A block version of a sparse-FGLM-like algorithm

Kevin Hyun, Vincent Neiger, Hamid Rahkooy, Éric Schost

November 26, 2017

Computing the Gröbner basis of an ideal with respect to a giving term ordering is an essential step in solving systems of polynomials. Certain term orderings, such as the degree reverse lexicographical ordering (*degrevlex*), tend to make the computation of the Gröbner basis faster: this has been observed empirically since the 1980's, and is now supported by theoretical results, at least for some "nice" families of inputs, such as complete intersections or certain determinantal systems [8, 13, 2]. On the other hand, other orderings, such as the lexicographical ordering (*lex*), make it easier to find the coordinates of the solutions, or to perform arithmetic operations in the corresponding residue class ring. For instance, for a zero-dimensional radical ideal I in generic coordinates in $\mathbb{K}[X_1, \ldots, X_n]$, for some field $\mathbb{K}$, the lexicographic Gröbner basis of I for the lexicographic order $X_1 > \cdots > X_n$ has the form

$$\{X_1 - R_1(X_n), \ldots, X_{n-1} - R_{n-1}(X_n), R_n(X_n)\}, \tag{1}$$

where $R_n \in \mathbb{K}[X_n]$ is squarefree and all R_i's are in $\mathbb{K}[X_n]$ of degree less than $\deg(R_n)$; this is known as the *shape lemma* [15]. The points in the variety $V(I)$ are then

$$\{(R_1(\alpha), \ldots, R_{n-1}(\alpha), \alpha) \mid \forall \alpha : R_n(\alpha) = 0\}.$$

As a result, the standard approach to solve a zero-dimensional system by means of Gröbner basis algorithms first computes a Gröbner basis for a degree ordering, then converts it to a more readable output, such as a lexicographic basis. As pointed out in [12], the latter step, while of polynomial complexity, can now be a bottleneck in practice. This paper will thus focus on this stage; in order to describe our contributions, we first discuss previous work on the question.

Let I be a zero-dimensional ideal in $\mathbb{K}[X_1, \ldots, X_n]$. As input, we assume that we know a monomial basis $\mathscr{B}$ of $Q = \mathbb{K}[X_1, \ldots, X_n]/I$, together with the multiplication matrices $\boldsymbol{M}_1, \ldots, \boldsymbol{M}_n$ of respectively $X_1, \ldots, X_n$ in this basis; we denote by D the degree of I, that is, the vector space dimension of Q. We should point out that knowing a degree Gröbner basis of I, computing the multiplication matrices efficiently is not a straightforward task: Faugère *et al.* showed how to do it in time $O(nD^3)$ in [11] and more recently, algorithms have been given with cost bound $O^{\sim}(nD^{\omega})$ [9, 10, 24], at least for some favorable families of inputs. Here, the notation $O^{\sim}$ hides polylogarithmic factors and ω is a feasible exponent for matrix multiplication. While improving these results in an interesting question in itself, we will not address it in this paper.

Given such an input, the FGLM algorithm [11] computes the lexicographic Gröbner basis of I in $O(nD^3)$ operations in $\mathbb{K}$. While the algorithm has an obvious relation to linear algebra, lowering the runtime to $O^{\tilde{}}(nD^\omega)$ was only recently achieved [9, 10, 24].

Polynomials as in (1) are a very useful data structure, but there is no guarantee that the lexicographic Gröbner basis of I possesses such a structure; when it does, we will say that I is in *shape position* (some sufficient conditions are in [3]). As an alternative, one may use Rouillier's Rational Univariate Representation algorithm [26] (see also [1, 4] for related considerations). The output is a description of the zero-set $V(I)$ by means of rational functions

$$\{F(T) = 0, \quad X_1 = \frac{G_1(T)}{G(T)}, \ldots, X_n = \frac{G_n(T)}{G(T)}\}, \tag{2}$$

where the multiplicities of a root τ of F equals that of I at the corresponding point $(G_1(\tau)/G(\tau), \ldots, G_n(\tau)/G(\tau)) \in V(I)$; the fact that we use rational functions makes it in particular possible to control precisely the bit-size of their coefficients, if we work over $\mathbb{K} = \mathbb{Q}$.

The algorithms of [1, 4, 26] rely on an idea that will be at the core of our algorithms as well: duality. Indeed, these algorithms compute sequences of values of the form $\tau_i = (\mathrm{trace}(X^i))_{i\geq 0}$ or $\tau_{i,j} = (\mathrm{trace}(X^i X_j))_{i\geq 0}$, where $\mathrm{trace} : Q \to \mathbb{K}$ is the trace form and X is typically of the form $X = t_1 X_1 + \cdots + t_n X_n$. From these values, it is then possible to recover the output in (2) by means of some structured linear algebra calculation.

A drawback of this approach is that we need to know the trace of all elements of the basis $\mathscr{B}$; while feasible in polynomial time, this is by no means straightforward. In [6], Bostan *et al.* introduced randomization to alleviate this issue: they show that computing values such as $\ell(X^i)$ and $\ell(X^i X_j)$, where ℓ is a random $\mathbb{K}$-linear form $Q \to \mathbb{K}$, allows one to compute a description of $V(I)$ of the form

$$\{P(T) = 0, \quad X_1 = V_1(T), \ldots, X_n = V_n(T)\}, \tag{3}$$

where P is a monic squarefree polynomial in $\mathbb{K}[T]$ and V_i is in $\mathbb{K}[T]$ of degree less than $\deg(P)$ for all i. In particular, this output will in general differ from the description in (2), since the latter keeps trace of the multiplicities of the solutions (the algorithm in [6] actually computes the *nil-indices* of the solutions).

The most costly part of such an algorithm is the computation of the values $\ell(X^i)$ and $\ell(X^i X_j)$; the rest essentially boils down to applying the Berlekamp-Massey algorithm and univariate polynomial arithmetic. In [12], Faugère and Mou pointed out that the multiplication matrices $\boldsymbol{M}_1, \ldots, \boldsymbol{M}_n$ can be expected to be sparse; they give precise estimates on their expected sparsity, assuming the validity of a conjecture due to Moreno-Socías [22]. On this basis, they designed several forms of *sparse FGLM* algorithms. For instance, if I is in shape position, the algorithms in [12] recover this basis, also by considering values of a linear form $\ell : Q \to \mathbb{K}$. For less favorable inputs, the algorithm falls back on the Berlekamp-Massey-Sakata algorithm [27], or plain FGLM.

The ideas at play in these algorithms are essentially Krylov space methods, using projections and Berlekamp-Massey techniques; they are also widely used in integer factorization

or discrete logarithm calculations [**?**, **?**]. It is then natural to adapt to our context the block version of such algorithms, as pioneered by Coppersmith [7] in the context of integer factorization; this makes it possible to easily parallelize the bottleneck of the algorithm. This was already discussed by Steel in [28], where he showed how to compute the analogue of polynomial P in (3). In that reference, one is only interested in the solutions in the base field $\mathbb{K}$ ($\mathbb{K}$ being a finite field in that context): the algorithm computes the roots of P in $\mathbb{K}$ and substitutes them in the input system, before computing a Gröbner basis in $n-1$ variables for each of them.

Our first contribution is to give a block version of the algorithm in [6] that extends the approach in [28] to compute all polynomials in (3) for essentially the same cost as the computation of P. **todo:** define the output

Basic algorithms. Over a field $\mathbb{K}$, recall that we can compute multiplication, division with remainder, extended GCD, and square free part of polynomial of degree at most n in $O^{\tilde{}}(n)$ field operations ($O^{\tilde{}}$ omits polylogarithmic factors) [14].

1 Basic results on linearly recurrent sequences

1.1 Scalar sequences

Let $\mathbb{K}$ be a field and consider a sequence $\mathcal{L} = (\ell_s)_{s\geq 0} \in \mathbb{K}^{\mathbb{N}}$ and the associated generating series $Z = \sum_{s\geq 0} \ell_s T^s \in \mathbb{K}[[T]]$. We say that a degree d polynomial $P \in \mathbb{K}[T]$ *cancels* a sequence $(\ell_s)_{s\geq 0}$ if $p_0\ell_s + \cdots + p_d\ell_{s+d} = 0$ for all $s \geq 0$, where $p_0, \ldots, p_d$ are the coefficients of P; this is equivalent to $\mathrm{rev}(P)Z$ being a polynomial of degree less than d, with $Z = \sum_{s\geq 0} \ell_s T^s$ and $\mathrm{rev}(P) = T^d P(1/T)$.

The sequence $(\ell_s)_{s\geq 0}$ is *linearly recurrent* if there exists a non-zero polynomial that cancels it; this is the case if and only if there exist polynomials A, B in $\mathbb{K}[T]$ such that $Z = A/B$; these polynomials are unique if we assume $\gcd(A, B) = 1$ and $B(0) = 1$. The *minimal polynomial* of a linearly recurrent sequence $\mathcal{L} = (\ell_s)_{s\geq 0}$ is the monic polynomial of lowest degree that cancels it. Given the closed form $Z = A/B$ as defined above, its minimal polynomial is $P = T^{\max(\deg(A)+1,\deg(B))}B(1/T)$. The *order* of $\mathcal{L}$ is the degree of this polynomial P.

It is often easier to work with a closed form that has the actual minimal polynomial as its denominator, rather than its reverse; this is done by working with generating series in the variable $1/T$. Let indeed $Z^* = \sum_{s\geq 0} \ell_s/T^{s+1}$ and P be any polynomial; then, P cancels the sequence $(\ell_s)_{s\geq 0}$ if and only if $Q = PZ^*$ is a polynomial of degree less than $\deg(P)$, so that $Z^* = Q/P$, with $\deg(Q) < \deg(P)$. Using generating series in $1/T$, the minimal polynomial of $(\ell_s)_{s\geq 0}$ is thus the monic polynomial P of minimal degree for which there exists $Q \in \mathbb{K}[T]$ that satisfies $Z^* = Q/P$.

In particular, given a sequence $(\ell_s)_{s\geq 0}$ and a polynomial that cancels it, there is a well-defined notion of associated numerator. The next definition makes this idea precise:

Definition 1.1. *Let $\mathcal{L} = (\ell_s)_{s\geq 0}$ be a sequence and let P be a polynomial that cancels $\mathcal{L}$. Then, the* numerator *of $\mathcal{L}$ with respect to P is defined as*

$$\Omega(\mathcal{L}, P) = P \sum_{s\geq 0} \frac{\ell_s}{T^{s+1}}.$$

In view of the discussion above, we deduce that $\Omega(S, P)$ is a polynomial of degree less than $\deg(P)$.

Remark 1.2.

- We may as well write this definition using expressions in variable T instead of $1/T$: given $\mathcal{L} = (\ell_s)_{s\geq 0}$ and P as above, we recover $\Omega(\ell, P)$ by considering $Z = \sum_{s\geq 0} \ell_s T^s$ and computing $A = \mathrm{rev}(P)L$, with $\mathrm{rev}(P) = T^d P(1/T)$, with $d = \deg(P)$; then

$$\Omega(\mathcal{L}, P) = T^{d-1} A(1/T).$$

- If $\deg(P) = d$, we only need to know the coefficients $\ell_0, \ldots, \ell_{d-1}$ to compute $\Omega(\mathcal{L}, P)$. Explicitly, we have

$$\Omega(\mathcal{L}, P) = (P \sum_{s=0}^{d} \ell_{d-s} T^s) \text{ div } T^d.$$

We give an example of this using the Fibonacci sequence $F = (1, 1, 2, 3, 5, 8, \ldots)$, which is linearly recurrent, with minimal polynomial $P = T^2 - T - 1$. Define $Z = \sum_{s\geq 0} F_s T^s$ and $\mathrm{rev}(P) = 1 - T - T^2$; then, we can write Z in closed form as:

$$Z = \frac{A}{\mathrm{rev}(P)} = \frac{1}{1 - T - T^2},$$

so that $\Omega(F, P) = T^{2-1} \cdot 1 = T$. Equivalently, we can define $Z^* = \sum_{s\geq 0} F_s / T^{s+1}$; then, we recover

$$\begin{aligned} \Omega(F, P) &= (T^2 - T - 1) \left(\frac{1}{T} + \frac{1}{T^2} + \frac{2}{T^3} + \frac{3}{T^4} + \cdots \right) \\ &= T. \end{aligned}$$

In terms of complexity, assuming that $\mathcal{L}$ has order at most D, given the first $2D$ terms of $\mathcal{L}$, we can recover its minimal polynomial P by means of the Berlekamp-Massey algorithm, or of Euclid's algorithm applied to rational function reconstruction; this can be done in $O(\mathsf{M}(D)\log(D))$ operations in $\mathbb{K}$. Once we know P, we can deduce $\Omega(\mathcal{L}, P)$ for the cost $\mathsf{M}(D)$ of one polynomial multiplication.

1.2 Linearly recurrent matrix sequences

Next, we discuss the analogue of these ideas for matrix sequences; our main goal is to give a cost estimate for the computation of a suitable *matrix generator* for a matrix sequence

$\mathcal{F}$, obtained by means of recent algorithms for approximants bases. A similar discussion (without a cost analysis) is in [30, Chapter 4].

We first present the notion of linear recurrence for sequences of matrices over a field $\mathbb{K}$, which extends the well-known notion for sequences in $\mathbb{K}^{\mathbb{N}}$; the definition comes from [20, Section 3] and [30, Definition 4.2].

Definition 1.3. *Let $\mathcal{F} = (F_s)_{s\geq 0} \subset \mathbb{K}^{m\times m'}$ be a matrix sequence. Then,*

- *a polynomial $P = p_0 + \cdots + p_d T^d \in \mathbb{K}[T]$ is a* scalar relation *for $\mathcal{F}$ if $p_0 F_s + \cdots + p_d F_{s+d} = 0$ holds for all $s \geq 0$;*
- *a polynomial vector $\boldsymbol{p} = p_0 + \cdots + p_d T^d \in \mathbb{K}[T]^{1\times m}$ is a* (left, vector) relation *for $\mathcal{F}$ if $p_0 F_s + \cdots + p_d F_{s+d} = 0$ holds for all $s \geq 0$;*
- *$\mathcal{F}$ is* linearly recurrent *if there exists a non-zero scalar relation for $\mathcal{F}$.*

For designing efficient algorithms, it will be useful to rely on operations on polynomials or truncated series; hence the following characterization of vector relations.

Lemma 1.4. *Consider a matrix sequence $\mathcal{F} = (F_s)_{s\geq 0} \subset \mathbb{K}^{m\times m'}$ and its generating series $\boldsymbol{Z} = \sum_{s\geq 0} F_s/T^{s+1} \in \mathbb{K}[\![T^{-1}]\!]^{m\times m'}$. Then, $\boldsymbol{p} \in \mathbb{K}[T]^{1\times m}$ is a vector relation for $\mathcal{F}$ if and only if the entries of $\boldsymbol{q} = \boldsymbol{p}\boldsymbol{Z}$ are in $\mathbb{K}[T]$; furthermore, in this case,* $\deg(\boldsymbol{q}) < \deg(\boldsymbol{p})$.

Proof. Let $\boldsymbol{p} = \sum_{0\leq k\leq d} p_k T^k$. For $s \geq 0$, the coefficient of $\boldsymbol{q}$ of degree $-s-1 < 0$ is $\sum_{0\leq k\leq d} p_k F_{s+k}$. Hence the equivalence, by definition of a relation. The degree comparison is clear since $\boldsymbol{Z}$ has only terms of negative degree. □

Concerning the algebraic structure of the set of vector relations, we have the following basic result, which can be found for example in [32, 20, 30].

