Algorithms for zero-dimensional ideals using linear recurrent sequences

Vincent Neiger[1], Hamid Rahkooy[2], and Éric Schost[2]

[1] Department of Applied Mathematics and Computer Science, Technical University of Denmark.
[2] Cheriton School of Computer Science, University of Waterloo, Canada.

Abstract. Inspired by Faugère and Mou's sparse FGLM algorithm, we show how using linear recurrent multi-dimensional sequences can allow one to perform operations such as the primary decomposition of an ideal, by computing of the annihilator of one or several such sequences.

1 Introduction

In what follows, $\mathbb{K}$ is a perfect field. We consider the set $\mathscr{S} = \mathbb{K}^{\mathbb{N}^n}$ of n-dimensional sequences $\boldsymbol{u} = (u_m)_{m\in\mathbb{N}^n}$, and the polynomial ring $\mathbb{K}[X_1, \ldots, X_n]$, and we are interested in the following question. Let $I \subset \mathbb{K}[X_1, \ldots, X_n]$ be a zero-dimensional ideal. Given a monomial basis of $Q = \mathbb{K}[X_1, \ldots, X_n]/I$, together with the corresponding multiplication matrices $\mathsf{M}_1, \ldots, \mathsf{M}_n$, we want to compute the Gröbner bases, for a target order $>$, of pairwise coprime ideals $J_1, \ldots, J_K$ such that $I = \cap_{1\le k\le K} J_k$.

Faugère *et al.*'s paper [11] shows how to solve this question with $K = 1$ (so J_1 is simply I) in time $O(nD^3)$, where $D = \deg(I)$; here, the degree $\deg(I)$ is the $\mathbb{K}$-vector space dimension of Q. More recently, algorithms have been given with the cost bound $O\tilde{~}(nD^\omega)$ [9, 10, 20], where the notation $O\tilde{~}$ hides polylogarithmic factors, still with $K = 1$. The algorithms in this paper allow splittings (so $K > 1$ in general) and assume that $>$ is a lexicographic order.

To motivate our approach, assume that the algebraic set $V(I)$ is in *shape position*, that is, the coordinate X_n separates the points of $V(I)$. Then, the Shape Lemma [14] implies that the Gröbner basis of the radical $\sqrt{I}$ for the lexicographic order $X_1 > \cdots > X_n$ has the form $\langle X_1-G_1(X_n), \ldots, X_{n-1}-G_{n-1}(X_n), P(X_n)\rangle$, for some squarefree polynomial P, and some $G_1, \ldots, G_{n-1}$ of degrees less than $\deg(P)$. The polynomials P and $G_1, \ldots, G_{n-1}$ can be deduced from the values $(\ell(X_n^i))_{0\le i\le 2D}$ and $(\ell(X_j X_n^i))_{1\le j<n, 0\le i<D}$, for a randomly chosen linear form $\ell : Q \to \mathbb{K}$, in time $O\tilde{~}(D)$ [4]. The algorithms in the latter reference use baby steps / giant steps techniques for the calculation of the values of ℓ.

Similar ideas were developed in [12]; the algorithms in this reference make no assumption on I but may fail in some cases, then falling back on the FGLM algorithm. For instance, if I itself (rather than $\sqrt{I}$) is known to have a lexicographic Gröbner basis of the form $\langle X_1 - H_1(X_n), \ldots, X_{n-1} - H_{n-1}(X_n), Q(X_n)\rangle$, the algorithms in [12] recover this basis, also by considering values of linear forms

$\ell_i : Q \to \mathbb{K}$. A key remark made in that reference is that the values of the linear forms ℓ_i that we need can be computed efficiently by exploiting the sparsity of the multiplication matrices $\mathsf{M}_1, \ldots, \mathsf{M}_n$; this sparsity is then analyzed, assuming the validity of a conjecture due to Moreno-Socías [18]. These techniques are related as well to Rouillier's Rational Univariate Representation algorithm [21], which uses values of a specific linear form $Q \to \mathbb{K}$ called the *trace*. However, computing the trace (that is, its values on the monomial basis of Q) is non-trivial, and using random choices instead makes it possible to avoid this issue.

In this paper, we work in the continuation of [4]. Assuming $V(I)$ is in shape position, the results in that reference allow us to compute the Gröbner basis of $\sqrt{I}$, and our goal here is to recover Gröbner bases corresponding to a decomposition of I as stated above. Following [12, 1], we discuss the relation of this question to instances of the following problem: given sequences $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s$ in $\mathscr{S}$, find the Gröbner basis of their *annihilator* $\mathrm{ann}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s) \subset \mathbb{K}[X_1, \ldots, X_n]$, for a target order $>$. The annihilator, discussed in the next section, is a polynomial ideal corresponding to the linear relations which annihilate all sequences.

A direct approach to solve the FGLM problem using such techniques would be to pick initial conditions at random; knowing multiplication matrices modulo I allows us to compute the values of a sequence $\boldsymbol{u}$, for which I is contained in $\mathrm{ann}(\boldsymbol{u})$. If $I = \mathrm{ann}(\boldsymbol{u})$ holds, computing sufficiently many values of $\boldsymbol{u}$ and feeding them into an algorithm such as Sakata's [22] would solve our problem. This is often, but not always, possible: there exists a sequence $\boldsymbol{u}$ for which $I = \mathrm{ann}(\boldsymbol{u})$ if and only if $Q = \mathbb{K}[X_1, \ldots, X_n]/I$ is a *Gorenstein* ring, a notion going back to [16, 15] (see e.g. [5, Prop. 5.3] for a proof of the above assertion). This is for instance the case if I is a complete intersection, or if I is radical over a perfect field [8]; however, an ideal such as $I = \langle X_1^2, X_1X_2, X_2^2 \rangle \subset \mathbb{K}[X_1, X_2]$ is not Gorenstein.

To remedy this, we may have to use more than one sequence, so as to be able to recover I as $I = \mathrm{ann}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s)$. However, proceeding directly in this manner, we do not expect the algorithm to be significantly better than applying directly the FGLM algorithm (the techniques we will use for computing annihilators follow essentially the same lines as the FGLM algorithm itself). We will see that starting from the Gröbner basis of $\sqrt{I}$, we will be able to decompose I into e.g. primary components (assuming we allow the use of factorization algorithms over $\mathbb{K}$), and that our approach is expected to be competitive in those cases where the multiple components of I have low degrees.

Acknowledgements. We thank the reviewers for their remarks and suggestions. The third author is supported by an NSERC Discovery Grant.

2 Generalities on sequences and their annihilators

Define the shift operators $s_1, \ldots, s_n$ on $\mathscr{S}$ in the obvious manner, by setting $s_i(\boldsymbol{u}) = (u_{m+e_i})_{m \in \mathbb{N}^n}$, where $e_1, \ldots, e_n$ are the unit vectors. This makes $\mathscr{S}$ a $\mathbb{K}[X_1, \ldots, X_n]$-module, by setting $f \cdot \boldsymbol{u} = f(s_1, \ldots, s_n)(\boldsymbol{u})$. For $f = \sum_m f_m \boldsymbol{X}^m$, the entries of $f \cdot \boldsymbol{u}$ are thus $(\langle \boldsymbol{u} \mid \boldsymbol{X}^m f \rangle)_{m \in \mathbb{N}^n}$, where we write $\boldsymbol{X}^m = X_1^{m_1} \cdots X_n^{m_n}$ and $\langle \boldsymbol{u} \mid f \rangle = \sum_{m'} f_{m'} u_{m'}$. To a sequence $\boldsymbol{u} = (u_m)_{m \in \mathbb{N}^n}$ in $\mathscr{S}$, we can then associate its *annihilator* $\mathrm{ann}(\boldsymbol{u})$, defined as the ideal of all polynomials f in $\mathbb{K}[X_1,\dots,X_n]$ such that $f\cdot\boldsymbol{u}=0$. If we consider several sequences $\boldsymbol{u}_1,\dots,\boldsymbol{u}_s$ in $\mathscr{S}$, we then define $\mathrm{ann}(\boldsymbol{u}_1,\dots,\boldsymbol{u}_s)=\mathrm{ann}(\boldsymbol{u}_1)\cap\cdots\cap\mathrm{ann}(\boldsymbol{u}_s)$.

We will also occasionally discuss *kernels* of sequences. For $\boldsymbol{u}\in\mathscr{S}$, the kernel $\ker(\boldsymbol{u})$ is the $\mathbb{K}$-vector space formed by all polynomials f in $\mathbb{K}[X_1,\dots,X_n]$ such that $\langle\boldsymbol{u}\mid f\rangle=0$; this is not an ideal in general. If we consider several sequences $\boldsymbol{u}_1,\dots,\boldsymbol{u}_s$, we will write $\ker(\boldsymbol{u}_1,\dots,\boldsymbol{u}_s)=\ker(\boldsymbol{u}_1)\cap\cdots\cap\ker(\boldsymbol{u}_s)$.

