Polynomial Interrupt Timed Automata

Béatrice Bérard (LIP6), Serge Haddad (LSV)
Claudine Picaronny (LSV), Mohab Safey El Din (LIP6)
Mathieu Sassolas (LACL)

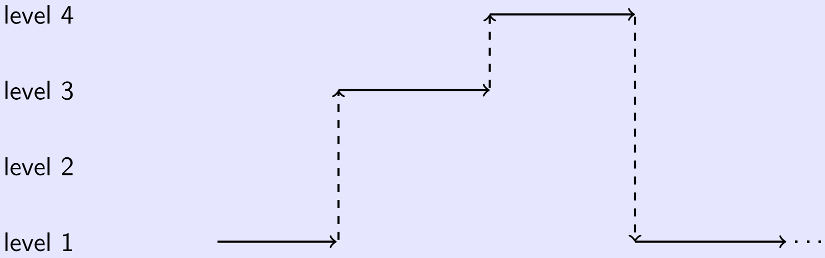
Cachan, December 2nd and 9th 2015
Reachability Problems 2015 (long version arXiv:1504.04541)

Motivations for Interrupt Clocks

- Theoretical: investigate subclasses of hybrid automata with stopwatches, to obtain decidability results in view of negative results, among them:
  - Henzinger et al. 1998: The reachability problem is decidable for rectangular initalized automata, but becomes undecidable for slight extensions, e.g. adding one stopwatch to timed automata.
  - Cassez, Larsen 2000: Linear hybrid automata and automata with stopwatches (and unobservable delays) are equally expressive.
  - Bouyer, Brihaye, Bruyère, Markey, Raskin 2006: Model checking timed automata with stopwatch observers is undecidable for WCTL (a weighted extension of CTL).
- Practical: Many real-time systems include interruptions (as in processors). An interrupt clock can be seen as a restricted type of stopwatch.

Interruptions and Real-Time

Several levels with exactly one active clock at each level

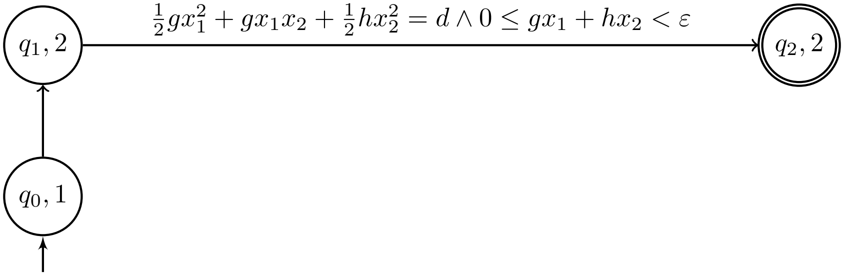


Execution : $\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{1.5} \begin{bmatrix} 1.5 \\ 0 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{2.1} \begin{bmatrix} 1.5 \\ 0 \\ 2.1 \\ 0 \end{bmatrix} \xrightarrow{1.7} \begin{bmatrix} 1.5 \\ 0 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{2.2} \begin{bmatrix} 3.7 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

Motivation for Polynomials

Landing a rocket

- First stage (lasting x_1): from distance d, the rocket approaches the land using gravitation g;
- Second stage (lasting x_2): the rocket approaches the land with constant deceleration $h < 0$;
- Third stage: the rocket must reach the land with small positive speed (less than ε).



For all $g \in [7, 10]$ does there exist an $h \in [-3, -1]$ such that the rocket is landing?

Outline

1. PolITA

Abstraction for Timed Systems

Abstraction for PolITA

Extensions

Conclusion and Perspectives

The Model of PolITA

A short history

- Interrupt Timed Automa (ITA) were introduced in (FOSSACS 2009 Bérard,H) with decision procedures for reachability and expressiveness results.
- The complexity of the decision procedure is improved (NEXPTIME and PTIME with a fixed number of clocks) and model checking is studied in (FMSD 2012 Bérard,H,Sassolas).
- ITA are enlarged with additive and multiplicative parameters while reachability remains decidable (2EXPSPACE and PSPACE with a fixed number of clocks) in (RP 2013 Bérard,H,Jovanovic,Lime).

Polynomial Interrupt Timed Automata (PolITA) in a nutshell

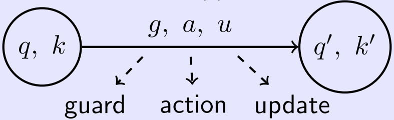
- clocks are ordered along hierarchical levels;
- their flows are restricted to $\dot{x} \in \{0, 1\}$ (stopwatches);
- guards and updates can be polynomials of clocks.

Main results. Reachability (and some quantitative model checking) is decidable in 2EXPTIME and PTIME with a fixed number of clocks.

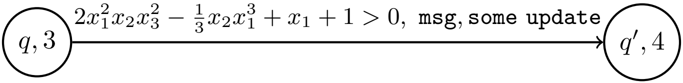
PolITA: Syntax (1)

$\mathcal{A} = (\Sigma, Q, q_0, X, \lambda, \Delta)$

- Σ, an alphabet;
- Q, a finite set of states with initial state q_0;
- $X = \{x_1, \ldots, x_n\}$, a set of clocks with x_k for level k;
- $\lambda : Q \to \{1, \ldots, n\}$ state level, with $x_{\lambda(q)}$ the active clock in state q;
- Transitions in Δ:



- Guards: conjunctions of constraints $P \bowtie 0$ with $\bowtie$ in $\{<, \leq, =, \geq, >\}$ and $P \in \mathbb{Q}[x_1, \ldots, x_k]$ at level k.