Lemma 1.5. *The sequence $\mathcal{F}$ is linearly recurrent if and only if the set of left vector relations for $\mathcal{F}$ is a $\mathbb{K}[T]$-submodule of $\mathbb{K}[T]^{1\times m}$ of rank m.*

Note however that a matrix sequence may admit nontrivial vector relations and have no scalar relation (and therefore not be linearly recurrent with the present definition); in this case the module of vector relations has rank less than m.

Definition 1.6. *Let $\mathcal{F} \subset \mathbb{K}^{m\times m'}$ be linearly recurrent. A* (left) matrix generator $\boldsymbol{P}$ *for $\mathcal{F}$ is a matrix in $\mathbb{K}[T]^{m\times m}$ whose rows form a basis of the module of left vector relations for $\mathcal{F}$. This basis is said to be*

- minimal *if the matrix is row reduced [34, 19];*
- canonical *if the matrix is in Popov form [25, 19].*

Note that the canonical generator is also a minimal generator; furthermore, all matrix generators $\boldsymbol{P} \in \mathbb{K}[T]^{m\times m}$ for $\mathcal{F}$ have the same determinantal degree $\deg(\det(\boldsymbol{P}))$, which we denote by $\Delta(\mathcal{F})$.

We now show that minimal matrix generators are denominators in some irreducible fraction description of the generating series of the sequence. This is a direct consequence of Lemmas 1.4 and 1.5 and of basic properties of polynomial matrices.

In what follows, for an $r \times s$ matrix $\boldsymbol{P}$ with entries in $\mathbb{K}[T]$, we denote by $\operatorname{rdeg}(\boldsymbol{P})$ and $\operatorname{cdeg}(\boldsymbol{P})$ respectively its row-degree and column-degrees, that is, the size-r vector of the degrees of its rows, respectively the size-s vector of the degrees of its columns.

Corollary 1.7. *A matrix sequence $\mathcal{F} = (F_s)_{s\geq 0} \subset \mathbb{K}^{m\times m'}$ is linearly recurrent if and only if its generating series $\boldsymbol{Z} = \sum_{s\geq 0} F_s/T^{s+1} \in \mathbb{K}[\![T^{-1}]\!]^{m\times m'}$ can be written as a matrix fraction $\boldsymbol{Z} = \boldsymbol{P}^{-1}\boldsymbol{Q}$ where $\boldsymbol{P} \in \mathbb{K}[T]^{m\times m}$ is nonsingular and $\boldsymbol{Q} \in \mathbb{K}[T]^{m\times m'}$. In this case, we have $\operatorname{rdeg}(\boldsymbol{Q}) < \operatorname{rdeg}(\boldsymbol{P})$ and $\deg(\det(\boldsymbol{P})) \geq \Delta(\mathcal{F})$, and $\boldsymbol{P}$ is a matrix generator of $\mathcal{F}$ if and only if $\deg(\det(\boldsymbol{P})) = \Delta(\mathcal{F})$ or, equivalently, the fraction $\boldsymbol{P}^{-1}\boldsymbol{Q}$ is irreducible (that is, $\boldsymbol{U}\boldsymbol{P} + \boldsymbol{V}\boldsymbol{Q} = \boldsymbol{I}$ for some polynomial matrices $\boldsymbol{U}$ and $\boldsymbol{V}$).*

We remark that we may also consider vector relations operating on the right: in particular, Lemma 1.4 shows that if the sequence is linearly recurrent then these right relations form a submodule of $\mathbb{K}[T]^{m'\times 1}$ of rank m'. Thus, a linearly recurrent sequence also admits a right canonical generator.

Now, we focus on our algorithmic problem: given a linearly recurrent sequence, find a minimal matrix generator. We assume the availability of bounds (d_ℓ, d_r) on the degrees of the left and right canonical generators, which allow us to control the number of terms of the sequence we will access during the algorithm. Since taking the Popov form of a reduced matrix does not change the degree, any minimal left matrix generator $\boldsymbol{P}$ has the same degree $\deg(\boldsymbol{P})$ as the left canonical generator: thus, d_ℓ is also a bound on the degree of any minimal left generator. The same remark holds for d_r and right minimal generators. The bounds d_ℓ, d_r are the same as γ_1, γ_2 in [30, Definitions 4.6 and 4.7]; see also δ_l, δ_r in [33, Section 4.2]; a result similar to the following lemma is in [30, Theorem 4.5].

Lemma 1.8. *Let $\mathcal{F} = (F_s)_{s\geq 0} \subset \mathbb{K}^{m\times m'}$ be linearly recurrent and let $d_r \in \mathbb{N}$ be such that the right canonical matrix generator of $\mathcal{F}$ has degree at most d_r. Then, $\boldsymbol{p} = p_0 + \cdots + p_d T^d \in \mathbb{K}[T]^{1\times m}$ is a left relation for $\mathcal{F}$ if and only if $p_0 F_s + \cdots + p_d F_{s+d} = 0$ holds for $s \in \{0, \ldots, d_r - 1\}$.*

Proof. Since the right canonical generator $\boldsymbol{P} \in \mathbb{K}[T]^{m'\times m'}$ is in column Popov form, we have $\boldsymbol{P} = \boldsymbol{L}\operatorname{Diag}(T^{t_1}, \ldots, T^{t_{m'}}) - \boldsymbol{Q}$ where $\operatorname{cdeg}(\boldsymbol{Q}) < \operatorname{cdeg}(\boldsymbol{P}) = (t_1, \ldots, t_{m'})$ componentwise and $\boldsymbol{L} \in \mathbb{K}^{m'\times m'}$ is unit upper triangular. We define the matrix $\boldsymbol{U} = \operatorname{Diag}(T^{d_r - t_1}, \ldots, T^{d_r - t_{m'}})\boldsymbol{L}^{-1}$, which is in $\mathbb{K}[T]^{m'\times m'}$ since $d_r \geq \deg(\boldsymbol{P}) = \max_j t_j$. Then, the columns of the right multiple $\boldsymbol{P}\boldsymbol{U} = T^{d_r}\boldsymbol{I}_{m'} - \boldsymbol{Q}\boldsymbol{U}$ are right relations for $\mathcal{F}$, and we have $\deg(\boldsymbol{Q}\boldsymbol{U}) < d_r$. As a consequence, writing $\boldsymbol{Q}\boldsymbol{U} = \sum_{0\leq k<d_r} Q_k T^k$, we have $F_{s+d_r} = \sum_{0\leq k<d_r} F_{s+k} Q_k$ for all $s \geq 0$.

Assuming that $\sum_{0\leq k\leq d} p_k F_{s+k} = 0$ holds for all $s \in \{0, \ldots, d_r - 1\}$, we prove by induction that this holds for all $s \in \mathbb{N}$. Let $s \geq d_r - 1$ and assume that this identity holds for all integers

up to s. Then, the identity concluding the previous paragraph implies that

$$\sum_{0\le k\le d} p_k F_{s+1+k} = \sum_{0\le k\le d} p_k \left(\sum_{0\le j<d_r} F_{s+1+k-d_r+j} Q_j \right)$$
$$= \sum_{0\le j<d_r} \underbrace{\left(\sum_{0\le k\le d} p_k F_{s+1-d_r+j+k} \right)}_{=0 \text{ since } s+1-d_r+j\le s} Q_j = 0,$$

and the proof is complete. □

The fast computation of matrix generators is usually handled via algorithms for computing minimal approximant bases [32, 30, 17]. The next result gives the main idea behind this approach. This result is similar to [30, Theorem 4.6] (see also [30, Theorems 4.7, 4.8, 4.9, 4.10]), but in some sense the reversal is on the input sequence rather than on the output matrix generator.

We recall from [31, 5] that, given a matrix $\boldsymbol{F} \in \mathbb{K}[T]^{m\times m'}$ and an integer $d \in \mathbb{N}$, the set of *approximants for* $\boldsymbol{F}$ *at order* d is defined as

$$\mathcal{A}(\boldsymbol{F}, d) = \{\boldsymbol{p} \in \mathbb{K}[T]^{1\times m} \mid \boldsymbol{p}\boldsymbol{F} = 0 \bmod T^d\}.$$

Then, the next theorem shows that relations for $\mathcal{F}$ can be retrieved as subvectors of approximants at order about $d_\ell + d_r$ for a matrix involving the first $d_\ell + d_r$ entries of $\mathcal{F}$.

Theorem 1.9. *Let $\mathcal{F} = (F_s)_{s\ge 0} \subset \mathbb{K}^{m\times m'}$ be a linearly recurrent sequence and let $(d_\ell, d_r) \in \mathbb{N}^2$ be such that the left (resp. right) canonical matrix generator of $\mathcal{F}$ has degree $\le d_\ell$ (resp. $\le d_r$). For $d > 0$, define*

$$\boldsymbol{F} = \begin{bmatrix} \sum_{0\le s<d} F_s T^{d-s-1} \\ -\boldsymbol{I}_{m'} \end{bmatrix} \in \mathbb{K}[T]^{(m+m')\times m'}. \tag{4}$$

Suppose that $d \ge d_r + 1$ and let $\boldsymbol{B} \in \mathbb{K}[T]^{(m+m')\times(m+m')}$ be a basis of $\mathcal{A}(\boldsymbol{F}, d_\ell + d_r + 1)$. Then,

- *if $\boldsymbol{B}$ is in Popov form then its $m \times m$ leading principal submatrix is the canonical matrix generator for $\mathcal{F}$;*
- *if $\boldsymbol{B}$ is row reduced then it has exactly m rows of degree $\le d_\ell$, and the corresponding submatrix $[\boldsymbol{P} \ \ \boldsymbol{R}]$ of $\boldsymbol{B}$ is such that $\boldsymbol{P} \in \mathbb{K}[T]^{1\times m}$ is a minimal matrix generator for $\mathcal{F}$.*

Proof. For any relation $\boldsymbol{p} \in \mathbb{K}[T]^{1\times m}$ for $\mathcal{F}$, there exists $\boldsymbol{r} \in \mathbb{K}[T]^{1\times m'}$ such that $\deg(\boldsymbol{r}) < \deg(\boldsymbol{p})$ and $[\boldsymbol{p} \ \ \boldsymbol{r}] \in \mathcal{A}(\boldsymbol{F}, d)$. Indeed, from Lemma 1.4, if $\boldsymbol{p}$ is a relation for $\mathcal{F}$ then $\boldsymbol{q} = \boldsymbol{p}\boldsymbol{Z}$ has polynomial entries, where $\boldsymbol{Z} = \sum_{s\ge 0} F_s T^{-s-1}$. Then, the vector $\boldsymbol{r} = -\boldsymbol{p}(\sum_{s\ge d} F_s T^{d-s-1})$ has polynomial entries, has degree less than $\deg(\boldsymbol{p})$, and is such that $[\boldsymbol{p} \ \ \boldsymbol{r}]\boldsymbol{F} = \boldsymbol{q}T^d$, hence $[\boldsymbol{p} \ \ \boldsymbol{r}] \in \mathcal{A}(\boldsymbol{F}, d)$.

Conversely, for any vectors $\boldsymbol{p} \in \mathbb{K}[T]^{1\times m}$ and $\boldsymbol{r} \in \mathbb{K}[T]^{1\times m'}$, if $[\boldsymbol{p} \ \ \boldsymbol{r}] \in \mathcal{A}(\boldsymbol{F}, d)$ and $\deg([\boldsymbol{p} \ \ \boldsymbol{r}]) \le d - d_r - 1$, then $\boldsymbol{p}$ is a relation for $\mathcal{F}$. Indeed, if $[\boldsymbol{p} \ \ \boldsymbol{r}] \in \mathcal{A}(\boldsymbol{F}, d)$ we have

$\boldsymbol{p}(\sum_{0\le s<d} F_s T^{d-s-1}) = \boldsymbol{r} \bmod T^d$. Since $d \ge d_r + 1$ and $\deg([\boldsymbol{p} \quad \boldsymbol{r}]) \le d - d_r - 1$, this implies that the coefficients of degree $d - d_r$ to $d-1$ of $\boldsymbol{p}(\sum_{0\le s<d} F_s T^{d-s-1})$ are zero. Then, Lemma 1.8 shows that $\boldsymbol{p}$ is a relation for $\mathcal{F}$.

The two items in the theorem are straightforward consequences. □

Using fast approximant basis algorithms, we obtain the main results from this section. They are stated in slightly more generality than needed, since in our main algorithm, we will always use $m' = m$. However, we believe that the more general form stated here may find further applications. The proof is a direct consequence of the previous theorem, using the algorithms of respectively [16], [35] and [18]. **todo:** choose $d \ge \dots$

Corollary 1.10. *Let $\mathcal{F} \subset \mathbb{K}^{m\times m'}$ be a linearly recurrent sequence and let $d = d_\ell + d_r + 1$, where $(d_\ell, d_r) \in \mathbb{N}^2$ are such that the left (resp. right) canonical matrix generator of $\mathcal{F}$ has degree $\le d_\ell$ (resp. $\le d_r$). Then, given $F_0, \dots, F_{d-1}$,*

- *if $m' \in \Omega(m)$, a minimal left matrix generator for $\mathcal{F}$ can be computed in $O(m'^{\omega}\mathsf{M}(d)\log(d))$ operations in $\mathbb{K}$;*
- *if $m' \in O(m)$, a minimal left matrix generator for $\mathcal{F}$ can be computed in $O(m^{\omega}\mathsf{M}(m'd/m)\log(m'd))$ operations in $\mathbb{K}$;*
- *the left canonical matrix generator for $\mathcal{F}$ can be computed in $O((m+m')^{\omega-1}\mathsf{M}(m'd)\log(m'd)^3)$ operations in $\mathbb{K}$.*

In particular, when $m' = m$, we can find the left canonical matrix generator of $\mathcal{F}$ in $O(m^{\omega}\mathsf{M}(d)\log(d))$ operations in $\mathbb{K}$.

1.3 The block Wiedemann algorithm

Finally, we apply the results seen above to a particular class of matrix sequences, namely the Krylov sequences used in the block Wiedemann algorithm. The claim below is not new; it can be found in [33] and [21]; we chose to give a close-to-self-contained presentation, that refers to the above references for the key properties.

