Comprehensive Triangular Decomposition

C. Chen, O. Golubitsky,
F. Lemaire, M. Moreno Maza, W. Pan

September 18, 2007

Triangular decompositions of parametric systems: which parameter values give finitely many solutions?

Let $F = \{f_1, f_2\} \subset \mathbb{Q}[u, v][x > y]$ with $f_1 = x^2 + y^2 - 1$ and $f_2 = uy - vx$. A triangular decomposition of $\mathbf{V}(F)$ is given by:

$$T_1 = \begin{cases} uy - vx \\ (u^2 + v^2)x^2 - u^2 \end{cases}, T_2 = \begin{cases} y - 1 \\ x \\ u \end{cases},$$

$$T_3 = \begin{cases} y + 1 \\ x \\ u \end{cases} \text{ and } T_4 = \begin{cases} y^2 + x^2 - 1 \\ u \\ v \end{cases}.$$

meanwhile $\mathbf{V}(F \cup \{u^2 + v^2\}) = \mathbf{W}(T_4)$. So the "minimal discriminant set" (minimal set of the "bad guys") is $\mathbf{V}(u^2 + v^2)$. Then, there are two good "cells" (or "'regions") in the solution space:

- $u \neq 0$ leading the solution $(x^2, y) = (\frac{u^2}{u^2+v^2}, \frac{vx}{u})$,
- $u = 0$ leading the solution $(x, y^2) = (0, 1)$.

Triangular decompositions of parametric systems: when do regular chains specialize to regular chains?

For $F = \{-(y+1)x + s, -(x+1)y + s\} \subset \mathbb{Q}[s][x > y]$, a triangular decomposition of $\mathbf{V}(F)$ w.r.t. $s < y < x$ is:

$$T_1 = \begin{cases} x+1 \\ y+1 \\ s \end{cases}, \quad T_2 = \begin{cases} (y+1)x - s \\ y(y+1) - s \end{cases}.$$

For some parameter values s and a regular chain T, the specialized triangular set $T(s)$ may not be a regular chain: for $s = 0$, T_2 specializes to triangular set

$$\begin{cases} (y+1)x \\ y(y+1) \end{cases},$$

where the initial of the first polynomial divides the second.

Objectives

For $F \subset \mathbb{K}[U][X]$, the following problems are of interest:

• Compute the values u of the parameters for which $F(u)$ has solutions, or has finitely many solutions.

• Compute the solutions of F as functions of the parameters.

• Provide an automatic case analysis for the number of solutions depending on the parameter values.

Related work

These questions have been approached by various techniques including comprehensive Gröbner bases (CGB), cylindrical algebraic decompositions (CAD), triangular decompositions (TD):

• CGB or GB only: (V. Weispfenning, 1992), (V. Weispfenning, 2002), (A. Montes, 2002), (D. Lazard & F. Rouillier, 2004), (A. Suzuki & Y. Sato, 2006), (Y. Kurata & M. Noro, 2007), (K. Nabeshima, 2007) and others.

• TD: (W.T. Wu, 1987), (S.C. Chou & X.S. Gao, 1991), (S.C. Chou & X.S. Gao, 1992), (T. Gómez Díaz, 1992), (D.M. Wang, 1998), (D.M. Wang, 2000), (L. Yang, X.R. Hou & B. Xia, 2000), (M. Moreno Maza, 2000) $=$ *Triade algorithm* $\subseteq$ `RegularChains` library.

CTD: Main idea

In broad words:

• this is a finite partition of the parameter space into regions, so that

• above each region C the "geometry" (number of irreducible components together with their dimensions and degrees) of $\mathbf{V}(F(u))$ is the same for all values $u \in C$.

On the first example, we have four cells:

- $u^2 + v^2 \neq 0, u \neq 0$ leading the solution $(x^2, y) = (\frac{u^2}{u^2+v^2}, \frac{vx}{u})$,
- $u^2 + v^2 \neq 0, u = 0$ leading the solution $(x, y^2) = (0, 1)$,
- $u^2 + v^2 = 0, u \neq 0$, leading to no solutions,
- $u = v = 0$, leading to infinitely many solutions.

