

**Canonical discussion of polynomial
systems with parameters**

ANTONIO MONTES

MA2-IR-06-00006

Caninical discussion of polynomial systems with parameters *

Antonio Montes

Departament de Matemàtica Aplicada 2,
Universitat Politècnica de Catalunya, Spain.

e-mail: antonio.montes@upc.edu
<http://www-ma2.upc.edu/~montes>

April 2006

Abstract

Given a parametric polynomial ideal I , the algorithm DISPGB, introduced by the author in 2002, builds up a binary tree describing a dichotomic discussion of the different reduced Gröbner bases depending on the values of the parameters. An improvement using a discriminant ideal to rewrite the tree was described by Manubens and the author in 2005. In this paper we describe how to iterate the use of discriminants to rebuild the tree and show that this leads to an ascending discriminant chain of ideals, and to the corresponding descending chain of varieties in the parameter space that characterizes the different kinds of solutions. This provides a discussion of the ideal I into canonical cases even if the tree is not completely canonical. It can also be used to obtain a canonical comprehensive Gröbner basis. With the new algorithm, we realize in some sense the objective proposed by Weispfenning in 1992. We also prove the conjectures formulated in the previous paper.

Keywords: minimal singular variety, canonical discussion, perfect specification, ascending discriminant chain, discriminant ideal, comprehensive Gröbner basis, parametric polynomial system.

MSC: 68W30, 13P10, 13F10.

1 Introduction

There are many authors [Be94, BeWe93, De99, Du95, FoGiTr01, Gi87, Gom02, GoRe93, GoTrZa00, GoTrZa05, HeMcKa97, Ka97, Kap95, MaMo05, Mo02, Mor97, Pe94, SaSu03, SaSuNa03, Sc91, Si92, We92, We03] that have studied the problem of specializing parametric ideals into a field and determining the specialized Gröbner bases. Many other authors [Co04, Em99, GoRe93, GuOr04, Mo95, Mo98, Ry00] have applied some of these methods to solve

*Work partially supported by the Ministerio de Ciencia y Tecnología under project BFM2003-00368, and by the Generalitat de Catalunya under project 2005 SGR 00692

concrete problems. In the previous paper [Mo02] we give more details of their contributions to the field. In the following we only refer to the papers directly related to the present work.

Let $I \subset K[\bar{a}][\bar{x}]$ be a parametric ideal in the variables $\bar{x} = x_1, \dots, x_n$ and the parameters $\bar{a} = a_1, \dots, a_m$, and $\succ_{\bar{x}}$ and $\succ_{\bar{a}}$ monomial orders in variables and parameters respectively. Weispfenning [We92] proved the existence of a Comprehensive Gröbner Basis (CGB) of I and gave an algorithm for computing it. Let K be a computable field (for example \mathbb{Q}) and K' an algebraically closed extension (for example \mathbb{C}). A CGB is a basis of I that specializes to a Gröbner basis of $\sigma_{\bar{a}}(I)$ for any specialization $\sigma_{\bar{a}} : K[\bar{a}][\bar{x}] \rightarrow K'[\bar{x}]$, that substitutes the parameters by concrete values $\bar{a} \in K'$. Nevertheless Weispfenning's algorithm was neither very efficient nor canonical. Using his suggestions, the author [Mo02] obtained a more efficient algorithm (DISPGB) for Discussing Parametric Gröbner Bases. DISPGB builds up a dichotomic binary tree, whose branches at each vertex correspond to the annihilation or not of a polynomial in $K[\bar{a}]$. It places at each vertex v a specification $\Sigma_v = (N_v, W_v)$ of the included specializations, that summarizes the null and non-null decisions taken before reaching v , and a specialized basis B_v of $\sigma_{\bar{a}}(I)$ for the specializations $\sigma_{\bar{a}} \in \Sigma_v$. At the terminal vertices the basis specializes to the Gröbner basis of the specialized ideal $\sigma_{\bar{a}}(I)$ that has a set of leading power products (lpp set) which is common to all specializations in Σ_v .

Inspired by DISPGB, Weispfenning [We03] was able to give a constructive method for obtaining a Canonical Comprehensive Gröbner Basis (CCGB) for parametric polynomial ideals. The algorithm is based on the direct computation of a discriminant ideal. Nevertheless his method is of high computational complexity and does not provide a true partition of the different cases of the parametric Gröbner system. Using Weispfenning's idea of discriminant ideal, Manubens and Montes drastically improved DISPGB. In [MaMo05] we showed that the tree T_{-1} , built up by DISPGB algorithm, can be rewritten into a new form T providing a more compact and effective discussion by computing a discriminant ideal that is easy to compute from T_{-1} . We proved that our discriminant contains Weispfenning discriminant ideal and we conjectured the equality. A redefinition of the discriminant allows now to prove the conjecture.

The old DISPGB is renamed BUILDTREE as it builds up the first discussion tree T_{-1} , and a new algorithm REBUILDTREE is added to rewrite the tree, and both are included in the new DISPGB algorithm outlined in Table 1.

The main contribution of this paper is the iteration of the rewrite process to deeper levels. Some theoretical problems remained open for this. In this paper all these problems are solved, proving that the iteration of REBUILDTREE to each level produces an *ascending chain of discriminant ideals*

$$J_{-1} = \{0\} \subsetneq J_0 \subsetneq J_1 \subsetneq \dots \subsetneq J_k \subsetneq K[\bar{a}] = J_{k+1}$$

and the corresponding *descending chain of minimal varieties*

$$K^m \supseteq \mathbb{V}(J_0) \supseteq \mathbb{V}(J_1) \supseteq \dots \supseteq \mathbb{V}(J_k) \supseteq \emptyset$$

in the parameter space K^m . A descending chain determines a partition

$$\Upsilon = \{Y_0, Y_1, \dots, Y_k\}$$

$T \leftarrow \mathbf{DISPGB}(B, \succ_{\bar{x}}, \succ_{\bar{a}})$ Input: $B \subseteq R[\bar{a}][\bar{x}]$: basis of I , $\succ_{\bar{x}}, \succ_{\bar{a}}$: term orders wrt the variables \bar{x} and the parameters \bar{a} respectively. Output: T : table with binary tree structure, containing (G_v, Σ_v) at vertex v BEGIN $T_{-1} := \mathbf{BUILDTREE}(B, \prec_{\bar{x}}, \prec_{\bar{a}})$ $T := \mathbf{REBUILDTREE}(T_{-1}, \phi, 0)$ END
--

Table 1: DISPGB algorithm.

of K^m into the sets $Y_i = \mathbb{V}(J_{i-1}) \setminus \mathbb{V}(J_i) \subset K^m$ for which the basis $G_i \subset K[\bar{a}][\bar{x}]$ provided by the algorithm has a characteristic lpp set ℓ_i and specializes to the reduced Gröbner basis of $\sigma_{\bar{a}}(I)$ for $\bar{a} \in Y_i$. The algorithm is able to represent the perfect specifications Y_i of the different cases in a canonical form, leading to a complete canonical discussion and a true partition.