Let $\boldsymbol{M}$ be in $\mathbb{K}^{D\times D}$ and $\boldsymbol{U}, \boldsymbol{V} \in \mathbb{K}^{D\times m}$ be two blocking matrices; we can then define the Krylov sequence $\mathcal{F}^{\boldsymbol{U},\boldsymbol{V}} = (F_s^{\boldsymbol{U},\boldsymbol{V}})_{s\ge 0}$ by

$$F_s^{\boldsymbol{U},\boldsymbol{V}} = \boldsymbol{U}^{\perp}\boldsymbol{M}^s\boldsymbol{V}, \quad s \ge 0.$$

This sequence is linearly recurrent, since the minimal polynomial of $\boldsymbol{M}$ is a scalar relation for it. The following theorem gives a bound on the degree of a minimal left generator of $\mathcal{F}^{\boldsymbol{U},\boldsymbol{V}}$, for generic choices of $\boldsymbol{U}$ and $\boldsymbol{V}$; we also state properties of the invariant factors of such a generator. **todo:** check Turner **todo:** define genericity

To do so, we let $s_1, \dots, s_r$ be the invariant factors of $T\boldsymbol{I}_D - \boldsymbol{M}$, ordered in such a way that $s_r | s_{r-1} | \dots | s_1$, and let $d_i = \deg(s_i)$ for all i; for $i > r$, we let $s_i = 1$, with $d_i = 0$. We define $\nu = d_1 + \dots + d_m \le D$ and $\delta = \lceil \nu/m \rceil \le \lceil D/m \rceil$. **todo:** unify subscripts $\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ or $\mathcal{F}^{\boldsymbol{U},\boldsymbol{V}}$

Theorem 1.11. *For a generic choice of $\boldsymbol{U}$ and $\boldsymbol{V}$, the following holds. Let $\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ be a minimal left generator for $\mathcal{F}^{\boldsymbol{U},\boldsymbol{V}}$ and denote by $\sigma_1, \ldots, \sigma_k$ the invariant factors of $\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$, for some $k \leq m$, ordered as above; write $\sigma_i = 1$ for $i > k$. Then,*

- *$\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ has degree δ;*
- *$s_i = \sigma_i$ for $1 \leq i \leq m$.*

Proof. Without loss of generality, we can assume that $\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ is a canonical left generator. Then, define the matrix sequence $\mathcal{F}^{\boldsymbol{V}} = (\boldsymbol{M}^s\boldsymbol{V})_{s\geq 0}$; this sequence is linearly generated as well, and we let $\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}$ be a canonical left generator for it.

We denote by $\langle \boldsymbol{V} \rangle$ the vector space generated by the columns of $\boldsymbol{V}, \boldsymbol{M}\boldsymbol{V}, \boldsymbol{M}^2\boldsymbol{V}, \ldots$, and we write $D_{\boldsymbol{V}} = \dim(\langle \boldsymbol{V} \rangle)$.

First, we prove that for any $\boldsymbol{V}$ in $\mathbb{K}^{D\times m}$, for a generic $\boldsymbol{U}$ in $\mathbb{K}^{D\times m}$, $\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}} = \boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$. Indeed, by [33, Lemma 4.2], there exists matrices $\boldsymbol{Q}_{\boldsymbol{V}}$ in $\mathbb{K}^{D\times D_{\boldsymbol{V}}}$ and $\boldsymbol{M}_{\boldsymbol{V}} \in \mathbb{K}^{D_{\boldsymbol{V}}\times D_{\boldsymbol{V}}}$, with $\boldsymbol{Q}_{\boldsymbol{V}}$ of full rank $D_{\boldsymbol{V}}$, and where $\boldsymbol{M}_{\boldsymbol{V}}$ is a matrix of the restriction of $\boldsymbol{M}$ to $\langle \boldsymbol{V} \rangle$, such that $\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} = \boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}$ if and only if the dimension of the span of $[\boldsymbol{Z}\ \boldsymbol{B}_{\boldsymbol{V}}\boldsymbol{Z}\ \boldsymbol{B}_{\boldsymbol{V}}^2\boldsymbol{Z}\ \cdots]$ is equal to $D_{\boldsymbol{V}}$, with $\boldsymbol{B}_{\boldsymbol{V}} = \boldsymbol{M}_{\boldsymbol{V}}^{\perp}$ and $\boldsymbol{Z} = \boldsymbol{P}_{\boldsymbol{V}}^{\perp}\boldsymbol{U} \in \mathbb{K}^{D_{\boldsymbol{V}}\times m}$.

By construction, one can find a basis of $\langle \boldsymbol{V} \rangle$ in which the matrix of $\boldsymbol{M}_{\boldsymbol{V}}$ is block-companion, with $m' \leq m$ blocks (take the $\boldsymbol{M}_{\boldsymbol{V}}$-span of the first column of $\boldsymbol{V}$, then of the second column, working modulo the previous vector space, etc.) Thus, $\boldsymbol{B}_{\boldsymbol{V}}$ is similar to a block-companion matrix with m' blocks as well; since $\boldsymbol{Z}$ has m columns, the span of $[\boldsymbol{Z}\ \boldsymbol{B}_{\boldsymbol{V}}\boldsymbol{Z}\ \boldsymbol{B}_{\boldsymbol{V}}^2\boldsymbol{Z}\ \cdots]$ has full dimension $D_{\boldsymbol{V}}$ for a generic $\boldsymbol{Z}$ (and for a generic $\boldsymbol{U}$, since $P_{\boldsymbol{V}}$ has rank $D_{\boldsymbol{V}}$). Thus, as claimed, for generic choices of $\boldsymbol{U}$ and $\boldsymbol{V}$, $\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} = \boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}$.

Let us next introduce a matrix $\mathscr{Y}$ of indeterminates of size $D \times m$, and let $\boldsymbol{P}^{\boldsymbol{M},\mathscr{Y}}$ be the canonical generating polynomial of the "generic" sequence $(\boldsymbol{M}^s\mathscr{Y})_{s\geq 0}$. The notation $\langle \mathscr{Y} \rangle$ and $D_{\mathscr{Y}}$ are defined as above. In particular, by [33, Proposition 6.1], the canonical generating polynomial $\boldsymbol{P}^{\boldsymbol{M},\mathscr{Y}}$ has degree δ and determinantal degree ν.

Now, for a generic $\boldsymbol{V}$ in $\mathbb{K}^{D\times m}$, $D_{\boldsymbol{V}} = D_{\mathscr{Y}}$. On the other hand, by [33, Lemma 4.3], for any $\boldsymbol{V}$ (including $\mathscr{Y}$), the degree of $\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}$ is equal to the first index d such that $\dim(\mathrm{span}([\boldsymbol{V}\ \boldsymbol{M}\boldsymbol{V}\ \cdots\ \boldsymbol{M}^{d-1}\boldsymbol{V}])) = D_{\boldsymbol{V}}$. As a result, for generic $\boldsymbol{V}$, $\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}$ and $\boldsymbol{P}^{\boldsymbol{M},\mathscr{Y}}$ have the same degree, that is, δ. The first item is proved.

We conclude by proving that for generic $\boldsymbol{U}, \boldsymbol{V}$, the invariant factors $\sigma_1, \ldots, \sigma_m$ of $\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ are $s_1, \ldots, s_m$. By [21, Theorem 2.12], for any $\boldsymbol{U}$ and $\boldsymbol{V}$ in $\mathbb{K}^{D\times m}$, for $i = 1, \ldots, m$, the i-th invariant factor σ_i of $\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ divides s_i, so that $\deg(\det(\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}})) \leq \nu$, with equality if and only if $\sigma_i = s_i$ for all $i \leq m$.

For $\boldsymbol{V}$ as above and any integers e, d, we let $\mathrm{Hk}_{e,d}(\boldsymbol{V})$ be the block Hankel matrix

$$\mathrm{Hk}_{e,d}(\boldsymbol{V}) = \begin{bmatrix} \boldsymbol{I}_D \\ \boldsymbol{M} \\ \boldsymbol{M}^2 \\ \vdots \\ \boldsymbol{M}^{e-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{V} & \boldsymbol{M}\boldsymbol{V} & \boldsymbol{M}^2\boldsymbol{V} & \cdots & \boldsymbol{M}^{d-1}\boldsymbol{V} \end{bmatrix}$$

By [21, Eq. (2.6)], $\mathrm{rank}(\mathrm{Hk}_{e,d}(\boldsymbol{V})) = \deg(\det(\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}))$ for $d \geq \deg(\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}})$ and $e \geq D$. We take $e = D$, so that $\mathrm{rank}(\mathrm{Hk}_{D,d}(\boldsymbol{V})) = \deg(\det(\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}))$ for $d \geq \deg(\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}})$. On the other

hand, the sequence $\mathrm{rank}(\mathrm{Hk}_{N,d}(\boldsymbol{V}))$ is constant for $d \geq D$; as a result, $\mathrm{rank}(\mathrm{Hk}_{D,D}(\boldsymbol{V})) = \deg(\det(\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}))$. For the same reason, we also have $\mathrm{rank}(\mathrm{Hk}_{D,D}(\mathscr{V})) = \deg(\det(\boldsymbol{P}^{\boldsymbol{M},\mathscr{V}}))$, so that for a generic $\boldsymbol{V}$, $\boldsymbol{P}^{\boldsymbol{M},\boldsymbol{V}}$ and $\boldsymbol{P}^{\boldsymbol{M},\mathscr{V}}$ have the same determinantal degree, that is, ν. As a result, for generic $\boldsymbol{U}$ and $\boldsymbol{V}$, we also have $\deg(\det(\boldsymbol{P}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}})) = \nu$, and the conclusion follows. □

2 Structure of the dual

this is a mess

For i in $\{1, \ldots, d\}$, let Q_i be the local algebra at $\boldsymbol{\alpha}_i$, that is $Q_i = \overline{\mathbb{K}}[X_1, \ldots, X_n]/I_i$, with I_i the $\mathfrak{m}_{\boldsymbol{\alpha}_i}$-primary component of I. By the Chinese Remainder Theorem, $Q \otimes_{\mathbb{K}} \overline{\mathbb{K}} = \overline{\mathbb{K}}[X_1, \ldots, X_n]/I$ is isomorphic to the direct product $Q_1 \times \cdots \times Q_d$. We let N_i be the *nil-index* of Q_i, that is, the maximal integer N such that $\mathfrak{m}_{\alpha_i}^N$ is not contained in I_i; for instance, $N_i = 0$ if and only if Q_i is a field, if and only if $\boldsymbol{\alpha}_i$ is a non-singular root of I. We also let $D_i = \dim_{\overline{\mathbb{K}}}(Q_i)$, so that we have $D_i \geq N_i$ and $D = D_1 + \cdots + D_d$.

The sequences we consider below are of the form $(\ell(t^s))_{s\geq 0}$, for ℓ a $\mathbb{K}$-linear form $Q \to \mathbb{K}$ and t in Q. For such sequences, the following standard result will be useful.

Lemma 2.1. *Let t be in Q and let $P \in \mathbb{K}[T]$ be its minimal polynomial. For a generic choice of ℓ in $\hom_{\mathbb{K}}(Q, \mathbb{K})$, P is the minimal polynomial of the sequence $(\ell(t^s))_{s\geq 0}$.*

Fix i in $1, \ldots, d$. There exists a basis of the dual $\hom_{\overline{\mathbb{K}}}(Q_i, \overline{\mathbb{K}})$ consisting of linear forms $(\lambda_{i,j})_{1\leq j\leq D_i}$ of the form

$$\lambda_{i,j} : f \mapsto (\Lambda_{i,j}(f))(\boldsymbol{\alpha}_i),$$

where $\Lambda_{i,j}$ is the operator

$$f \mapsto \Lambda_{i,j}(f) = \sum_{\mu=(\mu_1,\ldots,\mu_n)\in S_{i,j}} c_{i,j,\mu} \frac{\partial^{\mu_1+\cdots+\mu_n} f}{\partial X_1^{\mu_1} \cdots \partial X_n^{\mu_n}},$$

for some finite subset $S_{i,j}$ of $\mathbb{N}^n$ and non-zero constants $c_{i,j,\mu}$ in $\overline{\mathbb{K}}$. For instance, when $\boldsymbol{\alpha}_i$ is non-singular, we have $D_i = 1$, so there is only one function $\lambda_{i,j}$, namely $\lambda_{i,1}$, we write it $\lambda_{i,1}(f) = f(\boldsymbol{\alpha}_i)$.

More generally, we can always take $\lambda_{i,1}$ of the form $\lambda_{i,1}(f) = f(\boldsymbol{\alpha}_i)$; for $j > 1$, we can then also assume that $S_{i,j}$ does not contain $\mu = (0, \ldots, 0)$ (that is, all terms in $\Lambda_{i,j}$ have order 1 or more). Thus, introducing new variables $(U_{i,j})_{j=1,\ldots,D_i}$, we deduce the existence of non-zero homogeneous linear forms $P_{i,\mu}$ in $(U_{i,j})_{j=1,\ldots,D_i}$ such that for any λ in $\hom_{\overline{\mathbb{K}}}(Q_i, \overline{\mathbb{K}})$,

there exist $\boldsymbol{u}_i = (u_{i,j}) \in \overline{\mathbb{K}}^{D_i}$ such that we have

$$\begin{aligned}\lambda : f \mapsto \lambda(f) &= \sum_{j=1}^{D_i} u_{i,j}\lambda_{i,j}(f)\\ &= \sum_{j=1}^{D_i} u_{i,j}\big(\Lambda_{i,j}(f)\big)(\boldsymbol{\alpha}_i)\\ &= \sum_{j=1}^{D_i} u_{i,j} \sum_{\mu=(\mu_1,\ldots,\mu_n)\in S_{i,j}} c_{i,j,\mu}\frac{\partial^{\mu_1+\cdots+\mu_n} f}{\partial X_1^{\mu_1}\cdots\partial X_n^{\mu_n}}(\boldsymbol{\alpha}_i)\\ &= \sum_{\mu=(\mu_1,\ldots,\mu_n)\in S_i} P_{i,\mu}(\boldsymbol{u}_i)\frac{\partial^{\mu_1+\cdots+\mu_n} f}{\partial X_1^{\mu_1}\cdots\partial X_n^{\mu_n}}(\boldsymbol{\alpha}_i), \qquad (5)\end{aligned}$$

where S_i is the union of $S_{i,1},\ldots,S_{i,D_i}$, with in particular $P_{i,(0,\ldots,0)} = u_{i,1}$ and where $P_{i,\mu}$ depends only on $(u_{i,j})_{j=2,\ldots,D_i}$ for all μ in S_i, $\mu \neq (0,\ldots,0)$. Explicily, we can write $P_{i,\mu} = \sum_{j\in\{1,\ldots,D_i\} \text{ such that } \mu\in S_{i,j}} c_{i,j,\mu}U_{i,j}$.

Fix λ non-zero in $\hom_{\overline{\mathbb{K}}}(Q_i,\overline{\mathbb{K}})$. We can then define its *order* w and *symbol* π. The former is the maximum of all $|\mu| = \mu_1+\cdots+\mu_n$ for $\mu = (\mu_1,\ldots,\mu_n)$ in S_i such that $P_{i,\mu}(\boldsymbol{u}_i)$ is non-zero; by [23, Lemma 3.3] we have $w \leq N_i - 1$. Then, we let

$$\pi = \sum_{\mu\in S_i,\ |\mu|=w} P_{i,\mu}(\boldsymbol{u}_i)X_1^{\mu_1}\cdots X_n^{\mu_n}$$

be the *symbol* of λ; by construction, this is a non-zero polynomial. In the following paragraphs, we will need the next easy lemma.