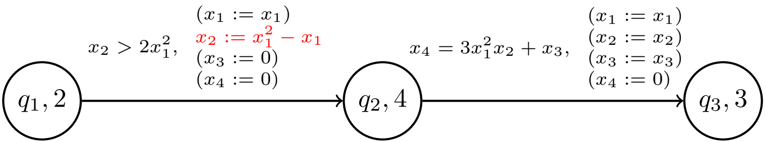
PolITA: Syntax (2)

A transition increasing level k to level $k' \geq k$

- If $i > k$ then x_i is reset;
- If $i < k$ then x_i is unchanged;
- x_k is unchanged or is updated by some $P \in \mathbb{Q}[x_1, \ldots, x_{k-1}]$.

A transition decreasing level k to level $k' < k$

- If $i > k'$ then x_i is reset;
- Otherwise x_i is unchanged.



Semantics of PolITA

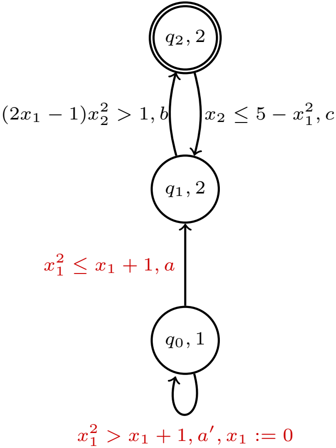
A transition system $\mathcal{T}_{\mathcal{A}} = (S, s_0, \rightarrow)$

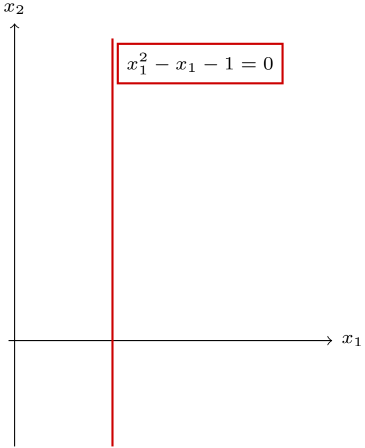
- $S = Q \times \mathbb{R}^n$, a set of configurations where a configuration is a pair (q, v) of a state q and a clock valuation $v = (v(x_1), \ldots, v(x_n))$;
- $s_0 = (q_0, v_0)$, the initial configuration with $v_0 = \mathbf{0} = (0, \ldots, 0) \in \mathbb{R}^n$;
- **Discrete step:** $(q, v) \xrightarrow{e} (q', v')$ for a transition $e : q \xrightarrow{g,a,u} q'$ if v satisfies the guard g and $v' = v[u]$;
- **Time step:** from q at level k: $(q, v) \xrightarrow{d} (q, v +_k d)$, with all clock values in $v +_k d$ unchanged except $(v +_k d)(x_k) = v(x_k) + d$.

An execution alternates time and discrete steps:

$$(q_0, v_0) \xrightarrow{d_0} (q_0, v_0 +_{\lambda(q_0)} d_0) \xrightarrow{e_0} (q_1, v_1) \xrightarrow{d_1} (q_1, v_1 +_{\lambda(q_1)} d_1) \xrightarrow{e_1} \cdots$$

An Execution

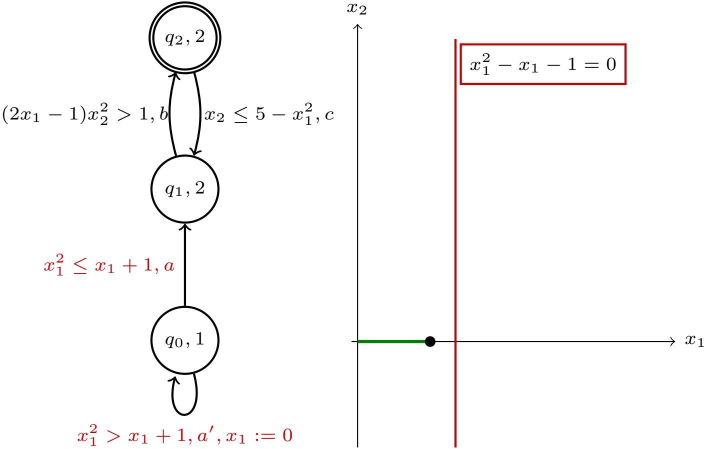




$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1)$
$\xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

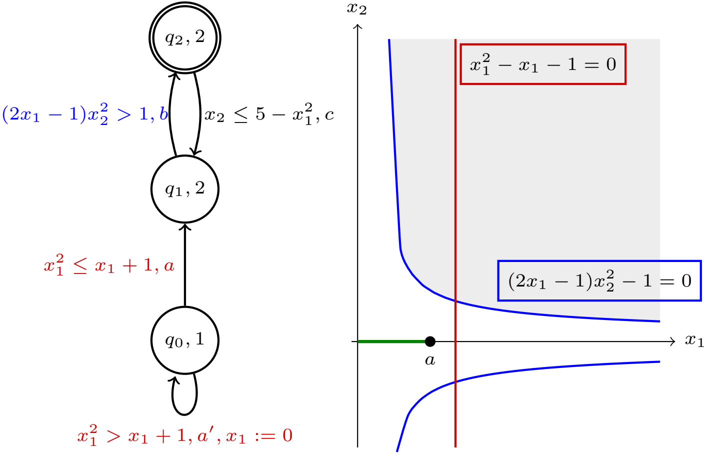
An Execution



$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1)$
$\xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

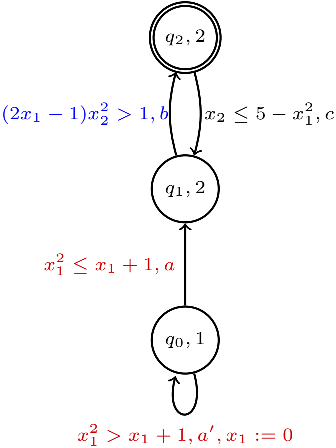
An Execution

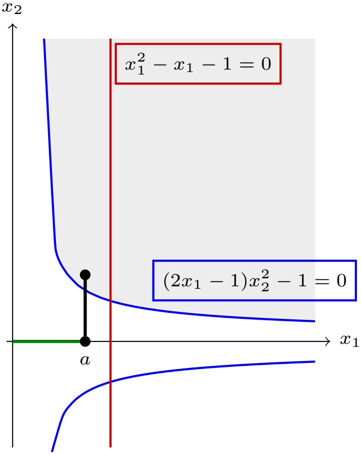


$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1) \xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

