Introduction

It is more than 10 years since we have published our first special issue devoted to the Computer Algebra. For us it was a starting point for the development of Computer Algebra in Moldova. Looking back we can say that this was a period during which we have found our own place in this very important part of modern Computer Science. It was, of course, impossible without the help of our colleagues from other countries, without the financial support from INTAS and Swedish Academy of Science. In fact the most part of articles, published in this issue was supported by INTAS project Nr 05-104-7553 and we are very grateful for this important help.

It is interesting to compare old and new issues. The systems which were just introduced in the previous issue (Singular, Anick) are now actively used to obtain much more sophisticated and fine results. Gröbner bases and non-commutative computations which were something new at that time, now are the standard topics for student courses. Today they became natural instruments and there are applications and user friendly implementations of those instruments that are of main interest. Differential algebraic equations, Cryptography, Integration - these are the topics of non-commutative Computer Algebra in this issue and such development is amusing!

Even "classic" topics, such as solving of the system of equations, optimization of algorithms, homogenization and studying of singularities are presented in this issue, but the level is much higher than ten years ago. In some sense the modern Computer Algebra has achieved the micro level: it helps us to study invisible details and the development of the software itself is also on the microscopic level: invisible (for the user) contributions for the essential improvements of the main algorithms. It is Fine Computer Algebra!

Svetlana Cojocaru, Victor Ufnarovski

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

A New Attempt On The F_5 Criterion

Christian Eder

Abstract

Faugère's criterion used in the F_5 algorithm is still not understand and thus there are not many implementations of this algorithm. We state its proof using syzygies to explain the normalization condition of a polynomial. This gives a new insight in the way the F_5 criterion works.

1 Introduction

In 2002 Faugère published a new algorithm for computing Gröbner bases [2]. He found a new criterion defining when a set is a Gröbner basis. This criterion can be used to compute Gröbner bases of ideals generated by arbitrary finite sequences of polynomials.

In the F_5 algorithm additional data on the polynomials is used to detect redundant critical pairs in advance to avoid computations of zero. In this paper we give a proof of the F_5 criterion with some easier and more general arguments.

The plan of the paper is as follows: In section 2 we give briefly the basic definitions for Gröbner basis computations as well as the main terminology for the F_5 criterion. In section 3 we prove the main theorem of this paper, the F_5 criterion.

©2008 by Ch. Eder

2 Basic Notations

Throughout this paper ring always means a commutative ring with identity, $\mathbb{N}$ is the set of non-negative integers. $\mathbb{K}$ denotes the ground field, $\mathbb{K}[\underline{x}]$ the polynomial ring over $\mathbb{K}$ in the finite sequence of n variables $\underline{x} = (x_1, \ldots, x_n)$. $\mathcal{T}$ denotes the set of terms of $\mathbb{K}[\underline{x}]$. Furthermore let $<$ be a total order on $\mathbb{K}[\underline{x}]$.

2.1 Gröbner basics

We briefly give the main definitions needed to define a Gröbner basis in a characterization useful for our purposes.

Definition 2.1. Let $t = x_1^{\alpha_1} \cdots x_n^{\alpha_n} \in \mathcal{T}$ where $\alpha_i \in \mathbb{N}$ for $i \in \{1, \ldots, n\}$. The *total degree of* t is defined to be $\deg(t) = \sum_{i=1}^{n} \alpha_i$.

Let

$$f = \sum_{\alpha} c_{\alpha_1,\ldots,\alpha_n} x_1^{\alpha_1} \cdots x_n^{\alpha_n} = \sum_{\alpha} c_\alpha x^\alpha \in \mathbb{K}[\underline{x}] \backslash \{0\}$$

where $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n$, $c_\alpha \in \mathbb{K}$, and only finitely many $c_\alpha \neq 0$. The *total degree of* f is defined as $\deg(f) = \max\{\alpha_1 + \cdots + \alpha_n \mid c_{\alpha_1,\ldots,\alpha_n} \neq 0\}$. Furthermore writing $f = c_\alpha x^\alpha + c_\beta x^\beta + \cdots + c_\gamma x^\gamma$, $x^\alpha > x^\beta > \cdots > x^\gamma$ in a unique way as a sum of non-zero terms we define

(a) the head monomial of f: $\mathrm{HM}(f) = c_\alpha x^\alpha$,

(b) the head term of f: $\mathrm{HT}(f) = x^\alpha$,

(c) the head coefficient of f: $\mathrm{HC}(f) = c_\alpha$.

Definition 2.2. Let $f, g \in \mathbb{K}[\underline{x}] \backslash \{0\}$. The *S-polynomial* of f and g is defined to be

$$\mathrm{Spol}(f, g) = \mathrm{HC}(g) \frac{\tau}{\mathrm{HT}(f)} f - \mathrm{HC}(f) \frac{\tau}{\mathrm{HT}(g)} g$$

where $\tau = \mathrm{lcm}(\mathrm{HT}(f), \mathrm{HT}(g))$.

Definition 2.3. Let $P \subset \mathbb{K}[\underline{x}]$ be a finite set, $0 \neq f \in \mathbb{K}[\underline{x}]$, and $t \in \mathcal{T}$. A representation

$$f = \sum_{p \in P} \lambda_p p,$$

where $\lambda_p \in \mathbb{K}[\underline{x}]$, $p \in P$ is called a *t-representation of f w.r.t. P* if for all $p \in P$ such that $\lambda_p \neq 0$ $\mathrm{HT}(\lambda_p p) \leq t$.

For $t = \mathrm{HT}(f)$ a t-representation of f is called a *standard representation*

There are a lot of equivalent characterizations of Gröbner bases, see for example [1]. The one we need in this paper is stated next.

Theorem 2.4. *Let $G = \{g_1, \ldots, g_{n_G}\}$ be a finite subset of $\mathbb{K}[\underline{x}]$ with $0 \notin G$. If for all $f \in I = \langle g_1, \ldots, g_{n_G} \rangle$ f has a standard representation, then G is a Gröbner basis of I.*

Proof. See [1]. □

2.2 F_5 basics

We extend given definitions and state new terminology needed to understand Faugère's F_5 criterion.

In the following let $F = (f_1, \ldots, f_m)$ be a sequence of polynomials in $\mathbb{K}[\underline{x}]$, $\mathbb{K}[\underline{x}]^m$ denotes the free $\mathbb{K}[\underline{x}]$-module of rank m.

Definition 2.5. Let $\mathbf{g} = \sum_{k=1}^{m} g_k \mathbf{e}_k \in \mathbb{K}[\underline{x}]^m$ where $\mathbf{e}_k$ denotes the k-th standard vector in $\mathbb{K}[\underline{x}]^m$. We define the evaluation map w.r.t. F $v_F : \mathbb{K}[\underline{x}]^m \to \mathbb{K}[\underline{x}]$ such that

$$v_F\left(\sum_{k=1}^{m} g_k \mathbf{e}_k\right) = \sum_{k=1}^{m} g_k f_k$$

An element $\mathbf{s} \in \mathbb{K}[\underline{x}]^m$ is called a syzygy w.r.t. F if $v_F(\mathbf{s}) = 0$. For $m \geq 2$ for each pair f_i, f_j with $1 \leq i < j \leq m$ we have a so-called principal syzygy w.r.t. F, $\pi_{i,j} = f_j \mathbf{e}_i - f_i \mathbf{e}_j$.

The set of all syzygies w.r.t. F is denoted $\mathrm{Syz}(F) = \ker(v_F)$ and generates an $\mathbb{K}[\underline{x}]$-module. The submodule generated by all principal syzygies w.r.t. F is denoted $\mathrm{PSyz}(F)$.

Next we define an ordering of $\mathbb{K}[\underline{x}]^m$.

Definition 2.6. Let $\mathbf{g} = \sum_{k=1}^{m} g_k \mathbf{e}_k \in \mathbb{K}[\underline{x}]^m$. The index of $\mathbf{g}$, denoted by $\mathrm{index}(\mathbf{g})$, is the smallest $i \in \{1, \ldots, m\}$ such that $g_i \neq 0$.

Suppose that $\mathbf{g}$ and $\mathbf{h} \in \mathbb{K}[\underline{x}]^m$ with $\mathrm{index}(\mathbf{g}) = i$ and $\mathrm{index}(\mathbf{h}) = j$. Then we can write $\mathbf{g} = \sum_{k=i}^{m} g_k \mathbf{e}_k$ and $\mathbf{h} = \sum_{k=j}^{m} h_k \mathbf{e}_k$.

$$\mathbf{g} \prec \mathbf{h} :\Leftrightarrow \begin{cases} i > j, \text{ or} \\ i = j \text{ and } \mathrm{HT}(g_i) < \mathrm{HT}(h_i) \end{cases}$$

For any $\mathbf{g} \in \mathbb{K}[\underline{x}]^m \backslash \{0\}$ it holds that $0 \prec \mathbf{g}$.

This leads to an extension of the terminology of head terms.

Definition 2.7. Let $\mathbf{g} \in \mathbb{K}[\underline{x}]^m \backslash \{0\}$ with $\mathrm{index}(\mathbf{g}) = i$. The module head term MHT of $\mathbf{g}$ is defined to be $\mathrm{MHT}(\mathbf{g}) = \mathrm{HT}(g_i)\mathbf{e}_i$.

Lemma 2.8. *The module ordering $\prec$ is well-founded.*

Proof. Let $\emptyset \neq P \subset \mathbb{K}[\underline{x}]^m$. The index of any element $\mathbf{p} = \sum_{i=1}^{m} p_i \mathbf{e}_i \in P$ is bounded by m, and $\leq$ is a well-ordering on the head terms of polynomials in $\mathbb{K}[\underline{x}]$. Thus

$$\begin{aligned} i_{\max} &:= \max\{\mathrm{index}(\mathbf{p}) \mid \mathbf{p} \in P\} \\ t_{\min} &:= \min\{\mathrm{HT}(p_k) \mid \mathbf{p} \in P, \mathrm{index}(\mathbf{p}) = k\} \end{aligned}$$

are well-defined. Then

$$\emptyset \neq M := \{\mathbf{p} \in P \mid \mathrm{index}(\mathbf{p}) = i_{\max}, \mathrm{HT}(p_{i_{\max}}) = t_{\min}\}$$

is the set of minimal elements of P. □

Next we define a connection between polynomials in $\mathbb{K}[\underline{x}]$ and module elements in $\mathbb{K}[\underline{x}]^m$. These are the main concepts for the F_5 criterion.

Definition 2.9.

(a) A *labeled polynomial* r is an element $r = (u\mathbf{e}_k, p)$ such that $u \in \mathcal{T}$, $p \in \mathbb{K}[\underline{x}]$.

(b) The *signature of* r is defined by $\mathcal{S}(r) := u\mathbf{e}_k$, the *polynomial of* r by $\text{poly}(r) := p$, and the *index of* r by $\text{index}(r) := k$. For a finite set G of labeled polynomials we define $\text{poly}(G) := \{\text{poly}(r) | r \in G\}$.

(c) If $t \in \mathcal{T}$ then $tr := (tu\mathbf{e}_k, tp)$, if $c \in \mathbb{K}$ then $cr := (u\mathbf{e}_k, cp)$.

(d) r is called *admissible w.r.t.* F if there exists a $\mathbf{g} \in \mathbb{K}[\underline{x}]^m \backslash \{0\}$ such that $v_F(\mathbf{g}) = p$ and $\text{MHT}(\mathbf{g}) = \mathcal{S}(r)$.

(e) Let G be a finite set of labeled admissible w.r.t. F polynomials. r is called *normalized w.r.t.* G if $u \notin \text{HT}(\langle\{p_i \in \text{poly}(G) \mid \text{index}(r_i) > \text{index}(r)\}\rangle)$.

(f) Let (r_1, r_2) be a pair of labeled polynomials with $\tau = \text{lcm}\big(\text{HT}(\text{poly}(r_1)), \text{HT}(\text{poly}(r_2))\big)$, $\tau_i = \frac{\tau}{\text{HT}(\text{poly}(r_i))}$ for $i \in \{1, 2\}$. Then (r_1, r_2) is called normalized if $\tau_1 r_1$, $\tau_2 r_2$ are normalized and $\mathcal{S}(\tau_2 r_2) \prec \mathcal{S}(\tau_1 r_1)$. For a pair of labeled polynomials (r_1, r_2) where r_1, r_2 are admissible to $\mathbf{g}_1, \mathbf{g}_2$ respectively, we define the S-polynomial to be

$$\text{Spol}(r_1, r_2) := \big(\text{MHT}(\tau_1\mathbf{g}_1 - \tau_2\mathbf{g}_2), c_2\tau_1\text{poly}(r_1) - c_1\tau_2\text{poly}(r_2)\big),$$

where $c_i = \text{HC}(\text{poly}(r_i))$ for $i \in \{1, 2\}$.

Corollary 2.10. *If r_1 and r_2 are admissible labeled polynomials w.r.t. F then Spol(r_1, r_2) is an admissible labeled polynomial w.r.t. F.*

3 F_5 criterion

Next we prove the F_5 criterion stated in [2]. For this purpose we need some lemmata and more notations.

Convention 3.1. In the following let $F = (f_1, \ldots, f_m)$, $f_i \in \mathbb{K}[\underline{x}]$, $G = \{r_1, \ldots, r_{n_G}\}$ a set of labeled admissible w.r.t. F polynomials such that

$$\{(\mathbf{e}_1, f_1), \ldots, (\mathbf{e}_m, f_m)\} \subset G.$$

Let $p_i = \text{poly}(r_i)$ for all $i \in \{1, \ldots, n_G\}$, $\text{poly}(G) = \{p_1, \ldots, p_{n_G}\}$.

When we write *admissible* we always mean *admissible w.r.t. F*.

Lemma 3.2. *If an admissible labeled polynomial $r = (ue_k, p)$ with $\mathbf{g} \in \mathbb{K}[\underline{x}]^m$ such that $MHT(\mathbf{g}) = ue_k$ and $v_F(\mathbf{g}) = p$ is non-normalized w.r.t. G then there exists $\mathbf{s} \in PSyz(F)$ with index$(\mathbf{s}) = k$ such that $MHT(\mathbf{g} - \mathbf{s}) \prec MHT(\mathbf{g})$.*

Proof. If $r = (ue_k, p)$ is non-normalized then there exists $r_i \in G$ with $p_i = \sum_{\ell=k_0}^m \lambda_\ell f_\ell \in G$ where $\lambda_\ell \in \mathcal{K}[\underline{x}]$ such that $\text{index}(r_i) = k_0 > k$ and $\text{HT}(p_i) \mid u$. So there exists $t \in \mathcal{T}$ such that $t\text{HT}(p_i) = u$. Let $\mathbf{z} := p_i\mathbf{e}_k - f_k \sum_{\ell=k_0}^m \lambda_\ell \mathbf{e}_\ell \in \text{Syz}(F)$. Now we can rewrite

$$\begin{aligned}
p_i\mathbf{e}_k - f_k \sum_{\ell=k_0}^m \lambda_\ell \mathbf{e}_\ell &= \left(\sum_{\ell=k_0}^m \lambda_\ell f_\ell\right)\mathbf{e}_k - f_k \sum_{\ell=k_0}^m \lambda_\ell \mathbf{e}_\ell \\
&= \lambda_{k_0} f_{k_0}\mathbf{e}_k - \lambda_{k_0} f_k \mathbf{e}_{k_0} + \lambda_{k_0+1} f_{k_0+1}\mathbf{e}_k - \\
&\quad - \lambda_{k_0+1} f_k \mathbf{e}_{k_0+1} + \cdots + \lambda_m f_m \mathbf{e}_k - \lambda_m f_k \mathbf{e}_m \\
&= \lambda_{k_0}\pi_{k,k_0} + \lambda_{k_0+1}\pi_{k,k_0+1} + \cdots + \lambda_m \pi_{k,m} \\
&= \sum_{\ell=k_0}^m \lambda_\ell \pi_{k,\ell}
\end{aligned}$$

where $\pi_{v,w}$ denotes the principal syzygy $f_w\mathbf{e}_v - f_v\mathbf{e}_w \in \text{PSyz}(F)$ for $v < w \in \{1, \ldots, m\}$. Set $\mathbf{s} = t\mathbf{z} \in \text{PSyz}(F)$. By construction $\text{index}(\mathbf{s}) = k$, $\text{MHT}(\mathbf{g} - \mathbf{s}) \prec \text{MHT}(\mathbf{g})$ and $v_F(\mathbf{g} - \mathbf{s}) = v_F(\mathbf{g})$. □

Lemma 3.3. *Let $r = (ue_k, p)$ and let $\tau_1, \tau_2 \in \mathcal{T}$. If $\tau_2\tau_1 r$ is normalized w.r.t. $G \Rightarrow \tau_1 r$ is normalized w.r.t. G.*

Proof. Let $\tau_2\tau_1 r = (\tau_2\tau_1 u e_k, \tau_2\tau_1 p)$ be normalized w.r.t. G.

Assume for contradiction that $\tau_1 r = (\tau_1 u e_k, \tau_1 p)$ is non-normalized w.r.t. G. Then there exists $r_0 \in G$ such that $\mathrm{index}(r_0) > k$ and $\mathrm{HT}(p_0) \mid \tau_1 u$. Then $\mathrm{HT}(p_0) \mid \tau_2\tau_1 u$ and it follows that $\tau_2\tau_1 r$ is non-normalized w.r.t. G, which contradicts our assumption that $\tau_2\tau_1 r$ is normalized w.r.t. G. □

The following definition of the ordering $\lessdot$ for representations of a labeled polynomials is similar to the one Faugère has stated in [2]. For a deeper insight we refer to [3].

Definition 3.4. Let $f \in I = \langle g_1, \ldots, g_{n_G}\rangle$. Then we define

$$\mathcal{R}_f := \Bigg\{ (\lambda, \sigma) \in \mathbb{K}[\underline{x}]^{n_G} \times \mathrm{Sym}_{n_G} \mid f = \sum_{i=1}^{n_G} \lambda_i p_{\sigma(i)}, \mathcal{S}(\lambda_1 r_{\sigma(1)}) \succeq \ldots \cdots \succeq \mathcal{S}(\lambda_{n_G} r_{\sigma(n_G)}) \Big\}$$

to be the set of *labeled representations of f w.r.t. G* where Sym_{n_G} denotes the symmetric group on $\{1, \ldots, n_G\}$. Next we define the ordering $\lessdot$ on labeled representations of f w.r.t. G.

For two labeled representations of f w.r.t. G, (λ, σ) and (λ', σ'), we define

$$\begin{aligned} \omega &= \left(\mathcal{S}(\mathrm{HT}(\lambda_1) r_{\sigma(1)}), \ldots, \mathcal{S}(\mathrm{HT}(\lambda_{n_G}) r_{\sigma(n_G)})\right), \\ \omega' &= \left(\mathcal{S}(\mathrm{HT}(\lambda'_1) r_{\sigma'(1)}), \ldots, \mathcal{S}(\mathrm{HT}(\lambda'_{n_G}) r_{\sigma'(n_G)})\right), \end{aligned}$$

respectively.
$(\lambda, \sigma) \lessdot (\lambda', \sigma')$ iff one of the following conditions holds:

(a) $\exists i$ such that $\forall 1 \leq j < i \leq n_G$: $\omega_j = \omega'_j$ and $\omega_i \prec \omega'_i$,

(b) $\forall j$: $\omega_j = \omega'_j$ and
$\max_{\ell=1,\ldots,n_G} \mathrm{HT}(\lambda_\ell p_{\sigma(\ell)}) < \max_{\ell'=1,\ldots,n_G} \mathrm{HT}(\lambda'_{\ell'} p_{\sigma'(\ell')})$,

(c) $\forall j$: $\omega_j = \omega'_j$,
$\max_{\ell=1,\ldots,n_G} \mathrm{HT}(\lambda_\ell p_{\sigma(\ell)}) = \max_{\ell'=1,\ldots,n_G} \mathrm{HT}(\lambda'_{\ell'} p_{\sigma'(\ell')}) =: t$
and $\#\{\ell \mid \mathrm{HT}(\lambda_\ell p_{\sigma(\ell)}) = t\} < \#\{\ell' \mid \mathrm{HT}(\lambda_{\ell'} p_{\sigma(\ell')}) = t\}$.

Lemma 3.5. *The ordering $\lessdot$ is well-founded.*

Proof. See [3], Lemma 3.17. □

Lemma 3.6. *Let $f \in I = \langle g_1, \ldots, g_{n_G} \rangle$. Let (λ, σ) be a minimal labeled representation for f w.r.t. G. Then for all indices $v \in \{1, \ldots, m\}$:*

$$\#\{k \mid (\lambda_k, \sigma(k)) \in (\lambda, \sigma), \lambda_k \neq 0, \mathit{index}(r_{\sigma(k)}) = v\} \leq 1.$$

Proof. We can assume σ to be the identity by renumbering G, $f = \sum_{i=1}^{m} \lambda_i g_i$. Choose $v \in \{1, \ldots, m\}$ arbitrarily. Denote

$$\begin{aligned} I &= \{k \mid (\lambda_k, \mathrm{id}(k)) \in (\lambda, \mathrm{id}), \mathrm{index}(r_k) = v\}, \\ I_< &= \{k \mid (\lambda_k, \mathrm{id}(k)) \in (\lambda, \mathrm{id}), \mathrm{index}(r_k) < v\} \text{ and} \\ I_> &= \{k \mid (\lambda_k, \mathrm{id}(k)) \in (\lambda, \mathrm{id}), \mathrm{index}(r_k) > v\}. \end{aligned}$$

Assume that $\#I > 1$.
Each $r_k \in G$ is admissible w.r.t. F, i.e. $g_k = \sum_{j=v}^{m} \eta_{k,j} f_j$ with $\eta_{k,j} \in \mathbb{K}[\underline{x}]$.
Thus we get a new representation of f:

$$\begin{aligned} f &= \sum_{i=1}^{m} \lambda_i g_i = \sum_{i \in I} \lambda_i g_i + \sum_{j \notin I} \lambda_j g_j \\ &= \sum_{i \in I_<} \lambda_i g_i + \left(\sum_{j \in I} \lambda_j \eta_{j,v} \right) f_v + \sum_{j \in I} \lambda_j \sum_{k=v+1}^{m} \eta_{j,k} f_k + \sum_{\ell \in I_>} \lambda_\ell g_\ell \end{aligned}$$

This new labeled representation $(\lambda', \sigma') \prec_{\mathrm{lex}} (\lambda, \mathrm{id})$: The first $\#I_<$ components remained unchanged, then there is one component $\lambda'_v f_v$ where $\lambda'_v = \sum_{j \in I} \lambda_j \eta_{j,v}$. By construction

$$\begin{aligned} &\mathcal{S}(\mathrm{HT}(\lambda'_v) r_{\sigma'(v)}) = \\ &= \max\{\mathcal{S}(\mathrm{HT}(\lambda_k) r_k) \mid (\lambda_k, \mathrm{id}(k)) \in (\lambda, \mathrm{id}), \mathrm{index}(r_k) = v\}, \end{aligned}$$

where $\mathrm{poly}(r_{\sigma'(v)}) = f_v$. So the signatures of the first $\#I_< + 1$ components of both labeled representations are equal. But the $\#I_< + 2$th component of (λ, id) has index v, as we assumed that there are at least two such components, whereas the $\#I_< + 2$th component of (λ', σ') has an index $< v$.

Thus we received a contradiction of the minimality of (λ, id) w.r.t. $\preccurlyeq$. □

Remark 3.7. Note that a labeled representation w.r.t. G does not restrict the number of possible representations of an element $f \in I$. A labeled representation w.r.t. G just orders the components of the corresponding representation of f so that representations can be compared w.r.t. $\preccurlyeq$.

Definition 3.8. Let $t \in \mathcal{T}$, (λ, σ) be a labeled representation w.r.t. G of a labeled polynomial r. W.l.o.g. we can assume $\sigma = \mathrm{id}$. Then (λ, id) is called a *t-representation of r* if

$$p = \sum_{\ell=1}^{n_G} \lambda_\ell p_\ell$$

such that for all components $\mathrm{HT}(\lambda_\ell p_\ell) \leq t$ and $\mathcal{S}(\mathrm{HT}(\lambda_\ell) r_\ell) \preceq \mathcal{S}(r)$.

Theorem 3.9. *If for all pairs (r_i, r_j) normalized w.r.t. G $\mathrm{Spol}(r_i, r_j)$ has a t-representation where $t < \mathrm{lcm}\big(\mathrm{HT}(p_i), \mathrm{HT}(p_j)\big)$ then $\mathrm{poly}(G)$ is a Gröbner basis of $I = \langle p_1, \ldots, p_n \rangle$.*

Proof. Let $f \in I$. Then f has a labeled representation (λ, σ) w.r.t. G. W.l.o.g. we can assume $\sigma = \mathrm{id}$ such that $f = \sum_{\ell=1}^{n_G} \lambda_\ell p_\ell$. By Lemma 3.5 let us assume (λ, id) to be a minimal labeled representation of f w.r.t. G.

If there is a component $(\lambda_k, \mathrm{id}(k)) \in (\lambda, \mathrm{id})$ such that $\lambda_k r_k$ is not normalized w.r.t. G then there exists a principal syzygy $\mathbf{s}$ by Lemma 3.2. $\lambda_k r_k$ is admissible, i.e. there exists $\mathbf{g} \in \mathbb{K}[\underline{x}]^m$ such that $\mathrm{MHT}(\mathbf{g}) = \mathcal{S}(\mathrm{HT}(\lambda_k) r_k)$ and $v_F(\mathbf{g}) = \lambda_k p_k$. So we can construct $\mathbf{g} - \mathbf{s}$ with $\mathrm{MHT}(\mathbf{g} - \mathbf{s}) \prec \mathrm{MHT}(\mathbf{g})$ and $\lambda_k r_k$ admissible to $\mathbf{g} - \mathbf{s}$. This gives a

labeled representation (λ', σ') of f w.r.t. G such that $(\lambda', \sigma') \lessdot (\lambda, \mathrm{id})$. This contradicts the minimality of (λ, id) w.r.t. $\lessdot$, so every $\lambda_k r_k$ such that $(\lambda_k, \mathrm{id}(k)) \in (\lambda, \mathrm{id})$ is normalized w.r.t. G.

By Lemma 3.6 there are no two components with the same index in (λ, id), i.e. all $\lambda_k r_k$ have different signatures.

Assume that there exist components $(\lambda_k, \mathrm{id}(k))$ such that $\mathrm{HT}(\lambda_k p_k) = t'$ where $t' \geq \mathrm{HT}(f)$. Note that $\#\{\ell \mid \mathrm{HT}(\lambda_\ell p_\ell) = t'\} \geq 2$. Choose two such components $(\lambda_i, \mathrm{id}(i))$, $(\lambda_j, \mathrm{id}(j))$.

Let $\tau = \mathrm{lcm}\big(\mathrm{HT}(p_i), \mathrm{HT}(p_j)\big)$, $\tau_i = \frac{\tau}{\mathrm{HT}(p_i)}$, and $\tau_j = \frac{\tau}{\mathrm{HT}(p_j)}$. Then $\tau \mid t'$, $\tau_i \mid \mathrm{HT}(\lambda_i)$, and $\tau_j \mid \mathrm{HT}(\lambda_j)$.

Define $m_i = \mathrm{HM}(\lambda_i)$ and $m_j = \frac{\mathrm{HC}(\lambda_i)}{\mathrm{HC}(\lambda_j)}\mathrm{HM}(\lambda_j)$. Now we compute

$$\begin{aligned} m_i p_i - m_j p_j &= \mathrm{HC}(\lambda_i)\mathrm{HT}(\lambda_i)p_i - \mathrm{HC}(\lambda_i)\mathrm{HT}(\lambda_j)p_j \\ &= \mathrm{HC}(\lambda_i)\left(\frac{\tau_i t'}{\tau}p_i - \frac{\tau_j t'}{\tau}p_j\right) \\ &= \mathrm{HC}(\lambda_i)\frac{t'}{\tau}\mathrm{Spol}(p_i, p_j). \end{aligned}$$

Since $\lambda_i r_i$ and $\lambda_j r_j$ are normalized w.r.t. G it follows with Lemma 3.3 that also $\tau_i r_i$ and $\tau_j r_j$ are normalized w.r.t. G.

Thus we get a new labeled representation (λ'', σ'') of f w.r.t. G:

$$\begin{aligned} f &= \sum_{\ell=1}^{n_G} \lambda_\ell p_\ell = \lambda_i p_i + \lambda_j p_j + \sum_{\substack{\ell=1 \\ \ell \neq i,j}}^{n_G} \lambda_\ell p_\ell \\ &= m_i p_i + \big(\lambda_i - \mathrm{HT}(\lambda_i)\big)p_i - m_j p_j - \frac{\mathrm{HC}(\lambda_i)}{\mathrm{HC}(\lambda_j)}\big(\lambda_j - \mathrm{HT}(\lambda_j)\big)p_j \\ &\quad + \left(1 + \frac{\mathrm{HC}(\lambda_i)}{\mathrm{HC}(\lambda_j)}\right)\lambda_j p_j + \sum_{\substack{\ell=1 \\ \ell \neq i,j}}^{n_G} \lambda_\ell p_\ell. \end{aligned}$$

As $\mathrm{Spol}(r_i, r_j)$ has a t-representation $\mathrm{Spol}(p_i, p_j) = \sum_{\ell=1}^{n_G} \eta_\ell p_\ell$ such that

$$\mathrm{HT}(\eta_\ell p_\ell) < \mathrm{HT}(\mathrm{lcm}(\mathrm{HT}(p_i), \mathrm{HT}(p_j)) \quad \text{and}$$
$$\mathcal{S}(\mathrm{HT}(\eta_\ell) r_\ell) \preceq \mathcal{S}(\mathrm{Spol}(r_i, r_j)).$$

It follows that $(\lambda'', \sigma'') \preccurlyeq (\lambda, \mathrm{id})$. This contradicts the minimality of (λ, id). □

Acknowledgement. I would like to thank John Perry for many useful discussions.

References

[1] T.Becker, V.Weispfennig, and H.Kredel. *Gröbner Bases.* Springer Verlag, 1993.

[2] J.C. Faugère. *A new efficient algorithm for computing Gröbner bases without reduction to zero(F5).* Symbolic and Algebraic Computation, Proc. Conferenz ISSAC 2002, pp. 75–83, 2002.

[3] Stegers, Till. *Faugère's F5 Algorithm Revisited.* Thesis for the degreee of Diplom-Mathematiker, 2005.

Christian Eder,

Fachbereich Mathematik, TU Kaiserslautern,
Postfach 3049, 67653
Kaiserslautern, Germany
E–mail: *ederc@rhrk.uni − kl.de*

Received November 7, 2007

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

Some Remarks on Blowing-Ups in a Computer Algebra System

Anne Frühbis–Krüger

Abstract

The aim of this short note is to provide detailed information on how to compute blowing ups in various settings by means of a computer algebra system. All examples are formulated using the system SINGULAR[5].

1 Introduction

Although the notion of blowing up is ubiquitous in algebraic geometry and singularity theory, the most common use of it is a blowing-up at a point. Consequently tools to compute blowing-ups at a point are implemented in a wide range of computer algebra systems.

For more complex applications on the other hand like e.g. studying examples of flops, considering Nash modifications or desingularization, restricting the choice of centers to sets of points is not an option: Considering the even the simplest example of a flop, the centers of the blow-ups for the small resolutions will be Weil divisors[1]; considering resolution of singularities, the singular locus has, in general, no reason to be zerodimensional. Thus implementations of blowing ups also need to cover the general case, which will be described in section 2 of this article. Allowing centers to be higher dimensional, we encounter problems of efficiency, which may be countered to some extent by considering embedded blowing up, whenever the centers are non-singular and we are using a covering by affine charts; allowing centers to even

©2008 by A. Frühbis-Krüger

[1] see section 3 for the step-by-step calculations in SINGULAR

be singular, as is necessary for Nash modifications, on the other hand, blocks this alternative for computations.

In sections 3 and 4, we consider examples of applications of the various variants mentioned – each time including a step by step SINGULAR-input and output for treating the respective task.

As usual for computational methods for algebraic geometry in characteristic zero, we assume the ground field to be $\mathbb{C}$ for all reasoning, although actual computations are performed over the rationals.

I should like to thank Lawrence Ein, Priska Jahnke, Patrick Popescu-Pampu, David Ploog and Ivo Radloff, whose questions on how to compute specific examples of blowing ups by means of a computer algebra system, led to this collection of remarks. I am also indebted to the Freie Universität Berlin and the ICTP Trieste for the invitations that provided the opportunity to meet the previously mentioned colleagues.

2 Implementation of Blowing Ups

2.1 Blowing Up a Scheme

Recall that given a noetherian scheme W and a closed subscheme Y of W, corresponding to a ideal sheaf $\mathcal{I}$ on W, the blowing-up of W along Y is defined as

$$\pi : \tilde{W} := Proj(\bigoplus_{d\geq 0} \mathcal{I}^d) \longrightarrow W.$$

This is a birational map, which is an isomorphism away from Y, i.e. $\tilde{W} \setminus \pi^{-1}(Y) \cong W \setminus Y$; the inverse image $\pi^{-1}(Y)$ is a Cartier divisor on $\tilde{W}$, called the exceptional divisor of the blowing up[2]. Unfortunately, this description is not well-suited for explicit calculations and implementations, which usually require objects to be represented by polynomial data, i.e. a free presentation or a set of generators of an ideal over a polynomial ring or a quotient thereof. To achieve this description, a

[2]For further details including the universal property of blowing up see any textbook on algebraic geometry like [6], II or [9]

convenient way is to pass to a covering of W by finitely many affine open sets. Then the initial situation in an affine chart $U \subset W$ can be formulated as follows: Working over the basering $A := \mathcal{O}_W(U)$, which is a polynomial ring or a quotient thereof, we can describe the center Y in this chart by the ideal $I = \mathcal{I}(U)$ which we now assume to be generated by $f_0, \ldots, f_s$. Then we can consider the graded morphism of A-algebras

$$\begin{array}{rcl} \varphi : A[y_0, \ldots, y_s] & \longrightarrow & A[t \cdot f_0, \ldots, t \cdot f_s] \subset A[t] \\ y_i & \longmapsto & t \cdot f_i \end{array}$$

whose image is obviously isomorphic to $\bigoplus_{d \geq 0} I^d$. Hence $\pi^{-1}(U)$, as a subset of $U \times \mathbb{P}^s$, allows a description by $A[y_0, \ldots, y_s]/ker(\varphi)$ which is precisely what we needed for computational purposes. The exceptional divisor of the blowing up, i.e. the inverse image of the center Y, then corresponds to the ideal $I \cdot A[y_0, \ldots, y_s]/ker(\varphi)$.

The ideal $ker(\varphi)$ unfortunately involves $s+1$ additional variables and hence it seems at first glance that e.g. the number of variables in the resolution process might constantly rise making effective standard bases calculations virtually impossible after just a few blow-ups. But passing once again to an appropriate affine covering helps us keep the number of variables sufficiently low; more precisely we use the usual covering of the newly introduced $\mathbb{P}^s$ by the sets $D(y_i)$, $1 \leq i \leq s$. Obviously, this is a trade-off and causes the calculations to branch which easily leads to duplicate calculations on the intersections of several charts and significantly increases the amount of data to be stored. On the other hand, treating several charts at the same time on different processors/computers allows a parallelization of e.g. the resolution algorithm which is only rarely possible for computational tasks in commutative algebra and may improve the performance.

Nevertheless, the disadvantages of passing to open covers largely outweigh the benefits in general and it is therefore desirable to keep the number of charts as low as possible e.g. by dropping charts which do not contribute any new information to the considered task.

