Triangular Systems and Factorized Gröbner Bases*

Hans-Gert Gräbe
Institut für Informatik, Universität Leipzig, 04109 Leipzig, Germany

April 4, 1995

Abstract

In a preceding paper [9] we reported on some experience with a new version of the well known Gröbner algorithm with factorization and constraint inequalities. Here we discuss, how this approach may be refined to produce triangular systems in the sense of [12] and [13]. Such a refinement guarantees, different to the usual Gröbner factorizer, to produce a quasi prime decomposition, i.e. the resulting components are at least pure dimensional radical ideals. As in [9] our method weakens the usual restriction to lexicographic term orders.

Triangular systems are a very helpful tool between factorization at a heuristical level and full decomposition into prime components.

Our approach grew up from a consequent interpretation of the algorithmic ideas in [5] as a *delayed* quotient computation in favour of early use of (multivariate) factorization. It is implemented in version 2.2 of the REDUCE package CALI [8].

1 Introduction

Solving systems of polynomial equations in an ultimate way means to find a decomposition of the variety of solutions into irreducible components and to present them in a way that is well suited for further computations. The only algorithms known nowadays for such a *prime decomposition* are based on the ideas developed in the fundamental paper [5]. There exist several implementations and reports about them, see e.g. [11] or the monograph [1]. The main tool is a reduction of the dimension of the underlying ideal either inverting one of the variables or intersecting with appropriate hypersurfaces. This needs several stable quotient computations to compute retractions etc. Only in the last part of the algorithm, in dimension zero and after a general (or moderate, as suggested in [11]) change of coordinates factorization (of univariate polynomials) is involved. Both the quotient computation and the change of coordinates tend to make things expensive with regard to computation time.

In this paper we investigate the opposite approach, i.e. how far one can proceed towards a prime decomposition, heavily using factorization (of multivariate polynomials), delaying the computation of stable ideal quotients to the end of the algorithm.

Such a delayed quotient computation may be represented as a pair (B, c) with $B \subset S$ generating the ideal I and $c \in S$ a polynomial non degeneracy condition. Since the zero set

*Appeared in *Proceedings AAECC-11*, Paris 1995, ed. G. Cohen, M. Giusti, T. Mora. Lecture Notes in Comp. Sci. 948, Springer 1995, 248 - 261.

of the stable quotient $I :< c >$ is the closure of $Z(I) \setminus Z(c)$, at a heuristic level this is exactly the well known Gröbner algorithm with factorization (FGB), see section 2 below or [9].

Practically important results are obtained with respect to a pure lexicographic term order, but such Gröbner bases are usually quite hard to compute. In [9] we stressed also an alternative approach and computed factorized Gröbner bases with respect to a "cheaper" term order. If the problem really factors in such a frame, it is often easy to compute lexicographic (factorized) Gröbner bases of each of these smaller pieces (either directly or by base change techniques).

Another observations of [9] was the fact, that even for polynomial systems, comimg from applications, and dimension zero FGB does not always split the corresponding zero set into irreducible components. Lazard proposed in [12] and [13] to weaken the irreducibility condition and to ask only for triangular systems. In dimension zero they generalize the notion of prime ideals and are well suited for further numerical evaluation, since they don't involve a change of coordinates. In general, given a triangular system for the (quasi) prime (i.e. at least radical and pure dimensional) ideal P in a polynomial ring S over the field k, one can extract a presentation for the (quasi) field $(S/P)_P$ as a finite extension of a pure transcendental extension of k. This is another way to present such a (quasi) prime component. The ideal basis may be recovered from this set by a (non zero dimensional) stable quotient computation if requested, see prop. 2.

Below we present a quasi prime decomposition algorithm. It is a modification of the prime decomposition algorithm in [5], but uses only factorized Gröbner bases with constraints and delays the computation of stable quotients until the ideal is radical and of dimension zero. The latter quotients are easier to compute than arbitrary stable quotients due to the linear algebra approach suggested by Möller in [14].

A first topic of our paper concerns the impact of the term order to be chosen in Möller's approach. Explaining in [14] the underlying idea for arbitrary (admissible) term orders the algorithm itself is formulated only for the pure lexicographic term order. As already for FGB, such an approach should be preferred, if the corresponding Gröbner basis may be calculated with reasonable effort. Otherwise multiple (factorized) Gröbner basis computations with respect to "cheap" term orders should be involved. We show by means of examples, that such a "slow turn to lex." may have some advantage.

The main topic of our paper is devoted to another generalization of the notion of triangular systems to positive dimension. It is different from both generalizations proposed in [13] and [16], and best suited, from our point of view, to be applied in a polynomial system solver. For a general problem (B, C) our algorithm computes a collection (T_k, V_k) of triangular systems T_k with respect to maximal independent sets V_k, such that, if we denote by $C_k := C(T_k, V_k)$ the set of leading coefficients of T_k in a representation with parameter set V_k, the ideals $I(T_k) :< \prod C_k >$ are pure dimensional radical ideals (and hence $\overline{Z(T_k, C_k)}$ quasi prime components), such that $\bigcup \overline{Z(T_k, C_k)} = \overline{Z(B, C)}$.

