The Big Mother of All the Dualities: Möller Algorithm

Maria Emilia Alonso
Dept. Algebra, Fac. Cie. Math.
Universitat Complutense Madrid
alonsog@eucmax.sim.ucm.es

Teo Mora
DIMA and DISI, Università di Genova
Dept. Math. Florida State University
theomora@dima.unige.it

February 12, 1999

1 The Stage

Let K be an effective field of characteristic zero; let V be a K-vector space and $V^* = \mathrm{Hom}_K(V, K)$.

Remark 1.1 The following are equivalent:

1. there exist n linear functionals $\mathfrak{X}_1, \ldots, \mathfrak{X}_n \in \mathrm{Hom}_K(V, V)$;
2. V is a left $K\langle X_1, \ldots, X_n\rangle$-module;
3. V^* is a right $K\langle X_1, \ldots, X_n\rangle$-module.

The relation is the obvious one, denoting

$$X_i v := \mathfrak{X}_i(v) \in V, \forall v \in V,$$

and

$$wX_i := \mathfrak{X}_i(w) \in V^*, \forall w \in V^*,$$

we have

$$(wX_i)(v) = w(X_i v), \forall v \in V, w \in V^*.$$

Of course, the following are also equivalent:

- $\mathfrak{X}_i(\mathfrak{X}_j(v)) = \mathfrak{X}_j(\mathfrak{X}_i(v)), \forall v \in V, \forall i, j;$

- V is a $K[X_1, \ldots, X_n]$-module;

- V^* is a $K[X_1, \ldots, X_n]$-module. $\square$

Denoting $\mathcal{P} = K\langle X_1, \ldots, X_n\rangle$, let us assume that V is given as a finite left $\mathcal{P}$-module, by assigning linear functionals $\mathfrak{X}_i \in \mathrm{Hom}_K(V, V)$ and a finite basis $\mathbf{1} := \{e_1, \ldots, e_\nu\} \in V$.

This induces the canonical $\mathcal{P}$-homomorphism

$$\Phi : \mathcal{P}^\nu \mapsto V, \Phi(f_1, \ldots, f_\nu) = \sum_i f_i e_i$$

–if needed, by abuse of notation we will denote by $\mathbf{1}$ also the basis of $\mathcal{P}^\nu$ – and allows to apply the Gröbner technology to V.

Let us denote $\mathbb{T} := \langle X_1, \ldots, X_n\rangle$ and $\mathbb{T}^\nu := \{\tau e_i : \tau \in \mathbb{T}, e_i \in \mathbf{1}\}$ which is a K-basis of $\mathcal{P}^\nu$ and let us impose on it an order $<$ s.t.

$$\forall \tau_1, \tau_2 \in \mathbb{T}^\nu, \forall i, 1 \leq i \leq n, \tau_1 < \tau_2 \Longrightarrow X_i\tau_1 < X_i\tau_2;$$

then each element $f \in \mathcal{P}^\nu$ can be uniquely expressed as

$$f = \sum c_i\tau_i, c_i \in K \setminus \{0\}, \tau_i \in \mathbb{T}^\nu, \tau_1 > \tau_2 > \cdots$$

and we can define its *maximal term* $T(f) := \tau_1$; in the same way for each set $F \in \mathcal{P}^\nu$, we will denote $T(F) := \{\tau T(f) : \tau \in \mathbb{T}, f \in F\}$.

Setting $U := \ker(\Phi)$ and $N(U) := \mathbb{T}^\nu \setminus T(U)$ we have

$$\mathcal{P}^\nu \cong U \oplus V, V \cong \mathcal{P}^\nu/U \cong \mathrm{Span}_K(N(U)).$$

Thanks to these congruences, we can consider $\mathbf{T} := N(U)$ as a K-basis of V so that the (linear algebra) Gröbner technology discussed in [13] and [11] can be projected from $\mathcal{P}^\nu$ to V.

In particular, $\mathbf{T}$ can be ordered (as a set only!) by $<$ and each element $f \in V$ can be uniquely expressed as

$$f = \sum c_i\tau_i, c_i \in K \setminus \{0\}, \tau_i \in \mathbf{T}, \tau_1 > \tau_2 > \cdots$$

and we can denote $T(f) := \tau_1$ its maximal term.

Lemma 1.2 $\forall f \in V, \forall j, X_jT(f) \in \mathbf{T} \Longrightarrow T(\mathfrak{X}_j(f)) = X_jT(f)$.

Proof: Let $f := \sum_{i=1}^t c_i\tau_i, c_i \in K \setminus \{0\}, \tau_i \in \mathbf{T}$ with $T(f) = \tau_1$. Let us partition $\{1, \cdots, t\} = I_N \uplus I_T$ where

$$I_N := \{i : X_j\tau_i \in N(U)\} \text{ and } I_T := \{i : X_j\tau_i \in T(U)\}.$$

Then

$$\begin{aligned}\mathfrak{X}_j(f) &= \sum_{i=1}^{t} c_i \mathfrak{X}_j(\tau_i) = \sum_{i\in I_N} c_i \mathfrak{X}_j(\tau_i) + \sum_{i\in I_T} c_i \mathfrak{X}_j(\tau_i) = \\ &= \sum_{i\in I_N} c_i X_j \tau_i + \sum_{i\in I_T} c_i \Phi(X_j\tau_i) = \sum_{i\in I_N} c_i X_j\tau_i + \sum_{i\in I_T}\sum_{\tau\in N(U)} c_i c_{i\tau}\tau,\end{aligned}$$

where $\Phi(X_j\tau_i) = \sum_{\tau\in N(U)} c_{i\tau}\tau$ and $X_j\tau - \sum_{\tau\in N(U)} c_{i\tau}\tau \in U$; the resut holds since

- $X_j\tau_1 = X_j T(f) \in N(U)$ by assumption,
- $X_j\tau_1 > X_j\tau_i, \forall i \in I_N$,
- $X_j\tau_1 > X_j\tau_i > \tau, \forall i \in I_T, \forall \tau : c_{i\tau} \neq 0.$ $\square$

For each K-vector subspace $P \subset V$, let us denote:

– $\mathbf{T}(P) := \{T(f) : f \in P\}$;

– $\mathbf{N}(P) := \mathbf{T} \setminus \mathbf{T}(P)$;

– $\mathbf{B}(P) := \mathbf{T}(P) \cap (\mathbf{1} \cup \{X_j\tau : 1 \le j \le n, \tau \in \mathbf{N}(P)\})$.