Lemma 2.2. *Fix i in $\{1,\ldots,d\}$. For a generic choice of λ in $\hom_{\overline{\mathbb{K}}}(Q_i,\overline{\mathbb{K}})$, and of $t_1,\ldots,t_n$ in $\overline{\mathbb{K}}^n$, $\pi_i(t_1,\ldots,t_n)$ is non-zero.*

Proof. Let Ω be the maximum of all $|\mu| = \mu_1+\cdots+\mu_n$ for $\mu = (\mu_1,\ldots,\mu_n)$ in S_i, and define

$$\Pi = \sum_{\mu\in S_i,\ |\mu|=\Omega} P_{i,\mu}X_1^{\mu_1}\cdots X_n^{\mu_n} \in \overline{\mathbb{K}}[U_{i,1},\ldots,U_{i,D_i},X_1,\ldots,X_n];$$

this is by construction a non-zero polynomial. Thus, for a generic choice of $\boldsymbol{u}_i = (u_{i,1},\ldots,u_{i,D_i})$, that define a linear form λ in $\hom_{\overline{\mathbb{K}}}(Q_i,\overline{\mathbb{K}})$ as in Eq. (5), and of $t_1,\ldots,t_n$ in $\overline{\mathbb{K}}^n$, the value $\Pi(u_{i,1},\ldots,u_{i,D_i},t_1,\ldots,t_n)$ is non-zero. As a result, the symbol of such a linear form λ is $\pi = \sum_{\mu\in S_i,\ |\mu|=\Omega} P_{i,\mu}(\boldsymbol{u}_i)X_1^{\mu_1}\cdots X_n^{\mu_n}$, and $\pi(t_1,\ldots,t_n)$ is then non-zero. □

Finally, we say a word about global objects. Fix a linear form $\ell : Q \to \mathbb{K}$. By the Chinese Remainder Theorem, there exist unique $\ell_1,\ldots,\ell_d$, with ℓ_i in $\hom_{\overline{\mathbb{K}}}(Q_i,\overline{\mathbb{K}})$ for all i, such that the extension $\ell_{\overline{\mathbb{K}}} : Q\otimes_{\mathbb{K}}\overline{\mathbb{K}} \to \overline{\mathbb{K}}$ decomposes as $\ell_{\overline{\mathbb{K}}} = \ell_1+\cdots+\ell_d$. We call *support* of ℓ the

subset $\mathfrak{S}$ of $\{1,\dots,d\}$ such that ℓ_i is non-zero exactly for i in $\mathfrak{S}$. As a consequence, for all f in Q, we have

$$\begin{aligned}\ell(f) &= \ell_1(f) + \cdots + \ell_d(f)\\ &= \sum_{i\in\mathfrak{S}} \ell_i(f).\end{aligned} \tag{6}$$

For i in $\mathfrak{S}$, we denote by w_i and π_i respectively the order and the symbol of ℓ_i. For such a subset $\mathfrak{S}$, we also write $Q_{\mathfrak{S}} = \prod_{i\in\mathfrak{S}} Q_i$ and $V_{\mathfrak{S}} = \cup_{i\in\mathfrak{S}}\{\boldsymbol{\alpha}_i\}$.

2.0.1 A fundamental formula.

The following lemma gives an explicit form for a generating series of the form $\sum_{\ell\geq 0}\ell(vt^i)T^\ell$, for a linear form $\ell : Q \to \mathbb{K}$. A slightly less precise version of it is in [6].

Lemma 2.3. *Let ℓ be in $\hom_{\mathbb{K}}(Q,\mathbb{K})$, with support $\mathfrak{S}$, and let $\{\pi_i \mid i\in\mathfrak{S}\}$ and $\{w_i \mid i\in\mathfrak{S}\}$ be as above.*

Let $t = t_1X_1 + \cdots + t_nX_n$, for some $t_1,\dots,t_n$ in $\mathbb{K}$ and let v be in $\mathbb{K}[X_1,\dots,X_n]$. Then, we have the equality

$$\sum_{s\geq 0}\ell(vt^s)/T^{s+1} = \sum_{i\in\mathfrak{S}} \frac{v(\boldsymbol{\alpha}_i)\, w_i!\, \pi_i(t_1,\dots,t_n) + (T - t(\boldsymbol{\alpha}_i))A_{v,i}}{(T-t(\boldsymbol{\alpha}_i))^{w_i+1}}, \tag{7}$$

for some polynomials $\{A_{v,i}\in\overline{\mathbb{K}}[T] \mid i\in\mathfrak{S}\}$ (that depend on the choice of v), with $A_{v,i}$ of degree less than w_i for all i in $\mathfrak{S}$.

Proof. Take v and t as above. Consider first an operator of the form $f \mapsto \frac{\partial^{|\mu|}f}{\partial X_1^{\mu_1}\cdots\partial X_n^{\mu_n}}$, where we write $|\mu| = \mu_1 + \cdots + \mu_n$. Then, we have the following generating series identities, with coefficients in $\mathbb{K}(X_1,\dots,X_n)$:

$$\begin{aligned}\sum_{s\geq 0}\frac{\partial^{|\mu|}(vt^s)}{\partial X_1^{\mu_1}\cdots\partial X_n^{\mu_n}}\frac{1}{T^{s+1}} &= \sum_{s\geq 0}\frac{\partial^{|\mu|}(vt^s/T^{s+1})}{\partial X_1^{\mu_1}\cdots\partial X_n^{\mu_n}}\\ &= \frac{\partial^{|\mu|}}{\partial X_1^{\mu_1}\cdots\partial X_n^{\mu_n}}\left(\sum_{s\geq 0} vt^s/T^{s+1}\right)\\ &= \frac{\partial^{|\mu|}}{\partial X_1^{\mu_1}\cdots\partial X_n^{\mu_n}}\left(\frac{v}{T-t}\right)\\ &= v\,|\mu|!\prod_{1\leq k\leq n}\left(\frac{\partial t}{\partial X_k}\right)^{\mu_k}\frac{1}{(T-t)^{|\mu|+1}} + \frac{P_{|\mu|}(\boldsymbol{X},T)}{(T-t)^{|\mu|}} + \cdots + \frac{P_1(\boldsymbol{X},T)}{(T-t)}\\ &= v\,|\mu|!\prod_{1\leq k\leq n} t_k^{\mu_k}\frac{1}{(T-t)^{|\mu|+1}} + \frac{P(\boldsymbol{X},T)}{(T-t)^{|\mu|}},\end{aligned}$$

for some polynomials $P_1, \ldots, P_{|\mu|}, P$ in $\mathbb{K}[\boldsymbol{X}, T]$ that depend on the choices of μ, v and t, with $\deg(P_i, T) < i$ for all i and thus $\deg(P, T) < |\mu|$.

Take now a linear combination of such operators, such as $f \mapsto \sum_{\mu \in R} c_\mu \frac{\partial^{|\mu|} f}{\partial X_1^{\mu_1} \cdots \partial X_n^{\mu_n}}$. The corresponding generating series becomes

$$\sum_{s \geq 0} \sum_{\mu \in R} c_\mu \frac{\partial^{|\mu|}(vt^s)}{\partial X_1^{\mu_1} \cdots \partial X_n^{\mu_n}} \frac{1}{T^{s+1}} = v \sum_{\mu \in R} c_\mu |\mu|! \prod_{1 \leq k \leq n} t_k^{\mu_k} \frac{1}{(T-t)^{|\mu|+1}} + \sum_{\mu \in R} \frac{P_\mu(\boldsymbol{X}, T)}{(T-t)^{|\mu|}},$$

where each P_μ has degree less than $|\mu|$ in T. Let w be the maximum of all $|\mu|$ for μ in R. We can rewrite the above as

$$v\, w! \sum_{\mu \in R, |\mu| = w} c_\mu \prod_{1 \leq k \leq n} t_k^{\mu_k} \frac{1}{(T-t)^{w+1}} + \frac{A(\boldsymbol{X}, T)}{(T-t)^w},$$

for some polynomial A of degre less than w in T. If we let $\pi = \sum_{\mu \in R,\ |\mu| = w} c_\mu X_1^{\mu_1} \cdots X_n^{\mu_n}$, this becomes

$$\sum_{s \geq 0} \sum_{\mu \in R} c_\mu \frac{\partial^{|\mu|}(vt^s)}{X_1^{\mu_1} \cdots X_n^{\mu_n}} \frac{1}{T^{s+1}} = v\, w!\, \pi(t_1, \ldots, t_n) \frac{1}{(T-t)^{w+1}} + \frac{A(\boldsymbol{X}, T)}{(T-t)^w}.$$

Applying this formula to the sum in Eq. (6), we obtain the claim in the lemma. □

The most useful consequence of the previous lemma is the following interpolation formula, where we fix a subset $\mathfrak{S}$ of $\{1, \ldots, d\}$. The mapping $t : V_{\mathfrak{S}} \to \overline{\mathbb{K}}$ defined by $\boldsymbol{\alpha}_i \mapsto t(\boldsymbol{\alpha}_i)$ plays a special role in the formula in the lemma; this leads us to the following definitions.

- We consider ℓ and t as in Lemma 2.3, such that ℓ has support $\mathfrak{S}$.
- $\mathfrak{T}$ is the subset of $\mathfrak{S}$ consisting of all indices i such that
  - $\pi_i(t_1, \ldots, t_n)$ is non-zero;
  - $t(\boldsymbol{\alpha}_{i'}) \neq t(\boldsymbol{\alpha}_i)$ for $i' \neq i$ in $\mathfrak{S}$.
- $\{r_1, \ldots, r_c\}$ are the pairwise distinct values taken by t on $V_{\mathfrak{S}}$, for some $c \leq |\mathfrak{S}|$.
- $\mathfrak{t}$ is the set of all indices j in $\{1, \ldots, c\}$ such that
  - the fiber $t^{-1}(r_j) \subset V_{\mathfrak{S}}$ contains a single point, written $\boldsymbol{\alpha}_{\sigma_j}$;
  - the point $\boldsymbol{\alpha}_{\sigma_j}$ is in $\mathfrak{T}$ (equivalently, $\pi_{\sigma_j}(t_1, \ldots, t_n)$ is non-zero).

  Remark that $j \mapsto \sigma_j$ induces a one-to-one correspondence between $\mathfrak{t}$ and $\mathfrak{T}$.

Lemma 2.4. *Let ℓ, t and all other notation be as above. Let further M be the minimal polynomial of t in $Q_{\mathfrak{S}}$. Suppose that M is also the minimal polynomial of the sequence $(\ell(t^s))_{s \geq 0}$, then there exist non-zero constants $\{c_j \mid j \in \mathfrak{t}\}$ such that for v in $\mathbb{K}[X_1, \ldots, X_n]$,*

$$\Omega((\ell(vt^s))_{s \geq 0}, M) = c_j v(\boldsymbol{\alpha}_{\sigma_j}) \quad \textit{for all } j \textit{ in } \mathfrak{t}.$$

Proof. For $j = 1, \dots, c$, we write T_j for the set of all indices i in $\mathfrak{S}$ such that $t(\boldsymbol{\alpha}_i) = r_j$; the sets $T_1, \dots, T_c$ form a partition of $\mathfrak{S}$. When T_j has cardinality 1, we thus have $T_j = \{\sigma_j\}$.

Take an arbitrary v in $\mathbb{K}[X_1, \dots, X_n]$ and let us collect terms in Eq. (7) as

$$\begin{aligned}\sum_{s\geq 0} \ell(vt^s)\frac{1}{T^{s+1}} &= \sum_{j\in\{1,\dots,c\}} \sum_{i\in T_j} \frac{v(\boldsymbol{\alpha}_i)w_i!\,\pi_i(t_1,\dots,t_n) + (T-r_j)A_{v,i}}{(T-r_j)^{w_i+1}} \\ &= \sum_{j\in\mathfrak{s}} \frac{v(\boldsymbol{\alpha}_{\sigma_j})w_{\sigma_j}!\,\pi_{\sigma_j}(t_1,\dots,t_n) + (T-r_j)A_{v,\sigma_j}}{(T-r_j)^{w_{\sigma_j}+1}} \\ &\quad + \sum_{j\in\{1,\dots,c\}-\mathfrak{s}} \frac{\sum_{i\in T_j}\Big(\big[v(\boldsymbol{\alpha}_i)w_i!\,\pi_i(t_1,\dots,t_n) + (T-r_j)A_{v,i}\big](T-r_j)^{y_j-(w_i+1)}\Big)}{(T-r_j)^{y_j}},\end{aligned}$$

where y_j is the maximum of all w_i for i in T_j. Remark that for $v = 1$, our condition that π_i is non-zero for i in $\mathfrak{T}$ implies that in the second line, together with our assumption on the characteristic of $\mathbb{K}$, imply that all terms in the first sum are non-zero and in reduced form.

After simplyfing terms in the second sum, we can rewrite the expression above as

$$\sum_{s\geq 0} \ell(vt^s)\frac{1}{T^{s+1}} = \sum_{j\in\mathfrak{t}} \frac{v(\boldsymbol{\alpha}_{\sigma_j})w_{\sigma_j}!\,\pi_{\sigma_j}(t_1,\dots,t_n) + (T-r_j)A_{v,\sigma_j}}{(T-r_j)^{w_{\sigma_j}+1}} + \sum_{j\in\{1,\dots,c\}-\mathfrak{s}} \frac{D_{v,j}}{(T-r_j)^{z_{v,j}}},$$

for some positive integers $\{z_{v,j} \mid j \in \{1,\dots,c\}-\mathfrak{s}\}$ and polynomials $\{D_{v,j} \mid j \in \{1,\dots,c\}-\mathfrak{s}\}$ such that for all j in $\{1,\dots,c\}-\mathfrak{s}$, we have $\deg(D_{v,j}) < z_{v,j}$ and $\gcd(D_{v,j}, (T-r_j)) = 1$; the integers $z_{v,j}$ are uniquely determined by these conditions, except if $r_j = 0$, in which case we set $z_{v,j} = \deg(D_{v,j})+1$. Some of the polynomials $D_{v,j}$ may vanish, so we let $\mathfrak{u}_v \subset \{1,\dots,c\}-\mathfrak{s}$ be the set of all j for which this is not the case. We then arrive at our final form for this sum, namely

$$\sum_{s\geq 0} \ell(vt^s)\frac{1}{T^{s+1}} = \sum_{j\in\mathfrak{t}} \frac{v(\boldsymbol{\alpha}_{\sigma_j})w_{\sigma_j}!\,\pi_{\sigma_j}(t_1,\dots,t_n) + (T-r_j)A_{v,\sigma_{j;}}}{(T-r_j)^{w_{\sigma_j}+1}} + \sum_{j\in\mathfrak{u}_v} \frac{D_{v,j}}{(T-r_j)^{z_{v,j}}}, \tag{8}$$

where all terms in the second sum are non-zero and in reduced form (and similarly for the first sum, for $v = 1$). This implies that the minimal polynomial of the sequence $(\ell(vt^s))_{s\geq 0}$ is

$$M_v = \prod_{j\in\mathfrak{t}} (T-r_j)^{\zeta_j} \prod_{j\in\mathfrak{u}_v} (T-r_j)^{z_{v,j}},$$

for some integers $\{\zeta_j \leq w_{\sigma_j}+1 \mid j \in \mathfrak{t}\}$; for $v = 1$, we actually have $\zeta_j = w_{\sigma_j}+1$ for all such j.