Regular chain

Let $Y = Y_1 < \cdots < Y_n$ be ordered variables and $\overline{\mathbb{K}}$ the algebraic closure of base field $\mathbb{K}$.
Let $T = f_1, \ldots, f_s$ be a triangular set in $\mathbb{K}[Y]$, with main variables $Y_{\ell_1} < \cdots < Y_{\ell_s}$.
For $1 \leq i \leq s$, the initial h_i is the lead. coeff. of f_i in Y_{ℓ_i}.
For $1 \leq i \leq s$, the rank $\mathrm{rank}(f_i)$ is the lead. monomial of f_i in Y_{ℓ_i}.
The saturated ideal is $\mathrm{Sat}(T) = (f_1, \ldots, f_s) : (h_1 \ldots h_s)^\infty$.
T is a regular chain if h_i is regular mod $\mathrm{Sat}(f_1, \ldots, f_{i-1})$ for all $i \geq 2$.
The quasi-component $\mathbf{W}(T) := \mathbf{V}(T) \setminus \mathbf{V}(h_1 \cdots h_s)$ satisfies $\overline{\mathbf{W}(T)} = \mathbf{V}(\mathrm{Sat}(T))$.
The algebraic variables are those which appear as main variables. The other ones are free.

Example

$$\left| \begin{array}{l} f_2 = (Y_1 + Y_2){Y_3}^2 + Y_3 + 1 \\ f_1 = {Y_1}^2 + 1. \end{array} \right. , \text{ with} \left| \begin{array}{l} \mathrm{mvar}(f_2) = Y_3 \\ \mathrm{mvar}(f_1) = Y_1 \end{array} \right. .$$

CTD: Definition

Let $U = U_1, \ldots, U_d$ be parameters, $X = X_1, \ldots, X_m$ variables, Π_U the projection from $\overline{\mathbb{K}}^{m+d}$ to the parameter space $\overline{\mathbb{K}}^d$.

Definition

A regular chain T *specializes well* at $u \in \overline{\mathbb{K}}^d$ if $T(u)$ is a regular chain in $\overline{\mathbb{K}}[X]$ and such that $\boxed{\mathrm{rank}(T(u)) = \mathrm{rank}(T_{>U_d}).}$

Definition

Let $F \subset \mathbb{K}[U, X]$ be a finite polynomial set. A *comprehensive triangular decomposition* of $\mathbf{V}(F)$ is given by:

- a finite partition $\mathcal{C}$ of $\Pi_U(\mathbf{V}(F))$,
- for each $C \in \mathcal{C}$ a set of regular chains $\mathcal{T}_C$ of $\mathbb{K}[U, X]$ such that for $u \in C$:
  - all regular chains $T \in \mathcal{T}_C$ specializes well at u and,
  - we have $\boxed{\mathbf{V}(F(u)) = \bigcup_{T \in \mathcal{T}_C} \mathbf{W}(T(u)).}$

CTD: Outline

- Iterated resultant
- Triade operations (implemented in `RegularChains`)
- The defining set of a regular chain
- PCTD: definition and algorithm
- Regular system and constructible set
- The Difference algorithm
- The coprime factorization for constructible sets
- The algorithm to compute CTD
- Experimentation

Iterated resultant

Definition

Let $p \in \mathbb{K}[Y]$ and $T \subset \mathbb{K}[Y]$ be a triangular set. The *iterated resultant* of p w.r.t. T, denoted by $\mathrm{res}(p, T)$, is defined below:

- if $p \in \mathbb{K}$ or all variables in p are free w.r.t. T, then

$$\mathrm{res}(p, T) = p,$$

- otherwise, if v is the largest variable of p which is algebraic w.r.t. T, then

$$\mathrm{res}(p, T) = \mathrm{res}(\mathrm{res}(p, T_v), T_{<v}).$$

Lemma

The triangular set T is a regular chain iff $\mathrm{res}(h_T, T) \neq 0$.