Lemma 1.3 *For each K-vector subspace $P \subset V$, it holds:*

1. $V = P \oplus \mathrm{Span}_K(\mathbf{N}(P))$;
2. *there is a K-vector space isomorphism between V/P and* $\mathrm{Span}_K(\mathbf{N}(P))$;
3. *for each $f \in V$ there is a unique $g := Can(f,P) \in \mathrm{Span}_K(\mathbf{N}(P))$ s.t. $f - g \in P$.*

Definition 1.4 *If $P \subset V$ is a finite dimensional K-vector subspace, the* border-basis *of P (shortly: the B-basis) w.r.t. $<$ is the subset*

$$\mathcal{B}(P) := \{\tau - Can(\tau, P) : \tau \in \mathbf{B}(P)\}.$$

Lemma 1.5 *Let $P \subset V$ be a finite dimensional left $\mathcal{P}$-submodule. For each $\tau \in \mathbf{T}$ it holds*

$$\tau \in \mathbf{T}(P) \iff \exists \tau_l \in \mathbb{T}, \tau_r \in \mathbf{B}(P) : \tau = \tau_l\tau_r.$$

Proof: Let $\tau := Y_m Y_{m-1}\cdots Y_1 e_j$, with $Y_i \in \{X_1,\dots,X_n\}$, $e_j \in \mathbf{1}$, and let us denote $\tau_0 := e_j, \tau_i := Y_i\tau_{i-1}, \forall i > 0$. Either $\tau \in \mathbf{N}(P)$ or there is a least i s.t. $\tau_i \in \mathbf{T}(P)$. If $i = 0$ the result holds with

$$\tau_l := Y_m Y_{m-1}\cdots Y_1, \tau_r := \tau_0 \in \mathbf{T}(P) \cap \mathbf{1} \subset \mathbf{B}(P).$$

Otherwise $\tau_{i-1} \in \mathbf{N}(P)$ and the result holds with

$$\tau_l := Y_m \cdots Y_{i+1}, \tau_r := \tau_i = Y_i\tau_{i-1} \in \mathbf{B}(P).$$

Conversely, if $\tau = \tau_l\tau_r$ with $\tau_l \in \mathbb{T}, \tau_r \in \mathbf{B}(P)$, let $f := \tau_l\,(\tau_r - Can(\tau_r, P)) \in P$. Then $\tau_l\tau_r = T(f) \in \mathbf{T}(P)$. $\square$

2 The Mother of Duality

For each K-vector subspace $P \subset V$, let

$$\mathfrak{L}(P) := \{\ell \in V^* : \ell(f) = 0, \forall f \in P\};$$

for each K-vector subspace $L \subset V^*$, let

$$\mathfrak{P}(L) := \{f \in V : \ell(f) = 0, \forall \ell \in L\}.$$

Lemma 2.1 *$P \subset V$ is a left $\mathcal{P}$-submodule if and only if $\mathfrak{L}(P)$ is a right $\mathcal{P}$-submodule.*

$L \subset V^*$ is a right $\mathcal{P}$-submodule if and only if $\mathfrak{P}(L)$ is a left $\mathcal{P}$-submodule.

Proof:

$$\begin{array}{lcl} P \text{ is a left } \mathcal{P}\text{-submodule} & \iff & \forall g \in P, \forall j, X_j g \in P \\ & \iff & \forall g \in P, \forall j, \forall \ell \in \mathfrak{L}(P), \\ & & \quad (\ell X_j)(g) = \ell(X_j g) = 0 \\ & \iff & \forall \ell \in \mathfrak{L}(P), \forall j, \ell X_j \in \mathfrak{L}(P) \\ & \iff & \mathfrak{L}(P) \text{ is a right } \mathcal{P}\text{-submodule.} \quad \square \end{array}$$

Lemma 2.2 *Let $P \subset V$ be a K-vector subspace and $g \in V$; then*

$$\ell(g) = 0, \forall \ell \in \mathfrak{L}(P) \Longrightarrow g \in P.$$

Let $L \subset V^*$ be a K-vector subspace and $\ell \in V^*$;then

$$\ell(g) = 0, \forall g \in \mathfrak{P}(L) \Longrightarrow \ell \in L.$$

Proof: Let B be a K-basis of P; assume $g \notin P$ and complete $B \cup \{g\}$ to a K-basis B' of V. Clearly, any $\ell \in V^*$ is uniquely characterized by giving $\ell(f), \forall f \in B'$.

So let ℓ be s.t. $\ell(g) \neq 0$, $\ell(f) = 0, \forall f \in B' - \{g\}$. Then $\ell \in \mathfrak{L}(P)$ and $\ell(g) \neq 0$.

The second statement comes by dualizing the proof of the first statement. $\square$

Theorem 2.3 *$\mathfrak{L}$ and $\mathfrak{P}$ define a duality between left $\mathcal{P}$-submodules of V and right $\mathcal{P}$-submodules of V^*. Under this duality, finite dimensional submodules correspond to finite dimensional ones.*

Proof: Let $P \subset V$ and let $L = \mathfrak{L}(P) \subset V^*$. P is a left $\mathcal{P}$-submodule of V iff $\mathfrak{L}(P)$ is a right $\mathcal{P}$-submodule of V^* by Lemma 2.1. Moreover $\mathfrak{P}(\mathfrak{L}(P)) = P$ and $\mathfrak{L}(\mathfrak{P}(L)) = L$ by Lemma 2.2 $\square$

Definition 2.4 *Let $\mathbb{L} = \{\ell_1, \ldots, \ell_s\} \subset V^*$ and $\mathbf{q} = \{q_1, \ldots, q_s\} \subset V$.*

$\mathbb{L}$ and $\mathbf{q}$ are said to be

- triangular *if* $L_i(q_j) = 0$ *if* $i < j$*;*
- biorthogonal *if* $L_i(q_j) = \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j. \end{cases}$

Lemma 2.5

1. *Given* $\mathbb{L} = \{\ell_1, \ldots, \ell_s\} \subset V^*$*, it is equivalent:*
   (a) there exists $\mathbf{q} = \{q_1, \ldots, q_s\} \subset V$ *biorthogonal to* $\mathbb{L}$*;*
   (b) there exists $\mathbf{q} = \{q_1, \ldots, q_s\} \subset V$ *triangular to* $\mathbb{L}$*;*
   (c) $\mathbb{L}$ *is linearly independent.*
2. *Given* $\mathbf{q} = \{q_1, \ldots, q_s\} \subset V$*, it is equivalent:*
   (a) there exists $\mathbb{L} = \{\ell_1, \ldots, \ell_s\} \subset V^*$ *biorthogonal to* $\mathbf{q}$*;*
   (b) there exists $\mathbb{L} = \{\ell_1, \ldots, \ell_s\} \subset V^*$ *triangular to* $\mathbf{q}$*;*
   (c) $\mathbf{q}$ *is linearly independent.*

Proof: Assume $\mathbb{L}$ is not linearly independent, so that there are $c_1, \ldots, c_s \in K$ not all zero, say $c_s \neq 0$, s.t. $\sum c_i \ell_i = 0$ and assume that $\mathbf{q} = \{q_1, \ldots, q_s\} \subset V$ is biorthogonal to $\mathbb{L}$. Then

$$0 = \sum c_i \ell_i(q_s) = c_s \ell_s(q_s) = c_s \neq 0,$$

giving a contradiction.