An Execution

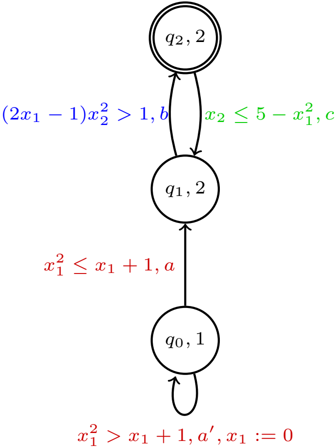


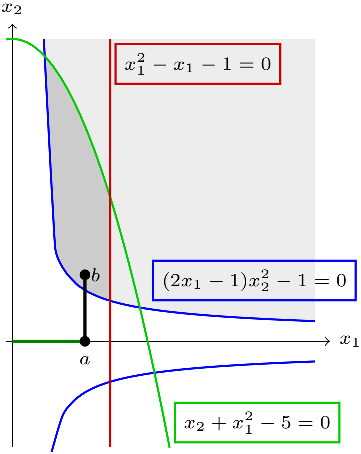


$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1)$
$\xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

An Execution

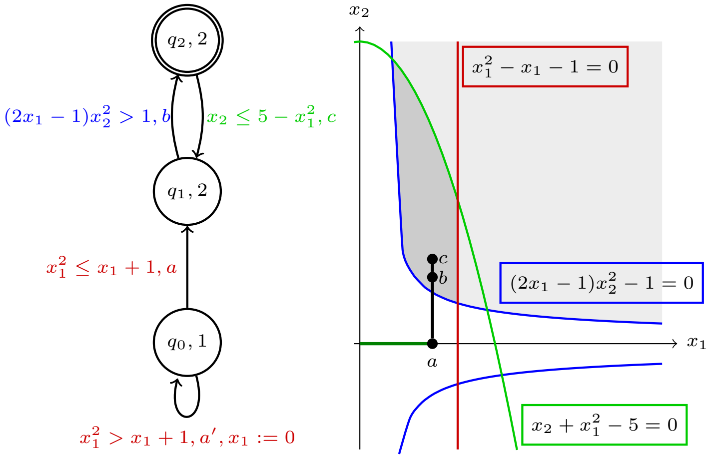




$$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1)$$
$$\xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

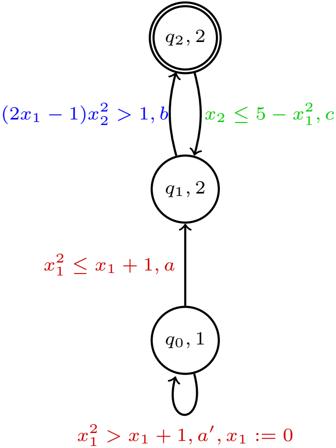
An Execution

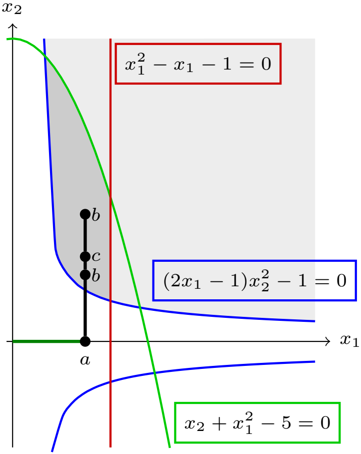


$$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1)$$
$$\xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

An Execution

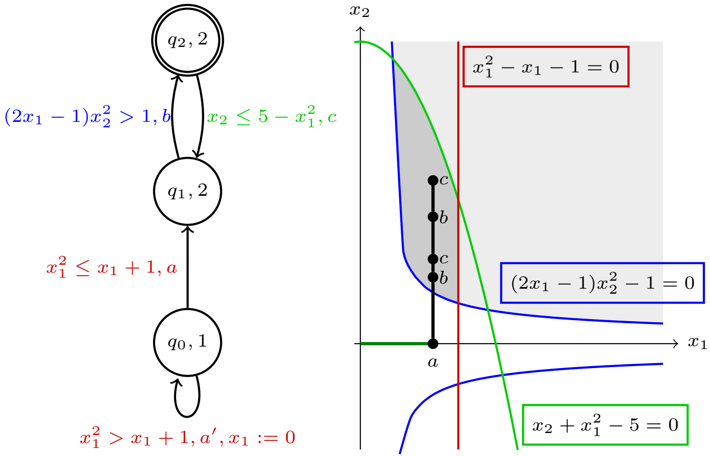




$$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1)$$
$$\xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

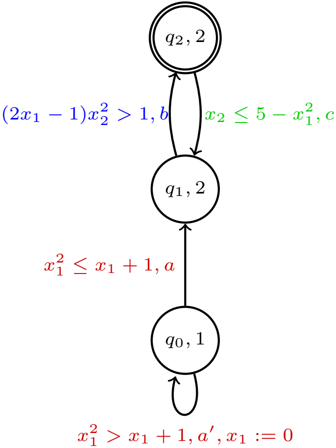
An Execution

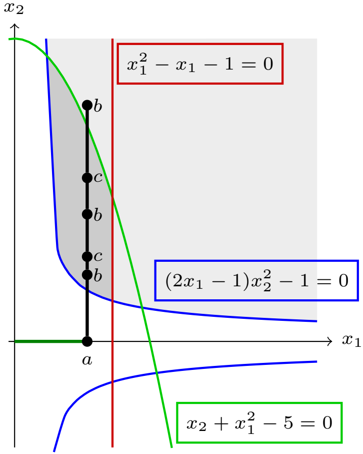


$$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1)$$
$$\xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

