Linear Algebra for Computing Gröbner Bases of Linear Recursive Multidimensional Sequences

Jérémy Berthomieu, Brice Boyer, Jean-Charles Faugère

Linear Algebra for Computing Gröbner Bases of Linear Recursive Multidimensional Sequences

Jérémy Berthomieu [a,b,c] jeremy.berthomieu@lip6.fr
Brice Boyer [a,b,c] brice.boyer@lip6.fr
Jean-Charles Faugère [c,a,b] jean-charles.faugere@inria.fr

[a] Sorbonne Universités, UPMC Univ Paris 06, Équipe POLSYS, LIP6, F-75005, Paris, France
[b] CNRS, UMR 7606, LIP6, F-75005, Paris, France
[c] INRIA, Équipe POLSYS, Centre Paris – Rocquencourt, F-75005, Paris, France

ABSTRACT

Sakata generalized the Berlekamp – Massey algorithm to n dimensions in 1988. The Berlekamp – Massey – Sakata (BMS) algorithm can be used for finding a Gröbner basis of a 0-dimensional ideal of relations verified by a table. We investigate this problem using linear algebra techniques, with motivations such as accelerating change of basis algorithms (FGLM) or improving their complexity.

We first define and characterize multidimensional linear recursive sequences for 0-dimensional ideals. Under genericity assumptions, we propose a randomized preprocessing of the table that corresponds to performing a linear change of coordinates on the polynomials associated with the linear recurrences. This technique then essentially reduces our problem to using the efficient 1-dimensional Berlekamp – Massey (BM) algorithm. However, the number of probes to the table in this scheme may be elevated. We thus consider the table in the *black-box* model: we assume probing the table is expensive and we minimize the number of probes to the table in our complexity model. We produce an FGLM-like algorithm for finding the relations in the table, which lets us use linear algebra techniques. Under some additional assumptions, we make this algorithm adaptive and reduce further the number of table probes. This number can be estimated by counting the number of distinct elements in a multi-Hankel matrix (a multivariate generalization of Hankel matrices); we can relate this quantity with the *geometry* of the final staircase. Hence, in favorable cases such as convex ones, the complexity is essentially linear in the size of the output. Finally, when using the LEX ordering, we can make use of fast structured linear algebra similarly to the Hankel interpretation of Berlekamp – Massey.

Categories and Subject Descriptors

I.1.2 [**Symbolic and Algebraic Manipulation**]: Algorithms

General Terms

Theory, Algorithms

Keywords

BMS and FGLM algorithms, Gröbner basis computation, 0-dimensional ideal, multidimensional linear recursive sequence

1. INTRODUCTION

A fundamental problem in Computer Science is to estimate the linear complexity of an infinite sequence S: this is the smallest length of a recurrence satisfied by S or the length of the shortest linear feedback shift register (LFSR) which generates it. From an algorithmic point of view, the Berlekamp – Massey algorithm (BM) [1, 14] solves this problem in the one dimensional case. Generalizations of linear recurrence sequences to n dimensions were proposed by several authors [3, 17, 19]. Sakata generalized the BM algorithm [19, 21] to n dimensions; in particular, the so-called BMS algorithm is able to compute a Gröbner basis [19, Lem. 5] of the ideal of relations satisfied by the input sequence.

Direct and important application of such generalization can be found in Coding Theory: the BMS algorithm can be used to decode n-dimensional cyclic codes [20] which are generalization of Reed Solomon codes. Another application is the computation of Gröbner bases since recent versions of the Sparse-FGLM algorithm [5] rely heavily on BM and BMS algorithms.

Related work

Linear Prediction dates back to Gauß in the 18th century: given a discrete set of original values $(u_i)_{i\in\mathbb{N}}$ the goal is to find the best coefficients, in the least-squares sense, $(\alpha_i)_{i\in\mathbb{N}}$ that will approximate u_i by $-\sum_{k=1}^{d}\alpha_{n-k}u_k$. This problem is equivalent to solving a linear system which is indeed a symmetric Toeplitz matrix. This problem has been extensively used in Digital Signal Processing theory and applications. In the numerical world, methods such as the Levinson – Durbin recursion can be used to solve this problem. Hence, to some extent, the original Levinson – Durbin problem in Norbert Wiener's Ph.D. thesis [13, 22] predates the Hankel interpretation of the Berlekamp – Massey algorithm (for instance [7]).

We refer to [9, 10] for a very nice classification of the BM algorithms for solving this problem and generalization to matrix sequences, see also [8]. Of particular importance for us is the solution of the underlying linear system in Toeplitz/Hankel form. Let us also mention that a call of BM on sequence $(u_0, u_1, \ldots, u_{2d-1})$ will behave as the extended Euclidean algorithm with input polynomials x^{2d} and $u_0x^{2d-1} + u_1x^{2d-2} + \cdots + u_{2d-1}$ making BM a simplified version of the extended Euclidean algorithm.

BMS extends the algebraic form of BM to n dimensions [18]. In the case of 0-dimensional ideals, this algorithm can be applied for Gröbner basis computations [19].

Contributions

First of all, we define and characterize linear recurrence sequences in n dimensions. More precisely, we link them with 0-dimensional ideals and define their order as the degree of the ideal generated by the relations satisfied by the sequence (see Sec. 2). Classically, this number is also the size of the staircase of the Gröbner basis (the canonical set of generators for the residue class ring).

A first idea is to try to use the standard BM algorithm to solve the n dimensional case: we give a randomized preprocessing on the input sequence; this preprocessing will yield a new table which,

with good probability, will only have one linear recurrence relation (see Sec. 3). Exploiting this property yields Th. 1.