Let I be a zero-dimensional ideal in $\mathbb{K}[X_1,\dots,X_n]$. Define the residue class ring $Q=\mathbb{K}[X_1,\dots,X_n]/I$ and let $D=\deg(I)=\dim_{\mathbb{K}}(Q)$. Consider also the dual $Q^*=\hom_{\mathbb{K}}(Q,\mathbb{K})$. To a linear form ℓ in Q^*, we associate the sequence $\boldsymbol{u}_\ell$ defined by $\boldsymbol{u}_\ell=(\ell(\boldsymbol{X}^m \bmod I))_{m\in\mathbb{N}^n}$.

For any linear form ℓ on Q, and any g in Q, define the linear form $g\cdot\ell\in Q^*$ by $(g\cdot\ell)(h)=\ell(gh)$. This induces a Q-module structure on Q^*, and we remark that we have the equality $g\cdot\boldsymbol{u}_\ell=\boldsymbol{u}_{(g \bmod I)\cdot\ell}$ for any g in $\mathbb{K}[X_1,\dots,X_n]$. Following [23] (where it is described with $n=1$), we call this operation *transposed product*.

For ℓ in Q^*, we can then define $\mathrm{ann}_Q(\ell)$ as the set of all g in Q such that $g\cdot\ell=0$; this is an ideal of Q. The following lemma clarifies the relation between $\mathrm{ann}(\boldsymbol{u}_\ell)\subset\mathbb{K}[X_1,\dots,X_n]$ and $\mathrm{ann}_Q(\ell)\subset Q$; it implies that $\mathrm{ann}(\boldsymbol{u}_\ell)$ is generated by I and any element of $\mathrm{ann}_Q(\ell)$ lifted to $\mathbb{K}[X_1,\dots,X_n]$.

Lemma 1 *With notation as above, for f in $\mathbb{K}[X_1,\dots,X_n]$, f is in $\mathrm{ann}(\boldsymbol{u}_\ell)$ if and only if $f \bmod I$ is in $\mathrm{ann}_Q(\ell)$.*

Proof. Take f in $\mathbb{K}[X_1,\dots,X_n]$. Then f is in $\mathrm{ann}(\boldsymbol{u}_\ell)$ if and only if $f\cdot\boldsymbol{u}_\ell=0$, that is, if and only if $\boldsymbol{u}_{(f \bmod I)\cdot\ell}=0$, if and only if $(f \bmod I)\cdot\ell$ itself is zero. □

When Q^* is a free Q-module of rank one, we say that Q is a Gorenstein ring, and that I is Gorenstein. In this case, there exists a linear form λ such that $Q^*=Q\cdot\lambda$; by the previous lemma, $\mathrm{ann}(\boldsymbol{u}_\lambda)=I$. Conversely, if $\mathrm{ann}(\boldsymbol{u}_\lambda)=I$, $\mathrm{ann}_Q(\lambda)=\{0\}$, so that $Q^*=Q\cdot\lambda$ (and Q^* is free of rank one). For instance, it is known that if I is radical, or I a complete intersection, then I is Gorenstein. On the other hand, if $I=\langle X_1^2,X_1X_2,X_2^2\rangle$, the inclusion $I\subset\mathrm{ann}(\boldsymbol{u}_\ell)$ is strict for any linear form ℓ. Using several sequences, we can however always recover I.

Lemma 2 *Let $\ell_1,\dots,\ell_D$ be linearly independent in Q^*, and let $\boldsymbol{u}_1,\dots,\boldsymbol{u}_D$ be the corresponding sequences. Then $\mathrm{ann}(\boldsymbol{u}_1,\dots,\boldsymbol{u}_D)=\ker(\boldsymbol{u}_1,\dots,\boldsymbol{u}_D)=I$.*

Proof. Note first that the inclusion $I\subset\mathrm{ann}(\boldsymbol{u}_1,\dots,\boldsymbol{u}_D)=\mathrm{ann}(\boldsymbol{u}_1)\cap\cdots\cap\mathrm{ann}(\boldsymbol{u}_D)$ is a direct consequence of Lemma 1, and that $\mathrm{ann}(\boldsymbol{u}_1,\dots,\boldsymbol{u}_D)$ is contained in $\ker(\boldsymbol{u}_1,\dots,\boldsymbol{u}_D)$. For the converse, let $\omega_1,\dots,\omega_D$ be the basis of Q dual to $\ell_1,\dots,\ell_D$. Suppose that f is in $\ker(\boldsymbol{u}_1,\dots,\boldsymbol{u}_D)$, and assume without loss of generality that f has been reduced by I, so that f is a linear combination of the form $f_1\omega_1+\cdots+f_D\omega_D$. Fix i in $1,\dots,D$ and apply ℓ_i to f; we obtain f_i. On the other hand, because f is in $\ker(\boldsymbol{u}_i)$, $\ell_i(f)$ must vanish. So we are done. □

We may however need less than D linear forms, as explained in the following discussion, which generalizes the comments we made in the Gorenstein case.

Let $B = (b_1, \ldots, b_D)$ be a monomial basis of Q. Given a linear form ℓ in Q^*, we define K_ℓ as the $D \times D$ matrix whose (i, j)th entry is $\ell(b_i b_j)$; this is the matrix of the mapping $f \in Q \mapsto f \cdot \ell \in Q^*$, so that its nullspace is $\mathrm{ann}_Q(\ell)$. More generally, given a positive integer s and linear forms $\ell_1, \ldots, \ell_s$, we define $K_{\ell_1,\ldots,\ell_s}$ as the $D \times sD$ matrix obtained as the concatenation of $K_{\ell_1}, \ldots, K_{\ell_s}$; this is the matrix of the mapping $(f_1, \ldots, f_s) \in Q^s \mapsto f_1 \cdot \ell_1 + \cdots + f_s \cdot \ell_s \in Q^*$.

Lemma 3 *For any linear forms $(\ell_1, \ldots, \ell_s)$, with all ℓ_i in Q^*, $\mathrm{ann}(\boldsymbol{u}_{\ell_1}, \ldots, \boldsymbol{u}_{\ell_s}) = I$ if and only if $(\ell_1, \ldots, \ell_s)$ are Q-module generators of Q^*.*

Proof. $(\ell_1, \ldots, \ell_s)$ are Q-module generators of Q^* if and only if $K_{\ell_1,\ldots,\ell_s}$ has rank D, if and only if $K^\perp_{\ell_1,\ldots,\ell_s}$ has a trivial nullspace. The nullspace of this matrix is the intersection of those of the matrices $K^\perp_{\ell_1}, \ldots, K^\perp_{\ell_s}$. All these matrices are symmetric, and we saw that for all i, the nullspace of $K^\perp_{\ell_i} = K_{\ell_i}$ is $\mathrm{ann}_Q(\ell_i)$; thus, the condition above is equivalent to $\mathrm{ann}_Q(\ell_1) \cap \cdots \cap \mathrm{ann}_Q(\ell_s) = \{0\}$. Lemma 1 shows that this is the case if and only if $\mathrm{ann}(\boldsymbol{u}_{\ell_1}) \cap \cdots \cap \mathrm{ann}(\boldsymbol{u}_{\ell_s}) = I$. □

Proposition 1 *There exists a unique integer $\tau \leq D$ such that for a generic choice of linear forms $(\ell_1, \ldots, \ell_\tau)$, with all ℓ_i in Q^*, the sequence of ideals $(\mathrm{ann}(\boldsymbol{u}_{\ell_1}, \ldots, \boldsymbol{u}_{\ell_t}))_{1 \leq t \leq \tau}$ is strictly decreasing, with $\mathrm{ann}(\boldsymbol{u}_{\ell_1}, \ldots, \boldsymbol{u}_{\ell_\tau}) = I$.*

Proof. Remark first that if τ exists with the properties above, it is necessarily unique. Let $(L_{1,1}, \ldots, L_{1,D}), \ldots, (L_{D,1}, \ldots, L_{D,D})$ be new indeterminates, let $\mathbb{L} = \mathbb{K}(L_{1,1}, \ldots, L_{D,D})$ and define the matrices $K_{L_1}, \ldots, K_{L_D}$ as follows. Let $Q_\mathbb{L} = Q \otimes_\mathbb{K} \mathbb{L}$; this allows us to define the linear forms $L_1, \ldots, L_D$ in $Q^*_\mathbb{L}$ by $L_t(b_j) = L_{t,j}$, for $1 \leq t \leq D$; then K_{L_t} is the matrix with entries $L_t(b_i b_j)$. The entries of K_{L_t} are linear forms in $L_{t,1}, \ldots, L_{t,D}$.