An Execution

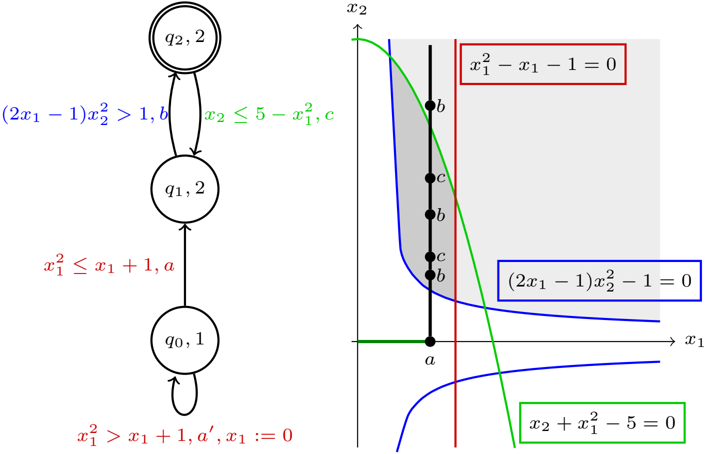




$$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1) \xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

An Execution



$(q_0, 0, 0) \xrightarrow{1.2} (q_0, 1.2, 0) \xrightarrow{a} (q_1, 1.2, 0) \xrightarrow{1.1} (q_1, 1.2, 1.1) \xrightarrow{b} (q_2, 1.2, 1.1)$
$\xrightarrow{0.3} (q_2, 1.2, 1.4) \xrightarrow{c} (q_1, 1.2, 1.4) \cdots$

Blue and green curves meet at real roots of $-2x_1^5 + x_1^4 + 20x_1^3 - 10x_1^2 - 50x_1 + 26$.

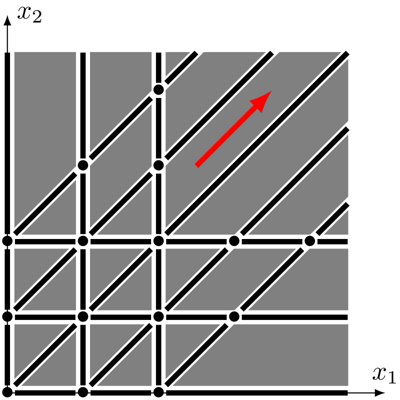
Outline

Reachability Analysis: the Key Idea

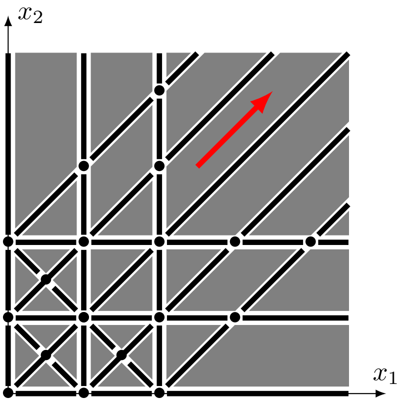
The number of (reachable) configurations is infinite (and even uncountable). So one wants to partition configurations into *regions* such that:

1. Two configurations in a region allow the same transitions and the new configurations belong to the same region.
2. If a configuration in a region letting time elapse reaches a new region every other configuration may reach the same region by time elapsing.
3. There is a finite representation of a region such that the discrete and time successors of the region are computable.
4. The number of regions is finite.

Regions for Timed Automaton (TA)



With ordinary
and diagonal constraints



With ordinary, diagonal
and additive constraints
(only valid for dimension 2)

Building the Region Automaton for TA

The "elimination" stage

- depends on the model;
- extracts the maximal constant.

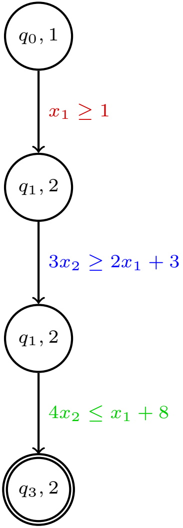
The "lifting" stage

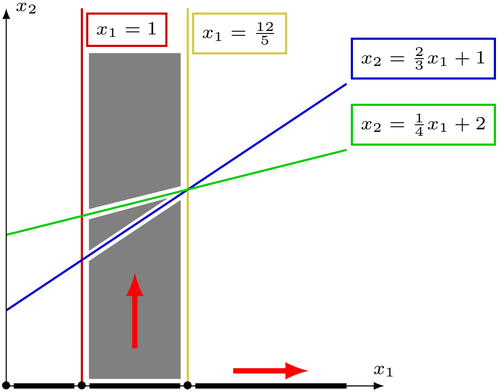
- depends on the class of TA and the elimination;
- partitions clocks space into regions.

The "synchronization" stage

- depends on the model and the lifting;
- performs some product of the model and the partition.

Regions for ITA





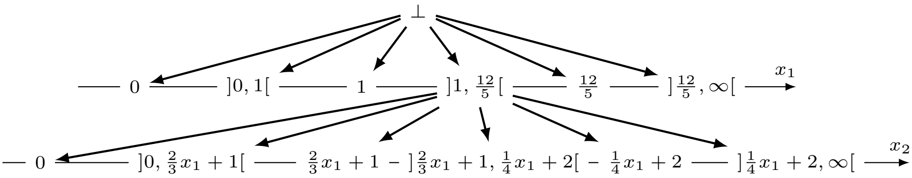
$$x_1 - \frac{12}{5} = \frac{12}{5}\left((\frac{2}{3}x_1 + 1) - (\frac{1}{4}x_1 + 2) \right)$$

Building the Region Automaton for ITA

The elimination stage produces a family of sets $\{E_k\}_{k \le n}$ of linear expressions initialized to $\{0, x_k\}$ by decreasing order:

- adding to E_k at level k all constraints $\sum_{i<k} a_i x_i + b$ that are compared to x_k or 0 at level at least k.
- enlarging this set by applying updates until saturation.
- producing at levels less than k linear expressions by difference between pairs in E_k taking into account possible updates.