THEOREM 1. *Let $\mathbf{u} = (u_{i_1,\dots,i_n})_{i_1,\dots,i_n\in\mathbb{N}}$ be a n-dimensional linear recursive sequence over $\mathbb{K}$. Let $d \in \mathbb{N}$. When the size of $\mathbb{K}$ is large enough, we can find an equivalent basis of its relations for all $i_1+\cdots+i_n \le 2d$ in randomized time in $O(n^{2d} + n\mathsf{M}(d)\log d)$ operations in $\mathbb{K}$, where $\mathsf{M}(d)$ is the complexity of multiplying two polynomials of degree at most $d-1$.*

Next, we propose two FGLM-like algorithms for computing Gröbner bases of the ideal of relations in Sec. 4 and 5. Both algorithms are based only on simple linear algebra operations: they search to extract maximal full rank submatrices of a multi-Hankel matrix (a multivariate generalization of Hankel matrices). Given a bound on the maximal degree of the elements in the final Gröbner basis, the first algorithm is able to compute it. This algorithm is efficient when the order of the sequence is relatively big. On the other hand, when the order of sequence is abnormally small we propose an output sensitive probabilistic algorithm: this time an estimate of the order of the sequence is given.

An important parameter of the complexity of the algorithms is the number of table queries. Indeed, in some applications, it is very costly to compute *one* element $u_{i_1,i_2,\dots}$ of the table; thus the number of table queries has to be minimized. For instance, in the FGLM application, each element of the table requires a matrix-vector product to be computed. This number can be estimated by counting the number of distinct elements in a multi-Hankel matrix; moreover, we can relate this quantity with the geometry of the final staircase:

THEOREM 2. *The number of queries to the table is the cardinal of set $2S = \{uv \mid (u,v) \in S^2\}$ where S is the staircase of the ideal.*

We show that in favorable cases such as convex ones, the complexity is essentially linear in the size of the output. However, we also exhibit pathological cases where the complexity grows quadratically.

In Sec. 5.2 and 5.3, to illustrate the efficiency of the proposed algorithms, we report experiments for two applications: Sparse-FGLM and the decoding of n-dimensional cyclic codes. The results of the experiments are fully in line with the theory: for instance, in coding theory, when t errors are generated randomly, they can be recovered in $O(t)$ evaluation of the syndromes.

For the LEX ordering, multi-Hankel matrices are heavily structured and can be solved with fast algorithms. In Sec. 6, we give two approaches based on the notion of displacement rank and the polynomial multiplication interpretation, we refer to [2] for both. If d_i is the maximal degree of the polynomials in x_i, then solving the mth multi-Hankel matrix can be done in $O(\mathsf{M}(2^{m-1}d_1\cdots d_m))$ operations in the base field.

Finally, we left as an open question whether our algorithms could be seen as a matrix version of BMS.

2. DEFINITION AND CHARACTERIZATION OF LINEAR RECURSIVE SEQUENCES

In Def. 2, we generalize the notion of a linear recursive sequence from the 1-dimensional case to the multidimensional case based on the terms of the sequence. We also give a characterisation of such a sequence based on the linear recurrence relations [17, Def. 21] the sequence sastifies and provide a proof in Prop. 3 that this characterization is equivalent to Def. 2. This characterization is related to but more restrictive than [3, Def. 2]. Finally, in Sec. 2.2, we adopt an FGLM viewpoint to describe a multidimensional sequence.

Let us recall that a 1-dimensional sequence $\mathbf{u} = (u_i)_{i\in\mathbb{N}}$ defined over a field $\mathbb{K}$ is linear recursive of *order* $d > 0$ if $\exists\, \alpha_0, \dots, \alpha_{d-1} \in \mathbb{K}$ s.t. for all $i \in \mathbb{N}$, $u_{i+d} + \sum_{k=0}^{d-1} \alpha_k u_{i+k} = 0$ and d is minimal. We shall denote $R(i) = u_{i+d} + \sum_{k=0}^{d-1} \alpha_k u_{i+k}$. In general, $\mathrm{Pol}_{\mathbf{u}}(R)(x) = x^d + \sum_{k=0}^{d-1} \alpha_k x^k$ is called the *characteristic polynomial* of the sequence. It is well-known that the vector space of linear sequences verifying the *linear recurrence relation* R has dimension d whose canonical basis is $(\mathbf{u}^{(0)}, \dots, \mathbf{u}^{(d-1)})$ verifying $\forall\, i,e,\ 0 \le i,e \le d-1$, $u_i^{(e)} = \delta_{i,e}$, Kronecker's delta. Other bases are natural, let us mention the one using the roots of $\mathrm{Pol}_{\mathbf{u}}(R)$: $(\mathbf{u}^{(0,0)}, \dots, \mathbf{u}^{(0,\mu_0-1)}, \dots, \mathbf{u}^{(r-1,\mu_{r-1})})$ verifying $u_i^{(e,m)} = i^m \zeta_e^i$, where ζ_e is a root of multiplicity μ_e. In other words, a linear sequence of order d is uniquely determined by its characteristic polynomial, or equivalently its minimal linear recurrence relation of order d, and its d first terms.

To simplify notations, let $\mathbf{i} = (i_1, \dots, i_n) \in \mathbb{N}^n$ and $\mathbf{x} = (x_1, \dots, x_n)$. As usual, we let $\mathbf{x}^{\mathbf{i}}$ denote $x_1^{i_1}\cdots x_n^{i_n}$ and $|\mathbf{i}| = i_1 + \cdots + i_n$. We also denote e_i the ith vector of the canonical basis of $\mathbb{Z}^n$. In the following, we consider a n-dimensional sequence $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$.

As in the one-dimensional case, we can define a linear recurrence relation and its associated polynomial. Such a relation was called *n-dimensional linear recursion relation* in [17, Def. 21].