Now, for $v = 1$, we assume that the minimal polynomial of the sequence $(\ell(t^s))_{s\geq 0}$ is the minimal polynomial M of t in $Q_{\mathfrak{S}}$. Writing $\mathfrak{u} = \mathfrak{u}_1$ and $z_k = z_{1,k}$ for all k in $\mathfrak{u}$, we can thus write it as

$$M = \prod_{j\in\mathfrak{t}} (T-r_j)^{w_{\sigma_j}+1} \prod_{j\in\mathfrak{u}} (T-r_j)^{z_j}.$$

Since it is the minimal polynomial of t in $Q_{\mathfrak{S}}$, it also cancels the sequence $(\ell(vt^s))_{s\geq 0}$ for any v, so that for all v, $\mathfrak{u}_v$ is contained in $\mathfrak{u}$ and M_v divides M. Remark also that the integer $\delta = \deg(M)$ is given by

$$\delta = \sum_{j\in\mathfrak{t}}(w_{\sigma_j}+1) + \sum_{j\in\mathfrak{u}} z_j.$$

For v arbitrary in $\mathbb{K}[X_1,\ldots,X_n]$, since M cancels the sequence $(\ell(vt^s))_{s\geq 0}$, $\Omega((\ell(vt^s))_{s\geq 0}, M)$ in the statement of the lemma is indeed a polynomial of degree less than δ (this proves our first claim). We can then rewrite the sum in Eq. (8) as $\sum_{s\geq 0}\ell(vt^s)/T^{s+1} = \Omega((\ell(vt^s))_{s\geq 0}, M)/M$, with

$$\begin{aligned}\Omega((\ell(vt^s))_{s\geq 0}, M) = &\sum_{j\in\mathfrak{t}}\Big(\big[v(\boldsymbol{\alpha}_{\sigma_j})w_{\sigma_j}!\,\pi_{\sigma_j}(t_1,\ldots,t_n) + (T-r_j)\tilde{A}_{v,\sigma_j}\big]\prod_{\iota\in\mathfrak{t}-\{j\}}(T-r_\iota)^{w_{\sigma_\iota}+1}\Big)\prod_{j\in\mathfrak{u}}(T-r_j)^{z_j}\\ &+\Big(\prod_{j\in\mathfrak{t}}(T-r_j)^{w_{\sigma_j}+1}\Big)\sum_{j\in\mathfrak{u}_v}\Big((T-r_j)^{z_j-z_{v,j}}D_{v,j}\prod_{\iota\in\mathfrak{u}-\{j\}}(T-r_\iota)^{z_\iota}\Big),\end{aligned}$$

In particular, for k in $\mathfrak{t}$, the value $\Omega((\ell(vt^s))_{s\geq 0}, M)(r_k)$ is

$$\begin{aligned}\Omega((\ell(vt^s))_{s\geq 0}, M)(r_k) &= v(\boldsymbol{\alpha}_{\sigma_k})w_{\sigma_k}!\,\pi_{\sigma_k}(t_1,\ldots,t_n)\prod_{\iota\in\mathfrak{t}-\{k\}}(r_\iota-r_k)^{w_{\sigma_\iota}+1}\prod_{j\in\mathfrak{u}}(r_j-r_k)^{z_k}\\ &= v(\boldsymbol{\alpha}_{\sigma_k})c_k,\end{aligned}$$

with

$$c_k = w_{\sigma_k}!\,\pi_{\sigma_k}(t_1,\ldots,t_n)\prod_{\iota\in\mathfrak{t}-\{k\}}(r_\iota-r_k)^{w_{\sigma_\iota}+1}\prod_{j\in\mathfrak{u}}(r_j-r_k)^{z_k}$$

for k in $\mathfrak{t}$. This is a non-zero constant, independent of v, which finishes the proof of the lemma. □

As an application, the following algorithm shows how to compute a zero-dimensional parametrization of $V_{\mathfrak{S}}$.

Algorithm 1 ParametrizationGeneric(ℓ, t)

Input:
- a linear form ℓ over $Q_{\mathfrak{S}}$
- $t = t_1X_1 + \cdots + t_nX_n$

Output: polynomials $(P, V_1, \ldots, V_n)$

1. let M be the minimal polynomial of the sequence $(\ell(t^s))_{s\geq 0}$ and let δ be its degree
2. let P be the squarefree part of M
3. let $C_1 = \Omega((\ell(t^s))_{s\geq 0}, M)$
4. **for** $i = 1,\ldots,n$ **do**
   (a) let $C_{X_i} = \Omega((\ell(X_it^s))_{s\geq 0}, M)$
5. **return** $((P, C_{X_1}/C_1 \bmod P, \ldots, C_{X_n}/C_1 \bmod P), t)$

Lemma 2.5. *Suppose that ℓ is a generic element of $\hom_{\mathbb{K}}(Q_{\mathfrak{K}}, \overline{\mathbb{K}})$ and that t is a generic linear form. Then the output $((P, V_1, \ldots, V_n), t)$ of* Parametrization(ℓ, t) *is a zero-dimensional parametrization of $V_{\mathfrak{S}}$.*

Proof. A generic choice of t separates the points of $V_{\mathfrak{S}}$, and we saw in Lemma 2.2 that for a generic choice of ℓ, $\pi_i(t_1, \ldots, t_n)$ vanishes for no i in $\mathfrak{S}$. As a result, $\mathfrak{T} = \mathfrak{S}$. Besides, we recall that for a generic ℓ in $\hom_{\mathbb{K}}(Q_{\mathfrak{S}}, \overline{\mathbb{K}})$, the minimal polynomials of $(\ell(t^s))_{s\geq 0}$ and of t are the same (Lemma 2.1).

Thus, the polynomial M we compute at step 1 is indeed the minimal polynomial of t, and we can apply the previous lemma, and for any root r_j of P, and $i = 1, \ldots, n$, we have

$$\frac{C_{X_i}(r_j)}{C_1(r_j)} = \frac{\Omega((\ell(X_i t^s))_{s\geq 0}, M)(r_j)}{\Omega((\ell(t^s))_{s\geq 0}, M)(r_j)} = \frac{c_j \alpha_{\sigma_j, i}}{c_j} = \alpha_{\sigma_j, i},$$

so that $C_{X_i}/C_1 \bmod P$ is the ith polynomial in the zero-dimensional parametrization of V corresponding to t. □

We demonstrate how this algorithm works through a small example. Let $I = \langle (X_1 - 1)(X_2 - 2), (X_1 - 3)(X_2 - 4)\rangle \subset GF(101)[X_1, X_2]$, then clearly $V(I) = \{(1, 4), (3, 2)\}$ and X_1 separates the points of V. We choose a random linear form

$$\ell : f \in I \mapsto \mathbb{N},\ \ell(f) = 17 f(1, 4) + 33 f(3, 2)$$

Then we have

$$\begin{aligned} \ell(X_1^i) &= 17 \cdot 1^i + 33 \cdot 3^i \\ \ell(X_2 X_1^i) &= 17 \cdot 4 \cdot 1^i + 33 \cdot 2 \cdot 3^i \end{aligned}$$

We define an infinite series for each sequence

$$\begin{aligned} Z_1 &= \sum_{i=0}^{\infty} \ell(X_1^i)/T^{i+1} = \frac{17}{T-1} + \frac{33}{T-3} = \frac{17(T-3) + 33(T-1)}{(T-1)(T-3)} \\ Z_2 &= \sum_{i=0}^{\infty} \ell(X_2 X_1^i)/T^{i+1} = \frac{17 \cdot 4}{T-1} + \frac{33 \cdot 2}{t-3} = \frac{17 \cdot 4(T-3) + 33 \cdot 2(T-1)}{(T-1)(T-3)} \end{aligned}$$

Z_1 and Z_2 have a common denominator $P = (t-1)(t-3)$, whose roots are the coordinates of X_1 in $V(I)$. Now, let

$$\begin{aligned} R_2 &= \frac{\Omega((\ell(X_2 X_1^i))_{i\geq 0}, P)}{\Omega((\ell(X_1^i))_{i\geq 0}, P)} \mod P \\ &= \frac{17 \cdot 4(T-3) + 33 \cdot 2(T-1)}{17(T-3) + 33(T-1)} \mod P \\ &= \frac{4(T-3) + 2(T-1)}{(T-3) + (T-1)} \mod P \end{aligned}$$

Now, $R_2(1) = 4$ and $R_2(3) = 2$ as needed.

In what follows, $\mathbb{K}$ is a field. Let I be an ideal in $\mathbb{K}[X_1, \ldots, X_n]$ and $Q = \mathbb{K}[X_1, \ldots, X_n]/I$ be the associated residue class ring. Suppose that $V = V(I)$ has dimension zero, and write it as $V = \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_d\}$, with all $\boldsymbol{\alpha}_i$'s in $\overline{\mathbb{K}}^n$, and $\boldsymbol{\alpha}_i = (\alpha_{i,1}, \ldots, \alpha_{i,n})$ for all i. We also let D be the dimension of Q, so that $d \leq D$, and *we assume that* $\mathrm{char}(\mathbb{K})$ *is greater than* D. In this section, we recall and generalize results from the appendix of [6], with the objective of computing a zero-dimensional parametrization of V.

The main novelty in our approach is to avoid generic coordinates as much as possible. The algorithm will decompose V into two parts: for the first part, we will be able to use X_1 as a separating element; the remaining points will be dealt with using a random linear form. Throughout, we rely only the following operations: evaluations of linear forms on successive powers of a given element in Q, say $1, t, t^2, \ldots$, and elementary operations on univariate polynomials.

The main algorithm is as follows. For the moment, we can only describe its main structure; the details of the subroutines are given in the next paragraphs.

Algorithm 2 Parametrization(ℓ, t)

Input:
- a linear form ℓ over Q
- $t = t_1X_1 + \cdots + t_nX_n$

Output:
- polynomials $(P, (V_1, \ldots, V_n))$

1. let $(F, (G_1, \ldots, G_n), X_1) = \mathsf{ParametrizationX}_1(\ell)$
2. let $\ell' = \mathsf{Update}(\ell, F, t)$
3. let $(Q, (W_1, \ldots, W_n), t) = \mathsf{ParametrizationGeneric}(\ell', t)$
4. let $(F^*, (G_1^*, \ldots, G_n^*), t) = \mathsf{ChangeCoordinate}(F, (G_1, \ldots, G_n), t)$
5. let $(P, (V_1, \ldots, V_n)) = \mathsf{Union}(F^*, (G_1^*, \ldots, G_n^*), Q, (W_1, \ldots, W_n))$
6. **return** $(P, (V_1, \ldots, V_n), t)$

The call to $\mathsf{ParametrizationX}_1(\ell)$ computes a zero-dimensional parametrization of a subset V' of V for which X_1 is a separating element, using values of the form $(\ell(X_1^s))_{s\geq 0}$. We then modify ℓ (which in effect removes from V the points we just found) and apply $\mathsf{ParametrizationGeneric}(\ell', t)$, to obtain a zero-dimensional parametrization of $V'' = V - V'$ using values of the form $(\ell'(t^s))_{s\geq 0}$. The last two steps involve changing coordinates in $(F, (G_1, \ldots, G_n)$ (to use t as a separating variable instead), and performing the union of the two components V' and V''.

3 Block Sparse-FGLM algorithm

In this section, we will show how to extend the Sparse-FGLM to using blocking methods. Steel's method also uses the Block Wiedemann algorithm to compute the minimal polynomial

of $\boldsymbol{M}_i$, for which the roots provide the appropriate values for X_i, but uses the "evaluation" method for the rest (another Gröbner Basis computation with that variable evaluated at each root of the minimal polynomial) [28]. Our algorithm computes the rest of the lex Gröbner basis directly.

Given:

$I \subset \mathbb{K}$:	zero dimensional ideal in shape position
$\mathbb{B} \subset \mathbb{K}[X_1, \ldots, X_n]/I$:	monomial basis of $\mathbb{K}[X_1, \ldots, X_n]/I$
D:	dimension of $\mathbb{B}$
$\boldsymbol{M}_1, \ldots, \boldsymbol{M}_n$:	multiplication matrices of $X_1, \ldots, X_n$ respectively
t:	random linear combination of X_i's
$\boldsymbol{M}$:	multiplication matrix of t

we compute a lex Gröbner basis that have the same points in its variety as the radical of I (note that we do not assume that I is radical). This is because we introduce another variable t which, generically, separates the points in the variety. We also assume that the base field $\mathbb{K}$ has characteristic larger than D. More precisely, we want to find polynomials $(R, R_1, \ldots, R_n)$ such that for all α that is a factor of R, $\{(R_1(\alpha), \ldots, R_n(\alpha))\} = V(I)$

The main idea is to compute the minimal polynomial of $\boldsymbol{M}$ by using the block Wiedemann algorithm. This is done by applying the Matrix Berlekamp-Massey algorithm of Section 1.2 on the sequence $S = (\boldsymbol{U}^\perp \boldsymbol{M}^i \boldsymbol{V})_{0 \le i < 2\lceil \frac{D}{m} \rceil}$, for some $m \in \mathbb{N}$ and $U, V \in \mathbb{K}^{D \times m}$, and taking the largest invariant factor. We find the rest of the polynomials in the output by extracting the scalar closed forms from the matrix closed forms. Lastly, we denote a matrix $\boldsymbol{N}$ with all its terms of negative exponents removed as $(\boldsymbol{N})_{T \ge 0}$

Algorithm 3 Block Sparse-FGLM$(\boldsymbol{M}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_n, m)$

Input:

- $\boldsymbol{M}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_n$ defined as above
- dimension of the blocks $m \in \mathbb{N}$

Output: polynomials $(R, R_1, \ldots, R_n)$

1. choose $\boldsymbol{U}, \boldsymbol{V} \in \mathbb{K}^{D \times m}$
2. $S = (\boldsymbol{U}^\perp \boldsymbol{M}^i \boldsymbol{V})_{0 \le i < 2d}$, with $d = \frac{D}{m}$
3. $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} = \mathsf{MatrixBerlekampMassey}(S)$
4. $\boldsymbol{N}^* = (\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \sum_{i=0}^{d-1} (\boldsymbol{U}^\perp \boldsymbol{M}^i e)/T^{i+1})_{T \ge 0}$
5. $P =$ largest invariant factor of $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$
6. $R = \mathsf{SquareFreePart}(P)$
7. $a = [0 \ \cdots \ 0 \ P](\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}})^{-1}$
8. $N = a\boldsymbol{N}^*$
9. for $j = 1 \ldots n$:
   - 9.1. $\boldsymbol{N}_j^* = (\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \sum_{i=0}^{d-1} (\boldsymbol{U}^\perp \boldsymbol{M}^i \boldsymbol{M}_j e)/T^{i+1})_{T \ge 0}$
   - 9.2. $N_j = a\boldsymbol{N}_j^*$
   - 9.3. $R_j = N_j/N \bmod R$

In lines 2,4, and 9.1 of the above algorithm, we use $(\boldsymbol{U}^\perp \boldsymbol{M}^i)_{i\geq 0}$, which we compute once in line 2 and store. Computing this sequence is the bottleneck of the algorithm, but this can be parallelized. Let $\boldsymbol{U} \in \mathbb{K}^{D\times m}$, then we can write $\boldsymbol{U}$ as $[u_1 u_2 \dots u_m]$ where $u_k \in \mathbb{K}^{D\times 1}$, $1 \leq k \leq m$, is a vector. Now,

$$\boldsymbol{U}^\perp \boldsymbol{M}^i = [u_1^\perp \dots u_m^\perp]\boldsymbol{M}^i = [u_1^\perp \boldsymbol{M}^i \dots u_m^\perp \boldsymbol{M}^i],$$

where each product $u_k^\perp \boldsymbol{M}^i$ is independent of others. Therefore, to generate the sequence $(\boldsymbol{U}^\perp \boldsymbol{M}^i)_{i\geq 0}$, we generate, in parallel, the sequences $(u_k^\perp \boldsymbol{M}^i)_{i\geq 0}$, $1 \leq k \leq m$, and concatenate them together as above. Let c be the sparsity of $\boldsymbol{M}$, then generating $(u_k^\perp \boldsymbol{M}^i)_{0\leq i<2d}$ costs $O(dcD^2) = O(cD^3/m)$ field operations. If we are able to compute m vector-matrix products in parallel at once, the total cost of generating $(\boldsymbol{U}^\perp \boldsymbol{M}^i)_{0\leq i<2d}$ is $O(cD^3/m)$. We can then compute $(\boldsymbol{U}^\perp \boldsymbol{M}^i \boldsymbol{V})_{0\leq i<2d}$ by a product

$$\begin{bmatrix} \boldsymbol{U}^\perp \\ \boldsymbol{U}^\perp \boldsymbol{M} \\ \boldsymbol{U}^\perp \boldsymbol{M}^2 \\ \vdots \\ \boldsymbol{U}^\perp \boldsymbol{M}^{2d} \end{bmatrix} \boldsymbol{V} = \begin{bmatrix} \boldsymbol{U}^\perp \boldsymbol{V} \\ \boldsymbol{U}^\perp \boldsymbol{M}\boldsymbol{V} \\ \boldsymbol{U}^\perp \boldsymbol{M}^2\boldsymbol{V} \\ \vdots \\ \boldsymbol{U}^\perp \boldsymbol{M}^{2d}\boldsymbol{V} \end{bmatrix}$$

in cost $O(m^{\omega-2}D^2)$.