We show that for a given ideal and monomial order there exists only a few set of possible descending chains of varieties even if there does not always exist a unique one. Even so, the representation of the cases can be described by a canonical representation independent from its situation in the tree (or chain).

The partition of K^m is almost independent of the particular descending chain. It is unique and canonical whenever the elements of the partition correspond to different cases that cannot be summarized into a single one (for example whenever they have all different lpp sets). In some discussions it happens that some cases could be summarized into a single one. Even if the actual implemented algorithm does not make this automatically, it is always possible to do it externally from the output of the tree. Frequently then this can also be realized using a different strategy with DISPGB.

A first implementation of the algorithm based on the definition of special cases given in [MaMo05] was distributed on the web. Using it, a counter-example has been provided to me by Wibmer [Wi06] that showed that the definition of special cases given in [MaMo05] was insufficient and give rise to erroneous discriminant ideal in some problems. In this paper we have corrected this point and now the singular cases include all the cases to which the generic Gröbner basis does not specialize to a Gröbner basis of the case, even if both basis have the same lpp sets. With this modification the discussion becomes correct and we can also prove the equality of Weispfenning and our top discriminant ideals. The algorithm works with ideals, as these are the algebraic objects that allow a Gröbner representation. Nevertheless, as it does not always use radical ideals, ideals do not represent varieties in a unique form. So, to prove our results we adopt frequently a geometrical view about the specifications of specializations considering the corresponding varieties. This leads to slightly different definitions and properties than those given in [MaMo05].

The *canonical Gröbner system* so obtained allows to compute also a *canonical compre-*

hensive Gröbner basis completing the objective of Weispfenning in [We92].

Manubens and Montes have implemented the new algorithm in *Maple* in release 4.1 of DPGB¹.

2 Reviewing BUILDTREE

BUILDTREE is described in [MaMo05] and improves the original DISPGB described in [Mo02]. So, it will not be described hereafter.

Nevertheless, we need to replace the concept of *reduced specification of specializations*² $\Sigma = (N, W)$ used inside BUILDTREE. Remember that N represents the null conditions, and W the non-null conditions. To attain now the geometrical perspective, the definition given in [MaMo05] must be substituted by the following:

Definition 1. $\Sigma = (N, W)$ is a *reduced specification of specializations* if

- (i) $N = \text{gb}(\langle N \rangle, \succ_{\bar{a}})$,
- (ii) W is a set of distinct irreducible polynomials over K ,
- (iii) No irreducible component of $\mathbb{V}(N) \subset K^m$ is contained in $\bigcup_{w \in W} \mathbb{V}(w)$.

We call X_{Σ} the semi-algebraic set specified by Σ :

$$X_{\Sigma} = \mathbb{V}(N) \setminus \left(\bigcup_{w \in W} \mathbb{V}(w) \right) \in K^m.$$

Lemma 2. *Let K be an infinite field and V, W be varieties of K^n . Then, if V is irreducible and $W \not\subseteq V$ then $\overline{V \setminus W} = V$.*

Proof.

\subseteq : $\overline{V \setminus W}$ is the minimal variety containing $V \setminus W$ and V is a variety containing $V \setminus W$. Thus $\overline{V \setminus W} \subseteq V$.

\supseteq : We have $V = W \cup (V \setminus W) \subseteq W \cup \overline{V \setminus W}$. As V is irreducible then either $V \subseteq W$ or $V \subseteq \overline{V \setminus W}$. But $V \subseteq W$ contradicts the hypothesis, thus $V \subseteq \overline{V \setminus W}$. □

Using the definition of reduced specification we can now prove the following

Theorem 3. *Let $\Sigma = (N, W)$ be a reduced specification of specializations, $Z = \bigcup_{w \in W} \mathbb{V}(w)$ and $X_{\Sigma} = \mathbb{V}(N) \setminus Z$. Then the Zariski closure of X is $\overline{X} = \mathbb{V}(N)$.*

¹Available on the web

²Definition 7 in [MaMo05]

Proof. Decompose $\mathbb{V}(N)$ into irreducibles $\mathbb{V}(N) = \bigcup_{i=1}^k V_i$. We have

$$\overline{\mathbb{V}(N) \setminus Z} = \overline{\bigcup_{i=1}^k V_i \setminus Z} = \bigcup_{i=1}^k \overline{V_i \setminus Z} = \bigcup_{i=1}^k \overline{V_i \setminus (Z \cap V_i)}$$

If no $V_i \subseteq Z$ then $Z \cap V_i \subsetneq V_i$ and thus, by Lemma 2, $\overline{V_i \setminus (Z \cap V_i)} = V_i$ and finally $\overline{\mathbb{V}(N) \setminus Z} = \bigcup_{i=1}^k V_i = V$. \square

Note that in contrast to the definition in [MaMo05], we do not require N to be radical. When we need to test whether a polynomial in $K[\bar{a}]$ vanishes as a consequence of $\bar{a} \in X$, it will not be sufficient to divide it by N , but we will need to test if it belongs to $\sqrt{\langle N \rangle}$. But this is simpler than computing the radical. Note also that properties (ii), (iii) and (iv) of the definition in [MaMo05] are simple consequences of Definition 1. Nevertheless property (iii) of Definition 1 is stronger, and REDSPEC (denoted CANSPEC in previous papers) must be designed to realize this new definition of reduced specification. If property (iii) is not held by the concrete used REDSPEC algorithm, then Theorem 3 will not be true, and the algorithm DISPGB will not always built up the best tree. We shall discuss this later.

We assume now that the algorithm BUILDTREE uses the adequate REDSPEC algorithm and, when applied to the ideal I , builds up a rooted binary tree with the following properties:

1. At each vertex v a dichotomic decision is taken about the vanishing or not of some polynomial $p(\bar{a}) \in K[\bar{a}]$.
2. A vertex is labelled by a list of zeroes and ones, whose root label is empty. At the null son vertex $p(\bar{a})$ is assumed null and a zero is appended to the father's label, whereas $p(\bar{a})$ is assumed non-null at the non-null son vertex, in which a 1 is appended to father's label.
3. At each vertex v , the tree stores Σ_v and B_v , where
 - $\Sigma_v = (N_v, W_v)$ is a reduced specification of specializations summarizing all the decisions taken in the preceding vertices starting from the root. By Definition 1 it represents all the specializations with $\bar{a} \in X_v = \mathbb{V}(N_v) \setminus \bigcup_{w \in W_v} \mathbb{V}(w)$.
 - B_v is reduced wrt Σ_v (not faithful in the sense of Weispfenning) and specializes to a basis of $\sigma_{\bar{a}}(I)$ for every $\bar{a} \in X_v$.
4. At the terminal vertices v ,
 - B_v specializes to the reduced Gröbner basis of $\sigma_{\bar{a}}(I)$ for every $\bar{a} \in X_v$ and has the same lpp set. We will denote such a basis GRGB(Σ_v) (Generic Reduced Gröbner Basis relative to Σ_v).
 - The specifications of the set of terminal vertices t_i represent subsets $X_{t_i} \subset K^m$ forming a partition of the whole parameter space K^m :

$$\mathcal{X} = \{X_{t_0}, X_{t_1}, \dots, X_{t_p}\},$$

and the sets X_{t_i} have characteristic lpp sets that do not depend on the algorithm.