Example 1 [Blowing up of $\mathbb{A}^3$ at the origin]

```
ring r=0,(t,x(1..3),y(1..3)),(dp(1),dp);
                      // A^3 x P^2 plus extra variable
                      // for elimination of t
                      // as usual in computation of
                      // preimage of zero
ideal I=y(1)-t*x(1),
        y(2)-t*x(2),
        y(3)-t*x(3);  // ideal describing map
ideal IW=eliminate(I,t);
                      // elimination step
ring r2=0,(x(1..3),y(1..3)),dp;
                      // A^3 x P^2
ideal IW=imap(r,IW);  // transfer the ideal to this ring
IW;                   // ideal of variety after blowing up
--> IW[1]=x(3)*y(2)-x(2)*y(3)
--> IW[2]=x(3)*y(1)-x(1)*y(3)
--> IW[3]=x(2)*y(1)-x(1)*y(2)

subst(IW,y(1),1);     // what does the chart
                      // D(y(1)) look like
--> _[1]=x(3)*y(2)-x(2)*y(3)
--> _[2]=-x(1)*y(3)+x(3)
--> _[3]=-x(1)*y(2)+x(2)
// As expected this is isomorphic to an A^3, getting rid
// of x(2) and x(3) using generators _[2] and _[3].
// The exceptional divisor is described by x(1)=0 in
// this chart.
//
// The same observations hold in the other charts,
// as the whole situation is blind to exchanging the roles
// of the variables x(i).
```

As already mentioned, we would like to blow up at more general

centers than point. Here is one such example:

Example 2 [Blowing up $\mathbb{A}^3$ in $V(z, x^2 + y^2 - 1)$]

```
ring r=0,(t,x(1..3),y(1..2)),(dp(1),dp);
                      // A^3 x P^1 plus extra variable
                      // for elimination of t
                      // as usual in computation of
                      // preimage of zero
ideal I=y(1)-t*x(3),
        y(2)-t*(x(1)^2+x(2)^2-1);
                      // ideal describing map
ideal IW=eliminate(I,t);
                      // elimination step
ring r2=0,(x(1..3),y(1..2)),dp;
                      // A^3 x P^1
ideal IW=imap(r,IW);  // transfer the ideal to this ring
IW;                   // ideal of variety after blowing up
--> IW[1]=x(1)^2*y(1)+x(2)^2*y(1)-x(3)*y(2)-y(1)

subst(IW,y(1),1);     // what does the chart
                      // D(y(1)) look like
--> _[1]=x(1)^2+x(2)^2-x(3)*y(2)-1
// This is obviously non-singular, but we cannot get rid
// of a fourth variable.
subst(IW,y(2),1);     // and D(y(2)) -->
_[1]=x(1)^2*y(1)+x(2)^2*y(1)-x(3)-y(1)
// Here we can get rid of x(3).
```

This sequence of computational steps to compute blowing ups is available as commands `blowUp` and `blowUp2` in SINGULAR, see the SINGULAR online manual for a description.

2.2 Notions of Transforms

Considering blowing ups, we are usually not just dealing with a single scheme, but additionally with one or several subschemes of it which are

also affected by the blowing up. This leads to the task of computing the total and the strict transform of such a subscheme (or depending on the context also the weak or the controlled transform).

To this end, let us recall that the total transform of a closed subscheme $Z \subset W$ (corresponding to an ideal sheaf $\mathcal{J}_Z \subset \mathcal{O}_W$) under the blowing up π is just the inverse image $\pi^{-1}(Z)$ and can hence be computed as

$$\mathcal{J}_{Z,total} = \mathcal{J}_Z \cdot \mathcal{O}_{\tilde{W}}.$$

Let us further recall that the strict transform $\tilde{Z}$ of Z is obtained by blowing up Z at the center given by $\mathcal{I} \cdot \mathcal{O}_Z$ according to the following commutative diagram:

$$\begin{array}{ccc} \tilde{Z} & \stackrel{i}{\hookrightarrow} & \tilde{W} \\ \downarrow & & \downarrow \pi \\ Z & \stackrel{i}{\hookrightarrow} & W. \end{array}$$

In the affine case, we can also obtain the strict transform of Z by forming the closure of $\pi^{-1}(Z \setminus (Z \cap Y))$ in $\tilde{W}$. By using again the previously introduced affine covering of $\tilde{W}$, this allows us to compute the strict transform from the total transform using a saturation[3]:

$$J_{Z,strict} = (J_Z \cdot \mathcal{O}_{\tilde{W}}(U) : I_E^{\infty}),$$

where J_Z is used as short hand notation of $\mathcal{J}_Z(U)$ and I_E denotes the ideal of the exceptional divisor of π on our chart U. Geometrically this saturation can be interpreted as dropping all components of the total transform which lie in the exceptional divisor or coincide with it.

For resolving singularities by the algorithmic approaches of Villamayor [1] and Encinas/Hauser [3] two other notions of transforms come into play which amount to ending the above saturation prematurely after a fixed number of ideal quotient computations. In the case

[3] Saturating, i.e. iterating the ideal quotient until it stabilizes (noetherian ring), is available in most computer algebra systems for algebraic geometry and commutative algebra as a built-in command. It is usually an expensive operation, but not if we are saturating by a principal ideal. For a detailed discussion of saturation and its geometric interpretation see [2] or [4]

of the weak transform, this number of iterations is the maximal order of the ideal J_Z prior to the blowing up (at a center contained in the locus of maximal order); geometrically speaking, the weak transform originates from the total transform by removing all copies of the exceptional divisor, but keeping the lower-dimensional components which lie inside the exceptional divisor. In the case of the controlled transform, the number of iterations is prescribed by the resolution algorithm and can be anything between 1 and the number of iterations for the weak transform.

Example 3 [Different notions of transforms of a space curve] Continuing the first example, we now consider the space curve $V(xz, yz, x^3 - y^4) \subset \mathbb{A}^3$ and compute its different transforms:

```
ideal J=x(1)*x(3),x(2)*x(3),x(1)^4-x(2)^3;
                                // ideal of space curve
ideal Jtotal=J,IW;              // ideal of total transform,
                                // before passing to charts
ideal Jt1=subst(Jtotal,y(1),1);
                                // ideal in chart D(y(1))
Jt1;
--> Jt1[1]=x(1)*x(3)
--> Jt1[2]=x(2)*x(3)
--> Jt1[3]=-x(1)^4+x(2)^3
--> Jt1[4]=x(3)*y(2)-x(2)*y(3)
--> Jt1[5]=-x(1)*y(3)+x(3)
--> Jt1[6]=-x(1)*y(2)+x(2)
// Obviously we can get rid of x(2) and x(3) by appropriate
// reductions. As the heuristic to do this automatically is
// lengthy, it is not printed here. Instead, we use our
// knowledge of what we want to replace:
ideal Jt2=subst(Jt1,x(3),x(1)*y(3));
                                // replace x(3) by x(1)*y(3)
                                // according to Jt1[5]
Jt2=subst(Jt2,x(2),x(1)*y(2));
                                // replace x(2) by x(1)*y(2)
```

```
                                // according to Jt1[6]
Jt2=interred(Jt2);              // drop unnecessary
                                // generators
Jt2;
--> Jt2[1]=x(1)^2*y(3)
--> Jt2[2]=x(1)^3*y(2)^3-x(1)^4

ring chart=0,(x(1),y(2),y(3)),dp;
ideal Jt2=imap(r2,Jt2);         // only keep necessary
                                // variables for this chart,
                                // by passing to appropriate
                                // ring
Jt2;
--> Jt2[1]=x(1)^2*y(3)
--> Jt2[2]=x(1)^3*y(2)^3-x(1)^4

ideal Jctrl1=quotient(Jt2,ideal(x(1)));
                                // controlled transform,
                                // #iterations=1
Jctrl1;
--> Jctrl[1]=x(1)*y(3)
--> Jctrl[2]=x(1)^2*y(2)^3-x(1)^3

ideal Jweak=quotient(quotient(Jt2,ideal(x(1))),
                     ideal(x(1)));
                                // weak transform
                                // #iterations=2, because
                                // ord(J)=ord(x(1)*x(3))=2
Jweak;
--> Jweak[1]=y(3)
--> Jweak[2]=x(1)*y(2)^3-x(1)^2

LIB"elim.lib";                  // saturation is in elim.lib
ideal Jstr=sat(Jt2,x(1));
                                // strict transform
```

```
Jstr;
[1]:
   _[1]=y(3)                    // ideal of strict transform
   _[2]=y(2)^3-x(1)
[2]:
   3                            // number of iterations when
                                // stabilizing
```

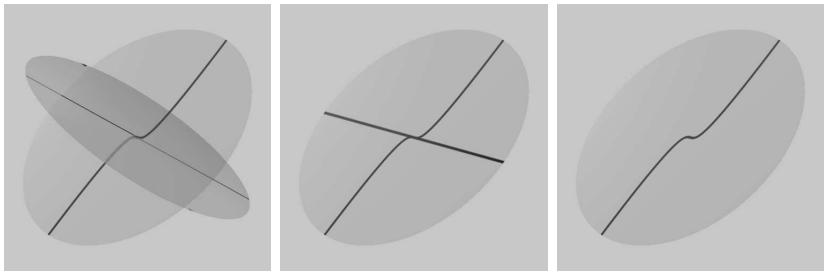


Figure 1. These three pictures illustrate the different notions of transforms computed in the example 3. From left to right, we see total transform, weak transform and strict transform. Due to technical reasons with the imaging tool surf, one additional plane is shown in each image: V(y(3)), of which we know that it contains the two curves.

The above considerations about the definition and the computation of the strict transform of a subscheme also imply that there are two equivalent ways of computing the blowing up of a scheme which can be embedded into a $\mathbb{A}^k$ or a $\mathbb{P}^k$ at a non-singular center:

- blowing up the scheme directly
- considering the scheme as embedded in an appropriate $\mathbb{A}^k$ (possibly after passing to an affine covering), blowing up the $\mathbb{A}^k$ and computing the strict transform

The first variant can be quite expensive in the elimination of the additional variables – depending on how complicated the equations for

the variety are[4]. The latter variant has to deal with a larger amount of data due to the affine covering; the expensive part here is the saturation which is, on the other hand, cheaper than a general saturation, because we saturate by a principal ideal.

If the center itself is singular, however, blowing up the ambient space is not an option, because the ambient space has no reason to be smooth after such a blowing up as the following example shows:

Example 4 [Blowing up at a singular center]

```
ring r=0,(t,x(1..3),y(1..2)),(dp(1),dp);
                                  // again A^3 x P^1 plus
                                  // additional variable t
ideal I=y(1)-t*x(1)*x(2),y(2)-t*x(3);
                                  // center is the union
                                  // of the x- and y-axes
ideal IW=eliminate(I,t);
IW;
--> IW[1]=x(1)*x(2)*y(2)-x(3)*y(1)
subst(IW,y(2),1);                 // chart D(y(2))
--> _[1]=x(1)*x(2)-x(3)*y(1)
// this obviously has a singular point at the origin!
```

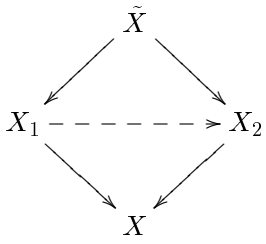
3 Application 1: A Flop

As the first application, we now consider the simplest example of a flop. It is however beyond the scope of this short note to explain exactly what a flop is; a good reference for the minimal model program (the context, in which the notions of flips and flops arose) and the precise definitions of flips and flops can be found in [7]. For our purpose here, which is

[4] Standard basis calculations w.r.t. elimination orderings are never really cheap, but (like standard basis calculations in general) they tend to become very expensive, if we are dealing with many variables and the equations are not of a particularly simple form.

an illustration on how to use a computer algebra system to deal with a concrete example, it suffices to consider the following situation:

Let $X = V(x_1x_2 - x_3x_4) \subset \mathbb{A}^4_{\mathbb{C}}$, which obviously has one isolated singularity at the origin. Blowing up X at this singular point, we obtain $\tilde{X} \subset \mathbb{A}^4_{\mathbb{C}} \times \mathbb{P}^3_{\mathbb{C}}$ whose exceptional locus turns out to be a $\mathbb{P}^1 \times \mathbb{P}^1$. On the other hand, blowing up at the Weil divisor $V(x_1, x_3) \subset X$ which is not $\mathbb{Q}$-Cartier, we obtain $X_1 \subset \mathbb{A}^4_{\mathbb{C}} \times \mathbb{P}^1_{\mathbb{C}}$. Analogously, blowing up at $V(x_2, x_4)$ yields another scheme X_2. Here it is interesting to observe that X_1 and X_2 may alternatively be constructed by blowing down one (and the other resp.) of the $\mathbb{P}^1_{\mathbb{C}}$ of the exceptional divisor of $\tilde{X}$. The resulting rational map $X_1 \dashrightarrow X_2$ is the well-known simplest example of a flop[5]. As a diagram, we have the following situation



The following sequence of SINGULAR commands mimics the main steps of the above construction:

```
ring r=0,(t,x(1..4)),dp;   // A^4 plus extra variable t,
                           // for checking singular locus,
                           // Weil divisors, not Cartier;
                           // extra variable t will be
                           // needed later on -
                           // explained there
ideal I=x(1)*x(2)-x(3)*x(4);
LIB"sing.lib";             // slocus is in sing.lib
std(slocus(I));            // ideal of singular locus
```

[5]Considered abstractly, the two varieties X_1 and X_2 are isomorphic in this very simple example. This is a coincidence and does not occur in general.

```
--> _[1]=x(4)
--> _[2]=x(3)
--> _[3]=x(2)
--> _[4]=x(1)              // as we expected

ideal IDiv1=x(1),x(3);     // first divisor
ideal IDiv2=x(2),x(4);     // second divisor
// as both V(IDiv1) and V(IDiv2) are obviously reduced,
// irreducible, closed subsets of A^4, it remains to check
// - V(IDiv1) contained in V(I) and of codimension 1
// - analogously for V(IDiv2) -- not shown here
size(reduce(I,std(IDiv1)));
                           // zero if ideal containment
                           // test succeeds
--> 0
dim(std(I))-dim(std(IDiv1));
                           // codimension of V(IDiv1)
                           // in V(I)
                           // remark: extra variable t
                           // causes both dimensions
                           //         to be raised by 1
                           //         which does not
                           //         affect this result
--> 1
// Hence we have prime divisors on X, which are of course
// Weil divisors.
//
// We now check that V(IDiv1) cannot be Q-Cartier, i.e.
// that there cannot be a power of V(IDiv1) which is
// locally principal. To this end, we pass to the
// localization at the only singular point. - If it
// fails there, this is sufficient to show that V(IDiv1)
// is not Q-Cartier.
ring rlocal=0,(t,x(1..4)),(dp(1),ds);
                           // ds ordering is local!
```

```
def I=imap(r,I);
ideal Itest=I,x(1),x(3)*t-1;
reduce(1,std(Itest));      // 0, if some power of x(3) is
                           // in I+<x(1)>; 1 otherwise
--> 1
Itest=I,x(3),x(1)*t-1;     // as above, but roles of x(1)
                           // and x(3) exchanged
reduce(1,std(Itest));
--> 1
// This implies that V(IDiv1) cannot be Q-Cartier.
//
// After checking the claimed properties of $X$, we now
// return to blowing up and blowing down.
ring r2=0,(t,x(1..4),u(1..4)),(dp(1),dp);
                           // for A^4 x P^3 + extra
                           // variable
ideal Ipt=x(1)*x(2)-x(3)*x(4),u(1)-t*x(1),u(2)-t*x(2),
                              u(3)-t*x(3),u(4)-t*x(4);
                           // ideal for blowing up point
ideal IWeil1=x(1)*x(2)-x(3)*x(4),u(1)-t*x(1),u(3)-t*x(3);
                           // for Weil-divisor V(x_1,x_3)
ideal IXtop=eliminate(Ipt,t);
                           // I blownup at point
size(IXtop);
--> 9                      // 9 generators

std(IXtop+ideal(x(1..4))); // ideal of except.locus
--> _[1]=x(4)
--> _[2]=x(3)
--> _[3]=x(2)
--> _[4]=x(1)
--> _[5]=u(1)*u(2)-u(3)*u(4)   // <-- P^1 x P^1 in P^3

ideal IX1=eliminate(IWeil1,t);
                           // I blown up at first
```

```
                              // Weil divisor
IX1;
--> IX1[1]=x(3)*u(1)-x(1)*u(3)
--> IX1[2]=x(2)*u(1)-x(4)*u(3)
--> IX1[3]=x(1)*x(2)-x(3)*x(4)

// Now we blow down contracting the P^1 specified by
// V(u(2),u(4)) to a point
// In general this can only be done, if the corresponding
// blow-up map is known - it is then a preimage
// calculation.
// Here, however, the situation is so simple that we can
// see that this contraction amounts to a projection to
// A^4 x P^1, i.e. to eliminating u(2) and u(4)
eliminate(IXtop,u(2)*u(4));
--> _[1]=x(3)*u(1)-x(1)*u(3)
--> _[2]=x(2)*u(1)-x(4)*u(3)
--> _[3]=x(1)*x(2)-x(3)*x(4)
// As expected this is the same as IX1.
```

4 Application 2: A Nash Modification

As the second application, we consider Nash modifications, which are known to locally be blowing ups. In particular, we shall consider two examples, only one of which is a complete intersection.

Let us recall that given a reduced separated algebraic scheme X of pure dimension r, a Nash modification $p : \tilde{X} \longrightarrow X$ is defined by the following process (which, for simplicity of presentation, we describe only in the special case that $X \subset \mathbb{A}^n$ and that X is defined by $\langle f_1, \ldots, f_m \rangle$):

Denoting by G_r^n the Grassmannian of r-planes in $\mathbb{A}^n$, by $Reg(X)$ the complement of the singular locus of X, and by $T_{X,x}$ the tangent

space of X at a point $x \in Reg(X)$, consider the morphism

$$\eta : Reg(X) \longrightarrow X \times G_r^n \quad (1)$$
$$x \longmapsto (x, T_{X,x}). \quad (2)$$

$\tilde{X}$ is then defined as the closure of the image of η in $X \times G_r^n$ and the Nash modification $p : \tilde{X} \longrightarrow X$ is the first projection. By a result of Nobile [8], p can locally be formulated as a blowing up at a center $J \subset \mathcal{O}_X$ where J is generated[6] by appropriate elements g_β of the ideal of $n-r$ minors of the Jacobian matrix $(\frac{\partial f_i}{\partial x_j})_{1 \leq i \leq m, 1 \leq j \leq n}$. More precisely, for each irreducible component X_i of X, we can find

- an $(n-r) \times n$ submatrix of this matrix of which at least one $n-r$ minor does not vanish on X_i (The minors of this submatrix will be denoted by $M_{i,\beta}$ where β indicates the columns involved in this particular minor)
- a global section $0 \neq h_i \in \Gamma(X, \mathcal{O}_X)$ vanishing along all other components X_j, $1 \leq j \leq d$, $i \neq j$.

The generators of J are then

$$g_\beta = \sum_{i=1}^{d} h_i M_{i,\beta}$$

where β runs through all $n-r$ tuples of column indices of the Jacobian matrix.

In the case of a complete intersection, the Jacobian matrix does not have more than $n-r$ rows, thus making the row selection and the h_i in the above construction unnecessary and implying that J is just the ideal of the singular locus.

```
// As a first example we consider a complete intersection:
ring r=0,(t,x,y,z,a(1..5)),(dp(1),dp);
                        // A^3 plus additional variables
```

[6] Under the above simplifications of $X \subset \mathbb{A}^k$ and $I(X) = \langle f_1, \ldots, f_m \rangle$

```
                        // for blowing up
ideal I1=x^2-y^2-z^4,yz;
                        // 2 lines V(x+y,z),V(x-y,z)
                        // 2 parabolas V(x-z^2,y),
                        //             V(x+z^2,y)
                        // all meeting in (0,0,0)

// Center of blowing up is singular locus
LIB"sing.lib";          // slocus is in sing.lib
ideal sL=mstd(slocus(I1))[2];
                        // minimal number of generators
                        // of ideal of singular locus
size(sL);
--> 5
ideal blow1=I1,a(1)-t*sL[1],a(2)-t*sL[2],a(3)-t*sL[3],
               a(4)-t*sL[4],a(5)-t*sL[5];
ideal Elim1=eliminate(blow1,t);
                        // do the blowing up

// Now we would like to check that we have indeed
// 2 single and a double point in the preimage
// of V(x,y,z)
LIB"primdec.lib";
primdecGTZ(Elim1+ideal(x,y,z));
--> [1]:                // double point
-->   [1]:              // the component
-->      _[1]=a(5)^2
-->      _[2]=a(3)
-->      _[3]=a(4)
-->      _[4]=a(1)
-->      _[5]=z
-->      _[6]=y
-->      _[7]=x
-->   [2]:              // its radical
-->       ....          // output omitted
```

```
-->[2]:                // single point
-->   [1]:             // the component
-->      _[1]=a(3)-a(5)
-->      _[2]=a(2)
-->      _[3]=a(4)
-->      _[4]=a(1)
-->      _[5]=z
-->      _[6]=y
-->      _[7]=x
-->   [2]:            // its radical
-->      ....         // output omitted

-->[3]:                // single point
-->   [1]:             // the component
-->      _[1]=a(3)+a(5)
-->      _[2]=a(2)
-->      _[3]=a(4)
-->      _[4]=a(1)
-->      _[5]=z
-->      _[6]=y
-->      _[7]=x
-->   [2]:            // its radical
-->      ....         // output omitted

// As a second example, we determine the center
// in the non-complete-intersection case:
ring r=0,(x,y,z),dp;
                       // A^3
ideal I2=xz,yz,x^2-y^4;
                       // 1 line V(x,y)
                       // 2 parabolas V(x-y^2,z)
                       //         and V(x+y^2,z)
                       // all meeting in (0,0,0)
list comps=minAssGTZ(I2);
```

```
                      // minimal associated primes
                      // of our ideal --
                      // coincides here obviously
                      // with prim. decomp.
matrix M[3][3]=diff(I2,x),diff(I2,y),diff(I2,z);
print(M);             // Jacobian matrix of I2
--> z,0,2*x,
--> 0,z,-4*y^3,
--> x,y,0
// To determine the appropriate generators
// of our center, we need to construct the
// g_beta=\sum h_i M_beta,i
// Step 1: define the three submatrices
//         and their respective ideals of minors:
matrix M12[2][3]=M[1,1..3],M[2,1..3];
matrix M13[2][3]=M[1,1..3],M[3,1..3];
matrix M23[2][3]=M[2,1..3],M[3,1..3];
ideal min12=minor(M12,2);
ideal min13=minor(M13,2);
ideal min23=minor(M23,2);
// Step 2: check for each component, which minors
//         do not vanish along the component
size(reduce(min12,std(comps[1])));
--> 0                 // all minors of M12 vanish
                      // along first component
size(reduce(min12,std(comps[2])));
--> 0                 // as before
size(reduce(min12,std(comps[3])));
--> 1                 // this is the good component
/* Important Aside:
   The numbering of the components in the output
   of minAssGTZ resp. primdecGTZ is not fixed and
   often changes when recomputing the decomp. */
...
// ... repeating these steps for the other ideals of
```

```
//     minors, we obtain:
//     comp1: M13 or M23
//     comp2: M13 or M23
//     comp3: M12

// Step 3: determine the h_i:
// check which generators of intersection of compi and
// compj does not vanish identically on compk
ideal inter12=intersect(comps[1],comps[2]);
reduce(inter12,std(comps[3]));
                       // study comp1 \cap comp2
                       // and comp3 ==> h3
--> _[1]=z
--> _[2]=0
--> _[3]=0
poly h3=inter12[1];    // inter12[1] does not
                       // vanish identically on comp1
...
// ... repeating these steps for the other two
//     components, we obtain:
//     h1 = y^2-x
//     h2 = y^2+x
//     h3 = z

// Step 4: combine information to obtain the center:
ideal center=(h1 * min13) + (h2 * min13) + (h3 * min12);
center;
--> center[1]=z^3
--> center[2]=x*z^2
--> center[3]=x*y*z
--> center[4]=x^2*y
--> center[5]=x^3
--> center[6]=y^3*z-x*y*z
--> center[7]=x*y^3-x^2*y
```

```
// Blowing up with this center now provides the
// desired Nash modification.
```

References

[1] Bravo,A., Encinas,S., Villamayor,O.:*A Simplified Proof of Desingularisation and Applications*, Rev. Math. Iberoamericana 21 (2005), 349–458.

[2] Decker,W., Lossen,C.: *Computing in Algebraic Geometry - A quick start using* SINGULAR, Algorithms and Computation in Mathematics 16, Springer Verlag (2006).

[3] Encinas,S., Hauser,H.: *Strong resolution of singularities in characteristic zero*, Comment. Math. Helv. 77 (2002), 821–845.

[4] Greuel,G.-M., Pfister,G.: *A* SINGULAR *Introduction to Commutative Algebra*, Springer (2002).

[5] Greuel,G.-M., Pfister,G., Schönemann,H.: SINGULAR 3.0, `http://www.singular.uni-kl.de/`.

[6] Hartshorne,R.: *Algebraic Geometry*, Springer (1977).

[7] Kollar,J., Mori,Sh.: *Birational geometry of algebraic varieties*, Cambridge Univ. Press (1998).

[8] Nobile,A.: *Some Properties of the Nash Blowing-Up*, Pac. J. Math. 60 (1975), 297–306.

[9] Shafarevich,I.: *Basic Algebraic Geometry*, Springer (1977).

Anne Frühbis–Krüger,

E–mail: *anne@math.uni – hannover.de*

Received January 9, 2008

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

Computing one of Victor Moll's irresistible integrals with computer algebra

Christoph Koutschan [*] Viktor Levandovskyy

Abstract

We investigate a certain quartic integral from V. Moll's book "Irresistible Integrals" and demonstrate how it can be solved by computer algebra methods, namely by using non-commutative Gröbner bases. We present recent implementations in the computer algebra systems SINGULAR and MATHEMATICA.

1 Introduction

The integral [1, (7.2.1)] which we deal with is

$$F(a,m) = \int_0^\infty \frac{1}{(x^4+2ax^2+1)^{m+1}}\,\mathrm{d}x. \tag{1}$$

From mathematical expert's view this integral might not look very challenging, and of course, Moll is able to compute its solution by hand. But nevertheless his computations are involved and need some quite special knowledge. From the software point of view both MAPLE and MATHEMATICA fail to evaluate (1) due to the presence of two parameters a, m (if they are set to concrete numbers the evaluation can be immediately done). We present computer algebra methods that allow to compute this integral in a purely automatic fashion with no expert's knowledge involved. The first approach is based on D-module theory whereas the second one follows Zeilberger's "holonomic systems approach". Our aim is to bring together these two directions since the underlying theoretical principles are identical. Moreover, we aim at a self-contained presentation of theory and algorithms.

©2008 by Ch. Koutschan, V. Levandovskyy

[*] supported by the Austrian FWF grant P20162

2 Preliminaries

Let $\mathbb{K}$ be a field. For the integration, we will need to deal with some special non-commutative algebras. It is common to define $\mathbb{K}$-algebras via generators and relations, especially if they have infinite dimension over $\mathbb{K}$. Let $\mathbf{X} = \{x_1, \ldots, x_n\}$ be a finite set of symbols, then by $\mathbb{K}\langle\mathbf{X}\rangle$ one denotes a free associative algebra. Given a finite set $R = \{r_1(x), \ldots, r_m(x)\} \subset \mathbb{K}\langle\mathbf{X}\rangle$, writing for an associative $\mathbb{K}$-algebra $A = \mathbb{K}\langle\mathbf{X} \mid R\rangle$ means $A \cong \mathbb{K}\langle\mathbf{X}\rangle/I_R$, where $I_R := \langle R\rangle$ is the two-sided ideal of $\mathbb{K}\langle\mathbf{X}\rangle$ generated by R. The elements of both R and I_R are often regarded as *relations* of A. This way of defining algebras has its roots in group theory, where a similar construction is performed. Since we are dealing with the algebras, which are in many sense close to commutative - in particular, each pair of variables is connected by some relation - we use shorter notation when writing the defining relations R. Namely, if we do not mention any relation between a pair of variables, these variables do commute.

Given two algebras $A = \mathbb{K}\langle\mathbf{X}\rangle/I$ and $B = \mathbb{K}\langle\mathbf{Y}\rangle/J$, we identify $A \otimes_{\mathbb{K}} B$ with the algebra $\mathbb{K}\langle\mathbf{X}, \mathbf{Y} \mid I + J\rangle$, since in $A \otimes_{\mathbb{K}} B$ any element $a \otimes 1$ for $a \in A$ commutes with every element $1 \otimes b$ for $b \in B$.

In this article we deal with Weyl algebras, shift algebras and their tensor products over a field $\mathbb{K}$ of characteristic 0. Given a natural number $n \geq 1$ and a set of variables (also called *coordinates*) $\mathbf{X} = \{x_1, \ldots, x_n\}$, we construct first a commutative ring $R_n = \mathbb{K}[\mathbf{X}]$. We identify a polynomial $f \in R_n$ with the operator of multiplication by f. Given n natural operators $\partial_i := \partial_{x_i} = \frac{\partial}{\partial x_i}$ of partial differentiation with respect to the coordinate variable x_i, we define the algebra of linear partial differential operators with *polynomial* coefficients (also called the n-th Weyl algebra) to be

$$A_n := \mathbb{K}\langle x_1, \ldots, x_n, \partial_1, \ldots, \partial_n \mid \{\partial_j x_i = x_i \partial_j + \delta_{ij}{}^1 \; \forall 1 \leq i, j \leq n\}\rangle.$$

Note, that the action of an operator on a function from an appropriate function space will be denoted by $\bullet$, while $\cdot$ will be used for

[1] δ_{ij} denotes the Kronecker symbol

multiplication in operator algebras. Thus,

$$\partial_{x_i} \bullet f(x_1, \ldots, x_n) := \frac{\partial f(x_1, \ldots, x_n)}{\partial x_i}.$$

To each coordinate x_i we can also associate a partial shift operator s_i, which acts on a function $f(x_1, \ldots, x_i, \ldots, x_n)$ as

$$s_i \bullet f(x_1, \ldots, x_i, \ldots, x_n) := f(x_1, \ldots, x_i + 1, \ldots, x_n).$$

Given n such operators, we define the algebra of linear partial shift operators with *polynomial* coefficients (also called the n-th shift algebra) to be

$$S_n := \mathbb{K}\langle x_1, \ldots, x_n, s_1, \ldots, s_n \mid \{s_j x_i = x_i s_j + \delta_{ij} s_j \; \forall 1 \leq i, j \leq n\}\rangle.$$

Both A_n and S_n share many nice properties, for instance

- $\{x_1^{\alpha_1} \ldots x_n^{\alpha_n} \partial_1^{\beta_1} \ldots \partial_n^{\beta_n} \mid \alpha_i, \beta_i \in \mathbb{N}_0\}$ is a $\mathbb{K}$-basis for A_n,
- $\{x_1^{\alpha_1} \ldots x_n^{\alpha_n} s_1^{\beta_1} \ldots s_n^{\beta_n} \mid \alpha_i, \beta_i \in \mathbb{N}_0\}$ is a $\mathbb{K}$-basis for S_n,
- A_n and S_n are Noetherian domains (in particular, every module is finitely generated and there are no zero divisors),
- for any $i, j \in \mathbb{N}$, $A_{i+j} \cong A_i \otimes_{\mathbb{K}} A_j$ and $S_{i+j} \cong S_i \otimes_{\mathbb{K}} S_j$,
- there is a Gröbner basis theory for both types of algebras, very close to the theory in the commutative case, see e.g. [11, 8].

Picking some nice $\mathbb{K}$-basis for an algebra as above, we call these basis elements *monomials*. As one can see, the monomials are in one-to-one correspondence with their *exponent vectors*, say, $(\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_n) \in \mathbb{N}^{2n}$. Hence, we can define a *monomial ordering* on A_n as follows (the cases of S_m and $A_n \otimes_{\mathbb{K}} S_m$ are completely analogous, see e.g. [8])

Definition 1 *A monomial ordering on A_n is a total ordering $\prec$ on the set of monomials, which satisfies for all $\alpha = (\alpha_x, \alpha_\partial), \beta = (\beta_x, \beta_\partial), \gamma = (\gamma_x, \gamma_\partial) \in \mathbb{N}^{2n}$*
(1) $\alpha \prec \beta \Rightarrow x^{\alpha_x}\partial^{\alpha_\partial} \prec x^{\beta_x}\partial^{\beta_\partial}$ *and*
(2) $x^{\alpha_x}\partial^{\alpha_\partial} \prec x^{\beta_x}\partial^{\beta_\partial} \;\Rightarrow\; x^{\alpha_x+\gamma_x}\partial^{\alpha_\partial+\gamma_\partial} \prec x^{\beta_x+\gamma_x}\partial^{\beta_\partial+\gamma_\partial}$.

Since every polynomial $f \in A_n$ can be uniquely written as a sum of monomials times coefficients, we call the highest monomial of f with respect to a given ordering the *leading monomial* of f. We denote the latter by $\mathrm{lm}(f)$.

Note that there is another requirement we need to be fulfilled in our class of algebras, namely $1 \prec x_i, \partial_j, s_k \ \ \forall\, i,j,k$, that is the monomial ordering is a *well-ordering*.

We say that $x^{\alpha_x}\partial^{\alpha_\partial}$ *divides* $x^{\beta_x}\partial^{\beta_\partial}$, if $\alpha_i \leq \beta_i$ for all i in the range. Note, that this just means, that there exist $\gamma \in \mathbb{N}^{2n}$ and $r \in A_n$, such that $x^{\beta_x}\partial^{\beta_\partial} = x^{\alpha_x}\partial^{\alpha_\partial} \cdot x^{\gamma_x}\partial^{\gamma_\partial} + r$ with $r = 0$ or $\mathrm{lm}(r) \prec x^{\alpha_x}\partial^{\alpha_\partial}$.

Definition 2 *Let $\prec$ be a monomial ordering on A_n and $G \subset A_n$ a finite set of polynomials. Let I be a left ideal, generated by G. G is called a left Gröbner basis of I if and only if for any $f \in I \setminus \{0\}$ there exists $g \in G$ satisfying* $\mathrm{lm}(g) \mid \mathrm{lm}(f)$.

Given a finite set of generators of a left ideal L, there is *Buchberger's algorithm* for computing a left Gröbner basis of L (see e.g. [11, 8]).

Let $M_n := R_n \setminus \{0\}$, then M_n is a multiplicatively closed subset of both A_n and S_n. Hence, using the algebraic formalism of "localization" and the fact that M_n is an Ore set, we can pass from A_n (resp. S_n) to its "Ore localization", that is an algebra $(A_n)_{M_n}$ (resp. $(S_n)_{M_n}$). In the language of systems of operator equations $(A_n)_{M_n}$ (resp. $(S_n)_{M_n}$) stays for the algebra of linear partial differential (resp. shift) operators with *rational* coefficients. The algebras with rational coefficients appear very often in practical applications. They - as well as the Weyl and the shift algebra - are special cases of Ore algebras. We refer to [10, 4, 3, 8] for more details on these algebras, their properties as well as computational aspects and Gröbner bases.

3 Integration with D-modules

Define $f := f(a,x) = x^4 + 2ax^2 + 1 \in \mathbb{K}[x,a]$, then we have to integrate the function $f^{-(m+1)}$ with respect to x.

D-module theory stands for "the theory of differential modules" and encompasses systems of linear partial differential equations with

polynomial and rational coefficients. One of the most important algorithms, obtained with D-module theory (see [11] and references therein for the full picture) is the algorithm for computing the s-parametric annihilator of $f \in \mathbb{K}[x_1, \ldots, x_n]$ for a symbolic s. That is, it is possible to compute a set of operators $\{P \in D[s] : P \bullet f^s = 0\} =: \mathrm{Ann}_{D[s]} f^s$, which is indeed a left ideal in the algebra $D[s] := A_n[s] = A_n \otimes_{\mathbb{K}} \mathbb{K}[s]$ (for historical reasons D stands for some n-th Weyl algebra). Additionally, there is an algorithm for computing $\mathrm{Ann}_D f^\lambda$ for any $\lambda \in \mathbb{C}$, which uses the previously mentioned one.

In the case of the integral (1), $f \in \mathbb{K}[x, a] = R_2$. Then $D = A_2 = \mathbb{K}\langle x, a, \partial_x, \partial_a | \partial_x x = x\partial_x + 1, \partial_a a = a\partial_a + 1\rangle$ is the 2nd Weyl algebra and $D[s] = A_2 \otimes_{\mathbb{K}} \mathbb{K}[s]$. First, we are going to compute the left ideal $L := \mathrm{Ann}_{D[s]} f^s \subset A_2 \otimes_{\mathbb{K}} \mathbb{K}[s]$ for $s := -(m+1)$ being symbolic. L corresponds to the system of linear partial differential equations in operators $\partial_x, \partial_a, s$ with coefficients in $\mathbb{K}[x, a]$, which has f^s as a solution. That is $\forall h \in L,\ h \bullet f^s = 0$.