DEFINITION 1. *Let $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ be a sequence with coefficients in $\mathbb{K}$. Let $\mathscr{K} \subset \mathbb{N}^n$ be finite. The set $\{\alpha_{\mathbf{k}},\ \mathbf{k} \in \mathscr{K}\}$ defines a* linear recurrence relation *R if for all $\mathbf{i} \in \mathbb{N}^n$, $R(\mathbf{i}) = \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}} u_{\mathbf{i}+\mathbf{k}} = 0$.*

The associated polynomial *is then $\mathrm{Pol}_{\mathbf{u}}(R)(\mathbf{x}) = \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}}\mathbf{x}^{\mathbf{k}}$. Conversely, from any polynomial $P \in \mathbb{K}[\mathbf{x}], P = \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}}\mathbf{x}^{\mathbf{k}}$, we define the non-instantiated associated relation $\mathrm{Rel}_{\mathbf{u}}(P)(\mathbf{i}) = \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}} u_{\mathbf{i}+\mathbf{k}}$.*

In the end we can always shift any polynomials so that it is enough to evaluate such a relation in $\mathbf{i} = (0,\dots,0)$, consequently we will use the following convention: $\left[x_1^{\alpha_1}\cdots x_n^{\alpha_n}\right]_{\mathbf{u}} = [x_1^{\alpha_1}\cdots x_n^{\alpha_n}] = u_{\alpha_1,\dots,\alpha_n}$, and in general $[P]_{\mathbf{u}} = [P] = u_{\mathbf{d}} + \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}} u_{\mathbf{k}}$.

EXAMPLE 1. *If $P(x,y) = xy - x - 1 \in \mathbb{K}[x,y]$ then $[P] = u_{1,1} - u_{1,0} - u_{0,0}$ and $[x^2yP] = u_{3,2} - u_{3,1} - u_{2,1}$.*

We are now in a position to define a linear recursive sequence.

DEFINITION 2. *Let $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ be a n-dimensional sequence with coefficients in $\mathbb{K}$. The sequence $\mathbf{u}$ is* linear recursive *if from a nonzero finite number of initial terms $u_{\mathbf{i}}, \mathbf{i} \in S$ and a finite number of linear recurrence relations, without any contradiction, one can compute any term of the sequence.*

For contradictions arising, we refer to Sakata's example with initial terms $\{u_{0,0}, u_{1,0}, u_{0,1}\}$ and relations $u_{i+2,j} - u_{i,j} = u_{i+1,j+1} - u_{i,j} = u_{i,j+2} - u_{i,j} = 0$, in [21, p. 147]. Indeed, one can derive a new relation $u_{i+1,j} - u_{i,j+1} = 0$ meaning that $u_{0,1}$ is determined by the knowledge of $u_{1,0}$ and the set of initial terms is only $\{u_{0,0}, u_{1,0}\}$.

In other words, a linear recursive sequence is a special case of a *holonomic* (or *P-recursive*) sequence whose recurrence relations only have constant coefficients (see [11]).

2.1 Gröbner bases and characterization of linear recursive sequences

Before giving another characterization of such sequences based on Gröbner bases, we recall some definitions and properties of Gröbner bases and admissible monomial orders.

An *admissible monomial order* $\prec$ is an order on monomials of $\mathbb{K}[\mathbf{x}]$ s.t. for any monomial $s \neq 1$, $1 \prec s$ and for any monomials t, m, s.t. $t \prec s$, $ms \prec mt$. This implies that there does not exist any infinite strictly decreasing sequences of monomials.

The leading term of a polynomial P for $\prec$, denoted $\mathrm{LT}_{\prec}(P)$ or $\mathrm{LT}(P)$ if no confusion can arise, is the greatest monomial of P multiplied by its coefficient.

Whenever an ideal I is *homogeneous*, i.e. spanned by homogeneous polynomials, a *truncated Gröbner basis of I up to degree d* for $\prec$, or *d-truncated Gröbner basis*, is a set of polynomials $\mathscr{G} = \{g_1, \dots, g_r\}$ s.t. for all $f \in I$, if $\deg f \le d$ then there exists $g_i \in \mathscr{G}$ s.t. $\mathrm{LT}(g_i)$ divides $\mathrm{LT}(f)$. This can be computed using any Gröbner basis algorithm by discarding critical pairs of degree greater than d.

For an affine ideal, we can also define a d-truncated Gröbner basis as the output of a Gröbner basis algorithm discarding critical pair of degree higher than d. That is to say, for any critical pair (f_i, f_j), if $\deg \mathrm{LT}(f_i) + \deg \mathrm{LT}(f_j) - \deg \mathrm{lcm}(\mathrm{LT}(f_i), \mathrm{LT}(f_j)) > d$, then (f_i, f_j) is not taken into account. In this situation, a truncated Gröbner basis $\mathscr{G} = \{g_1, ..., g_r\}$ up to degree d will span the subspace of polynomials $\sum_{i=1}^{r} h_i g_i$ with $\deg h_i \leq d - \deg g_i$.

PROPOSITION 3. *Equivalently, a n-dimensional sequence defined over a field $\mathbb{K}$ is linear recursive if the ideal I spanned by all the polynomials associated to its linear recurrence relations has dimension 0, i.e. has a finite number of solutions in the algebraic closure of $\mathbb{K}$.*