Assume now that $\mathbb{L}$ is linearly independent, and let us prove the existence of $\mathbf{q} = \{q_1, \ldots, q_s\} \subset V$ biorthogonal to $\mathbb{L}$, arguing by induction on s.

If $s = 1$, the linearly independence of $\{\ell_1\}$ means $\ell_1 \neq 0$ and so the existence of $g \in V$ s.t. $\ell_1(g) \neq 0$.

So let us assume the existence of $\{q_1, \ldots, q_{s-1}\} \subset V$ which is biorthogonal to $\{\ell_1, \ldots, \ell_{s-1}\}$. Since $\ell_s \notin \mathrm{Span}_K(\{\ell_1, \ldots, \ell_{s-1}\})$ then

$$\ell := \ell_s - \ell_s(q_1)\ell_1 - \cdots - \ell_s(q_{s-1})\ell_{s-1} \neq 0$$

and there is $g \in V : \ell(g) \neq 0$. Setting

$$g' := g - \ell_1(g)q_1 - \cdots - \ell_{s-1}(g)q_{s-1}$$

it holds

$$\ell_s(g') = \ell_s(g) - \ell_1(g)\ell_s(q_1) - \cdots - \ell_{s-1}(g)\ell_s(q_{s-1}) = \ell(g) \neq 0,$$

while for $i < s$

$$\ell_i(g') = \ell_i(g) - \ell_1(g)\ell_i(q_1) - \cdots - \ell_{s-1}(g)\ell_i(q_{s-1}) = \ell_i(g) - \ell_i(g)\ell_i(q_i) = 0,$$

so that $\mathbf{q} := \{q_1, \ldots, q_{s-1}, g'\} \subset V$ is triangular to $\mathbb{L}$.

In order to make it biorthogonal we have just to define $q'_s := \ell_s(g')^{-1} g'$ and, $\forall j := (s-1)..1$, $q'_j = q_j - \sum_{i>j} \ell_i(q_j) q'_i$. □

Let $\mathbb{L} = \{L_1, \ldots, L_s\} \subset V^*$ and $\mathbf{q} = \{q_1, \ldots, q_s\} \subset V$ and denote $L := \mathrm{Span}_K(\mathbb{L})$, $Q := \mathrm{Span}_K(\mathbf{q})$ and $\forall f \in V$

- $\rho(f, \mathbf{q}) := (c_1, \ldots, c_s) : f = \sum_i c_i q_i \bmod V/Q$;
- $v(f, \mathbb{L}) := (L_1(f), \ldots, L_s(f))$.

Lemma 2.6 *With the notation above, if $\mathbb{L}$ and $\mathbf{q}$ are biorthogonal, then:*

1. *L is a right $\mathcal{P}$-submodule iff $V/\mathfrak{P}(L) \cong Q$;*
2. *Q is a left $\mathcal{P}$-submodule iff $V^*/\mathfrak{L}(Q) \cong L$.*

Proof: Let $q \in V$ and let $q^{(1)} := \sum_i L_i(q) q_i$, $q^{(2)} := q - q^{(1)}$ so that $q^{(1)} \in Q$, $L_j(q^{(2)}) = 0\ \forall j$, $q^{(2)} \in \mathfrak{P}(L)$ and $V = Q \oplus \mathfrak{P}(L)$. □

Corollary 2.7 *With the same notation and assumptions of Lemma 2.6, it holds:*

1. *the vectors $v(q_i, \mathbb{L})$ are linearly independent;*
2. *the map $v : V \mapsto K^s$ satisfies $\ker(v) = \mathfrak{P}(L)$ so that*
3. *its restriction $v : Q \mapsto K^s$ is an isomorphism, and*
4. *$v(q) = \sum_i c_i v(q_i) \iff q = \sum_i c_i q_i$.*

Remark 2.8 Let $Q, W \subset V$ be s.t. $V = Q \oplus W$. For each $\ell \in V^*$ let $\bar{\ell}$ denote its restriction to W. Then $W^* = \{\bar{\ell}, \ell \in \mathfrak{L}(Q)\}$. In fact, each $L \in W^*$ can be extended to a linear functional $\ell \in V^*$ s.t. $L = \bar{\ell}$ by simply defining $\ell(q + w) = L(w), \forall q \in Q, \forall w \in W$. □

3 The Scion of the Algorithms

Möller Algorithm, that we are presenting now, solves the following

Problem 3.1 Given a finite sequence $\mathbb{L} = \{\ell_1, \ldots, \ell_s\} \subset V^*$ generating (as a K-vector space) a right $\mathcal{P}$-submodule $L \subset V^*$, compute:

– the finite dimensional left $\mathcal{P}$-submodule $P := \mathfrak{P}(L)$ by giving $\mathbf{N}(P)$ and the B-basis $\mathcal{B}(P)$ w.r.t. an ordering $<$;

– a sequence $\mathbb{L}' = \{\ell'_1, \ldots, \ell'_r\} \subset \mathbb{L}$ which is a linear basis of L;

– the sequence $\mathbf{q}$ which is triangular to $\mathbb{L}'$;

– $\forall i = 1..n, \forall j = 1..r, \rho(X_i q_j, \mathbf{q})$.

Möller Algorithm was originally introduced in 1982 in [3] in the case in which $V := K[X_1, \ldots, X_n]$ and the linear functionals ℓ were point evaluations.

It was reproposed in [5] (cf. also [6]), with important improvements and a complexity analysis, as a tool to solve the FGLM-problem: here again V is the polynomial ring and the functionals are the coefficients in a canonical form of a polynomial by a Gröbner basis.

[11] presents the complete theory in the polynomial ring case.

Möller Algorithm has been recently generalized in the non-commutative polynomial ring setting ([14]) as a tool to interpret Todd-Coxeter Algorithm via Gröbner technology, and to the two-sided ideal case ([2]).

However, hints to Möller Algorithm in non-commutative polynomial rings are alrealdy present in [9] which is even older than [6].