The lifting stage produces a tree of total preorders for E_k using the preorder of the ancestors.



The synchronization stage takes into account the level of the current state.

Outline

The Cylindrical Decomposition

The cylindrical decomposition was the first elementary decision method for the first-order theory of reals (2EXPTIME). It proceeds by elimination and lifting.

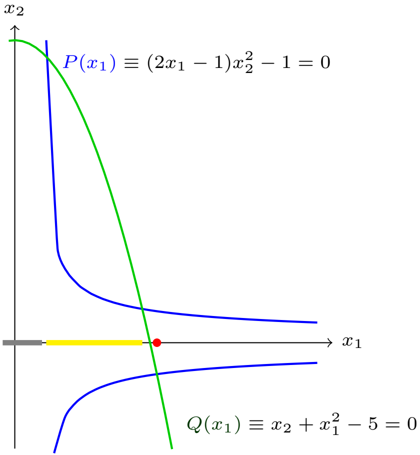
Given a family $\{\mathcal{P}_i\}_{i \leq n}$ where $\mathcal{P}_i$ is a finite subset of $\mathbb{Q}[x_1, \ldots, x_i]$, it builds a tree of *cells* with depth n and root $\perp = \mathbb{R}^0$ fulfilling:

- A cell of depth i is a connected subset of $\mathbb{R}^i$;
- Every cell C of depth less than n has an odd number (say $2k+1$) of children which constitute a partition of $C \times \mathbb{R}$;
- If $k > 0$ then there exist continuous mappings $f_1 < \cdots < f_k$ from C to $\mathbb{R}$ such that the children of C are:
  $\{(x, y) \mid x \in C, y < f_1(x)\}$, $\{(x, f_1(x)) \mid x \in C\}$,
  $\{(x, y) \mid x \in C, f_1(x) < y < f_2(x)\}$, ... , $\{(x, f_k(x)) \mid x \in C\}$,
  $\{(x, y) \mid x \in C, y > f_k(x)\}$;
- All $P \in \mathcal{P}_i$ has a constant sign inside a cell of depth $j \geq i$.

Observation: The construction for ITA is a cylindrical decomposition appropriate for polynomials of degree 1.

Elimination Stage for PolITA (1)

Let $P, Q \in \mathbb{Q}[x_1][x_2]$.



When x_1:

- belongs to the gray interval, P has no root and Q has a single root;
- belongs to the yellow interval, P has two roots and the single root of Q is greater than these roots;
- is the red point, P has two roots and the single root of Q is equal to the smaller root of P.

How to characterize such intervals and points?

Elimination Stage for PolITA (2)

The input. A family of sets $\{\mathcal{Q}_k\}_{k\leq n}$ with $\mathcal{Q}_k \subseteq \mathbb{Q}[x_1,\ldots,x_{k-1}][x_k]$ including x_k and those occurring in guards and updates.

The output. A family of sets $\{\mathcal{P}_k\}_{k\leq n}$ with $\mathcal{Q}_k \subseteq \mathcal{P}_k$ that fulfills a semantical property:

When the sign of all $P \in \mathcal{P}_k$ in a connected set $C \subseteq \mathbb{R}^k$ is constant, then for all z, z' in C and $P, Q \in \mathcal{P}_{k+1}$:

- the number of roots of the polynomials $P(z)$ and $P(z')$ in $\mathbb{R}[x_{k+1}]$ are equal;
- The order between the roots of $PQ(z)$ and $PQ(z')$ is the same.

The key concept. The *subresultant* $sRes(P,Q)$ of two polynomials $P, Q \in \mathbb{D}[X]$ is a $\mathbb{D}$-vector that can be computed by operations in the ring $\mathbb{D}$.

A syntactical sufficient condition. For all $P, Q \in \mathcal{P}_{k+1}$, with possible respective *truncations* R, S, the following polynomials should be in $\mathcal{P}_k$:

- the coefficients of P, Q;
- the items of $sRes(R, R')$ and $sRes(R, S)$.

Subresultants: Definition and Example

$Sres_j(P, Q)$ is the determinant of the matrix:

- whose lines are coefficients of $X^{q-1-j}P, \ldots, P, Q, \ldots, X^{p-1-j}Q$,
- truncated to its first $p + q - 2j$ columns.

Let $P = X^3 + X^2 + \alpha X + \beta$ and $Q = X^2 - 1$.

$$sRes_0(P, Q) = \begin{vmatrix} 1 & 1 & \alpha & \beta & 0 \\ 0 & 1 & 1 & \alpha & \beta \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & -1 & 0 \\ 1 & 0 & -1 & 0 & 0 \end{vmatrix} = \begin{vmatrix} 1 & 1 & \alpha & \beta \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \\ -1 & -1-\alpha & -\beta & 0 \end{vmatrix}$$

$$= \begin{vmatrix} 1 & 0 & -1 \\ -1 & -1-\alpha & -\beta \\ -\alpha & \alpha-\beta & \beta \end{vmatrix} = \begin{vmatrix} -1-\alpha & -1-\beta \\ \alpha-\beta & \beta-\alpha \end{vmatrix} = (\alpha - \beta)(\alpha + \beta + 2)$$

$$sRes_1(P, Q) = \begin{vmatrix} 1 & 1 & \alpha \\ 0 & 1 & 0 \\ 1 & 0 & -1 \end{vmatrix} \begin{matrix} \beta \\ -1 \\ 0 \end{matrix} = -1 - \alpha$$

$sRes_0(P, Q) = 0$ iff $deg(gcd(P, Q)) \geq 1$

$sRes_0(P, Q) = 0 \wedge sRes_1(P, Q) = 0$ iff $deg(gcd(P, Q)) \geq 2$

A Property of Subresultants

$$\boxed{sRes_0(P,Q) = 0 \wedge \cdots \wedge sRes_{j-1}(P,Q) = 0 \text{ iff } deg(gcd(P,Q)) \geq j}$$

Sketch of proof.