PROOF. We shall prove that both definitions are equivalent. First, let us consider a sequence verifying Def. 2. Let $\prec$ be a monomial ordering. For a linear recurrence relation R with support in $\mathscr{K}$, there is a maximal element $\mathbf{d}$ in $\mathscr{K}$ for $\prec$ s.t. R is reduced to $R'(\mathbf{i}) = u_{\mathbf{i}+\mathbf{d}} + \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}} u_{\mathbf{i}+\mathbf{k}} = 0, \forall \mathbf{i} \in \mathbb{N}^n$. Let $R_1, ..., R_m$ be a set of linear recurrence relations, written as above, sufficient to compute $\mathbf{u}$ together with $\{u_{\mathbf{i}}, \mathbf{i} \in S\}$. Let us show that for all $\mathbf{i}$, there exist $\gamma_{\mathbf{i},\mathbf{k}}, \mathbf{k} \in S$ s.t. $u_{\mathbf{i}} - \sum_{\mathbf{k}\in S} \gamma_{\mathbf{i},\mathbf{k}} = 0$. Obviously, this is true if $\mathbf{i} \in S$. Let us assume this is true for any $\mathbf{k}' \prec \mathbf{i}$. As $u_{\mathbf{i}}$ can be computed from the terms before, there exists a finite set S' s.t. for all $\mathbf{k}' \in S'$, $\mathbf{k}' \prec \mathbf{i}$ and $\beta_{\mathbf{k}'} \in \mathbb{K}$ s.t. $u_{\mathbf{i}} - \sum_{\mathbf{k}'\in S'} \beta_{\mathbf{k}'} u_{\mathbf{k}'} = u_{\mathbf{i}} - \sum_{\mathbf{k}'\in S'} \beta_{\mathbf{k}'} \sum_{\mathbf{k}\in S} \gamma_{\mathbf{k}',\mathbf{k}} u_{\mathbf{k}} = 0$. This leads to I being spanned by polynomials $\mathbf{x}^{\mathbf{i}} - \sum_{\mathbf{k}S} \gamma_{\mathbf{i},\mathbf{k}} \mathbf{x}^{\mathbf{k}}$ for all $\mathbf{k} \prec \mathbf{i}$. These polynomials form a Gröbner basis for $\prec$ with a finite staircase, namely monomials $\mathbf{x}^{\mathbf{k}}, \mathbf{k} \in S$. Hence $\dim I = 0$.

Conversely, let $\mathscr{G} = \{G_1, ..., G_m\}$ be a minimal reduced Gröbner basis of I for a monomial order $\prec$. There exists a finite subset S of $\mathbb{N}^n$ s.t. for all j, $1 \leq j \leq m$, $G_j = \mathbf{x}^{\mathbf{i}_j} - \sum_{\mathbf{k}\in S} \gamma_{\mathbf{i}_j,\mathbf{k}} \mathbf{x}^{\mathbf{k}}$ with $\gamma_{\mathbf{i}_j,\mathbf{k}} \in \mathbb{K}$. Let us prove we can set a finite number of terms of $\mathbf{u}$ and then compute any term. Let $u_{\mathbf{i}}$ be any term of the sequence. If $\mathbf{x}^{\mathbf{i}}$ is in the staircase of $\mathscr{G}$, then we set $u_{\mathbf{i}}$. Otherwise there exist j and $\mathbf{i}'$ s.t. $\mathbf{x}^{\mathbf{i}} = \mathbf{x}^{\mathbf{i}'} \mathbf{x}^{\mathbf{i}_j}$, hence $\mathbf{x}^{\mathbf{i}'} G_j = \mathbf{x}^{\mathbf{i}} - \sum_{\mathbf{k}\in S} \gamma_{\mathbf{i}_j,\mathbf{k}} \mathbf{x}^{\mathbf{i}'+\mathbf{k}} \in I$. By recurrence on the $\mathbf{x}^{\mathbf{i}'+\mathbf{k}} \prec \mathbf{x}^{\mathbf{i}}$, there exist $\alpha_{\mathbf{i},\mathbf{k}} \in \mathbb{K}$ s.t. $\mathbf{x}^{\mathbf{i}} - \sum_{\mathbf{k}\in S} \alpha_{\mathbf{i},\mathbf{k}} \mathbf{x}^{\mathbf{k}} \in I$. Therefore $u_{\mathbf{i}} - \sum_{\mathbf{k}\in S} \alpha_{\mathbf{i},\mathbf{k}} u_{\mathbf{k}} = 0$ and one can compute any $u_{\mathbf{i}}$ from a finite number of initial terms. □

In other words, if $\mathbf{u}$ is linear recursive, then $\mathbb{K}[\mathbf{x}]/I$ is a finite dimensional $\mathbb{K}$-vector space. This is related to the definition of *holonomic* function in an Ore algebra $\mathscr{A} = \mathbb{K}(\mathbf{z})\langle \partial_{\mathbf{z}} \rangle$, see [11, Def. 1]. If $\mathrm{Ann}(f)$ is the left ideal of polynomials vanishing on $f = \sum_{\mathbf{i}\in\mathbb{N}^n} u_{\mathbf{i}} \mathbf{z}^{\mathbf{i}}$, then $\mathscr{A}/\mathrm{Ann}(f)$ is a finite dimensional vector space over $\mathscr{A}$. This equivalent definition is also related to but more restrictive than [3, Def. 2], in which the author only assumes that the ideal is not reduced to zero.

EXAMPLE 2. *Consider the following sequences:*
(a) $u_{i,j} = (2^i + 3^i)5^j$ *for all $i, j \in \mathbb{N}$ is linear recursive of order 2, a minimal set of linear recurrence relations is $\{u_{i+2,j} - 5u_{i+1,j} + 6u_{i,j} = 0, u_{i,j+1} - 5u_{i,j} = 0\}$. The associated ideal $\langle (x-2)(x-3), y-5 \rangle$ has two solutions of multiplicity 1.*
(b) $u_{i,j} = (i+1)2^i 5^j$ *for all $i, j \in \mathbb{N}$ is linear recursive of order 2, a minimal set of linear recurrence relations is $\{u_{i+2,j} - 4u_{i+1,j} + 4u_{i,j} = 0, u_{i,j+1} - 5u_{i,j} = 0\}$. The associated ideal $\langle (x-2)^2, y-5 \rangle$ has one solution of multiplicity 2.*
(c) $u_{i,j} = \binom{i}{j}$ *for all $i, j \in \mathbb{N}$ is* not *linear recursive, however it is* holonomic. *Indeed, while calling* BMS *or our algorithms on this sequence for all $u_{i,j}$, $i + j \leq 2d$, one obtains relations $u_{i+1,j+1} - u_{i,j+1} - u_{i,j} = 0$, the famous Pascal's triangle relation, together with $\sum_{k=0}^{d} \binom{d}{k} (-1)^k u_{i+k,j} = 0$ and $u_{i,j+d} = 0$. From the polynomial point of view, they form a d-truncated Gröbner basis of $I = \langle xy - y - 1, (x-1)^d, y^d \rangle = \langle 1 \rangle$. Let us notice that 1 is reached by these polynomials only as a linear combinations of degree $d+1$ and the ideal $\langle 1 \rangle$ has not dimension 0 but -1. From the sequence point of view, one needs to add infinitely many initial conditions to compute the whole sequence.*