In order to compute a system I of such equations for the function $F(a, s)$, we use Theorem 5.5.1 of [11], which states the following: let J be the right ideal of A_2, generated by all partial differential operators, corresponding to variables, with respect to which we perform integration (in our case this is just ∂_x, but the Theorem, as well as the whole approach, which goes back to Takayama [13, 12], holds for the multiple variable case too). Then

$$I = (L + J) \ \cap \ \big(\mathbb{K}\langle a, \partial_a \mid \partial_a a = a\partial_a + 1\rangle \otimes_{\mathbb{K}} \mathbb{K}[s]\big),$$

where the latter algebra is a natural $\mathbb{K}$-subalgebra of $A_2 \otimes_{\mathbb{K}} \mathbb{K}[s]$.

In general the sum of a left and of a right ideals carries no left or right structure. However, in the setting we work with a right ideal is very special one and, as we can see, there is a structure of left ideal on the intersection I of the sum of ideals above with a subalgebra.

We work with the special $\mathbb{K}$-basis of the algebra $A_2 \otimes_{\mathbb{K}} \mathbb{K}[s]$, namely $\{\partial_x^\alpha x^\beta a^\gamma \partial_a^\delta s^\epsilon \mid \alpha, \beta, \gamma, \delta, \epsilon \in \mathbb{N}_0\}$. In particular, each monomial of any polynomial in a left Gröbner basis of $L = \mathrm{Ann}_{D[s]} f^s$ is presented in this form. Moreover, we compute a left Gröbner basis G of L with respect

to an ordering which eliminates x, ∂_x, i.e., any monomial containing ∂_x or x is bigger than one, which does not contain both of them.

Instead of summing L with the right ideal J (generated by ∂_x), we perform the right reduction of G with respect to J, what amounts to just skipping any monomial of every polynomial of G, if it is of the form $\partial_x^\alpha x^\beta a^\gamma \partial_a^\delta s^\epsilon$, where $\alpha \geq 1$. We may throw such a monomial away, because it belongs to the ideal J. After such a procedure we get a new set of polynomials G', where ∂_x does not appear. Since we used elimination ordering for both x and ∂_x for G and, moreover, monomials containing ∂_x are not present in G', it remains to pick those elements of G', which do not contain x. These elements then belong to the algebra $\mathbb{K}\langle a, \partial_a \mid \partial_a a = a\partial_a + 1\rangle[s]$ and, according to the Theorem 5.5.1 of [11], they generate the left ideal I we are looking for.

Now we illustrate the computation for the integral (1) with the computer algebra system Singular:Plural [5, 6]. This system has a library for computations with algebraic D-modules `dmod.lib` [9], which we are going to use.

```
LIB "dmod.lib";       // load the library for D-modules
ring r = 0,(a,x),dp;  // define a commutative ring
poly f = x^4 + 2*a*x^2 + 1;
def A = Sannfs(f); // A is a ring with the result object
                   // in it
setring A;
```

In the ring `A`, which stays for $D[s]$ (see above), there is an object called `LD` of the type `ideal`, which is the s-parametric annihilator ideal $L = \mathrm{Ann}_{D[s]} f^s$ as before. Its Gröbner basis consists of four operators

$$
\begin{aligned}
&2x^2\partial_a + 2a\partial_a - x\partial_x \ , \\
&x^3\partial_x - 2a^2\partial_a + ax\partial_x - 4x^2 s + 2\partial_a \ , \\
&4a^2\partial_a^2 - x^2\partial_x^2 - 8a\partial_a s + 4a\partial_a - 4\partial_a^2 + 4x\partial_x s - x\partial_x \ , \\
&2a^2 x\partial_a + ax^2\partial_x - 4axs - 2x\partial_a + \partial_x .
\end{aligned}
$$

Now, we change the order of variables into $\partial_x, x, a, \partial_a, s$; adjust the non-commutative relations respectively; set the monomial ordering,

eliminating ∂_x, x and compute the left Gröbner basis of the ideal L, mapped from the ring `A`.

```
ring rr = 0, (Dx, x, a, Da, s), (a(1,1),dp);
matrix @D[5][5];
@D[1,2] = -1; @D[3,4] = 1;
def RR = nc_algebra(1,@D);
setring RR; // a new non-commutative ring
map M = A, a, x, Da, Dx, s; // map from A to RR using names
ideal LD = M(LD); // the image of LD in the new ring
LD = groebner(LD); // left Groebner basis of LD
```

At this stage we have to perform the addition of the left ideal L with the right ideal J, generated by ∂_x and intersect the result with the subalgebra $\mathbb{K}\langle a, \partial_a \mid \partial_a a = a\partial_a + 1\rangle[s]$. We go along the lines, described above.

```
ideal DD = Dx ;
ideal J = rightNF(LD,DD); // reduce with Dx from the right
ideal NJ = nselect(J,1,2); // see below
NJ = groebner(NJ); // left Groebner basis of NJ
```

We achieve these operations by computing the right normal forms of generators of left Gröbner basis of `LD` with respect to ∂_x. Invoking `nselect` command we select those generators, which do not include the variables from 1 to 2, that is ∂_x and x. As we can see, the ideal called `NJ`, which stay for I as above, is a principal ideal indeed. It is generated by the polynomial

$$4a^2\partial_a^2 - 4\partial_a^2 - 8a\partial_a s + 4a\partial_a - 4s - 1.$$

Depending on the monomial ordering used, sometimes an invertible element might appear as a factor.

Now we substitute s by $-m-1$ and rewrite some terms, giving back the answer: the integral $F(a, m)$ is annihilated by the left principal

ideal of the algebra $\mathbb{K}\langle a, \partial_a \mid \partial_a a = a\partial_a + 1\rangle[m]$, which is generated by the operator

$$4(a-1)(a+1)\partial_a^2 + 4a(2m+3)\partial_a + (4m+3).$$

Of course, it is not yet a final answer, but an important part of it. In the next sections we show how we come to a closed form for the integral.

4 Holonomic systems and ∂-finite functions

We will now demonstrate how the symbolic evaluation of integrals like (1) can be performed in a different, more general framework, following D. Zeilberger's "holonomic systems approach" [14]. This theory was extended by F. Chyzak [2, 3, 4] who introduced the concept of ∂-finite functions and proposed Ore algebras to describe them. Moreover he implemented the underlying algorithms in the MAPLE package MGFUN.

For the construction of an Ore algebra, one starts with a commutative algebra like $\mathbb{K}[\mathbf{X}]$ or $\mathbb{K}(\mathbf{X})$ and adds one or several Ore extensions. These extensions introduce operators that necessarily commute with each other but usually do not commute with the variables $\mathbf{X}$. This setting is quite general (see e.g. [10]) and here we consider only special operators, namely the partial derivatives ∂_x, ∂_a and the shift s_m. For example, the Ore algebra that we will use here is $\mathbb{O} = \mathbb{K}(x,a,m)[\partial_x; 1, \partial_x][\partial_a; 1, \partial_a][s_m; s_m, 0]$. This algebra can also be realized as an Ore localization $(A_2 \otimes_{\mathbb{K}} S_1)_B$ where $A_2 = \mathbb{K}\langle x, a, \partial_x, \partial_a \mid \partial_x x = x\partial_x + 1, \partial_a a = a\partial_a + 1\rangle$, $S_1 = \mathbb{K}\langle m, s_m \mid s_m m = m s_m + s_m\rangle$, and B is the multiplicatively closed set $\mathbb{K}[x,a,m] \setminus \{0\} \subset A_2 \otimes_{\mathbb{K}} S_1$.

A function f is called ∂-finite w.r.t. an Ore algebra $\mathbb{K}(\mathbf{X})[\mathbf{P}; ., .]$ if the $\mathbb{K}(\mathbf{X})$-vector space spanned by all $(\mathbf{X^m P^n}) \bullet f$ is finite-dimensional over $\mathbb{K}(\mathbf{X})$. The following example will clarify this definition.

We want to find Ore operators in $\mathbb{O}$ that annihilate the integrand $g(x,a,m) = 1/(x^4 + 2ax^2 + 1)^{m+1}$. First observe that g is hyperexponential in x and a, i.e., $\frac{\partial_x \bullet g}{g}$ and $\frac{\partial_a \bullet g}{g}$ are rational functions in x and a

respectively, e.g.,

$$\frac{\partial_x \bullet g(x,a,m)}{g(x,a,m)} = \frac{(-m-1)\left(4x^3+4ax\right)}{x^4+2ax^2+1}.$$

Moreover g is hypergeometric in m which means that $\frac{s_m \bullet g}{g} = \frac{g(x,a,m+1)}{g(x,a,m)}$ is a rational function in m. Hence we can compute first order annihilating operators for $g(x,a,m)$ in $\operatorname{Ann}_{\mathbb{O}} g = \{R \in \mathbb{O} \mid R \bullet g = 0\}$. Note that we use the term "annihilator" for any ideal of annihilating operators.

```
g = 1/(x^4+2*a*x^2+1)^(m+1);
ann = Annihilator[g, {S[m], Der[a], Der[x]}]
```

$$\begin{aligned}\{&(x^4+2ax^2+1)\partial_x + 4mx^3 + 4x^3 + 4ax + 4amx,\\ &(x^4+2ax^2+1)\partial_a + 2mx^2 + 2x^2,\\ &(x^4+2ax^2+1)s_m - 1\}\end{aligned}$$

An easy check ensures that these polynomials indeed constitute a Gröbner basis of the left ideal they generate. Moreover all leading monomials have degree 1; hence the corresponding ideal is a left maximal ideal and we have $\dim_{\mathbb{K}(x,a,m)} \mathbb{O}/\operatorname{Ann}_{\mathbb{O}} g = 1$, so g is indeed ∂-finite w.r.t. $\mathbb{O}$.

In order to perform the integration w.r.t. x, we are interested in finding operators in $\operatorname{Ann}_{\mathbb{O}} g$ of the following special form:

$$P(a,m,\partial_a,s_m) + \partial_x Q(x,a,m,\partial_x,\partial_a,s_m),$$

since

$$\begin{aligned} 0 &= \int_0^\infty \left(P(a,m,\partial_a,s_m) + \partial_x Q(x,a,m,\partial_x,\partial_a,s_m)\right) \bullet g(x,a,m)\,\mathrm{d}x \\ &= P \bullet F(a,m) + \Big[Q \bullet g(x,a,m)\Big]_{x=0}^{x=\infty}. \end{aligned} \tag{2}$$

For this purpose we will use Takayama's algorithm [13, 12]. It is designed in a way that it computes P (the part one is mainly interested in) without computing Q. Informally spoken, one first divides out the right ideal generated by ∂_x and then eliminates x by performing a Gröbner

basis computation over a module. To this aim we have to compute in the Ore algebra $\mathbb{K}(a,m)[x][\partial_x;1,\partial_x][\partial_a;1,\partial_a][s_m;s_m,0]$ because otherwise we were not able to eliminate x. More details on Takayama's algorithm were given in the previous section.

The fact that Q is not considered at all leads to the prerequisite that the integral must have natural boundaries: An integral $\int_u^v h(x,\dots)\mathrm{d}x$ is said to have natural boundaries if $[R \bullet h]_{x=u}^{x=v} = 0$ for all operators R in the respective algebra. In particular, the inhomogeneous part in (2) will vanish. If the integral does not have natural boundaries, we can end up with an inhomogeneous equation.

If we now look at the integral (1) we see that unfortunately it does not have natural boundaries, e.g.,

$$\Big[1 \bullet g(x,a,m)\Big]_{x=0}^{x=\infty} = -1.$$

We nevertheless can apply Takayama's algorithm, but we have to use an extended version where also Q is computed. Such an extension is included in [7].

```
Takayama[ann, {x}, OreAlgebra[{x}, {Der[x], S[m], Der[a]}],
         Extended -> True]
```

$$\begin{aligned}&\{\{(-4m-4)s_m + 2a\partial_a + (4m+3),\\ &\quad (4a^2-4)\partial_a^2 + (8ma+12a)\partial_a + (4m+3)\},\\ &\{x, (-4m-4)xs_m + 2ax\partial_a + x\}\}.\end{aligned} \tag{3}$$

We are interested in the ordinary differential equation in a (the second operator). Note that it is the same as the result obtained with the first method. The corresponding Q is $(-4m-4)xs_m + 2ax\partial_a + x$. Now we verify that $[Q \bullet g]_{x=0}^{x=\infty}$ indeed vanishes although the integral does not have natural boundaries:

```
inhom = Simplify[ApplyOreOperator[%[[2,2]], g]]
```

$$x\left(x^4+2ax^2+1\right)^{-m-2}\left(-x^4+2ax^2+4m\left(ax^2+1\right)+3\right)$$

```
inhom /. x -> 0
```

$$0$$

```
Limit[inhom, x -> Infinity, Assumptions -> m >= 0]
```

$$0$$

Hence, we derived in a purely automatic fashion an ordinary differential equation in a that is satisfied by the integral.

5 Closed form solution

Up to now we did not present a closed form solution of the integral, but only a differential equation in the parameter a:

$$(4m+3)F(a,m)+4a(2m+3)F'(a,m)+4\left(a^2-1\right)F''(a,m)=0. \quad (4)$$

For solving this differential equation we can use standard tools. Since it has order 2, we need the initial values $F(0,m)$ and $F'(0,m)$:

```
in0 = Integrate[g /. a -> 0, {x, 0, Infinity},
                Assumptions -> m >= 0]
```

$$\frac{\Gamma\left(\frac{5}{4}\right)\Gamma\left(m+\frac{3}{4}\right)}{\Gamma(m+1)}$$

```
in1 = Integrate[D[g, a] /. a -> 0, {x, 0, Infinity},
                Assumptions -> m >= 0]
```

$$-\frac{2\Gamma\left(\frac{7}{4}\right)\Gamma\left(m+\frac{5}{4}\right)}{3\Gamma(m+1)}$$

We solve (4) with MATHEMATICA's command `DSolve`:

```
DSolve[{(4m+3)F[a] + 4a(2m+3)F'[a] + 4(a^2-1)F''[a] == 0,
       F[0] == in0, F'[0] == in1}, F[a], a]
```

After some simplification we end up with the final result:

$$F(a,m) = -\frac{(1+i)(-i)^m 2^{-m-1}\left(a^2-1\right)^{-\frac{m}{2}-\frac{1}{4}}\sqrt{\pi}Q_m^{(m+\frac{1}{2})}(a)}{\Gamma(m+1)},$$

where $Q_\lambda^{(\mu)}(z)$ denotes the associated Legendre function of the second kind.

Note that we computed this solution completely automatically with no necessity of human insight to the specific problem. V. Moll as an expert in the field of integrals gives the following slightly simpler solution involving Jacobi polynomials:

$$F(a,m) = 2^{-m-\frac{3}{2}}(a+1)^{-m-\frac{1}{2}}\pi P_m^{\left(m+\frac{1}{2},-m-\frac{1}{2}\right)}(a)$$

With our software [7] we can immediately prove the correctness of this solution:

```
Annihilator[Pi*JacobiP[m, m+1/2, -m-1/2, a]/2^(m+3/2)/
            (a+1)^(m+1/2),{Der[a], S[m]}]
```

$$\{-4m + (-2a)\partial_a + (4m+4)s_m - 3,$$
$$4m + \left(4a^2-4\right)\partial_a^2 + (8ma+12a)\partial_a + 3\}$$

Observe that this annihilator is exactly the same as (3). By comparing the initial values

$$F(0,0) = \frac{\pi}{2\sqrt{2}} \quad \text{and} \quad F'(0,0) = -\frac{\pi}{4\sqrt{2}}$$

we complete the proof.

6 Conclusion

We presented computer algebra methods for the automatic solution of a parametrized integral. We want to emphasize that these methods are applicable to a wide class of integration (and summation) problems.

In particular the second method works for the large class of holonomic functions. As a more challenging example let's just mention the integral

$$\int_0^\infty \frac{1}{(x^4+ax^3+bx^2+cx+d)^m}\mathrm{d}x$$

which contains more parameters, but nevertheless can be tackled in an analogous way. The problem here is only that the resulting differential equations are so involved that the standard tools are not able to find a closed form solution.

We are grateful to Victor Moll and Peter Paule for turning our attention towards this interesting problem.

We would like to acknowledge a partial financial support by the DFG Graduiertenkolleg "Hierarchie und Symmetrie in mathematischen Modellen" at RWTH Aachen, Germany, and by the Austrian FWF grant P20162 "Symbolische Integration und Spezielle Funktionen".

References

[1] George Boros and Victor H. Moll. *Irresistible Integrals*. Cambridge University Press, 2004.

[2] Frédéric Chyzak. An extension of Zeilberger's fast algorithm to general holonomic functions. In *Formal Power Series and Algebraic Combinatorics, 9th Conference*, pages 172–183. Universität Wien, 1997. Conference proceedings. Subsumed in [Chyzak, 2000c].

[3] Frédéric Chyzak. *Fonctions holonomes en Calcul formel*. PhD thesis, École polytechnique, 1998.

[4] Frédéric Chyzak and Bruno Salvy. Non-commutative elimination in Ore algebras proves multivariate identities. *Journal of Symbolic Computation*, 26:187–227, 1998.

[5] Gert-Martin Greuel, Gerhard Pfister, and Hans Schönemann. SINGULAR 3.0. A Computer Algebra System for polynomial computa-

tions. Centre for Computer Algebra, University of Kaiserslautern 2005. Available from `http://www.singular.uni-kl.de`.

[6] Gert-Martin Greuel, Viktor Levandovskyy, and Hans Schönemann. PLURAL. A SINGULAR 3.0 Subsystem for Computations with Non–commutative Polynomial Algebras. Centre for Computer Algebra, University of Kaiserslautern 2005. Available from `http://www.singular.uni-kl.de`.

[7] Christoph Koutschan. *Computer algebra algorithms for ∂-finite and holonomic functions*. PhD thesis, RISC-Linz, 2008. in preparation.

[8] Viktor Levandovskyy. *Non-commutative Computer Algebra for polynomial algebras: Gröbner bases, applications and implementation*. PhD thesis, Universität Kaiserslautern, 2005. Available from `http://kluedo.ub.uni-kl.de/volltexte/2005/1883/`.

[9] Viktor Levandovskyy and Jorge Morales. A SINGULAR 3.0 library for computations with algebraic D-modules `dmod.lib`, 2008.

[10] John C. McConnell and James C. Robson. *Noncommutative Noetherian rings. With the cooperation of L. W. Small.* Graduate Studies in Mathematics. 30. Providence, RI: American Mathematical Society (AMS), 2001.

[11] Mutsumi Saito, Bernd Sturmfels, and Nobuki Takayama. *Gröbner Deformations of Hypergeometric Differential Equations*, volume 6 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, 2000.

[12] Nobuki Takayama. An algorithm of constructing the integral of a module–an infinite dimensional analog of Gröbner basis. In *ISSAC '90: Proceedings of the international symposium on Symbolic and algebraic computation*, pages 206–211, New York, NY, USA, 1990. ACM.

[13] Nobuki Takayama. Gröbner basis, integration and transcendental functions. In *ISSAC '90: Proceedings of the international symposium on Symbolic and algebraic computation*, pages 152–156, New York, NY, USA, 1990. ACM.

[14] Doron Zeilberger. A holonomic systems approach to special function identities. *Journal of Computational and Applied Mathematics*, 32(3):321–368, 1990.

Christoph Koutschan, Viktor Levandovskyy Received March 17, 2008

Christoph Koutschan
RISC
Altenberger Str. 69
A-4040 Linz, AUSTRIA
E–mail: *Koutschan@risc.uni − linz.ac.at*

Viktor Levandovskyy
RWTH Aachen
Templergraben 64
D-52062 Aachen, GERMANY
E–mail: *Viktor.Levandovskyy@math.rwth − aachen.de*

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

An algebraic approach to a study of two-dimensional affine differential system

E.Naidenova*

Abstract

In a present paper a problem of classification of $Aff(2,\mathbb{R})$-orbits' dimensions is considered on example of an autonomous two-dimensional affine differential system of first order. Methods of Lie algebras are used in the work, as well as methods of group analysis. Computer algebra systems "Bergman" and "Mathematica 5.0" are widely used.

Mathematics Subject Classification 2000: 34C14, 34C05, 58F14.

Keywords and phrases: Differential equation, Lie algebra, affine invariant, $Aff(2,\mathbb{R})$-orbit.

1 Introduction

In a present work we consider autonomous polynomial differential system, written in general form as follows

$$\frac{dx^j}{dt} = \sum_{m_i \in \Gamma} P^j_{m_i}(x),\ (j = \overline{1,2}), \tag{1}$$

where $\Gamma = \{m_i\}_{i=1}^{l}$ is some finite set of different non-negative integers, and

$$P^j_{m_i}(x) = \sum_{k=0}^{m_i} \binom{m_i}{k} a^{ij}_k (x^1)^{m_i-k}(x^2)^k,\ (j = \overline{1,2}; i = \overline{1,l})$$

©2008 by E. Naidenova

*Supported by INTAS grant Ref. Nr. 05-104-7553, CSSDT grant Ref. Nr.07.411.05INDF

are homogeneous polynomials with order m_i in $(x^1, x^2, ..., x^n)$. Coefficients and variables of system (1) are defined over the field of real numbers $\mathbb{R}$. Further we will denote system (1) by $s^2(\Gamma)$ for special Γ. The variable t is independent one, and x^1, x^2 are dependent functions (variables) on t.

System (1) will be considered with group of affine transformations $Aff(2, \mathbb{R})$ given by equalities

$$\overline{x}^1 = \alpha x^1 + \beta x^2 + h_1,\ \overline{x}^2 = \gamma x^1 + \delta x^2 + h_2,\ \left(\Delta = \begin{vmatrix} \alpha & \beta \\ \gamma & \delta \end{vmatrix} \neq 0\right), \quad (2)$$

where $\alpha,\ \beta,\ \gamma,\ \delta,\ h_1,\ h_2$ are real parameters, ever varying in $\mathbb{R}$. Further we will consider transformations (2) given by matrix $q = \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$, and when we say "q belongs to group $Aff(2, \mathbb{R})$", we write this as $q \in Aff(2, \mathbb{R})$.

Note that the application of group $Aff(2, \mathbb{R})$ to qualitative investigation of systems (1) is remarkable as the system keeps its form after affine transformation. And coefficients of the system are varying in according to law of tensors, being basic geometrical objects of Invariant Theory. Thus, we can conclude that to perform complete qualitative investigation of system (1) it is necessary to apply the method of algebraic invariants. Remark, that this method was founded in works by K.Sibirsky [1].

Adaptation of Lie algebras of operators and techniques of group analysis in study of systems (1) has appeared as a certain step in development of this method. Results of such researches are quoted in works by M.Popa [2] and his disciples. These works are devoted to investigation of algebraic objects (finite-dimensional Lie algebras and corresponding algebras of invariants), obtained due to representation of linear groups of transformations in space of coefficients of systems (1). Besides, the classification's tasks are considered in these works, concerned with dimensions of orbits, as well as with problems of existence of invariant integrals.

As appeared, an answer to the question about existence of such integrals is thoroughly connected with classification of orbits' dimensions

and of invariant varieties of considering groups, particularly, group $Aff(2,\mathbb{R})$. Therefore it became necessary to construct such classifications for further investigation of systems (1).

Remark, that solution of classifications' questions for systems (1) with more than one homogeneity in right-hand sides requires implication of computer algebra systems and was impossible until nowadays due to intricate calculations.

2 Basic notions and definitions

Throughout the work we will need some notions.

Definition 2.1. *Call the linear space L_r over the field $\mathbb{R}$ a Lie algebra, if for any two of its elements X, Y the operation of commutation $[X,Y]$ is defined, which returns the element from L_r (commutator of elements X, Y) and satisfies the following axioms:*

1) bilinearity: for any $X,Y,Z \in L$ and $\alpha,\beta \in \mathbb{R}$

$$[\alpha X + \beta Y, Z] = \alpha[X,Z] + \beta[Y,Z],$$

$$[X, \alpha Y + \beta Z] = \alpha[X,Y] + \beta[X,Z];$$

2) anti-symmetry: for any $X,Y \in L$

$$[X,Y] = -[Y,X];$$

3) identity of Jacobi: for any $X,Y,Z \in L$

$$[[X,Y],Z] + [[Y,Z],X] + [[Z,X],Y] = 0.$$

It is shown in [2] that Lie algebra, corresponding to linear representation of group $Aff(2,\mathbb{R})$ in the space of coefficients and variables of system (1), is six-dimensional Lie algebra $L_6 = \{X_1, X_2, X_3, X_4, X_5, X_6\}$. This algebra can be given by Lie operators [2]:

$$X_1 = x^1\frac{\partial}{\partial x^1} - D_1,\ X_2 = x^2\frac{\partial}{\partial x^1} - D_2,\ X_3 = x^1\frac{\partial}{\partial x^2} - D_3,$$
$$X_4 = x^2\frac{\partial}{\partial x^2} - D_4,\ X_5 = \frac{\partial}{\partial x^1} - D_5,\ X_6 = \frac{\partial}{\partial x^2} - D_6, \qquad (3)$$

where

$$D_1 = \sum_{i=1}^{l}\sum_{k=0}^{m_i}\left[(m_i-k-1)\overset{i}{a}{}^1_k\frac{\partial}{\partial \overset{i}{a}{}^1_k} + (m_i-k)\overset{i}{a}{}^2_k\frac{\partial}{\partial \overset{i}{a}{}^2_k}\right],$$

$$D_2 = \sum_{i=1}^{l}\sum_{k=0}^{m_i}\left[k\left(\overset{i}{a}{}^1_{k-1}\frac{\partial}{\partial \overset{i}{a}{}^1_k} + \overset{i}{a}{}^2_{k-1}\frac{\partial}{\partial \overset{i}{a}{}^2_k}\right) - \overset{i}{a}{}^2_k\frac{\partial}{\partial \overset{i}{a}{}^1_k}\right],$$

$$D_3 = \sum_{i=1}^{l}\sum_{k=0}^{m_i}\left[(m_i-k)\left(\overset{i}{a}{}^1_{k+1}\frac{\partial}{\partial \overset{i}{a}{}^1_k} + \overset{i}{a}{}^2_{k+1}\frac{\partial}{\partial \overset{i}{a}{}^2_k}\right) - \overset{i}{a}{}^1_k\frac{\partial}{\partial \overset{i}{a}{}^2_k}\right],$$

$$D_4 = \sum_{i=1}^{l}\sum_{k=0}^{m_i}\left[k\overset{i}{a}{}^1_k\frac{\partial}{\partial \overset{i}{a}{}^1_k} + (k-1)\overset{i}{a}{}^2_k\frac{\partial}{\partial \overset{i}{a}{}^2_k}\right],$$

$$D_5 = \sum_{i=1}^{l}\sum_{k=0}^{i-1} i\left(\overset{i}{a}{}^1_k\frac{\partial}{\partial \overset{i-1}{a}{}^1_k} + \overset{i}{a}{}^2_k\frac{\partial}{\partial \overset{i-1}{a}{}^2_k}\right),$$

$$D_6 = \sum_{i=1}^{l}\sum_{k=0}^{i-1}\left(\overset{i}{a}{}^1_{k+1}\frac{\partial}{\partial \overset{i-1}{a}{}^1_k} + \overset{i}{a}{}^2_{k+1}\frac{\partial}{\partial \overset{i-1}{a}{}^2_k}\right). \qquad (4)$$

According to [2], in order to solve the problem of classification of orbits' dimensions, we will consider only operators D_1 - D_6, since they form six-dimensional Lie algebra L_6, corresponding to linear representation of group $Aff(2,\mathbb{R})$ in the space of coefficients of system (1).

Let $a = \left(\overset{1}{a}{}^1_0, \overset{1}{a}{}^1_1, \ldots, \overset{l}{a}{}^2_{m_l}\right) \in E(a)$, where $E(a)$ is Euclidean space of coefficients of right-hand sides of system (1).

Denote by $a(q)$ a point from $E(a)$ corresponding to a system, obtained from system (1) with coefficients a after transformation $q \in Aff(2,\mathbb{R})$.

Definition 2.2. *The set $O(a) = \{a(q); q \in Aff(2,\mathbb{R})\}$ is called an $Aff(2,\mathbb{R})$-orbit of a point* **a** *for system* (1).

Definition 2.3 *The set $M \subseteq E(a)$ is called an $Aff(2,\mathbb{R})$-invariant set if for any point $a \in M$ its orbits $O(a) \subseteq M$.*

It is known from [3] - [4] that space $\mathfrak{g}(a)$, constructed on coordinate vectors of operators (4), is the tangent space to $Aff(2,\mathbb{R})$ - orbit $O(a)$ in point $a \in E(a)$, such that

$$dim_{\mathbb{R}} O(a) = dim_{\mathbb{R}} \mathfrak{g}(a). \tag{5}$$

On the other hand,

$$dim_{\mathbb{R}} \mathfrak{g}(a) = rank M_1, \tag{6}$$

where M_1 is a matrix, constructed on coordinate vectors of operators (4).

From (5) - (6) it is evident

$$dim_{\mathbb{R}} O(a) = rank M_1. \tag{7}$$

Denote by

$$M = \begin{pmatrix} x^1 & 0 \\ x^2 & 0 \\ 0 & x^1 \\ 0 & x^2 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

We will denote the matrix (M, M_1) by $(\xi(x), \eta(a))$ when it represents a reflection in space of coefficients and variables $E(x,a)$ of system (1).

Further we will consider varieties Ψ given implicitly in finite-dimensional space $E(x,a)$ [4].

This means that an open set $U \subset E(x,a)$ is given together with reflection $\psi : U \to \mathbb{R}$ of class $C_\infty(U)$, and $\psi(x_0, a_0) = 0$ for some point $(x_0, a_0) \in U$ and the set $\psi(U_0)$ is open in $\mathbb{R}$ for any vicinity $U_0 \subset U$ of the point (x_0, a_0). Variety Ψ can be defined in these conditions as locus of $(x,a) \in U$, for which holds

$$\psi(x,a) = 0. \tag{8}$$

Equality (8) is called the equation of variety Ψ.

Definition 2.4. *Call the variety Ψ an invariant if for any point $a \in \Psi$ its orbit $O(a) \subseteq \Psi$.*

Definition 2.5. *Call the number*

$$r_* = r_*(\xi, \eta) = \max_{(x,a)\in U} rank(\xi(x), \eta(a))$$

a general rang of the reflection (ξ, η) onto open set $U \subset E(x, a)$.

Definition 2.6. *Call the point $(x, a) \in E(x, a)$ a singular point (of group $Aff(2, \mathbb{R})$ or its Lie algebra L_6), if*

$$rank(\xi(x), \eta(a)) < r_*,$$

and non-singular point (of group $Aff(2, \mathbb{R})$ or its Lie algebra L_6) if

$$rank(\xi(x), \eta(a)) = r_*.$$

Definition 2.7. *Call the variety $\Psi \subset U$ a singular variety of group $Aff(2, \mathbb{R})$ (or its Lie algebra $L_6(\xi, \eta)$) if all its points are singular and if the reflection (ξ, η) has the rang on Ψ, i.e. for any point $(x, a) \in \Psi$ we obtain*

$$rank(\xi(x), \eta(a)) = r_*(M|\Psi) < r_*.$$

Definition 2.8. *Call the variety $\Psi \subset U$ a non-singular variety of group $Aff(2, \mathbb{R})$ (or its Lie algebra $L_6(\xi, \eta)$) if all its points are non-singular, i.e. if the following equality holds*

$$r_*(M|\Psi) = r_*.$$

According to last definitions, all invariant varieties of group $Aff(2, \mathbb{R})$ can be divided into singular and non-singular $Aff(2, \mathbb{R})$-invariant varieties.

From this viewpoint, the classification of dimensions of $Aff(2, \mathbb{R})$-orbits of differential equations' system can be represented as a classification of invariant varieties of group $Aff(2, \mathbb{R})$. Remark, that $Aff(2, \mathbb{R})$-orbits of maximal dimension correspond to non-singular invariant varieties of group $Aff(2, \mathbb{R})$.

From Theorem of representation [4] follows

Theorem 2.1. *If non-singular variety of Lie algebra $L_6(\xi, \eta)$ is given regularly by equation* (8)*, then such invariant $F : E(x, a) \to \mathbb{R}$ of this algebra exists, that this variety can be given by equality $F(x, a) = 0$.*

Definition 2.9. *Call the integer rational function $K(x, a)$, in variables x and coefficients a of system* (1) *an affine comitant if it meets the condition*

$$K(\bar{x}, \bar{a}) = \Delta^{-g} K(x, a)$$

for any values of x and a and any transformations of group $Aff(2, \mathbb{R})$. Number g is called a weight of affine comitant.

Definition 2.10. *If an affine comitant $K(x, a)$ does not depend on variables x, it is called an affine invariant of system* (1).

From [2] and [4] it is known

Theorem 2.2. *The integer rational function $K(x, A)$ $(I(A))$ in variables x and coefficients a of system* (1) *is an affine comitant (invariant) of this system with weight g if and only if it meets conditions*

$$X_1(K) = X_4(K) = -gK,\ X_2(K) = X_3(K) = X_5(K) = X_6(K) = 0;$$
$$D_1(I) = D_4(I) = -gI,\ D_2(I) = D_3(I) = D_5(I) = D_6(I) = 0,$$

where X_1 - X_6 and D_1 - D_6 are defined in (3) *and* (4).

3 Classification of dimensions of $Aff(2, \mathbb{R})$ - orbits for system $s^2(0, 1)$.

Let us apply above stated theory to investigation of affine differential system $s^2(0, 1)$.

Consider system (1) for $\Gamma = \{0, 1\}$. According to [1] we will write it in tensor form as follows

$$\frac{dx^j}{dt} = a^j + a^j_\alpha x^\alpha, \quad (j, \alpha = 1, 2). \tag{9}$$

System (9) will be considered with group $Aff(2, \mathbb{R})$, defined in (2).

Further we will use affine comitants and invariants known from works [1], [5], [6]:

$$K_2 = a^p_\alpha x^\alpha x^q \varepsilon_{pq},\ K_{21} = a^p x^q \varepsilon_{pq},\ K_{22} = a^\alpha a^p_\alpha x^q \varepsilon_{pq},$$
$$I_1 = a^\alpha_\alpha,\ I_2 = a^\alpha_\beta a^\beta_\alpha,\ I_{21} = a^\alpha a^q a^p_\alpha \varepsilon_{pq},$$
$$Q = I_{21} + I_1 K_{22} - I_2 K_{21} + \frac{1}{2}(I_1^2 - I_2)K_2, \qquad (10)$$

where ε^{pq} and ε_{pq} are unit bi-vectors with coordinates $\varepsilon^{11} = \varepsilon^{22} = 0$, $\varepsilon^{12} = -\varepsilon^{21} = 1$ and $\varepsilon_{11} = \varepsilon_{22} = 0$, $\varepsilon_{12} = -\varepsilon_{21} = 1$.

Remark [6], that invariants I_1, I_2 and comitant Q form minimal polynomial basis of affine comitants for system (9).

In order to simplify further expressions we will use the following notations

$$x^1 = x,\ x^2 = y,\ a^1 = a,\ a^2 = b,\ a^1_1 = c,\ a^1_2 = d,\ a^2_1 = e,\ a^2_2 = f. \qquad (11)$$

According to (3) - (4) and (11), we will write Lie operators for system (9):

$$X_1 = x\frac{\partial}{\partial x} - D_1,\ X_2 = y\frac{\partial}{\partial x} - D_2,\ X_3 = x\frac{\partial}{\partial y} - D_3,$$
$$X_4 = y\frac{\partial}{\partial y} - D_4,\ X_5 = \frac{\partial}{\partial x} - D_5,\ X_6 = \frac{\partial}{\partial y} - D_6,$$

where

$$D_1 = -a\frac{\partial}{\partial a} - d\frac{\partial}{\partial d} + e\frac{\partial}{\partial e},\ D_2 = -b\frac{\partial}{\partial a} - e\frac{\partial}{\partial c} + (c-f)\frac{\partial}{\partial d} + e\frac{\partial}{\partial f},$$
$$D_3 = -a\frac{\partial}{\partial b} + d\frac{\partial}{\partial c} - (c-f)\frac{\partial}{\partial e} - d\frac{\partial}{\partial f},\ D_4 = -b\frac{\partial}{\partial b} + d\frac{\partial}{\partial d} - e\frac{\partial}{\partial e},$$
$$D_5 = c\frac{\partial}{\partial a} + e\frac{\partial}{\partial b},\ D_6 = d\frac{\partial}{\partial a} + f\frac{\partial}{\partial b}. \qquad (12)$$

Matrix M_1, constructed on coordinate vectors of operators (12), takes the form

$$M_1(0,1) = \begin{pmatrix} -a & 0 & 0 & -d & e & 0 \\ -b & 0 & -e & c-f & 0 & e \\ 0 & -a & d & 0 & f-c & -d \\ 0 & -b & 0 & d & -e & 0 \\ c & e & 0 & 0 & 0 & 0 \\ d & f & 0 & 0 & 0 & 0 \end{pmatrix} \tag{13}$$

Remark 3.1. *One can verify that rank of matrix* (13) *is less than 5. Therefore, according to* (7)*, the dimension of* $Aff(2,\mathbb{R})$*-orbit for system* (9) *is less than 5.*

Remark 3.2. *Using* (10)*, one can verify that* $K_2 \equiv 0$ *yields* $Q \equiv 0$*.*

To define a rank of matrix $M_1(0,1)$ it is necessary to construct all its minors of all possible orders. It is done using computer algebra system "Mathematica 5.0". In order to find affine-invariant conditions for rank of matrix $M_1(0,1)$ its minors of each order are considered separately along with invariants and semi-invariants (corresponding coefficients of affine comitants with each degree of variable x) of system (9). As these objects are polynomials depending on coefficients of system (9) and forming an ideal, the corresponding Gröbner bases [7] can be used to obtain linear dependency among them. Namely, the set of minors of each order is divided in subsets with respect to their types. All possible combinations of invariants, semi-invariants and their products of each type are composed. The corresponding Gröbner bases then has been constructed for them with the help of computer algebra system "Bergman" [8]. Analyzing such a bases one can figure out its element representing linear dependency between minors of matrix (13) and affine invariants and semi-invariants, as this element should contain only names of minors, invariants and semi-invariants, not the coefficients of system (9). According to this algorithm all types of minors of matrix (13) have been treated and corresponding Gröbner bases are constructed, therefore, desired dependencies are obtained. This technique is used throughout the proofs of Lemmas 3.1 - 3.4.