Triade operations

Let $p \in \mathbb{K}[Y]$ and let T be a regular chain.

- **Regularize**(p, T) returns regular chains $T_1, \ldots, T_e$ such that
  - either $e = 1$ and $T = T_1$,
  - or $e > 1$, $\mathrm{rank}(T_i) < \mathrm{rank}(T)$ for all $i = 1 \ldots e$ and

$$\mathbf{W}(T) \subseteq \bigcup_{i=1}^{e} \mathbf{W}(T_i) \subseteq \overline{\mathbf{W}(T)},$$

  - for all $1 \leq i \leq e$ the polynomial p is either 0 or regular modulo $\mathrm{Sat}(T_i)$.

- If a set of polynomials F such that $F \not\subset \mathrm{Sat}(T)$ then **Triangularize**(F, T) returns regular chains $T_1, \ldots, T_e$ such that:

$$\mathbf{V}(F) \cap \mathbf{W}(T) \;\subseteq\; \mathbf{W}(T_1) \cup \cdots \cup \mathbf{W}(T_e) \;\subseteq\; \mathbf{V}(F) \cap \overline{\mathbf{W}(T)}.$$

- When the above F consists of a single polynomial p then **Triangularize**(F, T) writes **Intersect**(p, T).

Defining set of a regular chain

Notation

Let $p \in \mathbb{K}[U][X]$ be a polynomial, $F \subset \mathbb{K}[U][X]$ be a polynomial set and $T \subset \mathbb{K}[U][X]$ be a regular chain.

- $\mathbf{V}^U(p) \subset \overline{\mathbb{K}}^d$ *is the common roots of the coefficients of p as a polynomial in X.*
- *if $T \subset \mathbb{K}[U]$ holds then $\mathbf{W}^U(T)$ is its quasi-variety in $\overline{\mathbb{K}}^d$.*

Definition

The *defining set* of T *w.r.t.* U is given by:

$$\mathbf{D}^U(T) = \mathbf{W}^U(T \cap \mathbb{K}[U]) \setminus \mathbf{V}^U(\mathrm{res}(h_{T_{>U_d}}, T_{>U_d})).$$

Proposition

Let $T \subset \mathbb{K}[U, X]$ be a regular chain and let $u \in \mathbf{W}^U(T \cap \mathbb{K}[U])$. We have: T specializes well at $u \in \overline{\mathbb{K}}^d$ if and only if $u \in \mathbf{D}^U(T)$.

PCTD: Algorithm

For a regular chain $T \subset \mathbb{K}[U, X]$ we define:

$$\mathbf{W}_C(T) = \mathbf{W}(T) \cap \Pi_U^{-1}(\mathbf{D}^U(T)).$$

Definition

A triangular decomposition $\mathcal{T}$ of $\mathbf{V}(F)$ is a called a *pre-comprehensive triangular decomposition* (PCTD) of $\mathbf{V}(F)$ if we have

$$\mathbf{V}(F) = \bigcup_{T \in \mathcal{T}} \mathbf{W}_C(T).$$

- **input**: $F \subset \mathbb{K}[U, X]$.
- **output**: A PCTD of $\mathbf{V}(F)$.

1. $\mathcal{T} \leftarrow$ **Triangularize**(F)
2. **while** $\mathcal{T} \neq \emptyset$ **repeat**
   - 2.1 let $T \in \mathcal{T}$, $\mathcal{T} \leftarrow \mathcal{T} \setminus \{T\}$
   - 2.2 **Output** T
   - 2.3 $G \leftarrow \textsc{Coefficients}(\mathrm{res}(h_{T_{>U_d}}, T_{>U_d}), U)$
   - 2.4 $\mathcal{T} \leftarrow \mathcal{T} \cup$ **Triangularize**(G, T)

Regular system: definition and notations

Definition (Regular System)

A pair $[T, h]$ is a regular system if T is a regular chain, and $h \in \mathbb{K}[Y]$ is regular w.r.t $\mathrm{Sat}(T)$. we write
$\boxed{\mathbf{Z}(T, h) := W(T) \setminus V(h).}$

Lemma

Let p and h be polynomials and T a regular chain. Assume that the product of initials h_T of T divides h. Then there exists an operation **Intersect**(p, T, h) *returning a set of regular chains $\{T_1, \ldots, T_e\}$ such that*

(*i*) *h is regular w.r.t $\mathrm{Sat}(T_i)$ for all i;*

(*ii*) $V(p) \cap \mathbf{Z}(T, h) = \bigcup_{i=1}^{e} \mathbf{Z}(T_i, h)$.