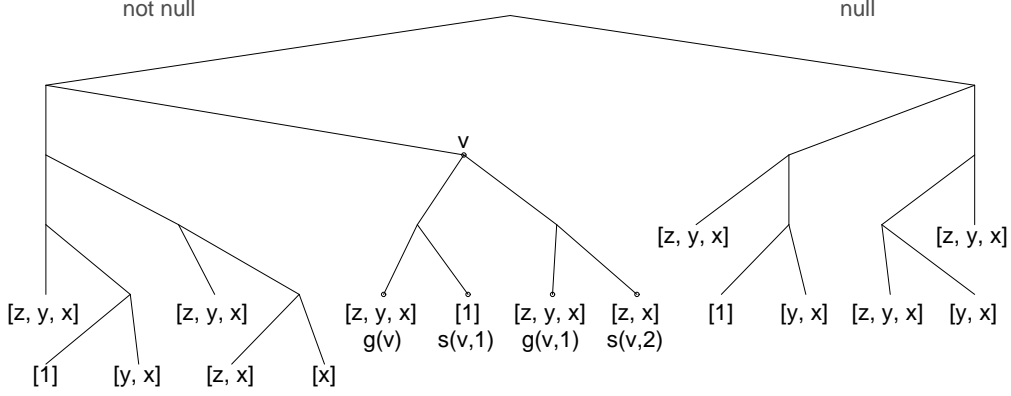


Figure 1: Vertices associated to a vertex v in BUILDTREE. Illustration of Definitions 4 in an example: $I = \langle x + cy + bz + a, cx + y + az + b, bx + ay + z + c \rangle$.

Definition 4. Let $T_{-1}(I, \succ_{\bar{x}}, \succ_{\bar{a}})$ be the tree built up by BUILDTREE and v be any vertex. Associated to v define

- (i) The *generic vertex relative to v* as the terminal vertex $g(v)$ descendent from v for which only non-null decisions have been taken, and correspondingly the *generic basis, specification and set of lpp's relative to v* .
- (ii) The *singular vertices relatives to v* as the terminal vertices $s_i(v)$ descendent from v whose lpp sets are different from that of vertex $g(v)$, as well as those terminal vertices having the same lpp set as $B_{g(v)} = \text{GRGB}(\Sigma_v)$ but for which the polynomials of $B_{g(v)}$, normalized to be polynomials of the specialized ideal, do not specialize to the polynomials of its own basis $B_{s_i(v)}$. These last vertices were not considered singular in [MaMo05] producing some erroneous discussions as has been noticed by Wibmer [Wi06], but must necessarily be considered so. The rest of terminal vertices $g_i(v)$ descendent from v whose lpp sets are equal to that of vertex $g(v)$ are *non-singular vertices*.
- (iii) The *minimal singular variety V_v relative to v* as the minimal variety containing all the singular specifications $X_{s_i(v)}$:

$$V_v = \overline{\bigcup_i X_{s_i(v)}} = \overline{\bigcup_i \left(\mathbb{V}(N_{s_i(v)}) \setminus \bigcup_{w \in W_{s_i(v)}} \mathbb{V}(w) \right)}.$$

- (iv) The *absolute generic vertex and basis* as the generic vertex and $\text{GRGB}(\Sigma_{[\]})$ relative to the root vertex.
- (v) The *discriminant ideal J_v relative to vertex v* by

$$J_v = \bigcap_i N_{s_i(v)},$$

where $N_{s_i(v)}$ are the null condition ideals of the specifications corresponding to the singular vertices relative to vertex v .

Note that the absolute generic basis is equal to the reduced Gröbner basis of I computed in $K(\bar{a})[\bar{x}]$ by the ordinary Buchberger algorithm, conveniently normalized to eliminate denominators.

In Fig. 1 a simple example of the tree built by BUILDTREE is shown. In the figure the lpp sets of the terminal vertices, a vertex v with label $[1, 0]$ and its associated generic $g(v)$, singular $s_1(v), s_2(v)$ and non-singular $g_1(v)$ vertices are illustrated.

Lemma 5. $N_v \subset J_v$.

Proof. By construction, the null condition ideal at any descendant vertex of v contains N_v , as some null condition has been added to reach it. The unique descendant terminal vertex having a null condition ideal equal to N_v is the generic vertex $g(v)$. All the singular terminal vertices have null-condition ideals that strictly contain N_v . Thus their intersection also contains N_v . \square

Theorem 6. Let $T_{-1}(I, \succ_{\bar{x}}, \succ_{\bar{a}})$ be the tree built up by BUILDTREE and v be any vertex. Then

- (i) the minimal singular variety V_v is $V_v = \mathbb{V}(J_v)$.
- (ii) $\sqrt{J_v} = \bigcap \{ \ker(\sigma_{\bar{a}}) : \sigma_{\bar{a}} \text{ is singular relative to vertex } v \}$

Proof. (i) Set $Z_i = \bigcup_{w \in W_{s_i(v)}} \mathbb{V}(w)$. By Theorem 3,

$$\begin{aligned} V_v &= \overline{\bigcup_i X_{s_i(v)}} = \overline{\bigcup_i (\mathbb{V}(N_{s_i(v)}) \setminus Z_i)} = \bigcup_i \left(\overline{\mathbb{V}(N_{s_i(v)}) \setminus Z_i} \right) \\ &= \bigcup_i \mathbb{V}(N_{s_i(v)}) = \mathbb{V}\left(\bigcap_i N_{s_i(v)}\right) = \mathbb{V}(J_v). \end{aligned}$$

- (ii) As K' is algebraically closed, $\mathbb{I}(\mathbb{V}(J_v)) = \sqrt{J_v}$ by the Nullstellensatz. Now $f \in \sqrt{J_v} = \bigcap_i \sqrt{N_{s_i(v)}}$ if and only if for all $\bar{a} \in \bigcup_i X_{\Sigma_{s_i(v)}}$ is $\sigma_{\bar{a}}(f) = 0$, which is equivalent to $f \in \bigcap \{ \ker(\sigma_{\bar{a}}) : \sigma_{\bar{a}} \text{ is singular relative to vertex } v \}$. \square

Note that if REDSPEC does not produce exactly a reduced specification, then Theorem 6 does not apply, and $\mathbb{V}(J_v)$ will include the minimal singular variety V_v but can be strictly greater. In this case unnecessary non-singular cases can remain inside the variety $\mathbb{V}(J_v)$.

In [MaMo05] we enunciate two forms of a conjecture that, with the new definition of special cases, becomes now a theorem. We proof it now.