Möller Algorithm can be informally described as follows:
three lists N, H and $List$ are considered. The first one contains terms in $\mathbf{N}(P)$, the second one contains elements in the border basis of P with respect to $<$. The third one is an auxiliary tool. Every step treats the minimal term $t := T(f) \in \mathbf{T}$ which is neither in N nor in $T(H)$. The three lists are updated until $List = \emptyset$, which implies that every term in $\mathbf{T}$ is either in N or in $T(H)$.

Algorithm 3.2 (Möller)

$N := \emptyset$; $H := \emptyset$; $List := \mathbf{1}$; $r := 0$
While $List \neq \emptyset$ **do**
  Choose $f \in List \mid T(f) = \min_<(T(g) : g \in List)$
  $List := List \setminus \{f\}$
  $t := T(f)$
  If $t \notin N \cup T(H)$ **then**
    $(p, \rho(f, \mathbf{q})) :=$ **Gauss-reduce**$(f; q_1, \ldots, q_r)$
    %% $L_{\pi(i)}(p) = 0, \forall i \leq r$
    If $v(p, \mathbb{L}) = 0$ **then**
      $H := H \cup \{p\}$
    else
      $r := r + 1$
      $\pi(r) := \min\{i : L_i(p) \neq 0\}$
      $q_r := L_{\pi(r)}(p)^{-1} \cdot p$
      $\rho(f, \mathbf{q}) := \rho(f, \mathbf{q}) - L_{\pi(r)}(p) e_r$
      $N := N \cup \{t\}$
      $List := List \cup \{X_j q_r : j = 1..n\}$
  else
    $(p, \rho(f, \mathbf{q})) :=$ **Gauss-reduce**$(f; q_1, \ldots, q_r)$
$G := H$

where $\{e_1, \ldots, e_s\}$ is the canonical basis of H^s and **Gauss-reduce** is

$(p, \rho(f, \mathbf{q})) :=$ **Gauss-reduce**$(f; q_1, \ldots, q_r)$
$p := f$, $\rho(f, \mathbf{q}) := 0$
For $i = 1..r$ **do**
$p := p - (L_{\pi(i)}p)q_i$
$\rho(f, \mathbf{q}) := \rho(f, \mathbf{q}) + (L_{\pi(i)}p)e_i$

In other words, $(p, \rho(f, \mathbf{q})) :=$ **Gauss-reduce**$(f; q_1, \ldots, q_r)$ performs Gaussian reduction on $v(f, \mathbb{L})$ with respect to the linearly independent set $\{v(q_i, \mathbb{L}), i = 1..r\}$ and performs the same linear operation on f using $q_1, \ldots, q_r$.

Let us prove the correctness of the algorithm. The terms $T(f)$ corresponding to the polynomials f which are inserted in $List$ are exactly the terms in the set

$$\mathbf{1} \cup \{X_j t \mid t \in \mathbf{N}(P),\ j \in \{1, \ldots n\}\} = \mathbf{N}(P) \uplus \mathbf{B}(P).$$

Each term t is inserted in

– $T(H) \subseteq \mathbf{B}(P)$, if it is linearly dipendent mod. P with the elements of N;

– N, otherwise, so that constantly $N \subseteq \mathbf{N}(P)$.

Therefore $\mathbf{N}(P) \uplus \mathbf{B}(P) = N \uplus T(H)$, $N \subseteq \mathbf{N}(P)$, $T(H) \subseteq \mathbf{B}(P)$, which implies $N = \mathbf{N}(P)$, $T(H) = \mathbf{B}(P)$ and the correctness of the algorithm.

Example 3.3 We will present thruout this note an elementary example in which $V := K[X, Y]$, $\mathbb{L} = \{\ell_1, \ell_2, \ell_3, \ell_4, \ell_5\}$ where $\forall f \in V$,

$$\begin{array}{lllllllll} \ell_1(f) & = & f(0,0), & \ell_2(f) & = & f(1,0), & \ell_3(f) & = & f(0,1), \\ \ell_4(f) & = & f(1,-1), & \ell_5(f) & = & f(0,-1). & & & \end{array}$$

Denoting f_i the polynomial treated in the i^{th} **While**-loop, and $\rho_i := \rho(f_i, \mathbf{q})$, $v_i := v(f_i, \mathbb{L})$, $V_i := v(q_i, \mathbb{L})$, the computation is:

$$
\begin{array}{rcl rcl rcl}
N & = & \emptyset & List & = & \{1\} & H & = & \emptyset \\
i & = & 1 & & & & & & \\
f_1 & = & 1 & \rho_1 & = & (1,0,0,0,0) & & & \\
V_1 & = & (1,1,1,1,1) & v_1 & = & (1,1,1,1,1) & q_1 & = & 1 \\
N & = & \{1\} & List & = & \{X,Y\} & & & \\
i & = & 2 & & & & & & \\
f_2 & = & X & \rho_2 & = & (0,1,0,0,0) & & & \\
V_2 & = & (0,1,0,1,0) & v_2 & = & (0,1,0,1,0) & q_2 & = & X \\
N & = & \{1,X\} & List & = & \{Y, Xq_2, Yq_2\} & & & \\
i & = & 3 & & & & & & \\
f_3 & = & Y & \rho_3 & = & (0,0,1,0,0) & & & \\
V_3 & = & (0,0,1,-1,-1) & v_3 & = & (0,0,1,-1,-1) & q_3 & = & Y \\
N & = & \{1,X,Y\} & List & = & \{Xq_2, Yq_2, Xq_3, Yq_3\} & & & \\
i & = & 4 & & & & & & \\
f_4 & = & X^2 & \rho_4 & = & (0,1,0,0,0) & & & \\
 & & & v_4 & = & (0,1,0,1,0) & & & \\
N & = & \{1,X,Y\} & List & = & \{Yq_2, Xq_3, Yq_3\} & H & = & H \cup \{X^2 - X\} \\
i & = & 5 & & & & & & \\
f_5 & = & XY & \rho_5 & = & (0,0,0,-1,0) & & & \\
V_4 & = & (0,0,0,1,0) & v_5 & = & (0,0,0,-1,0) & q_4 & = & -XY \\
N & = & \{1,X,Y,XY\} & List & = & \{Xq_3, Yq_3, Xq_4, Yq_4\} & & & \\
i & = & 6 & & & & & & \\
f_6 & = & XY & \rho_6 & = & (0,0,0,-1,0) & & & \\
 & & & v_6 & = & (0,0,0,-1,0) & & & \\
i & = & 7 & & & & & & \\
f_7 & = & Y^2 & \rho_7 & = & (0,0,1,2,2) & & & \\
V_5 & = & (0,0,0,0,1) & v_7 & = & (0,0,1,1,1) & q_5 & = & \frac{1}{2}Y^2 + XY - \frac{1}{2}Y \\
N & = & \{1,X,Y,XY,Y^2\} & List & = & \{Xq_4, Yq_4, Xq_5, Yq_5\} & & & \\
i & = & 8 & & & & & & \\
f_8 & = & -X^2Y & \rho_8 & = & (0,0,0,1,0) & & & \\
 & & & v_8 & = & (0,0,0,1,0) & & & \\
N & = & \{1,X,Y,XY,Y^2\} & List & = & \{Yq_4, Xq_5, Yq_5\} & H & = & H \cup \{-X^2Y + XY\} \\
i & = & 9 & & & & & & \\
f_9 & = & -XY^2 & \rho_9 & = & (0,0,0,-1,0) & & & \\
 & & & v_9 & = & (0,0,0,-1,0) & & & \\
N & = & \{1,X,Y,XY,Y^2\} & List & = & \{Xq_5, Yq_5\} & H & = & H \cup \{-XY^2 - XY\} \\
i & = & 10 & & & & & & \\
f_{10} & = & \frac{1}{2}XY^2 + X^2Y - \frac{1}{2}XY & \rho_{10} & = & (0,0,0,0,0) & & & \\
 & & & v_{10} & = & (0,0,0,0,0) & & & \\
i & = & 11 & & & & & & \\
f_{11} & = & \frac{1}{2}Y^3 + XY^2 - \frac{1}{2}Y^2 & \rho_{11} & = & (0,0,0,0,-1) & & & \\
 & & & v_{11} & = & (0,0,0,0,-1) & & & \\
N & = & \{1,X,Y,XY,Y^2\} & List & = & \emptyset & H & = & H \cup \{\frac{1}{2}Y^3 - \frac{1}{2}Y\}
\end{array}
$$

