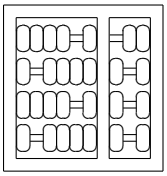

INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

Endomorphism for Non-Trivial Semi-Algebraic Loop Invariant Generation

Nadir Matringe *Arnaldo Vieira Moura*
Rachid Rebiha

Technical Report - IC-08-08-31 - Relatório Técnico

November - 2008 - Novembro

Endomorphism for Non-Trivial Semi-Algebraic Loop Invariant Generation

Nadir Matringe* Arnaldo Vieira Moura† Rachid Rebiha‡ § ¶

November 4, 2008

Abstract

Non-linear loop invariant generation have seen tremendous progress in recent years. However, the weakness of these approach is that they are limited to linear (affine) system, and they often relay on *trivial* polynomial invariant (null or constant). Moreover, for programs with loops that describe multivariate polynomial or multivariate fractional system, no method is known to lend itself to *non-trivial* non-linear invariant. In order to automate the generation of *non-trivial* multivariate polynomial invariant, one needs to handle *initiation* and *consecution* condition for non-linear (algebraic) loop. We demonstrate a powerful computational *complete* method that encodes these conditions for a candidate invariant (a multivariate polynomial assertion with indeterminate coefficients) into a set of *multi-parametric* constraints such that all solutions identify a *non-trivial* non-linear loop invariant. Then, we provide a *complete* decision procedure for this constraint-solving problem. For each type of loop (affine, polynomial, fractional system) we present necessary and sufficient conditions for the existence of *non-trivial* non-linear loop invariant and we identify a large decidable class together with undecidable class. Without computing Grobner bases or using quantifier elimination techniques we show that our method generates stronger invariant, hereby circumventing difficulties met by recent approach.

1 Introduction

In this paper, we present a new method that addresses the various deficiencies of the-state-of-the-art non-linear invariant generation method. An invariant at a location of a program is an assertion true of any reachable program state associated to this location. We provides mathematical techniques and design efficient algorithms to automate the discovery and

*Institue de Mathematiques de Jussieu (UMR 7586) Universit Paris 7-Denis Diderot, France.

†Institute of Computing, University of Campinas, 13081-970 Campinas, SP. Research supported in part by CNPq — Conselho Nacional de Desenvolvimento Científico e Tecnológico, grant #472504/2007-0

‡Institute of Computing, University of Campinas, 13081-970 Campinas, SP. Research supported in part by CNPq — Conselho Nacional de Desenvolvimento Científico e Tecnológico, grant #142170/2007-0

§Faculty of Informatics, University of Lugano, Switzerland

¶List of Authors in Alphabetic Order.

strengthening of non-linear interrelationships among the variables of a program containing non-linear loop (multivariate polynomial and fractional manipulation).

It is well-know that the automation and effectiveness of formal verification of program depend to the ease with which strong invariants can be automatically generated. (e.g. safety properties can be reduced to invariants properties). Furthermore, the standard technique use invariant assertion [16] directly to prove program properties or to provide lemmas to established other safety and liveness program properties. We also know that the weakest precondition method [8, 10] require loop invariant to be completely automatic if the considered program contains a loop.

To generate loop invariant, one need to discover *inductive* assertion that holds at any steps of the loop. Moreover, *inductive* assertion holds at the first time the loop location is reached (i.e. initiation condition), and is preserved under every instructions cycle back to the loop location (i.e. consecution condition). All invariant generation methods are base on inductive assertion discovery. In the case of loop describing a linear system, *Farka's lemma* [23] are used to encode the conditions for being a *linear* invariant. On the other hand, for *non-linear* invariant the difficulty for the automatic generation arise from the lack of *decision procedure* based on *completness*, *existence* and *decidability* results.

Non-linear loop invariant generation methods have seen tremendous progress [22, 1, 19, 21, 3, 13, 5, 20] in recent years. But these approaches are limited to linear (affine) system or relay on non scalable methods (with complexity at least doubly exponential). Moreover, they require Grobner Bases computation , first-order quantifier elimination [24],[4] or cylindrical algebraic decomposition. In [22], the non-linear invariant generation problem is reduced to a numerical constraint solving problem over indeterminate polynomial coefficients. In [19] similar forward propagation techniques use an abstract interpretation [7],[6] framework and Grobner bases construction to compute invariants as fixed points of operations on ideals. In [17], techniques from abstract interpretation and Grobner bases computation are used to calculate a polynomial ideal that represent the weakest precondition for the validity of the polynomial relations at a given program point. The main challenge for these techniques is that abstract interpretation introduce imprecision (*widening*) to assure termination. This is the main reason why these approaches often produce null or trivial invariant due to a too coarse abstraction. In [14],[13], the methods use techniques from algebra and combinatorics (Grobner bases, variable elimination, algebraic dependencies and symbolic summation). They attempt to generate (not in an completely automatic way) all polynomial invariants from a restricted class of linear (P-solvable) loops. Also conditional statement are "omitted"[13]. On the other hand, [22],[12],[20] use the theory of polynomial algebra to generate multivariate polynomial invariants. However, the main weakness of these approaches is that they relay on methods with at least a doubly exponential complexity: Grobner Bases computation , first-order quantifier elimination. Also, these approaches omit conditional statements and deal with conjunction of linear equations [22] and polynomial equations [12], [20]. Recently, [3] proposed the first methods handling conditional statement by directly solving semi-algebraic systems, but the authors of [3] pointed out that this is not a practical method since the complexity remind at least doubly exponential (single exponential w.r.t the number of program variables and parameters, and doubly exponential w.r.t the number of parameters).

For each type of semi-algebraic loop (affine, multivariate fractional and polynomial) we can summarize our contribution as follos:

1. In contrast with the mentioned approaches, our methods do not required computation of Grobner bases, quantifier elimination, cylindrical algebraic decomposition or direct resolution of semi-algebraic system and do not depend on any abstraction methods.

2. We succeeded in reducing the non-linear loop invariant generation problem to the intersection between eigenspaces of specific endomorphisms and initial linear or semi-affine/algebraic constraints.

3. We present the first non-linear invariant generation methods that handle multivariate fractional instructions (as far as it is our knowledge it is the first methods that handle multivariate fractional system), conditional statement, and nested inner-loops.

4. We present necessary and sufficient conditions for the existence of *non-trivial* non-linear loop invariants.

5. Considering the problem of invariant generation, we identify large decidable and undecidable classes.

Considering affine (linear), multivariate polynomial and multivariate fractional (Algebraic) loop, we provide (i) theorems for the existence of *non-trivial* non-linear invariant, (ii) theorems that identify Large decidable class and undecidable class and (iii) the first complete decision procedure for the induced constraint solving problem. Then, we obtain the first methods capable to generate *non-trivial* non-linear invariant for multivariate fractional (or polynomial) loop.

Computational and algorithmic algebra theory are recently used for invariant generation [22, 19, 14, 11, 17] by applying a method build upon Grobner Bases computation. In [22] non-linear invariant generation problem is reduce to numerical constraint solving problem over indeterminate polynomial coefficients. In [19] a similar forward propagation techniques use an abstract interpretation [7] framework and Grobner bases construction to compute invariants as fixed points and operations on ideals. In [17], techniques from abstract interpretation and Grobner bases computation are used to calculate a polynomial ideal that represent the weakest precondition for the validity of the considered polynomial relations at a given program point. In [14], the algorithm uses techniques from algebra and combinatorics (Grobner bases, variable elimination, algebraic dependencies and symbolic summation). They attempt to generate (not in an completely automatic way) all polynomial invariant from a restricted class of linear loops.

However, the weakness of these approach is that they are limited to linear (affine) system, and they often rely on *trivial* polynomial invariant (null or constant). As far as our knowledge, we provide the first methods for the generation of invariant for the general case of loop describing multivariate fractional system.

In Section 2 we present ideals of polynomials and their possible interaction with inductive assertion. In Section 3 we consider the case where the loop is linear. We present results for the existence of *non-trivial* invariant, decidable and undecidable class. Translated the

problem in term of linear algebra, we present a complete decision procedure for the automatic generation of *non-trivial* non-linear invariants. In Section 4 We present results for the existence of *non-trivial* invariant, decidable and undecidable class. Considering non-linear loop in section 4, we then present results for the existence of *non-trivial* invariant, decidable and undecidable class and a generalization of the decision procedure. In section 5 we provide a complete generalization by considering loop describing multivariate fractional system. We finally conclude our approach in Section 6.

2 Ideals of Polynomials and Inductive Assertion

2.1 Ideals of Polynomials

Let $A_n = K[X_1, .., X_n]$ be the ring of multivariate polynomials over the set of variables $\{X_1, .., X_n\} \subset K$. An ideal $I \in A_n$ is closed under addition, it include 0 and is closed by multiplication with each element in A_n (for all $P \in A_n$ and $Q \in I$, $PQ \in I$). Let $E \subseteq A_n$ a set of polynomials, the *ideal generated* by E is given by the following set of *finite* sums: $(E) = \{\sum_{i=1}^{k} P_i Q_i \mid P_i \in K[X_1, \ldots, X_n], Q_i \in E, k \geq 1\}$. In other words, the set of *finite* sums $\sum_{Q \in E} A_n Q$. A set of polynomials E is said to be a *basis* of an ideal I if $I = (E)$. By Hilbert's basis theorem, we know that for all Ideal, there exists a *finite basis*. An *algebraic* assertion is an assertion $\phi(x_1, ..., x_n)$ of the following form $\bigwedge_i p_i(x_1, ..., x_n) = 0$ where each $p_i \in A_n$. Let $\phi(x_1, ..., x_n) \equiv (\bigwedge_i p_i(x_1, ..., x_n) = 0)$ an algebraic assertion and the associated set $S_\phi \subseteq A_n$ of polynomials p_i that appear in ϕ. For S_ϕ we defined the *algebraic set* (or *variety*) as the common zeroes of all the polynomials in S_ϕ by: $V(S_\phi) = \{(x_1, ..., x_n) \in \mathbb{C}^n \mid \forall p \in S_\phi, p(x_1, ..., x_n) = 0\}$. Consider a polynomial $Q \in A_n$, an algebraic assertion ϕ with its associated polynomial set S_ϕ and the Ideal $I = (S_\phi)$, the *Ideal membership* problem ($Q \in I$) can be interpreted by the equivalent inclusion problem $V(I) \subseteq V(Q)$.