2 The Gröbner Algorithm with Factorization

Let $S := k[x_1, \ldots, x_n]$ be the polynomial ring in the variables $x_1, \ldots, x_n$ over the field k, $\bar{k}$ the algebraic closure of k, and $B := \{f_1, \ldots, f_m\} \subset S$ a finite system of polynomials. Denote by $I(B)$ the ideal generated by these polynomials, for $C := \{g_1, \ldots, g_k\}$ the *relative set of* *zeroes* by

$$Z(B,C) := \{a \in \bar{k}^n : \forall\, f \in B\ f(a) = 0 \text{ and } \forall g \in C\ g(a) \neq 0\},$$

and its Zariski closure by $\overline{Z(B,C)}$. The latter is the zero set of $I(B) :< \prod C >$ where $\prod C := \prod_{\alpha \in C} \alpha$.

In [9] we considered the following

General Problem

> *Given a system $B = \{f_1, \ldots, f_m\} \subset S$ of polynomials and a set of side conditions C find a collection (B_α, C_α) of polynomial systems B_α in "triangular" form (here: being a Gröbner basis) and side conditions C_α such that*
>
> $$Z(B,C) = \bigcup_\alpha Z(B_\alpha, C_\alpha)$$

and discussed, how it may be solved with the well known **Factorized Gröbner Bases Algorithm FGB(B,C)**. Its major steps are the following:

INPUT : A polynomial system with constraints (B, C).

OUTPUT : A list of polynomial systems with constraints (B_k, C_k), such that

– B_k are Gröbner bases and

– $\bigcup Z(B_k, C_k) = Z(B, C)$.

- During a preprocessing interreduce B and try to factor each polynomial $f \in B$. If f factors, replace B by a set of new problems, one for each factor of f. Update the side conditions and apply the preprocessing recursively. This ends up with a list of interreduced problems with non factoring base elements.
- For each basis in the problem list compute its list of critical pairs and start the corresponding Gröbner basis calculations. Each such calculation then consists of a polynomial list, a list of critical pairs not yet processed, and side conditions.
- Try each reduced (non zero) S-polynomial to factor before it will be added to the polynomial list. If it factors, split up the problem into as many subproblems as there are (different) factors, add each of the factors to the corresponding subproblem, and update the pair list and the side conditions.
- If the pair list is exhausted, extract the minimal Gröbner basis of the subproblem. If it is not yet interreduced (i.e. the reductum contains non standard terms), apply tail reduction to compute the minimal reduced Gröbner basis. This may cause some of the base elements to factor anew. Apply the preprocessing once more. If the result is stable then return it. Otherwise put the subproblems produced during the preprocessing back into the problem list.

Realizing this algorithm we used the following elementary operations:

1. **Updating after factorization**

   If (B, C) is a problem and $f \in I(B)$ factors as $f = g_1^{a_1} \dots g_m^{a_m}$ then replace the problem by the problem list

   $$\textbf{NewCon}(B, C, \{g_1, \dots, g_m\}) := \{(B \cup \{g_i\}, C \cup \{g_1, \dots, g_{i-1}\}) \mid i = 1, \dots, m\}$$

2. **Inconsistency check**

   (B, C) is inconsistent, i.e. $Z(B, C) = \emptyset$, if the normal form $NF(c, B) = 0$ for some $c \in C$.

3. **Subproblem removal check**

   (B_1, C_1) can be removed if there is a problem (or partial result) (B_2, C_2) such that $Z(B_1, C_1) \subset Z(B_2)$. This occurs if $NF(f, B_1) = 0$ for all $f \in B_2$. The second problem has to be replaced by $(B_2, C_1 \cap C_2)$.

3 Solving systems of polynomial equations

The algorithm presented so far may be applied to systems of polynomial equations with respect to arbitrary term orders. Since it is a heuristic approach, it doesn't guarantee to split all components. Especially with respect to the degrevlex term order, a nice order from a computational point of view, some or all components, even of different dimensions, usually keep glueing together. Even if the components are irreducible, but of positive dimension, their presentation through minimal ideal bases is quite difficult and not well suited for further numerical evaluation or to obtain more structural insight.

Triangular systems as defined below play an intermediate role in both directions. First, they present the (generalized) generic point of a component as a tower of (cyclic) algebraic extensions of a pure transcendental extension of k in a very nice form, well suited for further evaluation. Second, they form a decomposition of the zero set of the polynomial system, where each component is not necessarily irreducible, but is known to be at least radical and unmixed, i.e. quasi prime. Below we give an extension of the factorized Gröbner bases algorithm that produces such a collection of triangular sets from a given system of polynomial equations.