$\square$

4 His Siblings

4.1 Left multiplication by X_i

It is just a question of bookkeeping, deducing from the computations done by Möller Algorithm the matrices $\mathcal{M}(k) := \left(a_{ij}^k\right)_{ij}$ s.t. $\forall i, 1 \leq i \leq r, \mathfrak{X}_k(q_i) = \sum_j a_{ij}^k q_j$, i.e. the so-called "*Matphi*" matrix introduced in [5].

Example 4.1 Continuing the computation started in Ex. 3.3 we deduce easily:

$$
\mathcal{M}(1) = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \mathcal{M}(2) = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 2 & 2 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{pmatrix}
$$

$\square$

4.2 and Border Basis

Obviously, the knowledge of the matrices $\mathcal{M}(k)$ allow to apply Möller Algorithm to the finite sequence $\{\gamma_1, \ldots, \gamma_s\}$ of functionals defined by $Can(f,P) = \sum_{j=1}^{s} \gamma_i(f) q_i$ in order to solve (as it was originally proposed in [5]) the FGLM-problem:

Problem 4.2 ([5]) *Given a Gröbner basis of P w.r.t. the term-ordering $<_1$, to compute its Gröbner basis w.r.t. the term-ordering $<_2$.*

4.3 Duality (1)

The algorithm above produces a sequence $\mathbf{q}$ which is *triangular* to $\mathbb{L}'$. In order to get a sequence $\mathbf{q}' = \{q'_1, \ldots, q'_s\}$ which is *biorthogonal* to $\mathbb{L}'$ it is sufficient to perform Gaussian reduction to the matrix $(\ell'_i(q_j))_{ij}$ applying the same operations to the q_i's.

Of course, in order to preserve the multiplication structure, the matrices $\mathcal{M}(k)$ must be updated accordingly.

Example 4.3 Continuing the computation of Ex. 3.3 we have

$$(\ell'_i(q_j))_{ij} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

and it is therefore sufficient to define

$$\begin{array}{lclcl} q'_1 & := & q_1 - q_2 - q_3 - q_4 - 2q_5 & = & 1 - X - XY - Y^2, \\ q'_2 & := & q_2 - q_4 & = & X + XY, \\ q'_3 & := & q_3 + q_4 + q_5 & = & \frac{1}{2}Y + \frac{1}{2}Y^2, \\ q'_4 & := & q_4 & = & -XY, \\ q'_5 & := & q_5 & = & -\frac{1}{2}Y + XY + \frac{1}{2}Y^2, \end{array}$$

in order to obtain $\ell'_i(q'_j) = \delta_{ij}, \forall i, j$. □

This dualization has interested applications in the case in which the linear funcionals ℓ_i are point evaluations, i.e. when points $P_i = (x_{1i}, \ldots, x_{ni}) \in K^n$ are given s.t. $\ell_i(f) = f(x_{1i}, \ldots, x_{ni})$.

In this case, in fact, it holds $q'_i(P_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$ and $\mathbf{q}'$ is naturally related with Lagrange interpolation. In this case the polynomials q'_i, which are easy to compute, are called *separators* or *Forney functions*; as Forney functions they are used in coding theory to recover error magnitudes from syndromes.

A set of points $\{P_j : 1 \leq j \leq s\}$ are said to be *in Cayley-Bacharach position* when the separators have the same degree $deg(q'_i) = max\,(deg(t) : t \in N(P))$ the which can be easily tested as it was firstly remarked in [11].

4.4 Duality (2)

Above we dualized the matrix $(\ell'_i(q_j))_{ij}$ by operating over rows; of course we could have done it on columns, performing the same operations on the set $\mathbb{L}'$; in this case we would have obtained a set $\mathbf{\Lambda} := \{\lambda_1, \ldots, \lambda_s\}$ which is biorthogonal to $\mathbf{q}$. We will see below that also this approach thru columns has interesting applications.

Performing the same operations on the vectors $v_i = v(f_i, \mathbb{L})$, we obtain the values $w_i := v(f_i, \mathbf{\Lambda})$.

Example 4.4 In our example, performing column Gaussian reduction on

$$(\ell'_i(q_j))_{ij} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

we obtain

$$\begin{aligned} \lambda_1 &:= \ell_1, \\ \lambda_2 &:= \ell_2 - \ell_1, \\ \lambda_3 &:= \ell_3 - \ell_1, \\ \lambda_4 &:= \ell_4 + \ell_3 - \ell_2 - \ell_1, \\ \lambda_5 &:= \ell_5 + \ell_3 - 2\ell_1; \end{aligned}$$

and

$$\begin{aligned} w_1 &= (1,0,0,0,0), & w_2 &= (0,1,0,0,0), \\ w_3 &= (0,0,1,0,0), & w_4 &= (0,1,0,0,0), \\ w_5 &= (0,0,0,-1,0), & w_6 &= (0,0,0,-1,0), \\ w_7 &= (0,0,1,2,2), & w_8 &= (0,0,0,1,0), \\ w_9 &= (0,0,0,-1,0), & w_{10} &= (0,0,0,0,0), \\ w_{11} &= (0,0,0,0,-1). \end{aligned}$$