Remember the definitions leading to the top Weispfenning's discriminant. Let G be the generic Gröbner basis of the ideal I computed in $K(\bar{a})[\bar{x}]$. Associated to each $g \in G$, let d_g be the lcm of the denominators of g , and define the ideal

$$J_g = \{a \in K[\bar{a}] : ag \in I\} = d_g(I : d_g g) \bigcap K[\bar{x}].$$

Then Weispfenning's discriminant [We03] is the radical of the intersection $J_W = \sqrt{\bigcap_{g \in G} J_g}$. A specialization is said to be essential (for $I, \succ_{\bar{x}}$) if $J_g \subseteq \ker(\sigma)$ for some $g \in G$.

Theorem 7. *Consider the root vertex and its associated absolute generic case. Then*

- (i) *A specialization is essential if and only if it is singular.*
- (ii) *The radical of our top discriminant ideal is equal to Weispfenning's top discriminant.*

Proof.

- (i) If σ is essential, then it exists $g \in G$ such that $J_g \subseteq \ker(\sigma)$. Thus $\sigma(f) = 0$ for all $f \in J_g$. Let $h_g g$ be the polynomial in I that corresponds to g . Thus $h_g = \text{lc}(h_g g)$. Then $h_g \in J_g \subseteq \ker(\sigma)$ and $\sigma(h_g) = 0$. Thus g does not specialize to a polynomial with the same lpp wrt $\succ_{\bar{x}}$. Thus G does not specialize to a basis with the same lpp set. Thus σ is special as either $\sigma(G)$ is a Gröbner basis of $\sigma(I)$ and then it does not have the same lpp set as G , either it does not specialize to the reduced Gröbner basis of $\sigma(I)$. In both cases σ is singular. The argument can be reversed to prove that singular specializations are essential. This proves the strong formulation of the conjecture in [MaMo05].
- (ii) This constitutes the weak formulation of the conjecture and is a consequence of the strong formulation as proved in [MaMo05].

□

3 Iterative Rewriting of the Tree Using Discriminant Ideals

The design of the algorithms described in [MaMo05] uses radical null ideals as well as a radical discriminant ideal for the first rewrite of the initial tree $T_{-1}(I, \succ_{\bar{x}}, \succ_{\bar{a}})$. The actual design does not use necessary radicals, but instead the reduced specifications verify Theorem 3.

In order to rewrite the tree, we need a new kind of specification of specializations:

Definition 8. $\Sigma' = (N, M)$ is a *perfect specification of specializations* if

- (i) $N = \text{gb}(\langle N \rangle, \succ_{\bar{a}})$,
- (ii) $M = \text{gb}(\langle M \rangle, \succ_{\bar{a}})$,
- (iii) $\langle N \rangle \subsetneq \langle M \rangle$.

We call $Y_{\Sigma'}$ the algebraic set $\mathbb{V}(N) \setminus \mathbb{V}(M) \subset K^m$ specified by Σ' . The set $Y_{\Sigma'}$ corresponding to a perfect specification can be described in a canonical form using the algorithm CANPERFSPEC of Table 2 as states the following

Theorem 9. *Given two ideals $N \subsetneq M$, the algorithm CANPERFSPEC given in Table 2 generates a unique canonical description of*

$$Y_{\Sigma'} = \mathbb{V}(N) \setminus \mathbb{V}(M) \subset K^m$$

<p> $S \leftarrow \mathbf{PRIMEDECOMP}(N)$ Input: N: ideal (representing a variety) Output: S: The set of prime ideals of the decomposition of \sqrt{N} $(S, T) \leftarrow \mathbf{CANPERFSPEC}(N, M)$ Input: N, the null-condition ideal of the perfect specification M, the non-null condition ideal of the perfect specification $M \supset N$ Output: S, T, The sets of prime ideals corresponding to the minimal null and non-null conditions of the canonical description. BEGIN $M := M + N$ $S := \mathbf{PRIMEDECOMP}(N)$ $T := \mathbf{PRIMEDECOMP}(M)$ Eliminate from S the ideals that are also in T. Replace each ideal T_j in the decomposition of T by $\bigcap_k (S_k + T_j)$ $T := \bigcap_j \mathbf{PRIMEDECOMP}(T_j)$ Eliminate from S the ideals that are also in T. END </p>
--

Table 2: CANPERFSPEC algorithm.

in the form

$$Y_{\Sigma'} = \mathbb{V}(\bigcap_k N_k) \setminus \mathbb{V}(\bigcap_j (M_j)),$$

where the ideals N_k, M_j are prime (relative to K , but can be reducible wrt K') and each M_j strictly contains some N_k :

$$M_j \supsetneq N_{k(j)}.$$

Proof. By construction CANPERFSPEC produces the minimal decomposition with the required conditions. \square

In the iterative rewriting process two kinds of vertices of the rewritten trees become important: from here on, let us denote the vertices labelled by the all zero vector $[0, \overset{0}{\dots}, 0]$ by the number n of zeroes, and the vertices of the form $[0, \overset{n-1}{\dots}, 0, 1]$ by n' . With this notation, the root vertex $[]$ becomes vertex 0. The top discriminant ideal is now $J_0 = \bigcap_i N_{s_i(0)}$, and by Theorem 6, $\mathbb{V}(J_0)$ is the minimal singular variety of the root vertex.

The first step of REBUILDTREE, already described in [MaMo05], consists of the following:

1. put at the top vertex the decision about the discriminant J_0 ,
2. hang at the non null branch the generic basis $B_{g(0)}$,

<p>$T' \leftarrow \mathbf{REBUILDTREE}(T, n)$</p> <p>Input: T, the tree built by BUILDTREE or by REBUILDTREE($T, n - 1$) n, integer denoting the recursion level (initially 0).</p> <p>Output: T', the new rebuilt tree T.</p> <p>BEGIN</p> <p>REPEAT</p> <p>$T' :=$ copy T from the root cutting after vertex $[0, \overset{v}{\dots}, 0]$ $J_a =$ null condition ideal of vertex $[0, \overset{v}{\dots}, 0]$ of T, (initially empty) $v_g :=$ generic terminal vertex $[0, \overset{v}{\dots}, 0, 1, \dots, 1]$ relative to $[0, \overset{v}{\dots}, 0]$ of T $s_{pc} :=$ SPECCASES(n), $J := \bigcap_{s_i \in s_{pc}} N_{s_i}$. Add J to vertex n of T'.</p> <p>Place at vertex $[0, \overset{v}{\dots}, 0, 1]$ of T':</p> <ul style="list-style-type: none"> - $\Sigma_{v_g} = \mathbf{CANPERFSPEC}(J_a, J)$ (the canonical representation) - the basis B_{v_g} of T <p>Hang from the null branch of vertex $[0, \overset{v}{\dots}, 0]$ of T' the subtree of T starting with vertex $[0, \overset{v}{\dots}, 0]$ in T, rewrite it by adding the null condition J using REDSPEC, (some of the vertices v in T with $\text{lpp}_v = \text{lpp}_{v_g}$ will disappear in T' because they become incompatible)</p> <p>UNTIL the whole rebuilding is done</p> <p>END</p>

Table 3: REBUILDTREE algorithm.