Our guide is the prime decomposition algorithm proposed in [5] and refined in [11]. It uses several Gröbner basis computations to split the problem into smaller ones, recursively reducing the dimension either by inverting variables or by cutting with hypersurfaces. Finally, prime ideals are presented as recontractions from zero dimensional prime ideals, defined over a localization of S, considering some of the variables as parameters.

We follow the same lines, but make extensive use of (multivariate) factorization to split the problems as early as possible. On the other hand, we try to delay or even to skip (time consuming) nonzero dimensional quotient computations. This is possible since for a numerical evaluation along a prime ideal P one may use a zero dimensional parametric presentation of the prime field $(S/P)_P$ rather than the (more complicated) basis of the recontracted ideal. The recontraction can easily be obtained solely from the presentation of $(S/P)_P$ if requested.

Moreover, the various Gröbner basis computations in the algorithm in [5] are substituted by factorized Gröbner basis computations whenever possible. Third, we avoid another time consuming step (splitting off different zero dimensional prime ideals using the general position argument, see [11]) producing triangular systems instead of prime ideals.

4 Zero dimensional triangular systems

According to our general setting the input data are polynomial systems with constraints. If (B, C) is such a pair then the closure $\overline{Z(B,C)}$ is the zero set of the stable quotient of $I(B)$ by $c := \prod C$. This closure is different from $Z(B)$ iff $Z(B)$ has components in the hypersurface $Z(c)$. For *zero dimensional ideals* $I(B)$ all components are closed points and therefore either completely contained in $Z(c)$ or don't meet the hypersurface. Hence for such problems all constraints may be incorporated into the system of polynomial equations.

Lazard introduced in [12] the notion of triangular systems for zero dimensional ideals and extended it in [13] to positive dimension. For zero dimensional ideals he proposed to apply the D5 algorithm for their computation. We follow another approach, suggested in [14].

A set of polynomials $\{f_1(x_1), f_2(x_1, x_2), \ldots, f_n(x_1, \ldots, x_n)\}$ is called a (zero dimensional) *triangular system* (reduced triangular set in [12]) if, for $k = 1, \ldots, n$, $f_k(x_1, \ldots, x_k)$ is monic (i.e. has an invertible leading coefficient) regarded as a polynomial in x_k over $k[x_1, \ldots, x_{k-1}]$, and the ideal $I = I(f_1, \ldots, f_n)$ is radical. For such a triangular system S/I is a finite sum of algebraic field extensions of k. One can effectively compute in such extensions, as was discussed in [12].

Proposition 1 *Let (B, C) be a zero dimensional polynomial system with constraints. There is an algorithm that computes a finite number of triangular systems $T_1, \ldots, T_m$, such that*

$$Z(B, C) = \bigcup_i Z(T_i).$$

This result is due to Lazard [12] and was refined by Möller [14]. The algorithm **TriangSets(B,C)** used in our experimental version with CALI is the following:

FIRST STEP : Find by Buchberger's approach, cf. [1, 9.6], univariate polynomials in each of the variables.

Use a modification of FGB that incorporates these polynomials to compute a set $\{(B_k, C_k)\}$ of polynomial systems with $Z(B, C) = \bigcup Z(B_k, C_k)$ such that $I(B_k)$ is radical (by [1, 8.14.]).

SECOND STEP : Substitute (B_k, C_k) by a basis of $I(B_k) :< \prod C_k >$. This quotient can be computed by the linear algebra approach described in [14].

THIRD STEP : Compute recursively triangular systems as described in [14], but use FGB for intermediate Gröbner basis computations.

Let's add one more remark on the algorithm proposed in [14]. Its basics are formulated for arbitrary elimination orders, whereas in the applications the author restricts himself to the pure lexicographic term order. The advantage of that order is the fact that the Gröbner basis computation in the main step immediately yields a Gröbner basis of each recursion step. On the other hand such a Gröbner basis is usually hard to compute. If we use another ("cheaper") elimination order each recursion step of the main algorithm requires a new (one can use again factorized) Gröbner basis computation. Alternatively one can use the FGLM linear algebra approach [4] to compute the new Gröbner basis from the old one.

In table 2 we collected some computational results, comparing such a "slow turn" to the pure lexicographic term order with the unique "brute force" pure lexicographic Gröbner basis computation. Here

> *ZS* corresponds to the original TriangSets with respect to the pure lexicographic term order as proposed in [14]. It often leads to computations with huge coefficients.
>
> *FGB* corresponds to a sole FGB computation with respect to the degrevlex term order as the initial part of our modification. In most cases it doesn't split off the components deep enough.
>
> *ZS1* corresponds to TriangSets with respect to the elimination order (lex. in the variable to be eliminated, then degrevlex. in the remaining variables, this way performing several intermediate FGB computations), starting with the degrevlex term order,
>
> whereas
>
> *ZS2* corresponds to TriangSets, starting with a degrevlex FGB computation, followed by a repeated Gröbner basis computation as for ZS1, but using the FGLM approach. (Since the FGLM approach does not split a splitting ideal, another FGB call tries to factor the new base polynomials. Upon success it splits the new Gröbner basis into several smaller ones)

example	ZS		FGB		ZS1		ZS2	
	time	comp.	time	comp.	time	comp.	time	comp.
K3	2.2	3	0.3	1	1.9	3	3.2	3
K4	> 10000		3.2	1	98.6	4	30.5	4
K5	> 10000		31.2	1	> 10000		367.5	3
A5	15.7	15	8.6	8	27.3	18	40.6	18
R7	> 10000		8.9	1	> 10000		15.5	3