Define $K_{L_1,\ldots,L_t}$ as we did for $K_{\ell_1,\ldots,\ell_t}$. Then, for any linear forms $\ell_1, \ldots, \ell_t$ in Q^*, the matrix $K_{\ell_1,\ldots,\ell_t}$ is obtained by evaluating $K_{L_1,\ldots,L_t}$ at $L_{t,j} = \ell_t(b_j)$, for all t, j. The rank of $K_{\ell_1,\ldots,\ell_t}$ (over $\mathbb{K}$) is at most that of $K_{L_1,\ldots,L_t}$ (over $\mathbb{L}$).

We can then let τ be the smallest integer such that the matrix $K_{L_1,\ldots,L_\tau}$ has full rank D. Such an index exists, and is at most D, since by Lemma 2 (and by the remarks of the above paragraph) $K_{L_1,\ldots,L_D}$ has rank D.

Let $\ell_1, \ldots, \ell_\tau$ be such that $K_{\ell_1,\ldots,\ell_\tau}$ has rank D (this is our genericity condition); in this case, by the previous lemma, $\mathrm{ann}(\boldsymbol{u}_{\ell_1}, \ldots, \boldsymbol{u}_{\ell_\tau}) = I$. To conclude, it suffices to prove that the sequence of ideals $(\mathrm{ann}(\boldsymbol{u}_{\ell_1}, \cdots, \boldsymbol{u}_{\ell_t}))_{1 \leq t \leq \tau}$ is strictly decreasing. Suppose it is not the case, so that $\mathrm{ann}(\boldsymbol{u}_{\ell_1}, \ldots, \boldsymbol{u}_{\ell_t}) = \mathrm{ann}(\boldsymbol{u}_{\ell_1}, \ldots, \boldsymbol{u}_{\ell_{t+1}})$ for some $t < \tau$. Then, $\mathrm{ann}(\boldsymbol{u}_{\ell_1}, \ldots, \boldsymbol{u}_{\ell_t}, \boldsymbol{u}_{\ell_{t+2}}, \ldots, \boldsymbol{u}_{\ell_\tau}) = I$. Let us define $\ell'_1 = \ell_1, \ldots, \ell'_t = \ell_t, \ell'_{t+1} = \ell_{t+2}, \ldots, \ell'_{\tau-1} = \ell_\tau$. Then, we have $\mathrm{ann}(\boldsymbol{u}_{\ell'_1}, \ldots, \boldsymbol{u}_{\ell'_{\tau-1}}) = I$, so that $K_{\ell'_1,\ldots,\ell'_{\tau-1}}$ has rank D. This in turn implies (by the discussion above) that $K_{L_1,\ldots,L_{\tau-1}}$ has rank D, a contradiction. □

If Q is a local algebra with maximal ideal $\mathfrak{m}$, we can define the *socle* of Q as the $\mathbb{K}$-vector space of all elements f in Q such that $\mathfrak{m} f = 0$. For instance, if Q is local, the integer τ in the previous lemma is the dimension of the socle of Q. (we omit the proof, since we will not use this result in the rest of the paper).

3 Computing annihilators of sequences

Consider sequences $(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t)$ with $\boldsymbol{u}_i \in \mathscr{S}$ for all i, let J be the annihilator $\mathrm{ann}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t) \subset \mathbb{K}[X_1, \ldots, X_n]$, and suppose that it has dimension zero; our goal is to compute a Gröbner basis of it. We first review an algorithm due to Marinari, Möller and Mora [17], then introduce a modification of it that relaxes some of its assumptions. As a result, the algorithms in this section work under slightly different assumptions, and feature slightly different runtimes.

An algorithm with cost $(nt\deg(J))^{O(1)}$ would be highly desirable, but we are not aware of any such result. Most approaches (ours as well) involve reading a number of values of $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t$ and looking for dependencies between the columns of what is often called a generalized Hankel matrix, built using these values; the delicate question is how to control the size of the matrix.

Consider for instance the case $t = 1$, $\langle \boldsymbol{u}_1 \mid X_1^{m_1} \cdots X_n^{m_n} \rangle = 1$ for $m_1 + \cdots + m_n < \delta$ and $\langle \boldsymbol{u}_1 \mid X_1^{m_1} \cdots X_n^{m_n} \rangle = 0$ otherwise. The annihilator $J = \mathrm{ann}(\boldsymbol{u}_1)$ admits the lexicographic Gröbner basis $\langle X_1 - X_n, \ldots, X_{n-1} - X_n, X_n^{\delta} \rangle$, so we have $\deg(J) = \delta$; on the other hand, this sequence takes $\binom{\deg(J)+n-1}{n}$ non-zero values, so taking them all into account leads us to an exponential time algorithm.

In the case $t = 1$, Mourrain in [19] associates a Hankel operator to a sequence such that the kernel of the Hankel operator corresponds to the annihilator of the sequence. Algorithm 2 in this paper computes a border basis for the kernel of such a Hankel operator, taking as input its values over a finite set of monomials. As in the FGLM algorithm, this algorithm looks for linear dependencies between the monomials in the border of already computed linearly independent monomials. However, for examples as in the previous paragraph, we are not aware of how to avoid taking into account up to $\binom{\deg(J)+n-1}{n}$ values.

Several algorithms were also proposed in [1] for computing an annihilator $\mathrm{ann}(\boldsymbol{u}_1)$, and partly extended to arbitrary t in [2]. A first algorithm relies on the Berlekamp-Massey Algorithm, by means of a change of coordinates, which may require an exponential number of value of $\boldsymbol{u}_1$. The other algorithms extend the idea of FGLM, considering maximal rank sub-matrices of a truncated multi-Hankel matrix to compute a basis for the quotient algebra and a Gröbner basis. An algorithm with certified outcome (Scalar-FGLM) is presented; it considers the values of $\boldsymbol{u}_1$ at all monomials up to a given degree $\simeq \deg(J)$, so the issue pointed out above remains. An "adaptive" version uses fewer values of the sequence, but may fail in some cases (the conditions that ensure success of this algorithm seem to be close to the genericity assumptions we introduce in Subsection 3.2). A comparison of Scalar-FGLM and Sakata's algorithm is presented in [3].

3.1 A first algorithm

The first solution we discuss requires a strong assumption (written H_1 below): for any i and for any monomial b in $X_1, \ldots, X_n$, $b \cdot \boldsymbol{u}_i$ is in the $\mathbb{K}$-span of $(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t)$; as a result, the annihilator J of $(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t)$ equals the nullspace $\ker(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_t)$. For this situation, Marinari, Möller and Mora gave in [17] an algorithm that compute a Gröbner basis of J, for any order (for definiteness, we refer here to their second algorithm); it is an extension of both the Buchberger-Möller interpolation algorithm and the FGLM change of order algorithm.

Assumption H_1 above implies that $\deg(J) \le t$, and the runtime of the algorithm, expressed in terms of n and t, is $O(nt^3)$ operations in $\mathbb{K}$, together with the computation of all values $\langle \boldsymbol{u}_i \mid b \rangle$, $1 \le i \le t$, for $O(nt)$ monomials b. These evaluations are done in incremental order, in the sense that for any monomial b for which we need all $\langle \boldsymbol{u}_i \mid b \rangle$, there exists $j \in \{1, \dots, n\}$ such that $b = X_j b'$ and all $\langle \boldsymbol{u}_i \mid b' \rangle$ are known.

We will need the following property of this algorithm. Suppose that $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_t)$ is a subsequence of a larger family of sequences $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_{t'})$ that satisfies H_1, but that $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_t)$ itself may or may not, and that $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_t)$ and $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_{t'})$ have different $\mathbb{K}$-spans. Then, on input $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_t)$, the algorithm will still run its course, and at least one of the elements in the output will be a polynomial g that does not belong to $\operatorname{ann}(\boldsymbol{u}_1, \dots, \boldsymbol{u}_{t'})$.

3.2 An algorithm under genericity assumptions

We now give a second algorithm for computing $J = \operatorname{ann}(\boldsymbol{u}_1, \dots, \boldsymbol{u}_t)$, whose runtime is polynomial in $n, t, D = \deg(J)$ and an integer $B \le \deg(J)$ defined below. We do not assume that H_1 holds, but we will require other assumptions; if they hold, the output is the lexicographic Gröbner basis G of J for the order $X_1 > \cdots > X_n$. Our first assumption is:

H_2. We are given an integer B such that the minimal polynomial of X_j in $\mathbb{K}[X_1, \dots, X_n]/J$ has degree at most B for all j.