Lemma 3.1. *Rank of matrix* $M_1(0,1)$ *is equal to 4 if and only if holds*

$$K_2 Q \not\equiv 0, \tag{14}$$

where K_2 and Q are defined in (10).

Proof. Let us prove the necessity. Assume the contradiction. Namely, assume that for

$$K_2 Q \equiv 0 \tag{15}$$

even one non-zero minor of 4th order of matrix (13) exists. Equality (15) holds at least for $K_2 \equiv 0$ or $Q \equiv 0$.

Examine $K_2 \equiv 0$. Than, taking into consideration (10) and (11), we obtain the following values for coefficients of system (9)

$$e = d = 0,\ c = f. \tag{16}$$

After substitution of values (16) to matrix (13) one can verify that all 4th order's minors of this matrix are equal to zero. Thus, the assumption is not true in this case.

Examine $Q \equiv 0$. Than, taking into consideration (10) and (11), we obtain the following series of values for coefficients of system (9):

$$e = d = 0,\ c = f, \tag{17}$$

$$a = c = d = 0, \tag{18}$$

$$b = e = f = 0, \tag{19}$$

$$d = f = 0,\ e = \frac{bc}{a},\ a \neq 0, \tag{20}$$

$$a = b = 0,\ d = \frac{fc}{e},\ e \neq 0, \tag{21}$$

$$c = -f,\ d = \frac{af}{b},\ e = -\frac{bf}{a},\ ab \neq 0, \tag{22}$$

$$c = e = 0,\ d = \frac{af}{b},\ b \neq 0, \tag{23}$$

$$c = f,\ d = \frac{af}{b},\ e = \frac{bf}{a},\ ab \neq 0. \tag{24}$$

Case (17) coincides with case (16), obtained for $K_2 \equiv 0$, and will not be considered.

After substitution of each of series (18) - (24) to matrix (13) we obtain that all its 4th order minors are equal to zero. So, the above

stated assumption is not true in this case too. Therefore we conclude the necessity of conditions (14).

Sufficiency of conditions (14) is ensured by equality

$$K_2Q = \Delta^{1256}_{1235}x^4 + 2\Delta^{1256}_{1236}x^3y + (2\Delta^{1256}_{1234} - \Delta^{2356}_{1236})x^2y^2 + 2\Delta^{1356}_{1236}xy^3 + \\ +\Delta^{1356}_{1234}y^4 + (\Delta^{1245}_{1236} + 2\Delta^{2345}_{1235})x^3 + (\Delta^{2345}_{1236} - 2\Delta^{1236}_{1235})x^2y + (2\Delta^{2345}_{1234} - \\ -\Delta^{1236}_{1236})xy^2 + (\Delta^{1346}_{1236} - 2\Delta^{1345}_{1234})y^3 + \Delta^{1234}_{1235}x^2 - \Delta^{1234}_{1236}xy - \Delta^{1234}_{1234}y^2,$$

where Δ^{ijhk}_{lmnp} is 4th order minor of matrix (13), constructed on lines i, j, h, k $(1 \le i, j, h, k \le 6)$ and columns l, m, n, p $(1 \le l, m, n, p \le 6)$. Lemma 3.1 is proved.

Lemma 3.2. *Rank of matrix $M_1(0,1)$ is equal to 3 if and only if hold*

$$Q \equiv 0,\ K_2 \not\equiv 0, \tag{25}$$

where K_2 and Q are defined in (10).

Proof. Necessity of conditions (25) follows from Lemma 3.1. Let us prove sufficiency. We will consider each of cases (18) - (24) separately. Note, that case (17) contradicts to conditions of Lemma 3.2.

Denote by Δ^{ijk}_{lmn} a 3rd order minor of matrix (13) constructed on lines i, j, h, $(1 \le i, j, k \le 6)$ and columns l, m, n $(1 \le l, m, n \le 6)$.

As conditions (18) hold, comitant K_2 takes the form $K_2 = -ex^2 - fxy$. For $K_2 \not\equiv 0$ non-zero 3rd order's minors of matrix (13) will be at least $\Delta^{125}_{145} = -e^3$ or $\Delta^{236}_{245} = f^3$.

As conditions (19) hold, comitant K_2 takes the form $K_2 = cxy + dy^2$. For $K_2 \not\equiv 0$ non-zero 3rd order's minors of matrix (13) will be at least $\Delta^{136}_{134} = -d^3$ or $\Delta^{235}_{145} = c^3$.

As conditions (20) hold, comitant K_2 takes the form $K_2 = c(-\frac{b}{a}x^2 + xy)$. For $K_2 \not\equiv 0$ non-zero 3rd order's minor of matrix (13) will be at least $\Delta^{235}_{145} = c^3$.

As conditions (21) hold, comitant K_2 takes the form $K_2 = -ex^2 + (c - f)xy + \frac{cf}{e}y^2$. Remark, that $e \neq 0$. So, $K_2 \not\equiv 0$ and non-zero 3rd order's minor of matrix (13) will be at least $\Delta^{125}_{145} = -e^3$.

As conditions (22) or (24) hold, comitant K_2 takes the form $K_2 = f(-\frac{b}{a}x^2 - 2xy + \frac{a}{b}y^2)$ or $K_2 = f(-\frac{b}{a}x^2 + \frac{a}{b}y^2)$, correspondingly. In both

cases for $K_2 \not\equiv 0$ non-zero 3rd order's minor of matrix (13) will be at least $\Delta^{125}_{134} = f^3$.

As conditions (23) hold, comitant K_2 takes the form $K_2 = f(-xy + \frac{a}{b}y^2)$. For $K_2 \not\equiv 0$ non-zero 3rd order's minor of matrix (13) will be at least $\Delta^{236}_{245} = f^3$.

Sufficiency of conditions (25) is proved completely. Lemma 3.2 is proved.

Lemma 3.3. *Rank of matrix $M_1(0,1)$ is equal to 2 if and only if hold*

$$K_2 \equiv 0, \quad K_{21}^2 + I_1^2 \not\equiv 0, \tag{26}$$

where K_2, K_{21}, I_1 are defined in (10).

Proof. Denote by Δ^{ij}_{hk} a 2nd order minor of matrix (13) constructed on lines i, j $(1 \le i, j \le 6)$ and columns h, k $(1 \le h, k \le 6)$.

Necessity of equality from (26) follows from Lemmas 3.1 - 3.2 and Remark 3.2. Let us prove necessity of inequality from (26). Assume the contradiction. Namely, assume that for

$$K_{21}^2 + I_1^2 \equiv 0 \tag{27}$$

at least one non-zero 2nd order's minor of matrix (13) exists. For $K_2 \equiv 0$, taking into consideration (10) and (16), invariant I_1 takes the form

$$I_1 = 2f. \tag{28}$$

According to (10) and (11), comitant K_{21} can be written as follows

$$K_{21} = -bx + ay. \tag{29}$$

As $K_2 \equiv 0$ holds, all non-zero 2nd order's minors of matrix (13) will coincide to sign with one of the following

$$\Delta^{13}_{12} = a^2,\ \Delta^{14}_{12} = ab,\ \Delta^{24}_{12} = b^2,\ \Delta^{16}_{12} = af,\ \Delta^{26}_{12} = bf,\ \Delta^{56}_{12} = f^2. \tag{30}$$

As (27) holds, from (28) and (29) follows that $a = b = f = 0$ and all minors (30) are equal to zero. This contradiction confute our assumption and confirms the necessity of inequality from (26).

Sufficiency of conditions (26) is ensured by equality

$$K_{21}^2 + I_1^2 = \Delta_{12}^{24} x^2 - 2\Delta_{12}^{14} xy + \Delta_{12}^{13} y^2 + 4\Delta_{12}^{56}.$$

Lemma 3.3 is proved.

From Lemmas 3.1 - 3.3 evidently follows

Lemma 3.4. *Rank of matrix $M_1(0,1)$ is equal to 0 if and only if hold*

$$K_2 \equiv 0, \quad K_{21}^2 + I_1^2 \equiv 0, \tag{31}$$

where K_2, K_{21}, I_1 are defined in (10).

From Lemmas 3.1 - 3.4, Remark 3.1 and equality (7) follows

Theorem 3.1. *$Aff(2,\mathbb{R})$ - orbit of system* (9) *has the dimension*

$$4 \quad for \quad QK_2 \not\equiv 0; \tag{32}$$

$$3 \quad for \quad Q \equiv 0,\ K_2 \not\equiv 0; \tag{33}$$

$$2 \quad for \quad K_2 \equiv 0,\ K_{21}^2 + I_1^2 \not\equiv 0; \tag{34}$$

$$0 \quad for \quad K_2 \equiv 0,\ K_{21}^2 + I_1^2 \equiv 0, \tag{35}$$

where K_2, K_{21}, Q, I_1 are defined in (10).

According to Definition 2.3 from Theorem 3.1 follows

Theorem 3.2. *Sets M_1, M_2, M_3, M_4, defined by expressions* (32), (33), (34) *and* (35) *correspondingly, form $Aff(2,\mathbb{R})$-invariant partition of space $E(a)$ of coefficients of system* (9), *i.e.*

$$\bigcup_{i=1}^{4} M_i = E(a), \quad M_i \bigcap M_j = \emptyset$$

and each set M_1 ($i = \overline{1,4}$) is $Aff(2,\mathbb{R})$-invariant.

Remark 3.3. *Set M_1 with conditions* (32) *represents non-singular invariant variety of group $Aff(2,\mathbb{R})$.*

Remark 3.4. *Sets M_2-M_4 with conditions* (33) - (35) *correspondingly represent singular invariant varieties of group $Aff(2,\mathbb{R})$.*

Some results of this paper were announced in a common report with V.Orlov at the Conference "Algebraic systems and their applications in differential equations and other domains of mathematics", see [9].

References

[1] Sibirsky K., Introduction to Algebraic Theory of Invariants of Differential Equations, Chishinau, Shtiintsa, (1982) (in Russian, published in English in 1988)

[2] Popa M., Application of algebras to differential systems, Academy of Sciences of Moldova, Chishinau, (2001) (in Russian)

[3] Vinberg E. and Popov V., Theory of invariants, Itogi Nauki i Tehniki, series "Sovr. Probl. Matematiki", v 55, Moscow, (1989), pp.137–313 (in Russian)

[4] Ovsyannikov L., Group analysis of differential equations, Nauka, Moscow, (1978) (in Russian)

[5] Boularas D., Calin Iu., Tomichouk L., Vulpe N., T-comitants of quadratic systems: a study via the translation invariants, Report 96–90, Delft, University of Technology, Netherlands, (1996)

[6] Boularas D., Classification affine des systemes differentiels, These de magister, Institute de Mathematiques, Republique Algerienne Democratique et Populaire, (1992) (in French)

[7] Adams W.W., Loustaunau P., An introduction to Gröbner bases, Vol.3, Graduate studies in Mathematics, Providence, RI: AMS, 1966

[8] Backelin J., Cojocaru S., Ufnarovski V., Mathematical Computations using Bergman, Centre for Mathematical Sciences, Lund University, Sweden, (2005)

[9] Naidenova E., Orlov V., The classification of Aff(2,R)-orbits's dimensions for system s(0,1), Book of abstracts of International Conference "Algebraic systems and their applications in differential equations and other domains of mathematics" Aug., 21–23, 2007, Chisinau, Moldova

E.Naidenova,

Institute of Mathematics and Computer Science,
5 Academiei str.
Chişinău, MD−2028, Moldova.
E–mail: *hstarus@gmail.com*

Received December 17, 2007

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

A zero-dimensional approach to compute real radicals

Silke J. Spang

Abstract

The notion of real radicals is a fundamental tool in Real Algebraic Geometry. It takes the role of the radical ideal in Complex Algebraic Geometry. In this article I shall describe the zero-dimensional approach and efficiency improvement I have found during the work on my diploma thesis at the University of Kaiserslautern (cf. [6]). The main focus of this article is on maximal ideals and the properties they have to fulfil to be real. New theorems and properties about maximal ideals are introduced which yield an heuristic `prepare_max` which splits the maximal ideals into three classes, namely real, not real and the class where we can't be sure whether they are real or not. For the latter we have to apply a coordinate change into general position until we are sure about realness. Finally this constructs a randomized algorithm for real radicals. The underlying theorems and algorithms are described in detail.

1 Introduction

The original task arose from an article by Becker and Neuhaus written in 1998 (see [1]), where they present an idea to compute the real radical of a polynomial ideal. The following article speeds up the computation time of the algorithm which they described there:

Becker and Neuhaus idea was a coordinate change to reduce to the univariate case. Such coordinate changes cause a coefficient growth which slows down the computation.

©2008 by Silke J. Spang

Our idea is to study the properties of maximal ideals M and find a heuristic to decide whether they are real, i. e. if $\sqrt[re]{M} = M$ or not. This arose from the fact that the primary decomposition in SINGULAR is well implemented and very efficient in the average case.

The article is structured in three parts:

Section 1 gives a short overview of and motivation for the notion of τ-radicals. In particular the real radical is recalled. Some theory on how the $\sqrt[re]{}$-functor behaves and first properties of K-algebras A are stated. The real radical commutes with intersection and localisation. For an arbitrary ideal $I \trianglelefteq A$, we know $\sqrt[re]{I} = \sqrt[re]{\sqrt[re]{I}}$, and $\sqrt[re]{I}$ is a radical ideal by definition. A special form of the Real Nullstellensatz over $\mathbb{Q}$ is stated. One of the fundamental statements is Theorem 1 which tells us that the real radical of I is the intersection of all real prime ideals P containing I. In fact, giving rise to all real points, the real radical of I is the intersection of all real maximal ideals M containing I. The section finishes by sketching how the one-to-one correspondences from algebraic geometry over algebraically closed fields are translated to real algebraic geometry by means of the real radical. Thus a real maximal ideal corresponds to a zero-dimensional real zero-set which can be seen as finitely many conjugate points in the field extension of $\mathbb{Q}$ to $\mathbb{R}_{alg}$ (or $\mathbb{R}$ by the Tarski Seidenberg principle).

Prime ideals correspond to irreducible $\mathbb{Q}$-varieties in $\mathbb{R}^n$ and the primary decomposition is just the decomposition of a $\mathbb{Q}$-variety $V_{re}(I) \subset \mathbb{R}^n$ into its irreducible components.

The univariate case of polynomials $f \in \mathbb{Q}(y_1, \ldots, y_m)[x]$ which is a special case of zero-dimensional ideals is explained in Section 2. The main idea is the following: Let

$$f = \varepsilon \cdot p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdots p_r^{\alpha_r}.$$

If we could decide whether a prime polynomial p_i is real or not, then the real radical of the principal ideal $\langle f \rangle \trianglelefteq \mathbb{Q}(y_1, \ldots, y_m)[x]$ is

$$\sqrt[re]{\langle f \rangle} = \langle \prod_{p_i \text{ is real}} p_i \rangle.$$

This provides an idea how to compute the real radical of a univariate polynomial.

After describing the machinery for the univariate case, an algorithm for computing the zero-dimensional radical is explained in section 3. In contrast to the article of Becker and Neuhaus, the decision was to compute the primary decomposition of the zero-dimensional input and to give a heuristic for deciding whether a maximal ideal is real or not. This heuristic yields a procedure `prepare_max` which prepares a maximal ideal in such a way that we can avoid a coordinate change into general position as often as possible. If a coordinate change can't be avoided we use the procedure `GeneralPos`. Its input is a list of maximal ideals where a change can't be avoided. Here a suitably randomised coordinate change is computed such that we can check the properties of `prepare_max` for the transformed maximal ideals and afterwards we intersect all real maximal ideals of this list. The procedure `RealZero` gets a zero-dimensional input I and computes its primary decomposition. Then it considers separately every maximal ideal and tests if a change is needed to compute the real part. Afterwards it intersects the real radicals of all these 'nice' maximal ideals and restarts the procedure `GeneralPos` for the list of 'bad' ideals. To conclude the article section 3 is finished with one important Theorem of Becker and Neuhaus ([1] Theorem 4.5.) which explains the computation real radicals of general polynomial ideals via a reduction to the zero-dimensional case.

I would like to thank Dr. Anne Frühbis-Krüger and Prof. Dr. Gerhard Pfister for many fruitful discussions. I want to thank the SINGULAR team of the University in Kaiserslautern, especially Dr. Hans Schönemann, for supporting me with my SINGULAR problems while implementing the algorithms for my diploma thesis and giving good advise on the computation.

2 τ-real ideals and the real radical

This section uses some basics in real algebra which can be found in [5]. We define τ-radicals for pre-orderings σ of real fields K.

Definition 1 (τ-radicals and the real radical) *Let K be a formally real field and τ a pre-ordering of K. For any K-algebra A, we define the τ-radical of an ideal $I \trianglelefteq A$ by*

$$\sqrt[\tau]{I} = \{f \in A \ : \ f^{2r} + \sum_{i=1}^{m} a_i g_i^2 \in I \text{ with } r, m \in \mathbb{N}, g_i \in A \text{ and } a_i \in \tau \ \forall i\}.$$

An ideal I with the property $I = \sqrt[\tau]{I}$ is called τ-real.
If $\tau = \sum K^2 =: re$, then $\sqrt[re]{I}$ is called the real radical of I.

We can easily verify that $\sqrt[\tau]{I}$ is an ideal. For the special case of subfields K of $\mathbb{R}$ we get the following definition.

Definition 2 (Real radical) *Let A be an affine K-algebra, $I \trianglelefteq A$ any ideal. We define the real radical of I to be*

$$\begin{aligned}\sqrt[re]{I} := \langle f \in A : \exists r, m \in \mathbb{N} : \\ f^{2r} + \sum_{i=1}^{m} k_i g_i^2 \in I, \ k_i \in K_{\geq 0}, g_i \in A \rangle\end{aligned}$$

*I is called **real** if and only if $\sqrt[re]{I} = I$.*

To see that both definitions do not differ for $\mathbb{Q} \subseteq K \subseteq \mathbb{R}$ and the special case $\tau = re = \sum \mathbb{Q}^2$ we prove the following lemma:

Lemma 1 *Let $K = \mathbb{Q}$, then $re = \sum K^2 = K_{\geq 0}$ is an ordering of K.*

Proof 1 *$\sum \mathbb{Q}^2 \subseteq \mathbb{Q}_{\geq 0}$ is clear.*
Let $\frac{p}{q} \in \mathbb{Q}_{>0}$. Then

$$\frac{p}{q} = \frac{pq}{q^2} = \sum_{i=1}^{pq} \left(\frac{1}{q}\right)^2 \in \sum \mathbb{Q}^2.$$

Hence $\mathbb{Q}$ has a unique real closure and this closure is $\mathbb{R}_{alg} := \overline{\mathbb{Q}} \cap \mathbb{R}$, so we get the following corollary.

Corollary 1 *For every algebraic extension K of $\mathbb{Q}$ which is in $\mathbb{R}$ there exists only one possible ordering, i. e. $\sum K^2 = K_{\geq 0}$.*

2.1 Some properties of the $\sqrt[\tau]{}$-functor

For this subsection see Chapter 2 of [1].

Theorem 1 *Let (K,τ) be a pre-ordered field, I, J ideals in some K-algebra A and S a multiplicative closed subset of A satisfying $1 \in S$ and $0 \notin S$. Then we have:*

(a) $\sqrt[\tau]{I \cap J} = \sqrt[\tau]{I} \cap \sqrt[\tau]{J}$

(b) $\sqrt[\tau]{I_S} = (\sqrt[\tau]{I})_S$

Here $\sqrt[\tau]{I_S}$ denotes the τ-radical of the extension ideal I_S of I in the quotient ring A_S which naturally is a K-algebra.

For prime ideals and prime polynomials we get the following properties:

Lemma 2 *Let (K,τ) be a pre-ordered field and I a τ-real ideal of some K-algebra A. Then all minimal primes of I are τ-real as well.*

Corollary 2 *Let (K,τ) be a pre-ordered field and I an ideal of some K-algebra A. Then $\sqrt[\tau]{I} = \bigcap P$, where P ranges over all τ-real primes containing I.*

Proof 2 *The τ-real ideal $\sqrt[\tau]{I}$ is radical and thus the intersection of its minimal primes. These are τ-real by Lemma 2.*

The most important proposition which describes the relation between τ-realness and the possibility to extend pre-orderings is stated below.

Proposition 1 *Let (K,τ) be a pre-ordered fields and P a prime ideal of some K-algebra A. Then the following statements are equivalent:*

(a) *P is τ-real*

(b) *There is some $\alpha \in X(K)$ (which is the set of all orderings for any formally real field K.) satisfying $\alpha \supseteq \tau$ which can be extended to an ordering $\overline{\alpha}$ of the function field $k(P) := Q(A/P)$.*

(c) There is some $\alpha \in X(K)$ satisfying $\alpha \supseteq \tau$ such that P is α-real.

Moreover if A is an affine K-algebra and P a maximal ideal of A then the statements $(a) - (c)$ are equivalent to:

(d) There is some $\alpha \in X(K)$ satisfying $\alpha \supseteq \tau$ such that $k(P)$ can be embedded into some real closed field containing the real closure of (K, τ).

Finally the real radical describes a real variety as a collection of all real points respectively. conjugated points.

Proposition 2 *Let (K, τ) be a pre-ordered field and I an ideal of some affine K-algebra A. Then $\sqrt[r]{I} = \bigcap M$, where M ranges over all τ-real maximal ideals of A containing I.*

2.1.1 The behaviour of prime polynomials

The well-known **sign change criterion** of D. Dubois and G. Elfroymson (see [5] Chapter 2 12 Theorem 4) is:

Theorem 2 *Let (K, τ) be an ordered field with its unique real closure R and $f \in K[x_1, \ldots, x_n]$ be an irreducible polynomial. Then the following are equivalent:*

(a) The ordering τ can be extended to an ordering $\overline{\alpha}$ over the function field $k(f) = Q(K[x_1, \ldots, x_n]/\langle f \rangle)$.

(b) f is indefinite over R, i. e. there exists $a, b \in R^n$ such that $f(a) \cdot f(b) < 0$.

This leads us directly to the following remark about the situation over the special case that $K = \mathbb{Q}$.

Remark 1 *Let $f \in \mathbb{Q}[x_1, \ldots, x_n]$ be an irreducible polynomial. Then f is real (i. e. $\langle f \rangle$ is real) if and only if f is indefinite over $\mathbb{R}_{alg}$ and thus by the Tarski-Seidenberg principle indefinite over $\mathbb{R}$.*

Proof 3 *f is real if and only if the ordering $re = \mathbb{Q}_{\geq}$ can be extended in $Q(\mathbb{Q}[x_1, \ldots, x_n]/\langle f \rangle)$ by Proposition 1. By the sign change criterion this can be extended if and only if f is indefinite over $\mathbb{R}_{alg}$.*

As another remark for polynomials over $\mathbb{Q}(y_1, \ldots, y_m)$ we get:

Remark 2 *Let $f \in \mathbb{Q}(y_1, \ldots, y_m)[x_1, \ldots, x_n]$ be an irreducible polynomial. Then f is real if and only if there exists an ordering α of $\mathbb{Q}(y_1, \ldots, y_m)$ such that f is indefinite over the corresponding real closure R_α.*

Proof 4 *Let $F := \mathbb{Q}(y_1, \ldots, y_m)$.*
Let us first observe that since f is irreducible the ideal $\langle f \rangle$ is a prime ideal. Let now $\alpha \in X(F)$ be an ordering such that f is indefinite over R_α. This ordering α of F can be extended to an ordering $\overline{\alpha}$ in $k(f) = F[x_1, \ldots, x_n]/\langle f \rangle$. By Proposition 1 (b) this is equivalent to the statements that $\langle f \rangle$ is real. Thus f is real.

2.2 The Real Nullstellensatz

We now state the Real Nullstellensatz which was proved by Krivine in the 60s. We first recall the set of real points. For more detailed information see [5] or ([1] Definition 2.7 and Theorem 2.8)

Definition 3 *Let (K, τ) be a pre-ordered field and $I \trianglelefteq K[x_1, \ldots, x_n]$. For a ordering $\alpha \supseteq \tau$ let R_α denote the unique real closure of (K, α). Then we define the set of all τ-real points V_τ as follows:*

$$V_\tau(I) = \cup_{\alpha \supseteq \tau} V_{R_\alpha}(I).$$

Especially the set of all real points is denoted by $V_{re}(I)$.

We get the general Real Nullstellensatz:

Theorem 3 (The general Real Nullstellensatz) *Let (K, τ) be a pre-ordered field and $I \trianglelefteq K[x_1, \ldots, x_n]$ be an ideal. Then we have*

$$I_K(V_\tau(I)) = \sqrt[\tau]{I}.$$

The following lemma is useful for the computation in real closed fields. Note that it is a kind of specialisation of the Weak Nullstellensatz over algebraically closed fields.

Lemma 3 *Let R be any real closed field and $M \lhd \cdot R[x_1, \ldots, x_n]$ be a maximal ideal. Then we have the following 2 cases.*

i. M is not real, so $V_R(M) = \varnothing$.

ii. M is real and $V_R(M)$ consists of only one point.

Proof 5 *As M is a maximal ideal $R' := R[x_1, \ldots, x_n]/M$ is a field extension of R. As R is real closed, we know that $\overline{R} = R(i)$ and $[\overline{R} : R] = 2$. So we have the following 2 cases.*

$[R' : R] = 1$ *Then $R' = R$ and every zero of M is real thus M is real. Let $a = (a_1, a_2, \ldots, a_n) \in R^n$ so $a \in V_R(M)$. Now $I_R(a) = \langle x_1 - a_1, x_2 - a_2, \ldots, x_n - a_n \rangle$ is a maximal ideal which contains M as $\langle x_1 - a_1, x_2 - a_2, \ldots, x_n - a_n \rangle = I_R(a) \subset I_R(V_R(M)) = M$. Thus $M = \langle x_1 - a_1, x_2 - a_2, \ldots, x_n - a_n \rangle$. And hence $V_R(M) = \{a\}$ is exactly one point.*

$[R' : R] = 2$ *Then $R' = \overline{R}$ and $\overline{R}$ is not real, thus M is not real by Proposition 1. Hence by the Real Nullstellensatz (Theorem 3) $V_R(M) = \varnothing$.*

2.3 One-to-one correspondences in real algebraic geometry

Let K be any subfield of $\mathbb{R}$ and $A = K[x_1, \ldots, x_n]$. Here the following special form of Theorem 3 holds:

Theorem 4 (Special Real Nullstellensatz) *Let $J \trianglelefteq K[x_1, \ldots, x_n]$, then:*

$$I_K(V_{\mathbb{R}}(J)) = \sqrt[re]{J}$$

This yields the well-known one-to-one correspondences.

$$\begin{aligned}
\text{real ideals} &\overset{1:1}{\longleftrightarrow} K\text{-varieties in } \mathbb{R}^n \\
\text{real prime ideals} &\overset{1:1}{\longleftrightarrow} \text{irreducible } K\text{-varieties in } \mathbb{R}^n \\
\text{real maximal ideals} &\overset{1:1}{\longleftrightarrow} \text{irreducible 0-dim. } K\text{-varieties in } \mathbb{R}^n
\end{aligned}$$

So every correspondence over $\mathbb{C}$ occurs in a natural way by means of real radicals in real algebraic geometry.

3 The univariate case

To obtain an algorithm for the zero-dimensional case, we first consider the univariate case, i. e. ideals in the principal ideal domain $F[x]$ where $F = \mathbb{Q}(y_1, \ldots, y_m)$. The main idea for the univariate case is the following: If we compute the real radical of $\langle f \rangle \trianglelefteq K[x]$, we know that factorising f corresponds to a primary decomposition. So if

$$f = \varepsilon p_1^{m_1} \cdot p_2^{m_2} \cdots p_r^{m_r}$$

then the $\langle p_i \rangle$, for all $i = 1, \ldots, r$ are precisely the minimal primes of $\langle f \rangle$. Such a minimal prime is real if and only if $V_{\mathbb{R}}(p_i) \neq \varnothing$, i. e. if p has a real root. So $\langle p_i \rangle$ is real if and only if p_i is real.

Hence the real radical of $\langle f \rangle$ is:

$$\sqrt[re]{\langle f \rangle} = \langle \prod_{p_i \text{ real}} p_i \rangle.$$

This leads us directly to the demand of a criterion to know whether an irreducible polynomial p is real or not.

Here we have two cases:

In the easier first case $F = \mathbb{Q}$ i.e. $m = 0$; the general case $m > 0$ requires more knowledge of real algebra.

3.1 The special univariate case

Definition 4 *Let $p \in \mathbb{Q}[x]$ be an irreducible polynomial. We call p* ***real*** *if p has a real root $\alpha \in \mathbb{R}$. Then p is the minimal polynomial of this root α.*

Note that p is real if and only if $V_{\mathbb{R}}(p) \neq \varnothing$, that is p is real if and only if $\langle p \rangle$ is real, since $\langle p \rangle$ is a maximal ideal and $\sqrt[re]{\langle p \rangle} \supseteq \langle p \rangle$. Hence the decision of being real for prime polynomials reduces to a root counting problem.

The solution to this problem is the following:
If the degree of p is odd the fundamental theorem of algebra over $\mathbb{R}$ states that p has a real root. But if the degree of p is even, we can't be sure if p has a real root. In this case we use the theorem of Sturm, which counts the number of all distinct real roots of a non–constant polynomial $f \in K[x]$ in an interval $[a, b]$, where $a < b$. The best a and b can be found by computing the Cauchy bound for polynomials. For detailed description of Sturm's theorem and its applications see [2].

3.2 The general univariate case

Contrary to the special case $F = \mathbb{Q}$ the general case of polynomials in $\mathbb{Q}(y_1, \ldots, y_m)[x]$ is not a real root counting problem as we do not know about sign or when a root is real. Thus we need some tools of real algebra.

The following special form of Lemma 4.1 in [1] gives a solution to the decision problem of realness for prime polynomials:

Lemma 4 *Let $p \in \mathbb{Q}[y_1, \ldots, y_m, x]$, where $m \in \mathbb{N}_0$ and $\deg_x p > 0$ be an irreducible polynomial. Then the following conditions are equivalent:*

(a) $\langle p \rangle \cdot \mathbb{Q}(y_1, \ldots, y_m)[x]$ *is real.*

(b) $\langle p \rangle \cdot \mathbb{Q}[y_1, \ldots, y_m, x]$ *is real.*

(c) p is indefinite over $\mathbb{R}$, i. e. there are points $\underline{a}, \underline{b} \in \mathbb{R}^{m+1}$ satisfying $p(\underline{a}) \cdot p(\underline{b}) < 0$.

This reduces our problem to decision whether a polynomial has a sign change i. e. whether it is indefinite or not. For a detailed solution of this problem see the article of G. Zeng and X. Zeng [4].

3.3 Example for the procedure `RealPoly`

The algorithm `RealPoly` (cf. SINGULAR Release 3-0-3) computes the real part of a polynomial in the univariate case. We conclude this section with some examples.

Example 1 *1. Let* $f = x^9+x^7+2x^6+x^5+2x^4-7x^3+4x^2-8x+4 \in \mathbb{Q}[x]$. *Factorising yields* $f = (x-1)\cdot(x^3+x^2+x-1)\cdot(x3+4)\cdot(x^2+1) = p_1\cdot p_2\cdot p_3\cdot p_4$. *The prime factors* p_1, p_2, p_3 *are real as they have real roots by the fundamental theorem of algebra, but* p_4 *has no real root. Hence* p_4 *is not real. So the real part of* f *is:* $\overline{f} = p_1\cdot p_2\cdot p_3 = x7+2x4+x3-8x+4$.

Let

$$\begin{aligned} f =& x^8y^2z^4 - 2x^7y^3z^2 + x6y^4z^4 + x^6y^4 + x^6y^2z^4 + 2x^6yz^5 - 2x^5y^5z^2 - \\ & 2x^5y^3z^2 - 4x^5y^2z^3 + x^4y^6 + x^4y^4 + 2x^4y^3z^5 + 2x^4y^3z + 2x^4yz^5 + \\ & x4z^6 - 4x^3y^4z^3 - 4x^3y^2z^3 - 2x^3yz^4 + 2x^2y^5z + 2x^2y^3z + x^2y^2z^6 + \\ & x^2y^2z^2 + x^2z^6 - 2xy^3z^4 - 2xyz^4 + y^4z^2 + y^2z^2 \in \mathbb{Q}(y,z)[x]. \end{aligned}$$

Factorising yields that

$$f = (x^2y+z)^2\cdot(xz^2-y)^2\cdot(x^2+y^2+1) = p_1^2\cdot p_2^2\cdot p_3.$$

As p_1 *and* p_2 *have odd degree in* z *(resp. in* y*) they are indefinite and thus real.* x^2+y^2+1 *is positive semi-definite. The real polynomial computed from* f *is* $g = p_1\cdot p_2 = x^3yz^2 - x^2y^2 + xz^3 - yz$.

4 The zero-dimensional radical computation

To explain the main idea used in the algorithm for the zero-dimensional real radical via reduction to the univariate case consider the following example. Let $F := \mathbb{Q}(y_1,\ldots,y_m)$ as in the last section.

Example 2 *Let* $I = \langle x_1 - g_1(x_n), x_2 - g_2(x_n), \ldots, x_{n-1} - g_{n-1}(x_n), g_n(x_n)\rangle \trianglelefteq F[x_1, \ldots, x_n]$ *be given. If* $\overline{g_n}$ *is the real part of* g_n *obtained by the procedure* `RealPoly` *the real radical of* I *is:*

$$\sqrt[re]{I} = \langle x_1 - g_1(x_n), x_2 - g_2(x_n), \ldots, x_{n-1} - g_{n-1}(x_n), \overline{g_n}(x_n)\rangle$$

Proof 6 *Let* $g_n = \prod_{i=1}^{r} p_i^{\alpha_i}$ *be the factorisation of* g_n *in* $F[x_n]$*. Then every ideal* $\langle x_1 - g_1, x_2 - g_2, \ldots, x_{n-1} - g_{n-1}, p_i\rangle$ *is maximal because of the isomorphism*

$$F[x_1, \ldots, x_n]/\langle x_1 - g_1, x_2 - g_2, \ldots, x_{n-1} - g_{n-1}, p_i\rangle \cong F[x_n]/\langle p_i\rangle.$$

As p_i *is prime we conclude that* $F[x_1, \ldots, x_n]/\langle x_1 - g_1, x_2 - g_2, \ldots, x_{n-1} - g_{n-1}, p_i\rangle$ *is a field.*
Now $\langle x_1 - g_1, x_2 - g_2, \ldots, x_{n-1} - g_{n-1}, p_i\rangle$ *is real if and only if* p_i *is real because* $F[x_n]/\langle p_i\rangle$ *is real if and only if* p_i *is real by Propostion 1. Hence*

$$\begin{aligned}
\sqrt[re]{I} &\overset{Cor.2}{=} \bigcap_{M \in \mathrm{Min}(I)\ \text{real}} M \\
&= \bigcap_{p_i \text{ is real}} \langle x_1 - g_1, x_2 - g_2, \ldots, x_{n-1} - g_{n-1}, p_i\rangle \\
&= \langle x_1 - g_1, x_2 - g_2, \ldots, x_{n-1} - g_{n-1}, \prod_{p_i \text{ is real}} p_i\rangle \\
&= \langle x_1 - g_1(x_n), x_2 - g_2(x_n), \ldots, x_{n-1} - g_{n-1}(x_n), \overline{g_n}(x_n)\rangle
\end{aligned}$$

The most important theorem for the zero-dimensional computation in the article of Becker and Neuhaus is the Shape lemma which gives a detailed information on the shape of the reduced Gröbner basis of a radical ideal satisfying the property of being in general position in some way, so that we can obtain the position of an ideal given in the example above.