Constructible sets

Definition (Constructible set)

A constructible subset of $\overline{\mathbb{K}}^n$ is any finite union

$$(A_1 \setminus B_1) \cup \cdots \cup (A_e \setminus B_e)$$

where $A_1, \ldots, A_e, B_1, \ldots, B_e$ are algebraic varieties over $\mathbb{K}$.

Theorem

Every constructible set can be represented by a finite set of regular systems.

Remark

The conclusion comes from applying the operation **Triangularize** *and the algorithm* **DifferenceLR**.

Specification of "Difference"

Algorithm 1 **Difference**$([T, h], [T', h'])$

Input $[T, h]$, $[T', h']$ two regular systems.

Output Regular systems $\{[T_i, h_i] \mid i = 1 \ldots e\}$ such that

$$\mathbf{Z}(T, h) \setminus \mathbf{Z}(T', h') = \bigcup_{i=1}^{e} \mathbf{Z}(T_i, h_i),$$

and $\mathrm{rank}(T_i) \leqslant_r \mathrm{rank}(T)$.

Algorithm 2 **DifferenceLR**$(\mathcal{L}, \mathcal{R})$

Input $\mathcal{L} := \{[L_i, f_i] \mid i = 1 \ldots r\}$ and $\mathcal{R} := \{[R_j, g_j] \mid j = 1 \ldots s\}$ two lists of regular systems.

Output Regular systems $\mathcal{S} := \{[T_i, h_i] \mid i = 1 \ldots e\}$ such that

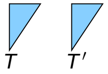
$$\left(\bigcup_{i=1}^{r} \mathbf{Z}(L_i, f_i)\right) \setminus \left(\bigcup_{j=1}^{s} \mathbf{Z}(R_j, g_j)\right) = \bigcup_{i=1}^{e} \mathbf{Z}(T_i, h_i),$$

with $\mathrm{rank}(\mathcal{S}) \leqslant_r \mathrm{rank}(\mathcal{L})$.

“Difference”: Efficient algorithm

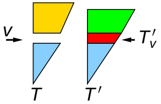
- Computing $Z(T,h) \setminus Z(T',h')$ with $h_T = h_{T'} = h = h' = 1$ by exploiting the triangular structure level by level.

Case 1:



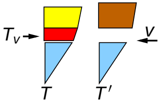
$\langle T \rangle = \langle T' \rangle$, Easy!

Case 2:



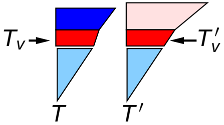
Output $[T, T'_v]$;
Output `Difference` $(\mathbf{V}(T'_v) \cap \mathbf{V}(T), T')$;

Case 3:



Output `Difference` $(T, \mathbf{V}(T_v) \cap \mathbf{V}(T'))$;

Case 4:



- $g :=$ GCD$(T_v, T'_v, T_{<v})$;
- $g \in \mathbb{K} \Rightarrow$ Output $[T, 1]$;
- $\mathrm{mvar}(g) < v \Rightarrow$ { Output $[T, g]$;
  Output `Difference`$(\mathbf{V}(g) \cap \mathbf{V}(T), T')$ };
- Output `Difference`$(T_{<v} \cup \{g\} \cup T_{>v}, T')$;
- Output `Difference`$(T_{<v} \cup \{T_v / g\} \cup T_{>v}, T')$;

MakePairwiseDisjoint (MPD)

We assume that **DifferenceLR**$(\mathcal{L}, \mathcal{R})$ returns a list of regular systems sorted by increasing rank.