For j in $1, \dots, n$, we will denote by J_j the ideal $\operatorname{ann}(\pi_j(\boldsymbol{u}_1), \dots, \pi_j(\boldsymbol{u}_t)) \subset \mathbb{K}[X_j, \dots, X_n]$, where for all i, $\pi_j(\boldsymbol{u}_i)$ is the sequence $\mathbb{N}^{n-j+1} \to \mathbb{K}$ defined by $\langle \pi_j(\boldsymbol{u}_i) \mid (m_j, \dots, m_n) \rangle = \langle \boldsymbol{u}_i \mid (0, \dots, 0, m_j, \dots, m_n) \rangle$ for all $(m_j, \dots, m_n)$ in $\mathbb{N}^{n-j+1}$; in particular, $J_1 = J$. We write $\deg(J_j) = D_j \le D$, we let G_j be the lexicographic Gröbner basis of J_j, and we let $\mathscr{B}_j$ be the corresponding monomial basis of $\mathbb{K}[X_j, \dots, X_n]/J_j$.

We can then introduce our genericity property; by contrast with H_2, we will not necessarily assume that it holds, and discuss the outcome of the algorithm when it does not. We denote this property by $\mathsf{H}_3(j)$, for $j = 1, \dots, n-1$.

$\mathsf{H}_3(j)$. We have the equality $J_j \cap \mathbb{K}[X_{j+1}, \dots, X_n] = J_{j+1}$.

Remark that the inclusion $J_j \cap \mathbb{K}[X_{j+1}, \dots, X_n] \subset J_{j+1}$ always holds.

Suppose that for some j in $1, \dots, n$, we have computed a sequence of monomials $\mathscr{B}'_{j+1}$ in $\mathbb{K}[X_{j+1}, \dots, X_n]$ (if $j = n$, we let $\mathscr{B}'_{j+1} = (1)$). Since we will use them repeatedly, we define properties P and P' as follows, the latter being stronger than the former.

$\mathsf{P}(j+1)$. The cardinality D'_{j+1} of $\mathscr{B}'_{j+1}$ is at most D_{j+1}.
$\mathsf{P}'(j+1)$. The equality $\mathscr{B}'_{j+1} = \mathscr{B}_{j+1}$ holds.

We describe in the following paragraphs a procedure that computes a new family of monomials $\mathscr{B}'_j$, and we give conditions under which they satisfy $\mathsf{P}(j)$ and $\mathsf{P}'(j)$.

We call a family of monomials $\mathscr{B}$ in $\mathbb{K}[X_j,\ldots,X_n]$ *independent* if their images are $\mathbb{K}$-linearly independent modulo J_j (we call it *dependent* otherwise). We denote by $\mathsf{M}_{\mathscr{B}}$ the matrix with entries $\langle \boldsymbol{u}_i \mid bb'\rangle$, with rows indexed by $i=1,\ldots,t$ and b' in $\mathscr{C}_{j+1} = \mathscr{B}'_{j+1} \times (1, X_j, \ldots, X_j^{B-1})$, and columns indexed by b in $\mathscr{B}$ (for any monomial b in $\mathbb{K}[X_j,\ldots,X_n]$, M_b is the column vector defined similarly).

Lemma 4 *If $\mathscr{B}$ is dependent, the right nullspace of $\mathsf{M}_{\mathscr{B}}$ is non-trivial. If both $\mathsf{P}'(j+1)$ and $\mathsf{H}_3(j)$ hold, the converse is true.*

Proof. Any $\mathbb{K}$-linear relation between the elements of $\mathscr{B}$ induces the same relation between the columns of $\mathsf{M}_{\mathscr{B}}$, and the first point follows.

By definition, a polynomial f in $\mathbb{K}[X_j,\ldots,X_n]$ belongs to J_j if and only if it annihilates $\pi_j(\boldsymbol{u}_1),\ldots,\pi_j(\boldsymbol{u}_t)$, that is, if $\langle \pi_j(\boldsymbol{u}_i) \mid X_j^{m_j}\ldots X_n^{m_n} f\rangle = 0$ for all $(m_j,\ldots,m_n)$ in $\mathbb{N}^{n-j+1}$ and all $i=1,\ldots,t$. Now, assumptions $\mathsf{P}'(j+1)$, H_2 and $\mathsf{H}_3(j)$ imply that $\mathscr{C}_{j+1}$ generates $\mathbb{K}[X_j,\ldots,X_n]/J_j$, so that f is in J_j if and only if $\langle \boldsymbol{u}_i \mid bf\rangle = 0$, for all b in $\mathscr{C}_{j+1}$ and all $i=1,\ldots,t$. $\square$

The following lemma, that essentially follows the argument used in the proof of the FGLM algorithm [11], will be useful to justify our algorithm as well.

Lemma 5 *Suppose that $b_1 < \cdots < b_u < b_{u+1}$ are the first $u+1$ standard monomials of $\mathbb{K}[X_j,\ldots,X_n]/J_j$, for the lexicographic order induced by $X_j > \cdots > X_n$, with $b_1 = 1$. Then for any monomial b such that $b_u < b < b_{u+1}$, $\{b_1,\ldots,b_u,b\}$ is a dependent family.*

Proof. We prove the result by induction on $u \geq 0$, the case $u=0$ being vacuously true. Assuming the claim is true for some index $u \geq 0$, we prove it for $u+1$. We proceed by contradiction, and we let b be the smallest monomial such that $b_u < b < b_{u+1}$ and $\{b_1,\ldots,b_u,b\}$ is an independent family (b exists by the well-ordering property of monomial orders).

We will use the fact that any monomial c less than b can be rewritten as a linear combination of $b_1,\ldots,b_i$, with $b_i < c$, for some $i \leq u$: if $c < b_u$, this is by the induction assumption; if $c = b_u$, this is obvious; if $b_u < c < b$, this is by the definition of b.

Now, either b is the leading term of an element in the Gröbner basis of J_j, or it must be of the form $b = X_e b'$, for some monomial b' not in $\{b_1,\ldots,b_u\}$. We prove that in both cases, b can be rewritten as a linear combination of $b_1,\ldots,b_u$, which is a contradiction. In the first case, b rewrites as a linear combination of smaller monomials, say $c_1,\ldots,c_v$, and by the previous remark, all of them can be rewritten as linear combinations of $b_1,\ldots,b_u$. Altogether, b itself can be rewritten as a linear combination of $b_1,\ldots,b_u$, a contradiction.

In the second case, $b = X_e b'$, for some monomial b' not in $\{b_1,\ldots,b_u\}$. As above, b' can be rewritten modulo J_j as a linear combination of monomials $b_1,\ldots,b_i$, for some $i \leq u$, with $b_i < b'$. Then, $b = X_e b'$ is a linear combination of $X_e b_1,\ldots,X_e b_i$. Since $b_i < b'$, we get $X_e b_1 < \cdots < X_e b_i < X_e b' = b$, so all of

$X_e b_1, \ldots, X_e b_i$ can be rewritten as linear combinations of $b_1, \ldots, b_u$. As a result, this is also the case for b itself, so we get a contradiction again. □

Suppose that $\mathsf{P}(j+1)$ holds. Then, the algorithm at step j proceeds as follows. We compute the reduced row echelon form of $\mathsf{M}_{\mathscr{C}_{j+1}}$. Using assumption $\mathsf{P}(j+1)$, this matrix has at most tBD_{j+1} rows and at most BD_{j+1} columns, and it has rank at most D_j (by the first item of Lemma 4). This computation can be done in time $O(tB^2D_{j+1}^2D_j) \in O(tB^2D^3)$. The column indices of the pivots allow us to define the monomials $\mathscr{B}'_j = (b'_1 < \cdots < b'_{D'_j})$, for some $D'_j \le D_j$.