Lemma 5 (Shape-Lemma) *Let* I *be a zero-dimensional radical ideal in* $F[x_1, \ldots, x_n]$ *with all* d *roots in* $\overline{F}^n$ *having distinct* x_n *coordinates.*

Then the reduced Gröbner basis of I in the lexicographical ordering has the shape

$$G = \{x_1 - g_1(x_n), x_2 - g_2(x_n), \ldots, x_{n-1} - g_{n-1}(x_{n-1}), g_n(x_n)\},$$

where g_n is a square-free polynomial of degree d and the g_i, $i < n$, are polynomials of degree $d - 1$.

Proof 7 *See Lemma 4.5 of [6].*

A naive idea for an algorithm could be:

1. Compute the radical $\sqrt{I}$ of the given ideal I.
2. Test if $\sqrt{I}$ fulfils the shape condition with respect to one variable x_i and compute a reduced Gröbner basis of $\sqrt[re]{I}$ w. r. t. a lexicographical ordering with lowest variable x_i. If not use a random change into general position until this condition is fulfilled.
3. Compute the real radical of $\sqrt{I}$ as described in Example 2 and undo the coordinate change.

As a coordinate change into general position causes a growth of coefficients and terms which slows down the Gröbner bases computations it is important to avoid this change as often as possible. Therefore we give some heuristics, i. e. some kinds of special cases in which we do not have to apply a random coordinate change.

The idea for the algorithm due to Becker and Neuhaus ([1]) has been presented in Example 2 and Lemma 5. In the rest of this section I will present my own algorithm:

As in SINGULAR the primary decomposition of zero-dimensional ideal, in the average case, is very efficient, we can use this algorithm as a black box. The main idea of the primary decomposition due to Gianni/Trager/Zacharias (the command is `primdecGTZ`) was presented in [3] chapter 4.2. Hence we can assume the maximality of all ideals we are dealing with. The next subsection presents some properties for maximal ideals I found.

4.1 How to decide whether a maximal ideal is real

For a maximal ideal there are only two possibilities – either it is real or its real radical is the whole ring. This is the reason why getting criteria for maximal ideals is not difficult. The main idea of this section is to find an heuristic which fulfils the following criteria:

1. Its costs have to be lower in the average case than the costs that a random coordinate change would cost.

2. The decision of realness must be an easy test, i. e. it shouldn't cost too many operations.

3. Our heuristic must cancel out maximal ideals M which are not real as early as possible in the computations.

Here are some properties of maximal ideals that I found during the work on my diploma thesis ([6]). For the definition of orderings and real closed fields I refer to [5].

One obvious property of real maximal ideals is the following corollary.

Corollary 3 *Let $M \lhd \cdot F[x_1, \ldots, x_n]$ be maximal and $f_1, \ldots, f_n$ be the univariate polynomials such that $\langle f_i \rangle = M \cap F[x_i]$. If M is real then every f_i is real too.*

Another simple remark is:

Remark 3 *If $M = \langle f_1, \ldots f_n \rangle \lhd \cdot \mathbb{Q}[x_1, \ldots, x_n]$ is a maximal ideal with every $f_i \in \mathbb{Q}[x_i]$ real, then M is real.*

Proof 8 *This is clear as every f_i has a zero a_i in the common real closed field $\mathbb{R}$. Thus $(a_1, \ldots, a_n) \in \mathbb{R}^n$ is in the real zeros of M.*

Note that this simple remark for the rational numbers is not true for an arbitrary real field F. This remains only true if F is an ordered field. The problem for arbitrary real fields is the following:
A polynomial $f_i \in F[x_i]$ is real if and only if there exist orderings

$\alpha_1, \ldots, \alpha_r$ and the corresponding real closures $R_{\alpha_1}, \ldots, R_{\alpha_r}$ such that f_i has zeros in every R_{α_i}.

But these orderings α_i could occur in a way that there exists no common real closed ground field R_α and no corresponding ordering α of F such that the polynomials f_i all have a root in R_α, which would yield that M is real. The following counter-example for arbitrary real fields clarifies the problem:

Example 3 *Let $M = \langle x^2 + 1 + t, y^2 - t\rangle \lhd \cdot\mathbb{Q}(t)[x, y]$. Then $m_1 = x^2+1+t$ is real in every real closed extension R_α of $\mathbb{Q}(t)$ which admits an ordering α in which $t < -1$ (note that we conclude that m_1 is real as it is indefinite over $\mathbb{R}$), $m_2 = y^2 - t$ is real in every real closed extension R_β which admits an ordering β satisfying $t > 0$. Both types of orderings, the α– and β-orderings, contradict each other.*
In fact M is not real as

$$1^2 + x^2 + y^2 = m_1 + m_2 \in M$$

and hence $1 \in \sqrt[re]{M}$.

Analogous to the Shape Lemma, there holds a stronger property for maximal ideals that can be tested very easily:

Proposition 3 *Let $M \lhd \cdot F[x_1, \ldots, x_n]$ be a maximal ideal and $G = \{g_1, \ldots, g_n\}$ the reduced Gröbner basis of M with respect to any lexicographical ordering with smallest variable x_i. If G has the following properties:*

- *$g_1 \in F[x_i]$ and g_1 is real.*[1]
- *every g_i for $i = 2, \ldots, n$ has odd degree in its leading variable*[2]*.*

[1] G is a triangular set as it is a reduced lexicographical Gröbner basis, wlog we can assume that the univariate polynomial in smallest variable in G is g_1.

[2] Let $f \in \mathbb{Q}[x_1, \ldots, x_n]$. The leading variable of f (short $lvar(f)$) is the largest variable in f, i. e. if

$$f = a_s(x_1, \ldots, x_{k-1})x_k^s + a_{s-1}(x_1, \ldots, x_{k-1})x_k^{s-1} + \ldots + a_0(x_1, \ldots, x_{k-1}),$$

$a_s \in \mathbb{Q}[x_1, \ldots, x_{k-1}] \setminus \{0\}$, for a $k \leq n$, then $lvar(f) = x_k$ and the pseudo leading coefficient of f is $ini(f) = a_s(x_1, \ldots, x_{k-1})$.

Then the maximal ideal M is real.

Proof 9 *Assume for simplicity that $G = \{g_1, \ldots, g_n\}$ is a Gröbner basis satisfying the properties above w. r. t. the ordering $x_1 < x_2 < \ldots < x_n$.*
As $g_1 \in F[x_1]$ is real there exists a real closed field $R \supset F$ such that g_1 has a zero $\alpha_1 \in R$. Now $g_2(x_2, \alpha_1) \in R[x_2]$ has odd degree and thus has a zero α_2 in R by the fundamental theorem of algebra. By the same reason $g_3(x_3, \alpha_2, \alpha_1) \in R[x_3]$ has a zero $\alpha_3 \in R$. Inductively there exists an $\alpha \in V_{R^n}(M)$.
Thus $V_R(M) \neq \varnothing$ and hence, by the definition of the real zero-set of M, $V_{re}(M) \neq \varnothing$. Now by the Real Nullstellensatz $\sqrt[re]{M} = I_F(V_R(M)) = I_F(\alpha) \subset M$. As M is maximal and $V_{re}(M) \neq \varnothing$ we conclude the realness of M.

A last non-trivial condition to test the realness of M is:

Lemma 6 *Let $M = \langle m_1, \ldots, m_n \rangle$ be a maximal ideal in $F[x_1, \ldots, x_n]$ written as a reduced lexicographical Gröbner basis w.r.t to the ordering $x_1 < x_2 < \ldots < x_n$. If M is real, every generator m_i is real.*

Proof 10 *Assume contrary: Thus let i be the smallest index such that m_i is not real. As M is a lexicographical Gröbner basis we get the following cases:*

Case 1: $i = 1$ then $m_1 \in F[x_1]$ and has no real root. So

$$\langle 1 \rangle = \sqrt[re]{m_1} \subset \sqrt[re]{\langle m_1, \ldots, m_n \rangle} = \sqrt[re]{M}.$$

Thus M is not real which is a contradiction.

Case 2: $i > 1$. Let R be an arbitrary real closure of (F, α) w. r. t. an ordering α of F such that $a = (a_1, \ldots, a_n) \in R^n$ is a real point of M (i. e. $a \in V_{re}(M)$). Then we have the following situation:

- *$M' := \langle m_1, \ldots, m_i \rangle = M \cap F[x_1, \ldots, x_i] \lhd \cdot F[x_1, \ldots x_i]$ is real since $(a_1, \ldots, a_i) \in V_R(M') \subset V_{re}(M')$.*

- $M'' := \langle m_1, \ldots, m_{i-1} \rangle = M \cap F[x_1, \ldots, x_{i-1}] \lhd \cdot F[x_1, \ldots x_{i-1}]$ *is real since* $(a_1, \ldots, a_{i-1}) \in V_R(M'') \subset V_{re}(M'')$.

As M' is real, the ordering α of F can be extended in $k(M) = F[x_1, \ldots, x_n]/M$, i. e. $k(M)$ is a formally real field (see Proposition 1). From the first isomorphism theorem, we get:

$$\begin{aligned} F[x_1, \ldots, x_i]/M' \cong & (F[x_1, \ldots, x_{i-1}, x_i]/M'')/(M'/M'') \\ = & ((F[x_1, \ldots, x_{i-1}]/M'')[x_i])/((\langle m_i \rangle + M'')/M''). \end{aligned}$$

Now as $(a_1, \ldots, a_{i-1})$ is a (real) root of the maximal M'' we get that

$$F[x_1, \ldots, x_{i-1}]/M'' \cong F(a_1, \ldots, a_{i-1})$$

which is ordered by $F(a_1, \ldots, a_{i-1}) \cap R^2$. Hence

$$k(M) \cong F(a_1, \ldots, a_{i-1})[x_i]/\langle m_i(a_1, \ldots, a_{i-1}, x_i) \rangle$$

and $k(M)$ is real. Thus the ordering $F(a_1, \ldots, a_{i-1}) \cap R^2$ can be extended to $F(a_1, \ldots, a_{i-1}, a_i) \cap R^2$ (as a_i is a real root of $m_i(a_1, \ldots, a_{i-1}, x_i)$ by the definition of a). But then $m_i(a_1, \ldots, a_{i-1}, x_i)$ is indefinite over R by the sign change criterion (Theorem 2) and thus $m_i(x_1, \ldots, x_i)$ is indefinite over R, too. Now we get from Remark 2 that m_i is real which contradicts the assumption.

Lemma 6 is no equivalence as we can see in the following example:

Example 4 *Let $M = \langle x^3 - 2, y^2 + x^2 - x \rangle \lhd \cdot \mathbb{Q}[x, y]$. Now $x^3 - 2$ is real since $\sqrt[3]{2}$ is in $\mathbb{R}$ and $y^2 + x^2 - x$ is real by Lemma 4 as it is indefinite. But M is not real as $y^2 + \sqrt[3]{2}^2 - \sqrt[3]{2}$ has no real root since $\sqrt[3]{2}^2 - \sqrt[3]{2} > 0$.*

The following corollary is useful to test the realness of prime polynomials $f \in F[x_1, \ldots, x_n]$.

Corollary 4 *Let $f \in \mathbb{Q}[y_1, \ldots, y_m, x_1, \ldots, x_n]$ be an irreducible polynomial. Then f is real considered as polynomial in $F[x_1, \ldots, x_n]$ if and only if f considered as a polynomial in $\mathbb{Q}[y_1, \ldots, y_m, x_1, \ldots, x_n]$ is real.*

Proof 11 $\Rightarrow$**:** *As* $\langle f\rangle F[x_1,\ldots,x_n]$ *is real in* $F[x_1,\ldots,x_n]$*, there exists an* x_i *such that* $\deg_{x_i} f > 0$*. Without loss of generality let* x_n *be this* x_i*. By Theorem 1 we conclude that* $\langle f\rangle F(x_1,\ldots,x_{n-1})[x_n] = \langle f\rangle \mathbb{Q}(y_1,\ldots,y_m,x_1,\ldots,x_{n-1})[x_n]$ *is real. Thus by Lemma 4* $\langle f\rangle \mathbb{Q}\ [y_1,\ldots,y_m,x_1,\ldots,x_n]$ *is real and hence* f *is real considered over* $\mathbb{Q}[x_1,\ldots,x_n,y_1,\ldots,y_m]$*.*

$\Leftarrow$**:** *This is clear as reality commutes with localisation (see Lemma 1).*

Combining all these conditions yields a good heuristic to decide the property of being real for maximal ideals M. Let us first consider a large example in which it was possible to avoid the change into general position completely.

Example 5 *Let*

$$I = \langle (y^3+3y^2+y+1)(y^2+4y+4)(x^2+1),\\ (x^2+y)(x^2-y^2)(x^2+2xy+y2)(y^2+y+1)\rangle \trianglelefteq \mathbb{Q}[x,y]$$

The primary decomposition of I *yields 10 maximal ideals.*

1. $M_1 = \langle y^2+1, x-y\rangle$ *which is not real as* y^2+1 *is not real. Hence it does not satisfy the conditions in Proposition 3 and Corollary 3.*

2. $M_2 = \langle y-1, x^2+1\rangle$ *does not satisfy the Corollary 3 and is thus not real.*

3. $M_3 = \langle y^2+y+1, x^2+1\rangle$ *does not satisfy Corollary 3 and is thus not real.*

4. $M_4 = \langle y^2+1, x+y\rangle$ *does not satisfy Corollary 3 and is thus not real.*

5. $M_5 = \langle y+2, x-2\rangle$ *is real by Proposition 3 or Remark 3.*

6. $M_6 = \langle y+2, x^2-2\rangle$ *is real by Proposition 3 for the ordering* $x < y$ *with the reduced Gröbner basis* $G = \{x^2-2, y+2\}$*.*

7. *$M_7 = \langle y+2, x+2\rangle$ is real by Proposition 3 or Remark 3.*

8. *$M_8 = \langle y^3+3y^2+y+1, x+y\rangle$ is real by Proposition 3 w. r. t. the ordering $y < x$ under which M is a reduced Gröbner bases.*

9. *$M_9 = \langle y^3+3y^2+y+1, x^2+y\rangle$. Here it is not obvious to see if M_9 is real or not. So we have to compute the Gröbner bases w. r. t. both orderings $x < y$ and $y < x$.*
   The Gröbner basis w. r. t. to the lexicographical ordering $x < y$ of M_9 is
   $$G_M = \langle x^6 - 3x^4 + x^2 - 1, y + x^2\rangle.$$
   First we have to test if $x^6 - 3x^4 + x^2 - 1$ is real. We know that $x^6 - 3x^4 + x^2 - 1$ is prime and after applying the `RealPoly` procedure introduced in the last section we get that $x^6-3x^4+x^2-1$ is real. Now we know that M_9 is real by Proposition 3 w. r. t. to the ordering $x < y$.

10. *$M_{10} = \langle y^3+3y^2+y+1, x-y\rangle$ is real by Proposition 3.*

So the real radical of I is

$$\begin{aligned}\sqrt[re]{I} &= M_5 \cap M_6 \cap M_7 \cap M_8 \cap M_9 \cap M_{10} \\ &= \langle y^4+5y^3+7y^2+3y+2, x^4 - x^2y^2 + x^2y - y^3\rangle\end{aligned}$$

In the next subsection I describe a procedure using the criteria introduced above.

After giving this procedure it is easy to describe the algorithm for the zero-dimensional case using a coordinate change into general position.

4.1.1 The procedure `prepare_max`

The procedure `prepare_max` which uses the properties introduced above acts in the following way:

It gets as input a maximal ideal M and returns a list $erg = \overline{M}, j$, where

$$\overline{M} = \begin{cases} \sqrt[re]{M} & \text{if } j = 1, \text{ the change into general position can be avoided} \\ M & \text{if } j = 0, \text{ the change into general position cannot be avoided} \end{cases}$$

I explain my algorithm in pseudo-code. The proof of the correctness of this algorithm follows from the criteria explained above. In the algorithm itself there is no need to check Corollary 3 explicitly. This criterion is checked implicitly in the check of Proposition 3 as we will see.

The procedure `prepare_max` is written as follows:

Algorithm 1
(An heuristic to check if a coordinate change can be avoided)

proc *prepare_max(M)*

INPUT : *a maximal ideal* $M \lhd \cdot F[x_1, \ldots, x_n]$

OUTPUT: *a list* $erg = (\overline{M}, j)$ *s.t.:*

$$\overline{M} = \begin{cases} \sqrt[re]{M} & \text{if } j = 1, \text{ the change into general position can be avoided} \\ M & \text{if } j = 0, \text{ the change into general position can't be avoided} \end{cases}$$

BEGIN

Initialise $P := \{\lambda : \lambda \text{ is a permutation of the variables } \{x_1, \ldots, x_n\}\}$

while $(P \neq \varnothing)$ *do* {

Choose a $\lambda = (x_{j_1}, x_{j_2}, \ldots, x_{j_n}) \in P$

$P := P \setminus \{\lambda\}$

Compute the lexicographical Gröbner basis $M_\lambda = \{f_1, f_2, \ldots, f_n\}$ *of* M *w. r. t. the ordering* $x_{j_1} < x_{j_2} < \ldots < x_{j_n}$. *Now* f_1 *is univariate in the variable* x_{j_1}.

Let $\overline{f_1} :=$ `RealPoly`(f_1) *the real part of* f_1. *As* f_i *is prime there are two possibilities* $\overline{f_1} = 1$ *or* $\overline{f_1} = f_1$.

if $(\overline{f_1} = 1)$

{

$erg := \langle 1 \rangle, 1$

$return(erg)$;

}

According to Proposition 3 search the first position $k \geq 2$ *such that* m_k *has even degree in* x_{j_k}. *Set* $k = n+1$ *if there exists none.*

if $(k > n)$

{

$erg := M, 1$; *(Correctness is clear from Prop. 3)*

$return(erg)$;

}

According to Lemma 6 search from position $(k+1)$ *in* M_λ, *the first non-real generator* m_i.

If there exists a position $i \leq n$ *set* $erg = \langle 1 \rangle, 1$ *and return erg.*

}

If F *is non parametric, i. e.* $F = \mathbb{Q}$ *and every generator of* M *is univariate use Remark 3 and return* $erg := M, 1$.

$erg := M, 0$;

$return(erg)$;

END

In many cases the realness of maximal ideals can be checked only using the procedure `prepare_max`. But it may happen that an ideal fails this test, i. e. the result of $prepare\_max(M)$ is $erg = M, 0$. In this case we have to apply a coordinate change into general position.

Here I used the already well-optimised coordinate change implemented in the primdec.lib.

The method I implemented during my diploma thesis is called `GeneralPos`. It gets a list of maximal ideals which failed the test `prepare_max` as input and returns the intersection of all real maximal ideals of this input.

Let us consider an example. An ideal in which we have to apply a coordinate change into general position was presented in Example 3. Lets have a look at this.

Example 6 *Let* $M = \langle x^2 + 1 + t, y^2 - t\rangle \lhd \cdot\mathbb{Q}(t)[x, y]$. *Choosing the coordinate change*

$$\begin{aligned}\varphi : \mathbb{Q}(t)[x, y] &\to \mathbb{Q}(t)[x, y]\\ x &\mapsto x\\ y &\mapsto y + x + t\end{aligned}$$

we get:

$$\begin{aligned}\varphi(M) &= \langle x^2 + 1 + t, (y + x + t)^2 - t\rangle\\ &= \langle x^2 + 1 + t, x^2 + 2xy + 2tx + y^2 + 2ty + t^2 - t\rangle\end{aligned}$$

Its lexicographical Gröbner basis w. r. t. the ordering $y < x$ *is:*

$$\begin{aligned}G_\varphi = \{&y^4 + 4ty^3 + (6t^2 + t)y^2 + (4t^3 + 4t)y + (t^4 + 6t^2 + 4t + 1),\\ &(-4t - 2)x - y^3 + (-3t)y^2 + (-3t^2 - 2t - 3)y + (-t^3 - 2t^2 - 3t)\}.\end{aligned}$$

Now $y^4 + 4ty^3 + (6t^2 + 2)y^2 + (4t^3 + 4t)y + (t^4 + 6t^2 + 4t + 1)$ *is not real in* $\mathbb{Q}(t)[y]$ *as* $y^4 + 4ty^3 + (6t^2 + 2)y^2 + (4t^3 + 4t)y + (t^4 + 6t^2 + 4t + 1)$ *is positive semi-definite (which can be seen using Lemma 4). Hence as in Example 3 we get that* M *is not real.*

In all my tests it didn't happen often that I had to change into general position for the test of being real. In fact the only examples I found in which there is a need to apply this change are ideals over

transcendent extensions of $\mathbb{Q}$ which are of the form in Example 3, i. e. every generator is univariate and real. For these cases I have not yet found any property to check realness without applying this change. A simple example for an ideal in which this change yields the realness of a maximal ideal is the following:

Example 7 *Let* $M = \langle x^2 + 1 - t, y^2 - t\rangle \lhd \cdot\mathbb{Q}(t)[x, y]$. *Here the same coordinate change as in the example above yields:*

$$\begin{aligned}\varphi(M) &= \langle x^2 + 1 - t, (y + x + t)^2 - t\rangle \\ &= \langle x^2 + 1 - t, x^2 + 2xy + 2tx + y^2 + 2ty + t^2 - t\rangle\end{aligned}$$

Here the Gröbner basis w. r. t. the lexicographical ordering $y < x$ *is:*

$$\begin{aligned}G_\varphi =&\{y^4 + 4ty^3 + (6t^2 - 4t + 2)y^2 + (4t^3 - 8t^2 + 4t)y + (t^4 - 4t^3 + \\ &+ 2t^2 + 1), 2x + y^3 + 3ty^2 + (3t^2 - 4t + 3)y + (t^3 - 4t^2 + 3t)\}.\end{aligned}$$

Now $y^4 + 4ty^3 + (6t^2 - 4t + 2)y^2 + (4t^3 - 8t^2 + 4t)y + (t^4 - 4t^3 + 2t^2 + 1)$ *is real as it is indefinite and the degree of* $2x + y^3 + 3ty^2 + (3t^2 - 4t + 3)y + (t^3 - 4t^2 + 3t)$ *in* x *is odd. Hence* $\varphi(M)$ *is real by Proposition 3, thus* M *is real. In fact* M *is* α*-real in every ordering* α *of* $\mathbb{Q}(t)$ *satisfying the condition* $t \geq 1$.

To see the algorithm `GeneralPos` I recommend looking at Algorithm 4.2 in [6].

4.2 An algorithm to compute the zero-dimensional radical

From the explanation in the last subsections, it is not difficult to get an algorithm which computes the real radical of a zero-dimensional ideal J in $F[x_1, \ldots, x_n]$.

Algorithm 2

proc *RealZero(I)*

INPUT : *a zero-dimensional ideal* $I \trianglelefteq F[x_1, \ldots, x_n]$

OUTPUT: *an ideal* $\overline{J}$ *s.th.* $\overline{J} = \sqrt[re]{I}$

Simplify the ideal $I = \langle f_1, \ldots, f_r \rangle$ *to* $J = \langle g_1, \ldots, g_r \rangle$ *as described in [6] Remark 4.16,*[4]

Compute the associated primes of $Max := \mathrm{Min}(I)$ *with* `primdecGTZ` *or* `primdecSY`*. (This depends on which algorithm is faster.*[4]*).*

Initialise $Prep := \varnothing$ *and* $NonPrep := \varnothing$

while $Max \neq \varnothing$ *do*

{

Choose an $M \in Max$

$Max := Max \setminus \{M\}$

Compute $erg = \overline{M}, j$ *with Algorithm 1.*

If $j = 1$ *and* $\overline{M} \neq \langle 1 \rangle$

{

$Prep := Prep \cup \{\overline{M}\}$

}

else

{

$NonPrep := NonPrep \cup \{\overline{M}\}$

}

$Prepared := \bigcap_{\overline{M} \in Prep} \overline{M}$*:*

$NonPrepared := GeneralPos(NonPrep)$*;*[5]

[4]These operations are applied with a time limit by the aid of the `watchdog` command. watchdog(command, timer) returns the result of the command if the time for the command finishes before the timer.

[5]The idea of this approach was explained with 2 examples in the previous subsection.

According to Theorem 1 we get that

$$\sqrt[re]{I} = \sqrt[re]{J} = Prepared \cap NonPrepared =: \overline{J}.$$

$return(\overline{J})$;

To finish this chapter I give an example in which every path of Algorithm 2 is taken.

Example 8 *Let*

$$\begin{aligned} I = \langle & (x^2y^3 - tx^2y + y^6 - y^5 - ty^4 + t^2 + 1) \cdot (y^3 - t^2y^2 + (-t^3 + t^2 - \\ & - t)y + t^3), (-2t)x^4 - 4tx^2 + (-t + 1)y^6 + (-t^2 + t)y^5 + (t^2 - \\ & - t)y^4 + (-t^4 + t^3)y^2 + (t^4 - t^3)y + (t^5 - t^4 + 2t^3 - 2t), y^7 + \\ & + t^2y^4 - t^2y^3 - t^4, (-t)x^2y^2 + t^2x^2 - y^6 - ty^5 + ty^4 + (-t^3 + \\ & + t^2 - t)y^2 + t^3y + (t^4 - t^3 + t^2)\rangle. \end{aligned}$$

Then every generator of I is simplified in the sense of Remark 4.16.

1. *The primary decomposition of I provides 4 minimal primes which are*

   - $M_1 = \langle x^2 + 1 - t, y^3 + t^2 \rangle$
   - $M_2 = \langle x^2 + t^2 + 1, y^2 + t \rangle$
   - $M_3 = \langle x^2 + 1 - t, y^2 - t \rangle$
   - $M_4 = \langle x^2 + 1 + t, y^2 - t \rangle$

   We set $Max := \{M_1, M_2, M_3, M_4\}$.

2. *$Prep := \varnothing$ and $NonPrep := \varnothing$*

3. *As Max is not empty choose $M_1 \in Max$ and set*

$$Max := Max \setminus \{M_1\} = \{M_2, M_3, M_4\}.$$

4. *prepare_max*$(M_1) = M_1, 1$ *because of Proposition 3. Hence set:*

$$Prep := Prep \cup \{M_1\} = \{M_1\}$$
$$NonPrep := NonPrep = \varnothing$$

5. *As* Max *is not empty choose* $M_2 \in Max$ *and set*

$$Max := Max \setminus \{M_2\} = \{M_3, M_4\}.$$

6. *prepare_max*$(M_2) = \langle 1 \rangle, 1$ *by [6] Lemma 3.2 w. r. t. the lexicographical ordering* $y < x$*. Hence set:*

$$Prep := Prep = \{M_1\}$$
$$NonPrep := NonPrep = \varnothing$$

7. *As* Max *is not empty choose* $M_3 \in Max$ *and set*

$$Max := Max \setminus \{M_3\} = \{M_4\}.$$

8. *prepare_max*$(M_3) = M_3, 0$*. Hence we have to apply a coordinate change and set:*

$$Prep := Prep = \{M_1\}$$
$$NonPrep := NonPrep \cup \{M_3\} = \{M_3\}$$

9. *As* Max *is not empty choose* $M_4 \in Max$ *and set*

$$Max := Max \setminus \{M_4\}.$$

10. *prepare_max*$(M_4) = M_4, 0$*. Hence we have to apply a coordinate change and set:*

$$Prep := Prep = \{M_1\}$$
$$NonPrep := NonPrep \cup \{M_4\} = \{M_3, M_4\}$$

11. *Now* Max *is empty and we set* $Prep = \{M_1\}$.

12. *From the examples 6 and 7 we conclude with the coordinate change* φ *satisfying* $\varphi(x) = x, \varphi(y) = y + x + t$ *that* M_3 *is real and* M_4 *is not real. Hence*

$$NonPrep = \{M_3\}$$

13. *Set*

$$\begin{aligned}\overline{J} &= Prep \cap NonPrep = M_1 \cap M_3 \\ &= \langle y^5 - ty^3 + t^2y^2 - t^3, x^2 + (-t+1)\rangle\end{aligned}$$

Hence the real radical of I *is*

$$\overline{J} = \langle y^5 - ty^3 + t^2y^2 - t^3, x^2 + (-t+1)\rangle.$$

4.3 The general case as reduction

To conclude I shall explain shortly how to compute the real radical with the preparations of this article.

The main theorem for the higher dimensional computation, adapted from [1] Theorem 4.5., is:

Theorem 5 *Let* $I \trianglelefteq F[x_1, \ldots, x_n]$. *For any* $S \subsetneq \{x_1, \ldots, x_n\}$ *let* $J^{(S)}$ *denote an ideal of the quotient ring* $F[x_1, \ldots, x_n] \cdot F(S)$ *satisfying*

$$\dim J^{(S)} \leq 0 \text{ and } I \cdot F(S) \subseteq J^{(S)} \subseteq (I \cdot F(S))_{Iso}.$$

Then

$$\sqrt[re]{I} = \bigcap_{S \subsetneq \{x_1,\ldots,x_n\}} (\sqrt[re]{J^{(S)}} \cap F[x_1, \ldots, x_n])$$

As every $J^{(S)}$ has a dimension less then equal zero we are able to compute there real radicals. Theorem 5 now tells us how to intersect all these ideals properly so that our result will be the real radical. The theory of finding the $J^{(S)}$ uses real isolated points for arbitrary formally real fields. It is explained in detail in [1] chapter 4 or in chapter 5 of [6].

5 Conclusions

Following a short introduction of the basics on real algebra and real radicals, I described how to compute the real radical in the univariate case and in the zero-dimensional case. The univariate case corresponds to the leaves of the reduction tree for computing real radicals. While the univariate case uses theory which can already be found in literature, like Sturm's Theorem (cf. [2]) or the decision of indefiniteness (cf. [4]), section 4, the zero-dimensional case, introduces newly found properties. The decision was to compute the primary decomposition of the zero-dimensional input and to give a heuristic for deciding whether a maximal ideal is real or not. This heuristic yield a procedure `prepare_max` which prepares a maximal ideal in such a way that we can avoid a coordinate change into general position as often as possible. If we can not avoid a coordinate change we use the procedure `GeneralPos`. Its input is a list of maximal ideals where a change can't be avoided. Here a suitably randomised coordinate change is computed such that we can check the properties of `prepare_max` for the transformed maximal ideals and afterwards we intersect all real maximal ideals of this list. Finally, the procedure `RealZero` gets a zero-dimensional input I and computes its primary decomposition. Then it considers separately every maximal ideal and tests if a change is needed to compute the real part. Afterwards it intersects the real radicals of all these 'nice' maximal ideals and restarts the procedure `GeneralPos` for the list of 'bad' ideals. Since the primary decomposition is well-optimised in SINGULAR the advantage of this is a time improvement during the computations. This is because coordinate changes into general position cause a growth of coefficients and terms which slows the Gröbner bases computations down. The idea presented in this abstract avoid such changes as often as possible. Finally the article closes with the description how to compute the arbitrary radical as a reduction to the zero-dimensional case. We have presented an algorithm to compute real radicals which uses the new introduced heuristic `prepare_max` and is thus a time improvement to the algorithm presented by Becker and Neuhaus in [1].

References

[1] E. Becker and R. Neuhaus. On the computation of the real radical. *Journal of Pure and Applied Algebra*, 124:261–280, 1998.

[2] Henri Cohen. *A course in computational algebraic number theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, 1993.

[3] Gert-Martin Greuel and Gerhard Pfister. *A* **Singular** *introduction to commutative algebra*. Springer-Verlag, Berlin, 2002. With contributions by Olaf Bachmann, Christoph Lossen and Hans Schönemann, With 1 CD-ROM (Windows, Macintosh, and UNIX).

[4] Zeng Guangxing and Zeng Xiaoning. An effective decision method for semidefinite polynomials. *J. Symb. Comput.*, 37(1):83–99, 2004.

[5] Manfred Knebusch and Claus Scheiderer. *Einführung in die reelle Algebra*, volume 63 of *Vieweg Studium: Aufbaukurs Mathematik [Vieweg Studies: Mathematics Course]*. Friedr. Vieweg & Sohn, Braunschweig, 1989.

[6] Silke J. Spang. *On the Computation of the real radical*. Diploma Thesis. University of Kaiserslautern, March 2007.

Silke J. Spang,

Received November 9, 2007

Fraunhofer Institute for Industrial Mathematics (ITWM)
Department System Analysis, Prognosis and Control
Kaiserslautern, Germany
E–mail: *silke.spang@itwm.fraunhofer.de*

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

Gröbner Bases for Nonlinear DAE Systems of Analog Circuits

Silke J. Spang

Abstract

Systems of differential equations play an important role in modelling and analysis of many complex systems e. g. in electronics and mechanics. The following article is concerned with a symbolic analysis approach for reduction of the differential index of nonlinear differential algebraic equation (DAE) systems, which occur in the modelling and simulation of analog circuits.

1 Introduction

Systems of differential equations play an important role in modelling and analysis of many complex systems e. g. in electronics and mechanics. For example, the simple oscillator circuit of figure 1, which is part of nearly all analog electronic devices, yields the following DAE

$$C1\left(V_1'(t) - V_2'(t)\right) - I_{L1}(t) = 0 \tag{1.1}$$

$$\frac{V_2(t)}{R1} + C1(V_2'(t) - V_1'(t)) = 0 \tag{1.2}$$

$$V_1(t) + L1 \cdot I_{L1}'(t) = 0 \tag{1.3}$$

where I_{L1} denotes the current through the inductor $L1$ and V_i the voltage between the node i and the ground.

Unlike ordinary differential equation systems (short ODE), proper DAE systems are subject to hidden constraints. These constraints are not explicitely stated in the system of equations, but they constrain the solution within a certain manifold. For instance, in the above DAE

©2008 by Silke J. Spang

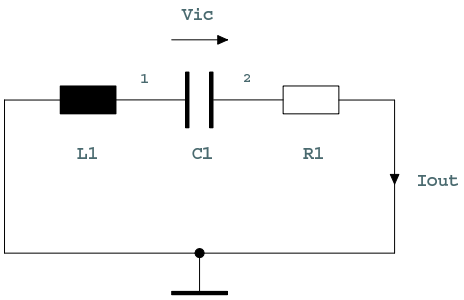


Figure 1. Analog oscillator circuit

there is no possibility to compute an explicit formula for V_2' which does not depend on V_1' and vice versa using algebraic deformations only. Deriving the whole system, we obtain:

$$C1\left(V_1'(t) - V_2'(t)\right) - I_{L1}(t) = 0 \tag{1}$$

$$\frac{V_2(t)}{R1} + C1(V_2'(t) - V_1'(t)) = 0 \tag{2}$$

$$V_1(t) + L1 \cdot I_{L1}'(t) = 0 \tag{3}$$

$$C1\left(V_1''(t) - V_2''(t)\right) - I_{L1}'(t) = 0 \tag{4}$$

$$\frac{V_2'(t)}{R1} + C1(V_2''(t) - V_1'(t)) = 0 \tag{5}$$

$$V_1'(t) + L1 \cdot I_{L1}''(t) = 0 \tag{6}$$

Adding (4) and (5) we get

$$\frac{V_2'(t)}{R1} - I_{L1}'(t) = 0, \tag{7}$$

multiplying (7) with $L1$ and adding it to (3) we end up with

$$\frac{V_2'(t)}{R1} \cdot L1 - V_1(t) = 0. \tag{8}$$

In the same way we get another equation for $V_1'(t)$ and get the following

equation:

$$V_1'(t) = -\frac{R1}{L1} \cdot V_1(t) + \frac{1}{C1} \cdot I_{L1}(t) \tag{9}$$

So the system can be reformulated as an ODE of the following form:

$$V_1'(t) = -\frac{R1}{L1} \cdot V_1(t) + \frac{1}{C1} \cdot I_{L1}(t) \tag{2.1}$$

$$V_2'(t) = -\frac{R1}{L1} \cdot V_1(t) \tag{2.2}$$

$$I_{L1}'(t) = \frac{V_1(t)}{L1} \tag{2.3}$$

In general hidden constraints for such systems can be handled using methods from commutative algebra.