Recall, from Section 1.2, that we can compute the minimum generating polynomial matrix in $O(m^\omega M(D/m)\log(D/m))$. The largest invariant factor P and the quotient a in lines 5 and 6 respectively by using the high-order lifting algorithm of [29, Algorithm 5]. By [29, Corollary 16], the cost of this is **todo:** $O(m^\omega M(D/m)\log(D/m)\log(m))$. Now, for any $v \in \mathbb{K}^{D\times 1}$, computing $(\boldsymbol{U}\boldsymbol{M}^i v)_{0\leq i<d}$ costs $O(D^2)$ by using the above method. Recall, from Section 1.1, that we can compute $\boldsymbol{N}_v^* = \boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \sum_{i=0}^{d-1} (\boldsymbol{U}^\perp \boldsymbol{M}^i v)/T^{i+1}$ by first doing a product $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \sum_{i=0}^{d-1} (\boldsymbol{U}^\perp \boldsymbol{M}^i v)T^i$. Since both $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ and $\sum_{i=0}^{d-1} (\boldsymbol{U}^\perp \boldsymbol{M}^i v)T^i$ have degree at most d, this product can be done in cost $O(m^2 M(D/m))$.

Now, $\boldsymbol{N}_v^*$ has degree less than D/m and a has degree at most D, so computing $a\boldsymbol{N}_v^*$ costs $O(m^2 M(D/m))$. Therefore, computing $N, N_1, \dots, N_j$ costs $O(n(m^2 M(D/m)))$. Since P, R, N, N_j, $1 \leq j \leq n$ have degree at most D, lines 6 and 9.3 costs $\tilde{O}(D)$. Finally, assuming perfect parallelization and $m << D$, we get the cost

$$O(cD^3/m + m^\omega M(D/m)\log(D/m)\log(m) + nm^2 M(D/m)),$$

3.1 Proof of Correctness

The largest invariant factor of $T\boldsymbol{I} - \boldsymbol{M}$ is the minimal polynomial of $\boldsymbol{M}$. Therefore, the above theorem shows that if we choose the entries of $\boldsymbol{U}$ and $\boldsymbol{V}$ randomly, we will have that the largest invariant factor of $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ is the minimal polynomial of $\boldsymbol{M}$ with high probability.

Next, we prove that one can recover a scalar numerator through a matrix numerator. **todo:** Vincent: the next lemma is straightforward and doesn't deserve a statement/proof.

The inverse can be written $(\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}})^{-1} = \boldsymbol{A}\boldsymbol{D}^{-1}\boldsymbol{B}$ where $\boldsymbol{A}, \boldsymbol{B}$ are (unimodular) polynomial matrices, and $\boldsymbol{D}$ is the Smith form which is a diagonal of divisors of the largest invariant factor P. So there's nothing to prove.

Lemma 3.1. *Let $\boldsymbol{a}$ be defined as line 7 of Algorithm 3, then $\boldsymbol{a}$ has polynomial entries.*

Proof. Let $\mathscr{D} = \boldsymbol{A}\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}\boldsymbol{B}$ be the Smith normal form of $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ and $s_1, \cdots s_m$ be invariant factors of $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ such that $s_m | s_{m-1} | \cdots | s_1$. Let $[b_1, \cdots, b_m]$ be the last row of $\boldsymbol{B}$ and $w = [\frac{s_1 b_1}{s_m}, \frac{s_1 b_2}{s_{m-1}}, \cdots, \frac{s_1 b_{m-1}}{s_2}, b_m]$ (since $s_i | s_1$), then

$$\begin{aligned}(w\boldsymbol{A})\boldsymbol{A}^{-1}\mathscr{D} &= [\frac{s_1 b_1}{s_m}, \frac{s_1 b_2}{s_{m-1}}, \cdots, \frac{s_1 b_{m-1}}{s_2}, b_m] \begin{bmatrix} s_m & & \\ & \ddots & \\ & & s_1 \end{bmatrix} \\ &= [s_1 b_1, s_1 b_2, \cdots, s_1 b_m] \\ &= [0, \cdots, 0, s_1]\boldsymbol{B}\end{aligned}$$

Therefore, if $a = w\boldsymbol{A}$, we get $a\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} = (w\boldsymbol{A})\boldsymbol{A}^{-1}\mathscr{D}\boldsymbol{B}^{-1} = [0, \cdots, 0, s_1]$ as needed. Since both w and $\boldsymbol{A}$ have polynomial entries, a must also have polynomial entries.

□

Theorem 3.2. *Let $U = [u_1, u_2, \ldots, u_m]$ be in $\mathbb{K}^{D\times m}$, $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ be the minimum generator of $(\boldsymbol{U}^t\boldsymbol{M}^i\boldsymbol{V})_{i\geq 0}$, and P be the minimal polynomial of $\boldsymbol{M}$. For any $v \in \mathbb{K}^D$, if $\boldsymbol{N}^* = (\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \sum_{i=0}^{d-1}(\boldsymbol{U}^t\boldsymbol{M}^i v)/T^{i+1})_{T\geq 0}$ and $N = a\boldsymbol{N}^*$, then $N = \Omega((u_m^t\boldsymbol{M}^i v)_{i\geq 0}, P)$.*

Proof. Let $Z = \sum_{i=0}^{\infty}(\boldsymbol{U}^t\boldsymbol{M}^i v)/T^{i+1}$ and $\boldsymbol{N}^{*'} = \boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}Z$. Since the highest power of the entries in Z is T^{-1}, the entries of the product $\boldsymbol{N}^{*'}$ must have degree less than $d = deg(\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}})$. Furthermore, since $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ cancels the sequence $(\boldsymbol{U}\boldsymbol{M}^i\boldsymbol{V})_{i\geq 0}$, $\boldsymbol{N}^{*'}$ is a polynomial matrix and does not have any terms of negative exponents. This means that all terms in Z with degree less than T^{-d} must vanish in the product. Therefore,

$$\boldsymbol{N}^* = \boldsymbol{N}^{*'}$$

Now, rewrite $\boldsymbol{U}$ as $\boldsymbol{U} = [u_1 u_2 \cdots u_m]$, then

$$\boldsymbol{N}^* = \boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \sum_{i=0}^{\infty} \boldsymbol{U}^{\perp}\boldsymbol{M}^i v/T^{i+1} = \begin{bmatrix} \sum_{i\geq 0} u_1^{\perp}\boldsymbol{M}^i v/T^{i+1} \\ \vdots \\ \sum_{i\geq 0} u_m^{\perp}\boldsymbol{M}^i v/T^{i+1} \end{bmatrix}$$

Recall Ω from Definition 1.1. By rewriting each $\sum_{i\geq 0} u_j\boldsymbol{M}^i v$ in its closed form, we get

$$\boldsymbol{N}^* = \boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \begin{bmatrix} \Omega((u_1^{\perp}\boldsymbol{M}^i v)_{i\geq 0}, P)/P \\ \vdots \\ \Omega((u_m^{\perp}\boldsymbol{M}^i v)_{i\geq 0}, P)/P \end{bmatrix}$$

By Theorem 1.11, the i^{th} invariant factor of $T\boldsymbol{I} - \boldsymbol{M}$ is equal to the i^{th} invariant factor of $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ for generic choice of $\boldsymbol{U}, \boldsymbol{V}$. Thus, $s_1 = P$ and by Lemma 3.1

$$\begin{aligned} a\boldsymbol{N}^* &= a\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \begin{bmatrix} \Omega((u_1^\perp \boldsymbol{M}^i v)_{i\geq 0}, P)/P \\ \vdots \\ \Omega((u_m^\perp \boldsymbol{M}^i v)_{i\geq 0}, P)/P \end{bmatrix} \\ &= [0, \cdots, 0, P] \begin{bmatrix} \Omega((u_1^\perp \boldsymbol{M}^i v)_{i\geq 0}, P)/P \\ \vdots \\ \Omega((u_m^\perp \boldsymbol{M}^i v)_{i\geq 0}, P)/P \end{bmatrix} \\ &= \Omega((u_m^\perp \boldsymbol{M}^i v)_{i\geq 0}, P) \end{aligned}$$

Therefore, $N = \Omega((u_m^\perp \boldsymbol{M}^i v)_{i\geq 0}, P)$ as needed. □

Finally, we conclude by seeing what happens when we pick specific values for v and applying Theorem 3.2. In line 4 of Algorithm 3, we have $v = e$, where e is the coordinate vector for $1 \in \mathbb{B}$. By applying Theorem 3.2, we have that

$$\boldsymbol{N}^* = \boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \sum_{i\geq 0} \boldsymbol{U}^t \boldsymbol{M} e/T^{i+1},$$

and

$$N = a\boldsymbol{N}^* = \Omega((u_m^\perp \boldsymbol{M}^i e)_{i\geq 0}, P).$$

By construction, $\boldsymbol{M}^i e$ gives the coordinate vector of t^i in $\mathbb{B}$, so

$$\sum_{i\geq 0} u_m^\perp \boldsymbol{M}^i e/T^{i+1} = \sum_{i\geq 0} \ell_{u_m}(t^i)/T^{i+1}$$

where ℓ_{u_m} is the linear form associated with u_m. Therefore,

$$N = \Omega((u_m^\perp \boldsymbol{M}^i e)_{i\geq 0}, P) = \Omega((\ell_{u_m}(t^i))_{i\geq 0}, P)$$

.

Similarly, in line 7.1 of Algorithm 3, we compute $\boldsymbol{N}_j^* = \boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} \sum_{i\geq 0} \boldsymbol{U}^\perp \boldsymbol{M} \boldsymbol{M}_j e/T^{i+1}$. By Theorem 3.2,

$$N_j = a\boldsymbol{N}_j^* = \Omega((u_m^\perp \boldsymbol{M}^i \boldsymbol{M}_j e)_{i\geq 0}, P)$$

Again, by construction, $\boldsymbol{M}_j e$ is the coordinate vector for X_j in $\mathbb{B}$ and $\boldsymbol{M}^i \boldsymbol{M}_j e$ is the coordinate vector for $X_j t^i$ in $\mathbb{B}$. Therefore,

$$N_j = \Omega((u_m^\perp \boldsymbol{M}^i \boldsymbol{T}_j e)_{i\geq 0}, P) = \Omega((\ell_{u_m}(X_j t^i))_{i\geq 0}, P)$$

By the correctness of Algorithm 1, we have the proof of correctness for the Block Sparse-FGLM algorithm.

3.2 Example

First, we give an example in the radical case. Let

$$I = \langle -16X_1^2 - 15X_1X_2 - 14X_2^2 - 48X_1 + 26, 35X_1X_2 + 47X_1 - 46X_2 - 47\rangle \subset GF(101)[X_1, X_2]$$

We choose $t = 2X_1 + 53X_2$, with multiplication matrices:

$$\boldsymbol{M}_1 = \begin{bmatrix} 85 & 0 & 37 & 0 \\ 69 & 85 & 15 & 0 \\ 100 & 91 & 19 & 1 \\ 1 & 10 & 68 & 0 \end{bmatrix} \quad \boldsymbol{M}_2 = \begin{bmatrix} 36 & 1 & 0 & 0 \\ 42 & 0 & 85 & 1 \\ 51 & 0 & 91 & 0 \\ 95 & 0 & 10 & 0 \end{bmatrix} \quad \boldsymbol{M} = \begin{bmatrix} 58 & 53 & 74 & 0 \\ 41 & 69 & 91 & 53 \\ 75 & 81 & 13 & 2 \\ 88 & 20 & 60 & 0 \end{bmatrix}$$

We choose $m = 2$ and choose $U, V \in GF(101)^{D\times m}$ of random entries:

$$\boldsymbol{U}^\perp = \begin{bmatrix} 84 & 38 \\ 29 & 58 \\ 80 & 43 \\ 7 & 82 \end{bmatrix} \quad \boldsymbol{V} = \begin{bmatrix} 6 & 97 \\ 83 & 58 \\ 101 & 95 \\ 59 & 89 \end{bmatrix}$$

We compute the matrix sequence $S = (\boldsymbol{U}^\perp \boldsymbol{M}^i \boldsymbol{V})_{i\geq 0}$ and its minimum generating matrix polynomial $\boldsymbol{G}$

$$S = (\begin{bmatrix} 92 & 75 \\ 83 & 51 \end{bmatrix}, \begin{bmatrix} 57 & 82 \\ 23 & 16 \end{bmatrix}, \begin{bmatrix} 54 & 93 \\ 70 & 66 \end{bmatrix}, \begin{bmatrix} 50 & 77 \\ 26 & 76 \end{bmatrix})$$

$$\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}} = \begin{bmatrix} T^2 + 76T + 8 & 87T + 31 \\ 100T + 46 & T^2 + 87T + 44 \end{bmatrix}$$

The biggest invariant factor of $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ is

$$P = T^4 + 62T^3 + 85T^2 + 69T + 37 = R$$

since P is square free. Now, we compute $\boldsymbol{N}^*$ and a:

$$\begin{aligned} \boldsymbol{N}^* &= [84T + 46, 38T + 65] \\ a &= [T + 55, T^2 + 76T + 8] \end{aligned}$$

Finally, we find the scalar numerator $N = a\boldsymbol{N}^*$:

$$N = 100T^3 + 26T^2 + 33T + 18$$

To get $R_1(T)$, we compute $\boldsymbol{N}_1^*$ and $N_1 = a\boldsymbol{N}_1^*$:

$$\begin{aligned} \boldsymbol{N}_1^* &= [79T + 8, 100T + 23] \\ N_1 &= 100T^3 + 26T^2 + 33T + 18 \end{aligned}$$

Lastly,

$$\begin{aligned} R_1(T) &= N_1/N \mod R \\ &= 61T^3 + 75T^2 + 85T + 23 \end{aligned}$$

We compute $R_2(t) = 32T^3 + 41T^2 + 94T + 22$ in the same way. As a sanity check, we see that $V(I)$ has one point in $GF(101)^2$: $(54, 79)$ and P has one factor in $GF(101)$: 53. Now, $R_1(53) = 54$ and $R_2(53) = 79$ as expected.