3. hang at the non null branch the perfect specification $\Sigma' = (\{0\}, J_0)$ representing the set $Y_0 = K^m \setminus \mathbb{V}(J_0)$, (or better its canonical representation provided by CANPERFSPEC)
4. hang from the null branch of the new rebuilt tree, the old tree T_{-1} . To do this
 - add the null condition J_0 to the reduced specifications in all the descending vertices,
 - use REDSPEC to recompute the specifications,
 - reduce the bases using the new specifications.

Given I and the order $\succ_{\bar{x}}$, the specification of the absolute generic vertex $g(0)$ satisfies $\overline{X_{g(0)}} = K^m$ by Theorem 3, and its lpp set is intrinsic by Gianni's Theorem [Gi87]. Thus V_0 is also intrinsic, because $\bigcup_i X_{s_i(0)}$ is the region of K^m containing not generic special cases, and this does not depend on the algorithm. So the radical of the top discriminant ideal is also intrinsic. We have already proved that it is equal to Weispfenning[We03] top discriminant ideal.

```

Spc ← SPECCASES(n)
Input: n, integer denoting the recursion level (initially 0).
Output: Spc, the special cases descending from vertex n of T.
BEGIN
  c := Set of terminal vertices descending from vertex n
  c1 := Select from c the cases with  $\text{lpp}(B_i) \neq \text{lpp}(B_{v_g})$ 
  c2 :=  $\phi$ 
  IF c ≠ c1 THEN
    FOR g ∈ Bvg DO hg :=  $\text{lc}(ag)$  with  $ag \in \sigma_{v_g}(I)$ 
    FOR ALL ti ∈ c \ c1 DO
      test := true
      FOR ALL g ∈ Bvg WHILE test DO
        IF  $\sigma_{t_i}(h_g) = 0$  THEN test := false
        IF test THEN c2 := c2 ∪ {ti}
  Spc := c1 ∩ c2
END

```

Table 4: SPECCASES routine called inside REBUILDTREE.

After the first rebuild the specification at the new vertex 1 is $\Sigma'_1 = (J_0, \langle 1 \rangle)$ and corresponds to the root vertex of the old tree T_{-1} . That specification represents the subset of values of the specializations included $Y_0 = \mathbb{V}(J_0)$ for which B_0 is the GRGB. The non null branch will summarize the largest set of generic specifications of T_{-1} given by a perfect specialization that were before under the null branch. Most of the non-special cases of the old tree will be included now in the new enlarged generic specification and correspondingly will become incompatible and disappear under the null branch in the new tree when adding the discriminant condition. The new specifications under the null branch will continue to verify Theorems 3, 6. The old singular cases at the terminal vertices will neither disappear nor change, because these null conditions already contain the new discriminant condition.

The rewrite process can be iterated now starting from vertex 1, and in general from vertex *n* until the complete rewriting process has been finished. The iterated REBUILDTREE algorithm is shown in Table 3 and the routine SPECCASES used by it to determine the special cases hanging from vertex *n* of the *n* – 1 rebuilt tree is shown in Table 4. As it can be seen in SPECCASES routine, we need to use a quotient of ideals similar to that used to determine Weispfenning’s discriminant.

The whole rewrite process should now be clear. At the second rebuild the discriminant J_1 will be computed and set as decision in vertex $1 = [0]$. Thus the specification at the new vertex $2' = [0, 1]$ will be $\Sigma'_{2'} = (J_0, J_1)$ which is clearly a perfect specification, and at vertex $2 = [0, 0]$, it will be $\Sigma'_2 = (J_1, \langle 1 \rangle)$, which is also a perfect specification.

The complete rebuilt tree *T* will thus have the vertices and specifications given in Figure 2.

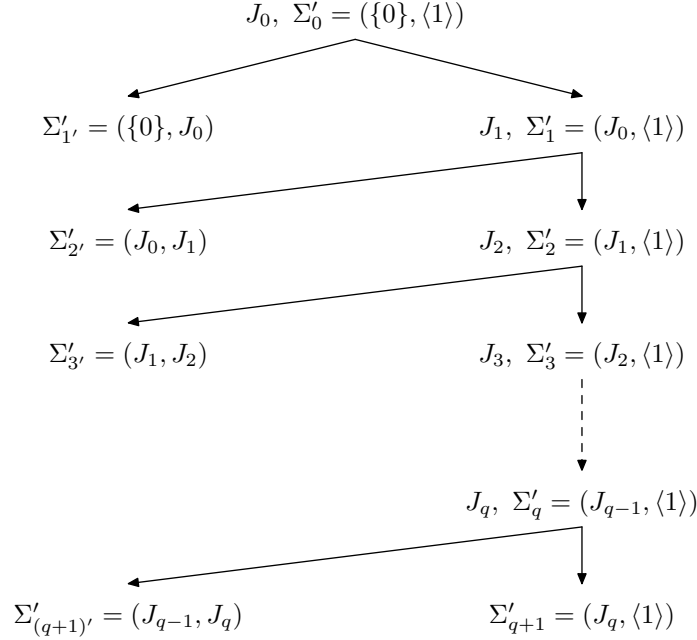


Figure 2: Discriminants and Specifications in the rebuilt tree.

The discriminant ideals J_i and the associated varieties verify the inclusions

$$\begin{aligned} \{0\} \subsetneq \langle J_0 \rangle \subsetneq \langle J_1 \rangle \subsetneq \langle J_2 \rangle \subsetneq \cdots \subsetneq \langle J_q \rangle \subsetneq \langle 1 \rangle \\ K'^m \supsetneq \mathbb{V}(J_0) \supsetneq \mathbb{V}(J_1) \supsetneq \mathbb{V}(J_2) \supsetneq \cdots \supsetneq \mathbb{V}(J_q) \supsetneq \phi, \end{aligned} \quad (1)$$

and the set of values of the parameters corresponding to the specification of the terminal vertices $(i+1)' = [0, \overset{i}{\dots}, 0, 1]$ are $Y_i = \mathbb{V}(J_{i-1}) \setminus \mathbb{V}(J_i)$. For the last terminal vertex $q+1 = [0, \overset{q+1}{\dots}, 0]$ it is $Y_{q+1} = \mathbb{V}(J_q) \setminus \mathbb{V}(\langle 1 \rangle) = \mathbb{V}(J_q)$. Formula (1) gives a discriminant chain of descending varieties associated to the polynomial ideal I , as it defines the regions of K'^m space having different reduced Gröbner bases. The strict inclusions in the chains arises as a consequence of $Y_i = \mathbb{V}(J_{i-1}) \setminus \mathbb{V}(J_i)$ being non-empty, as it contains the generic case $g(i)$.

4 Canonical Character of the Discussion

Note that even the first tree $T_1(I, \succ_{\bar{x}}, \succ_{\bar{a}})$ built by BUILDTREE provides a finite partition $\mathcal{X} = \{X_1, \dots, X_p\}$ of K'^m and the corresponding $G_i = \text{GRGB}(\Sigma_i) \in R[\bar{x}]$. BUILDTREE is algorithm depending, and so a different partition $\mathcal{X}' = \{X'_{t_0}, X'_{t_1}, \dots, X'_{t_p}\}$ using a different strategy in DISPGB. But the union of the sets specified in the cases corresponding to a unique GRGB specializing to the reduced Gröbner basis of the specialized ideal (with the same lpp set), must be the same in \mathcal{X} and in \mathcal{X}' , as these sets are not algorithm depending.