□

4.5 Right multiplication by X_i

Lemma 4.5 *Let $\mathbf{\Lambda} = \{\lambda_1, \ldots, \lambda_s\} \subset V^*$ and $\mathbf{q} = \{q_1, \ldots, q_s\} \subset V$ be biorthogonal. Then*

$$\forall k = 1..n, \forall j = 1..s, \mathfrak{X}_k(\lambda_j) = \sum_{i=1}^{s} \lambda_j(\mathfrak{X}_k(q_i))\lambda_i.$$

Proof: In fact, $\forall l = 1..s$, it holds

$$\mathfrak{X}_k(\lambda_j)(q_l) = (\lambda_j X_k)(q_l) = \lambda_j(X_k q_l) = \lambda_j(\mathfrak{X}_k(q_l)) = \sum_{i=1}^{s} \lambda_j(\mathfrak{X}_k(q_i))\lambda_i(q_l).\square$$

As a corollary of this result, the already computed matrices $\mathfrak{M}(k) = \left(a^k_{ij}\right)_{ij}$ which satisfy

$$\forall i, 1 \leq i \leq r, \mathfrak{X}_k(\lambda_j) = \sum_i a^k_{ij}\lambda_i$$

are also the one whose i^{th}-row is $v(X_k q_i, \mathbf{\Lambda})$.

Example 4.6 In our case, the previous computation of the w_i's produces:

$$\mathcal{N}(1) = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \mathcal{N}(2) = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 2 & 2 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

□

4.6 A Vandermonde Criterium

In our setting it is worthwhile to remark the following result:

Theorem 4.7 (Möller)
Let $\mathbb{L} := \{\ell_1, \ldots, \ell_s\} \subset V*$ be a finite sequence generating a right $\mathcal{P}$-submodule $L \subset V^*$. Let $\mathbf{p} := \{p_1, \ldots, p_s\} \subset V$ be a finite sequence genereting a left $\mathcal{P}$-submodule $P \subset V$. Then the following are equivalent:

1. $P = \mathfrak{P}(L)$;
2. $\det(\ell_i(p_j)) \neq 0$;
3. $L = \mathfrak{L}(P)$.

Proof:

1. $\Longrightarrow$ 2. Let us apply Alg. 3.2 but with the modification in which we perform the first s^{th} **While**-loops on $f := p_i$, $i = 1..s$. In each $(r+1)^{th}$ **While**-loop the assumption that $\det(\ell_i(p_j)) \neq 0$ allows us to conclude that f is linearly independent w.r.t. $q_1, \ldots, q_r$ and so that $\mathrm{Span}_K(\{q_1, \ldots, q_{r+1}\}) = \mathrm{Span}_K(\{p_1, \ldots, p_{r+1}\})$.

2. $\Longrightarrow$ 1. Let $\mathbf{q}$ be the biorthogonal sequence to $\mathbb{L}$. Since $\mathrm{Span}_K(\mathbf{q}) = \mathrm{Span}_K(\mathbf{p})$ and $(\ell_i(p_j))_{ij}$ is the matrix which expresses the representation of $\mathbf{p}$ in terms of $\mathbf{q}$, the conclusion is obvious.

□

5 And His Dual

Let us start with two silly remarks: First of all, the duality between the biorthogonal sets $\mathbf{\Lambda} = \{\lambda_1, \dots, \lambda_s\}$ and $\mathbf{q} = \{q_1, \dots, q_s\}$ involves a *third* set: $\mathbf{N} := \mathbf{N}(P) = \{t_1, \dots, t_s\}$ under the obvious relation

$$\forall i, j : t_i = T(q_i), \lambda_j(q_i) = \delta_{ji}.$$

As funny as it can see, we will see soon that we could even claim that $t_i = T(\lambda_i)$.

The second remark we can do is that the elementary duality $(V^*)^* = V$ suggests to apply Möller Algorithm in a dual way: in Alg. 3.2 we have deduced the left module structure of $P := \mathfrak{P}(L)$ by performing linear algebra on the vectors of the evaluation of its polynomials at the functionals in $\mathbb{L}$ or $\mathbf{\Lambda}$; therefore linear algebra on the vectors of the evaluation of the functionals of $L = \mathfrak{L}(P)$ at the polynomials in $\mathbf{q}$ should deduce the right module structure of L, the *canonical module* of V/P.

For the left $\mathcal{P}$-module $V/\mathbf{T}(P)$ we call *socle* the set of terms

$$\mathfrak{S}(P) := \{t \in \mathbf{N}(P) : X_k t \in \mathbf{T}(P), \forall k = 1..n\}.$$

Let us assume that $\mathrm{Card}(\mathfrak{S}(P)) = \nu$, $\mathfrak{S}(P) = \{t_{i_1}, \dots, t_{i_\nu}\}$ and let us define $\mathbf{\Sigma} : \mathcal{P}^\nu \mapsto L$ by $\mathbf{\Sigma}(\sum_{j=1}^{\nu} f_j e_j) = \sum_{j=1}^{\nu} \lambda_{i_j} f_j$. We claim that $\mathbf{\Sigma}$ is surjective and that the following algorithm, which essentially is the dual version of Möller's one, computes

- the module $U := \ker(\mathbf{\Sigma})$ by giving $\mathbf{N}(U)$ and its B-bases $\mathcal{B}(U)$;
- an explicit vector space isomorphism

$$\sigma : L = \mathrm{Span}_K(\mathbf{\Lambda}) \mapsto \mathrm{Span}_K(\mathbf{N}(U)) \subset \mathcal{P}^\nu$$

  which is a splitting homomorphism for $\mathbf{\Sigma}$.