4 Experimental Results

name	n	**D**	**Sparsity**	**m = 1**	**m = 4**	**m = 8**	**deg(P) = D**
rand(3,10)	3	1000	0.06	16.550	4.45	3.305	yes
rand(3,12)	3	1728	0.05	70.703	18.74	13.250	yes
rand(3,14)	3	2744	0.05	240.992	64.619	44.737	yes
rand(3,16)	3	4096	0.04	697.91	191.327	129.598	yes
rand(3,18)	3	5832	0.03	1779.89	473.091	329.231	yes
rand(3,20)	3	8000	0.03	4128.45	1099.08	759.426	yes
bannwarth	5	284	0.27	1.027	0.364	0.220	yes
cyclic7	7	924	0.02	7.038	1.954	1.448	yes
eco12	12	1024	0.13	62.294	16.427	11.539	yes
katsura8	9	256	0.28	1.207	0.338	0.249	yes
katsura9	10	512	0.26	9.210	2.453	1.865	yes
katsura10	11	1024	0.25	71.243	18.867	13.218	yes
sot1	5	8694	0.001	762.063	208.25	143.281	yes
vor2	6	574	0.24	10.238	2.749	1.972	yes

5 Improvements for Special cases

5.1 Choice of the blocking size

In general, experiments show that choosing $1 \leq m \leq D$ that matches the number of cores is optimal in Section 4. However, when $m = 1$, blocking has no effect and there is no parallelization to be had. In particular, the minimum generating polynomial matrix $\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}}$ is equal to the minimum polynomial P of $\boldsymbol{M}$. Therefore, $a = P(\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}})^{-1} = 1$ and we can omit steps 7,8, and 9.2 in Algorithm 3.

If the multiplication matrices are dense, then it is better to compute the product $\boldsymbol{U}\boldsymbol{M}^i$ using a single matrix multiplication, rather than in parallel. When $m = D$, $\boldsymbol{U}, \boldsymbol{V} \in \mathbb{K}^{D\times D}$ are square matrices and we can write $\boldsymbol{U}\boldsymbol{M}\boldsymbol{V} = \boldsymbol{U}'\boldsymbol{M}$; therefore, we can generate sequences of the form $(\boldsymbol{U}\boldsymbol{M}^i)_{i\geq 0}$ rather than $(\boldsymbol{U}\boldsymbol{M}^i V)_{i\geq 0}$. Now, $d = D/m = 1$ and we generate the sequence $S = (\boldsymbol{U}, \boldsymbol{U}\boldsymbol{M})$. The minimum generating polynomial matrix of S is

$$\boldsymbol{F}^{U,M} = T - \boldsymbol{U}\boldsymbol{M}\boldsymbol{U}^{-1},$$

and

$$((T - \boldsymbol{U}\boldsymbol{M}\boldsymbol{U}^{-1})(\boldsymbol{U}/T + \boldsymbol{U}\boldsymbol{M}/T^2))_{T\geq 0} = \boldsymbol{U}.$$

For $\boldsymbol{M}_j$, $1 \leq j \leq n$, we get the sequence $(\boldsymbol{U}\boldsymbol{M}_j, \boldsymbol{U}\boldsymbol{M}\boldsymbol{M}_j)$ and

$$((T - \boldsymbol{U}\boldsymbol{M}\boldsymbol{U}^{-1})(\boldsymbol{U}\boldsymbol{M}_j/T + \boldsymbol{U}\boldsymbol{M}\boldsymbol{M}_j/T))_{T\geq 0} = \boldsymbol{U}\boldsymbol{M}_j.$$

Thus, we get the following algorithm:

Algorithm 4 Dense Block Sparse-FGLM($\boldsymbol{M}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_n$)

Input:

- $\boldsymbol{M}, \boldsymbol{M}_1, \ldots, \boldsymbol{M}_n$ defined as above

Output: polynomials $(R, R_1, \ldots, R_n)$

1. choose $\boldsymbol{U} \in \mathbb{K}^{D\times D}$
2. $\boldsymbol{F}^{U,M} = T - \boldsymbol{U}\boldsymbol{M}\boldsymbol{U}^{-1}$
3. $\boldsymbol{N}^* = \boldsymbol{U}$
4. $P =$ largest invariant factor of $\boldsymbol{F}^{U,M}$
5. $R =$ SquareFreePart(P)
6. $a = [0\ \cdots\ 0\ P](\boldsymbol{F}^{\boldsymbol{U},\boldsymbol{M},\boldsymbol{V}})^{-1}$
7. $N = a\boldsymbol{N}^* e$
8. for j $= 1 \ldots n$:
  8.1. $\boldsymbol{N}_j^* = \boldsymbol{U}\boldsymbol{M}_j$
  8.2. $N_j = a\boldsymbol{N}_j^* e$
  8.3. $R_j = N_j/N \bmod R$
9. return $((R, R_1, \ldots, R_n), t)$

Let ω be an exponent such that matrices of size n can be multiplied in $O(n^\omega)$. In this case, computing $\boldsymbol{F}^{U,M}$ costs $O(D^\omega)$ and computing $N, N_1, \ldots, N_n$ costs $O(nD^2)$ since entries of a has roughly degree D. **todo:** We can solve for a by using the high-order lifting algorithm of [29, Algorithm 5]. By [29, Corollary 16], the cost of this is $O(D^\omega \log(D))$. Putting everything together, we get the final cost of $O(D^\omega \log(D) + nD^2)$.

5.2 Decomposition of linear forms.

In this paragraph, we work over the whole V (so $\mathfrak{S} = \{1, \ldots, d\}$). Specializing our previous discussion to the case $t = X_1$, we let $r_1, \ldots, r_c$ be the pairwise distinct values taken by X_1 on V, for some $c \leq d$. For $j = 1, \ldots, c$, we write T_j for the set of all indices i in $\{1, \ldots, d\}$ such that $\alpha_{i,1} = r_j$; the sets $T_1, \ldots, T_c$ form a partition of $\{1, \ldots, d\}$. When T_j has cardinality 1, we denote it as $T_j = \{\sigma_j\}$, for some index σ_j in $\{1, \ldots, d\}$, so that $\alpha_{\sigma_j,1} = r_j$.

For $i = 1, \ldots, d$, let us write ν_i for the degree of the minimal polynomial of X_1 in Q_i; thus, this polynomial is $(T - \alpha_{i,1})^{\nu_i}$. For j in $\{1, \ldots, c\}$, we define m_j as the maximum of all ν_i, for i in T_j. As a result, the minimal polynomial of X_1 in $\prod_{j \in T_j} Q_j$ is $(T - r_j)^{m_j}$, and the minimal polynomial of X_1 in Q is $M = \prod_{j \in \{1,\ldots,c\}} (T - r_j)^{m_j}$.

Recall that a linear form $\ell : Q \to \overline{\mathbb{K}}$ can be written uniquely as $\ell = \sum_{i \in \{1,\ldots,d\}} \ell_i$, with $\ell_i : Q_i \to \overline{\mathbb{K}}$; collecting terms, ℓ may also be written as $\ell = \sum_{j \in \{1,\ldots,c\}} \lambda_j$, with $\lambda_j = \sum_{i \in T_j} \ell_i$. Given such an ℓ, we first explain how to compute values of the form $\lambda_j(1)$. We will do this for some values of j only, namely those j for which $m_j = 1$.

Lemma 5.1. *Let ℓ be in $\hom_{\mathbb{K}}(Q, \mathbb{K})$, let M be the minimal polynomial of X_1 in Q and let δ be its degree. Then, the polynomial $\Omega((\ell(X_1^s))_{s \geq 0}, M)$ satisfies*

$$\Omega((\ell(X_1^s))_{s \geq 0}, M)(r_j) = \lambda_j(1) M'(r_j) \quad \text{for all } j \text{ such that } m_j = 1.$$

Proof. Let $\mathfrak{e}$ be the set of all indices j in $\{1, \ldots, c\}$ such that $m_j = 1$, and let $\mathfrak{f} = \{1, \ldots, c\} - \mathfrak{e}$; this definition allows us to split the sum as

$$\begin{aligned} \sum_{s \geq 0} \ell(X_1^s)/T^{s+1} &= \sum_{j \in \{1,\ldots,c\}} \sum_{i \in T_j} \sum_{s \geq 0} \ell_i(X_1^s)/T^{s+1} \\ &= \sum_{j \in \mathfrak{e}} \sum_{i \in T_j} \sum_{s \geq 0} \ell_i(X_1^s)/T^{s+1} + \sum_{j \in \mathfrak{f}} \sum_{i \in T_j} \sum_{s \geq 0} \ell_i(X_1^s)/T^{s+1}. \end{aligned}$$

Using Lemma 2.3 with $t = X_1$ and $v = 1$, any sum $\sum_{s \geq 0} \lambda_j(X_1^s)/T^{s+1}$ in the second summand can be rewritten as

$$\frac{C_j}{(T - r_j)^{v_j}},$$

for some integer v_j, and for some polynomial C_j of degree less than v_j. Next, take j in $\mathfrak{e}$. Since $m_j = 1$, $\nu_i = 1$ for all i in T_j, so that each such ℓ_i takes the form

$$\ell_i : f \mapsto (\Lambda_i(f))(\boldsymbol{\alpha}_i),$$

where Λ_i is a differential operator that does not involve $\partial/\partial X_1$. Since all terms of positive order in Λ_i involve one of $\partial/\partial X_2, \ldots, \partial/\partial X_n$, they cancel X_1^s for $s \geq 0$. Thus, $\ell_i(X_1^s)$ can be rewritten as $\ell_{i,1}\alpha_{i,1}^s$, for some constant $\ell_{i,1}$, and the generating series of these terms is

$$\frac{\ell_{i,1}}{T - \alpha_{i,1}} = \frac{\ell_{i,1}}{T - r_j}.$$

Remarking that we can write $\ell_{i,1} = \ell_i(1)$, altogether, the sum in question can be written

$$\begin{aligned}\sum_{s\geq 0} \ell(X_1^s)/T^{s+1} &= \sum_{j\in\mathfrak{e}} \frac{\sum_{i\in T_j} \ell_i(1)}{T - r_j} + \sum_{j\in\mathfrak{f}} \frac{D_j}{(T-r_j)^{x_j}} \\ &= \sum_{j\in\mathfrak{e}} \frac{\lambda_j(1)}{T - r_j} + \sum_{j\in\mathfrak{f}} \frac{D_j}{(T-r_j)^{x_j}}\end{aligned}$$

for some integers $\{x_j \mid j \in b\}$ such that $\deg(D_j) < x_j$ holds, and with D_j and $T - r_j$ coprime; if $r_j = 0$, we take $x_j = \deg(D_j) + 1$. In particular, the minimal polynomial of $(\ell(X_1^s))_{s\geq 0}$ is $N = \prod_{j\in\mathfrak{e}}(T - r_j)\prod_{j\in\mathfrak{f}}(T - r_j)^{x_j}$.

On the other hand, the minimal polynomial M of X_1 can be rewritten as $M = \prod_{j\in\mathfrak{e}}(T - r_j)\prod_{j\in\mathfrak{f}}(T - r_j)^{m_j}$, so that $\delta = \sum_{j\in\mathfrak{e}} 1 + \sum_{j\in\mathfrak{f}} m_j$. The minimal polynomial of the sequence $\ell(X_1^s)$ divides M, so that $x_j \leq m_j$ holds for all j in $\mathfrak{f}$. As a result, $\Omega((\ell(X_1^s))_{s\geq 0}, M)$ exists and is given by

$$\begin{aligned}\Omega((\ell(X_1^s))_{s\geq 0}, M) = &\sum_{j\in\mathfrak{e}} \Big(\lambda_j(1) \prod_{\iota\in\mathfrak{e}-\{j\}} (T - r_\iota)\Big)\Big(\prod_{j\in\mathfrak{f}} (T - r_j)^{m_j}\Big) \\ &+ \sum_{j\in\mathfrak{f}} \Big((T - r_j)^{m_j - x_j} D_j \prod_{\iota\in\mathfrak{f}-\{j\}} (T - r_j)^{m_\iota}\Big)\Big(\prod_{j\in\mathfrak{e}} (T - r_j)\Big).\end{aligned}$$

This implies that

$$\Omega((\ell(X_1^s))_{s\geq 0}, M)(r_k) = \lambda_k(1) \prod_{\iota\in\mathfrak{e}-\{k\}} (r_k - r_\iota) \prod_{j\in\mathfrak{f}} (r_k - r_j)^{m_j} = \lambda_k(1) M'(r_k)$$

holds for all k in $\mathfrak{e}$. □

We then show how to use this result to avoid (as much as possible) using a generic linear form $t = t_1X_1 + \cdots + t_nX_n$, and how to use (say) X_1 instead to compute a zero-dimensional parametrization of a subset of V; this is motivated by the fact that the multiplication matrix by X_1 is expected to be sparser than that of t (since the matrix of t is a combination of those of $X_1, \ldots, X_n$), sometimes by a substantial amount. Of course, there is no guarantee that X_1 is a separating element for V. As a result, we will compute a decomposition of V into two components V' and V''; X_1 will be a separating element for V', whereas we will use a generic linear form to describe V''.

More precisely, we characterize the set V' mentioned above as follows: for i in $\{1, \ldots, d\}$, $\boldsymbol{\alpha}_i$ is in V' if and only if:

- for i' in $\{1,\dots,d\}$, with $i' \neq i$, $\alpha_{i',1} \neq \alpha_{i,1}$;
- Q_i is a reduced algebra (equivalently, I_i is radical).

We denote by $\mathfrak{A} \subset \{1,\dots,d\}$ the set of corresponding indices i, and we let $\mathfrak{B} = \{1,\dots,d\} - \mathfrak{a}$, so that we have $V' = V_{\mathfrak{A}}$ and $V'' = V_{\mathfrak{B}}$. Remark that X_1 is a separating element for V'.

Correspondingly, we define $\mathfrak{a}$ as the set of all indices j in $\{1,\dots,c\}$ such that σ_j is in $\mathfrak{A}$. In other words, j is in $\mathfrak{a}$ if and only if T_j has cardinality 1 and Q_{σ_j} is reduced. The algorithm in this paragraph will compute a zero-dimensional parametrization of $V_{\mathfrak{A}}$; we use the following lemma to perform this decomposition of V.

Lemma 5.2. *Let j be in $\{1,\dots,c\}$ such that $m_j = 1$, let λ be a linear form over $\prod_{i\in T_j} Q_i$ and let $t = t_2X_2 + \cdots + t_nX_n$. Define constants a, b, c in $\overline{\mathbb{K}}$ by*

$$a = \lambda(1), \quad b = \lambda(t), \quad c = \lambda(t^2).$$

Then, j is in $\mathfrak{a}$ if and only if, for a generic choice of λ and t, $ac = b^2$.