Lemma 6 *Property* $\mathsf{P}(j)$ *holds, and if* $\mathsf{P}'(j+1)$ *and* $\mathsf{H}_3(j)$ *hold, then* $\mathsf{P}'(j)$ *holds.*

Proof. The first item is a restatement of the inequality $D'_j \le D_j$. To prove the second item, assuming that $\mathsf{P}'(j+1)$ and $\mathsf{H}_3(j)$ hold, we deduce from Lemma 4 that the columns indexed by the genuine $\mathscr{B}_j$ form a column basis of $\mathsf{M}_{\mathscr{C}_{j+1}}$, and we claim that it is actually the lexicographically smallest column basis (this will prove that $\mathscr{B}_j = \mathscr{B}'_j$). Indeed, write $\mathscr{B}_j = (b_1, \ldots, b_{D_j})$, and let $(f_1, \ldots, f_{D_j})$ be another subsequence of $\mathscr{C}_{j+1}$ whose corresponding columns form a column basis of $\mathsf{M}_{\mathscr{C}_{j+1}}$. Let m be the smallest index such that $b_m \ne f_m$. Then, applying Lemma 5 to $(b_1, \ldots, b_{m-1})$ and f_m, we deduce that $b_m < f_m$ (otherwise, since they are different, we must have $b_{m-1} < f_m < b_m$, which implies that f_m is a linear combination of $(b_1, \ldots, b_{m-1}) = (f_1, \ldots, f_{m-1})$, a contradiction). □

Thus, running this procedure for $j = n, \ldots, 1$, we maintain $\mathsf{P}(j)$; this implies that the running time is $O(ntB^2D^3)$, computing the values $\langle \boldsymbol{u}_i \mid b\rangle$, for $1 \le i \le t$, for $O(nB^2D^2)$ monomials b (with the same monotonic property as in the previous subsection). If $\mathsf{H}_3(j)$ holds for all j, the second item in the last lemma proves that $\mathscr{B}'_1 = \mathscr{B}_1$, the monomial basis of $\mathbb{K}[X_1, \ldots, X_n]/J$.

Once $\mathscr{B}'_1$ is known, we compute and return a family of polynomials G' defined as follows. We determine the sequence Δ of elements in $X_1\mathscr{B}'_1 \cup \cdots \cup X_n\mathscr{B}'_1 - \mathscr{B}'_1$, all of whose factors are in $\mathscr{B}'_1$ (finding them does not require any operation in $\mathbb{K}$; this can be done by using e.g. a balanced binary search tree with the elements of $\mathscr{B}'_1$, using a number of comparisons that is quasi-linear time in nD). Then, we rewrite each column M_b, for b in Δ, as a linear combination of the form $\sum_{1\le i\le D'_1} c_i \mathsf{M}_{b'_i}$ and we put $b - \sum_{1\le i\le D'_1} c_i b'_i$ in G'. If the reduction is not possible, the algorithm halts and returns fail. Using the reduced row echelon form of $\mathsf{M}_{\mathscr{C}_2}$, each reduction takes time $O(D_1^2) \in O(D^2)$ operations in $\mathbb{K}$, for a total of $O(nD^3)$.

If $\mathsf{H}_3(j)$ holds for all j, since $\mathscr{B}_1 = \mathscr{B}'_1$, the fact that $G' = G$ follows from Lemma 4. Assume now that G' differs from G; we prove that there exists an element in G not in J (we will use this in our main algorithm to detect failure cases). Indeed, in this case, $\mathscr{B}'_1$ must be different from $\mathscr{B}_1$, and since $\mathscr{B}'_1$ has cardinality at most equal to that of $\mathscr{B}_1$, there exists a monomial b in $\mathscr{B}_1$ not in $\mathscr{B}'_1$. This in turn implies that there exists an element g in G' that divides b, and thus with leading term in $\mathscr{B}_1$. Reducing g modulo G, we must then obtain a non-zero remainder, so that g does not belong to J.

4 Main algorithm

4.1 Representing primary zero-dimensional ideals

Let I be a zero-dimensional ideal in $\mathbb{K}[X_1, \dots, X_n]$; we assume that I is $\mathfrak{m}$-primary, for some maximal ideal $\mathfrak{m}$, and we write $D = \deg(I)$. In this paragraph, we briefly mention some possible representations for I (our main algorithm will compute either one of these representations).

The first, and main, option we will consider is simply the Gröbner basis G of I, for the lexicographic order induced by $X_1 > \cdots > X_n$. As an alternative, consider the following construction. Our assumption on I implies that the minimal polynomial R of X_n in $\mathbb{K}[X_1, \dots, X_n]/I$ takes the form $R = P^e$, for some irreducible polynomial P in $\mathbb{K}[Z]$, of degree say f (remark that $R(X_n)$ is also the last polynomial in G). Let $\mathbb{L} = \mathbb{K}[Z]/\langle P\rangle$; this is a field extension of degree f of $\mathbb{K}$, and the residue class ζ of Z in $\mathbb{L}$ is a root of P. We then let I' be the ideal $I + \langle (X_n - \zeta)^e\rangle$ in $\mathbb{L}[X_1, \dots, X_n]$, and let D' be its degree. Then, a second option is to compute the lexicographic Gröbner basis G' of I', for the order $X_1 > \cdots > X_n$. The following lemma relates D and D'.

Lemma 7 *The ideal I' has degree $D' = D/f$.*

Proof. Let $\mathbb{M}$ be the splitting field of P and let $\zeta_1, \dots, \zeta_f$ be the roots of P in $\mathbb{M}$. The ideals $J_i = I + \langle (X_n - \zeta_i)^e\rangle \subset [X_1, \dots, X_n]$ are such that $\deg(J_1) + \cdots + \deg(J_f) = \deg(I)$. On the other hand, there exist f embeddings $\sigma_1, \dots, \sigma_f$ of $\mathbb{L}$ into $\mathbb{M}$, with σ_i given by $\zeta \mapsto \zeta_i$; as a result, $\deg(I') = \deg(J_i)$ holds for all i, and the claim follows. □

The point behind this construction is to lower the degree of the ideal we consider, at the cost of working in a field extension of $\mathbb{K}$. This may be beneficial, as the cost of the main algorithm (which essentially relies on the one in the previous section) will be a polynomial of rather large degree with respect to the degree of the ideal, whereas computation in a field extension such as $\mathbb{K} \to \mathbb{L}$ is a well-understood task of cost ranging from quasi-linear to quadratic.

Our last option aims at producing a "simpler" Gröbner basis, by means of a change of coordinates. For this, we will assume that X_n separates the points of $V(\mathfrak{m})$ (over an algebraic closure of $\mathbb{K}$). As a result, the ideal $\mathfrak{m}$ being maximal, it admits a lexicographic Gröbner basis of the form $\langle X_1 - G_1(X_n), \dots, X_{n-1} - G_{n-1}(X_n), P(X_n)\rangle$. Define $\xi_1 = G_1(\zeta), \dots, \xi_{n-1} = G_{n-1}(\zeta), \xi_n = \zeta$, for $\zeta \in \mathbb{L}$ as above; then, $(\xi_1, \dots, \xi_n)$ is the unique zero of I' (in fact, I' is $\mathfrak{m}'$-primary, with $\mathfrak{m}' = \langle X_1 - \xi_1, \dots, X_n - \xi_n\rangle$). We can then apply the change of coordinates that replaces X_i by $X_i + \xi_i$ in I', for all i, and call I'' the ideal thus obtained (so that I'' is generated by the polynomials $f(X_1 + \xi_1, \dots, X_n + \xi_n)$, for f in I, and X_n^e). Now, I'' is $\mathfrak{m}''$-primary, with $\mathfrak{m}'' = \langle X_1, \dots, X_n\rangle$; one of our options will be to compute the Gröbner basis G'' of I''.

Example 1 *Consider the polynomials in* $\mathbb{Q}[X_1, X_2]$

$$\begin{aligned} X_1^2 - 2X_1X_2 - 2X_1 + X_2^2 + 2X_2 + 1, \\ X_1X_2^2 + X_1X_2 + 2X_1 - X_2^3 - 2X_2^2 - 3X_2 - 2, \\ X_2^4 + 2X_2^3 + 5X_2^2 + 4X_2 + 4, \end{aligned}$$

the last of them being $P(X_2)^2 = (X_2^2 + X_2 + 2)^2$*, and let* I *be the ideal they define. The polynomials above are the lexicographic Gröbner basis* G *of* I *for the order* $X_1 > X_2$*. Let* $\mathbb{L} = \mathbb{Q}[Z]/\langle Z^2 + Z + 2\rangle$*, and let* ζ *be the image of* Z *in* $\mathbb{L}$*; then, the ideal* $I' = I + \langle (X_2 - \zeta)^2\rangle$ *in* $\mathbb{L}[X_1, X_2]$ *admits the Gröbner basis* G'

$$\begin{aligned} X_1^2 - 2X_1\zeta - 2X_1 + \zeta - 1, \\ X_1X_2 - X_1\zeta - X_2\zeta - X_2 - 2, \\ X_2^2 - 2X_2\zeta - \zeta - 2. \end{aligned}$$

Here, we have $e = 2$*,* $f = 2$*,* $D = 6$ *and* $D' = 3$*. The ideal* I *is* $\mathfrak{m}$*-primary, where* $\mathfrak{m}$ *admits the Gröbner basis* $\langle X_1 - X_2 - 1, X_2^2 + X_2 + 2\rangle$*, so that we have* $(\xi_1, \xi_2) = (\zeta + 1, \zeta)$*, and* I' *is* $\mathfrak{m}'$*-primary, with* $\mathfrak{m}' = \langle X_1 - \xi_1, X_2 - \xi_2\rangle$*. Applying the change of coordinates* $(X_1, X_2) \leftarrow (X_1 + \xi_1, X_2 + \xi_2)$*, the resulting ideal* I'' *admits the Gröbner basis* $G'' = \langle X_1^2, X_1X_2, X_2^2\rangle$*, from which we can readily confirm that it is* $\langle X_1, X_2\rangle$*-primary.*

4.2 The algorithm

We consider a zero-dimensional ideal I in $\mathbb{K}[X_1, \ldots, X_n]$. We assume that we know a monomial basis $B = (b_1, \ldots, b_D)$ of $Q = \mathbb{K}[X_1, \ldots, X_n]/I$, so that we let $D = \dim_{\mathbb{K}}(Q)$, together with the corresponding multiplication matrices $\mathsf{M}_1, \ldots, \mathsf{M}_n$ of respectively $X_1, \ldots, X_n$. We assume that the last variable X_n has been chosen generically; in particular, X_n separates the points of $V = V(I)$.