Theorem 1. *(Weak version of Hilbert's Nullstellensatz) In the context described just above, the Hilbert's Nullstellensatz theorem states that if* $(Q \in I)$ *then* $\phi(x_1, \ldots, x_n) \models (Q(x_1, \ldots, x_n) = 0)$.

Constraint-based approaches and their related work: If we consider a n multivariate polynomial, $Q = \sum_{i_1,..,i_n} a_{i_1,..,i_n} {x_1}^{i_1} ... {x_n}^{i_n}$, where $a_{i_1,...,i_n}$ are in a field K. How do we know if p is in an ideal I of $K[X_1, ..., X_n]$ (*Ideal membership* problem)? One can consider the Grobner Bases $G = \{g_1, ..., g_s\}$ of I (There exist an algorithm [2, 9] that compute such bases as long as we know a finite generating bases of I). We can compute the normal form of Q for I using its Grobner G, denoted $NF_G(Q)$ (in dimension 1 it is the rest of the euclidean division). Grobner Bases can be used because they guarantee the confluence and termination of those reductions. Then $NF(Q) = \sum_{i_1,..,i_n} f(a)_{i_1,..,i_n} {x_1}^{i_1} ... {x_n}^{i_n}$, here $f(a)_{i_1,..,i_n}$ are combination of $a_{i_1,..,i_n}$. Then $(Q \in I)$ is equivalent to $(NF(Q) = 0)$, in other words all coefficients $f(a)_{i_1,...,i_n} = 0$.

However the Grobner Bases and normal forms computation remains doubly exponential. Moreover the constraint systems generated are often non-linear for linear loops and always non-linear for algebraic loops which restrict direct resolution (using quantifier elimination

or SMT solver) as non-scalabe appraoches. We succeed to generate non-trivial non-linear invariant equivalent set of constraint without using Grobner bases and without normal form reduction computation or direct consideration of such constraint systems.

2.2 Inductive Assertion and Invariants

We use transition system as representation of programs.

Definition 1. *A transition system $\langle V, L, \mathcal{T}, l_0, \Theta\rangle$, where V is a set of variables, L is a set of locations and l_0 is the initial location. A state is given by an interpretation of the variables in V. A transition $\tau \in \mathcal{T}$ is given by a tuple $\langle l_{pre}, l_{post}, \rho_\tau\rangle$, with l_{pre} and l_{post} to name the τ's pre- and post- locations. The transition relation ρ_τ is a first-order assertion over $V \cup V'$, where V correspond to current-state variable and V' to the next-state variables. Θ is the initial condition given as a first-order assertion over V.*

Definition 2. *Consider a transition system $W = \langle V, L, \mathcal{T}, l_0, \Theta\rangle$, an* invariant *at location $l \in L$ is defined by an assertion over V which holds on all reachable state at location l. An* invariant *of W is an assertion over V that holds at all locations.*

Definition 3. *Let $W = \langle V, L, \mathcal{T}, l_0, \Theta\rangle$ a transition system and a given domain of assertion D. An assertion map for W is a map $\eta : L \to D$. We said that η is* inductive *if and only if the* Initiation *and* Consecution *conditions hold:*

- $\Theta \models \eta(l_0)$ *(Initiation)*
- $\forall \tau \in \mathcal{T}$ *s.t* $\tau = \langle l_i, l_j, \rho_\tau\rangle$ *we have* $\eta(l_i) \wedge \rho_\tau \models \eta(l_j)'$ *(Consecution).*

From Floyd and Hoare, we know that if η is and inductive assertion map then $\eta(l)$ is an invariant at l. We will use the following notions of consecution.

Definition 4. *Let $\tau = \langle l_i, l_j, \rho_\tau$ be a transition for given algebraic transition system and η be an algebraic inductive map. We identify the following* complete *notion of consecution:*

1. *η satisfies* Fractional-*scale consecution for τ if and only if there exist a multivariate fractional $\frac{T}{Q}$ such that: $\rho_\tau \models (\eta(l_j)' - \frac{T}{Q}\eta(l_j) = 0)$*
2. *η satisfies* Polynomial-*scale consecution for τ if and only if there exist a multivariate polynomial T such that : $\rho_\tau \models (\eta(l_j)' - T\eta(l_j) = 0)$*
3. *η satisfies* Constant-*scale consecution for τ if and only if there exist a constant $\lambda \in K$ such that : $\rho_\tau \models (\eta(l_j)' - \lambda\eta(l_j) = 0)$*

Constant-scale [22] consecution encodes the fact that the numerical value of the assertion after the transition τ is a λ constant multiple of the numerical value prior the transitionτ. On the other hand, *Polynomial-scale* consecution encodes the fact that the numerical value of the assertion after the transition τ is a T multivariate polynomial multiple of the numerical value prior the transition. We are able to handle the most general case (the loop describes a multivariate fractional system) with *Fractional-scale* consecution. *Fractional-scale* consecution encodes the fact that the numerical value of the assertion after the transition τ is a $\frac{T}{Q}$ multivariate fractional multiple of the numerical value prior the transition.

3 When to Use Constant Scale Consecution

3.1 Decision Procedure for λ-Invariant

Definition 5. *We consider transition system corresponding to a* loop $\tau = \langle l_i, l_i, \rho_\tau \rangle$*, with:*

$$\rho_\tau = \begin{bmatrix} x'_1 = L_1(x_1, \ldots, x_n) \\ \vdots \\ x'_n = L_n(x_1, \ldots, x_n) \end{bmatrix} \tag{1}$$

A Polynomial $Q \in K[X_1, ..., X_n]$ *is said to be a* λ-invariant for constant-scale consecution *with parameter* λ *for the loop* τ *if and only if*

$$Q(X'_1, .., X'_n) - \lambda Q(X_1, .., X_n) = 0,$$

modulo the Ideal of $K[X'_1, .., X'_n, X_1, .., X_n]$ *corresponding to the loop, generated by the grobner base* $(X'_1 - L_1(X_1, ..., X_n), ..., X'_n - L_n(X_1, ..., X_n))$*.*

Theorem 2. ***(λ-invariant's characterization)*** *Consider a transition system corresponding to a loop* τ *described as in definition 5. Let* $Q(X_1, ..., X_n)$ *be a multivariate polynomial with indeterminate coefficients (a template).* Q *is a* λ*-invariant for constant-scale consecution with parameter* $\lambda \in K$ *for* τ *if and only if*

$$Q(L_1(X_1, \ldots, X_n), .., L_n(X_1, \ldots, X_n)) = \lambda Q(X_1, \ldots, X_n)$$

Proof. if $Q(X'_1, .., X'_n) - \lambda Q(X_1, .., X_n)$ belongs to the ideal I generated bey the family $(X'_1 - L_1, \ldots, X'_n - L_n)$, then there exists a family $(A_1, \ldots, A_n)$ of polynomials in $K[X'_1, .., X'_n, X_1, .., X_n]$ such that $Q(X'_1, .., X'_n) - \lambda Q(X_1, .., X_n) = (X'_1 - L_1)A_1 + \cdots + (X'_n - L_n)A_n$. Letting $X'_i = L_i$, we obtain that $Q(L_1(X_1, ..., X_n), .., L_n(X_1, ..., X_n)) = \lambda Q(X_1, ..., X_n)$.

Conversely suppose $Q(L_1(X_1, \ldots, X_n), .., L_n(X_1, \ldots, X_n)) = \lambda Q(X_1, \ldots, X_n)$, then as $Q(X'_1, .., X'_n)$ is equal to $Q(L_1, .., L_n)$ modulo the ideal I, we get that $Q(X'_1, .., X'_n) = \lambda Q(X_1, \ldots, X_n)$ modulo I. □

Consider the case of *affine* transition system corresponding to a loop described as in definition 5 where $L_i(x_1, ..., x_n) = \sum_{k=1}^{n} c_{i,k-1}x_k + c_{i,k}$ are affine forms. In this case let $Q \in A_n$ be a multivariate polynomial of degree r, with indeterminate coefficients (a template) which is going to be a λ-invariant candidate for constant-scale consecution with parameter λ. We are going to show that for good choices of λ, there always exists such a λ-invariant that won't be trivial. As Q is of degree r, it has coefficients listed $a_0, .., a_t$ corresponding each to a monomial in the expansion of Q with respect to total-degree lexicographic ordering ($t+1$ being the number of monomials of degree inferior to r, so that a_t is the coefficient of the constant term). Let's notice that $Q(L_1(X_1, .., X_n), .., L_n(X_1, .., X_n)$ is also of degree r because all L_i's are of degree one. Translating the problem in terms of linear algebra, using the canonical basis of $K[X_1, .., X_n]$, we immediately see that this is equivalent to resolve the following:

$$(M - \lambda I)\ \vec{a} = \vec{0}$$

where $\vec{a}$ is the column vector with coordinates a_i, and M a $(t+1) \times (t+1)$ matrix whose coefficients depend on the $c_{i,k}$'s. To be more precise, let V_r be the subspace of $K[X_1, .., X_n]$ consisting of polynomials of degree less than r, then M is the matrix of the endomorphism of V_r given by

$$(P(X_1, .., X_n) \mapsto P(L_1(X_1, ..., X_n), .., L_n(X_1, ..., X_n))$$

in the canonical basis of V_r consisting of the terms of degree less then r with total-degree lexicographic ordering (this is in fact an endomorphism because all L_i's are of degree one). In other words, the parameter λ must be an eigenvalue of M if we want to find a non null λ-invariant whose coefficients will be those of an eigenvector.