Proof. The assumption that $m_j = 1$ means that for all i in T_j, $\nu_i = 1$. The linear form λ can be uniquely written as a sum $\lambda = \sum_{i\in T_j} \ell_i$, where each ℓ_i is in $\hom_{\overline{\mathbb{K}}}(Q_i, \overline{\mathbb{K}})$. The fact that all ν_i are equal to 1 then implies that each ℓ_i takes the form

$$\ell_i : f \mapsto (\Lambda_i(f))(\boldsymbol{\alpha}_i),$$

where Λ_i is a differential operator that does not involve $\partial/\partial X_1$. Thus, as in Eq. (5), we can write a general Λ_i of this form as

$$\Lambda_i : f \mapsto u_{i,1}f + \sum_{2\le r\le n} P_{i,r}(u_{i,2},\dots,u_{i,D_i})\frac{\partial}{\partial X_j}f + \sum_{2\le r\le s\le n} P_{i,r,s}(u_{i,2},\dots,u_{i,D_i})\frac{\partial^2}{\partial X_j \partial X_k}f + \tilde{\Lambda}_i(f),$$

where all terms in $\tilde{\Lambda}_i$ have order at least 3, $\boldsymbol{u}_i = (u_{i,1},\dots,u_{i,D_i})$ are parameters and $(P_{i,r})_{2\le r\le n}$ and $(P_{i,r,s})_{2\le r\le s\le n}$ are linear forms in $u_{i,2},\dots,u_{i,D_i}$. We obtain

$$\begin{aligned}
\Lambda_i(1) &= u_{i,1} \\
\Lambda_i(t) &= u_{i,1}t + \sum_{2\le r\le n} P_{i,r}t_r \\
\Lambda_i(t^2) &= u_{i,1}t^2 + 2t\sum_{2\le r\le n} P_{i,r}t_r + 2\sum_{2\le r\le s\le n} P_{i,r,s}t_rt_s,
\end{aligned}$$

which gives

$$\begin{aligned}
a &= \sum_{i\in T_j} u_{i,1} \\
b &= \sum_{i\in T_j} u_{i,1}t(\boldsymbol{\alpha}_i) + \sum_{i\in T_j, 2\le r\le n} P_{i,r}t_r \\
c &= \sum_{i\in T_j} u_{i,1}t(\boldsymbol{\alpha}_i)^2 + 2\sum_{i\in T_j, 2\le r\le n} t(\boldsymbol{\alpha}_i)P_{i,r}t_r + 2\sum_{i\in T_j, 2\le r\le s\le n} P_{i,r,s}t_rt_s.
\end{aligned}$$

Suppose first that j is in $\mathfrak{a}$. Then, $T_j = \{\sigma_j\}$, so we have only one term Λ_{σ_j} to consider, and Q_{σ_j} is reduced, so that all coefficients $P_{\sigma_j,r}$ and $P_{\sigma_j,r,s}$ vanish. Thus, we are left in this case with

$$a = u_{\sigma_j,1}, \quad b = u_{\sigma_j,1} t(\boldsymbol{\alpha}_{\sigma_j}), \quad c = u_{\sigma_j,1} t(\boldsymbol{\alpha}_{\sigma_j})^2,$$

so that we have $ac = b^2$, for *any* choice of λ and t. Now, we suppose that j is not in $\mathfrak{a}$, and we prove that for a generic choice of λ and t, $ac - b^2$ is non-zero. The quantity $ac - b^2$ is a polynomial in the coefficients $(\boldsymbol{u}_i)_{i \in T_j}$, and $(t_i)_{i \in \{2,\dots,n\}}$, and we have to show that it is not identically zero. We discuss two cases; in both of them, we prove that a suitable specialization of $ac - b^2$ is non-zero.

Suppose first that for at least one index σ in T_j, Q_σ is not reduced. In this case, there exists as least one index ρ in $\{2, \dots, n\}$ such that $P_{\sigma,\rho}(u_{\sigma,2}, \dots, u_{\sigma,D_\sigma})$ is not identically zero **todo:** explain better. Let us set all $\boldsymbol{u}_{\sigma'}$ to 0, for σ' in $T_j - \{\sigma\}$, as well as $u_{\sigma,1}$, and all t_r for $r \neq \rho$. Then, under this specialization, $ac - b^2$ becomes $-(P_{\sigma,\rho}(u_{\sigma,2}, \dots, u_{\sigma,D_\sigma}) t_\rho)^2$, which is non-zero, so that $ac - b^2$ itself must be non-zero.

Else, since j is not in $\mathfrak{a}$, we can assume that T_j has cardinality at least 2, with Q_σ reduced for all σ in T_j (so that $P_{\sigma,r}$ and $P_{\sigma,r,s}$ vanish for all such σ and all r, s). Suppose that σ and σ' are two indices in T_j; we set all indices $u_{\sigma'',1}$ to zero, for σ'' in $T_j - \{\sigma, \sigma'\}$. We are left with

$$a = u_{\sigma,1} + u_{\sigma',1}, \quad b = u_{\sigma,1} t(\boldsymbol{\alpha}_\sigma) + u_{\sigma',1} t(\boldsymbol{\alpha}_{\sigma'}), \quad c = u_{\sigma,1} t(\boldsymbol{\alpha}_\sigma)^2 + u_{\sigma',1} t(\boldsymbol{\alpha}_{\sigma'})^2.$$

Then, $ac - b^2$ is equal to $2u_{\sigma,1}u_{\sigma',1}(t(\boldsymbol{\alpha}_\sigma) - t(\boldsymbol{\alpha}_{\sigma'}))^2$, which is non-zero, since $\boldsymbol{\alpha}_\sigma \neq \boldsymbol{\alpha}_{\sigma'}$. □

The previous lemmas allow us to write Algorithm ParametrizationX$_1$. After computing M, we determine its factor $P = \prod_{j \in \{1,\dots,c\}, m_j = 1}(T - r_j)$. We split this polynomial further using the previous results in order to find $\prod_{j \in \mathfrak{a}}(T - r_j)$, and we conclude using the same kind of calculations as in ParametrizationGeneric.

Algorithm 5 ParametrizationX$_1(\ell, t)$

Input:
- a linear form ℓ over Q
- a linear form $t = t_2 X_2 + \cdots + t_n X_n$

Output: polynomials $((P, V_1, \dots, V_n), X_1)$

1. let M be the minimal polynomial of the sequence $(\ell(X_1^s))_{s \geq 0}$ and let δ be its degree
2. let $P = \prod_{r \text{ root of } M \text{ of multiplicity } 1}(T - r)$
3. let t be a random linear form in $X_2, \dots, X_n$
4. **for** $i = 0, 1, 2$ **do**
   (a) let $A_i = \Omega((\ell(t^i X_1^s))_{s \geq 0}, M)$
5. let $P = \gcd(P, A_0 A_2 - A_1^2)$
6. **for** $i = 2, \dots, n$ **do**
   (a) let $A_{X_i} = \Omega((\ell(X_i X_1^s))_{s \geq 0}, M)$
7. **return** $((P, T, A_{X_2}/A_1 \bmod P, \dots, A_{X_n}/A_1 \bmod P), X_1)$

Lemma 5.3. *Suppose that ℓ is a generic element of $\hom_{\mathbb{K}}(Q, \overline{\mathbb{K}})$ and that t is a generic linear form. Then the output $((P, V_1, \ldots, V_n), X_1)$ of $\mathsf{ParametrizationX}_1(\ell, t)$ is a zero-dimensional parametrization of $V_{\mathfrak{A}}$.*

Proof. Lemma 2.1 shows that for a generic choice of ℓ, M is the minimal polynomial of X_1, so that we indeed have $P = \prod_{j\in\{1,\ldots,c\}, m_j=1}(T - r_j)$. Let then r_j be one of these roots; by Lemma 5.1, for $i = 0, 1, 2$ we have $A_i(r_j) = M'(r_j)(t^i \cdot \lambda_j)(1)$, where $\lambda_j = \sum_{i\in T_j} \ell_i$, and the ℓ_i's are the components of ℓ.

As a result, the value of $A_0A_2 - A_1^2$ at r_j is (up to the non-zero factor $M'(r_j)^2$) equal to the quantity $ac - b^2$ defined in Lemma 5.2, so for a generic choice of ℓ and t, it vanishes if and only if j is in $\mathfrak{a}$. Thus, after Step 5, P is equal to $\prod_{j\in\mathfrak{a}}(T - r_j)$.

The last step is to compute the zero-dimensional parametrization of $V_{\mathfrak{A}}$. This is done using again Lemma 5.1. Indeed, for j in $\mathfrak{a}$, T_j is simply equal to $\{\sigma_j\}$, so that we have, for $i = 2, \ldots, n$,

$$A_1(r_j) = M'(r_j)\lambda_j(1) \quad \text{and} \quad A_{X_i}(r_j) = M'(r_j)(X_i \cdot \lambda_j)(1) = M'(r_j)\lambda_j(X_i).$$

Now, since j is in $\mathfrak{a}$, Q_{σ_j} is reduced, so that there exists a constant $\lambda_{j,1}$ such that for all f in $\overline{\mathbb{K}}[X_1, \ldots, X_n]$, $\lambda_j(f)$ takes the form $\lambda_{j,1}f(\boldsymbol{\alpha}_{\sigma_j})$. This shows that, as claimed,

$$\frac{A_{X_j}(r_j)}{A_1(r_j)} = \frac{\Omega((\ell(X_jX_1^s)_{s\geq 0}, M)(r_j)}{\Omega((\ell(tX_1^s))_{s\geq 0}, M)(r_j)} = \frac{M'(r_j)\lambda_{j,1}\alpha_{j,i}}{M'(r_j)\lambda_{j,1}} = \alpha_{j,i}.$$

For $i = 1$, since we use X_1 as a separating variable for $V_{\mathfrak{A}}$, we simply add the polynomial T to our list. □

References

[1] M. E. Alonso, E. Becker, M.-F. Roy, and T. Wörmann. Zeroes, multiplicities and idempotents for zerodimensional systems. In *MEGA'94*, volume 142 of *Progress in Mathematics*, pages 1–15. Birkhäuser, 1996.

[2] M. Bardet, J.-C. Faugère, and B. Salvy. On the complexity of the F5 Gröbner basis algorithm. *Journal of Symbolic Computation*, 70:49–70, 2015.

[3] E. Becker, T. Mora, M. Marinari, and C. Traverso. The shape of the Shape Lemma. In *ISSAC'94*, pages 129–133. ACM, 1994.

[4] E. Becker and T. Wörmann. Radical computations of zero-dimensional ideals and real root counting. *Mathematics and Computers in Simulation*, 42(4-6):561–569, 1996.

[5] B. Beckermann and G. Labahn. A uniform approach for the fast computation of matrix-type Padé approximants. *SIAM J. Matrix Anal. Appl.*, 15(3):804–823, 1994.

[6] A. Bostan, B. Salvy, and É. Schost. Fast algorithms for zero-dimensional polynomial systems using duality. *Appl. Algebra Engrg. Comm. Comput.*, 14:239–272, 2003.

[7] D. Coppersmith. Solving homogeneous linear equations over GF(2) via block Wiedemann algorithm. *Math. Comp.*, 62(205):333–350, 1994.

[8] J.-C. Faugère. A new efficient algorithm for computing Gröbner bases without reductions to zero (F5). In *ISSAC'02*, pages 75–83. ACM, 2002.

[9] J.-C. Faugère, P. Gaudry, L. Huot, and G. Renault. Polynomial Systems Solving by Fast Linear Algebra. https://hal.archives-ouvertes.fr/hal-00816724, 2013.

[10] J.-C. Faugère, P. Gaudry, L. Huot, and G. Renault. Sub-cubic change of ordering for Gröbner basis: a probabilistic approach. In *ISSAC'14*, pages 170–177. ACM, 2014.

[11] J.-C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *J. Symbolic Comput.*, 16(4):329–344, 1993.

[12] J.-C. Faugère and C. Mou. Sparse FGLM algorithms. *J. Symbolic Comput.*, 80(8):538–569, 2017.

[13] J.-C. Faugère, M. Safey El Din, and P.-J. Spaenlehauer. On the complexity of the generalized MinRank problem. *Journal of Symbolic Computation*, pages 30–58, 2013.

[14] J. Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, Cambridge, third edition, 2013.

[15] P. Gianni and T. Mora. Algebraic solution of systems of polynomial equations using Gröbner bases. In *AAECC'5*, volume 356 of *Lecture Notes in Comput. Sci.*, pages 247–257. Springer, 1989.

[16] P. Giorgi, C.-P. Jeannerod, and G. Villard. On the complexity of polynomial matrix computations. In *ISSAC'03*, pages 135–142. ACM, 2003.

[17] P. Giorgi and R. Lebreton. Online order basis algorithm and its impact on the block Wiedemann algorithm. In *ISSAC'14*, pages 202–209. ACM, 2014.

[18] C.-P. Jeannerod, V. Neiger, É. Schost, and G. Villard. Fast computation of minimal interpolation bases in Popov form for arbitrary shifts. In *ISSAC'16*, pages 295–302. ACM, 2016.

[19] T. Kailath. *Linear Systems*. Prentice-Hall, 1980.

[20] E. Kaltofen and G. Villard. On the complexity of computing determinants. In *ISSAC'01*, pages 13–27. ACM, 2001.

[21] E. Kaltofen and G. Villard. On the complexity of computing determinants. *Comput. Complexity*, 13(3-4):91–130, 2004.

[22] G. Moreno-Socías. *Autour de la fonction de Hilbert-Samuel (escaliers d'idéaux polynomiaux)*. PhD thesis, École polytechnique, 1991.

[23] B. Mourrain. Isolated points, duality and residues. *Journal of Pure and Applied Algebra*, 117/118:469–493, 1997. Algorithms for algebra (Eindhoven, 1996).

[24] V. Neiger. *Bases of relations in one or several variables: fast algorithms and applications*. PhD thesis, École Normale Supérieure de Lyon, November 2016.

[25] V. M. Popov. Invariant description of linear, time-invariant controllable systems. *SIAM Journal on Control*, 10(2):252–264, 1972.

[26] F. Rouillier. Solving zero-dimensional systems through the rational univariate representation. *Appl. Algebra Engrg. Comm. Comput.*, 9(5):433–461, 1999.

[27] S. Sakata. Extension of the Berlekamp-Massey algorithm to N dimensions. *Information and Computation*, 84(2):207–239, 1990.

[28] Allan Steel. Direct solution of the (11,9,8)-MinRank problem by the block Wiedemann algorithm in Magma with a Tesla GPU. In *PASCO'15*, pages 2–6. ACM, 2015.

[29] A. Storjohann. High-order lifting and integrality certification. *J. Symbolic Comput.*, 36:613–648, 2003.

[30] W. J. Turner. *Black box linear algebra with the LINBOX library*. PhD thesis, North Carolina State University, 2002.

[31] M. Van Barel and A. Bultheel. A general module theoretic framework for vector M-Padé and matrix rational interpolation. *Numer. Algorithms*, 3:451–462, 1992.

[32] G. Villard. Further analysis of Coppersmith's block Wiedemann algorithm for the solution of sparse linear systems. *ISSAC'97*, pages 32–39, 1997.

[33] G. Villard. A study of Coppersmith's block Wiedemann algorithm using matrix polynomials. Technical report, LMC-IMAG, Report 975 IM, 1997.

[34] W. A. Wolovich. *Linear Multivariable Systems*, volume 11 of *Applied Mathematical Sciences*. Springer-Verlag New-York, 1974.

[35] W. Zhou and G. Labahn. Efficient algorithms for order basis computation. *J. Symbolic Comput.*, 47(7):793–819, 2012.