The algorithm in this section computes a decomposition of I into primary components $J_1, \ldots, J_K$. Each such component J_k will be given by means of one of the representations described in the previous subsection; we will emphasize the first of them, the lexicographic Gröbner basis of J_k, and mention how to modify the algorithm in order to obtain the other representations. In order to find the primary components of I, we cannot avoid the use of factorization algorithms over $\mathbb{K}$; if desired, one may avoid this by relying on *dynamic evaluation techniques* [7], replacing for instance the factorization into irreducibles used below by a squarefree factorization (thus producing a decomposition of I into ideals that are not necessarily primary). In that case, if one wishes to compute descriptions such as the second or third ones introduced above, involving algebraic numbers as coefficients, one should take into account the possibility of splittings the defining polynomials, as is usual with this kind of approach (a complete description of the resulting algorithm, along the lines of [6], is beyond the scope of this paper).

The ideal I and its primary decomposition. Let $P_{\min} \in \mathbb{K}[X_n]$ be the minimal polynomial of X_n in Q, let P be its squarefree part, and let polynomials

$G_1, \ldots, G_{n-1}$ in $\mathbb{K}[X_n]$, with $\deg(G_i) < \deg(P)$ for all i, be such that $\sqrt{I}$ admits the lexicographic Gröbner basis $\langle X_1 - G_1(X_n), \ldots, X_{n-1} - G_{n-1}(X_n), P(X_n)\rangle$. We write $P_{\min} = P_1^{e_1} \cdots P_K^{e_K}$, with the P_k's pairwise distinct irreducible polynomials in $\mathbb{K}[X_n]$ and $e_k \geq 1$ for all k. In particular, the factorization of P is $P_1 \cdots P_K$; we write $f_k = \deg(P_k)$ for all k.

Correspondingly, let $V_1, \ldots, V_K$ be the $\mathbb{K}$-irreducible components of V and for $k = 1, \ldots, K$, let $\mathfrak{m}_k$ be the maximal ideal defining V_k; hence, the reduced lexicographic Gröbner basis of $\mathfrak{m}_k$ is $\langle X_1 - (G_1 \bmod P_k), \ldots, X_{n-1} - (G_{n-1} \bmod P_k), P_k\rangle$. We can then write $I = J_1 \cap \cdots \cap J_K$, with J_k $\mathfrak{m}_k$-primary for all k; note that the ideal J_k is defined by $J_k = I + \langle P_k^{e_k}\rangle$. In what follows, we explain how to compute a Gröbner basis of this ideal by means of the results of the previous section. Without loss of generality, assume that L is such that $e_k = 1$ for $k > L$ and $e_k \geq 2$ for $k = 1, \ldots, L$. The fact that X_n is a generic coordinate implies that for $k > L$, $J_k = \mathfrak{m}_k$, so there is nothing left to do for such indices; hence, we are left with showing how to use the algorithms of the previous section to compute Gröbner bases of $J_1, \ldots, J_L$.

Data representation. An element f of Q is represented by the column vector v_f of its coordinates on the basis B, whereas a linear form $\ell : Q \to \mathbb{K}$ is represented by the row vector $\mathsf{w}_\ell = [\ell(b_1), \ldots, \ell(b_D)]$. Computing $\ell(f)$ is then done by means of the dot product $\mathsf{w}_\ell \cdot \mathsf{v}_f$. Multiplying f by X_i amounts to computing $\mathsf{M}_i \mathsf{v}_f$, and the linear form $X_i \cdot \ell : g \mapsto \ell(X_i g)$ is obtained by computing the vector $\mathsf{w}_{X_i \cdot \ell} = \mathsf{w}_\ell \mathsf{M}_i$.

In terms of complexity, we assume that multiplying any matrix M_i by a vector (either on the left or on the right) can be done in m operations in $\mathbb{K}$. The naive bound on m is $O(D^2)$, but the sparsity properties of these matrices often result in much better estimates; see [12] for an in-depth discussion of this question. On the other hand, we assume $D \leq \mathsf{m}$.

Computing $P_{\min}$ and $G_1, \ldots, G_{n-1}$. First, we compute generators of $\sqrt{I}$. We choose a random linear form $\ell_1 : Q \to \mathbb{K}$, and we compute the values $(\ell_1(X_n^i))_{0 \leq i < 2D}$ and $\ell_1(X_1 X_n^i), \ldots, \ell_1(X_{n-1} X_n^i)$, for $0 \leq i < D$. This is done by computing $1, X_n, \ldots, X_n^{2D-1}$ by repeated applications of M_n, which amounts to $O(D\mathsf{m})$ operations, and doing the corresponding dot products with $\ell, X_1 \cdot \ell, \ldots, X_{n-1} \cdot \ell$. For the latter, we have to compute the linear forms $X_i \cdot \ell$ in $O(n\mathsf{m})$ operations, then do a $D \times D$ by $D \times (n+1)$ matrix product, which costs $O(nD^2)$ operations (without using fast linear algebra).

Using the algorithm given in [4], given these values, we can compute the minimal polynomial $P_{\min}$, as well as the polynomials $G_1, \ldots, G_{n-1}$ describing $V(I)$ in $O\tilde{}(D)$ operations in $\mathbb{K}$. Then, as per the discussion in the preamble, we assume that we have an algorithm for factoring polynomials over $\mathbb{K}$, so that $(P_1, e_1), \ldots, (P_K, e_K)$ and P can be deduced from $P_{\min}$.

Constructing the orthogonal of J_k. For $k = 1, \ldots, K$, we will write $Q_k = \mathbb{K}[X_1, \ldots, X_n]/J_k$. Any linear form $\ell : Q \to \mathbb{K}$ induces a linear form $\varphi_k(\ell) : Q_k \to \mathbb{K}$, defined as follows.

Let T_k be the polynomial $P_{\min} / P_k^{e_k}$. For f in Q_k, let $\hat{f}$ be any lift of f to $\mathbb{K}[X_1, \ldots, X_n]$, and define $\varphi_k(\ell)(f) = \ell(T_k \hat{f} \bmod I)$. Notice that this expression is well-defined: indeed, any two lifts of f differ by an element δ of $J_k = I + \langle P_k^{e_k} \rangle$, so that $T_k\delta$ is in I, since $T_k P_k^{e_k} = P_{\min}$ is.

Lemma 1. *The mapping $\varphi_k : Q^* \to Q_k^*$ is $\mathbb{K}$-linear and onto.*

Proof. Linearity is clear by construction; we now prove that φ_k is onto. Let indeed A_k, B_k in $\mathbb{K}[X_n]$ be such that $A_k T_k + B_k P_k^{e_k} = 1$ (they exist by definition of T_k). Consider λ in Q_k^*, and define ℓ in Q^* by $\ell(f) = \lambda(A_k f \bmod J_k)$. Since $P_k^{e_k}$ vanishes modulo J_k, we have $A_k T_k = 1 \bmod J_k$, so $\ell(f) = \lambda(f \bmod J_k)$ holds for all f in Q; this in turn readily implies that $\varphi_k(\ell) = \lambda$. □

We saw in Subsection 2 how to associate to an element $\ell \in Q^*$ a sequence $\boldsymbol{u}_\ell \in \mathscr{S}$, by letting $\langle \boldsymbol{u}_\ell \mid m \rangle = \ell(m \bmod I)$. The following tautological observation will then be useful below: for ℓ in Q^*, the sequences $\boldsymbol{u}_{T_k \cdot \ell}$ and $\boldsymbol{u}_{\varphi_k(\ell)}$ coincide, where $\boldsymbol{u}_{\varphi_k(\ell)}$ is defined starting from the linear form $\varphi_k(\ell) \in Q_k^*$. Indeed, take any monomial m in $X_1, \dots, X_n$; then, $\varphi_k(\ell)(m \bmod J_k)$ is defined as $\ell(T_k m \bmod I)$, which is equal to $(T_k \cdot \ell)(m \bmod I)$. We will use this remark to compute values of $\varphi_k(\ell)$, through the computation of values of $T_k \cdot \ell$ instead.