$sRes_j(P,Q) = 0$ iff $\exists U, V \neq 0$ with $deg(U) < q - j$, $deg(V) < p - j$ and $deg(UP + VQ) < j$.

- Assume that $deg(gcd(P,Q)) \geq j$.

Then $deg(lcm(P,Q)) \geq p + q - j$.

So there exist $U, V \neq 0$ with $deg(U) \leq q - j$, $deg(V) \leq p - j$ and $UP = -VQ$.

This implies that for $k < j$, $sRes_j(P,Q) = 0$.

- The other direction is established by induction.

Base. $sRes_0(P,Q) = 0$
$\Rightarrow \exists U, V \neq 0$ with $UP + VQ = 0$, $deg(U) < q$ and $deg(V) < p \Rightarrow deg(gcd(P,Q)) \geq 1$.

Induction. $sRes_0(P,Q) = \cdots = sRes_j(P,Q) = 0$
By induction, $sRes_0(P,Q) = \cdots = sRes_{j-1}(P,Q) = 0 \Rightarrow deg(gcd(P,Q)) \geq j$.
$sRes_j(P,Q) = 0 \Rightarrow \exists U, V \neq 0$ with $deg(U) < q - j$, $deg(V) < p - j$ and $deg(UP + VQ) < j$.
$gcd(P,Q) | UP + VQ \Rightarrow UP + VQ = 0 \Rightarrow deg(lcm(P,Q)) < p + q - j \Rightarrow deg(gcd(P,Q)) \geq j + 1$.

Continuity of roots

Let $P \in \mathbb{C}[X_1, \ldots, X_{k-1}][X_k]$, $S \subseteq \mathbb{C}^{k-1}$ with $deg(P(x))$ constant over $x \in S$.
Let $a \in S$ such that $\{z_i\}_{i \leq m}$ are the roots of $P(a)$ with multiplicities $\{\mu_i\}_{i \leq m}$.
Let $0 < r < \min_{i \neq j}(|z_i - z_j|/2)$.

Then there exists an neighborhood U of a such that for $x \in U$,
$P(x)$ has exactly μ_i roots counted with multiplicities in $D(z_i, r)$ for all $i \leq m$.

Proof.

• Let $P = X^\mu$ and $Q = X^\mu - \sum_{i<\mu} b_i X^i$ with $\delta = \max_{i<\mu} |b_i| < \frac{\min(1, r^\mu)}{\mu}$.
Let z be a root of Q, $\delta < \frac{1}{\mu} \Rightarrow |z| < 1$. $z^\mu = \sum_{i<\mu} b_i z^i \Rightarrow |z^\mu| \leq \mu\delta < r^\mu \Rightarrow |z| < r$.

• Let $\varphi(Q, R) = QR$ where Q, R have degree q and r, are monic and coprime.
φ is differentiable with Jacobian $\pm Sres_0(Q, R)$. φ locally admits a differentiable inverse.
So there exist neighborhoods $\mathcal{V}_Q$, $\mathcal{V}_R$ of Q and R, such that:

$\mathcal{V} = \varphi(\mathcal{V}_Q \times \mathcal{V}_R)$ is a neighborhood of QR

• By induction, $P_0 = (X - z_1)^{\mu_1} \cdots (X - z_m)^{\mu_m}$ admits an open neighborhood $\mathcal{V}$ such that for all monic $P_1 \in \mathcal{V}$:

$P_1 = Q_1 \ldots Q_m$ with all Q_i of degree μ_i and whose roots belong to $D(z_i, r)$

• Since the coefficients of P are rational functions of $X_1, \ldots X_{k-1}$ and so continuous, there is an appropriate neighborhood U of a.

Root mappings (1)

Let $P_1, \ldots, P_s \in \mathbb{R}[X_1, \ldots, X_{k-1}][X_k]$, $S \subseteq \mathbb{R}^{k-1}$ connected with for all i, j, $deg(P_i(x)) > -\infty$, $deg(gcd(P_i(x), P_j(x))$, $deg(gcd(P_i(x), P_i'(x))$ are constant.

Then there exist continuous functions $f_1 < \cdots < f_\ell$ from S to $\mathbb{R}$ such that the set of real roots of $\prod_{j \le s} P_j(x)$ is $\{f_1(x), \ldots, f_\ell(x)\}$.

Moreover for all i, j, the multiplicity of $f_i(x)$ for $P_j(x)$ is constant.

Proof.

Let $a \in S$ and $\{z_i(a)\}_{i \le m}$ be roots of $\prod_{j \le s} P_j(a)$ with μ_i^j, multiplicity of $z_i(a)$ for $P_j(a)$.

Let $R_{jk}(a) = gcd(P_j(a), P_k(a))$, $deg(R_{jk}(a)) = \sum_{i \le m} \min(\mu_i^j, \mu_i^k)$ and $\min(\mu_i^j, \mu_i^k)$ is the multiplicity of $z_i(a)$ for $R_{jk}(a)$.

$deg(P_j(x)$ and $deg(gcd(P_j(x), P_j'(x))$ constant implies the number of distinct roots of $P_j(x)$ is constant.

One root of $\prod_{j \le s} P_j(x)$ in the neighborhood of $z_i(a)$

- Pick $r > 0$ such that $D(z_i(a), r)$ are disjoint.
- Let i, j such that $\mu_i^j > 0$, there is a neighborhood U of a such that for all $x \in U$, $D(z_i(a), r)$ contains exactly a root, denoted $z_i^j(x)$, of $P_j(x)$ with multiplicity μ_i^j.
- Assume there exists $k \neq j$ with $\mu_i^k > 0$, since $deg(R_{jk}(x))$ is constant, $z_i^j(x) = z_i^k(x)$ for all $x \in U$. So we can omit the superscript j in $z_i^j(x)$ (defined when $\mu_i^j > 0$).