Theorem 3. ***(Existence of λ-invariant)*** *Considering V_r and M as described just above, A polynomial Q of V_r is λ-invariant for scale consecution if and only if there exists an eigenvalue λ of M, such that Q belongs to the eigenspace corresponding to λ.*

We also notice that the last column of M is always $(0, \cdots, 0, 1)^{\mathcal{T}}$ by definition of the matrix M. Thus 1 is always an eigenvalue of M, a corresponding eigenvector being $\vec{a}$ with $a_i = 0$ except for a_t , this gives the trivial λ-invariant $Q(X_1, .., X_n) = a_t$, i.e. constant polynomial. Eigenvalue one always gives the constant polynomial as a λ-invariant, but might give better invariants for different eigenvectors if $dim(Ker(M - \lambda I)) \geq 2$, as we will see in the following examples.

3.2 Examples

Example 1. ***General Case for 2 Variables*** *We first treat the general case where the transition system has only two variables, we will look for a λ-invariant candidate of degree two.*

$$\rho_\tau = \begin{bmatrix} x_1' = c_{1,0}x_1 + c_{1,1}x_2 + c_{1,2} \\ x_2' = c_{2,0}x_1 + c_{2,1}x_2 + c_{2,2} \end{bmatrix}$$

And we look for an invariant polynomial $Q(X_1, X_2) = a_0{X_1}^2 + a_1X_1X_2 + a_2{X_2}^2 + a_3X_1 + a_4X_2 + a_5$ for constant scaling with parameter λ. We remind that we must solve the equation $Q(c_{1,0}X_1 + c_{1,1}X_2 + c_{1,2}, c_{2,0}X_1 + c_{2,1}X_2 + c_{2,2}) = \lambda Q(X_1, X_2)$. Thus for M we get the following matrix:

$$\begin{pmatrix} {c_{1,0}}^2 & c_{1,0}c_{2,0} & {c_{2,0}}^2 & 0 & 0 & 0 \\ 2c_{1,0}c_{1,1} & c_{1,0}c_{2,1} + c_{1,1}c_{2,0} & 2c_{2,0}c_{2,1} & 0 & 0 & 0 \\ {c_{1,1}}^2 & c_{1,1}c_{2,1} & {c_{2,1}}^2 & 0 & 0 & 0 \\ 2c_{1,0}c_{1,2} & c_{1,0}c_{2,2} + c_{1,2}c_{2,0} & 2c_{2,0}c_{2,2} & c_{1,0} & c_{2,0} & 0 \\ 2c_{1,1}c_{1,2} & c_{1,1}c_{2,2} + c_{1,2}c_{2,1} & 2c_{2,1}c_{2,2} & c_{1,1} & c_{2,1} & 0 \\ {c_{1,2}}^2 & c_{1,2}c_{2,2} & {c_{2,2}}^2 & c_{1,2} & c_{2,2} & 1 \end{pmatrix}$$

We see that the last column is as predicted, plus the matrix is block diagonal, thus its characteristic polynomial is $P(\lambda) = (1 - \lambda)P_1(\lambda)P_2(\lambda)$, with P_1 being the characteristic polynomial of

$$\begin{pmatrix} c_{1,0} & c_{2,0} \\ c_{1,1} & c_{2,1} \end{pmatrix}$$

and P_2 being the one of

$$\begin{pmatrix} {c_{1,0}}^2 & c_{1,0}c_{2,0} & {c_{2,0}}^2 \\ 2c_{1,0}c_{1,1} & c_{1,0}c_{2,1}+c_{1,1}c_{2,0} & 2c_{2,0}c_{2,1} \\ {c_{1,1}}^2 & c_{1,1}c_{2,1} & {c_{2,1}}^2 \end{pmatrix}$$

. Here P_2 is of degree three and has at least one real root, which can be calculated by Lagrange's resolvent method. So if we choose the parameter λ to be this root, the corresponding eigenvectors will give non-trivial λ-invariants of degree two as at least a_0, a_1 or a_2 must be non null for such an eigenvector.

Example 2. *To fix things, suppose the transition system is given by $\tau = \langle l_i, l_i, \rho_\tau \rangle$ with $\rho_\tau \equiv [x'_1 = 2x_1 + x_2 + 1 \wedge x'_2 = 3x_2 + 4]$ M will be equal to:*

$$\begin{pmatrix} 4 & 0 & 0 & 0 & 0 & 0 \\ 4 & 6 & 0 & 0 & 0 & 0 \\ 1 & 3 & 9 & 0 & 0 & 0 \\ 4 & 8 & 0 & 2 & 0 & 0 \\ 2 & 7 & 24 & 1 & 3 & 0 \\ 1 & 4 & 16 & 1 & 4 & 1 \end{pmatrix}$$

Then $P_2(\lambda) = (4-\lambda)(6-\lambda)(9-\lambda)$, so fix λ to be 4, we get that the corresponding eigenspace is generated by the following vector: $(1, -2, 1, -6, 6, 9)^T$. So that as a λ-invariant polynomial for scale consecution with parameter 4, we get $1{X_1}^2 - 1X_1X_2 + {X_2}^2 - 6X_1 + 6X_2 + 9$.

Example 3. ***With 4 Variables***

We study the following transition system [22] corresponding to the multiplication of 2 numbers and the transition considered is $\tau = \langle l_i, l_i, \rho_\tau \rangle$ with $\rho_\tau \equiv [s' = s+i \wedge j' = j+1 \wedge i' = i \wedge j'_0 = j_0]$.

Here to keep the same notations as in [22], we let s stand for X_1, j for X_2, i for X_3 and j_0 for X_4). We are looking for a degree two invariant of the form $Q(s, j, i, j_0) = a_0{s}^2 + a_1sj + a_2si + a_3sj_0 + a_4{j}^2 + a_5ji + a_6jj_0 + a_7{i}^2 + a_8ij_0 + a_9{j_0}^2 + a_{10}s + a_{11}j + a_12i + a_{13}j_0 + a_{14}$. We want to resolve $Q(s+i, j+1, i, j_0) = \lambda Q(s, j, i, j_0)$. Here an evident eigen value is 1, it is clear in view of the matrix M:

$$\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & -1 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1
\end{pmatrix}$$

that $dim(Ker(M-I)) \geq 2$*, for example the vector* $(1,0,0,0,0,1,0,0,-1,0,0,0,0,0,0)^{\mathcal{T}}$ *is the eigenvector corresponding to the* λ*-invariant* $s + ji - ij_0$ *(without grobner bases and quantifier elimination we find the invariant obtained in [22]).*

3.3 What Happens in Practice

We have seen that for an affine transition system of the type of the one describe in definition 5 where $L_i(x_1, ..., x_n) = \sum_{k=1}^{n} c_{i,k-1}x_k + c_{i,k}$ are affine forms, the scaling consecution technique with parameter λ works if and only if λ is an eigenvalue of M. Eigenvalues are calculated as the roots of the characteristic polynomial of M, we thus state 3 facts.

1. 1 is always an eigenvalue as we have seen, but if the corresponding eigenspace is of dimension one exactly, the eigen vectors corresponds to constant λ-invariants, which are trivial.

2. Apart from 1, other eigenvalues might not be real, but complex.

3. From Galois theory, we know that as soon as a polynomial is of degree equal or greater than five, one can generally not calculate his roots, so even if there are eigenvalues different than 1, one is not always sure to be able to calculate it.

So the technique might be to check, for each calculable root of M (which is a non empty set as 1 is always in it), the ones that have eigenvectors with first coordinates non null (in order to have a high degree invariant). So in general, we are not guaranteed to get something else than trivial λ-invariants. Nevertheless, when M is block triangular (with blocks 4×4 or less), the scale consecution technique works (the two examples given in section 1), it is also the case when the eigenspace corresponding to one is of dimension more than 1 (example given in section 3).

Theorem 4. ***(Undecidability of constant scale consecution)***

Let M *the matrix introduce in this section and* ϕ_λ *its characteristic polynomial. Finding a non trivial* λ *invariant is equivalent to finding a root of* ϕ_λ *different than* 1 *if* 1 *has*

multiplicity one.
If the degree of ϕ_λ is equal or greater than six, then $\phi_\lambda/(X-1)$ has degree greater equal or greater than five, and roots of such polynomials are usually incalculable by Galois theory on resolvability of polynomial equations.

Theorem 5. ***(Some decidable classes)*** *Let M the matrix introduce in this section. The problem of finding a* non-trivial λ-invariant *is* decidable *if one of the following assertions are true:*

- *M is block triangular (with blocks 4×4 or less) ,*
- *The eigenspace corresponding to the eigenvalue one is of dimension strictly more than* 1.

Proof. Suppose M is block triangular with blocks 4×4 or less, then it's characteristic polynomial will a product of polynomials of degree less than four, whose roots can be calculated by Lagrange's resolvent method [15].

For the second assertion, we already know that 1 is an eigenvalue, suppose that the corresponding eigenspace is of dimension exactly one, then the only vectors in that space are the constant polynomials. Whereas if it is of dimension two or more, than we get polynomials that are non trivial in the eigenspace. Looking at theorem 7 to come, we see that it is particularly interesting case. □

3.4 Initiation step

3.4.1 Intersection with an Initial Hyperplane

During the previous paragraphs, we didn't take account of the initiation step. Let's consider the affine system associated to an affine loop (as describe in definition 5), and an invariant candidate Q of degree r. Q being a λ-invariant for constant scale consecution meaning that $Q(L_1(X_1,...,X_n),...,L_n(X_1,...,X_n)) = \lambda Q(X_1,...,X_n)$. Now let $u_1,...,u_n$ be the initial values of the variables $X_1,...,X_n$, for the initial step, we need $Q(u_1,...,u_n) = 0$.