- **input**: $\mathcal{S}$ be a list of regular systems sorted by increasing rank.
- **algorithm**:
  - If $|\mathcal{S}| < 2$ then **MPD**$(\mathcal{S}) := \mathcal{S}$.
  - Let $\mathcal{L}, \mathcal{R}$ be list of regular systems such that

$$\mathcal{S} = \mathcal{L} + \mathcal{R} \ \text{ and } \ (|\mathcal{L}| = |\mathcal{R}| \ \text{ or } \ |\mathcal{L}| = |\mathcal{R}| + 1)$$

then **MPD**$(\mathcal{S}) =$ **MPD**(**DifferenceLR**$(\mathcal{L}, \mathcal{R})) +$ **MPD**$(\mathcal{R})$.

Proposition

For all distinct regular systems $D, D' \in \mathcal{D} =$ **MPD**$(\mathcal{S})$*, we have* $\mathbf{Z}(D) \cap \mathbf{Z}(D') = \varnothing$*, and*

$$\bigcup_{S \in \mathcal{S}} \mathbf{Z}(S) = \bigcup_{D \in \mathcal{D}} \mathbf{Z}(D).$$

SymmetricMakePairwiseDisjoint (SMPD)

- **algorithm**:
  - If $|\mathcal{S}| < 2$ then **return** $\mathcal{S}$.
  - Let $\mathcal{L}, \mathcal{R}$ such that $\mathcal{R} = [T, h]$ and $\mathcal{S} = \mathcal{L} + \mathcal{R}$.
  - Let $\mathcal{L} := \textbf{SMPD}(\mathcal{L})$.
  - **output** $\textbf{MPD}(\textbf{DifferenceLR}(\mathcal{R}, \mathcal{L}))$
  - **output** $\textbf{MPD}(\textbf{DifferenceLR}(\mathcal{L}, \mathcal{R}))$
  - **output** $\bigcup_{[T',h'] \in \mathcal{L}} \textbf{MPD}(\textbf{DifferenceLR}([T', h'], \textbf{Difference}([T', h'], \mathcal{R})))$

Proposition

Let $\mathcal{D} = \textbf{SMPD}(\mathcal{S})$, *we then have*

- *The output of* **SMPD** *satisfies the property of* **MPD**.
- *Moreover, for any two distinct regular systems* S, S' *in* $\mathcal{S}$, *the sets* $\mathbf{Z}(S) \setminus \mathbf{Z}(S')$, $\mathbf{Z}(S) \cap \mathbf{Z}(S')$ *and* $\mathbf{Z}(S') \setminus \mathbf{Z}(S)$ *can be represented by some regular systems in D respectively.*

CTD: Algorithm

- **input**: $F \subset \mathbb{K}[U, X]$.
- **output**: A CTD of $\mathbf{V}(F)$.

1. Compute a PCTD $\mathcal{T}_0$ of F.
2. For each $T \in \mathcal{T}_0$, compute $D^U(T)$ as a set of regular systems $\mathcal{R}^U(T)$.
3. Using **SMPD**, refine $\cup_{T \in \mathcal{T}_0} \mathcal{R}^U(T)$ into disjoint cells, each of which represented by a set of regular systems, obtaining a partition $\mathcal{C}$ of $\Pi_U(\mathbf{V}(F))$,
4. Each cell $C \in \mathcal{C}$ is associated with a subset $\mathcal{T}_C$ of $\mathcal{T}_0$.
5. Return all pairs $(C, \mathcal{T}_C)$.

Remark

- *The third step can be seen as a* set theoretical instance *of the coprime factorization problem.*
- *the PCTD step computes* $\cup_{C \in \mathcal{C}} \cup_{T \in \mathcal{T}_C} \mathbf{W}_C(T)$
- *the* **SMPD** *step creates the partition* $\mathcal{C}$*.*

Minimal discriminant set

Definition

The *discriminant set of F* is defined as the set of all points $u \in \overline{\mathbb{K}}^d$ for which $\mathbf{V}(F(u))$ is empty or infinite.