Root mappings (2)

Proof (continued.)

$z_i(a)$ real if and only if $z_i(x)$ real

- $z_i(a)$ real $\Rightarrow$ $z_i(x)$ real otherwise its conjugate would be another root in $D(z_i(a), r)$.
- $z_i(a)$ complex $\Rightarrow$ $\overline{z_i(a)}$ root.

$D(z_i(a), r)$ and $D(\overline{z_i(a)}, r)$ disjoint implies $z_i(x)$ complex.

The number of real roots of $\prod_{j \leq s} P_j(x)$ is globally constant.

- Hence the number of real roots of $\prod_{j \leq s} P_j(x)$ is constant over $x \in U$.

S is connected implies the number of real roots of $\prod_{j \leq s} P_j(x)$ is constant over S, say ℓ.

The real roots of $\prod_{j \leq s} P_j(x)$ are continuous mappings of x.

- Let $f_i(x)$, for $i \leq \ell$ be the function that associates with x the i^{th} real root of $\prod_{j \leq s} P_j(x)$ in increasing order.

Since r could be chosen arbitrarily small, f_i is continuous.

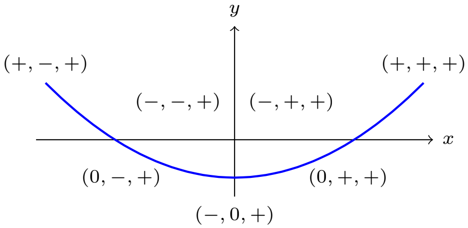
The multiplicity of a real root of $\prod_{j \leq s} P_j(x)$ w.r.t. any $P_j(x)$ is globally constant.

As the multiplicity of $f_i(x)$ w.r.t. any $P_j(x)$ is locally constant, it is constant over $x \in S$.

Thom Encodings (1)

Let $P \in \mathbb{R}[X]$ of degree p and $x \in \mathbb{R}$, the P-code of x is:
$(sign(P(x)), sign(P'(x)), \ldots, sign(P^{(p)}(x)))$ whose basic properties are:

- The values associated with a P-code are either an open interval or a point. *(thus P-codes of roots of P are "identifiers")*
- Given two P-codes, one can decide which corresponding values are bigger.



$(s_p, \ldots, s_0) \prec (s'_p, \ldots, s'_0)$ if there exists i with:

- for all $j < i$, $s_j = s'_j$;
- $(s_{i-1} > 0$ and $s_i < s'_i)$ or $(s_{i-1} < 0$ and $s_i > s'_i)$.

Thom Encodings (2)

Effectiveness properties.

Given $P, Q \in \mathbb{D}[X]$ where $\mathbb{D} \subseteq \mathbb{R}$ is *sign-effective*, one can compute:

- the number of roots of P;
- the Q-encodings of the roots of P.

Thus one can merge and order the roots of P and Q.

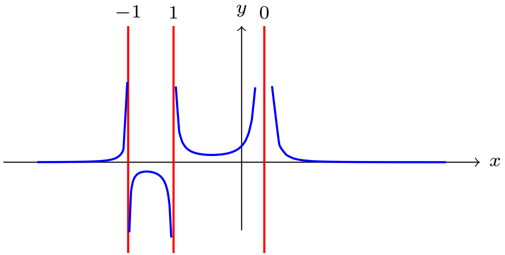
Cauchy Index

Let $P, Q \in \mathbb{D}[X]$. Then the *Cauchy index* of Q/P is defined by:

$$Ind(Q/P) = \frac{1}{2} \sum_{z \in Pole(Q/P)} sign((Q/P)(z^+)) - sign((Q/P)(z^-))$$

where $sign((Q/P)(z^+))$ and $sign((Q/P)(z^-)))$ denote respectively the sign of the rational function Q/P at the right and at the left of z.

Let $Q/P = \frac{1}{(X+2.5)(X+1.5)(X-0.5)^2}$. Then $Ind(Q/P) = 0$.



Tarsky Query

Let $P, Q \in \mathbb{D}[X]$. Then the *Tarsky query* of (Q, P) is defined by:

$$TaQ(Q,P) = \sum_{z \in Zer(P)} sign(Q(z)).$$

$\boxed{\text{Let } P, Q \in \mathbb{D}[X]. \text{ Then: } TaQ(Q,P) = Ind(P'Q/P)}$

Proof.

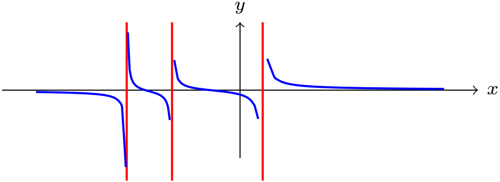
Let z be a root of P with multiplicity μ.

Then $P'Q/P = Q(\frac{\mu}{X-z} + R)$ with R a rational function with no pole at z.

If $Q(z) = 0$ then $P'Q/P$ has no pole in z.

Otherwise $sign((P'Q/P)(z^+)) = sign(Q(z))$ and $sign((P'Q/P)(z^-)) = -sign(Q(z))$.

Let $P = (X+2.5)(X+1.5)(X-0.5)^2$. Then $TaQ(P', P) = 3$.