DEFINITION 3. *Let $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ be a recursive linear sequence and I be its ideal of relations. The* order *of $\mathbf{u}$ is equivalently the minimal number of initial terms of $\mathbf{u}$ needed to compute any term or the degree of I.*

2.2 Matrix multiplications in the quotient ring point of view

In this section, with a FGLM point of view, we shall show that given the ideal of relations and the initial terms of a sequence, one can express all the terms of said sequence as a scalar product. This point of view will be a key for the adaptive algorithm designed in Sec. 5.

Let $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ be a linear recursive sequence over $\mathbb{K}$. Let S be the staircase of a Gröbner basis of its ideal of relations I, then $\mathbb{K}[\mathbf{x}]/I$ is a $\mathbb{K}$-algebra whose canonical basis as a vector space is made of the monomials $\mathbf{x}^{\mathbf{i}} \in S$. Defining $T_1, ..., T_n$ the multiplication matrices by respectively $x_1, ..., x_n$ in $\mathbb{K}[\mathbf{x}]/I$, then $u_{\mathbf{i}} = \langle \mathbf{r}, T_1^{i_1} \cdots T_n^{i_n} \cdot \mathbf{1} \rangle$, where $\mathbf{r} = (u_{\mathbf{0}}, ...) = (u_s, s \in S)$, the vector of initial conditions, and $\mathbf{1} = (1, 0, ..., 0)^{\mathrm{T}}$, the vector representing 1 in the canonical basis of $\mathbb{K}[x_1, ..., x_n]/I$.

Indeed, let $\mathbf{i} \in \mathbb{N}^n$. By definition, $T_1^{i_1} \cdots T_n^{i_n} \cdot \mathbf{1}$ is the vector representing $\mathbf{x}^{\mathbf{i}}$ in the canonical basis of $\mathbb{K}[x_1, ..., x_n]/I$. Then, the scalar product allows one to map this vector to the corresponding linear combination of the initial terms of the sequence.

3. RANDOMIZED REDUCTION: BMS/BM

In this section, we exploit a randomized preprocessing on the table which can simplify the computation of the linear recurrence relations of the sequence. This preprocessing will yield a new table which, with good probability, will have one linear recurrence relation of the form $u_{\mathbf{i}+de_1} - \sum_{k=0}^{d-1} \alpha_k u_{\mathbf{i}+ke_1} = 0$, for all $\mathbf{i}$ and other relations of the type $u_{\mathbf{i}+e_j} - \sum_{k=0}^{d-1} \beta_{j,k} u_{\mathbf{i}+ke_1} = 0$. In other words, all the other variables can be deduced from the first one. Therefore, the bottleneck of the execution of BMS on this sequence would be the computation of the first relation. This comes down essentially to running BM on the subsequence $(u_{ie_1})_{i\in\mathbb{N}}$.

3.1 Linear Transformation of the Table

This preprocessing can be seen as a linear transformation on the variables appearing in the ideal of relations.

Let $A \in \mathrm{GL}_n(\mathbb{K})$. Let us denote $\ell_i = \sum_{j=1}^{n} a_{i,j} x^j$, the ith linear form of $A\mathbf{x}$. Then $(A\mathbf{x})^{\mathbf{i}} = \prod_{j=1}^{n} \ell_j^{i_j}$, with $\mathbf{x}^{\mathbf{i}} = x_1^{i_1} \cdots x_n^{i_j}$. We define the action of A on an n-dimensional sequence as follows.

DEFINITION 4. *Let $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ be a linear recurrent sequence. We define a* change of basis *on $\mathbf{u}$ as an invertible matrix $A \in \mathrm{GL}_n(\mathbb{K})$. The sequence $\mathbf{v} = (v_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n} = A \cdot \mathbf{u}$ is defined as $v_{\mathbf{i}} = \left[(A\mathbf{x})^{\mathbf{i}}\right]_{\mathbf{u}}$.*

In other words, for instance, $v_{\mathbf{0}} = u_{\mathbf{0}}, v_{e_i} = \sum_{j=1}^{n} a_{i,j} u_{e_j}$, etc. The following proposition links the ideals of relations of $\mathbf{u}$ and $\mathbf{v}$.

PROPOSITION 4. *Let P be a polynomial associated to a relation of $\mathbf{u}$. Let $\mathbf{v} = A \cdot \mathbf{u}$ for A invertible. Then $P(A^{-1}\mathbf{x})$ is a polynomial associated to a relation of $\mathbf{v}$.*

PROOF. Since $v_{\mathbf{i}}$ is merely the polynomial $(A\mathbf{x})^{\mathbf{i}}$ evaluated in $\mathbf{u}$. Any polynomial $P(A^{-1}\mathbf{x})$ evaluated in $\mathbf{v}$ will yield $P(\mathbf{x})$ evaluated in $\mathbf{u}$. Therefore, $\left[P(A^{-1}\mathbf{x})\right]_{\mathbf{v}} = 0$ iff $\left[P(\mathbf{x})\right]_{\mathbf{u}} = 0$. □