Algorithm 5.1 (Gröbner-Kreuzer)

$I := \emptyset$; $H := \emptyset$
For all $i = 1..\nu$ **do**
  $\sigma(\lambda_{j_i}) := e_i, \tau(\lambda_{j_i}) := T(q_{j_i})$
  $I := I \cup \{j_i\}, T(\lambda_{j_i}) = e_i$
$List := \{\lambda_{i_j} X_k, j = 1..\nu, k = 1..n\}$
While $List \neq \emptyset$ **do**
  For all $\lambda_i X_k \in List \mid X_k$ doesn't divide $\tau(\lambda_i)$ **then**
    %% $\mathfrak{X}_k(\lambda_i) = 0$
    $p := \sigma(\lambda_i) X_k$
    $T(p) = T(\lambda_i) X_k$
    $p := \mathbf{B} - \mathbf{reduce}(p, H)$
    $H := H \cup \{p\}$

$List := List \setminus \{\lambda_{i_j} X_k\}$
Choose $\lambda_i X_k \in List$ such that
$\tau(\lambda_i)X_k^{-1} = max_< \left(\tau(\lambda_j)X_l^{-1} : \lambda_j X_l \in List\right)$
$List := List \setminus \{\lambda_i X_k\}$
Let $\mathbf{j} : T(q_\mathbf{j}) = \tau(\lambda_i)X_k^{-1}$
Let $\mathfrak{X}_k(\lambda_i) := \sum_{j=1}^r c_j \lambda_j$
$\%\%\ c_j \neq 0 \Longrightarrow j \in I \cup \{\mathbf{j}\}$
If $\mathbf{j} \in I$ then
$p := \sigma(\lambda_i)X_k - \sum_{j=1}^r c_j \sigma(\lambda_j)$
$T(p) = T(\lambda_i)X_k$
$p := \mathbf{B} - \mathbf{reduce}(p, H)$
$H := H \cup \{p\}$
else
$$\sigma(\lambda_\mathbf{j}) := c_\mathbf{j}^{-1}\sigma(\lambda_i)X_k - \sum_{\substack{j=1\\ j\neq\mathbf{j}}}^{r} c_j c_\mathbf{j}^{-1}\sigma(\lambda_j)$$
$\tau(\lambda_\mathbf{j}) := T(q_\mathbf{j})$,
$I := I \cup \{\mathbf{j}\}, T(\lambda_\mathbf{j}) = T(\lambda_i)X_k$
$List := List \cup \{\lambda_\mathbf{j} X_k : k = 1..n\}$

while **B-reduce** is the procedure discussed in [11]: when it is called, each element in $f \in H$ is s.t. $f - T(f)$ is a linear combination of terms in$\{T(\lambda_i), i \in I\}$ and p is s.t. $p - T(f)$ is a linear combination of terms in $\{T(\lambda_i), i \in I\} \cup \{T(f), f \in H\}$; $\mathbf{B} - \mathbf{reduce}(p, H)$ substitutes in p each occurence of a term $T(f)$ with $f - T(f)$; as a consequence $f := \mathbf{B} - \mathbf{reduce}(p, H)$ is s.t. $f - T(f)$ is a linear combination of terms in$\{T(\lambda_i), i \in I\}$.

Example 5.2 In our example we have $\mathfrak{S}(P) = \{XY, Y^2\}$. Denoting l_i the linear functional treated in the i^{th} **While**-loop the computation (where $T(\cdot)$ is indicated in **bold**) is:

$$\begin{array}{rclrclrcl}
l_1 & = & \lambda_5 & & & & \sigma(\lambda_5) & = & (\mathbf{1}, 0) \\
T(\lambda_5) & = & Y^2 & I & = & \{5\} & & & \\
l_2 & = & \lambda_4 & & & & \sigma(\lambda_4) & = & (0, \mathbf{1}) \\
T(\lambda_4) & = & XY & I & = & \{5, 4\} & & & \\
l_3 & = & \lambda_5 X & p & = & (X, 0) & H & = & \{(\mathbf{X}, 0)\} \\
l_4 & = & \lambda_5 Y & \lambda_5 Y & = & 2\lambda_3 - \lambda_5 & \sigma(\lambda_3) & = & (\frac{1}{2}\mathbf{Y} + \frac{1}{2}, 0) \\
T(\lambda_3) & = & Y & I & = & \{5, 4, 3\} & & & \\
l_5 & = & \lambda_3 X & p & = & (\frac{1}{2}XY + \frac{1}{2}X, 0) & H & = & H \cup \{(\mathbf{XY}, 0)\} \\
l_6 & = & \lambda_4 X & \lambda_4 X & = & -\lambda_3 + \lambda_4 & H & = & H \cup \{(\frac{1}{2}Y + \frac{1}{2}, \mathbf{X} - 1)\} \\
l_7 & = & \lambda_4 Y & \lambda_4 Y & = & -\lambda_2 + 2\lambda_3 - \lambda_4 & \sigma(\lambda_2) & = & (Y + 1, -\mathbf{Y} - 1) \\
T(\lambda_2) & = & X & I & = & \{5, 4, 3, 2\} & & & \\
l_8 & = & \lambda_2 Y & p & = & (Y^2 + Y, -Y^2 - Y) & H & = & H \cup \{(-Y^2 - Y, \mathbf{Y^2} + Y) \\
l_9 & = & \lambda_3 Y & \lambda_3 Y & = & \lambda_3 + \lambda_1 & \sigma(\lambda_1) & = & (\frac{1}{2}\mathbf{Y^2} - \frac{1}{2}, 0) \\
T(\lambda_1) & = & 1 & I & = & \{5, 4, 3, 2, 1\} & & & \\
l_{10} & = & \lambda_1 X & p & = & (\frac{1}{2}XY^2 - \frac{1}{2}X, 0) & H & = & H \cup \{(\mathbf{XY^2}, 0)\} \\
l_{11} & = & \lambda_1 Y & p & = & (\frac{1}{2}Y^3 - \frac{1}{2}Y, 0) & H & = & H \cup \{(\mathbf{Y^3} - Y, 0)\} \\
l_{12} & = & \lambda_2 X & \lambda_2 X & = & \lambda_2 + \lambda_1 & H & = & H \cup \{(\frac{1}{2}Y^2 + \frac{1}{2}Y, \mathbf{XY} - Y)\}
\end{array}$$

$\square$

Remark 5.3 1. Hints of this algorithm can be found in a footnote of Gröbner book [7], where he used this idea to decompose a $(X_1, \ldots, X_n)$-primary ideal $P \subset K[X_1, \ldots, X_n]$ into minimal components; essentially the same idea is proposed by Kreuzer in the case in which $P \subset K[X_1, \ldots, X_n]$ is a Gorenstein ideal defined by points.

2. As I claimed, $\tau(\lambda_i) = T(q_i) = t_i, \forall i$; also

$$\tau(\lambda_k) = \tau e_i = \tau\sigma(\lambda_{j_i}) \Longrightarrow \tau T(\lambda_k) = \tau(e_i), \forall k$$

holds.

3. It is worthwhile to remark that if we apply the algorithm not to the module V/P but directly to the module $V/\mathbf{T}(P)$ obtaining the canonical module structure of $\mathfrak{L}(\mathbf{T}(P))$ the computation is essentially the same, except that at each **While**-loop instead of producing p (resp. $\sigma(\lambda_\mathbf{j})$) the algorithm produces $T(p)$ (resp. $T(\lambda_\mathbf{j})$). This is another, unexpected, instance of the usual general pattern in which Gröbner technology in order to obtain an information on $V \cong \mathfrak{P}^\nu/U$ computes the information on $\mathfrak{P}^\nu/T(U)$ and "lifts" it from $\mathfrak{P}^\nu/T(U)$ to V.