The rebuild process groups sets of X_i 's, having a unique GRGB that specializes to the reduced Gröbner basis of the specialized ideal (with the same lpp's), into a single Y_j .

Moreover, CANPERFSPEC provides a canonical description of the Y_i .

Thus, when REBUILDTREE finishes with a partition where none of the generic reduced Gröbner bases are equivalent by specialization, (either they have different lpp sets or they do not specialize in the same way), than the discussion is described in a complete canonical form.

Even when REBUILDTREE is not able to summarize every generic reduced Gröbner basis into a single case, we can use CANPERFSPEC in an external routine applied to the final tree to achieve the unification of cases. So it is always possible to obtain a canonical representation of the different specialization cases.

Nevertheless the order in which the different cases appear in the tree is still algorithm depending to some extent. Let us comment this point.

The absolute generic basis $B_{g(0)}$ does not depend on the algorithm, and so the same is true for the set $\cup_i X_{s_i(0)} = \cup_i X'_{s_i(0)}$. Thus $\mathbb{V}(J_0)$ is canonical.

The first step of REBUILDTREE algorithm then rewrites the tree setting as new generic case the perfect specification given by $K^m \setminus \mathbb{V}(J_0)$ that is thus not algorithm depending. All the special cases remain under the null branch of the new top decision.

At the new step of REBUILDTREE the new generic basis $B_{g(1)}$ relative to the new vertex 1, for which only non-null conditions have been added is chosen, and the new discriminant $J_1 \supset J_0$ is determined. So the new perfect specification $\Sigma'_{1'} = (J_0, J_1)$ is obtained for the new generic vertex relative to the new vertex 1, that we denote 1'. It corresponds to the set of parameter values $\mathbb{V}(J_0) \setminus \mathbb{V}(J_1)$. Only the possibility of a different choice of the basis for the new generic case relative to vertex 1 that can appear in a different tree obtained by the non canonical BUILDTREE, can destroy (misstand) the canonical character of the rebuilding process.

To reach the generic case of a vertex in BUILDTREE, only compatible non-null decisions have been taken in the sense of reduced specifications. Thus the dimension of the Zariski closure of the generic case must be equal to that of $\mathbb{V}(J_0)$.

If $\dim(\mathbb{V}(J_1)) < \dim(\mathbb{V}(J_0))$ then the generic basis of case 1' is also canonical as it corresponds to the unique specification under vertex 1 whose Zariski closure has dimension equal to $\dim(\mathbb{V}(J_0))$, and thus does not depend on the particular tree built up by BUILDTREE.

The algorithm continues working so and whenever $\dim(\mathbb{V}(J_i)) < \dim(\mathbb{V}(J_{i-1}))$ the choice of the generic basis associated to vertex i does not depend on the algorithm. If $\dim(\mathbb{V}(J_i)) < \dim(\mathbb{V}(J_{i-1}))$ happens for all i then the discussion tree will be absolutely canonical too.

Nevertheless it can happen that, for some i , $\dim \mathbb{V}(J_{i-1}) = \dim \mathbb{V}(J_i)$. In this case $\mathbb{V}(J_i)$ is formed by a subset of the irreducible components of $\mathbb{V}(J_{i-1})$, some of them having the same dimension as the correspondent in $\mathbb{V}(J_{i-1})$. In this case a different possible choice of the generic case is possible and this would produce a new choice J'_i with $\mathbb{V}(J'_i)$ in $\overline{\mathbb{V}(J_{i-1})} \setminus \mathbb{V}(J_i)$. This will produce an inversion in the order of the cases inside $\mathbb{V}(J_{i-1})$. Thus the discussion tree is not absolutely canonical and can produce minor predictable changes in the order of the discussion. This will become clear in the examples.

Let $G = \text{gb}(I, \succ_{\bar{x}, \bar{a}})$ be the reduced Gröbner basis of I computed wrt the product order $\succ_{\bar{x}, \bar{a}}$. This is a tentative Comprehensive Gröbner Basis of I . The canonical Gröbner system obtained by DISPGB can be used to complete it to a true CGB, that will then also be Canonical. For this, it suffices to verify for which cases and polynomials no polynomial in G does specialize to it. For each of them we can compute pre-images in I using an

appropriate algorithm. This will be discussed in detail in another paper.

5 Examples

5.1 Example 1

Consider the following linear system with ideal

$$I = \langle x + cy + bz + a, cx + y + az + b, bx + ay + z + c \rangle$$

having three parameters. Take monomial orders $\text{lex}(x, y, z)$ and $\text{lex}(a, b, c)$.

DISPGB obtains the following discriminant ideal chain:

$$\begin{aligned} J_0 &= [a^2 + b^2 + c^2 - 2abc - 1] \\ J_1 &= [b^4c^2 - b^4 - b^2c^4 + b^2 + c^4 - c^2, ac^3 - ac + b^3c^2 - b^3 - bc^4 + b, \\ &\quad abc^2 - ab - c^3 + c, ab^2 - ac^2 - b^3c + bc^3, a^2 + b^2 + c^2 - 2abc - 1] \\ J_2 &= [c^2 - 1, a - bc] \\ J_3 &= [c^2 - 1, b^2 - 1, a - bc] \end{aligned}$$

with $J_0 \subsetneq J_1 \subsetneq J_2 \subsetneq J_3$. Decomposing the associated minimal varieties using CANPERF-SPEC we have:

$$\begin{aligned} \mathbb{V}(J_0) &= \mathbb{V}(c^2 + b^2 - 1 - 2acb + a^2) \\ \mathbb{V}(J_1) &= \mathbb{V}(a + c, b + 1) \cup \mathbb{V}(c - b, a - 1) \cup \mathbb{V}(c - a, b - 1) \cup \mathbb{V}(b + c, a + 1) \cup \\ &\quad \cup \mathbb{V}(a + b, c + 1) \cup \mathbb{V}(a - b, c - 1) = r_1 \cup r_2 \cup r_3 \cup r_4 \cup r_5 \cup r_6 \\ \mathbb{V}(J_2) &= \mathbb{V}(a + b, c + 1) \cup \mathbb{V}(a - b, c - 1) = r_5 \cup r_6 \\ \mathbb{V}(J_3) &= \mathbb{V}(a - 1, b - 1, c - 1) \cup \mathbb{V}(a - 1, b + 1, c + 1) \cup \mathbb{V}(a + 1, b - 1, c + 1) \cup \\ &\quad \cup \mathbb{V}(a + 1, b + 1, c - 1) = \{(1, 1, 1), (1, -1, -1), (-1, 1, -1), (-1, -1, 1)\} \\ &= \{P_1, P_2, P_3, P_4\}. \end{aligned}$$

The varieties verify $\mathbb{V}(J_0) \supseteq \mathbb{V}(J_1) \supseteq \mathbb{V}(J_2) \supseteq \mathbb{V}(J_3)$ cf. Figure 3. The corresponding lpp sets of the reduced Gröbner bases of the Y 's are all different and are summarized in the following table

Specification	$\mathbb{C}^3 \setminus V_0$	$V_0 \setminus V_1$	$V_1 \setminus V_2$	$V_2 \setminus V_3$	V_3
lpp sets	$[x, y, z]$	$[1]$	$[x, y]$	$[z, y]$	$[y]$

V_0 is a surface (dimension 2). It contains V_1 consisting of the six straight lines r_1, \dots, r_6 . V_2 consists of two of these lines $r_5 \cup r_6$. Finally V_3 consists of the four points P_1, \dots, P_4 that are the intersection of $V_2 = r_5 \cup r_6$ and $V_1 = r_1 \cup r_2 \cup r_3 \cup r_4$. In this case, an alternative discriminant chain is possible with $V'_0 = V_0$, $V'_1 = V_1$, $V'_2 = \overline{V_1 \setminus V_2}$, $V'_3 = V_3$, for which the positions 2' and 3' will be permuted, being the unique change in the tree.