3.2 Essential reduction to BM

Let $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ be a n-dimensional linear recursive sequence. Let $\mathbf{v} = (v_{\mathbf{i},j})_{(\mathbf{i},j)\in\mathbb{N}^n\times\mathbb{N}}$ be the $(n+1)$-dimensional sequence defined by $v_{\mathbf{i},j} = u_{\mathbf{i}}$ for all $\mathbf{i} \in \mathbb{N}^n, j \in \mathbb{N}$. If $I \in \mathbb{K}[\mathbf{x}]$ is the ideal of relations of $\mathbf{u}$ and t is a new variable representing the last coordinate then $J = I + (t-1)$ is the ideal of relations of $\mathbf{v}$. Let us now apply the change of coordinates in which each x_i remains the same

and t is mapped onto $t+\sum_{i=1}^{n} c_i x_i$ for some $c_i \in \mathbb{K}$. Generically, the minimal reduced Gröbner basis of the new ideal J' obtained from J for the LEX order with $t < x_1 < \cdots < x_n$ is in *shape position*, i.e. is $\langle f(t), x_1 - f_1(t), \ldots, x_n - f_n(t)\rangle$, with $\deg f_i < \deg f$, see [6, 12]. As $f(t) = \sum_{j=0}^{d} \alpha_j t^j$ depends only on t, it is found by running BM on the subsequence $(v_{\mathbf{0},j})_{j\in\mathbb{N}}$. Each polynomial $x_i - f_i(t)$, for $1 \le i \le n$, is then found by solving the linear system $u_{e_i,k+d} - \sum_{j=0}^{d-1} \beta_{i,j} u_{e_i,k+j}$ for $0 \le k \le d-1$, whose matrix is Hankel.

Therefore, after applying a linear transformation on the table, finding its relations essentially reduces to running BM on a 1-dimensional subsequence. This is summed up in Alg. 1.

ALGORITHM 1. BM *for n-dimensional sequences*
Input: *a n-dimensional sequence* $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$
Output: *an equivalent basis of relations of* $\mathbf{u}$
Pick at random $c_1, \ldots, c_n \in \mathbb{K}$
Compute $\tilde{\mathbf{v}} = (\tilde{v}_j)_{j\in\mathbb{N}}$ *with*
$\tilde{v}_j = \left[(t + c_1x_1 + \ldots + c_nx_n)^j\right]_{\mathbf{v}}$ // *with* $v_{\mathbf{i},j} = u_{\mathbf{i}}$, $\forall(\mathbf{i}, j) \in \mathbb{N}^n \times \mathbb{N}$
$= \left[(1 + c_1x_1 + \ldots + c_nx_n)^j\right]_{\mathbf{u}}$
Compute f *with* BM *on* $\tilde{\mathbf{v}}$ *of degree* d.
For i ***from*** 1 ***to*** n
Solve the Hankel linear system

$$\begin{pmatrix} u_{e_i,0} & \cdots & u_{e_i,d-1} \\ \vdots & \iddots & \vdots \\ u_{e_i,d-1} & \cdots & u_{e_i,2d-2} \end{pmatrix} \begin{pmatrix} \beta_{i,0} \\ \vdots \\ \beta_{i,d-1} \end{pmatrix} = \begin{pmatrix} u_{e_i,d} \\ \vdots \\ u_{e_i,2d-1} \end{pmatrix}$$

Return $f, x_1 - \sum_{j=0}^{d-1} \beta_{1,j} t^j, \ldots, x_n - \sum_{j=0}^{d-1} \beta_{n,j} t^j$.

Let us mention that one can also retrieve the relations of the original sequence, i.e. the polynomials in $x_1, \ldots, x_n$ not in t, by computing a new Gröbner basis of the ideal for an order eliminating t, e.g. LEX with $x_1 < \cdots < x_n < t$. Poteaux and Schost [15] proved that there is a Las Vegas algorithm to change the order of a triangular set whose complexity is essentially that of modular composition $O(\mathsf{C}(d)) \subseteq O(d^{(\omega+1)/2})$ operations, as d is the degree of I.

3.3 Complexity results

PROPOSITION 5. *Let* $d \in \mathbb{N}$. *Computing terms* $v_{\mathbf{i}}$ *for all* $\mathbf{i} \in \mathbb{N}^n$ *such that* $|\mathbf{i}| \le d$ *can be done in* $O(n^{2d})$ *operations in* $\mathbb{K}$, $O(n^{2d})$ *memory space and* $O(n^d)$ *queries to the table elements.*

PROOF. As seen in Sec. 3.1, to compute $v_{\mathbf{i}}$, one needs to compute $\ell_1^{i_1} \cdots \ell_n^{i_n}$, where ℓ_j is the jth row of $A\mathbf{x}$.

Let $z_0, \ldots, z_n$ be new variables; expanding $(z_0 + \ell_1 z_1 + \ldots + \ell_n z_n)^d$ yields terms of the form $c\ell_1^{i_1} \cdots \ell_n^{i_n} z_0^{d-|\mathbf{i}|} z_1^{i_1} \cdots z_n^{i_n}$ with $i_1, \ldots, i_n \ge 0$ and s.t. $|\mathbf{i}| \le d$. This allows us to directly determine all the polynomials we need to compute $v_{\mathbf{i}}$, $|\mathbf{i}| \le d$.

Since $z_0 + \ell_1 z_1 + \ldots + \ell_n z_n$ has n^2+1 monomials, its dth power has $\binom{n^2+1}{d} \in O(n^{2d})$. Using the square and multiply algorithms, one needs to perform $O(n^{2d})$ operations in $\mathbb{K}$ to compute this power. This method also needs to store every coefficient of our polynomials which is in $O(n^{2d})$.