Table 1 : Comparing different approaches to zero dimensional ideals

All computation times are CPU times on an IBM-RS/6000, obtained with version 2.2 of our REDUCE package CALI [8] and with integer coefficients. The number of components *comp* produced with the corresponding version of the algorithm gives a measure for the quality of the result beyond CPU time.

The examples are the following:

K3 – The Katsura example, [2], with 4 variables and primes of degree (1 1 6).

K4 – The Katsura example, [2], with 5 variables and primes of degree (1 1 2 12).

K5 – The Katsura example, [2], with 6 variables and primes of degree (1 1 30).

A5 – The Arnborg example, [6, 3.2.], with 5 variables and 20 prime components.

R7 – The rudimentary Arnborg example, [6, 3.3.]. It has prime components of degree (2 6 12).

We conclude that Möller's approach ZS should be preferred for easy examples, whereas the modifications $ZS1$ and $ZS2$ are worth to be tried if ZS fails.

5 Reduction to dimension zero

To describe the reduction to dimension zero we have to recall the notion of independent sets: For a given ideal $I \subset S$ the set of variables $(x_v, v \in V)$ is an *independent set* iff $I \cap k[x_v, v \in V] = (0)$. See [1] for the definition and also a guideline to the history of this notion. [7] contains another explanation of this notion, its connection to strongly independent sets, and discusses algorithms for an effective computation of strongly independent sets.

Let $B = \{f_1, f_2, \ldots, f_m\}$ be a set of polynomials in S. We say that they form a *triangular system with respect to the maximal independent set* $(x_v, v \in V)$ of I, if the extension $\tilde{B}$ of B to $\tilde{S} := k(x_v, v \in V)[x_v, v \notin V]$ forms a triangular system for the (zero dimensional) extension ideal $\tilde{I} := I \cdot \tilde{S}$. Note that in this case $\tilde{B}$ is a Gröbner basis of $\tilde{I}$ with respect to the lexicographic term order.

This definition is, up to a reordering of the variables, essentially the same as in [13]. Reordering variables yields a better distinction between the algebraic and transcendental parts of the extension, presenting the quotient ring $Q(\tilde{S}/\tilde{I})$ as a finite extension of $k(x_v : v \in V)$ also on the level of data structures.

Note that our triangular systems are automatically perfect triangular forms and regular chains with respect to the reordered variables, as defined in [16] resp. [10].

If I is prime then $I = \tilde{I} \cap S$. In general, the retraction ideal can be found by a stable quotient computation from a Gröbner basis (with respect to an *arbitrary term order* on $\tilde{S}$) of $\tilde{I}$. For this purpose let's remark, that one can compute denominator-free in $\tilde{S}$ using the well known pseudo normal form algorithm **PNF(p,B)**. It returns a denominator-free pseudo $\tilde{S}$-normal form $f \in S \subset \tilde{S}$ of the polynomial $p \in S$ with respect to the basis $B \subset S$, i.e. satisfying $z \cdot f \equiv p \ (mod\ I(B)\tilde{S})$ for a certain unit $z \in \tilde{S}$. z can be chosen to be a product of leading coefficients of the elements in B.

In the following a *denominator-free basis* B of $\tilde{I}$ is a set of polynomials in S such that they generate $\tilde{I}$ regarded as elements of $\tilde{S}$. Denote by $I(B)$ as before the ideal generated by B in the ring S. Note that B must not be contained in I if $I \neq \tilde{I} \bigcap S$.

Proposition 2 *Let B be a denominator-free Gröbner basis of $\tilde{I}$ over $\tilde{S}$ and c the product of the leading coefficients of the elements of B. Then*

$$\tilde{I} \cap S = I(B) :< c > .$$

Especially, if $dim(\tilde{S}/\tilde{I}) = 0$ then $I(B) :< c >$ is pure dimensional of dimension $|V|$.

Proof: Since c is invertible in $\tilde{S}$ we have only to show, that $\tilde{I} \cap S \subset I(B) :< c >$. But for a denominator-free element $f \in \tilde{I}$ we get $PNF(f, B) = 0$ and hence $f \in I(B) :< c >$. □

This is a slight modification of [5, 3.8.], where c is the product of all leading coefficients in a Gröbner basis of I instead of $\tilde{I}$.

By some abuse of notation we denote for a maximal independent set V of I and $B, \tilde{I}, \tilde{S}$ as above the set of leading coefficients of B considered as elements in $\tilde{S}$ (with respect to a given term order on $\tilde{S}$) by **C(B,V)**.