Theorem

If $\mathcal{T}$ is a pre-comprehensive triangular decomposition of $\mathbf{V}(F)$, then the following is the discriminant set of F:

$$\left(\bigcup_{\substack{T \in \mathcal{T} \\ X \not\subseteq \mathrm{mvar}(T)}} \mathbf{D}^U(T) \right) \cup \left(\bigcap_{\substack{T \in \mathcal{T} \\ X \subseteq \mathrm{mvar}(T)}} \overline{\mathbb{K}}^d \setminus \mathbf{D}^U(T) \right).$$

Example (1/3)

Example

Let $F = \{vxy + ux^2 + x, uy^2 + x^2\}$ be a parametric polynomial system with parameters $u > v$ and unknowns $x > y$. Then a comprehensive triangular decomposition of $\mathbf{V}(F)$ is:

$$\begin{array}{rl} C_1 = \{u(u^3 + v^2) \neq 0\} : & \mathcal{T}_{C_1} = \{T_3, T_4\} \\ C_2 = \{u = 0\} : & \mathcal{T}_{C_2} = \{T_2, T_3\} \\ C_3 = \{u^3 + v^2 = 0, v \neq 0\} : & \mathcal{T}_{C_3} = \{T_1, T_3\} \end{array}$$

where
$$\begin{aligned} T_1 &= \{vxy + x - u^2y^2, 2vy + 1, u^3 + v^2\} \\ T_2 &= \{x, u\} \\ T_3 &= \{x, y\} \\ T_4 &= \{vxy + x - u^2y^2, u^3y^2 + v^2y^2 + 2vy + 1\} \end{aligned}$$

Here , C_1, C_2, C_3 is a partition of $\Pi_U(\mathbf{V}(F))$ and $\mathcal{T}_{C_i}$ is a triangular decomposition of $\mathbf{V}(F)$ above C_i.

Example (2/3)

Example

By `RegSer` (D.M. Wang, 2000), $\mathbf{V}(F)$ can be decomposed into a set of regular systems:

$$R_1 = \left\{ \begin{array}{rcl} ux + vy + 1 & = & 0 \\ (u^3 + v^2)y^2 + 2vy + 1 & = & 0 \\ u(u^3 + v^2) & \neq & 0 \end{array} \right. , \quad R_2 = \left\{ \begin{array}{rcl} x & = & 0 \\ y & = & 0 \\ u & \neq & 0 \end{array} \right. ,$$

$$R_3 = \left\{ \begin{array}{rcl} x & = & 0 \\ vy + 1 & = & 0 \\ u & = & 0 \\ v & \neq & 0 \end{array} \right. , \quad R_4 = \left\{ \begin{array}{rcl} 2ux + 1 & = & 0 \\ 2vy + 1 & = & 0 \\ u^3 + v^2 & = & 0 \\ v & \neq & 0 \end{array} \right. , \quad R_5 = \left\{ \begin{array}{rcl} x & = & 0 \\ u & = & 0 \end{array} \right. .$$

For each regular system, one can directly read its dimension when parameters take corresponding values. However, the dimension of the input system could not be obtained immediately, since there is not a partition of the parameter space.

Example (3/3)

Example

By DISPGB (Montes02), one can obtain all the cases over the parameters leading to different reduced Gröbner bases with parameters:

$$
\begin{array}{rl}
u(u^3+v^2)\neq 0: & \{ux+(u^3v+v^3)y^3+(-u^3+v^2)y^2, \\
 & (u^3+v^2)y^4+2vy^3+y^2\} \\
 & \\
u(u^3+v^2)=0, u\neq 0: & \{ux+2v^2y^2, 2vy^3+y^2\} \\
 & \\
u=0, v\neq 0: & \{x^2, vxy+x\} \\
 & \\
u=0, v=0: & \{x\}
\end{array}
$$

Here for each parameter value, the input system specializes into a Gröbner basis. Since Gröbner bases do not necessarily have a triangular shape, the “geometry” may not be read directly either. For example, when $u=0, v\neq 0$, $\{x^2, vxy+x\}$ is not a triangular set.