Finally, we need to evaluate the polynomials by replacing $\mathbf{x}^{\mathbf{i}}$ by $u_{\mathbf{i}}$. For this, we only need to access each element of the table once, and update all our evaluations at the same time, hence $O(n^d)$ queries. As each of the $O(n^d)$ polynomials of degree d has $O(n^d)$ coefficients, $O(n^{2d})$ multiplications must be performed. □

PROOF OF TH. 1. Besides the change of basis in $O(n^{2d})$, we need one call of BM in $\tilde{O}(n^d)$ operations in $\mathbb{K}$. Finally, each Hankel system can be solved in $O(\mathsf{M}(d)\log d)$ operations (see [2]). □

4. MULTI-HANKEL SOLVER

This section is devoted to the design of a FGLM-like algorithm for computing the Gröbner bases of the ideal of relations of a sequence $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$, with coefficients in $\mathbb{K}$. From now on, $\mathscr{T}$ is the ordered set of terms that we can make from $x_1, \ldots, x_n$. We fix $\prec$ to be an admissible monomial ordering. If $P \in \mathbb{K}[x_1, \ldots, x_n]$ then $\mathscr{T}(P)$ is the set of terms composing P and $\mathrm{LT}_{\prec}(P)$ is the maximum of $\mathscr{T}(P)$. For any D, $\mathscr{T}_D$ is the set of all terms of degree $\le D$ sorted by increasing order (wrt. $\prec$). Since we want to design an algorithm we will be unable to check that a relation is valid for all $\mathbf{i} \in \mathbb{N}^n$. Indeed, at some point of the algorithm we will have a finite subset of indices $T \subset \mathbb{N}^n$ and we will try to find relations that are valid for those indices: for $\mathbf{i} \in T$, $u_{\mathbf{i}+\mathbf{d}} + \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}} u_{\mathbf{i}+\mathbf{k}} = 0$.

DEFINITION 5. *Let* T *be a finite subset of* $\mathbb{N}^n$. *We say that a polynomial* $P \in \mathbb{K}[x_1, \ldots, x_n]$ *is valid up to* T *if* $\mathrm{Rel}_{\mathbf{u}}(P)(\mathbf{i}) = 0$ *for all* $\mathbf{i} \in T$. *In that case we write that* $\mathrm{NF}(P, \mathbf{u}, T) = 0$.

Let T *be a finite subset of* $\mathscr{T}$. *We say that a polynomial* $P \in \mathbb{K}[x_1, \ldots, x_n]$ *is valid up to* T *if* $[tP] = 0$ *for all* $t \in T$. *In that case we write that* $\mathrm{NF}(P, \mathbf{u}, T) = 0$.

4.1 Staircase

We assume now that $T \subset \mathscr{T}$ is a finite set of terms. Equivalently, T' is the set of exponents of all $t \in T$. Any BMS-style algorithm will generate *minimal* relations; hence when we try to establish a new relation $u_{\mathbf{i}+\mathbf{d}} + \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}} u_{\mathbf{i}+\mathbf{k}}$ we must check two properties: *(a)* There are scalars $\alpha_{\mathbf{k}} \in \mathbb{K}$ so that $\forall \mathbf{i} \in T', u_{\mathbf{i}+\mathbf{d}} + \sum_{\mathbf{k}\in\mathscr{K}} \alpha_{\mathbf{k}} u_{\mathbf{i}+\mathbf{k}} = 0$; *(b)* There are no nonzero relations $\sum_{\mathbf{k}\in\mathscr{K}} \beta_{\mathbf{k}} u_{\mathbf{i}+\mathbf{k}} = 0$ which are valid for all $\mathbf{i} \in T'$.

We can translate these properties in polynomial terms: *(a)* There is a monic polynomial $P \in \mathbb{K}[x_1, \ldots, x_n]$ of leading term $\mathbf{x}^{\mathbf{d}}$ s.t. $\mathrm{NF}(P, \mathbf{u}, T) = 0$; *(b)* There are no nonzero relations $\sum_{t\in\mathscr{T}(P-\mathbf{x}^{\mathbf{d}})} \beta_t \mathrm{Rel}_{\mathbf{u}}(t)(\mathbf{i}) = 0$ which are valid for all $\mathbf{i} \in T'$. Equivalently, there are no nonzero relations $\sum_{t\in\mathscr{T}(P-\mathbf{x}^{\mathbf{d}})} \beta_t [mt] = 0$ which are valid for all $m \in T$.

Hence it is important to identify a set of terms for which there is *no* linear relations.

DEFINITION 6. *Let* T *be a finite subset of* $\mathscr{T}$. *We say that a finite set* $S \subset T$ *of terms is a useful staircase wrt.* $\mathbf{u}$, T *and* $\prec$ *if* $\sum_{t\in S} \beta_t [mt] = 0$, $\forall m \in S$ *implies that* $\beta_t = 0$ *for all* $t \in S$, S *is maximal for the inclusion and minimal for* $\prec$. *We compare two ordered sets for* $\prec$ *by seeing them as tuples of their elements and then comparing them lexicographically.*

Note that these "useful staircases" are not staircases in the sense of Gröbner bases since they are not always stable under division.

EXAMPLE 3. *In dimension* 2, *consider the set* $T = \{1, x, y, x^2, xy, y^2\}$ *of all degree* ≤ 2 *monomials and the table* $\mathbf{u} = \begin{pmatrix} 0&0&0&\cdots \\ 0&0&0&\cdots \\ 0&1&0&\cdots \\ 0&0&0&\ddots \end{pmatrix}$ *Then, we can check (see Ex. 4) that* $\{y, x, xy, x^2\}$ *is a useful staircase. Since* 1 *is not in this set, it is not stable under division. However, using the stability criterion, we recover the complete staircase* $\{1, x, y, xy, x^2\}$.

Alg. 2 transforms a useful staircase into a staircase.