Considering the space V_r of polynomial of degree less or equal to r, we have the following linear form on this space $P \mapsto P(u_1,...,u_n)$, so that initial values correspond in terms of linear algebra , to a hyperplane of V_r, given by the kernel of $P \mapsto P(u_1,...,u_n)$. Now if we add the initiation step, $Q(X_1,...,X_n) = 0$ will be an invariant in the sense of definition 2 if and only if there exists an eigenvalue λ of M, such that Q belongs to the intersection of the eigenspace corresponding to λ, and the hyperplane $Q(u_1,\ldots,u_n) = 0$ given by the initial values $(u_1,\ldots,u_n)$. Thanks to our initiation and consecution encoding, the algebraic assertion map $\eta : L \mapsto (Q(X_1,\ldots,X_n) = 0)$ is inductive (see Definition 3).

Theorem 6. ***(Existence of invariant using constant scale consecution)*** *A polynomial Q in V_r is a true invariant for the affine loop (describe in definition 5) with initial values $(u_1,\ldots,u_n)$, if and only if there exists an eigenvalue λ of M, such that Q belongs to the intersection of the eigenspace corresponding to λ, and the hyperplane $Q(u_1,\ldots,u_n) = 0$.*

We state the most important:

Theorem 7. ***(Existence of non-null invariant using constant scale consecution for any given initial values)***

There will be a non-null invariant polynomial for any given initial values if and only if there exists an eigenspace of M with dimension more than 2.

Proof. As any hyperplane is of codimension one in V_r, it must have a nonzero intersection with any subspace of dimension strictly greater than one. Applying this to the hyperplane given by the initial values, it must have a nonzero intersection with any eigenspace of M of dimension greater than two. □

Let's get back to the following example: $\rho_\tau \equiv [x'_1 = 2x_1 + x_2 + 1 \wedge x'_2 = 3x_2 + 4]$. It's matrix M (see appendix) has six distinct eigenvalues so that each eigenspace is of dimension one, we note E_λ the eigenspace corresponding to λ. E_4 has basis $(1, -2, 1, -6, 6, 9)^\mathcal{T}$, E_6 has basis $(0, 1, -1, 2, -5, 6)^\mathcal{T}$, E_9 has basis $(0, 0, 1, 0, 4, 4)^\mathcal{T}$, E_2 has basis $(0, 0, 0, 1, -1, -3)^\mathcal{T}$, E_3 has basis $(0, 0, 0, 0, 1, 2)^\mathcal{T}$, and E_1 has basis $(0, 0, 0, 0, 0, 1)^\mathcal{T}$ Suppose the initiation step is given by: $(x_1 = 0, x_2 = -2)$, i.e. in previous notations $(u_1, u_2) = (0, 2)$, which corresponds to the hyperplane $Q(0, 2) = 0$ in V_2, or $4a_2 - 2a_4 + a_5 = 0$ in $\mathbb{R}^6$ using the canonical basis of V_2. It is clear that $(0, 0, 1, 0, 4, 4)$ belongs to this hyperplane, so that ${X_2}^2 + 4X_2 + 4$ is an invariant polynomial for the loop with initiation step $(x_1 = 0, x_2 = -2)$. With the same transition system, choose initiation step $(x_1 = 1, x_2 = 1)$, corresponding to $Q(1, 1) = 0$ or $a_0 + a_1 + a_2 + a_3 + a_4 + a_5 = 0$, we see that none of the eigenvectors above is contained in this hyperplane, so that the only invariant polynomial we will get is zero with such initial values. We finally get back to our last example 3. The corresponding matrix verifies that $dim(Ker(M - I)) \geq 2$, so that according to the previous subsection, consecution scale technique will give a non-null invariant whatever the initial values are (which explain why a non-trivial invariant was fund in [22]).

3.5 When constant scale consecution never works

Let's consider an algebraic transition system deduced from a loop of the following forms:

$$\rho_\tau \equiv \begin{bmatrix} x'_1 = P_1(x_1, \ldots, x_n) \\ \vdots \\ x'_m = P_m(x_1, \ldots, x_n) \end{bmatrix} \tag{2}$$

Where $P_1, \ldots, P_m \in A_n$. In the case where each polynomial P_i have a degree greater than 1, the constant-scale consecution encoding proposed by existing methods [22] unfortunately generate trivial (constant or null) invariants. Moreover, if each P_i is of degree greater or equal than 2, the previous methods can only generate trivial invariant.

Example 4. *Let consider the following loop:*

$$\rho_\tau \equiv \begin{bmatrix} x' = x(y + 1) \\ y' = y^2 \end{bmatrix}$$

At the step k of the iteration this loop compute the sum: $1 + y + \cdots + y^{2^k - 1}$. We consider $P(x, y) = a_0x^2 + a_1xy + a_2y^2 + a_3x + a_4y + a_5$ as candidate λ-invariant. Modulo the loop ideal

of $\mathbb{K}[x', y', x, y]$ *which is generated by the following Grobner Bases* $\{x' - x(y+1), y' - y^2\}$ *(with the total-degree lexicographic ordering with the precedence:* $x' > y' > x > y$*.), we have* $P(x', y') = P(x(y+1), y^2)$ *and we denote* $P'(x, y) = P(x(y+1), y^2)$*, after expanding we get* $P'(x, y) = a_0x^2y^2 + a_1xy^3 + a_2y^4 + 2a_0x^2y + a_1xy^2 + a_0x^2 + a_3xy + a_4y^2 + a_3x + a_5$*. If we try the constant-scale consecution with parameter* λ *we obtain:*

$$\begin{cases} a_0 = 0 & a_1 = 0 & a_3 = \lambda a_3 \\ a_1 = 0 & a_0 = \lambda a_0 & \lambda a_4 = 0 \\ a_2 = 0 & a_3 = \lambda a_1 & a_5 = \lambda a_5 \\ 2a_0 = 0 & a_4 = \lambda a_2 & \end{cases}$$

By simplification we get : $a_0 = a_1 = a_2 = a_3 = a_4 = 0$ *and* $a_5 = \lambda a_5$*. if* $\lambda \neq 1$ *then* $a_5 = 0$ *which leads to a* null *invariant. Otherwise* $\lambda = 1$ *and we obtain a* constant *invariant (*a_5*). Also, by considering the initial condition, we remark that it will implies that the constant invariant* a_5 *is null.*

4 Polynomial consecution to handle non-linear algebraic transition systems

4.1 T-Invariant Generation and a First Example

Definition 6. *We consider algebraic transition system corresponding to an* algebraic *loop* $\tau = \langle l_i, l_j, \rho_\tau \rangle$ *as describe in Section 3.5, equation 2 where* $P_1, ..., P_n \in \mathbb{K}[x_1, ..., x_n]$*. A Polynomial* $Q \in K[X_1, ..., X_n]$ *is said to be a* T*-invariant for polynomial-scale consecution for the loop* τ *if and only if there exists a polynomial* $T \in K[X_1, ..., X_n]$*, verifying*

$$Q(X'_1, \ldots, X'_n) = T(X_1, \ldots, X_n)Q(X_1, \ldots, X_n)$$

modulo the Ideal of $K[X'_1, ..., X'_n, X_1, ..., X_n]$ *corresponding to the loop, generated by the grobner base* $(X'_1 - P_1(X_1, ..., X_n), ..., X'_n - P_n(X_1, ..., X_n))$*.*

Theorem 8. ***(*T*-invariant's characterization)*** *Consider an algebraic transition system corresponding to an algebraic loop* τ *as describe in Section 3.5, equation 2. Let* $Q \in K[X_1, ..., X_n]$ *be a multivariate polynomial with indeterminate coefficients (a template).* Q *is a* T*-invariant for polynomial scale consecution with* parametric polynomial $T \in K[X_1, ..., X_n]$ *for* τ *if and only if*

$$Q(P_1(X_1, \ldots, X_n), \ldots, P_n(X_1, \ldots, X_n)) = T(X_1, \ldots, X_n)Q(X_1, \ldots, X_n)$$

Proof. If $Q(X'_1, .., X'_n) - TQ(X_1, .., X_n)$ belongs to the ideal I generated by the family $(X'_1 - P_1, \ldots, X'_n - P_n)$, then there exists a family $(A_1, \ldots, A_n)$ of polynomials in $K[X'_1, .., X'_n, X_1, .., X_n]$ such that

$$Q(X'_1, .., X'_n) - \lambda Q(X_1, .., X_n) = (X'_1 - P_1)A_1 + \cdots + (X'_n - P_n)A_n$$

. Letting $X'_i = P_i$, we obtain that $Q(P_1(X_1, ..., X_n), ..., P_n(X_1, ..., X_n)) = TQ(X_1, ..., X_n)$ Conversely suppose $Q(P_1(X_1, \ldots, X_n), .., P_n(X_1, \ldots, X_n)) = TQ(X_1, \ldots, X_n)$, then as $Q(X'_1, .., X'_n)$ is equal to $Q(P_1, .., P_n)$ modulo the ideal I, we get that $Q(X'_1, .., X'_n) = \lambda Q(X_1, \ldots, X_n)$ modulo I. □

Let's get back to the example 4. We take $(x = x_0, s = 1)$ as initial values. We propose to use *polynomial scale consecution* with parametric polynomial $T(s, x) = b_0x^2 + b_1s + b_2x + b_3$. We thus obtain the following equation: $P'(s, x) = (b_0x^2 + b_1s + b_2x + b_3) \cdot P(s, x)$. In other words we obtain the following multi-parametric linear system (with parameters b_0, b_1, b_2, b_3):

$$\left\{\begin{array}{lll} a_0 = b_0a_0 & 0 = b_2a_5 + b_3a_4 & a_3 = b_1a_4 + b_2a_3 + b_3a_1 \\ a_1 = b_0a_1 & 0 = b_0a_4 + b_2a_2 & a_4 = b_0a_5 + b_2a_4 + b_3a_2 \\ a_2 = b_0a_2 & a_3 = b_1a_5 + b_3a_3 & a_1 = a_3b_0 + b_1a_2 + b_2a_1 \\ a_5 = b_3a_5 & a_0 = b_1a_3 + b_3a_0 & \\ 0 = b_1a_0 & 2a_0 = b_1a_1 + b_2a_0 & \end{array}\right.$$

Now we explain a first decision procedure for the valuations of the parameters. Considering the three first equations, we choose $b_0 = 1$ in order to keep a high degree invariant (otherwise the coefficients a_0, a_1, a_2, which are those of highest degree terms would be *null*). Then we obtain an other system with the equation $b_1a_0 = 0$. For the same degree conserving reason we choose $b_1 = 0$. Then, in the resulting system we have the equation $b_2a_0 = 2a_0$. As a direct consequence, the parameter b_2 is set to 2. Because of the presence of the equation $b_3a_0 = a_0$ in the resulting system, b_3 is set to 1. We finally obtain the following system :

$$\left\{\begin{array}{c} a_3 + a_1 = 0 \\ a_4 + 2a_2 = 0 \\ a_2 - a_5 = 0 \end{array}\right.$$

As we have less equations than variables. we already can state that the existence of a *non-trivial* solution for the generation of T-invariant. Now we add the hyperplane corresponding to the initial values: $a_2{x_0}^2 + (a_1 + a_4)x_0 + a_0 + a_1 + a_5 = 0$. As there are six variables and four equations, we already can state the existence of a *non-trivial* solution for the problem of invariant generation. A possible solutions is the vector $(x_0(1 - x_0), 1, 1, -1, -2, 1)^{\mathcal{T}}$, in other words, $x_0(1 - x_0)x^2 + xy + y^2 - x - 2y + 1 = 0$ is an invariant. We've been using polynomial scale consecution with polynomial $T(t, y) = y^2 + y + 1$.