To find $\tilde{I} \cap S$ we have to remove all components of I that vanish in the localization $\tilde{S}$. Hence, given a problem (B, C) and a maximal independent set V for the ideal $I = I(B)$ we ask for all components of I, that don't pass through the generic point $(x_v, v \in V)$. They can be found as in [5, 8.2.], computing a (factorized) Gröbner basis of (B, C) with respect to an elimination order for $(x_v, v \notin V)$, i.e. where $x_v >> x_w$ for $v \notin V, w \in V$:

Proposition 3 *Let B be a Gröbner basis of I with respect to an elimination order for $(x_v, v \notin V)$, C a set of polynomial constraints, $\tilde{S} = k(x_v, v \in V)[x_v, v \notin V]$ the extension ring, $B' \subset B$ a subset that is a minimal Gröbner basis of $\tilde{I} = I \cdot \tilde{S}$, and D the set of leading coefficients of elements of B' regarded as polynomials in $\tilde{S}$ with respect to the induced term order. Then*

$$Z(B, C) = Z(\tilde{I} \cap S, C) \cup \bigcup \{Z(B_i, C_i) \ : \ (B_i, C_i) \in NewCon(B, C, D)\}.$$

This is a slight refinement of [5, 8.2.].

Proof: Indeed, the first component is a decomposition of $Z(B, C \cup D)$ and the second collection covers all branches of $Z(B, C) \cap Z(d)$ for $d \in D$. □

6 The Extended Gröbner Factorization Algorithm

Altogether we get the following algorithm for the decomposition of a polynomial system with constraints into triangular systems, that define quasi prime ideals:

The Extended Gröbner Factorization Algorithm EFGB

Input : A problem (B, C).

Output : A list of sets (T_k, C_k, V_k), such that

- T_k is a triangular system with respect to the maximal independent set V_k,
- $C_k = C(T_k, V_k)$ and
- $\overline{Z(B, C)} = \bigcup \overline{Z(T_k, C_k)}$

- Compute a factorized Gröbner basis and initialize the postprocessing :
  - results:=FGB(B,C),
  - problems:=∅.

REPEAT

- If there are new problems, convert them with FGB into results
- else take a result (B', C') of highest dimension,

- compute a maximal independent set V for B',
- compute a factorized Gröbner basis $\{(B_i, C_i)\}$ of (B', C') with respect to an elimination order for the variables outside V.
- convert all results (B_i, C_i), for which V remains an independent set, into triangular systems, i.e.
    - extract from B_i a minimal Gröbner basis B_i' in $\tilde{S} = k(x_v : v \in V)[x_v : v \notin V]$,
    - collect the leading coefficients of B_i' into the set D_i.
    - compute (denominator-free) in $\tilde{S}$ the collection `TriangSets(`B_i', C_i`)`, i.e. a decomposition into triangular systems $\{T_{ij}\}$ for the zero dimensional extension ideal (possibly empty, if $I(\tilde{B_i'}) :< \prod\ C_i >= (1)$).
    - add the sets $(T_{ij}, C(T_{ij}, V), V)$ to the output collection.
    - join $NewCon(B_i, C_i, D_i)$ with the problem list, since these problems are covered by (B_i, C_i) but not by the quasi primes obtained from it. Their dimension doesn't exceed $|V|$.
- add all other results (that were obtained during the additional Gröbner basis computation and are either of less or equal dimension or V failed to be an independent set) to the problem list.[1]

UNTIL

all problems are treated and all results are converted into triangular systems.

- Return the list of triangular systems.

From the discussion above it follows easily, that this algorithm terminates and produces a list of triangular systems with the desired property:

Proposition 4 *Let (B, C) be a polynomial system with side conditions over $S = k[x_1, \ldots, x_n]$. Then EFGB computes a decomposition (T_k, C_k, V_k), where*

- *T_k is a triangular system with respect to V_k,*
- *$I_k := I(T_k) :< \prod C_k >$ is a pure dimensional radical ideal with V_k as a maximal strongly independent set,*
- *$\overline{Z(B, C)} = \bigcup Z(I_k)$.*

There are some obvious improvements of the algorithm along the lines, explained for FGB. E.g. one can apply the subproblem removal check and the inconsistency check to the problems, obtained during the postprocessing, to keep this list as short as possible. On the other hand, the subproblem removal check can not be applied to the triangular systems directly, since their presentation does not support a direct comparison between sets attached to different independent sets. Hence the result of *EFGB* may be non minimal.

[1] Note that they are Gröbner bases, but with respect to another term order.

To compare different triangular systems one has to find their recontraction ideals I_k, a step that we tried to avoid during our algorithm. Denote for further reference the corresponding modification of $EFGB$, where for each triangular system a retraction is computed and used for subproblem removal checks in the spirit of [9] to keep the list of problems and results as short as possible, by **EFGB1**. Note that these computations may be done with respect to an arbitrary term order in S.