In algorithmic terms, computing a single transposed product by a polynomial $T(X_n)$, that is, $T \cdot \ell$, can be done using Horner's rule, using d right-multiplications by M_n, with $d = \deg(T)$; this takes $O(d\mathsf{m})$ operations in $\mathbb{K}$. If several transposed products are needed, such as for instance computing $T_1 \cdot \ell, \dots, T_L \cdot \ell$ as below, the cost becomes $O(LD\mathsf{m})$, using D as an upper bound on $\deg(T_1), \dots, \deg(T_L)$. One can actually do better, by computing inductively and storing the products $X_n^i \cdot \ell$, for $i = 0, \dots, D-1$. Then, the coefficients of $T_1 \cdot \ell, \dots, T_L \cdot \ell$ can be computed as the product of the $D \times d'$ matrix of coefficients of $(X_n^i \cdot \ell)_{0 \le i < D}$ by the matrix of coefficients of $T_1, \dots, T_L$; the cost is $O(D\mathsf{m} + LD^2)$.

One can improve this idea further using *subproduct tree* techniques, since the polynomials $T_1, \dots, T_L$ have a very specific structure. Recall that we defined $T_k = P_{\min}/P_k^{e_k}$. Hence, all of $T_1, \dots, T_L$ share a common factor $R = P_{L+1}^{e_{L+1}} \cdots P_K^{e_K}$. We can then treat the common factor R separately, by writing $T_k = RU_k$ for all these indices k, and computing $U_1 \cdot \ell', \dots, U_L \cdot \ell'$ instead, with $\ell' = R \cdot \ell$. The cost to compute ℓ' is $O(D\mathsf{m})$.

The polynomials $U_1, \dots, U_L$ have no common factor anymore, but they are all of the form $P_1^{e_1} \cdots P_{k-1}^{e_{k-1}} P_{k+1}^{e_{k+1}} P_L^{e_L}$. We can then define a subproduct tree as in [13, Chapter 10], that is, a binary tree $\mathcal{T}$ having the polynomials $(P_k^{e_k})_{1 \le k \le L}$ at its leaves, and where each node is labeled by the product of the polynomials at its two children. We proceed in a top-down manner: we associate ℓ' to the root of the tree, and recursively, if a linear form λ has been assigned to an inner node of $\mathcal{T}$, we associate to each of its children the transposed product of λ by the polynomial labelling the other child. At the leaves, this gives us $U_L \cdot \ell', \dots, U_K \cdot \ell'$, as claimed. The total cost at each level is $O(D\mathsf{m})$, for a total of $O(D\log(L)\mathsf{m})$.

The main procedure, using the algorithm of Subsection 3.1. The first version of the main procedure determines the Gröbner bases of $J_L, \dots, J_K$ by applying the algorithm of Subsection 3.1 to successive families of linear forms.

We maintain a list of "active" indices S, initially set to $S = (1, \dots, L)$; these are the indices for which we are not done yet. The algorithm proceeds iteratively; at step $i \geq 1$, we pick a random linear form $\ell_i \in Q^*$, and compute all $\ell_{k,i} = T_k \cdot \ell_i$, for k in S. We then apply the algorithm of Subsection 3.1 to $(\boldsymbol{u}_{\ell_{k,1}}, \ldots, \boldsymbol{u}_{\ell_{k,i}})$, for all k independently, and obtain families of polynomials $G_{k,i}$ as output. For verification purposes, we also choose a random $\ell_0 \in Q^*$, and compute the corresponding $\ell_{k,0}$.

Write $D_k = \deg(J_k)$, for $k \leq K$. Combining Lemma 2 and the equality $\boldsymbol{u}_{\ell_{k,i}} = \boldsymbol{u}(\varphi_k(\ell_i))$ seen above, we deduce that for a generic choice of $\ell_1, \ldots, \ell_{D_k}$, $(\ell_{k,1}, \ldots, \ell_{k,D_k})$ satisfies assumption H_1 needed for our algorithm, and that G_{k,D_k} is a Gröbner basis of J_k. In view of the discussion in Subsection 3.1, for any $i < D_k$, $G_{k,i}$ contains a polynomial g not in J_k. Since ℓ_0 was chosen at random, $\ell_{k,0}$ will in general not vanish at g; hence, at every step i, we evaluate $\ell_{k,0}$ at all elements of $G_{k,i}$, and continue the algorithm for this index k if we obtain a non-zero value; else, we remove k from our list S, and append $G_{k,i}$ to the output.

In terms of complexity, we will have to apply the process in the previous paragraph to μ linear forms $\ell_{D_1}, \ldots, \ell_\mu$, with $\mu = \max_{k \leq L}(D_k)$, for a cost $O(\mu D \mathsf{m} \log(L))$. Then, we will exploit a feature of Marinari-Möller-Mora's second algorithm: it is incremental in the number of linear forms given as input, so that the overall runtime of our D_k successive invocations is the same as if we called it once with $\ell_1, \ldots, \ell_{D_k}$. For a given k, it adds up to $O(nD_k^2\mathsf{m} + nD_k^3) = O(nD_k^2\mathsf{m})$, where the first term describes the cost of the evaluations of the linear forms we need (since each new value requires the product by one of the M_i). Overall, the runtime is $O(\mu D \log(L)\mathsf{m} + n \sum_{k \leq L} D_k^2 \mathsf{m})$. This supports the comment made in the introduction: if the degrees of the multiple components are small, say $D_k = O(1)$ for all k, this is $O(nD\log(D)\mathsf{m})$.

Using the algorithm of Subsection 3.2. We can adapt our main procedure in order to use the algorithm of Subsection 3.2 instead; the main difference is that we expect to use fewer linear forms.

For $k \leq K$, let indeed $t_k \leq D_k$ be the maximum of $\tau(Q_{k,\geq 1}), \ldots, \tau(Q_{k,\geq n})$, with $Q_{k,\geq j} = \mathbb{K}[X_j, \ldots, X_n]/J_k \cap \mathbb{K}[X_j, \ldots, X_n]$, and with τ defined as in Proposition 1 (for instance, if I is a complete intersection ideal, $t_k = 1$ for all k). The main algorithm proceeds as in the previous variant: we choose random linear forms $\ell_1, \ldots$ and deduce $\ell_{k,i} = T_k \cdot \ell_i$; we will compute the Gröbner basis G_k of J_k as $\mathrm{ann}(\boldsymbol{u}_{\ell_{k,1}}, \boldsymbol{u}_{\ell_{k,2}}, \ldots)$. We claim that we only need t_k linear forms $\ell_1, \ldots, \ell_{t_k}$ in order to recover G_k.

To confirm this, we consider again assumptions H_2 and H_3 made in Subsection 3.2. The appendix of [4] implies that the minimal polynomial of any variable X_i in Q_k has degree at most e_k, except for X_n. We already know the minimal polynomial $P_k^{e_k}$ of X_n in Q_k, so we skip the first pass in the loop of the algorithm of Subsection 3.2, and use the value $B = e_k$.

Regarding H_3, we prove that if $\ell_1, \ldots, \ell_{t_k}$ are chosen generically, assumption $\mathsf{H}_3(j)$ holds for $j = 1, \ldots, n$. For $i \geq 1$ and $j = 1, \ldots, n$, define $\ell_{k,i,j}$ as the linear form in $Q_{k,\geq j}^*$ induced by restriction of $\varphi_k(\ell_i) \in Q_k^*$. Applying Proposition 1 to $Q_{k,\geq j}$ shows that there exists a Zariski open $\Omega_{k,j} \subset {Q_{k,\geq j}^*}^{t_k}$ such that if $\ell_{k,1,j}, \ldots, \ell_{k,t_k,j}$ are in $\Omega_{k,j}$, they generate $Q_{k,\geq j}^*$ as a $Q_{k,\geq j}$-module, and thus (Lemma 3) $J_k \cap \mathbb{K}[X_j, \ldots, X_n] = \mathrm{ann}(\boldsymbol{u}_{\ell_{k,1,j}}, \ldots, \boldsymbol{u}_{\ell_{k,t_k,j}})$. If this is true for some index k and all j, $\mathsf{H}_3(j)$ follows as well for these indices. Now, the mapping $\Delta_{k,j} : (\ell_1, \ldots, \ell_{t_k}) \mapsto (\ell_{k,1,j}, \ldots, \ell_{k,t_k,j})$ is $\mathbb{K}$-linear and onto (we proved above that $(\ell_1, \ldots, \ell_{t_k}) \mapsto (\varphi_k(\ell_1), \ldots, \varphi_k(\ell_{t_k}))$ is onto, and the surjectivity of the projection is straightforward), so that the preimage $\Delta_{k,j}^{-1}(\Omega_{k,j})$ is Zariski open in Q^{*t_k} for all k, j. In other words, for generic $\ell_1, \ldots, \ell_{t_k}$, $\mathsf{H}_3(j)$ holds for all j and all k, so the algorithm of Subsection 3.2 computes G_k for all k.