Remark 1. *We gave a quite simple decision procedure that did work luckily, but in more complex cases, it might fail, we will give a better technique, with a more global point of view in the next paragraphs.*

4.2 General Theory for Polynomial Scale Consecution

We are looking for a T-invariant of degree r for an algebraic loop $\tau = \langle l_i, l_i, \rho_\tau \rangle$ (as describe in Section 3.5, equation 2), that is, a polynomial Q of degree r such that there exists a polynomial T, verifying

$$Q(P_1(X_1, .., X_n), .., P_n(X_1, .., X_n)) = T(X_1, .., X_n)Q(X_1, .., X_n)$$

To translate that in terms of linear algebra, we write again the T-invariant candidate's coefficients ordered $a_0, ..., a_t$ ($t + 1$ being the number of monomials of degree inferior to r). Let d be the maximal degree of the P_i's, we are thus going to look for T of degree $e = dr - r$. Let's write its ordered coefficients $\lambda_0, ..., \lambda_s$ ($s + 1$ being the number of monomials of degree inferior to e). We remind that V_m designs the subspace of $\mathbb{K}[x_1, \ldots, x_n]$ of degree inferior

or equal to m. Let M be the matrix in the canonical basis of V_r and V_{dr}, of the morphism from V_r to V_{dr} given by

$$P(X_1, \ldots, X_n) \mapsto P(P_1(X_1, \ldots, X_n), \ldots, P_n(X_1, \ldots, X_n)$$

, the coefficients of M will be polynomials in those of the P_i's. Let L be the matrix in the canonical basis of V_r and V_e, of the morphism from V_r to V_{dr} given by:

$$P \mapsto TP$$

The matrix L has a very simple form which will be clear after our examples, right now we just say it's non zero coefficients are λ_i's, and it has a natural block decomposition. Let u be the number of terms of degree less than dr (i.e. the dimension of V_{rd}), our problem comes to the same thing as finding such a matrix L, such that $M - L$ has a non trivial kernel, in other words such that $M - L$ is of rank less than u. Due to Galois's Linear Algebra theory, we know that a Matrix $\mathcal{M}$ is of rank less than $k \in \mathbb{N}$ if and only if there exist an *invert* $k * k$ sub-matrix of $\mathcal{M}$

Theorem 9. ***(Existence of T-invariant vectorspace)*** *Considering M as described just above. There will be a T-invariant polynomial if and only if there exists a matrix L (corresponding to $P \mapsto TP$) such that $M - L$ has a nontrivial kernel. In this situation, any vector in the kernel of $M - L$ will give a T-invariant polynomial.*

Proof. In fact, a polynomial Q is T- invariant if and only if $Q(P_1(X_1, .., X_n), .., P_n(X_1, .., X_n)) = T(X_1, .., X_n)Q(X_1, .., X_n)$, i.e. if and only if $m(Q) = l(Q) \Leftrightarrow (m - l)(Q) = 0$ from definition of endomorphisms m and l. Writing this into matrices, we get the statement of the theorem. □

Again the last column of M is $(0, ..., 0, 1)^{\mathcal{T}}$, and the last column of L is $(0, .., 0, \lambda_0, .., \lambda_s)^{\mathcal{T}}$, so that if we choose every λ_i to be zero, except $\lambda_s = 1$, the last column of M - L will be null. Thus for this choice of L (or $T = 1$), we at least always get a T-invariants corresponding to constant polynomials. Now let's notice that $M - L$ having non trivial kernel is equivalent for it to have rank strictly less than the dimension $d(r)$ of V_r. By classical theorem of linear algebra [15], it is equivalent to the fact that each $d(r) \times d(r)$ subdeterminant of $M - L$ is equal to zero. Those determinants are polynomials with variables $(\lambda_0, \lambda_1, \cdots, \lambda_s)$, which we will note $D_1(\lambda_0, \lambda_1, \cdots, \lambda_s), \ldots, D_t(\lambda_0, \lambda_1, \cdots, \lambda_s)$.

Theorem 10. ***(Undecidability of finding T-invariants)*** *There will be a non trivial T-invariant if and only if the polynomials $(D_1, .., D_r)$ described just above admit a common root, other than the trivial one $(0, \cdots, \cdots, \cdots, 0, 1)$. Those roots are in general not calculable.*

We will give some examples of decidable classes in the following section.

Example 5. ***Loop with Two Variables, T-Invariant of Degree Two*** *To get an idea of the situation, we first study the general case of degree two algebraic transition systems with two variables in the loop. The transition system has the form:*

$$\rho_\tau = \begin{bmatrix} x' = c_0x^2 + c_1xy + c_2y^2 + c_3x + c_4y + c_5 \\ y' = d_0x^2 + d_1xy + d_2y^2 + d_3x + d_4y + d_5 \end{bmatrix}$$

then

$$M = \begin{pmatrix} {c_0}^2 & c_0d_0 & {d_0}^2 & 0 & 0 & 0 \\ 2c_0c_1 & c_0d_1 + c_1d_0 & 2d_0d_1 & 0 & 0 & 0 \\ 2c_0c_2 + {c_1}^2 & c_0d_2 + c_1d_1 + c_2d_0 & 2d_0d_2 + {d_1}^2 & 0 & 0 & 0 \\ 2c_1d_1 & c_1d_2 + c_2d_1 & 2d_1d_2 & 0 & 0 & 0 \\ {c_2}^2 & c_2d_2 & {d_2}^2 & 0 & 0 & 0 \\ 2c_0c_3 & c_0d_3 + c_3d_0 & 2d_0d_3 & 0 & 0 & 0 \\ 2(c_0c_4 + c_1c_3) & c_0d_4 + c_1d_3 + c_3d_1 + c_4d_0 & 2(d_0d_4 + d_1d_3) & 0 & 0 & 0 \\ 2(c_1c_4 + c_2c_3) & c_1d_4 + c_2d_3 + c_3d_2 + c_4d_1 & 2(d_1d_4 + d_2d_3) & 0 & 0 & 0 \\ 2c_2c_4 & c_2d_4 + c_4d_2 & 2d_2d_4 & 0 & 0 & 0 \\ 2c_0c_5 + {c_3}^2 & c_0d_5 + c_3d_3 + c_5d_0 & 2d_0d_5 + {d_3}^2 & c_0 & d_0 & 0 \\ 2(c_1c_5 + c_3c_4) & c_1d_5 + c_3d_4 + c_4d_3 + c_5d_1 & 2(d_1d_5 + d_3d_4) & c_1 & d_1 & 0 \\ 2c_2c_5 + {c_4}^2 & c_2d_5 + c_4d_4 + c_5d_2 & 2d_2d_5 + {d_4}^2 & c_2 & d_2 & 0 \\ 2c_3c_5 & c_3d_5 + c_5d_3 & 2d_3d_5 & c_3 & d_3 & 0 \\ 2c_4c_5 & c_4d_5 + c_5d_4 & 2d_4d_5 & c_4 & d_4 & 0 \\ {c_5}^2 & c_5d_5 & {d_5}^2 & c_5 & d_5 & 1 \end{pmatrix}$$

$$L = \begin{pmatrix} \lambda_0 & 0 & 0 & 0 & 0 & 0 \\ \lambda_1 & \lambda_0 & 0 & 0 & 0 & 0 \\ \lambda_2 & \lambda_1 & \lambda_0 & 0 & 0 & 0 \\ 0 & \lambda_2 & \lambda_1 & 0 & 0 & 0 \\ 0 & 0 & \lambda_2 & 0 & 0 & \\ \lambda_3 & 0 & 0 & \lambda_0 & 0 & 0 \\ \lambda_4 & \lambda_3 & 0 & \lambda_1 & \lambda_0 & 0 \\ 0 & \lambda_4 & \lambda_3 & \lambda_2 & \lambda_1 & 0 \\ 0 & 0 & \lambda_4 & 0 & \lambda_2 & 0 \\ \lambda_5 & 0 & 0 & \lambda_3 & 0 & \lambda_0 \\ 0 & \lambda_5 & 0 & \lambda_4 & \lambda_3 & \lambda_1 \\ 0 & 0 & \lambda_5 & 0 & \lambda_4 & \lambda_2 \\ 0 & 0 & 0 & \lambda_5 & 0 & \lambda_3 \\ 0 & 0 & 0 & 0 & \lambda_5 & \lambda_4 \\ 0 & 0 & 0 & 0 & 0 & \lambda_5 \end{pmatrix}$$

In this case, to choose L such that the rank of $M - L$ is less than 6, one has to calculate each subdeterminant 6×6 obtained by cancelling 9 lines of $M - L$. Those determinant will be polynomials in variables $(\lambda_0, ..., \lambda_5)$ of degree less than 6. Now L is such that $M - L$ will be of degree less than 6, if and only if $(\lambda_0, ..., \lambda_5)$ are roots of each of those polynomials.