6.1 Some Examples

Examle 1 : Consider the graph of the space curve $C = \{(x^{31} - x^6 - x,\, x^8,\, x^{10}) : x \in \mathbf{C}\}$, i.e. the curve generated by $B = \{x^{31} - x^6 - x - y,\, x^8 - z,\, x^{10} - t\}$, but with respect to the variable order $x > y > z > t$, see [15] or [6, 3.4.]. Wang used it in [16] to illustrate his approach to triangular systems. Note that his aim was the construction of a full *stratification* $Z(B) = \bigcup Z(B_k, C_k)$ with (his) triangular systems (B_k, C_k), whereas we ask only for a *decomposition* into (our) triangular systems, from which all (i.e. here : the only) components of $Z(B)$ may be reconstructed (by prop. 2). For practical purposes it seems to be sufficient to restrict the effort to such a question.

Since $I(B)$ is a prime ideal, it can be described by a single triangular system with respect to the maximal independent set $\{t\}$. We get

$$\begin{aligned} B' = \{&(t^4 - t)\,x - t\,y - z^2\,,\\ &t^3\,y^2 + 2\,t^2\,y\,z^2 - (t^6 - 2\,t^3 - t + 1)\,z^4\,,\\ &z^5 - t^4\}. \end{aligned}$$

All variations of the Extended Gröbner Factorizer produce it as the essential part of the answer. Note that, different to Wang's representation T_1 (p. 91) of that part of the solution, the leading coefficients depend only on t.

With $EFGB1$ this is already the full output collection, since it detects superfluous components. $EFGB$ produces some auxiliary components, namely

$$\begin{aligned} &\{x\,,\,y\,,\,z\,,\,t\},\\ &\{x + z^2\,,\,y - z^2\,(t+1)\,,\,z^4 - z^3\,(t+1) + z^2\,t + z - t - 1\,,\,t^2 + t + 1\},\\ &\{x - z^2\,,\,y - z^2\,(t+1)\,,\,z^4 - z^3\,(t+1) + z^2\,t + z - t - 1\,,\,t^2 + t + 1\},\\ &\{x + t\,,\,y + 1\,,\,z + t + 1\,,\,t^2 + t + 1\},\\ &\{x - t\,,\,y + 1\,,\,z + t + 1\,,\,t^2 + t + 1\},\\ &\{x - z^2\,,\,y + z^2\,,\,z^4 + z^3 + z^2 + z + 1\,,\,t - 1\},\\ &\{x + z^2\,,\,y + z^2\,,\,z^4 + z^3 + z^2 + z + 1\,,\,t - 1\},\\ &\{x + 1\,,\,y + 1\,,\,z - 1\,,\,t - 1\},\\ &\{x - 1\,,\,y + 1\,,\,z - 1\,,\,t - 1\} \end{aligned}$$

for the combination with ZS and

$$\{x\,,\,y\,,\,z\,,\,t\}$$

for the combination with $ZS1$.

The following two examples come from the area of geometry theorem proving.

Example 2 : Apollonius' Circle Theorem (cf. [10]):
The altitude pedal of the hypotenuse of a right-angled triangle and the midpoints of the three sides of the triangle lie on a circle.

With vertices $O(0,0), A(a,0), B(0,b)$ and the pedal point $P(c,d)$ the geometric situation may be described by the following equations:

$$B := \{ -4\,a\,b + a\,d + b\,c,\ a\,c - b\,d\}.$$

The conclusion of the theorem may be expressed as $(a-c)^2 + (b-d)^2 - a^2 - b^2 = 0$ on the "geometrically relevant" part of $Z(B)$.

We ask for formulas that express the coordinates of P in a, b. For this purpose we compute triangular systems with respect to the variable order $c > d > a > b$. We obtain two essential solutions

$$T_1 := \{(a^2+b^2)\,c - 4\,a\,b^2,\ (a^2+b^2)\,d - 4\,a^2\,b\} \quad \text{and} \quad T_2 := \{\,a,\ b\,\},$$

where only the first one is geometrically relevant. The geometric non degeneracy condition is $C := C(T_1, \{a\,,\ b\}) = \{a^2+b^2\}$. Since $Z(B \bigcup C) = \{\,a,\ b\,\}$ this condition is equivalent to $a\,b \neq 0$, the "expected" one. In general, it is not clear how to compare different non degeneracy conditions and find a minimal or canonical one (in a sense to be made precise), cf. [17].

Example 3 : The midpoint perpendicular's intersection theorem, cf. [17].