We still need to discuss what happens when applying this algorithm to $\ell_{k,1}, \ldots, \ell_{k,i}$ for some $i < t_k$. In this case, as per the discussion in Subsection 3.2, either we get generators of $\mathrm{ann}(\boldsymbol{u}_{\ell_{k,1}}, \ldots, \boldsymbol{u}_{\ell_{k,i}})$, which is a strict superset of J_k, or at least one of the polynomials in the output does not belong to $\mathrm{ann}(\boldsymbol{u}_{\ell_{k,1}}, \ldots, \boldsymbol{u}_{\ell_{k,i}})$. In any case, the output contains at least one polynomial g not in J_k, so we can use the same stopping criterion as in the previous paragraph, using a linear form ℓ_0 to test termination.

To control the complexity, at the ith step, we now use linear forms $\ell_1, \ldots, \ell_{2^i}$; as a result, we need to go up to $i = t$, with $t = \max_k(t_k)$, and the overall runtime is proportional to that at $i = t$. The cost of preparing the linear forms $\ell_{k,i}$ is $O(tD\mathsf{m}\log(L))$, and the cost of computing annihilators is $O(nt\sum_{k\le L} e_k^2 D_k^2 \mathsf{m})$. The first term is better than the equivalent term for our first algorithm, but the second one is obviously worse. On the other hand, the analysis in Subsection 3.2 can be refined significantly, and possibly lead to improved estimates.

Using a scalar extension. To conclude, we discuss (without giving proofs) how to put to practice the idea introduced in Subsection 4.1 of computing Gröbner bases of ideals of smaller degree over larger base fields, in the context (for definiteness) of the algorithm of the previous paragraph.

Let $\ell_{k,1}, \ldots, \ell_{k,t_k}$ be defined as before, let $\boldsymbol{u}_{\ell_{k,1}}, \ldots, \boldsymbol{u}_{\ell_{k,t_k}}$ be the corresponding sequences, and assume that these linear forms are such that the annihilator of $\boldsymbol{u}_{\ell_{k,1}}, \ldots, \boldsymbol{u}_{\ell_{k,t_k}}$ is J_k. Let further $\mathbb{L}_k$ be the field extension $\mathbb{K}[Z]/P_n(Z)$, and let ζ_k be the residue class of Z in $\mathbb{L}_k$ Then, the annihilator of $J'_k = J_k + \langle X_n - \zeta_k \rangle^{e_k}$ in $\mathbb{L}[X_1, \ldots, X_n]$ has degree D_k/f_k by Lemma 7, so we might want to compute it instead of J_k. To accomplish this, we need sequences whose annihilator would be J'_k; we do this following the same strategy as above. Define $S_k = P_k/(X_n - \zeta_k) \in \mathbb{L}_k[X_n]$, as well as the linear form $\ell'_{k,i} = S_k^{e_k} \cdot \ell_{k,i} : \mathbb{L}[X_1, \ldots, X_n]/I \to \mathbb{L}$, for $i \ge 1$. Then, one verifies that $\mathrm{ann}(\boldsymbol{u}_{\ell'_{k,1}}, \ldots, \boldsymbol{u}_{\ell'_{k,t_k}})$ is indeed J'_k.

Our last comment discusses the translation mentioned in Subsection 4.1. The ideal J'_k is $\mathfrak{m}'$-primary, with $\mathfrak{m}' = \langle X_1 - \xi_1, \ldots, X_n - \xi_n \rangle$, as in Subsection 4.1. To replace J'_k by a $\langle X_1, \ldots, X_n \rangle$-primary ideal, we need to modify the sequences $\boldsymbol{u}_{\ell'_{k,1}}, \ldots, \boldsymbol{u}_{\ell'_{k,t_k}}$. For $i \ge 1$, let $U_{k,i} \in \mathbb{L}[[X_1, \ldots, X_n]]$ be the generating series of $\boldsymbol{u}_{\ell'_{k,i}}$, and let $\tilde{U}_{k,i} = \frac{1}{(1+\xi_1 X_1)\cdots(1+\xi_n X_n)} U_{k,i}(\frac{X_1}{1+\xi_1 X_1}, \ldots, \frac{X_n}{1+\xi_n X_n})$. Letting $\tilde{\boldsymbol{u}}_{k,i}$ be the sequence whose generating series is $\tilde{U}_{k,i}$, $\mathrm{ann}(\tilde{\boldsymbol{u}}_{k,1}, \ldots, \tilde{\boldsymbol{u}}_{k,t_k})$ is indeed the $\langle X_1, \ldots, X_n \rangle$-primary ideal J''_k obtained by translation by $(\xi_1, \ldots, \xi_n)$ in J'_k.

References

1. J. Berthomieu, B. Boyer, and J.-C. Faugère. Linear Algebra for Computing Gröbner Bases of Linear Recursive Multidimensional Sequences. *Journal of Sym-*

bolic Computation, In press, 2016.

2. J. Berthomieu and J.-C Faugère. Guessing Linear Recurrence Relations of Sequence Tuples and P-recursive Sequences with Linear Algebra. In *ISSAC'16*, pages 95–102. ACM, 2016.
3. J. Berthomieu and J.-C Faugère. In-depth comparison of the Berlekamp-Massey-Sakata and the Scalar-FGLM algorithms: the non adaptive variants. hal-01516708, May 2017.
4. A. Bostan, B. Salvy, and É. Schost. Fast algorithms for zero-dimensional polynomial systems using duality. *AAECC*, 14:239–272, 2003.
5. J. Brachat, P. Comon, B. Mourrain, and E. Tsigaridas. Symmetric tensor decomposition. *Linear Algebra and its Applications*, 433(11):1851–1872, 2010.
6. X. Dahan, M. Moreno Maza, É. Schost, and Y. Xie. On the complexity of the D5 principle. In *Transgressive Computing*, pages 149–168, 2006.
7. J. Della Dora, C. Dicrescenzo, and D. Duval. About a new method for computing in algebraic number fields. In *EUROCAL'85*, pages 289–290. Springer-Verlag, 1985.
8. D. Eisenbud. *Commutative Algebra: with a view toward algebraic geometry*, volume 150. Springer Science & Business Media, 2013.
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. *Journal of Symbolic Computation*, 16(4):329–344, 1993.
12. J.-C. Faugère and C. Mou. Sparse FGLM algorithms. *Journal of Symbolic Computation*, 80(3):538–569, 2017.
13. J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, Cambridge, third edition, 2013.
14. P. Gianni and T. Mora. Algebraic solution of systems of polynomial equations using Groebner bases. In *AAECC'87*, volume 356 of *Lecture Notes in Comput. Sci.*, pages 247–257. Springer, Berlin, 1989.
15. W. Gröbner. Über irreduzible Ideale in kommutativen Ringen. *Mathematische Annalen*, 110(1):197–222, 1935.
16. F. S. Macaulay. Modern algebra and polynomial ideals. *Mathematical Proceedings of the Cambridge Philosophical Society*, 30(1):27–46, 1934.
17. M. G. Marinari, T. Mora, and H. M. Möller. Gröbner bases of ideals defined by functionals with an application to ideals of projective points. *AAECC*, 4:103–145, 1993.
18. G. Moreno-Socías. *Autour de la fonction de Hilbert-Samuel (escaliers d'ideaux polynomiaux)*. PhD thesis, École polytechnique, 1991.
19. B. Mourrain. Fast algorithm for border bases of artinian gorenstein algebras. *ArXiv e-prints*, May 2017.
20. V. Neiger. *Bases of relations in one or several variables: fast algorithms and applications*. PhD thesis, École Normale Supérieure de Lyon, November 2016.
21. F. Rouillier. Solving zero-dimensional systems through the rational univariate representation. *AAECC*, 9(5):433–461, 1999.
22. S. Sakata. Extension of the Berlekamp-Massey algorithm to N dimensions. *Information and Computation*, 84(2):207–239, 1990.
23. V. Shoup. A new polynomial factorization algorithm and its implementation. *Journal of Symbolic Computation*, 20(4):363–397, 1995.