Remark 2. ***(Decidable classes)*** *In many particular cases, it's easy to find a matrix L such that $M - L$ has non trivial kernel. Here we describe two decidable classes:*

- *for example suppose that in the previous case, c_2, c_4 and c_5 are null, then one can choose $(\lambda_0, \ldots, \lambda_s)$ in order to make first column zero.*

- *the third column can be cancelled for good choices of the λ_i's if d_0, d_3 and d_5 are zero.*

Remark 3. *In many particular cases, it's easy to find a matrix L such that $M - L$ has non trivial kernel.*

For example suppose that in the previous case, c_2, c_4 and c_5 are null, then one can choose $(\lambda_0, \ldots, \lambda_s)$ in order to make first column zero. Now we consider the example 4 in Section 3.5. Here we have $(c_0 = 0, c_1 = 1, c_2 = 0, c_3. = 1, c_4. = 0, c_5. = 0)$, and $(d_0 = 0, d_1 = 0, d_2 = 1, d_3. = 0, d_4. = 0, d_5. = 0)$.

Then the matrix M is:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

And the matrix $M - L$ is:

$$\begin{pmatrix} -\lambda_0 & 0 & 0 & 0 & 0 & 0 \\ -\lambda_1 & -\lambda_0 & 0 & 0 & 0 & 0 \\ 1-\lambda_2 & -\lambda_1 & -\lambda_0 & 0 & 0 & 0 \\ 0 & 1-\lambda_2 & -\lambda_1 & 0 & 0 & 0 \\ 0 & 0 & 1-\lambda_2 & 0 & 0 & 0 \\ -\lambda_3 & 0 & 0 & -\lambda_0 & 0 & 0 \\ 2-\lambda_4 & -\lambda_3 & 0 & -\lambda_1 & -\lambda_0 & 0 \\ 0 & 1-\lambda_4 & -\lambda_3 & -\lambda_2 & -\lambda_1 & 0 \\ 0 & 0 & -\lambda_4 & 0 & -\lambda_2 & 0 \\ 1-\lambda_5 & 0 & 0 & -\lambda_3 & 0 & -\lambda_0 \\ 0 & -\lambda_5 & 0 & 1-\lambda_4 & -\lambda_3 & -\lambda_1 \\ 0 & 0 & -\lambda_5 & 0 & 1-\lambda_4 & -\lambda_2 \\ 0 & 0 & 0 & 1-\lambda_5 & 0 & -\lambda_3 \\ 0 & 0 & 0 & 0 & -\lambda_5 & -\lambda_4 \\ 0 & 0 & 0 & 0 & 0 & 1-\lambda_5 \end{pmatrix}$$

Now we directly see that taking $\lambda_0 = \lambda_1 = \lambda_3 = 0, \lambda_2 = 1, \lambda_4 = 2, \lambda_5 = 1$, the first column of $M - L$ is zero, and the second column is equal to the fourth. So with this choice of L (i.e. $T(x, y) = y^2 + 2y + 1$), $M - L$ will be of rank equal or less than four, i.e. with kernel of dimension equal or more than two. Any vector in the kernel will be T-invariant.

4.3 Initiation step

It is the same thing as previously. If we are looking for an invariant $Q \in k[X_1, ..., X_n]$, Let $u_1, \ldots, u_n$ be the initial values of the variables $X_1, ..., X_n$, for the initial step, we need $Q(u_1, \ldots, u_n) = 0$. Considering the space V_r of polynomial of degree less or equal to r, we have the following linear form on this space $P \mapsto P(u_1, \ldots, u_n)$, so that initial values correspond in terms of linear algebra , to a hyperplane of V_r, given by the kernel of $P \mapsto P(u_1, \ldots, u_n)$.

Theorem 11. ***(Existence of non trivial invariant using Polynomial scale consecution)*** *We will have a non trivial invariant if and only if there exists a matrix L (the one of $P \mapsto TP$ in canonical basis' with non null coefficients of T being $\lambda_0, ..., \lambda_s$), such that the intersection of the kernel of $M - L$ and the hyperplane given by the initial values is not zero, the invariants corresponding to vectors in the intersection.*

Proof. We refer to the one of theorem 6. □

The most practical case is given by:

Theorem 12. ***(Existence of non trivial invariant using Polynomial scale consecution for any given initial values)*** *If one can find T (i.e. L) such that $dim(Ker(M - L)) \geq 2$, for any initiation step, there will always be non trivial invariants.*

Proof. We refer to the one of theorem 7. □

If we get back to our preceding example, with $T(x, y) = y^2 + 2y + 1$, $M - L$ verifies the hypothesis of the theorem, so that we are guaranteed of the existence of an invariant, whatever the initial values are. We remind that for initial step $(x = x_0, s = 1)$, a possible invariant is given by $x_0(1 - x_0)x^2 + xy + y^2 - x - 2y + 1$.

5 Fractional Scale Consecution

5.1 Theory for fractional scale consecution

We want to deal with transition systems of the following type:

$$\rho_\tau = \begin{bmatrix} x'_1 = P_1(x_1, \ldots, x_n)/Q_1(x_1, \ldots, x_n) \\ \vdots \\ x'_n = P_n(x_1, \ldots, x_n)/Q_n(x_1, \ldots, x_n) \end{bmatrix} \tag{3}$$

where P_i's and Q_i's belong to $K[X_1, ..., X_n]$, and P_i being relatively prime to Q_i.

Definition 7. *A Polynomial $Q \in K[X_1, \ldots, X_n]$ is said to be a T-invariant for polynomial-scale consecution for the loop τ if and only if there exists a rational function $F \in K(X_1, ..., X_n)$, verifying*

$$Q(X'_1, ..., X'_n) = F(X_1, ..., X_n)Q(X_1, ..., X_n)$$

modulo the fractional Ideal of $K(X'_1, ..., X'_n, X_1, ..., X_n)$ corresponding to the loop, generated by $\left(X'_1 - \frac{P_1(X_1,...,X_n)}{Q_1(X_1,...,X_n)}, \ldots, X'_n - \frac{P_n(X_1,...,X_n)}{Q_n(X_1,...,X_n)}\right)$.

Theorem 13. ***(F-invariant's characterization)*** *Consider an algebraic transition system corresponding to an algebraic loop τ as describe in definition 7.*

Let $Q \in K[X_1, \ldots, X_n]$ be a multivariate polynomial with indeterminate coefficients (a template). Q is a F-invariant for polynomial scale consecution with parametric polynomial *$F \in K(X_1, \ldots, X_n)$ for τ if and only if*

$$Q\left(\frac{P_1(X_1, ..., X_n)}{Q_1(X_1, ..., X_n)}, \ldots, \frac{P_n(X_1, ..., X_n)}{Q_n(x_1, ..., x_n)}\right) = F(X_1, ..., X_n)Q(X_1, ..., X_n)$$

Proof. If $Q(X'_1, .., X'_n) - FQ(X_1, .., X_n)$ belongs to the fractional ideal J generated bey the family $(X'_1 - P_1/Q_1, \ldots, X'_n - P_n/Q_n)$, then there exists a family $(A_1, \ldots, A_n)$ of fractional functions in $K(X'_1, .., X'_n, X_1, .., X_n)$ such that $Q(X'_1, .., X'_n) - FQ(X_1, .., X_n) = (X'_1 - P_1/Q_1)A_1 + \cdots + (X'_n - P_n/Q_n)A_n$. Letting $X'_i = P_i/Q_i$, we obtain that $Q(P_1/Q_1, .., P_n/Q_n) = \lambda Q(X_1, \ldots, X_n)$.

Conversely suppose $Q(P_1/Q_1, .., P_n/Q_n) = FQ(X_1, \ldots, X_n)$, then as $Q(X'_1, .., X'_n)$ is equal to $Q(P_1/Q_1, .., P_n/Q_n)$ modulo the ideal J, we get that $Q(X'_1, .., X'_n) = FQ(X_1, \ldots, X_n)$ modulo J. □

Let d be the maximal degree of the P_i's and the Q_i's, and let Π be the lcm of the Q_i's. Now let $U = {X_1}^{i_1}..{X_n}^{i_n}$ be a monomial of degree less than r (i.e. $i_1 + .. + i_n \leq r$), then: $\Pi^r U(P_1/Q_1, \ldots, P_n/Q_n) = \Pi^r (P_1/Q_1)^{i_1}...(P_n/Q_n)^{i_n}$. But as $Q_j^{i_j}$ divides Π^{i_j}, for all j, we see that $Q_1^{i_1}...Q_n^{i_n}$ divides $\Pi^{i_1+\ldots+i_r}$ which divides itself Π^r. We deduce that $\Pi^r Q(P_1/Q_1, \ldots, P_n/Q_n)$ is a polynomial for every Q in V_r. Now suppose $F = T/S$ (T relatively prime to S) satisfies the equality of the previous theorem and suppose we are looking for an invariant Q of degree r. Then multiplying by Π^r, we get $\Pi^r Q(P_1/Q_1, \ldots, P_n/Q_n) = (\Pi^r TQ)/S$. As we have a priory no information on Q, in most of the cases Q will be relatively prime to S. In this case we see that S will divide Π^r, so that rewriting F, we can suppose it has denominator Π^r. From now on we will suppose that F is of the form T/Π^r, as we just saw that this constraint is very little restrictive. Now let's call m the morphism of vector spaces $Q \mapsto \Pi^r Q(P_1/Q_1, \ldots, P_n/Q_n)$ from V_r to V_{nrd}, and let M be it's matrix in canonical basis. Let T be a polynomial in V_{nrd-r}, and let's note l the morphism of vector spaces $Q \mapsto TQ$ from V_r to V_{nrd}, and let L be it's matrix in canonical basis. Combining theorem 11, and the preceding discussion, we have the following theorem:

Theorem 14. *Let M be as described just above. There will exist a F-invariant (with the restriction that F is of the form T/Π^r) polynomial if and only if there exists a matrix L (corresponding to $Q \mapsto TQ$) such that $M - L$ has a nontrivial kernel. In this situation, any vector in the kernel of $M - L$ will give a F-invariant polynomial.*

We refer to the preceding section for decidability, as the theorem is of the same form as theorem 9. For the initiation step, as before initial values give a hyperplane of V_r, but in order to the transition system to have sense, the n-tuple of initial values must not be a root of any of the Q_i's, and so must be their iterates as long as the loop is applied. Then we are guaranteed that they won't cancel Π^r. As before we have the following theorem:

Theorem 15. ***(Existence of non trivial invariant using Fractional scale consecution)*** *We will have a non trivial invariant if and only if there exists a matrix L (the one of $Q \mapsto TQ$ in canonical basis' with coefficients of T being $\lambda_0, ..., \lambda_s$), such that the intersection of the kernel of $M - L$ and the hyperplane given by the initial values (good initial values) is not zero, the invariants corresponding to vectors in the intersection.*

In this case we also have the important theorem:

Theorem 16. ***(Existence of non trivial invariant using Polynomial scale consecution for any initial value)*** *We will have a non trivial invariant for any "good"(non-trivial) initial value if there exists a matrix L , such that the kernel of $M-L$ is of dimension equal or greater than 2.*

Example 6. *We consider the following system:*

$$\rho_\tau = \begin{bmatrix} x_1' = x_2/(x_1 + x_2) \\ x_2' = x_1/(x_1 + 2x_2) \end{bmatrix}$$

We are looking for a F-invariant polynomial of degree two. Here the least common multiple of $(x_1 + x_2)$ and $(x_1 + 2x_2)$ is their product, so that m is given by: $[Q \in V_2 \mapsto [(x_1 + x_2)(x_1 + 2x_2)]^2 Q(x_1/(x_1 + x_2), x_2/(x_1 + 2x_2))]$. Here as both $x_2/(x_1 + x_2)$ and $x_1/(x_1 + 2x_2)$ have "degree" zero, $[(x_1 + x_2)(x_1 + 2x_2)]^2 Q(x_2/(x_1 + x_2), x_1/(x_1 + 2x_2))$ will be exactly a linear combination of degree four terms if it is non null. Hence m has values in $Vect(X_1^4, X_1^3X_2, X_1^2X_2^2, X_1X_2^3, X_2^4)$. For a polynomial T and $Q \in V_2$ to verify: $[(x_1 + x_2)(x_1 + 2x_2)]^2 Q(x_2/(x_1 + x_2), x_1/(x_1 + 2x_2)) = TQ$, as the left member is in $Vect(X_1^4, X_1^3X_2, X_1^2X_2^2, X_1X_2^3, X_2^4)$, T must be of the form $\lambda_0 X_1^2 + \lambda_1 X_1X_2 + \lambda_3 X_2^2$ and Q of the form $a_0X_1^2 + a_1X_1X_2 + a_3X_2^2$. From this little preliminary discussion, we see that we already can take Q in $Vect(X_1^2, X_1X_2, X_2^2)$, and T also. Then both m and $l : (Q \mapsto TQ)$ will be morphisms from $Vect(X_1^2, X_1X_2, X_2^2)$ in $Vect(X_1^4, X_1^3X_2, X_1^2X_2^2, X_1X_2^3, X_2^4)$. In the corresponding canonical basis, the matrix M is :

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 2 \\ 1 & 3 & 1 \\ 4 & 2 & 0 \\ 4 & 0 & 0 \end{pmatrix}$$

and L will be:

$$\begin{pmatrix} \lambda_0 & 0 & 0 \\ \lambda_1 & \lambda_0 & 0 \\ \lambda_2 & \lambda_1 & \lambda_0 \\ 0 & \lambda_2 & \lambda_1 \\ 0 & 0 & \lambda_2 \end{pmatrix}$$

In other words, $M - L$ will be the following matrix:

$$\begin{pmatrix} -\lambda_0 & 0 & 1 \\ -\lambda_1 & 1-\lambda_0 & 2 \\ 1-\lambda_2 & 3-\lambda_1 & 1-\lambda_0 \\ 4 & 2-\lambda_2 & -\lambda_1 \\ 4 & 0 & -\lambda_2 \end{pmatrix}$$

Taking $(\lambda_0 = 1, \lambda_1 = 3, \lambda_2 = 2)$ *actually cancels the second column, hence with this choice of* L, $M - L$ *has kernel equal to* $Vect(0, 1, 0)$. *It was in fact possible to see from the beginning that the corresponding polynomial* X_1X_2 *is* $(X_1^2+3X_1X_2+2X_2^2)/[(X_1+X_2)(X_1+2X_2)]^2$*-invariant. It is an invariant (see definition 2) for initial values* $(0, 1)$ *(which iterates clearly never cancel* X_1+X_2 *and* X_1+2X_2, *because they are of the form* $(a, 0)$ *or* $(0, b)$ *with* a *and* b *strictly positive)*

6 Branching Conditions and Nested Loops

Using the methods described so far we obtain eigenspaces where each point corresponds to a non-trivial non-linear invariant. Then we can consider the eigenvectors to form a basis such that the induced ideal is an Ideal of non-trivial non-linear loop invariants. Here, we show how our method deals with the conditional statements inside loops. Let's consider the following type of loop : ... `While(B_1){ [I_1;] If(B_2){ [I_2;] } Else{ [I_3;] } [I_4;] }` ... where I_is are notations to represent a block of multivariate fractional instructions. Our algorithm will first represent the loop with the two following transitions $\tau_1 = \langle l_i, l_i, (\mathcal{B}_1 \wedge \mathcal{B}_2), \rho_{\tau_1}\rangle$, and $\tau_2 = \langle l_i, l_i, (\mathcal{B}_1 \wedge \neg\mathcal{B}_2), \rho_{\tau_2}\rangle$ where: $\rho_{\tau_1} \equiv [x'_1 = F_{1,[I_1;I_2;I_4;]_\circ}(x_1, ...x_n), \ldots, x'_n = F_{n,[I_1;I_2;I_4;]_\circ}(x_1, ..., x_n)]$ and $\rho_{\tau_2} \equiv [x'_1 = F_{1,[I_1;I_3;I_4;]_\circ}(x_1, ...x_n) \ldots, x'_n = F_{n,[I_1;I_3;I_4;]_\circ}(x_1, ..., x_n)]$ with $[;;]_\circ$ denoting our operator based on separation rewritting rules to compose blocks of instructions. Our algorithm first generates independently the ideals of invariant $\xi_1 = (\mu_1, ..., \mu_n)$ and $\xi_2 = (\kappa_1, ..., \kappa_p)$ for the respective transitions τ_1 and τ_2. Any element $\mu_i \in \xi_1$ refers to an inductive invariant $\mu_i(X_1, ..., X_n) = 0$ corresponding to the *partial loops* described by transition τ_1. Respectively, any $\kappa_i \in \xi_2$ refers to an inductive invariant $\kappa_i(X_1, ..., X_n) = 0$ for the loop described by the transition τ_2. Then we can take $\mu_i(X_1, ..., X_n) * \kappa_i(X_1, ..., X_n) = 0$ as global loop invariants, since these invariant will remind true in any sequences of transitions during the execution of the loop. As our approach generates *ideals of non-trivial and non-linear invariants*, we also propose a `Join` operator to finally generate the *ideal of global loop invariants.*

Example 7. *A running example which is beyond the limit of state-of-the-art aches. The result given by our methods is on the right. The invariant generated is* $u\_0 * (1 - u\_0) * X * Y * Z^2 * + X * Y * Z * U + X * Y * U^2 - X * Y * Z - 2 * X * Y * U + X * Y = 0$

```
1  //initialization
2          ...
3  int u_0;
4          ...
5  ((M > 0)&&(Z = 1)&&(U = u_0)...)
6          ...
7    While ((X>=1) || (Z>=z_0)){
8          ...
9          If(Y > M){
10               X = Y / (X + Y);
11               Y = X / (X + 2 * Y);
12         }
13         Else{
14              Z = Z * (U + 1);
15              U = U^2;
16         }
17   }
18     ...
```

Then we generate the following invariants :

```
          ...
==================================
```

```
[Ordering:=] Lexicographic
==================================
              ...
==================================
Dumping results for analysis mode
[Fractional-Scaling]
>From [Line 7] to [Line 10]
Relationships between vars
of func [File_Test_0] cond [_IF_1]
              ...
[Decision:=] [Lambda[0]=1;Lambda[1]=3;Lambda[2]=2]
[Kernel(M-L):=] [Vect(0,1,0)]
[Fractional consecution:=]
[[(X^2+3*X*Y+2*Y^2 )/((X+Y)(X+2*Y))^2]-Invariant]
[_IF_1][Invariant:=]
[X*Y]
==================================
Dumping results for analysis mode
[Polynomial-Scaling]
>From [Line 11] to [Line 14]
Relationships between vars
of func [File_Test_0] cond [_ELSE_1]
                 ...
[Polynomial consecution:=]
[[U^2+U+1]-Invariant]
[_ELSE_1][Invariant:=]
[u_0*(1-u_0)*Z^2+Z*U+U^2-Z-2*U+1]
==================================
Dumping results for Join mode
[Ideal Product]
[Asking for only one invariant]
>From [Line 5] to [Line 15]
Relationships between all vars
of [File_Test_0] loop [_WHILE_1]
                 ...
[_WHILE_1][Invariant:=]
[u_0*(1-u_0)*X*Y*Z^2*+X*Y*Z*U
+X*Y*U^2-X*Y*Z-2*X*Y*U+X*Y]
```

In the following theorem, we formalize and generalize the method just described.