With vertices $A(0,0), B(b_1,0), C(c_1,c_2)$ and $M(m_1,m_2)$ as a candidate for the intersection point the theorem can be formulated as the existence of (again geometrically meaningful) solutions of the following polynomial system of equations:

$$\begin{aligned} B = \{&-2\,m_1\,c_1 - 2\,m_2\,c_2 + c_1^2 + c_2^2\,, \\ &2\,m_1\,b_1 - 2\,m_1\,c_1 - 2\,m_2\,c_2 - b_1^2 + c_1^2 + c_2^2\,, \\ &b_1\,(-2\,m_1 + b_1)\} \end{aligned} \quad .$$

with respect to the variable order $m_1 > m_2 > b_1 > c_1 > c_2$. Computing triangular systems we get

$$\begin{aligned} T_1 &:= \{(2\,c_1)\,m_1 + 2\,m_2\,c_2 - (c_1^2 + c_2^2)\,,\ b_1\} \text{ and} \\ T_2 &:= \{2\,m_1 - b_1\,,\ 2\,c_2\,m_2 + (b_1\,c_1 - c_1^2 - c_2^2)\}, \end{aligned}$$

where the second solution is the desired proof. We get also the geometric non degeneracy condition $c_2 \neq 0$ as $C(T_2, \{b_1\,,\ c_1\,,\ c_2\})$.

The last example is a slight modification of example 2 in [16].

Example 4 :

$$B := \{(x-u)^2 + (y-v)^2 - 1\,,\ v^2 - u^3\,,\ 2\,v\,(x-u) + 3\,u^2\,(y-v)\}$$

As for the original example it is quite hard to compute the corresponding triangular systems (for $v > u > y > x$) with respect to the pure lexicographic term order. As already mentioned, our approach is not restricted to such an assumption. If we use the deglex. term order instead, *EFGB* produces a component with $\{u\,,\ v\,,\ x^2 + y^2 - 1\}$ and

$$\begin{aligned} B' := \big\{ & 729\,y^6 + y^4\,(-1458\,x^3 + 729\,x^2 - 4158\,x - 1685) + y^2\,(729\,x^6 - 1458\,x^5 - \\ & 2619\,x^4 - 4892\,x^3 - 297\,x^2 + 5814\,x + 427) + (729\,x^8 + 216\,x^7 - 2900\,x^6 - \\ & 2376\,x^5 + 3870\,x^4 + 4072\,x^3 - 1188\,x^2 - 1656\,x + 529), \\ & u\,(59049\,x^6 + 91854\,x^5 - 45198\,x^4 + 145152\,x^3 + 63549\,x^2 + 60922\,x + 21420) + \\ & 2187\,y^4\,(18\,x - 1) + 3\,y^2\,(-32805\,x^4 - 5832\,x^3 - 68283\,x^2 - 29520\,x - 5848) + \\ & (-72171\,x^6 - 45198\,x^5 - 128763\,x^4 - 4452\,x^3 + 173411\,x^2 + 49194\,x + 19731), \\ & v\,(1594323\,x^{10} + 2716254\,x^9 - 4041576\,x^8 - 3347568\,x^7 + 2788506\,x^6 - \\ & 2199348\,x^5 - 8874644\,x^4 - 2153376\,x^3 - 1888245\,x^2 + 630086\,x + 492660) + \\ & 729\,y^5\,(2187\,x^4 + 486\,x^3 - 1332\,x^2 - 1126\,x - 711) + 2\,y^3\,(-1594323\,x^7 - \\ & 177147\,x^6 - 3588867\,x^5 - 1380726\,x^4 + 3478059\,x^3 + 2984211\,x^2 + 2461087\,x + \\ & 523566) + y\,(-2716254\,x^9 - 2158569\,x^8 - 5824710\,x^7 - 1845180\,x^6 + \\ & 14549238\,x^5 + 4733958\,x^4 - 3887082\,x^3 - 4986900\,x^2 - 4101320\,x - 528813)\big\}. \end{aligned}$$

$C := \overline{Z(B', D)}$ with $D = C(B', \{x\})$ is a plane curve at distance 1 from the curve $v^2 - u^3 = 0$. B' presents the quotient field of this curve as an algebraic extension of degree 6 over $k(x)$.

The curve, originally considered in [16], is a twofold cover of C defined by another coordinate with values $w_1 = \frac{1}{3u^2}$ resp. $w_2 = \frac{1}{2v}$. We get by Lazard's method for these inverses in an algebraic extensions the presentation

$$w_1 = \frac{\begin{aligned} & 2187\,y^4\,(2187\,x^4 - 972\,x^3 - 1746\,x^2 - 1004\,x - 765) + 6\,y^2\,(-1594323\,x^7 + \\ & 708588\,x^6 - 3615111\,x^5 + 1678158\,x^4 + 4569615\,x^3 + 2921904\,x^2 + \\ & 2070347\,x + 365670) + (4782969\,x^{10} - 2125764\,x^9 - 16592769\,x^8 - \\ & 6403536\,x^7 + 35474598\,x^6 + 31452840\,x^5 + 4069914\,x^4 - 3931472\,x^3 - \\ & 3729375\,x^2 - 2207172\,x + 1214055) \end{aligned}}{\begin{aligned} & 48\,x^4\,(59049\,x^6 + 91854\,x^5 - 45198\,x^4 + 145152\,x^3 + 63549\,x^2 + 60922\,x + \\ & 21420) \end{aligned}}$$

and a similar one for w_2.