This situation can occur in a generalized situation. Suppose that in a problem we have a variety $V = \bigcup_{i=1}^s V_i$, where the V_i 's have the same dimension and $V_i \cap V_j = W$ for every $i \neq j$ and $\dim W < \dim V$. Suppose that the problem admits the chain

$$V \supseteq \bigcup_{i=2}^s V_i \supseteq \bigcup_{i=3}^s V_i \supseteq \dots \supseteq V_s \supseteq W.$$

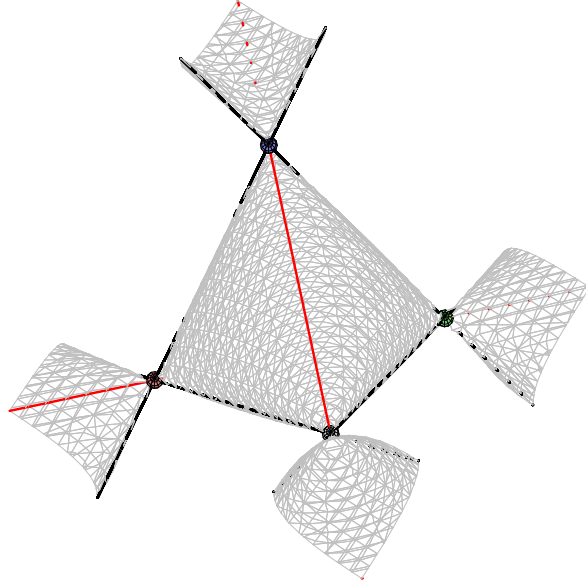


Figure 3: Varieties of the discriminant chain.

The specializations of the different cases that will determine CANPERFSPEC will be $V_i \setminus W$, and they are independent of the order in which the embedded varieties are chosen by a strategy in DISPGGB. Each order of the V_i 's is then possible for the tree, but the description of the cases is canonical and independent of the order in the tree.

5.2 Example 2

In order to illustrate distinct possible alternatives for the discussion tree consider the following ideal with three parameters

$$I = [abc(x^5y + z), bc(x^2y^2 + yz), ac(xy^2 + y), ab(xy + z), axz^3 + y]$$

Take monomial orders $\text{lex}(x, y, z)$ and $\text{lex}(a, b, c)$. Depending on the strategy in DISPGGB we obtain three different trees that can be seen in Figure 4 and the corresponding ascending discriminant chains:

Tree 1	$\langle (a-1)abc \rangle \subsetneq \langle abc \rangle \subsetneq \langle bc \rangle \subsetneq \langle ac, bc \rangle \subsetneq \langle ab, ac, bc \rangle \subsetneq \langle a, bc \rangle$
Tree 2	$\langle (a-1)abc \rangle \subsetneq \langle abc \rangle \subsetneq \langle ab \rangle \subsetneq \langle ab, ac \rangle \subsetneq \langle bc \rangle \subsetneq \langle a, b, c \rangle$
Tree 3	$\langle (a-1)abc \rangle \subsetneq \langle abc \rangle \subsetneq \langle ac \rangle \subsetneq \langle ab, ac \rangle \subsetneq \langle a \rangle$

Nevertheless, as we have discussed in previous section, using CANPERFSPEC all the chains lead to the same canonical specification and generic reduced Gröbner bases of the cases shown in the next table.

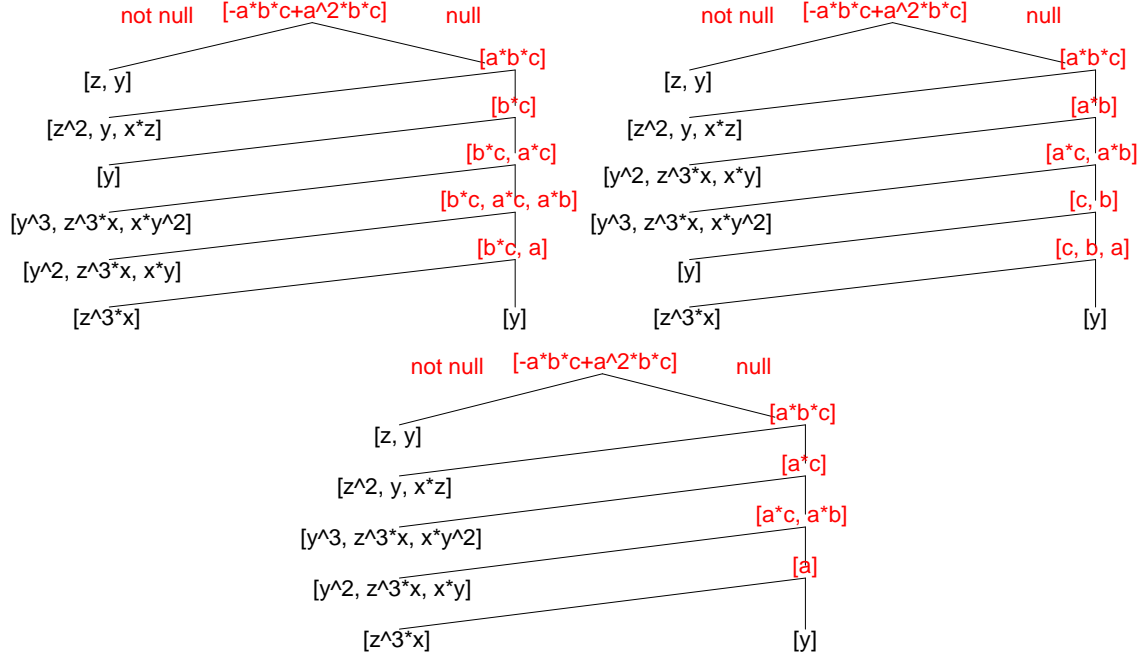


Figure 4: Discussion trees 1,2,3 in example 2.

