\frac{5}{2}, \frac{5}{16}, \frac{1}{8}, \frac{1}{2} \right], \left[\frac{45}{16}, \frac{5}{32}, \frac{1}{16}, \frac{9}{16} \right], \left[\frac{95}{32}, \frac{5}{64}, \right. \right. \\
 & \left. \left. \frac{1}{32}, \frac{19}{32} \right], \left[\frac{95}{32}, \frac{5}{128}, \frac{1}{64}, \frac{19}{32} \right], \left[\frac{95}{32}, \frac{5}{256}, \frac{1}{128}, \frac{19}{32} \right], \left[\frac{765}{256}, \frac{5}{512}, \frac{1}{256}, \frac{153}{256} \right], \left[\frac{1535}{512}, \right. \right. \\
 & \left. \left. \frac{5}{1024}, \frac{1}{512}, \frac{307}{512} \right], \left[\frac{1535}{512}, \frac{5}{2048}, \frac{1}{1024}, \frac{307}{512} \right], \left[\frac{1535}{512}, \frac{5}{4096}, \frac{1}{2048}, \frac{307}{512} \right], \left[\frac{12285}{4096}, \frac{5}{8192}, \right. \right. \\
 & \left. \left. \frac{1}{4096}, \frac{2457}{4096} \right], \left[\frac{24575}{8192}, \frac{5}{16384}, \frac{1}{8192}, \frac{4915}{8192} \right], \left[\frac{24575}{8192}, \frac{5}{32768}, \frac{1}{16384}, \frac{4915}{8192} \right], \left[\frac{24575}{8192}, \right. \right. \\
 & \left. \left. \frac{5}{65536}, \frac{1}{32768}, \frac{4915}{8192} \right], \left[\frac{196605}{65536}, \frac{5}{131072}, \frac{1}{65536}, \frac{39321}{65536} \right], \left[\frac{393215}{131072}, \frac{5}{262144}, \frac{1}{131072}, \right. \right. \\
 & \left. \left. \frac{78643}{131072} \right], \left[\frac{393215}{131072}, \frac{5}{524288}, \frac{1}{262144}, \frac{78643}{131072} \right], \left[\frac{393215}{131072}, \frac{5}{1048576}, \frac{1}{524288}, \frac{78643}{131072} \right], \right. \\
 & \left. \left[\frac{3145725}{1048576}, \frac{5}{2097152}, \frac{1}{1048576}, \frac{629145}{1048576} \right], \left[\frac{6291455}{2097152}, \frac{5}{4194304}, \frac{1}{2097152}, \frac{1258291}{2097152} \right] \right]
 \end{aligned}$$

Maple session: verifying the program

```
> mplfile := cat(getenv("MXHOME"), "/mx-2012/programs/wensley2.mpl") :
precond := [[Q>P, P>=0, E>0]];
postcond := [[P >= Q*y , Q*y > P - Q*E  ]];
guard := [[E<=d]];
ineq_invs := [ P - Q*d < Q*y, Q*y <= P, y>=0];
```

$$\begin{aligned}
 \text{precond} &:= [[P < Q, 0 \leq P, 0 < E]] \\
 \text{postcond} &:= [[Qy \leq P, P - Q E < Qy]] \\
 \text{guard} &:= [[E \leq d]] \\
 \text{ineq_invs} &:= [-dQ + P < Qy, Qy \leq P, 0 \leq y]
 \end{aligned}
 \tag{2.3.1}$$

```
> st := time():
eq_invs := LoopEqInv(mplfile); # compute equation invariants
time()-st;
```

$$\begin{aligned}
 \text{eq_invs} &:= [yQ - a, dQ - 2b, -2by + ad] \\
 &0.210
 \end{aligned}
 \tag{2.3.2}$$

```
> # verify the specification of the program
st:=time():
LoopVerify(precond, guard, [[op(eq_invs), op(ineq_invs)]], postcond);
time()-st;
```

$$\begin{aligned}
 &\text{true} \\
 &1.380
 \end{aligned}
 \tag{2.3.3}$$

Xie Xie!