Benchmark (1)

We provide comparative benchmarks with MAPLE implementations of related methods:

- *decomposition into regular systems* by Wang and,
- *discussing parametric Gröbner bases* by Montes.

The corresponding MAPLE functions are *RegSer* and *DISPGB*, respectively.

Note that the specifications of these three methods are different:

- The output of `CTD` and *DISPGB* depends on the choice of the parameter set.
- *RegSer* does not require to specify parameters.
- `CTD` computes a comprehensive triangular decomposition, and thus a family of triangular decompositions with a partition of the parameter space,
- *DISPGB* computes a family of comprehensive Gröbner bases with a partition of the parameter space,
- *RegSer* computes a triangular decomposition.

Benchmark (1)

Sys	Name	`Triangularize`	`PCTD`	`SMPD`	`CTD`	#Cells
10	MontesS10	5.325	0.684	1.138	7.147	10
11	MontesS11	0.757	0.208	12.302	13.267	28
12	MontesS12	14.199	2.419	10.114	26.732	10
13	MontesS13	0.415	0.143	1.268	1.826	9
14	MontesS14	41.167	31.510	0.303	72.980	4
15	MontesS15	6.919	0.579	1.123	8.621	5
16	MontesS16	6.963	0.083	2.407	9.453	21
17	AlkashiSinus	0.716	0.191	0.574	1.481	6
18	Bronstein	2.526	0.017	0.548	3.091	6
19	Gerdt	3.863	0.006	0.733	4.602	5
20	Hereman-2	1.826	0.019	0.020	1.865	2
21	Lanconelli	2.056	0.336	3.430	5.822	14
22	genLinSyst-3-2	1.624	0.275	25.413	27.312	32
23	genLinSyst-3-3	9.571	1.824	1097.291	1108.686	116
24	Wang93	6.795	37.232	11.828	55.855	8
25	Maclane	12.955	0.403	54.197	67.555	21
26	Neural	15.279	19.313	0.530	35.122	4
27	Leykin-1	1261.751	86.460	27.180	1375.391	57
28	Lazard-ascm	60.698	2817.801	–	–	–
29	Pavelle	–	–	–	–	–

Table 1 Solving timings and number of cells of `CTD`

Benchmark (2)

	DISPGB		RegSer		CTD	
Sys	Time (s)	# Cells	Time (s)	# Components	Time (s)	# Cells
10	9.659	5	0.329	5	7.147	10
11	0.489	3	0.260	9	13.267	28
12	259.730	5	2.381	23	26.732	10
13	5.830	9	0.199	9	1.826	9
14	–	–	–	–	72.980	4
15	30.470	7	0.640	10	8.621	5
16	61.831	7	6.060	22	9.453	21
17	4.619	6	0.150	5	1.481	6
18	8.791	5	0.319	6	3.091	6
19	20.739	5	3.019	10	4.602	5
20	101.251	2	0.371	7	1.865	2
21	43.441	4	0.330	7	5.822	14
22	–	–	0.350	18	27.312	32
23	–	–	2.031	61	1108.686	116
24	–	–	4.040	6	55.855	8
25	83.210	11	–	–	67.555	21
26	–	–	–	–	35.122	4
27	–	–	–	–	1375.391	57
28	–	–	–	–	–	–
29	–	–	–	–	–	–

Table 2 Solving timings and number of components/cells in three algorithms

Conclusions

- CTD is a new tool for the analysis of parametric polynomial systems: its purpose is to partition the parameter space into regions, so that within each region the "geometry" of the algebraic variety of the specialized system is the same for all values of the parameters.
- As the main technical tool, we proposed an algorithm that represents the difference of two constructible sets as finite unions of regular systems.
- We have reported on an implementation of our algorithm computing CTDs, based on the `RegularChains` library in MAPLE. Our comparative benchmarks, with MAPLE implementations of related methods for solving parametric polynomial systems, illustrate the good performances of our CTD code.

Thanks!