The treatment of linear DAE systems using algebraic methods is straight-forward, but this is not the case for nonlinear terms, e. g. DAEs containing exponential functions. Here some further development of computational methods is necessary to match the needs of this equations which arise from nonlinear circuits.

We describe a method for the detection of such hidden constraints and reformulate the DAE in an ODE like manner. In section 2 some background theory of differential systems and their differential index is explained and an algebraic framework by meanings of rings, ideals and Gröbner bases for the properties of local solvability and being formally integrable are given. The ring of all differential equations up to order q w. r. t. the independent variable t will be reinterpreted as the polynomial ring $A^{(q)}$. In section 3 the computational development during the work with SINGULAR and *Mathematica* on this subfield is described. We will see how such problems can be tackled using polynomial systems in SINGULAR. Systems containing the latter give rise to electrical circuits describing the behavior of transistors and diodes.

Section 4 expands our view to some new classes of functions. We will get some feeling how to tackle exponential functions, sines and cosines and in particular square-roots in an algebraic and polynomial frame. Systems containing exponential functions may give rise to electrical circuits describing the behavior of transistors and diodes. We

will embed all these classes in a new ring D for which we define derivative map φ_D. To this end will see in section 5 that our gained theory applied to some DAE systems good solutions in Analog Insydes. After our preprocessing the last example we detected some equations which gave sufficient constraints to compute the solution with the nonlinear DAE-Solver of *Mathematica*. Concluding with section six we will give some outlook for further development.

I would like to thank the Analog Insydes Team especially Dr. Alexander Dreyer for many fruitful discussions and their support with problems in Analog Insydes. I also want to thank the Fraunhofer Institute for Industrial Mathematics especially my department System Analysis, Prognosis and Control with the department chief Dr. Patrick Lang for giving my the opportunity to work this field. Last I want to thank my advisor Prof. Dr. Gerhard Pfister for being a good friend and advisor which always has an ear for me and my problems.

2 Basics and mathematical background

In this section we will present some algebraic and analytic basics which shall help to understand the next sections.

Suppose a DAE (F) of order q is given. We introduce the **differential index** (cf. [4]) of (F) to be r if a minimum of $r+1$ geometric differentiations of (F) is required until no new constraint is found. Note that this index definition is one out of a group of indices measuring the difficulty of solving DAE systems (cf. [6]).

As already mentioned above, proper DAE systems yield additional constraints to the solution, which are not stated explicitely in terms of equations.

Example 1
Consider the following system (cf. [6]) with functions x_i in the independent variable t

$$x_1' + x_1 = 0 \tag{3.1}$$
$$x_2 x_2' - x_3 = 0 \tag{3.2}$$
$$x_1^2 + x_2^2 - 1 = 0. \tag{3.3}$$

This system admits a hidden constraint $x_1^2 - x_3 = 0$ *which appears after a differentiation of equation* (3.3) *and the elimination* x_1' *and* $x_2 \cdot x_2'$ *by using the equations* (3.1) *and* (3.2). *The above turns out to have differential index two as after two steps there occur no more hidden constraints. In this case the system can be transformed to an ODE.*

Systems of high index are algebraically underdetermined as they have a gap of constraints which only appear after differentiation. These hidden constraints may slow down numerical computations, or make them even impossible. Systems of lower indices have less of these hidden equations and it turns out to be desirable to transform a higher indexed system into one of a lower index. Among the approaches to decrease the differential index is the theory of **locally solvable** and **involutive** systems. (cf. [5]) Here we prolongate and project the given DAE until no new constraint can be found.

We define the **prolongation** and the **projection** of q-th order systems (cf. [3] chapter 2, [4] chapter 2-3), where the prolongation coincides with differentiation and the projection with the elimination of the highest order part.

Definition 2
Let $f_1, .., f_n \in C^m(T, t)$ *be m-differentiable functions in the time* $t \in T \subset K$ *for an interval* T *in a field* K. *If* $f_i^{(j)} = \frac{d^j f_i}{d^j t}$ *denotes the* j*-th derivation of* f_i, *then we denote the space of all* ***differential algebraic equations up to order*** q *of* $f = (f_1, \ldots, f_n)$ *over* K *by* $A^{(q)} = K[f^{(q)}, f^{(q-1)}, \ldots, f', f, t]$.

Now we are able to give a formal definition of projection and prolongation in terms of ring maps and elimination.

Definition 3
Let $D_t : A^{(q)} \to A^{(q+1)}$ *be a formal differentiation, i.e.:*

- $D_t(p \cdot q) = D_t(p) \cdot q + p \cdot D_t(q)$ *(chain rule)*
- $D_t(p + q) = D_t(p) + D_t(q)$
- $D_t(f_i^{(j)}) = f_i^{(j+1)}$ *for all* $f_i^{(j)} \in A^{(q)}$.

- $D_t(a) = 0$ *for all* $a \in K$.

The field $Const(A) = \{a \in A^{(0)} : D_t(a) = 0\}$ *is called the* ***field of constants****. Note that* K *is a subfield of* $Const(A)$*, but they need not be equal.*

Now a DAE system (F) can be transformed into an ideal I of $A^{(q)}$. Recall that an ideal is a subset which is invariant under addition and scalar multiplication and is denoted by $I \trianglelefteq A$. Note that the solutions of (F) do not coincide with the solutions of I. Of course, every solution of (F) corresponds to a solution in I but not vice versa. This is because algebraically the derivative of an f_i is another variable and we have no a priori knowledge about their analytical relationship. The map D_t has its natural extension for ideals $I = \langle g_1, \dots g_r \rangle \trianglelefteq A^q$ given by

$$D_t(I) = \langle D_t(g_1), \dots, D_t(g_r) \rangle.$$

Definition 4
Let I *be an ideal in* $A^{(q)}$*:*

- *The* ***algebraic prolongation*** *of* I *is defined to be*

$$\mathcal{P}(I) = \langle I, D_t(I) \rangle \trianglelefteq A^{q+1}.$$

- *The* ***algebraic projection*** *of* I *is given by*

$$\mathcal{E}(I) = I \cap A^{q-1}.$$

The next definition gives some properties of systems which have a low index.

Definition 5
Let $I \trianglelefteq A^{(q)}$ *be an ideal then*

1. I *is called* ***locally solvable*** *if*

$$\mathcal{E} \circ \mathcal{P}(I) = I.$$

2. *I is called* ***formally integrable*** *if for all* $k \geq 0$

$$\mathcal{E} \circ \mathcal{P}(\mathcal{P}^k(I)) = \mathcal{P}^k(I).$$

We try to use Gröbner basis methods to obtain such forms. First of all we define orderings and Gröbner bases (cf. [2]).

Definition 6 (Ordering)
Let $A = K[x_1, \ldots, x_n]$ *be an affine* K*-algebra. A total ordering* $>$ *on the set of monomials of* A*.*

$$Mon(A) = \{x_1^{a_1} \cdot x_2^{a_2} \cdots x_n^{a_n} : a_i \in \mathbb{N}\}$$

is an antisymmetric binary relation

$$> (x^\alpha, x^\beta) = \begin{cases} 1 & \alpha >_{\mathbb{N}^n} \beta \\ 0 & \alpha = \beta \\ -1 & \beta >_{\mathbb{N}^n} \alpha \end{cases}$$

for an ordering $>_{\mathbb{N}^n}$ *of* $\mathbb{N}^n$*. Additionally*

$$> (x^\alpha, x^\beta) = > (x^\gamma \cdot x^\alpha, x^\gamma \cdot x^\beta)$$

holds. For simplicity we write:

- $x^\alpha > x^\beta$ *if* $> (x^\alpha, x^\beta) = 1$
- $x^\alpha = x^\beta$ *if* $> (x^\alpha, x^\beta) = 0$
- $x^\alpha < x^\beta$ *if* $> (x^\alpha, x^\beta) = -1$.

An ordering is a ***well- or global ordering*** *if* $x^\alpha > 1$ *for all* $\alpha \in \mathbb{N}^n \setminus \{(0, \ldots, 0)\}$*, a* ***local ordering*** *if every* $x^\alpha < 1$ *and* ***mixed ordering*** *otherwise.*

Definition 7 (leading monomial and leading ideal)
Let A *be an affine* K*-algebra and* $>$ *an ordering on* $Mon(A)$*, then:*

1. *For every polynomial*
$$f = f_1 x^{\alpha_1} + f_2 x^{\alpha_2} + \ldots + f_m x^{\alpha_m} \in A \setminus \{0\}$$
with $f_1 \neq 0$ *and* $x^{\alpha_1} > x^{\alpha_2} > \ldots > x^{\alpha_m}$ *let* $\mathrm{LM}(f) = x^{\alpha_1}$ *denote the* ***leading monomial****, which is the biggest monomial w. r. t.* $>$ *in* f*.*

2. *For any ideal* $I \trianglelefteq A$ *let* $\mathrm{L}(I) = \langle \mathrm{LM}(f) : f \in I \rangle$ *denote the* ***leading ideal*** *of* I*.*

Now we are able to define a **Gröbner basis** as a so called "fine form" of an ideal I, see [2, Def. 1.6.1].

Definition 8 (Gröbner basis)
Let $I = \langle f_1, \ldots, f_r \rangle \trianglelefteq A$ *be any ideal. A* ***standard basis*** *is a representation* $\langle g_1, \ldots, g_m \rangle$ *of* I *such that the equality* $\mathrm{L}(I) = \langle \mathrm{LM}(g_1), \ldots, \mathrm{LM}(g_m) \rangle$ *holds. If the underlying ordering* $>$ *is a global one then we call a standard basis just* ***Gröbner basis.***

To represent ideals on computers, we can use Gröbner bases. This form is suitable for computations as it provides a reduction to the monomial case. Note that computing with monomials is only a combinatorial problem. The so-called **normal form** w. r. t. to a set $\{g_1, .., g_m\}$ is defined as follows:

Definition 9 (Normal form, standard representation)
Let $\mathcal{G}$ *denote the set of all finite subsets* $G \subset A$*. A map*
$$\mathrm{NF} : A \times \mathcal{G} \to A, \ (f, G) \mapsto \mathrm{NF}(f|G)$$
is called ***normal form*** *on* A *if* $\mathrm{NF}(0|G) = 0$ *for all* $G \in \mathcal{G}$ *and for all* $f \in R$ *and* $G \in \mathcal{G}$*:*

1. $\mathrm{NF}(f|G) \neq 0 \implies \mathrm{LM}(\mathrm{NF}(f|G)) \notin \mathrm{L}(G)$.

2. *If* $G = \{g_1, \ldots, g_m\}$*, then* $r := f - \mathrm{NF}(f|G)$ *has a* ***standard representation*** *w. r. t.* G*, that is, either it holds* $r = 0$*, or*
$$r = \sum_{i=1}^{r} a_i \cdot g_i, \ a_i \in A,$$
satisfying $\mathrm{LM}(r) \geq \mathrm{LM}(a_i \cdot g_i)$*, such that* $a_i \cdot g_i \neq 0$*, for all* i*.*

Most of the classical problems of ideal theory, e. g. ideal membership, variable elimination, equality of two ideals, etc. can be easily solved using Gröbner bases. To eliminate variables, we use **elimination orderings**.

Simply expressed, we can view them as a separator that makes everything we want to eliminate larger than the elements we want to keep. The best elimination orderings for fast Gröbner basis computations are the so called block orderings (cf. [2] Example 1.2.8.(3)).

Because of the elimination property of Gröbner bases it seems advisable to use the Gröbner basis theory to obtain a good formulation for a given DAE system. We finish this section proving the following lemma.

Lemma 10
Let $I \trianglelefteq A^{(q)}$ be a linear locally solvable DAE. Then I is formally integrable too.

PROOF
We have to show the $\mathcal{E} \circ \mathcal{P}(\mathcal{P}^k(I)) = \mathcal{P}^k(I)$ for all k. As the prolongation of a linear DAE is again linear we conclude that every $\mathcal{P}^k(I)$ is linear. Thus it suffices to show that $\mathcal{E} \circ \mathcal{P}(\mathcal{P}(I)) = \mathcal{P}(I)$. As I is linear all polynomials in I are of degree one. Hence $D_t(I)$ is simply a substitution of variables. So let I be written as Gröbner basis w. r. t. a block elimination ordering $>$ on the ring variables satisfying

$$\{f^{(0]}\} < \{f'\} < \{f^{(2)}\} < \ldots < \{f^{(q)}\}.$$

Then the Gröbner basis G of I can be written in block form:

$$G = \{\underbrace{f_{q1}, \ldots, f_{qn_q}}_{order\ q}, \ldots, \underbrace{f_{11}, \ldots, f_{1,n_1}}_{order\ 1}, \underbrace{f_{01}, \ldots, f_{0n_0}}_{order\ 0}\}$$

Let $F_j = \{f_{j1}, \ldots, f_{jn_j}\}$. Now $D_t(G)$ is again a Gröbner basis as all variables are substituted. In fact

- $D_t(G) = \{D_t(F_q), D_t(F_{q-1}), \ldots, D_t(F_0)\}$
- $D_t^2(G) = \{D_t^2(F_q), D_t^2(F_{q-1}), \ldots, D_t^2(F_0)\}$.

Now

$$\begin{aligned}\mathcal{E} \circ \mathcal{P}(\mathcal{P}(I)) &= \mathcal{E} \circ \mathcal{P}(I + D_t(I))) \\ &= \mathcal{E}(I + D_t(I) + D_t(I + D_t(I))) \\ &= \mathcal{E}(I + D_t(I) + D_t^2(I)) \\ &= I + D_t(I) + \mathcal{E}(D_t^2(I)) = \mathcal{P}(I) + \mathcal{E}(D_t^2(I))\end{aligned}$$

As $P(I) \subseteq \mathcal{E} \circ \mathcal{P}(\mathcal{P}(I))$ always holds, it suffices to show the inclusion $\mathcal{E} \circ \mathcal{P}(\mathcal{P}(I)) \subseteq \mathcal{P}(I)$. This reduces to $\mathcal{E}(D_t^2(I)) \subseteq \mathcal{P}(I)$. Now

$$\begin{aligned}\mathcal{E}(D_t^2(I)) &= \langle D_t^2(F_{q-1}), \ldots, D_t^2(F_1), D_t^2(F_0)\rangle \\ &= D_t(\langle D_t(F_{q-1}), \ldots, D_t(F_1), D_t(F_0)\rangle) \\ &= D_t(\mathcal{E}(D_t(I))) \subseteq D_t(\mathcal{E} \circ \mathcal{P}(I)) \\ &= D_t(I) \subseteq \mathcal{P}(I)\end{aligned}$$

This proves our claim.

3 A computational approach

In the following section a computational approach for interacting the *Mathematica*-based tool Analog Insydes [1] with SINGULAR is explained. One of the main difficulties is to construct a communication bridge between both systems that come from different mathematical application domains. SINGULAR is well optimised for polynomials and Gröbner bases, while Analog Insydes is used for modelling and mixed numeric/symbolic approximation of analog circuits.

3.1 Differentiation and Prolongation

A natural way to implement differentiation is dealing with word rewriting systems. The derivative of a variable is simply represented by another variable. This rewriting process is obtained by the definition of new variables df_i for f_i', ddf_i for f_i'' etc. Then differentiation is obtained by a left shift to the formal derivative. So, to obtain a correct differentiation we simply introduce a map φ defining the derivative of every function. The core of the whole differentiation of pure polynomials in the $A^{(q)}$ is the product rule (see Algorithm 1).

Algorithm 1 PROC productrule ($poly\ f, map\ \varphi$)

Require: a polynomial (resp. monomial) $f \in A^q$ and the derivative map φ of variables
Ensure: a polynomial df which is $f' \in A^{(q+1)}$
  if $(\deg f = 0)$ **then**
    return 0;
  if $(\deg f = 1)$ **then**
    return $\varphi(f)$;
  else
    pick a prime factor p of f;
    $g := \frac{f}{p}$
    $df := g \cdot \varphi(p)$+**productrule**$(g, \varphi) \cdot p$
  return df

Using this underlying core it is possible to obtain a procedure for the differentiation of an ideal. This procedure is called **derivideal**. It gets an ideal I as input and the definition of the derivative map φ, the output is $D_t(I)$, the derivative of I. The prolongation is simply defined by the procedure **Prolongation** which takes the ideal *dae* and a natural number $functionanz$ as arguments where the latter denotes the number of involved functions and is just to generate the derivative map automatically. The following example which is derived from Example 1 shows how the procedure **Prolongation** works.

Example 11

```
> ring r=0,(dx(1..3),x(1..3),t),dp;
> ideal dae=dx(1)+x(1),x(2)*dx(2)-x(3),
  x(1)^2+x(2)^2-1;
//x(1),x(2),x(3) are the 3 functions
> def difring=Prolongation(dae,3);
> setring difring;
> dae;
dae[1]=dx(1)+x(1)
dae[2]=dx(2)*x(2)-x(3)
dae[3]=x(1)^2+x(2)^2-1
```

```
//ddx(i) is x(i)''
> prol;
prol[1]=dx(1)+x(1)
prol[2]=dx(2)*x(3)+x(2)*x(3)-dx(2)
prol[3]=x(2)^2+x(3)-1
prol[4]=dx(2)*x(2)-x(3)
prol[5]=x(1)^2-dx(2)*x(2)
prol[6]=ddx(2)+dx(2)^3+dx(2)^2*x(2)
        -dx(2)*dx(3)-dx(3)*x(2)
prol[7]=ddx(1)+dx(1)
```

The ideal **prol** is the Gröbner basis of the prolongation from Definition 4. The equation $x_1^2 - x_3 = 0$ (cf. Example 1) can be easily derived from **prol[4]** and **prol[5]**. The ring *difring* is ordered by a block ordering admitting $\{t, x_1, \ldots dx_3\} < \{ddx_1, ddx_2, ddx_3\}$. Hence, we see that prol[1..5] is the elimination of the highest derivatives in *prol*. So we see that after defining the prolongation with Gröbner bases it is an easy task to compute the elimination.

3.2 Computing locally solvable systems

Algorithm 2 PROC LocallySolvableDAE(ideal dae, int n)

Require: a DAE dae of q-th order in $A^{(q)}$, N the number of functions
Ensure: a DAE locs of q-th order which is locally solvable and the differential index of dae

```
int difindex = 0;
ideal locs = dae;
ideal buffer = 0;
while buffer ≠ locs do
  buffer = locs;
  locs = InvolutionStep(locs);
  difindex = difindex + 1;
return (locs, difindex);
```

In the previous section we saw that the computation of $\mathcal{E} \circ \mathcal{P}(I)$ for

every I can be implemented easily. This is done in the auxilliary procedure **InvolutionStep**. The natural way to extend this to a procedure which returns a locally solvable system is described in Algorithm 2.

To finish the computations of Example 1 we see the following example:

Example 12

```
> ring r=0,(dx(1..3),x(1..3),t),dp;
> ideal dae=dx(1)+x(1),x(2)*dx(2)-x(3),
            x(1)^2+x(2)^2-1;
> LocallySolvableDAE(dae,3);
[1]:
   _[1]=dx(3)+2*x(3)
   _[2]=dx(1)+x(1)
   _[3]=dx(2)*x(3)+x(2)*x(3)-dx(2)
   _[4]=x(2)^2+x(3)-1
   _[5]=dx(2)*x(2)-x(3)
   _[6]=x(1)^2-x(3)
[2]:
   2
```

Here we see that the differential index of our system is 2 and the equality $x_1^2 - x_3 = 0$ (c.f Example 1) appears as the sixth in the result. The advantage is that we can now derive every hidden constraint in the original DAE from the resulting one. The next two sections will deal with some extensions to new functions that may be included in the nonlinear systems like exponential functions, sines, cosines and squareroots.

4 Integration of more function types

This section describes the main ideas how to extend the algorithmic approach from the last section to the case of rings including more function classes like exponential functions, sines, cosines and squareroots as extensions of $A^{(q)}$. These allow us to formalize the concept of DAE systems including the latter.

4.1 Special extension rings for Exponential functions

First, let us consider systems with exponential functions. Such systems occur in simple analog circuit consisting of transistors and diodes. The special diode equations are called Shockley equations. They explain the connection between current and voltage. The Shockley diode equation is

$$I = I_S \cdot (e^{q \cdot \frac{V_D}{k \cdot T}} - 1),$$

where I is the diode current, I_S is a scalar factor called the saturation current, q - the elementary charge, V_D is the voltage across the diode, k - the Boltzmann constant and T - the temperature.

The extension ring $B^{(q)}$ is defined as follows:

Definition 13
Let n_e be a natural number and $arg = arg_1, \ldots, arg_{n_e}$ be a list of polynomials in $A^{(q)}$. Then we define $e_i := e^{arg_i}$. Now the ring of special exponential DAEs of q-th order is denoted by

$$B^{(q)}_{arg} := A^{(q)}[e_1, \ldots, e_{n_e}] = K[f^{(q)}, \ldots, f', f, t, e].$$

If there is no confusion about arg, we simply write $B^{(q)}$ instead of $B^{(q)}_{arg}$.

Note that for every list arg there is another ring. So the prolongation and of course the index and the local solvability depend on arg. With the additional definition

$$D_t(e_i) = D_t(arg_i) \cdot e_i$$

we get our natural extensions of the above discussed theory. In the next subsection the programs defined in the previous section will be extended for these special rings.

4.2 A solution to the computational task of exponential functions

As written in Section 3 the core of prolongation is how to define the derivative map. To this end we have to get a method to get the exponents of the e_i and to define the derivatives. As we already have

an algorithm representing a derivative map to compute derivatives in $A^{(q)}$, call it $\varphi_{A^{(q)}}$ and **DerivPoly**$(\cdot, \varphi)$, we can describe the derivative map for $B^{(q)}$.

Definition 14
Let x be one of the variables in $B^{(q)}$. We define the derivative map $\varphi_{B^{(q)}}$ as follows

$$\begin{aligned}\varphi_{B^{(q)}} : & B^{(q)} \to B^{(q+1)} \\ & x \mapsto \begin{cases} \varphi_{A^{(q)}}(x) & \text{if } x \in A^{(q)} \\ \textbf{DerivPoly}(arg_i, \varphi_{A^{(q)}}) \cdot e_i & \text{if } x = e_i \end{cases}\end{aligned}$$

Now the prolongation can be extended to exponential functions easily if we know their number and arg.

4.3 How to expand to sines and cosines

As sine and cosine depend on each other by means of their derivatives we extend our ring $B^{(q)}_{arg}$ simultaneously with both sine and cosine on arguments in $B^{(q)}_{arg}$ as follows:

Definition 15
Let n_{trig} be any natural number and $trig_{arg} = trig_{arg_1}, \ldots, trig_{arg_{n_{trig}}}$ a list of polynomials in $B^{(q)}_{e_{arg}}$. We define $s_i := \sin(trig_{arg_i})$ and $c_i := \cos(trig_{arg_i})$ as the ring of special trigonometric functions of q-th order

$$\begin{aligned} C^{(q)}_{trig_{arg}, e_{arg}} := B^{(q)}_{e_{arg}}[s_1, \ldots, s_{n_{trig}}, c_1, \ldots, c_{n_{trig}}] = \\ = K[f^{(q)}, \ldots, f', f, t, e, s, c]. \end{aligned}$$

If there is no confusion about $trig_{arg}$ and e_{arg} we simply write $C^{(q)}$ instead of $C^{(q)}_{trig,earg}$.

To extend our theory of locally solvable systems we have to extract our derivative map.

Definition 16
Let x be one of the variables in $C^{(q)}$. We define the derivative map $\varphi_{C^{(q)}}$ as follows

$$\varphi_{C^{(q)}} : C^{(q)} \to C^{(q+1)}$$
$$x \mapsto \begin{cases} \varphi_{C^{(q)}}(x) & \text{if } x \in B^{(q)} \\ \mathbf{DerivPoly}(trig_{arg_i}, \varphi_{B^{(q)}}) \cdot c_i & \text{if } x = s_i \\ \mathbf{DerivPoly}(trig_{arg_i}, \varphi_{B^{(q)}}) \cdot (-s_i) & \text{if } x = c_i \end{cases}$$

As a last extension we present squareroots.

4.4 Extension to square-roots

As we consider polynomials and no rational functions we need to adjoint both, the squareroot $\sqrt{\cdot}$ and its multiplicative inverse $\frac{1}{\sqrt{\cdot}}$ to the ring $C^{(q)}$. This yields the following definition:

Definition 17
Let $C_{e_{arg}, trig_{arg}}$ as in Definition 15 and let $sqrt_{arg} = sqrt_{arg_1}, \ldots, sqrt_{arg_{n_{sqrt}}} \in C^{(q)}$ be the list of the arguments n_{sqrt} of the squareroot functions $\sqrt{\cdot}_i$. We define for the list $mf := e_{arg}, trig_{arg}, sqrt_{arg}$ the ring of special functions including exponential functions, sines, cosines and squareroots as follows:

$$D_{mf} = C^{(q)}_{trig_{arg}, e_{arg}}[\sqrt{\cdot}_1, \ldots, \sqrt{\cdot}_{n_{sqrt}}, \frac{1}{\sqrt{\cdot}_1}, \ldots, \frac{1}{\sqrt{\cdot}_{n_{sqrt}}}].$$

Additionally extracting the derivative map for the new ring D with

$$\varphi_D(\sqrt{\cdot}_i) := \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot \frac{1}{2} \cdot \frac{1}{\sqrt{\cdot}_i}$$
$$\varphi_D(\frac{1}{\sqrt{\cdot}_i}) := \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot \frac{-1}{2} \cdot \left(\frac{1}{\sqrt{\cdot}_i}\right)^3$$

we get the desired prolongation.

4.5 The structure of D

D fits to all algebraic combinations that are possible with exponential functions, sines, cosines and square-roots. Unfortunately, we have no a priori knowledge about their analytical and geometric relationship. The following subsection tries to fill this gap. First of all let us recall the geometric relationship between sines and cosines, that is

$$\sin(x)^2 + \cos(x)^2 = 1 \ \forall x.$$

Therefore, we have naturally $s_i^2 + c_i^2 = 1$ for all $i \in \{1, \ldots, n_{trig}\}$. Because of the multiplicative relation between the square-roots and their reciprocal we have additionally the conditions $\sqrt{\cdot}_i \cdot \frac{1}{\sqrt{\cdot}} = 1$ for every i. Hence we get the following definition:

Definition 18
Let D_{mf} be defined as above (Definition 17), then we define the ideal I_0 to be

$$I_0 = \langle s_i^2 + c_i^2 - 1, \sqrt{\cdot}_j \cdot \frac{1}{\sqrt{\cdot}_j} - 1, \sqrt{\cdot}_j^2 - sqrt_{arg_j} \ : \\ i \in \{1, \ldots, n_{trig}\}, j \in \{1, \ldots, n_{sqrt}\}\rangle$$

The ideal I_0 represents every algebraic combination which is trivially zero for this special classes. The main question is: does this suffice in general? Hence we want to show that I_0 is already locally solvable. Therefore we consider the following lemma.

Lemma 19
Let D and I_0 be defined as above, then I_0 is formally integrable, especially for the theory of locally solvable set in D it is sufficient to compute the normal forms w. r. t. I_0 after every prolongation step.

PROOF
We will show that $D_t(I_0) \subset I_0$. Therefore we show that $D_t(g) \in I_0$ for every generator g of I_0. We have the following cases:

1. $g = s_i^2 + c_i^2 - 1$ for some suitable i. Then

$$\begin{aligned} D_t(g) &= D_t(s_i^2 + c_i^2 - 1) \\ &= D_t(s_i^2) + D_t(c_i^2) - D_t(1) \\ &= 2 \cdot s_i \cdot D_t(s_i) + 2 \cdot c_i \cdot D_t(c_i) - 0 \\ &= 2 \cdot \mathbf{DerivPoly}(trig_{arg_i}, \varphi_{B^{(q)}}) \cdot s_i \cdot c_i - \\ &\quad - 2 \cdot \mathbf{DerivPoly}(trig_{arg_i}, \varphi_{B^{(q)}}) \cdot c_i \cdot s_i \\ &= 0 \in I_0 \end{aligned}$$

2. $g = \sqrt{\cdot}_j \cdot \frac{1}{\sqrt{\cdot}_j} - 1$ for a suitable j. Then

$$\begin{aligned} D_t(g) &= D_t(\sqrt{\cdot}_j \cdot \frac{1}{\sqrt{\cdot}_j} - 1) \\ &= D_t(\sqrt{\cdot}_j \cdot \frac{1}{\sqrt{\cdot}_j}) - D_t(1) \\ &= D_t(\sqrt{\cdot}_j) \cdot \frac{1}{\sqrt{\cdot}_j} + \sqrt{\cdot}_j \cdot D_t(\frac{1}{\sqrt{\cdot}_j}) - 0 \\ &= \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot \frac{1}{2} \cdot \frac{1}{\sqrt{\cdot}_i} \cdot \frac{1}{\sqrt{\cdot}_j} + \\ &\quad + \sqrt{\cdot}_j \cdot \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot \frac{-1}{2} \cdot \left(\frac{1}{\sqrt{\cdot}_i}\right)^3 \\ &= \frac{1}{2} \cdot \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot [\left(\frac{1}{\sqrt{\cdot}_i}\right)^2 - \\ &\quad - (\sqrt{\cdot}_i \left(\frac{1}{\sqrt{\cdot}_i}\right)) \left(\frac{1}{\sqrt{\cdot}_i}\right)^2] \\ &= \frac{1}{2} \cdot \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot \left(\frac{1}{\sqrt{\cdot}_i}\right)^2 \cdot [1 - (\sqrt{\cdot}_i \left(\frac{1}{\sqrt{\cdot}_i}\right))] \\ &= -\frac{1}{2} \cdot \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot \left(\frac{1}{\sqrt{\cdot}_i}\right)^2 \cdot g \in I_0 \end{aligned}$$

3. $g = \sqrt{\cdot}_j^2 - sqrt_{arg_j}$ for a suitable j. Then

$$\begin{aligned}
D_t(g) &= D_t(\sqrt{\cdot}_j^2 - sqrt_{arg_j}) \\
&= D_t(\sqrt{\cdot}_j^2) - D_t(sqrt_{arg_j}) \\
&= 2 \cdot \sqrt{\cdot}_j \cdot D_t(\sqrt{\cdot}_j) - \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \\
&= 2 \cdot \sqrt{\cdot}_j \cdot \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot \frac{1}{2} \cdot \frac{1}{\sqrt{\cdot}_i} - \\
&\quad - \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \\
&= \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot [2 \cdot \frac{1}{2} \cdot \sqrt{\cdot}_j \cdot \frac{1}{\sqrt{\cdot}_i} - 1] \\
&= \mathbf{DerivPoly}(sqrt_{arg_i}, \varphi_{C^{(q)}}) \cdot g \in I_0
\end{aligned}$$

Hence $\mathcal{P}(I_0) = \langle I_0, D_t(I_0) \rangle = I_0$ and thus $\mathcal{P}^k(I_0) = I_0$ for all $k \in \mathbb{N}$ and especially $\mathcal{E} \circ \mathcal{P}(\mathcal{P}^k(I_0)) = I_0 = \mathcal{P}^k(I_0)$ for all k and thus I_0 is formally integrable.

Algorithm 3 extends Algorithm 2 following Lemma 19.

Algorithm 3 PROC LocallySolvableDAE(ideal dae,ideal I_0)

Require: a DAE dae of q-th order in $D^{(q)}$ and the ideal I_0 from Definition 18

Ensure: a DAE locs of q-th order which is locally solvable and the differential index of dae

```
int difindex = 0;
ideal locs = dae;
I_0=groebner(I_0);
ideal buffer = 0;
while buffer ≠ locs do
  buffer = locs;
  locs = NF(InvolutionStep(locs),I_0);
  difindex = difindex + 1;
return (locs, difindex);
```

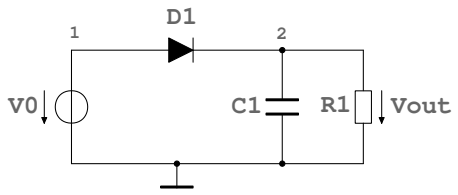


Figure 2. Analog rectifier circuit

5 Computational examples and outlook

To finish this article with the last section we will see how the gained theory applies to the following two examples. This section will be concluded with an outlook for further development.

5.1 Analog rectifier circuit

Example 20
We have given the analog circuit in figure 2. This circuit contains both nonlinear as well as dynamic components, namely the diode $D1$ *and the capacitor* $C1$*. Given numerical element values and a custom input voltage waveform* $V_0 = V_{in}(t)$ *we shall compute the transient response* $V_{\mathrm{out}}(t)$ *across the load resistor* $R1$*. The model parameters* I_S *(saturation current) and* $V_T = kT/q$ *(thermal voltage) are given as* $I_S = 1\,\mathrm{pA}$ *and* $V_T = 26\,\mathrm{mV}$*. The values of the circuit elements are assumed to be* $R1 = 100\,\Omega$ *and* $C1 = 100\,\mathrm{nF}$*. This yields the following DAE system* (F)*:*

$$Ib_{ACdD1}(t) + Ib_{V0}(t) = 0 \tag{4.1}$$

$$-Ib_{ACdD1}(t) + \frac{V_{n2}(t)}{100} + \frac{V'_{n2}(t)}{10^7} = 0 \tag{4.2}$$

$$\frac{e^{\frac{V_{n1}(t)-V_{n2}(t)}{V_T}} - 1 + V_{n2}(t) - V_{n1}(t)}{10^{12}} = Ib_{ACdD1}(t) \tag{4.3}$$

and the input condition $V_{n1}(t) = V_{in}(t)$*.*
The three equalities are transferred from Analog Insydes to SIN-

GULAR, *which continues with computations in the polynomial ring* $\mathbb{Q}[V'_{n1}, V'_{n2}, Ib'_{V0}, Ib'_{ACdD1}, V_{n1}, V_{n2}, Ib_{V0}, Ib_{ACdD1}, t, e_1]$.