4. This dual Möller algorithm seems apparently sufficient to study the structure of the Gröbner basis of the ideals of a set of points in $K[X, Y]$, merging and proving the results by Lazard [10] and Cerlienco-Mureddu [4].

6 Gröbner Representation

With the discussion developed above, we hope that the following definition looks natural.

Definition 6.1 *Let* $U \subset \mathcal{P}^\nu$ *be a right* $\mathcal{P}$*-module such that* $\dim_K(\mathcal{P}^\nu/U)$ *is finite.*

We call a Gröbner representaion *of* U *the assignemnt of*

- *a sequence* $\mathbf{q} = \{q_1, \ldots, q_s\} \subset \mathcal{P}^\nu$ *s.t.* $\mathcal{P}^\nu/U = \mathrm{Span}_K(\mathbf{q})$*, and*
- *the matrices* $\mathcal{M}(k) := \left(a_{ij}^k\right)_{ij}$ *s.t.* $\forall i, 1 \le i \le r, \mathfrak{X}_k(q_i) = \sum_j a_{ij}^k q_j$.

In this setting, if U is given by a Gröbner representaion and $W, U \subset W \subset \mathcal{P}^\nu$ is a submodule generated by $(g_1, \ldots, g_h)$ the following algorithm allows to compute a Gröbner representation of W.

Algorithm 6.2 (Traverso)
$List := \{g_1, \ldots, g_h\}, I = \{1, \ldots, s\}$
While $List \neq \emptyset$ **do**
  Choose $g \in List$
  $List := List \setminus \{g\}$
  Compute $c_i, i \in I : \mathrm{Can}(g, U) = \sum_{i \in I} c_i q_i$

Let $l := \max\{i \in I : c_i \neq 0\}$
$I := I \setminus \{l\}$
For all $i, j \in I, k = 1..n$ **do**
$a^k_{ij} := a^k_{ij} - c_l^{-1} c_j a^k_{il}$
$List := List \cup \{\mathfrak{X}_k(g) : k = 1..n\}$
$\mathbf{q} = \{q_i, i \in I\}$

7 Solving?

In [1] the setting presented here was already applied to compute the roots of a zero-dimensional squarefree ideal $I \subset K[X_1, \ldots, X_n]$ and was later generalized removing the restriction that the roots are not simple in [12].

Theorem 7.1 (Auzinger-Stetter)
Let $A := K[X_1, \ldots, X_n]/I$ *and* $\mathbf{q} := \{q_1, \ldots q_s\}$ *be s.t.* $A \cong \mathrm{Span}_K(\mathbf{q})$. *For each* $g \in K[X_1, \ldots, X_n]$ *let* $\mathfrak{M}(g) := (a_{ij})_{ij}$ *be the matrix s.t.* $gq_i = \sum_j a_{ij} q_j$.

Assume that $P_1, \ldots, P_s \in K^s$ *are the roots of I; then the eigenvalues of* $\mathfrak{M}(g)$ *are* $g(P_1), \ldots, g(P_s)$ *and (if the eigenvalues are all different)* $(q_1(P_i), \ldots q_s(P_i))$ *is the eigenvector corresponding to* $g(P_i)$.

Example 7.2 In our setting let us consider $g := X + 2Y$ so that

$$\mathfrak{M}(g) = \begin{pmatrix} 0 & 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & -2 & 0 \\ 0 & 0 & 2 & 3 & 4 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -2 \end{pmatrix}$$

It is easy to check that $\mathfrak{M}(g)$ has the following eigenvalues e_i and eigenvectors v_i corrsponding to the points P_i:

$$\begin{array}{lllllllll} P_1 & = & (0,0) & e_1 & = & 0 & v_1 & = & (1,\mathbf{0},\mathbf{0},0,0) \\ P_2 & = & (1,0) & e_2 & = & 1 & v_2 & = & (1,\mathbf{1},\mathbf{0},0,0) \\ P_3 & = & (0,1) & e_3 & = & 2 & v_3 & = & (1,\mathbf{0},\mathbf{1},0,0) \\ P_4 & = & (1,-1) & e_4 & = & -1 & v_4 & = & (1,\mathbf{1},-\mathbf{1},1,0) \\ P_5 & = & (0,-1) & e_5 & = & -2 & v_5 & = & (1,\mathbf{0},-\mathbf{1},0,1) \end{array}$$

References

[1] Auzinger, W., Stetter, H.J. *An elimination algoriothm for the computation of all zeros of a system of multivariate polynomial equations.* In: Intern. Series in Numer. Math. **86** (1988) 11-30

[2] Borges-Trenard M.A., Borges-Quintana M., Mora T. *Computing Gröbner Bases by FGLM Techniques in a Noncommutative Settings.* Preprint (1999).

[3] Buchberger B., Möller H.M. *The construction of Multivariate Polynomials with Preassigned Zeros.* EUROCAM'82, Marseille. L.N.Comp.Sci **144** (1982),24–31.

[4] Cerlienco, L,, Mureduu, M. *Algoritmi combinatori per l'interpolazione polinomiale in dimensione* ≥ 2.

[5] Faugère J., Gianni P., Lazard D., Mora T. *Efficient Computation of Zero-dimensional Gröbner Bases by Change of Ordering.* J. Symb. Comp. **16** (1993), 329–344.

[6] Gianni P., Mora T. *Algebraic Solution of Systems of Polynomial Equations Using Grbner Bases.* AAECC-5, Minorca. L.N.Comp.Sci **356** (1989),247–257.

[7] Gröber

[8] Kreuzer

[9] Labonté G.*An Algorithm for the Construction of Matrix Representations for Finite Presented Non-commutative Algebras.* J. Symb. Comp. **9** (1990), 27–38.

[10] Lazard, D. *Ideal Basis and Primary Decomposition: Case of two variables.* J. Symb. Comp. **1** (1985),261–270.

[11] Marinari M.G., Möller H.M., Mora T. *Gröbner Bases of Ideals Defined by Functionals with an Application to Ideals of Proyective Points.* AAECC **4** (1993), 103–145.

[12] Möller. H.M., Stetter, H.J. *Multivariate polynomial equations with multiple zeros solved by matrix eigenproblems.* Numer. Math. **70** (1995), 311–329

[13] Mora T. *An Introduction to Commutative and Noncommutative Gröbner Bases.* Th. Comp. Ssci. **134** (1994), 131–173.

[14] Reinert B., Madlener K., Mora T. *A Note on Nielsen Reduction and Coset Enumeration.* Proc. ISSAC 98, Rostock (1998),171–178.