Theorem 17. *Let $I = \{I_1, ..., I_k\}$ a set of ideals in $K[X_1, ..., X_n]$ such that $I_j = ({f^{(j)}}_1, ..., f^{(j)}_{n_j})$ where $j \in [1, k]$. Let's $\nabla(I_1, ..., I_k) = \{\delta_1, ..., \delta_{n_1 n_2 ... n_k}\}$ such that all elements δ_i in $\nabla(I_1, ..., I_k)$ are formed by the product of one element from each ideal in I. Assume that all I_js are ideals of invariants for a loop at location l_j described by a transition τ_j. Now, if all l_j describe the same location program point, then we have several transitions* looping *at the same point. So we obtain an encoding of possible execution paths of a loop containing* conditional statements.

Then $\nabla(I_1, ..., I_k)$ is an ideal of non-trivial non-linear invariants for the entire loop located at l_j.

Once again, here there are no need for Grobner Basis computation and the complexity of the steps described remain linear. Example 7 illustrate our method for the case where the loop contains two conditional statements. Our algorithm first generates an invariant for the loop corresponding to the first condition If at line 6 using Fractional-Scaling ($[\_IF\_1][Invariant :=][X * Y]$). See Example 6 for more details. Then it computes the invariant ($[\_ELSE\_1][Invariant :=][u\_0 * (1 - u\_0) * Z^2 + Z * U + U^2 - Z - 2 * U + 1]$) corresponding to the other alternative transition τ_2 of the loop (`Else` at line 10). In that case we were only asked for one invariant as the `Join` operator only returned the product of the two previously computed invariants ($[\_WHILE\_1][Invariant :=][u\_0 * (1 - u\_0) * X * Y * Z^2 * + X * Y * Z * U + X * Y * U^2 - X * Y * Z - 2 * X * Y * U + X * Y]$).

In presence of nested loops, our methods generates ideals of invariants for each inner-loop and then generats a global invariant considering the non-linear system composed of pre-computed invariants.

7 $\sum u_i f_i \in H$?

Let $H = (h_1, .., h_r)$ be an ideal of $K[x_1, .., x_n]$, and $F = \{f_1, .., f_k\}$ be a family of polynomials of $K[x_1, .., x_n]$, we want a criterion determining if a k-uple $(u_1, .., u_k)$ verifies $\sum u_i f_i \in H$.

To do so, we first notice that $(u_1, .., u_k)$ verifies $\sum_{i=1}^{k} u_i f_i \in H$ iff there exists $(u_{k+1}, .., u_{k+r})$ in $K[x_1, .., x_n]^r$ which verifies $u_1 f_1 + .. + u_k f_k = u_{k+1} h_1 + .. + u_{k+r} h_r$.
Rewriting this equality $u_1 f_1 + .. + u_k f_k - u_{k+1} h_1 - .. - u_{k+r} h_r = 0$, we see that $\sum_{i=1}^{k} u_i f_i \in H$ iff there exists $(u_{k+1}, .., u_{k+r})$ in $K[x_1, .., x_n]^r$ such that the vector $(u_1, u_2, \ldots, u_{k+r}^{\mathcal{T}}$ belongs to de the module $Syz(f_1, .., f_k, -h_1, .., -h_r)$.

Now let $W_1, .., W_p$ be a syzygy basis of $Syz(f_1, .., f_k, -h_1, .., -h_r) \subset K[x_1, .., x_n]^{k+r}$, then we see that $(u_1, .., u_k)$ verifies $\sum u_i f_i \in H$ iff $(u_1, u_2, \ldots, u_k)^{\mathcal{T}}$ belongs to the submodule of $K[x_1, .., x_n]^k$ generated by $(V_1, .., V_p)$ where V_i is the vector obtained from erasing the r last coordinates of W_i.

We thus obtained the following theorem:

Theorem 18. *Let $H = (h_1, .., h_r)$ be an ideal of $K[x_1, .., x_n]$, and $F = \{f_1, .., f_k\}$ be a family of polynomials of $K[x_1, .., x_n]$.*
Let also $W_1, .., W_p$ be a syzygy basis of $Syz(f_1, .., f_k, -h_1, .., -h_r)$, and V_i be the vector obtained from erasing the r last coordinates of W_i.

If $(u_1, .., u_k)$ is a k-uple of $K[x_1, .., x_n]^k$, then $\sum_{i=1}^{k} u_i f_i \in H$ if and only if the vector $(u_1, u_2, \ldots, u_k)^{\mathcal{T}}$ belongs to the submodule of $K[x_1, .., x_n]^k$ generated by $(V_1, .., V_p)$.

8 Conclusion

Our methods do not required computation of Grobner bases, quantifier elimination, cylindrical algebraic decomposition or direct resolution of semi-algebraic system, as well as they do not depend on any abstraction methods. We succeeded in reducing the non-linear loop invariant generation problem to the intersection between eigenspaces of specific endomorphisms and initial linear or semi-affine/algebraic constraints. Our non-trivial non-linear invariant generation method is *sound* and *complete* as we provide a complete encoding to handle multivariate fractional Loop (algebraic system with multivariate rational functions) where variable are constrained initially with parameters. As far as it is our knowledge, these are the first non-linear invariant generation methods that handle multivariate fractional instructions, conditional statement, and nested inner-loops. Also, for each type of system, we presented necessary and sufficient conditions for the existence of *non-trivial* non-linear loop invariants. Considering the problem of invariant generation, we identified a large decidable class together with an undecidable class. Finally, our methods generates ideals of non-trivial non-linear loop invariants (in polynomial steps).

References

[1] S. Bensalem, M. Bozga, J.-C. Ghirvu, and L. Lakhnech. A transformation approach for generating non-linear invariants. *Static Analysis Symposium*, 5:101–114, June 2000. 1

[2] B. Buchberger. Symbolic computation: Computer algebra and logic. In *Frontiers of Combining Systems: Proceedings of the 1st Int. Workshop, Munich (Germany)*, pages 193–220, 1996. 2.1

[3] Y. Chen, B. Xia, L. Yang, and N. Zhan. Generating polynomial invariants with discoverer and qepcad. In *Formal Methods and Hybrid Real-Time Systems*, pages 67–82, 2007. 1

[4] G. E. Collins. *Quantifier Elimination for the Elementary Theory of RealClosed Fields by Cylindrical Algebraic Decomposition.* LNCS, 1975. 1

[5] P. Cousot. Proving program invariance and termination by parametric abstraction, lagrangian relaxation and semidefinite programming. In *Sixth Int. Conf. on Verification, Model Checking and Abstract Interpretation (VMCAI'05)*, pages 1–24, Paris, France, LNCS 3385, Jan. 17–19 2005. 1

[6] P. Cousot and R. Cousot. Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Conf. Record of the 4th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 238–252, Los Angeles, California, 1977. ACM Press, NY. 1

[7] P. Cousot and R. Cousot. Abstract interpretation and application to logic programs. *Journal of Logic Programming*, 13(2–3):103–179, 1992. 1, 1

[8] E. W. Dijkstra. *A Discipline of Programming.* Prentince-Hall, 1976. 1

[9] J.-C. Faugere. A new efficient algorithm for computing grobner bases (f4). *Journal of Pure and Applied Algebra*, 139(1-3):61–88, June 1999. 2.1

[10] R. W. Floyd. Assigning meanings to programs. In *Proceedings of the 19th Symphosia in Applied Mathematics*, pages 19–37, 1967. 1

[11] T. Jebelean, L. Kovacs, and N. Popov. Experimental Program Verification in the Theorema System. *Int. Journal on Software Tools for Technology Transfer (STTT)*, 2006. in press. 1

[12] D. Kapur. Automatically generating loop invariants using quantifier elimination. *Proc. IMACS Intl. Conf. on Applications of Computer Algebra*, 2004. 1

[13] L. Kovacs. Reasoning algebraically about p-solvable loops. In *TACAS 2008: Proc. of the 14th Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems*, volume 4963, pages 249–264. LNCS, 2008. 1

[14] L. Kovacs and T. Jebelean. Finding polynomial invariants for imperative loops in the theorema system. In *Proc.of Verify'06 Workshop*, pages 52–67, August 15-16 2006. 1, 1

[15] S. Lang. *Algebra.* Springer, January 2002. 3.3, 4.2

[16] Z. Manna and A. Pnueli. *Temporal Verification of Reactive Systems: Safety.* Springer-Verlag, New York, 1995. 1

[17] M. Mller-Olm and H. Seidl. Polynomial constants are decidable. In *Static Analysis Symposium*, pages 4–19. LNCS, 2002. 1, 1

[18] R. Rebiha, N. Matringe, and A. Vieira-Moura. Non-trivial non-linear loop invariant generation. In *Technical-Report-IC-07-045*, Dec. 2007.

[19] E. Rodríguez-Carbonell and D. Kapur. Automatic generation of polynomial invariants of bounded degree using abstract interpretation. *Sci. Comput. Program.*, 64(1):54–75, 2007. 1, 1

[20] E. Rodríguez-Carbonell and D. Kapur. Generating all polynomial invariants in simple loops. *J. Symb. Comput.*, 42(4):443–476, 2007. 1

[21] S. Bensalem, Y. Lakhnech, and H. Saidi. Powerful techniques for the automatic generation of invariants. In Rajeev Alur and Thomas A. Henzinger, editors, *Proc. of the 8th Int. Conf. on Computer Aided Verification CAV*, volume 1102, pages 323–335, NJ, USA, 1996. 1

[22] S. Sankaranarayanan, H. B. Sipma, and Z. Manna. Non-linear loop invariant generation using grobner bases. In *POPL '04: Proc. of the 31st ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pages 318–329, New York, NY, USA, 2004. ACM Press. 1, 1, 2.2, 3, 3.4.1, 3.5

[23] A. Schrijver. *Theory of Linear and Integer Programming.* Wiley, 1986. 1

[24] V. Weispfenning. Quantifier elimination for real algebra - the quadratic case and beyond. *Applicable Algebra in Engineering, Communication and Computing*, 8(2):85–101, 1997. 1