Now the procedure ***LocallySolvableDAE*** *returns the following system to* Mathematica

$$Ib_{ACdD1}(t) + Ib_{V0}(t) = 0 \tag{5.1}$$
$$Ib'_{ACdD1}(t) + Ib'_{V0}(t) = 0 \tag{5.2}$$
$$1 + 10^{12} \cdot Ib_{ACdD1}(t) + V_{n2}(t) - e_1 = V_{n1}(t) \tag{5.3}$$
$$10^5 \cdot (100 \cdot Ib_{ACdD1}(t) - V_{n2}(t)) = V'_{n2}(t) \tag{5.4}$$
$$(e_1 + 1)(-10^5 \cdot (100 \cdot Ib_{ACdD1}(t) - V_{n2}(t)) + V'_{n1}(t)) = 10^{12} \cdot V_T \cdot Ib'_{ACdD1}(t) \tag{5.5}$$

where $e_1 = e^{\frac{V_{n1}(t) - V_{n2}(t)}{V_T}}$. *Using normal forms and our knowledge about* V_{n1} *the system can be written as*

$$V'_{n1}(t) = V'_{in}(t) \tag{6.1}$$
$$V'_{n2}(t) = f(t) \tag{6.2}$$
$$Ib'_{ACdD1}(t) = -\frac{(e_1 + 1)(f(t) + V_{n2}(t)) - V'_{in}(t)}{10^{12} \cdot V_T} \tag{6.3}$$
$$Ib'_{V0}(t) = -Ib'_{ACdD1}(t) \tag{6.4}$$
$$V_{n1}(t) = V_{in}(t) \tag{6.5}$$
$$Ib_{ACdD1}(t) = -Ib_{V0}(t) \tag{6.6}$$

where $f(t) = 10^5 \cdot (100 \cdot Ib_{ACdD1}(t) - V_{n2}(t))$. *This constrains the equations of an explicit ODE formulation in the variables* V_{n1}, V_{n2} *and* Ib_{ACdD1}. *This is because* $V_{in}(t)$, *and hence* $V'_{in}(t)$, *have to be explicitly provided by the user. Therefore, the first three equations give formulas for* V'_{n1}, V'_{n2} *and* Ib'_{ACdD1}, *which do not depend on unknown derivatives.*

5.2 A system including sines and cosines

Example 21

$$Ib_{ABdLC}(t) + Ib_{V0}(t) = 0 \tag{7.1}$$
$$D_{IdLC}(t) = Ib'_{ABdLC}(t) \tag{7.2}$$
$$-16\cos(Ib_{ABdLC}(t)) + D_{IdLC}(t) = 4 \tag{7.3}$$

Although the system contains all necessary conditions, it can hardly be solved numerically without preprocessing. Processing the equation system above with the approach of locally solvable sets described earlier, we find the following equations:

$$16\cos(Ib_{ABdLC}(t)) + 4 = D_{IdLC}(t) \tag{8.1}$$
$$Ib_{ABdLC}(t) + Ib_{V0}(t) = 0 \tag{8.2}$$
$$16\cos(Ib_{ABdLC}(t)) + 4 = Ib'_{ABdLC}(t) \tag{8.3}$$
$$Ib_{ABdLC}(t) + Ib'_{V0}(t) + 4 = 0 \tag{8.4}$$
$$64(\sin(Ib_{ABdLC}(t)) + 2\sin(2Ib_{ABdLC}(t))) + D'_{IdLC}(t) = 0 \tag{8.5}$$

If we look at the system, we see that constraint (8.4) is redundant, as it is implicitly given by (8.2) and (8.3). If we additionally cut constraint (8.1) which is implicitly given in constraint (8.5), we get the following system (F):

$$Ib_{ABdLC}(t) + Ib_{V0}(t) = 0 \tag{9.1}$$
$$16\cos(Ib_{ABdLC}(t)) + 4 = Ib'_{ABdLC}(t) \tag{9.2}$$
$$64(\sin(Ib_{ABdLC}(t)) + 2\sin(2Ib_{ABdLC}(t))) = D'_{IdLC}(t) \tag{9.3}$$

With the initial conditions $D_{IdLC}(0) = 1, Ib_{ABdLC}(0) = 0$ *and* $Ib_{V0}(0) = 0$ *the solution to the system computed by Analog Insydes can be seen in figure 3.*

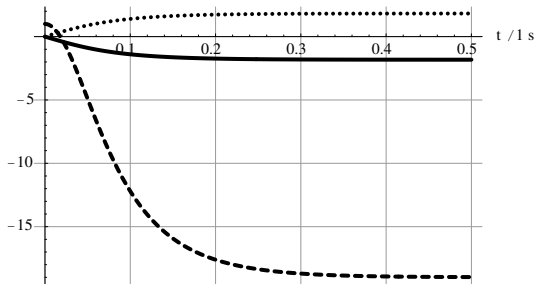


Figure 3. Time integration of 9.1–9.3, where Ib_{ABdLC} $(\cdots)$, D_{IdLC} $(--)$, and Ib_{V0} (—).

5.3 Conclusion and outlook

We have embedded an important class of nonlinear DAE systems into a polynomial frame. This enables us to apply the theory of commutative algebra and Gröbner bases for modelling problems arising from analog circuit analysis. Therefore, we recalled some algebraic basics. We introduced algorithmic procedures for transforming DAEs to systems which are as close as possible to ODEs. After discussing polynomial nonlinear DAEs our approach was extended to systems containing exponential terms. This is an improvement of the known theory of local solvability and formal integrability (cf. [5], [4], [3]). This enables the analysis of important nonlinear components like diodes and transistors.

In further developments, because integrating further function types in Singular would only entail unnecessary work, we have decided to move the prolongation implementation to *Mathematica*. On one hand, this allows us to consider a larger spectrum of DAEs, and on the other hand, the prolongation process may use the specialized *Mathematica* differentiation functions. To transform between Mathematica and Singular representation, we define a mapping between *Mathematica* DAEs and Singular ideals.

The initial results of my research are very promising, more practical

applications will be tackled in the future by using more sophisticated approaches. This will be published in a forthcoming article.

References

[1] J. Broz, A. Dreyer, T. Halfmann, E. Hennig, M. Thole, and T. Wichmann. *Analog Insydes – Release 2.1.1 Manual.* Fraunhofer-Institut für Techno- und Wirtschaftsmathematik, Kaiserslautern, Germany, 2007.

[2] G.-M. Greuel and G. Pfister. *A* **Singular** *introduction to commutative algebra.* Springer-Verlag, Berlin, 2002. With contributions by Olaf Bachmann, Christoph Lossen and Hans Schönemann, With 1 CD-ROM (Windows, Macintosh, and UNIX).

[3] M. Hausdorf. Geometrisch-algebraische Vervollständigung allgemeiner Systeme von Differentialgleichungen. *Diplomarbeit*, 2000.

[4] G. J. Reid, P. Lin, and A. D. Wittkopf. Differential Elimination-Completion Algorithms for DAE and PDAE. *Studies in Applied Mathematics*, 106(1):1–45, 2001.

[5] W. M. Seiler. Indices and solvability for general systems of differential equations. In *Computer algebra in scientific computing—CASC'99 (Munich)*, pages 365–385. Springer, Berlin, 1999.

[6] T. Wichmann. *Symbolische Reduktionsverfahren für nichtlineare DAE-Systeme.* Shaker Verlag, Aachen, Germany, 2004.

Silke J. Spang,

Received January 21, 2008

Fraunhofer Institute for Industrial Mathematics (ITWM)
Department System Analysis, Prognosis and Control
Kaiserslautern, Germany
E–mail: *silke.spang@itwm.fraunhofer.de*

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

Private Key Extension of Polly Cracker Cryptosystems

Nina Taslaman

Abstract

In 1993 Koblitz and Fellows proposed a public key cryptosystem, *Polly Cracker*, based on the problem of solving multivariate systems of polynomial equations, which was soon generalized to a Gröbner basis formulation. Since then a handful of improvements of this construction has been proposed.

In this paper it is suggested that security, and possibly efficiency, of any Polly Cracker-type cryptosystem could be increased by altering the premises regarding private- and public information.

1 Introduction

In 1993, Koblitz and Fellows [1] proposed a public key cryptosystem, *Polly Cracker*, based on the NP-complete problem of solving multivariate systems of polynomial equations over a finite field. This was immediately generalized to a Gröbner basis formulation, where the problem of solving polynomial equations was replaced by the EXPSPACE-complete problem of computing a Gröbner basis for an ideal. Using some general NP- or EXPSPACE-complete problem as the basis for a public key cryptosystem was a daring move, since the failure of Merkle and Hellman's knapsack-based cryptosystem from 1978 [4] had resulted in high scepticism among cryptographers regarding this type of construction. Indeed, a title like *Why you cannot even hope to use Gröbner Bases in Public Key Cryptography* [3] suggests it met a harsh response. The main criticism against the idea was single-break attacks (i.e. individual-message recovery) based on linear algebra.

©2008 by N. Taslaman

However, over the years a plethora of possible countermeasures against these attacks and others has been proposed, as well as different modifications to improve and generalize the initial idea - the most general version as of now seeming to be Ackermann and Kreutzer's generalization to module Gröbner bases over general monoid rings, which allows commonly used public key schemes such as RSA and ElGamal to be formulated as special cases [9].

Rather than continuing in this direction of generalizing the setting, V. Ufnarovski suggested author to investigate altering the rules for private and public information in the Polly Cracker setup. This is the subject of this paper.

To introduce the actors: Alice - intended receiver of secret messages, Bob - sender of such messages, and Eve - enemy, who tries to recover Bob's messages. Messages are restricted to some message space M and encrypted by Bob using some encryption function $F : M \to C$ into ciphertext space C. In a public-key cryptosystem (Williamson 1974 [5], Diffie and Hellman 1976 [6]) there may be many Bob's but only one Alice, i.e. F is publicly known (the *public key*) and anyone may encrypt messages, but (hopefully) only Alice can decipher them. This requires F to be a *trapdoor one-way function*, i.e. while encryption $F(m) = c$ may be computed in polynomial time, the decryption $F^{-1}(c) = m$ may not - except for someone (Alice) knowing some additional trapdoor information which simplifies the computation (the *private key*). As for Eve's part of the game, one distinguishes between *total break attacks*, in which she tries to find the secret key (or some equivalent information) so that she may decrypt any future ciphertext, and *single break attacks*, aimed at decrypting specific individual messages. The basic assumption is always that Eve has access to any encrypted message sent by Bob. One also has to consider the situation that she has temporary access to some decryption black box (e.g. in the form of a compiled decryption program), which she may use to decrypt any finite number of ciphertexts of her choosing. This is the scenario for a *chosen-ciphertext attack*, where Eve's goal is to use this information for a total break attack.

1.1 The Polly Cracker Public Key System

Let $\mathbb{F}_q[X]$ be the set of multivariate polynomials over a finite field $\mathbb{F}_q$ generated by the alphabet $X = \{x_1, ..., x_n\}$. Given a subset F of polynomials, let $\langle F \rangle$ denote the ideal they generate over $\mathbb{F}_q[X]$. Also, given a Gröbner basis $G \subset \mathbb{F}_q[X]$, under some monomial ordering $\preceq$, and a polynomial $f \in \mathbb{F}_q[X]$, let $\bar{f}$ denote the normal form of f over $\langle G \rangle$ with respect to $\preceq$, i.e. $\bar{f} := r_G(f)$ is the unique remainder of f over G under the given monomial ordering. The Gröbner basis version of Polly Cracker may then be described like so:

Cryptosystem 1.1 (Polly Cracker).

KEY GENERATION To set up the system, Alice chooses a Gröbner basis $G \subset \mathbb{F}_q[X]$ under some monomial ordering $\preceq$ and selects a finite subset $P \subset \langle G \rangle$ of the corresponding ideal.

$$\text{PRIVATE KEY: } G \qquad \text{PUBLIC KEY: } P$$

MESSAGE SPACE A subset of all G-normal forms:

$$M \subset \{\ \bar{f} \mid f \in \mathbb{F}_q[X]\ \}$$

ENCRYPTION Bob encrypts a message $m \in M$ by choosing some $p \in \langle P \rangle$ and computing the ciphertext

$$c := m + p \in m + \langle G \rangle$$

DECRYPTION Alice decrypts c by computing its normal form over $\langle G \rangle$:

$$\bar{c} = r_G(c) = r_G(m) + r_G(p) = m + 0$$

1.2 Main Attacks

A total break attack on this cryptosystem generally amounts to computing an equivalent Gröbner basis G' for the public key ideal $\langle P \rangle$ - this would be an equivalent secret key. The general problem of computing a Gröbner basis for a given ideal is NP-complete (see e.g. [7]), and Alice

may choose her P and $\preceq$ from some class of known hard instances, for example by encoding well-studied problems from logic, to ensure giving Eve (the attacker) a hard time here.

Now, if Eve does not succeed in the above she could always try to exploit some possible weakness in Bob's choice of $p \in \langle P \rangle$, letting her decipher at least some of his messages. The most severe criticism against Polly Cracker has been its vulnerability to such single break attacks based on linear algebra, mentioned already in Fellows and Koblitz's original paper [1]. With public key $P = \{p_1, ..., p_s\}$, Bob's p will have the form $p = \sum_{i=1}^{s} h_i p_i$ for some ephemeral polynomials h_i. The main idea is then to consider

$$c = m + \sum_{i=1}^{s} h_i p_i \tag{1}$$

as a linear system of equations, whose unknowns are the coefficients of the polynomials h_i's and m. By guessing the support of these, the linear system might be solvable by usual Gaussian elimination, retrieving m. The countermeasure here is for Alice to choose the setting parameters so as to ensure infeasible system sizes (there is a security/efficiency-tradeoff here), and for Bob to choose his h_i's so as to ensure a certain amount of cancellation in the sum. This calls for quite clever constructions.

1.3 Efficiency Issues

The main problem for implementing Polly Cracker instances stems from the above mentioned security/efficiency-tradeoff. In particular, the so called *message expansion* is an issue here: a message m will be encrypted into a ciphertext polynomial c of, most likely, larger support, so even though $supp(m)$ may be as small as a single constant term, $supp(c)$ may be very big if parameter sizes are not properly restricted, implying issues in storage, transfer and decryption. For example, in [2] Koblitz presents a study-example of a Polly Cracker instance (the Graph Perfect Code Instance) based on a perfect code problem from graph theory, and for sufficient security suggests using a polynomial

ring with 500 indeterminates x_i. However, even narrowing it down to 200 one gets ciphertexts of about 60'000 monomials for this instance (see [8]). All serious attempts at practical, implementable Polly Cracker instances have to deal with this issue, which tends to make them somewhat technical.

2 Related Work

In [10] (2004), Levy-dit-Vehel and Perret describe how to construct Polly Cracker instances based on 3-SAT problems from logic, i.e. so that a total break attack may be P-reduced to some well-studied hard 3-SAT instance, while at the same time providing resistance against the classical linear algebra attacks. The latter is achieved by the use of an elaborate generating algorithm for $p \in \langle P \rangle$, together with suggested parameter sizes resulting in a message expansion of about 1500 terms, which is at least manageable but still not suitable for practical use.

The efficiency issue is addressed more directly in [11] (2002), where Ly presents a cleverly constructed, however somewhat technical, modification of Polly Cracker called Polly Two. This cryptosystem can be viewed in three different polynomial settings via a ring homomorphism: domain- goal- or quotient ring, each setting providing security in its own way and simultaneously taking care of the efficiency/sequrity trade-off. In the goal-ring setting this cryptosystem reduces to a Polly Cracker instance with very large parameter sizes, thus handling the linear algebra attacks. Legal users operate in the domain ring where parameter sizes are quite small, with a message expansion of less than 100 terms. This would be acceptable for practical use, however setting up concrete instances seems to be somewhat difficult (e.g. finding a suitable homomorphism).

In [12] (2004), T. Rai generalizes Polly Cracker to noncommutative polynomial rings, inspiration being that this allows ideals for which no finite Gröbner bases exist. The idea here is for Alice to take a secret key Gröbner basis G, finite as usual, but with a public key subset $P \subset G$ so that no finite Gröbner basis exists for $\langle P \rangle$. This means that Eve cannot even theoretically succeed in the usual total break attack. Another

benefit comes from the use of two-sided ideals, leading to quadratic (rather than linear) systems of coefficients in the single break attacks. Unfortunately, finding suitable ideals for concrete instances turns out to be a challenging task. Also, no experimental data is provided, so it is unclear how efficient instances of this system would be.

Going further along the generalizing path, Ackermann and Kreuzer in [9] take the Polly Cracker scheme all the way up to a setting of modules (generalizing the ideals in Polly Cracker) over general monoid rings (generalizing the standard polynomial rings). This could be a promising framework for future cryptosystems (no such instances are provided), but even in its abstract formulation it is of direct interest since most well-known public key schemes seem to let themselves be formulated as special cases, e.g. RSA, ElGamal and even recent attempts at group-based public key schemes.

3 Extending The Private Key in Polly Cracker

Studying the Polly Cracker construction (Cryptosystem 1.1), we make the following observations:

1. The monomial ordering $\preceq$ used is seemingly assumed to be a public domain parameter - at least the advantages of keeping it private is, to our knowledge, never pointed out. The idea here is the following:

   > Alice could choose a Gröbner basis G under some ordering $\preceq$ so that $\langle G\rangle$-normal words with respect to $\preceq$ are not necessarily $\langle G\rangle$-normal with respect to other orderings.

   This would imply that even if Eve managed to find some Gröbner basis $\tilde{G}$ for $\langle P\rangle$, unless she guesses the correct monomial ordering, she cannot expect messages to be preserved in an attempted decryption, i.e. it might be that

   $$r_{\tilde{G}}(c) = r_{\tilde{G}}(m) \neq r_G(m) = m$$

2. The public setting for Polly Cracker is a polynomial ring over some finite field $\mathbb{F}_q$. It is never motivated why the cardinality of this field should be public information. In fact, Bob could encrypt messages perfectly well in $\mathbb{Z}[X]$, with Alice taking the ciphertext $(mod\ p)$ before proceeding as usual with decryption, if we just require the coefficients of messages to be bounded so that they are not destroyed by the $(mod\ p)$ computation.

While the idea of private monomial ordering works with the usual Polly Cracker scheme, keeping the field cardinality private requires some adjustments of the scheme.

3.1 Polly Goes Private - With p

To concretize these ideas, let us first for simplicity of discussion consider the case $\mathbb{F}_q = \mathbb{Z}_p$ for some large prime number p. For a set of polynomials $F \subset \mathbb{Z}_p[X]$, let $\langle F \rangle_p$ denote the usual ideal they generate in $\mathbb{Z}_p[X]$, and let $\langle F \rangle_{\mathbb{Z}}$ denote the ideal F generates when lifted to $\mathbb{Z}[X]$, i.e.

$$\langle F \rangle_{\mathbb{Z}} := \{ \sum_{f \in F} f h_f \mid h_f \in \mathbb{Z}[X] \}$$

Note that

$$\langle F \rangle_{\mathbb{Z}}\ (mod\ p) = \langle F \rangle_p \tag{2}$$

Cryptosystem 3.1 (Polly Cracker with Private $\preceq$ and p).

KEY GENERATION Alice chooses some big prime p, a positive integer $q < p$, a finite Gröbner basis $G \subset \mathbb{Z}_p[X]$ under some monomial ordering $\preceq$, and a finite subset $P \subset \langle G \rangle_{\mathbb{Z}}$.

PRIVATE KEY: p, G, $\preceq$ PUBLIC KEY: P

MESSAGE SPACE M: G-normal forms under $\preceq$ in $\mathbb{Z}_p[X]$ with coefficients bounded by q.

ENCRYPTION Bob chooses $f \in \langle P \rangle_{\mathbb{Z}}$ and encrypts a message $m \in M$ into the ciphertext

$$c := m + f \in \mathbb{Z}[X]$$

DECRYPTION Alice decrypts c by first computing

$$c' = c \ (mod\ p) = m + f_p \in \mathbb{Z}_p[X]$$

where $f_p := f \ (mod\ p)$ and then

$$\bar{c'} = r_G(c') = r_G(m) + r_G(f_p) = m + 0$$

Decryption follows from (2):

$$f \in \langle P \rangle_{\mathbb{Z}} \quad \Rightarrow \quad f \ (mod\ p) \in \langle P \rangle_p \subset \langle G \rangle_p$$

Before proceeding with the case of higher prime-power cardinality, let us first discuss the effects of this private key alteration.

3.1.1 Security gain

The main idea of keeping p private is that it blows up the complexity of a total break attack. As before, this attack amounts to finding a Gröbner basis (under some lucky monomial ordering) for $\langle P \rangle_p$. While this can be made hard even when p is known, without this knowledge Eve could at best try searching through primes $p' > q$, and for each try finding a Gröbner basis for $\langle P \rangle_{p'}$.

Also, forcing users to compute over $\mathbb{Z}[X]$, rather than $K[X]$ for some field K, Eve cannot use scalar inverses in her attacks. Since Gaussian elimination without using scalar inverses leads to intermediate coefficient swell, this means that linear algebra attacks grow more costly.

3.1.2 Efficiency possibilities

The decryption procedure now consists of two steps: first a modulo operation, which is fast, and then the usual reduction, which may be costly. Alice has a possibility to speed up the decryption procedure

here by choosing some public key polynomials $p_i \equiv 0 \ (mod\ p)$, so that much of the ciphertext is simplified in the first (fast) decryption step. While tempting for very efficient decryption, Alice should not take every $p_i \equiv 0 \ (mod\ p)$, however, since this would make p a common factor of all public-key coefficients, which could be detected by Eve.

3.1.3 Issues and countermeasures

By limiting message coefficients to $q < p$, there is a trade-off between the size of the message space and the additional security provided by keeping p secret. However, if q and p are large enough, this should not be a major concern.

A more serious effect is that, since Bob encrypts over $\mathbb{Z}[X]$, the coefficients of the ciphertext may grow big, which can be cumbersome. To limit this effect he should not choose ephemeral key polynomials with too big coefficients. The Chinese remainder theorem could also be used for more efficient transmission:

With α the largest coefficient of a ciphertext polynomial c, Bob multiplies relatively prime numbers n_i, of manageable size, so that the product $N := n_1 \cdots n_r \geq \alpha$. He then computes

$$\begin{cases} c_1 = c & (mod\ n_1) \\ \quad ... & \\ c_r = c & (mod\ n_r) \end{cases} \tag{3}$$

and sends the ciphertext tuple

$$C := \{(c_1, ..., c_r), (n_1, ..., n_r)\}$$

Here coefficients of the c_i's are bounded by $max\{n_1, ..., n_r\}$. Alice then uses the Chinese remainder theorem to solve (3), recovering $c \ (mod\ N) = c$ with full coefficients, and she may proceed as before.

Note, however, that while coefficient sizes may be controlled by this method, we have to pay in the number r of ciphertext polynomials.

3.1.4 Chosen-ciphertext attack

In a private letter, Rai suggests a chosen-ciphertext attack aimed at finding our secret p: Eve could e.g. enumerate primes $q_i > q$ and encrypt fake messages of the form

$$\tilde{m} = q_1 m_1 + ... + q_k m_k$$

where each m_i is a monomial in the message space. If it would happen that some $q_i = p$, the corresponding term would decrypt to zero and the decryption black box she has temporary access to would return $\tilde{m} - q_i m_i$, revealing $p = q_i$.

Note that such a fake message after decryption would contain some coefficients $q_j > q$, which was not allowed in the message space. Hence, to avoid this attack, the decryption black box should be set to detect any such fake ciphertexts (decrypting to terms with coefficients larger than q) and return an error message if that happens.

3.2 Polly Goes Private - With p^n

Now suppose $\mathbb{F}_q = \mathbb{F}_{p^n}$ for some prime p and $n > 1$, and let α denote a generating element for this field via some primitive degree-n-polynomial in $\mathbb{Z}_p[\alpha]$. We use α-power notation as default for nonzero field elements. Let us define a homomorphism from the ring of univariate polynomials $f(s)$ over $\mathbb{Z}$ into $\mathbb{F}_{p^n}$ by

$$\tilde{\varphi} : \mathbb{Z}[s] \to \mathbb{F}_{p^n}; \quad s \mapsto \alpha$$

and extend it to a homomorphism from $\mathbb{Z}[s][X]$ into $\mathbb{F}_{p^n}[X]$ as:

$$\varphi : \mathbb{Z}[s][X] \to \mathbb{F}_{p^n}[X]; \quad f(s)w \mapsto \tilde{\varphi}(f)w \tag{4}$$

where w denotes a word with letters from X. This φ will be used by Alice to translate Bob's messages in $\mathbb{Z}[s][X]$ into the ordinary Polly Cracker setting $\mathbb{F}_{p^n}[X]$. We will need some notation here in order to recognize corresponding key polynomials in these two settings.

Given $f = \sum \alpha^k w_k$ in $\mathbb{F}_{p^n}[X]$, let f_s denote the polynomial obtained in $\mathbb{Z}[s][X]$ by simply replacing every α by s. Then, corresponding to

the definitions in the prime-cardinality case, for $F \subset \mathbb{F}_{p^n}[X]$ let $\langle F \rangle_{p^n}$ be the usual ideal generated by F over $\mathbb{F}_{p^n}[X]$, i.e.

$$\langle F \rangle_{p^n} := \{ \sum_{f \in F} f g_f \mid g_f \in \mathbb{F}_{p^n}[X] \}$$

and let

$$\langle F \rangle_{\mathbb{Z}[s]} := \{ \sum_{f \in F} f_s h_f \mid h_f \in \mathbb{Z}[s][X] \}$$

be the ideal generated by the corresponding polynomials f_s over $\mathbb{Z}[s][X]$. Since

$$\varphi(f_s h_f) = \varphi(f_s)\varphi(h_f) = f\varphi(h_f)$$

we have

$$f \in \langle F \rangle_{\mathbb{Z}[s]} \quad \Rightarrow \quad \varphi(f) \in \langle F \rangle_{p^n} \tag{5}$$

Now, Alice may keep p and n secret while letting Bob compute over $\mathbb{Z}[s][X]$. Using φ she may then translate his ciphertext into a standard Polly Cracker ciphertext in $\mathbb{F}_{p^n}[X]$. By 5, this works if the message space is restricted properly. The details are as follows:

Cryptosystem 3.2 (Polly Cracker with Private $\preceq$ and p^n).

KEY GENERATION Alice chooses a prime number p, some $n > 1$, a finite Gröbner basis $G \subset \mathbb{F}_{p^n}[X]$ under some monomial ordering $\preceq$, a finite subset $P \subset_R \langle G \rangle_{\mathbb{Z}[s]}$ and some $r < p^n - 1$.

PRIVATE KEY: $\mathbb{F}_{p^n}$, G, $\preceq$ PUBLIC KEY: P

MESSAGE SPACE Linear combinations of G-normal words $w_i \in \mathbb{F}_{p^n}[X]$ with coefficients s^k where $k < r$, i.e.

$$M = \{ \sum s^{k_i} w_i \mid k_i \leq r,\ r_G(w_i) = w_i \}$$

ENCRYPTION Bob chooses $f \in \langle P \rangle_{\mathbb{Z}[s]}$ and encrypts a message $m = \sum s^{k_i} w_i \in M$ into the ciphertext

$$c := m + f \in \mathbb{Z}[s][X]$$

DECRYPTION Alice decrypts c as

$$r_G(\varphi(c)) = r_G(m_\alpha + \varphi(f)) = m_\alpha + 0$$

where $m_\alpha = \sum \alpha^{k_i} w_i$ is the message m only with the symbol s replaced by α.

Here decryption follows from 5:

$$f \in \langle P \rangle_{\mathbb{Z}[s]} \subset \langle G \rangle_{\mathbb{Z}[s]} \quad \Rightarrow \quad \varphi(f) \in \langle G \rangle_{p^n}$$

Note that the message is preserved in two steps: First it is preserved by φ since its coefficients are of form s^k for $k < q^n - 1$ (so there is no modulo-effect in the exponent), and then it is preserved in reduction over G, as usual for Polly Cracker, being a normal form.

In this description we have, for clarity, used the different symbols s and α to distinguish Bob's computations over $\mathbb{Z}[s][X]$ from field computations. Of course we might as well let Bob use the same symbol α and compute over $\mathbb{Z}[\alpha][X]$ - the important thing is that he is not able to interpret α as the field element in $\mathbb{F}_{p^n}$.

Example 3.1 (Toy Example). For demonstration, we give a very small example in $\mathbb{F}_{2^3}[x, y]$. A translation table for power/polynomial representation of the field elements in $\mathbb{F}_{2^3}$ is given by:

α^k	$r_k(\alpha)$	α^k	$r_k(\alpha)$
-	0	α^3	$\alpha + 1$
1	1	α^4	$\alpha^2 + \alpha$
α	α	α^5	$\alpha^2 + \alpha + 1$
α^2	α^2	α^6	$\alpha^2 + 1$

KEY GENERATION Take the Gröbner basis

$$G = \{x - \alpha^5, y - \alpha^2\} \in \mathbb{F}_{2^3}[x, y]$$

and preliminary public key polynomials

$$\hat{p}_1 = x^2 + \alpha xy + 1, \quad \hat{p}_2 = \alpha^2 xy + \alpha y^2 + \alpha^3$$

Over $\mathbb{F}_{2^3}$ we have $\hat{p}_1(\alpha^5, \alpha^2) = \hat{p}_2(\alpha^5, \alpha^2) = 0$, so

$$\hat{p}_1, \hat{p}_2 \in \langle G \rangle_{p^n}$$

We multiply these by some polynomials in $\mathbb{Z}[\alpha][x, y]$ to form public key polynomials $p_1, p_2 \in \langle G \rangle_{\mathbb{Z}[\alpha]}$, for example:

$$p_1 = \hat{p}_1 \cdot (5\alpha^7 x + 1) = 5\alpha^7 x^3 + 5\alpha^8 x^2 y + x^2 + \alpha xy + 5\alpha^7 x + 1$$

$$p_2 = \hat{p}_2 \cdot (4\alpha^2 y - \alpha) = 4\alpha^4 xy^2 + 4\alpha^3 y^3 - \alpha^3 xy - \alpha^2 y^2 + \alpha^5 y - \alpha^4$$

For message restriction we choose $r = 6 < 2^3 - 1$.

PRIVATE KEY: $\mathbb{F}_{2^3}$, $G = \{\, x - \alpha^5,\ y - \alpha^2 \,\}$

PUBLIC KEY: $P = \{\, p_1,\ p_2 \,\}$ from above

MESSAGE SPACE G-normal forms in this case are just constants:

$$M = \{\alpha^k \mid k \leq 6\}$$

ENCRYPTION Suppose Bob wants to send us the message $m = \alpha^6$. He chooses ephemeral polynomials in $\mathbb{Z}[\alpha][x, y]$:

$$h_1 = 3y - \alpha, \quad h_2 = xy + \alpha^2$$

and computes the ciphertext in $\mathbb{Z}[\alpha][x, y]$:

$$c = m + p_1 h_1 + p_2 h_2 =$$

$$4\alpha^4 x^2 y^3 + 4\alpha^3 y^4 x - \alpha^3 x^2 y^2 - \alpha^2 xy^3 + (15\alpha^7 + 3)x^2 y +$$

$$(15\alpha^8 + 4\alpha^6 + 4\alpha^5 + 3\alpha)xy^2 + 4\alpha^5 y^3 - (5\alpha^8 + \alpha)x^2 -$$

$$(5\alpha^9 + \alpha^5 + \alpha^4 + \alpha^2)xy - \alpha^4 y^2 + (19\alpha^7 + 3)y - (5\alpha^8 + \alpha)$$

Note that Bob's choice of the last term α^2 in h_2 gives cancellation of the message $m = \alpha^6$ in c.

DECRYPTION Upon receiving c as above, we first compute in $\mathbb{F}_{2^3}$ (using the translation table):

$$\varphi(c) = 0 + 0 + \alpha^3x^2y^2 + \alpha^2xy^3 + (1+1)x^2y +$$
$$(\alpha + 0 + 0 + \alpha)xy^2 + 0 + (\alpha + \alpha)x^2 +$$
$$(\alpha^2 + \alpha^5 + \alpha^4 + \alpha^2)xy + \alpha^4y^2 + (1+1)y + (\alpha + \alpha)$$
$$= \alpha^3x^2y^2 + \alpha^2xy^3 + xy + \alpha^4y^2$$

Then, with $G = \{x - \alpha^5, y - \alpha^2\}$ we have:

$$r_G(\varphi(c)) = \varphi(c)(\alpha^5, \alpha^2) = \alpha^3 + \alpha^6 + 1 + \alpha = \alpha^6 = m$$

⌟

4 Conclusion

An extension of the private key in Polly Cracker has been suggested. In particular, an adjustment of the scheme to private field cardinality could be used to increase complexity of standard attacks (total- as well as single break), while at the same time providing means to control efficiency of decryption by introducing a fast preliminary decryption step before the usual reduction. This scheme adjustment is very simple in Polly Cracker instances over $\mathbb{Z}_p[X]$. The case of higher prime power coefficient fields requires a bit more theory, but in the end does not increase the complexity of the system. An issue that arises is the possible occurrence of large integer coefficients in the ciphertext. Modular techniques could be used to handle this effect.

It would remain to test these ideas on realistic Polly Cracker instances.

References

[1] M. Fellows, N. Koblitz, *Combinatorial cryptosystems galore!*, Finite fields: theory, applications and algorithms, Contemporary Mathematics Volume 168, 1994.

[2] N. Koblitz: *Algebraic aspects of cryptography*, Algorithms and Computation in mathematics, 3. Springer Verlag, 1998.

[3] B. Barkee, D. C. Can, J. Ecks, T. Moriarty, R.F. Ree: *Why you cannot even hope to use Gröbner Bases in Public Key Cryptography - An open letter to a scientist who failed and a challenge to those who have not yet failed*, Journal of Symbolic Computation, 18, pp. 497–501, 1994.

[4] R. Merkle, M. Hellman: *Hiding Information and Signatures in Trapdoor Knapsacks*, IEEE Trans. Information Theory, 24(5), pp.525–530, 1978.

[5] M. J. Williamson: *Non-Secret Encryption Using a Finite Field*, 1974, `http://www.mirrors.wiretapped.net/security/info/reference/cesg-publications/History/secenc.pdf`.

[6] W. Diffie, M. Hellman: *New Directions in Cryptography*, IEEE Transactions on Information Theory, vol. IT-22, pp: 644–654, 1976.

[7] E. Mayr, A. Meyer, *The complexity of the word problems for commutative semigroups and polynomial ideal*, Adv. Math. Vol 46 no.3, pp. 305–329, 1982.

[8] D. Hofheintz, R. Steinwandt: *A "Differential" Attack on Polly Cracker*, IEEE International Symposium on Information Theory, Proceedings of ISIT 2002, p.211.

[9] P. Ackermann, M. Kreuzer: *Gröbner Basis Cryptosystems*, Universität Dortmund, 2006, `http://www.springerlink.com/content/174x321n05136859/fulltext.pdf`

[10] F. Levy-dit-Vehel, L. Perret, *A Polly Cracker system based on Satisfiability*, Progress in Computer Science and Applied Logic, Vol. 23, Birkhäuser, pp.177-192, 2004.

[11] L. V. Ly: *Polly Two - a public-key cryptosystem based on Polly Cracker*, Ruhr-Universität, 2002.

[12] T. S. Rai: *Infinite Gröbner Bases and Noncommutative Polly Cracker Cryptosystems*, Ph.D thesis, Virginia Polytechnic Institute and State University, 2004, http://scholar.lib.vt.edu/theses/available/etd-03262004-082608/unrestricted/rai_etd.pdf

Nina Taslaman,

Received February 18, 2008

E–mail: *ninus.af.quark@gmail.com*

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

On the Cancellation Rule in the Homogenization

Victor Ufnarovski

Abstract

We consider the possible ways of the homogenization of non-graded non-commutative algebra and show that it should be combined with the cancellation rule to get the mathematically adequate correspondence between graded and non-graded algebras.

1 Introduction

The homogenization is a standard instrument in the commutative algebra. From the computational point of view it is useful because homogeneous algorithms are often more efficient, allowing to save memory (for example cleaning a lot when the current degree is done). In the non-commutative case the situation is much less trivial, because the connection between non-graded algebra and graded algebra obtained by the homogenization is not so obvious as in the commutative case. First of all there are several ways to homogenize. If t is a homogenizing variable and one wants to homogenize a non-commutative polynomial f of the degree k the obvious way is to multiply all the monomials in f that have the degree less than k by the corresponding power of t. But how to do it? From the left? From the right? In the middle?

The answer depends on our aim. Suppose we want to calculate the Gröbner basis G of given non-graded algebra and our goal is to obtain it from the Gröbner basis G^* of the corresponding graded algebra which we get using the homogenization of the relations. It would be nice to get it using the dehomogenization procedure as in the commutative

©2008 by V. Ufnarovski

case, i.e. simply putting $t = 1$. Is it possible? Do we really get the Gröbner basis of our non-graded algebra?

An easy example $x^2 = x$ shows that we should be careful about the choice of the ordering: if $t > x$ then $tx > x^2$ and the leading word tx in $tx - x^2$ will be not the leading word after dehomogenization. But suppose that we have solved this problem (and it is not so difficult). Suppose even more that we know that after the dehomogenization we get the correct Gröbner basis. There are still some problems. The first one reflects the fact that 1 commutes with all other variables, but t does not. From the computational point of view it means that the calculating of Gröbner basis G^* may be much more complicated than in the corresponding non-graded algebra. A couple of tests shows that this is the case: almost any non-trivial example creates a huge Gröbner basis G^*, almost always we get infinite Gröbner basis even in the case where the non-graded Gröbner basis is finite. One of the explanation of this phenomena is that though we get Gröbner basis G after dehomogenization, normally it is not minimal, because the reduction works differently in graded and non-graded case. As example, suppose that the leading terms of Gröbner basis in our graded algebra look as txy^kt for all $k > 0$. It is obvious that we get a minimal Gröbner basis G^*. But after dehomogenization we get the set of leading terms xy^k of Gröbner basis G, which is far from being minimal. The term xy alone should be the leading term of the minimal Gröbner basis, but how to avoid the unnecessary calculations of the infinite set in G^*?

One more or less evident attempt to solve this problem is to introduce extra commuting relations: $tx = xt$ for any variable x and demand $tx > xt$. Then all other words in the Gröbner basis of our graded algebra will have the form ft^k, where the word f does not contain t. Words in the example above should be replaced by xy^kt^2 and we can do the reduction already on the level G^*, so xyt^2 be the only leading word term remaining in the minimal Gröbner basis and we achieved our goal in this case. Can we in general hope that the minimal Gröbner basis be still minimal Gröbner basis after dehomogenization? Much more often, but it is still not the case! To see the reason, consider the following example.