Root Counters

Let $P, Q \in \mathbb{D}[X]$. Then:

- $\mathbf{nb}_P(Q)[-1] = |\{z \in Zer(P) \mid Q(z) < 0\}|$;
- $\mathbf{nb}_P(Q)[0] = |\{z \in Zer(P) \mid Q(z) = 0\}|$.
- $\mathbf{nb}_P(Q)[1] = |\{z \in Zer(P) \mid Q(z) > 0\}|$;

The Tarski queries and root counters are related by:

- $TaQ(1, P) = \mathbf{nb}_P(Q)[-1] + \mathbf{nb}_P(Q)[0] + \mathbf{nb}_P(Q)[1]$;
- $TaQ(Q, P) = -\mathbf{nb}_P(Q)[-1] + \mathbf{nb}_P(Q)[1]$;
- $TaQ(Q^2, P) = \mathbf{nb}_P(Q)[-1] + \mathbf{nb}_P(Q)[1]$.

$$\begin{pmatrix} TaQ(1,P) \\ TaQ(Q,P) \\ TaQ(Q^2,P) \end{pmatrix} = \mathbf{M}_1 \begin{pmatrix} \mathbf{nb}_P(Q)[-1] \\ \mathbf{nb}_P(Q)[0] \\ \mathbf{nb}_P(Q)[1] \end{pmatrix} \text{ with } \mathbf{M}_1 = \begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

Generalized Counters and Tarski Queries

Let $P \in \mathbb{D}[X]$ and $\mathcal{Q} = (Q_1, \ldots, Q_m)$ be a finite sequence of $\mathbb{D}[X]$. Then:
$\mathbf{nb}_P(\mathcal{Q})$ is an integer vector whose support is $\{-1, 0, 1\}^{\{1,\ldots,m\}}$ such that:

$$\mathbf{nb}_P(\mathcal{Q})[i_1, \ldots, i_m] = |\{z \in Zer(P) \mid \forall j \leq m \; sign(Q_j(z)) = i_j\}|$$

$\mathbf{TaQ}_P(\mathcal{Q})$ is an integer vector whose support is $\{0, 1, 2\}^{\{1,\ldots,m\}}$ such that:

$$\mathbf{TaQ}_P(\mathcal{Q})[i_1, \ldots, i_m] = TaQ(Q_1^{i_1} \cdots Q_m^{i_m})$$

$$\boxed{\mathbf{TaQ}_P(\mathcal{Q}) = \mathbf{M}_m \cdot \mathbf{nb}_P(\mathcal{Q}) \text{ where } \mathbf{M}_m = \mathbf{M}_1 \otimes \cdots \otimes \mathbf{M}_1}$$

A useful application.
The Q-code of the roots of P is simply deduced from $\mathbf{nb}_P(Q, Q', \ldots, Q^{(q)})$.

Computing the Cauchy Index

Let $s = (s_p, \ldots, s_0)$ be a list of reals such that $s_p \neq 0$. Define s' as the shortest list such that $s = (s_p, 0, \ldots, 0) \cdot s'$. Then we inductively define:

$$PmV(s) = \begin{cases} 0 & \text{if } s' = \emptyset \\ PmV(s') + \varepsilon_{p-q} sign(s_p s_q) & \text{if } s' = (s_q, \ldots, s_0) \text{ and } p - q \text{ is odd} \\ PmV(s') & \text{otherwise} \end{cases}$$

where $\varepsilon_i = (-1)^{\frac{i(i-1)}{2}}$

Let $P, Q \in \mathbb{D}[X]$ with $p = deg(P) > q = deg(Q)$. Then:

$$PmV(sRes(P, Q)) = Ind(Q/P)$$

Once again the subresultants!

Triangular Systems

A *triangular system* $(n_i, P_i)_{i \leq k}$ with $n_i \in \mathbb{N}$ and $P_i \in \mathbb{Q}[x_1, \ldots, x_i]$ represents the algebraic point α in $\mathbb{R}^k$ if:

- α_1 is the n_1^{th} root of $P_1 \in \mathbb{Q}[x_1]$;
- for all $i < k$, α_{i+1} is the n_{i+1}^{th} root of $P_{i+1}(\alpha_1, \ldots, \alpha_i) \in \mathbb{Q}[\alpha_1, \ldots, \alpha_i][x_{i+1}]$.

Effectiveness properties.

- One can decide whether $(n_i, P_i)_{i \leq k}$ is a triangular system;
- One can decide the sign of an item of $\mathbb{Q}[\alpha_1, \ldots, \alpha_k]$ when α is given by a triangular system;
- or equivalently the sign of $P(\alpha)$ for $P \in \mathbb{Q}[x_1, \ldots, x_k]$.

Lifting Stage for PolITA: General Overview

The tree is built top-down as follows.

Due to the invariance property of cells w.r.t. the sign of polynomials, a cell is represented by a sample point given by a triangular system.

Let C be a cell at depth $k < n$ represented by $(\alpha_1, \ldots, \alpha_k)$. In order to find its children in $\mathbb{Q}[\alpha_1, \ldots, \alpha_k]$,

- one determines the (number of) roots of the polynomials in $\mathcal{P}_{k+1}$;
- one globally orders them;
- an interval $](i, P), (j, Q)[$ is represented by an appropriate root of $(PQ)'$;
- an interval $]-\infty, (1, P)[$ (resp. $](i, P), \infty[$) is represented by $(1, P[x_{k+1} \leftarrow x_{k+1} + 1])$ (resp. $(i, P[x_{k+1} \leftarrow x_{k+1} - 1])$);
- The triangular system associated with the children is the original one extended by the root corresponding to the children.

Synchronization Stage for PolITA

The synchronization can be done on-the-fly during the lifing stage.

This may produce considerable time and space savings.

However the elimination stage is already doubly exponential.

Outline

Decidability and Extensions

The PolITA model may be extended while reachability remains decidable.

- Parameters may be modelled by low level additional clocks.
- A set of auxiliary clocks may be added per level with restrictions.
  This model is strictly more expressive than the original one.
- Updates of clocks with a lower level than the current one may be allowed.
- A PolITA may be "synchonized" with a TA:
  the PolITA interrupts the TA.

Outline

Conclusion and Perspectives

Summary of results

- Another model of hybrid systems (extending TA) with parameters.
- Decidability of the reachability and quantitative model checking problems.

Perspectives

- Experimentations since a prototype already exists.
  Thanks to Rémi Garnier and Mathieu Huot, L3 students of ENS Cachan!
- Adapting more efficient methods for first-order theory of reals
  to (subclasses of) PoliTA.
- Extensions of expressions in o-minimal decidable theories.