Specification	Reduced Gröbner Basis
$\mathbb{C}^3 \setminus (\mathbb{V}(a) \cup \mathbb{V}(a-1) \cup \mathbb{V}(b) \cup \mathbb{V}(c))$	$[z, y]$
$\mathbb{V}(a-1) \setminus (\mathbb{V}(a-1, b) \cup \mathbb{V}(a-1, c))$	$[z^2 - z, y + z, xz - z]$
$\mathbb{V}(b) \setminus (\mathbb{V}(b, c) \cup \mathbb{V}(a, b))$	$[y^3 - y a z^3, a x z^3 + y, x y^2 + y]$
$\mathbb{V}(c) \setminus (\mathbb{V}(b, c) \cup \mathbb{V}(a, c))$	$[y^2 - z^4 a, a x z^3 + y, x y + z]$
$\mathbb{V}(b, c) \setminus \mathbb{V}(a, b, c)$	$[a x z^3 + y]$
$\mathbb{V}(a)$	$[y]$

Even the case with $\text{GRGB} = [y]$ that appears divided in two subcases in trees 1 and 2, but is summarized in one single case in the third tree, have the same representation after using `CANPERFSPEC` to add them. This illustrates well the canonical character of the discussion built up by `DISPGB`.

6 Acknowledgements

I would like to thank Drs. Pelegrí Viader and Julian Pfeifle for their many helpful comments and his insightful perusal of our first draft. I also thank my Ph. D. student M. Manubens for helping in all the discussions and implementation details. I am indebted to M. Wibmer for his interesting counter-example that lead me to correct the definition of singular specifications.

References

- [Be94] T. Becker. On Gröbner bases under specialization. *Appl. Algebra Engrg. Comm. Comput.* (1994) 1–8.
- [BeWe93] T. Becker, V. Weispfenning. Gröbner Bases: A Computational Approach to Commutative Algebra. Springer, New-York, (1993).
- [Co04] M. Coste. Classifying serial manipulators: Computer Algebra and geometric insight. Plenary talk. (Personal communication). *Proceedings of EACA-2004* (2004) 323–323.
- [De99] S. Dellière. Triangularisation de systèmes constructibles. Application à l'évaluation dynamique. Thèse Doctorale, Université de Limoges. Limoges, (1995).
- [Du95] D. Duval. Évaluation dynamique et clôture algébrique en Axiom. *Journal of Pure and Applied Algebra* **99** (1995) 267–295.
- [Em99] I. Z. Emiris. Computer Algebra Methods for Studying and Computing Molecular Conformations. *Algorithmica* **25** (1999) 372–402.
- [FoGiTr01] E. Fortuna, P. Gianni and B. Trager. Degree reduction under specialization. *Jour. Pure and Applied Algebra*, **164**(1-2) (2001) 153–164. *Proceedings of MEGA 2000*.
- [Gi87] P. Gianni. Properties of Gröbner bases under specializations. In: EUROCAL'87. Ed. J.H. Davenport, Springer LCNS **378** (1987) 293–297.
- [Gom02] T. Gómez-Díaz. Dynamic Constructible Closure. *Proceedings of Posso Workshop on Software*, Paris, (2000) 73–93.
- [GoRe93] M.J. González-López, T. Recio. The ROMIN inverse geometric model and the dynamic evaluation method. In: Computer Algebra in Industry. Ed. A.M. Cohen, Wiley & Sons, (1993) 117–141.
- [GoTrZa00] M.J. González-López, L. González-Vega, C. Traverso, A. Zanoni. Gröbner Bases Specialization through Hilbert Functions: The Homogeneous Case. *SIGSAM BULL* (Issue 131) **34**:1 (2000) 1–8.
- [GoTrZa05] L. González-Vega, C. Traverso, A. Zanoni. Hilbert Stratification and Parametric Gröbner Bases. *Proceedings of CASC-2005* (2005) 220–235.
- [GuOr04] M. de Guzmán, D. Orden. Finding tensegrity structures: geometric and symbolic approaches. *Proceedings of EACA-2004* (2004) 167–172.
- [HeMcKa97] P. Van Hentenryck, D. McAllester and D. Kapur. Solving polynomial systems using a branch and prune approach. *SIAM J. Numer. Anal.* **34**:2 (1997) 797–827.
- [Ka97] M. Kalkbrenner. On the stability of Gröbner bases under specializations. *Jour. Symb. Comp.* **24**:1 (1997) 51–58.

- [Kap95] D. Kapur. An Approach for Solving Systems of Parametric Polynomial Equations. In: Principles and Practices of Constraints Programming. Ed. Saraswat and Van Hentenryck, MIT Press, (1995) 217–244.
- [MaMo05] M. Manubens, A. Montes. Improving DISPGB Algorithm Using the Discriminant Ideal. Short abstract in Proceedings of Algorithmic Algebra and Logic (2005) 159–166. arXiv: math.AC/0601763. To be published in *Jour. Symb. Comp.*
- [Mo95] A. Montes. Solving the load flow problem using Gröbner bases. *SIGSAM Bull.* **29** (1995) 1–13.
- [Mo98] A. Montes. Algebraic solution of the load-flow problem for a 4-nodes electrical network. *Math. and Comp. in Simul.* **45** (1998) 163–174.
- [Mo02] A. Montes. New algorithm for discussing Gröbner bases with parameters. *Jour. Symb. Comp.* **33**:1-2 (2002) 183–208.
- [Mor97] M. Moreno-Maza. Calculs de Pgcd au-dessus des Tours d'Étendements Simples et Résolution des Systèmes d'Équations Algébriques. Doctoral Thesis, Université Paris 6, 1997.
- [Pe94] M. Pesh. Computing Comprehensive Gröbner Bases using MAS. User Manual, Sept. 1994.
- [Ry00] M. Rychlik. Complexity and Applications of Parametric Algorithms of Computational Algebraic Geometry. In: Dynamics of Algorithms. Ed. R. del la Llave, L. Petzold, and J. Lorenz. IMA Volumes in Mathematics and its Applications, Springer-Verlag, **118** (2000) 1–29.
- [SaSu03] Y. Sato and A. Suzuki. An alternative approach to Comprehensive Gröbner Bases. *Jour. Symb. Comp.* **36** (2003) 649–667.
- [SaSuNa03] Y. Sato, A. Suzuki, K Nabeshima. ACGB on Varieties. *Proceedings of CASC 2003*. Passau University, (2003) 313–318.
- [Sc91] E. Schönfeld. Parametrische Gröbnerbasen im Computeralgebrasystem ALDES/SAC-2. Dipl. thesis, Universität Passau, Germany, May 1991.
- [Si92] W. Sit. An Algorithm for Solving Parametric Linear Systems. *Jour. Symb. Comp.* **13** (1992) 353–394.
- [We92] V. Weispfenning. Comprehensive Gröbner Bases. *Jour. Symb. Comp.* **14** (1992) 1–29.
- [We03] V. Weispfenning. Canonical Comprehensive Gröbner bases. *Proceedings of ISSAC 2002*, ACM-Press, (2002) 270–276. *Jour. Symb. Comp.* **36** (2003) 669–683.
- [Wi06] M. Wibmer. (Private communication.) Gröbner bases for families of affine schemes. <http://relationsproject.at.tt> (2006).