Example 1 *The algebra* $A = \langle x, y | x^2 - 1, xy^2 - 1 \rangle$ *has the set* $G = \{y^2 - x, x^2 - 1\}$ *as a Gröbner basis if* $y > x$. *If we homogenize the relations using the commuting homogenizing variable* $t > y > x$ *we get the graded algebra*

$$\langle t, x, y | x^2 - t^2, xy^2 - t^3, tx - xt, ty - yt \rangle.$$

Its Gröbner basis is infinite. Even it contains such elements as $y^2t - xt^3, x^2 - t^2$, *which should be sufficient to obtain* G, *it contains also infinitely many other elements, for example, of form*

$$y^2(xy)^{4k-2}t^2 - t^{8k}, k = 1, 2, \ldots$$

The reason for the trouble is the presence of t in the leading word y^2t. Because of it the leading monomials containing y^2 cannot be reduced (as they are in G).

The remedy for this trouble is far from the being trivial and the main aim of this article is to find it. Shortly the idea is that it is not sufficient to homogenize the relations. We should work in another factor-algebra, where leading terms of the corresponding Gröbner basis do not contain t (commutativity relations $tx = xt$ are the only exceptions). We describe this algebra below. Shortly the rule is as follows: during the Gröbner basis calculations cancel t, if it appears in all the terms. The resulting reduced Gröbner basis will be minimal after the dehomogenization. Let us discuss all the details more carefully (but more formally).

2 Homogenization and dehomogenization

Let $K\langle X \rangle$ be a free algebra over the field K and t be an additional (homogenizing) variable. For any homogenous element $u \in K\langle X \rangle$ of the degree k and any $m \geq k$ we define $u^{*(m)} \in K\langle X, t \rangle$ as ut^{m-k}. If $u \in K\langle X \rangle$ is an arbitrary element, written as the sum of its homogeneous components $u = \sum u_i$, and still having degree $k \leq m$ we define $u^{*(m)}$ as $u = \sum u_i^{*(m)}$ and u^* as $u^{*(k)}$. In other words $u^* = \sum u_i t^{k-i}$, if $\deg u_i = i$. So, $u^* = u$ if and only if u is homogeneous.

To dehomogenize some element $v \in K\langle X, t\rangle$ we simply replace all occurrences of t by 1. In other words, if $v = v(X, t)$ we define $v_* = v(X, 1)$.

For example,

$$(x^2 + y)^* = x^2 + yt; (x^2 + y)^{*(3)} = x^2t + yt^2;$$

$$(x^2 + yt)_* = x^2 + y; (tx - xt)_* = 0.$$

The following statement is trivial, but useful.

Lemma 1 *a) The map $v \to v_*$ is a homomorphism from $K\langle X, t\rangle$ to $K\langle X\rangle$.*

b) $(u^*)_* = u$ for any $u \in K\langle X\rangle$. ■

Note that the map $u \to u^*$ is not a homomorphism and in general not always $(v_*)^* = v$. The following definition helps to choose elements that almost have this property.

Definition 1 *A word $g = ft^l$ is canonical, if $l \geq 0$ and f does not contain variable t. A canonical element of $K\langle X, t\rangle$ is a linear combination of some canonical words of the same length.*

Note that canonical elements are by the definition homogeneous. The following lemma shows their importance.

Lemma 2 *a) Every homogeneous element in $K\langle X, t\rangle$ can be uniquely written as a sum of the canonical element and the element belonging to the ideal, generated by the set $S = \{tx - xt | x \in X\}$.*

b) If v is a canonical element then $v = (v_*)^*t^d$, where d is the minimal power of t dividing some word in v. In particular, $v = (v_*)^*$ if and only if v cannot be written as wt.

Proof. a) is evident and is a trivial application of the Gröbner bases theory.

b) is sufficient to check for a canonical word: if $g = ft^i$ and $|g| = k$ then

$$g_* = f, f^{*(m)} = ft^{m-(k-i)} = gt^{m-k}$$

for any $m \geq k - i$. So, if $v = \sum_j \alpha_j g_j = \sum_j \alpha_j f_j t^{i_j}$ is a canonical element of the degree k, then $v_* = \sum_j \alpha_j f_j$ has degree $k - d$, and

$$(v_*)^* = \sum_j \alpha_j g_j t^{(k-d)-k} = vt^{-d}.$$

■

3 Homogenized ideal

Let $A = K\langle X\rangle/I$, where I is some ideal which will be fixed for the rest of this article. In general I (and A) are not graded and our idea is to study A with the help of graded algebra $B = K\langle X, t\rangle/I^*$, where I^* contains all homogenized elements of I and (to be able to work with the canonical elements only) all the commutators $tx - xt$. More formally, I^* is an ideal in $K\langle X, t\rangle$, generated by all homogenized elements $u^*, u \in I$ and the set $S = \{tx - xt | x \in X\}$. We want to prove some elementary properties of I^*.

Lemma 3 *a) If $u \in I$ is homogeneous, then $u \in I^*$.*
b) If $v \in I^*$ then $v_* \in I$.
c) If $v \in K\langle X, t\rangle$ is homogeneous, then $v \in I^* \Leftrightarrow v_* \in I$.
d) If $vt \in I^*$ then $v \in I^*$.

Proof. a) $u = u^*$ and belongs to I^*.
b) Consider a map ϕ which is the composition

$$K\langle X, t\rangle \to K\langle X\rangle \to A = K\langle X\rangle/I,$$

where the first arrow corresponds to the homomorphism $v \to v_*$, and the second is the natural homomorphism. Then $v_* \in I \Leftrightarrow v \in \ker\phi$. Because $S \subset \ker\phi$ and for every $u \in I$, according to Lemma 1, $u^* \in \ker\phi$, we have that $I^* \subset \ker\phi$, which proves b).

c) The implication $v \in I^* \Rightarrow v_* \in I$ follows from b). On the other hand, according to Lemma 2, $v = w + s$, where w is a canonical element and s belongs to the ideal, generated by S. Now $v_* = w_* + s_* = w_*$ so $v_* \in I \Leftrightarrow w_* \in I$ and, according to Lemma 2, $v = w + s = (w_*)^* t^d + s \in I^*$ if $v_* \in I$.

d) follows from c) because I^* is a homogeneous ideal. ■

4 Eliminating ordering

Suppose that $>$ is an admissible ordering on free monoid $\langle X\rangle$ such that $|f| > |g| \Rightarrow f > g$, where $|f|$ is the length of a word f. We will extend it to the eliminating ordering on free monoid $\langle X, t\rangle$, namely for any two words $f, g \in \langle X, t\rangle$ we put

$$f > g \Leftrightarrow \begin{cases} |f| > |g| \\ \quad or \\ |f| = |g|, \quad f_* > g_* \\ \quad or \\ |f| = |g|, \quad f_* = g_*, \quad f >_{lex} g, \end{cases}$$

where $>_{lex}$ is a pure lexicographical ordering, extending $>$ such that the letter t is larger than any letter from X. Note that $t < x$, but $tx > xt$ for any $x \in X$. This ordering is also admissible and has some special properties that we want to use.

Lemma 4 *Let $v \in K\langle X, t\rangle$ be a canonical element, g be its leading word. Then*

a) If $\deg_t g = k$ then $v = wt^k$, for some canonical element w.

b) Leading term of v_* is g_*.

c) If $u \in K\langle X\rangle$ then the leading word of u in $K\langle X\rangle$ is the same as leading word of u^* in $K\langle X, t\rangle$.

Proof. Recall that v is homogeneous.

a) If h is another word in v then $deg_t h \geq \deg_t g$, otherwise $|h_*| > |g_*|$. So, $h = h't^l$ with $l \geq k$ and $v = wt^k$.

b) In the same notations, if $l > k$ then $|g_*| > |h'| = |h_*|$. Otherwise $l = k$ and $g > h \Leftrightarrow g_* > h_*$ (we can cancel t^k).

c) The leading term of u^* does not contain t according to a). Because it depends only on the words of highest length in u we can use b). ■

5 Normal words and Gröbner basis

From now we fix the eliminating ordering. We want to study the relation between the Gröbner basis for I and Gröbner basis for I^*. Let us recall that the subset G of I is its Gröbner basis if for any $u \in I$ there exists an element $g \in G$ such that its leading word (or leading monomial in another terminology) $lm(g)$ is a subword of the leading word $lm(u)$. Words that are not divisible by any $lm(g), g \in G$ (or equivalent by any $lm(u), u \in I$) are called normal and if we denote the set of the normal words by N then $K\langle X\rangle = KN \oplus I$ (direct sum of vector spaces), so N can serve as a basis for factor-algebra $A = K\langle X\rangle/I$ (see e.g. [2] for the details). Suppose that G is a minimal Gröbner basis for I. Our aim is to describe a minimal Gröbner basis G^* for I^* and the corresponding set of normal words N^* in $K\langle X, t\rangle$. Note that N^* is not the same set as $\{n^*|n \in N\}$, which is the same as N.

Theorem 1 *a) A word $f \in \langle X, t\rangle$ is normal relative I^* (i.e $f \in N^*$) if and only if it is canonical and $f_* \in N$.*

b) If G is a minimal Gröbner basis for I then $G^* = S \cup \{g^*|g \in G\}$ is a Gröbner basis for I^*. It is minimal, if G does not contain elements of degree 1 or constants.

c) If $G = \{1\}$ then $\{1\}$ is a minimal Gröbner basis for I^* too.

d) If $Y \subset X$ is the set of leading monomials in G that have degree 1, then to obtain a minimal Gröbner basis for I^* from that one in b) we need only to take away all the commutators $ty - yt, y \in Y$.

Proof. a) Because S is a subset of I^* a normal word should be canonical. Let f be a canonical word, $f = ht^k, f_* = h$.

If f is not normal then it is a leading word of some homogeneous $v \in I^*$ (because I^* is homogeneous). Then by Lemma 3 $v_* \in I$ and according to Lemma 4 h is its leading term, so $f_* = h$ is not normal.

On the other hand if $f_* = h$ is not normal, then h is the leading word of some $u \in I$. According to Lemma 4 $u^* \in I^*$ has h as the leading term, so f is the leading term of $u^*t^k \in I^*$. This conclusion finishes the proof that $f \in N^*$ if and only if $f_* \in N$.

b) Because $g^* \in I^*$ for every $g \in G$ the set G^* is a subset of I^* and it remains to proof that every leading word f of some $u \in I^*$ is divisible by some leading term of G^*. Because f is not normal it is evident for non-canonical words: $tx = lm(tx - xt)$ is a subword for some $x \in X$. If $f = ht^k$ is canonical then, according to a), $h \notin N$ and is divisible by the leading word of some $g \in G$. But $g^* \in G^*$ has the same leading word by Lemma 4 and word is a subword of f too.

If G does not contain any element of degree less then two then no leading term of G^* can be a subword of the leading term of some $s \in S$. Because G is minimal, G^* should be minimal too.

c) is evident and for d) we need only to note that $ty - yt$ can be written in the factor-algebra as linear combination of other commutators and we do not need it. ■

6 Rabbit Strategy in the Calculating of Gröbner basis

Now, when we get the good definition of the homogenization ideal the question is how to get Gröbner basis for the ideal I^* practically, starting from the generating set R for the ideal I? We know, that we need to homogenize the elements in R, we know, that we need to add the commuting relations $xt - tx$ from S, but it is not sufficient to get all the canonical elements in I^*, as Example 1 shows. Fortunately we need only to slightly modify the main algorithm for Gröbner basis calculations to get the desired result.

Definition 2 *The cancellation rule: if $u = vt^k$ is a canonical element and $k \geq 0$ is as maximal as possible then replace u by v. Formally: replace u by $(u_*)^*$.*

Theorem 2 *Let $R \subset K\langle X\rangle$ be the generating set of the ideal I. Consider the eliminating ordering (as above) and the following algorithm. Homogenize R, add $S = \{tx - xt | x \in X\}$ and use the standard Gröbner basis calculation algorithm (Mora's algorithm) with the following modification: every time when we get a new canonical element u that should*

be added to the Gröbner basis add instead the element, obtained by the cancellation rule.

The resulting set G^* is the Gröbner basis for the ideal I^*. After dehomogenization (setting $t = 1$) we get the Gröbner basis G for the ideal I. Moreover, G^* is minimal if and only if G is minimal. In particular if I has a finite Gröbner basis we get it after finitely many steps.

Proof. Consider the process of calculating the Gröbner basis for I and compare it with the modified algorithm creating G^*. By the construction and according to Lemma 4 all leading monomials from G^* (except those that correspond to S) do not contain t. From this follows that those two processes deal with the same leading monomials. The only possible difference could be in the reduction, but the cancellation rule, commutativity rules for t and ordering are specially designed to take care about this problem: the reduction process looks similar too (see example below). So, for every $g \in G$ we get g^* added to the Gröbner basis. According to the previous theorem we get Gröbner basis for I^* (and no other elements, because we are always inside I^*). Thus G is obtained from G^* using the dehomogenization, which proves all the statements in the theorem. ■

Let us check how this algorithm works in the Example 1. As above we suppose that $y > x > t$, but work in the eliminating ordering. We start from the same set:

$$tx - xt, ty - yt, x^2 - t^2, xy^2 - t^3.$$

Rewriting x^2y^2 in two different ways we get the element

$$x(xy^2 - t^3) - (x^2 - t^2)y^2 = t^2y^2 - xt^3 \to y^2t^2 - xt^3 = u.$$

The main difference now is that we should apply the cancellation rule and add the cancelled element $v = y^2 - xt$ to our Gröbner basis. Now we can throw away the element $xy^2 - t^3$ (it is reduced to zero using v) and we are done: no more new elements appear. The dehomogenization gets the desired result.

The algorithm described in this theorem was used in the Computer Algebra package Bergman (see [3]). Initially Bergman was elaborated for the graded algebras only. This restriction makes it more efficient. To be able to use Bergman in the non-graded situations we introduced so called Rabbit strategy, close to the strategy, described in the last theorem. More exactly, dealing with non-graded algebras, Bergman homogenize them and uses the cancellation rule during the calculations. This means that the calculations cannot be done degree by degree as for graded case, but sometimes (when we used the cancellation rule) we need to go back to the lower degrees. This jumping between the degrees explains the name of the strategy and in fact is organized using three parameters: maximum degree, starting degree and step s. We do all the calculations degree after degree until the maximum degree. But when we pass the starting degree we are ready to jump. We pass s degrees and, if we have found that the cancellation rule was used, we jump back to the corresponding degree and pass next s degrees and so on until the maximum degree will be achieved. In the case we get Gröbner basis completely the dehomogenized set G is the minimal Gröbner basis for our non-graded algebra. If not, the user is informed that obtained set G may be incomplete. The important property of the Rabbit strategy is that if we have a finite Gröbner basis in our non-graded algebra than using sufficiently large maximum degree we will obtain this Gröbner basis and the user will be informed about this.

7 n-chains and Anick resolution

As we have seen above the ideal I^* is the correct way to work with the homogenization. We want to underline this fact even more by showing (without complete proofs) that in fact we can use I^* and G^* to work with the homological properties. For simplicity we restrict ourselves by the case when Gröbner basis G has no elements of the degree less then two, so both G and G^* are minimal. We also suppose that the elements in I have no constant terms, so K be a trivial module both for graded and non-graded algebra. We want to compare Anick resolutions for them.

Let us recall that the sets C_n of n-chains are defined recursively. First of all, $C_{-1} = 1, C_0 = \{X\}$, where X is our alphabet and for every $x \in X$ its tail is x itself.

The set C_{n+1} consists of those words fr with $f \in C_n, 1 \neq r \in N$ which have the following properties:

- If $f = gs$, where s is the tail of f then $sr \notin N$.
- If $r = r'x$, where $x \in X$ then $sr' \in N$.

The normal word r is uniquely determined by the word fr and is its tail.

Recall that the set C_1 is exactly the set of the leading words of any minimal Gröbner basis (and depends on ideal I and ordering only). Now we want to describe the set of n-chains for the ideal I^*.

Theorem 3 *a) The set of n-chains for the ideal I^* is the union of two different sets for $n \geq 0$: $C_n^* = C_n \cup tC_{n-1}$.*

b) Every element of C_n has the same tail as for ideal I.

c) If $f = tg \in tC_{n-1}$ then for $n > 0$ it has the same tail as g and for $n = 0$ the tail is the word t itself.

Proof. Easy induction. Base for $n = 0$ is trivial, for $n = 1$ follows from the Theorem 1. In general, if fr is $(n+1)$-chain for I^* with $n \geq 1$, then $f = gs$ is n-chain for I^* and r, s are normal (for I^*), but sr is not. If $r = r'y, y \in X \cup t$, then sr' is normal. According to Theorem 1 a) we have $y \neq t$ (otherwise sr and sr' are normal simultaneously). Because r is normal r' does not contain t neither. At last, by the induction, the tail s does not contain t. So we decide the question of normality exactly as in I. If $g \in C_n$ we can conclude that $f \in C_{n+1}$, but if $g \in tC_n$, say $g = th, h \in C_{n-1}$, then th and h have the same tale and the fact that ths is $(n+1)$-chain is equivalent to the fact that hs is n-chain for I. ■

Let us recall that n-chains are used for the constructing of Anick resolution (see [1, 2]), namely for the trivial module K over algebra $A = K\langle X\rangle/I$. It looks as

$$\cdots C_n \otimes A \to C_{n-1} \otimes A \cdots \to C_{-1} \otimes A \to K$$

The differentials d_n are recursively defined for any n-chain f, which we identify with $f \otimes 1$. The last theorem allow us to see how in fact Anick resolution is lifted from the non-graded algebra A to the graded algebra $B = K\langle X, t\rangle / I^*$. We skip the proof of the technical details of this process, and only formulate its most important properties.

Theorem 4 *If d_n^* are differentials in the Anick resolution for trivial $B-$module K then*

a)If $f \in C_n$ then $d_n^*(f) = (d_n(f))^*$

b)If $f = tg \in tC_{n-1}$ then $d_n^*(tf) = td_{n-1}^*(g) + (-1)^n gt$.

c) $v \in Ker\ d_n^* \Leftrightarrow v_* \in Ker\ d_n$ for any canonical element v.

This and previous theorem gives also some hint how to extract the information about the homology of A from the homology of B. We see for example that in the monomial case the Betti numbers are nothing else than the differences of the corresponding Betti numbers for B, because in the monomial case the Betti numbers are equal to the number of the corresponding n-chains. Of course, we do not need to homogenize monomial algebras, but the last theorem shows that we can calculate the Betti numbers in the similar way in general case. It does not work if we only homogenize the relations. This again shows that the homogenization should be combined with the cancellation rule to get the correct mathematical connection between non-graded and graded algebras.

This article takes its origin from the discussion of the properties of Anick resolution with Ed Green and the author is very grateful to him for all his ideas that have helped to write this article.

References

[1] Anick, D., On the homology of associative algebras, Trans. Am. Math. Soc.,296, No 2, (1986), pp.641–659.

[2] Ufnarovski ,V.: Combinatorial and Asymptotic Methods of Algebra in "Algebra-VI" (A.I.Kostrikin and I.R.Shafarevich, Eds), En-

cyclopaedia of Mathematical Sciences, Vol. 57 , Springer,(1995), pp.5–196.

[3] http://servus.math.su.se/bergman

Victor Ufnarovski,

Received February 19, 2008

Centre for Mathematical Sciences, Mathematics, Lund
Institute of Technology, Lunds University,
P:O. Box 118, SE-22100,
Lund, Sweden
E–mail: *ufn@math.su.se*

Computer Science Journal of Moldova, vol.**16**, no.1(46), 2008

Approaches to automated construction of graphical shells for computer algebra systems

Alexander Colesnicov Svetlana Cojocaru
Ludmila Malahova

Abstract

The paper proposes a calculator model of the graphical shell to be used for computer algebra systems. The calculator shell model is described. Then the techniques of semi-automated construction of such shell are discussed. The motivation of the approach based on the domain model is given. We describe also two possible component assembly methods, static and dynamic, and our experience with them. We motivate the selection of dynamic component assembly.

1 Introduction

We suppose the existence of programs executing symbolic computations in computer algebra *(engines)* whose developers need to provide modern graphical shell with their systems. Computer algebra is widely used in many areas, including pure and applied mathematics, theoretical physics, chemistry, engineering, technology, etc. Multitude of solved problems makes investigators to create specialized engines in the cases when use of general purpose systems is inefficient, or the necessary functionality is not implemented even in commercial systems. As a rule, creators of such systems have not enough time, resources, and qualification to develop shells for them. It isn't unusual that rich mathematical ideas implemented in an engine are enveloped in poorly designed interface. Our own experience with the Bergman computer algebra system (CAS) and review of other systems illustrate this [1].

©2008 by A.Colesnicov, S.Cojocaru, L.Malahova

The absence of the user-friendly standard shells makes such systems less popular because of requiring special knowledge and skills, e.g., in programming, to use them.

Another problem of computer algebra engines is multitude of their data formats and the implied difficulty in communication between different engines.

Investigations show that CAS interface developer provides some or all of the following features:

- 2-D presentation of mathematical expressions,
- Editing of mathematical expressions that includes sub-expression manipulation,
- Windows that model sheets of paper and combine texts, formulas, and graphics,
- Processing and presentation of long expressions,
- Simultaneous use of several CAS, which implies the necessity to solve problems of data conversion, configuration management, and communication protocols,
- Interface extensibility providing additions of new menus, new fragments of on-line documentation, etc.,
- Guiding of the user during the whole period of his/her problem solving,
- The system should be self-ex'planatory; its operational mode should be understandable directly from the experience of interaction with the system,
- Control over problem formulation correctness and over information necessary to solve it.

The primary scope of a shell is creation of a comfortable environment for a mathematician or another specialist that uses mathematical apparatus. It would be preferable for these users to input data and to

obtain mathematical results in their natural 2-dimensional form. The linear form of input can be used also as the linear input is faster but it imposes additional conventions to enter powers, indices, fractions, etc., or uses additional characters. It is necessary also to provide possibilities to edit expressions, integrate them with a usual text, and obtain results in a form suitable for publication of an article (e.g., LaTeX) or in Internet (e.g., MathML).

The syntactic check of the entered mathematical expressions and the spelling check of accompanying text would be also desired features.

We see that functions of the graphical shell are almost independent of the engine.

We propose therefore a universal shell implementing the calculator model and constructed from the ready-made components [2]. Moreover, we successfully used several engines at once with such shell. This solves many problems of incompatibility of data formats in different CASs, and solves partially the problem of their interconnection.

Sec. 2 defines and discusses the calculator model of the graphical shell. We describe there the details of its work and its interaction with the CAS engines.

There are two approaches to the automated construction of such shells. Both approaches are based on the component programming (CP) and differ mainly in the technique of component assembly that can be static or dynamic.

At the first approach (static component assembly), we successfully combined the CP and the aspect-oriented programming (AOP). This technique is described in Sec. 3.

It is possible to construct the shell using dynamic component assembly. This second technique was developed over the Eclipse platform. The details are described in Sec. 4.

In the Conclusion (Sec. 5) we compare both approaches, describe their advantages and shortcomings, and motivate our decision to use the second approach.

2 The calculator model of the user interface

A usual numerical calculator works step-by-step: you enter numbers, and select one of possible operations. The calculator executes the operation and shows the result that can be used as an operand for the next operation.

The calculator model of the CAS shell behaves quite similarly. You enter a mathematical object (e.g., an ideal that is presented as a list of polynomials with coefficients from some field), select a possible action (e.g., calculation of the Gröbner basis of the corresponding algebra) and start a CAS engine that executes the operation. The result is a new object (in our example, a new list of polynomials), and it can be used for further calculations (e.g., for reduction of polynomials).

The shell implements the input and output of mathematical object in the form suitable for the user; the engines implement all calculations.

Fig. 1 shows one of variants of our shell that supports two CASs: Bergman[1] and Singular[2].

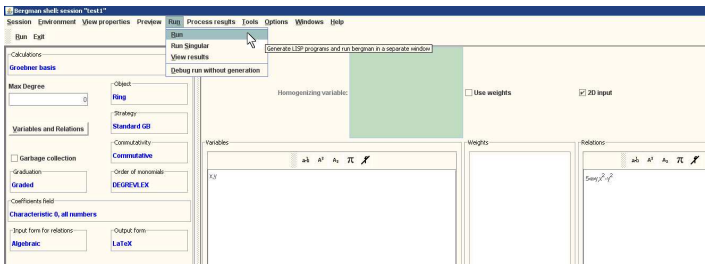


Figure 1. A graphical shell that supports Bergman and Singular

Both these engines support only the console interface. For Bergman, it is the underlying Lisp console. For Singular, it is a console with the Singular programming language. Our shell permits to enter mathematical objects, converts them to the Singular or Bergman input files, and runs the corresponding engine to execute calculations.

[1] http://www.math.su.se/bergman/
[2] http://www.singular.uni-kl.de/

The CAS shell design can have two different starting points: the set of engine operations, or the set of processed mathematical objects. In [5, Sec. 3.3] these approaches are called correspondingly "noun-verb" and "verb-noun" (or "object-action" and "action-object"). V. López-Jaquero and F. Montero [4] refer to these variants as to "domain model" and "task model".

There is no common opinion on applicability of these two approaches. J. Raskin [5] shows several advantages of the first approach over the second one. V. López-Jaquero and F. Montero [4] motivate the advantage of the second approach, but their argumentation applies mainly to the case when the objects are containing in the databases. They note also that "the derivation of the user interface out of a task model adds an additional view to the design process: the user".

In the beginning we built shells for the Bergman CAS originating them from the engine operations ("the task model"). This seemed naturally for Bergman that has only a small set of objects, but a big and growing set of actions. Later we planned to use several CAS engines with the same shell. We wanted to support the usage of an engine that would not be even taken into account at the shell programming, i.e., to make the shell as independent of the engines as possible. The corresponding investigation permitted us to construct the calculator model of the shell. In this case, the shell actions are restricted by the object input, the result output, the object conversions between different representations, the selection of the engine and its operation, and several common tasks like the session support. We see that most tasks are defined by objects and that user tasks are quite restricted. In the meantime [1] the old "verb-noun" model began to hinder the shell development.

Therefore we decided to originate the CAS shell from the mathematical objects we want to proceed ("the domain model"). This approach permitted us to create a shell that is really independent of the used engines, and to provide researchers-mathematicians with a unified shell for several CASs.

Within the domain model, the CAS shell development begins with the listing of used mathematical objects. For each object, we provide

a procedure or procedures of its input that produces its internal XML representation. The entered object is dispalyed on a tab. The current set of tabs with objects, and the current parameter settings can be stored as a *session* and restored later.

The object is displayed on a tab in one of its external representations (e.g., LaTeX). A set of convertors from internal XML representation to different external representations exists for each object. The user selects a representation of an object through a menu. There is also a possibility to store an object in any of its representations in a file.

Except of input procedures and convertors to external representations, each object is associated with actions that can be performed over it by the existing engines. As the user opens object's tab, the associated actions became visible in a menu.

Some actions use more than one object. These additional objects should be entered and visible on other tabs. If there are several combinations of objects for one or several actions, the corresponding request is made to select one of these variants.

So the user selects object(s) and the action. The shell converts objects from their internal XML representation to the engine input files and starts the necessary engine.

After the engine run termination the calculated result is kept usually in a file. After the calculation is finished a shell module converts the result in its internal XML representation and shows it on a new tab. Being a mathematical object, this result can be converted in different external (visual) representations, saved in a file, and used for further calculations.

We see that a calculator CAS shell model supposes the object-independent part (session support, tab manipulations, dynamic menus support, etc.) and the object-dependent part. The object-dependent part contains input modules and convertors. Most convertors are engine-independent but the convertors used to generate input files before the engine start are engine-dependent. There are also engine-dependent convertors that scan output files after the engine finishes its calculations and produce internal XML representation of resulting objects.

Action menus are both object- and engine-dependent. To change these menus dynamically, the shell uses XML descriptions that exist for each type of object, each engine, and each allowed combination of those.

3 Static shell assembly

Our technique of the static shell assembly is described in details in [3]. We combined in it the component and aspect-oriented programming.

Having a set of ready-made modules described above we need three operations to assembly a shell:

1. to generate the *glue code* (the additional code that is necessary to assemble components together);
2. to generate code that tunes adaptable components;
3. to generate variable menus.

Aspect oriented programming (AOP) is a technique to add a new behavior to an existing program without changing its sources and even binaries. It is mostly used to handle cross-cutting concerns like logging or debugging. E.g., we need to add almost the same code in regularly selected places of the program to trace it. AOP concentrates templates of additional code and insertion points in *aspects*. Aspects are compiled separately, and the *code weaving* is performed during the execution of the program.

To apply AOP for the semi-automated assembly of a shell from components, we noted that the glue code is regular and repeating, and that it can be generated from a formal description of the shell. With AOP, we use an unchanged shell template and unchanged components, and generate only aspects containing the glue code or the code to tune adaptable modules. The menu is also generated as an aspect. We have checked this idea by implementing it.

A shell consists of the constant part and the variable part. The constant part contains, in particular, the session management: storing data for each session, their modification, etc. We also found useful a

notion of *environment*, or partially defined session [1]. Each session can be based on an environment where some data are already defined. The environment management is implemented like the session management.

Other features of the constant part of a shell are possibilities to create the list of engines, to start external programs, to check collected data, to show help, etc.

Modules that enter the data and convertors form the variable part of a shell.

During the assembly of a shell its constant part is taken as the base. The developer prepares list of objects and defines how they have to be entered in the shell (by selection from several variants, by marking, by text editing, by 2D formula input, by entering parameters of a mathematical object using a wizard, etc.) Each possible method of the data input is implemented as a customizable component. The necessary modules pass the customization and are glued together with the constant part of the shell. Menus are also generated and included as an aspect.

The whole system consists therefore of a pre-implemented constant part, a set of data input components and convertors, and a shell generator that adapts and assembles all parts together producing CAS shells.

4 Dynamic shell assembly

The shell with dynamic module assembly is based on the open source Eclipse[3] platform.

An Eclipse-based application consists of the Eclipse *platform* and a set of *plugins*. Each plugin is a module that contains in itself its XML description as a resource. The system tunes itself (e.g., adds new menu items) using the XML description of the new module. To add a module, it is enough to copy its JAR archive in the plugin directory and to restart.

The visual part of the Eclipse platform is the Eclipse *workbench*.

[3] http://www.eclipse.org/

The workbench provides a window that contains tabbed *views* (e.g., lists of settings) and *editors*. Such window is called in Eclipse the *perspective* (corresponds to *session* as we defined before). Editors are plugins that edit texts; a usual text editor is already provided with the Eclipse platform. The workbench supports also *projects*; for a project, we can store and restore its current perspective.

The construction and work of the CAS shell based on Eclipse remains the same as described before.

The Eclipse platform will form the constant part of the shell.

Modules to input mathematical object should be implemented as plugins-editors. Convertors should be also implemented as plugins. Eclipse supports the dynamic change of menus at the activation of each editor.

A separate plugins are necessary to conduct engines. It includes engine list support, engine start, consoles, etc., and, especially, the support of correspondence between the engine functions and mathematical objects. This last feature is new for Eclipse.

5 Conclusions

The first approach permits to implement a platform-independent system. We work in Java with Swing graphics. The deployed shell consists of a single executable JAR that contains the compiled classes, resources, and additional libraries. This archive can be executed on any platform with the suitable version of the Java VM. At the second approach we use SWT graphic library from Eclipse that is platform dependent. We are in this case to deploy different archives for different platforms, or to require the user to install Eclipse or, at least, its libraries.

However the Swing graphics was criticized for its visual appearance that does not correspond the platform standards. Eclipse uses the native graphics on each platform that can slightly accelerate the graphical operations and guarantees the native appearance.

At the first approach we implement session support in the exact necessary volume. With Eclipse, we are to use the Eclipse framework

that is more general and may seem more complicated: some base features of Eclipse are superfluous for us.

Any shell expansion (adding a new mathematical object, new engine or new action of the existing engine, etc.) implies recompilation to add new features at the static assembly. Eclipse adds new plugins dynamically.

The factors listed till now balance one another; none of them is decisive. The main advantage of Eclipse is its richness, especially in the current Eclipse 3 "Europa". This version of Eclipse contains more than 900 ready-made plugins. A big part of common GUI functions is already implemented or can be adapted from existing plugins. After the appearance of Eclipse 3 we decided to use this approach in our project. However the general system structure and functions are common for both approaches.

6 Acknowledgements

The work was supported by the INTAS grant Ref. Nr. 05–104–7553 "Interface generating toolkit for symbolic computation systems".

References

[1] S. Cojocaru, L. Malahova, and A. Colesnicov. Interfaces to symbolic computation systems: Reconsidering experience of bergman. *Computer Science Journal of Moldova*, 13(2(28)):232–244, 2005.

[2] S. Cojocaru, L. Malahova, and A. Colesnicov. Providing modern software environments to computer algebra systems. In V.G. Ganzha, E.W. Mayr, and E.V. Vorozhtsov, editors, *Computer Algebra in Scientific Computing. 9th International Workshop, CASC 2006. Chişinău, Moldova, September 11–15, 2006. Proceedings*, number 4194, pages 129–140. Springer-Verlag, 2006.

[3] A. Colesnicov and L. Malahova. Aspect oritented programming and component assembly. *Computer Science Journal of Moldova*, 15(1(43)):38–53, 2007. ISSN 1561–4042.

[4] V. Lopez-Jaquero and F. Montero. Comprehensive task and dialog modelling. In J. Jacko, editor, *Human Computer Interaction, Part I. HCII 2007. Beijing, China, July 22–27, 2007.*, number 4550, pages 1149–1158. Springer-Verlag. ISSN 0302–9743.

[5] J. Raskin. *The Humane Interface. New Direction for Designing Interactive Systems*. Pearson Education, Inc. (Addison Wesley Longman), 2000. ISBN 0–201–37937–6.

A. Colesnicov, S. Cojocaru, L. Malahova,

Received February 19, 2008

Institute of Mathematics and Computer Science,
5 Academiei str.
Chişinău, MD−2028, Moldova.
E–mail: *kae@math.md*, *sveta@math.md*, *mal@math.md*

Congratulations

The Society "Academician Constantin Sibirschi", which activates in Republic of Moldova under the chairmanship of the USA business man, Dr. Val Sibirschi, organizes annually the contest of most valuable works in the field of mathematics. It was the corresponding member of the Academy of Sciences of Moldova, Professor C.Gaindric who became the 2007 laureate. He was awarded by the Premium of "Academician Constantin Sibirschi" for the series of works "Mathematical models and information systems of decisions underlying in information society".

The series consists from 42 works (2 monographs, 3 chapters in collective monographs, 13 articles in journals, 21 articles in collections, the 10 of which are abroad, 4 overviews and brochures) summarizing the following results:

- The solutions of the problems which can be formulated as general-

ization of the problem of m-traveling salesmen are proposed. The algorithms of solution on the base of the scheme *branch and bound* and *tabu search* are proposed. An information heuristic system for calculation of transportation routes for different goods within radius of Moscow and Baku was elaborated and experimentally applied in Moldova for spare parts transportation in association Moldselhoztechnica.

- A structure of Decision Support Systems (DSS) was proposed, which permits a unified approach irrespective of the problem domain and its nature, assures their functionality and convenience in decision making process.

- Under Prof. C.Gaindric leadership and at his direct participation there were elaborated:

  - DSS for dispatcher of vehicle transport enterprise;
  - DSS for activity plan formation for transport enterprise;
  - DSS for financing and monitoring of the projects of a scientific and technical program.

    In these DSS the methods and algorithms for solution of the problems described in the works of the series are used.
  - At present the system SonaRes for ultrasonographic diagnostics is being elaborated.

In the problems of Information Society (IS) creation the analysis of the process of IS development was made, corresponding indices were revealed and analyzed, concentrating henceforth on the problem of digital divide overcoming. The solutions were proposed which can serve as a draft for a future DSS, which will monitor indices of IS development, highlight strong and weak aspects, offer concrete solutions for creation of public access points in some community, starting from existing unbiassed situation. In this context:

The researches were made on the role which science, electronic culture, electronic education, electronic governance play in information

society; system of indices of preparation for integration into information society *(e-readiness)* and those which monitor the process of digital divide overcoming (*digital divide*) were investigated also.

The problem of population providing for the nondiscriminatory access to information was examined. Its solution was argued by public Internet access points formation at schools.

Editorial board of the journal ,,Computer Science Journal of Moldova" congratulates heartily its colleague, Prof. Constantin Gaindric, on getting this high award.

Editorial board of CSJMol