In table 2 we collected for different examples and variants of $EFGB$ the same data as in table 1. E1 - E4 are the examples discussed so far. The remaining examples we took from [2]: G1 and G2 are two variants of Gonnet's example, the original one (G1) and the homogenized as considered in [9] (G2), and H1 is the example Hairer 1. We combined both $EFGB$ and $EFGB1$ with ZS (i.e. pure lexicographic intermediate computations) and $ZS1$ (degrevlex. intermediate computations), as described in 3.1.

H1 demonstrates that it may be of real value not to compute the retraction ideals. On the other hand, for examples that split into many triangular systems as e.g. Gonnet's, the computation of the retraction ideals helps to pick up the essential ones. It needs further study to find the breakpoint between both approaches.

As already mentioned, $EFGB$ must not start with the pure lexicographic term order. It is of great value to have more freedom in choise, as demonstrates the second part of table 2. Here we collected the results for the hard examples from the first part, when computed with respect to the deglex. term order.

example	EFGB + ZS		EFGB1 + ZS		EFGB + ZS1		EFGB1 + ZS1	
	time	comp.	time	comp.	time	comp.	time	comp.
E1	21.0	10	22.3	1	17.7	2	18.2	1
E2	0.20	3	0.25	2	0.39	3	0.32	2
E3	0.44	6	0.26	2	0.56	6	0.26	2
G1	10.8	5	9.3	3	23.1	7	12.9	3
G2	265	98	168	7	273	68	228	7
H1	4.33	2	> 20000		13.1	2	> 20000	
E4	> 20000		> 20000		19.9	2	> 1000	
H1	5.0	2	41.0	1	7.53	2	333	2

Table 2 : Comparing different versions of the Extended Gröbner Factorizer

We conclude, that both the modification of the definition of triangular systems in positive dimension and the method for their computation proposed in this paper are well suited for the application in polynomial system solvers. Of great value are both the stronger definition of triangular systems, that is different from those proposed by other authors in connection with the characteristic set method and their variations, and the greater freedom in the choise of term orders to carry out the corresponding computations.

References

[1] Becker, T., Weispfenning, V., Kredel, H.: A computational approach to commutative algebra. Graduate Texts in Math. 141, Springer, New York, 1993.

[2] Boege, W., Gebauer, R., Kredel, H.: Some examples for solving systems of algebraic equations by calculating Gröbner bases. J. Symb. Comp. **2** (1986), 83 - 98.

[3] Chou, S. C.: Automated theorem reasoning in geometries using the characteristic set method and Gröbner basis method. In: Proc. ISSAC'90, ACM Press 1990, 255 - 260.

[4] Faugere, J., Gianni, P., Lazard, D., Mora, T.: Efficient computations of zerodimensional Gröbner bases by change of ordering. J. Symb. Comp. **16** (1993), 329 - 344.

[5] Gianni, P., Trager, B., Zacharias, G.: Gröbner bases and primary decomposition of polynomial ideals. J. Symb. Comp. **6** (1988), 149 - 167.

[6] Giovini, A. *et al.*: "One sugar cube, please" or selection strategies in the Buchberger algorithm. In: Proc. ISSAC'91, ACM Press, 1991, 49 - 54.

[7] Gräbe, H.-G.: Two remarks on independent set. J. Alg. Comb. **2** (1993), 137 - 145.

[8] Gräbe, H.-G.: CALI – A REDUCE package for commutative algebra. Version 2.2, Febr. 1995. Available through the REDUCE library e.g. at `redlib@rand.org`.

[9] Gräbe, H.-G.: On factorized Gröbner bases. To appear in: Proc. "Computer algebra in Science and Engineering", Bielefeld 1994.

[10] Kalkbrener, M.: A generalized Euclidean algorithm for geometry theorem proving. J. Symb. Comp. **15** (1993), 143 - 167.

[11] Kredel, H.: Primary ideal decomposition. In: Proc. EUROCAL-87, LNCS 378 (1989), 270 - 281.

[12] Lazard, D.: Solving zero dimensional algebraic systems. J. Symb. Comp. **13** (1992), 117 - 131.

[13] Lazard, D.: A new method for solving algebraic systems of positive dimension. Discr. Appl. Math. **33** (1991), 147 - 160.

[14] Möller, H.-M.: On decomposing systems of polynomial equations with finitely many solutions. J. AAECC **4** (1993), 217 - 230.

[15] Traverso, C., Donati, L.: Experimenting the Gröbner basis algorithm with the AlPi system. In: Proc. ISSAC'89, ACM Press 1989.

[16] Wang, D.: An elimination method for solving polynomial systems. J. Symb. Comp. **16** (1993), 83 - 114.

[17] Winkler, F.: Gröbner bases in geometry theorem proving and simplest non degeneracy conditions. Math. Pannonica **1** (1990), 15 - 32.