ALGORITHM 2. *Stabilize*
Input: S' *a useful staircase*
Output: *a staircase*
$S := []$
For $s \in S'$ ***do*** $S := S \cup \{t \mid t \in \mathscr{T} \text{ and } t \text{ divides } s\}$
Return S

4.2 Linear Algebra to find relations

We give a simple algorithm to check that a finite set $S \subset T$ is a useful staircase wrt. $\mathbf{u}$, T and $\prec$. Let us start with a simple example:

EXAMPLE 4 (EX. 3 CONT.). *We look for a relation* $P(x,y) = a_5x^2 + a_4xy + a_3y^2 + a_2x + a_1y + a_0$ *that is to say we try to find* $a_0, \ldots, a_5$ *s.t.* $[tP] = 0$ *for all* $t \in T$: $a_0u_{k1,k2} + a_1u_{k1,k2+1} + a_2u_{k1+1,k2} + a_3u_{i_1,i_2+2} + a_4u_{i_1+1,i_2+1} + a_5u_{i_1+2,i_2} = 0$, *for all* (i_1, i_2) *s.t.* $i_1 + i_2 \le 2$. *To find a useful staircase* S *it is equivalent to extracting a full rank matrix in the following multi-Hankel matrix:*

$$H_T = \begin{array}{c} \\ 1 \\ y \\ x \\ y^2 \\ xy \\ x^2 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & y & x & y^2 & xy & x^2 \end{array} \\ \begin{pmatrix} u_{0,0} & u_{0,1} & u_{1,0} & u_{0,2} & u_{1,1} & u_{2,0} \\ u_{0,1} & u_{0,2} & u_{1,1} & u_{0,3} & u_{1,2} & u_{2,1} \\ u_{1,0} & u_{1,1} & u_{2,0} & u_{1,2} & u_{2,1} & u_{3,0} \\ u_{0,2} & u_{0,3} & u_{1,2} & u_{0,4} & u_{1,3} & u_{2,2} \\ u_{1,1} & u_{1,2} & u_{2,1} & u_{1,3} & u_{2,2} & u_{3,1} \\ u_{2,0} & u_{2,1} & u_{3,0} & u_{2,2} & u_{3,1} & u_{4,0} \end{pmatrix} \end{array}$$

or equivalently

$$H_T = \begin{array}{c} \\ 1 \\ y \\ x \\ y^2 \\ xy \\ x^2 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & y & x & y^2 & xy & x^2 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}.$$

In this example, 1 *and* y^2 *are clearly useless so that* $S' = \{y, x, xy, x^2\}$ *is the useful staircase and* $\det(H_{S'}) = 1 \neq 0$*. In addition we can try to find a relation* $Q(x,y) = y^2 - a_3 x^2 - a_2 xy - a_1 x - a_0 y$*. Again this is equivalent to finding* $a_0, \ldots, a_3$ *s.t.* $\mathrm{Rel}(Q)(i_1, i_2) = 0$ *for all* (i_1, i_2) *with* $i_1 + i_2 \leq 2$*. In turn, this reduces to solving the linear system:*

$$H_{S'} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} u_{0,3} \\ u_{1,2} \\ u_{1,3} \\ u_{2,2} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Since $H_{S'}$ *is full rank we find* $a_0 = \cdots = a_3 = 0$ *and the relation* $Q(x,y) = y^2$.

In the general case we can now define the structured matrix associated to two lists of terms:

DEFINITION 7. *Let* T *and* S *be two finite subsets of* $\mathscr{T}$*. We consider the polynomial* $P_S(\mathbf{x}) = \sum_{s \in S} a_s s$ *and the linear equations* $[t\, P_S] = 0$ *for all* $t \in T$*. Then, we generate the coefficient matrix* $H_{T,S}$ *from the previous linear system of equations in the unknown variables* a_s *for* $s \in S$*:*

$$H_{T,S} = \begin{array}{c} \\ \vdots \\ t \in T \\ \vdots \end{array} \begin{array}{c} \begin{array}{ccc} \cdots & s \in S & \cdots \end{array} \\ \begin{pmatrix} \cdots & & \cdots \\ \cdots & [t\,s]_{\mathbf{u}} & \cdots \\ \cdots & & \cdots \end{pmatrix} \end{array}.$$

When $T = S$ *we simply write* H_T *for the multi-Hankel matrix* $H_{T,S}$*.*

These two crucial operations involve linear algebra:
(a) Checking that a finite set of terms $S \subset \mathscr{T}$ is a useful staircase wrt. T (we assume that $\#T \geqslant \#S$) is equivalent to checking that the matrix $H_{T,S}$ has full rank;
(b) Finding a monic polynomial $P \in \mathbb{K}[x_1, \ldots, x_n]$ of given support s.t. $\mathrm{NF}(P, \mathbf{u}, T) = 0$ is equivalent to solving a linear system

$$H_{T,S} \times \mathbf{a} + H_{T,\{\mathrm{LT}(P)\}} = 0$$

where $S = \mathscr{T}(P - \mathrm{LT}(P))$ is the support of the polynomial P except the leading term. If $\mathbf{a}$ is a solution then $P = \mathrm{LT}(P) + \sum_{s \in S} a_s s$ is a polynomial s.t. $\mathrm{NF}(P, \mathbf{u}, T) = 0$.

PROPOSITION 6. *Let* T *be a finite subset of* $\mathscr{T}$*. If the finite set of terms* $S \subset \mathscr{T}$ *is a useful staircase wrt.* $\mathbf{u}$*,* T *and* $\prec$ *then:*

$$\det(H_S) \neq 0 \text{ and } \operatorname{rank} H_S = \operatorname{rank} H_{T,S} = \operatorname{rank} H_T.$$

PROOF. This is another wording of Def. 6. □

The two statements of Prop. 7 are easy to prove but they are the basis of the algorithm: according to them we know that we can proceed degree by degree.

PROPOSITION 7. *If* $S' \subset S$ *are finite set of terms, then* $\operatorname{rank} H_S \geq \operatorname{rank} H_{S'}$*. Moreover, assume* S *is a finite subset of terms s.t.* $\det(H_S) \neq 0$ *and* $t \in \mathscr{T} \backslash S$ *then we have:* $\operatorname{rank} H_{S \cup \{t\}} = \mathrm{ColRank}\, H_{S, S \cup \{t\}} = \mathrm{RowRank}\, H_{S \cup \{t\}, S}$.

PROOF. $H_{S'}$ is a submatrix of H_S and H_S is symmetric. □

We rely on the following naive strategy to extract a full rank matrix of a multi-Hankel matrix. We start with $H_{\{\}}$ and we proceed by induction, assuming that at some point we have found a useful S s.t. H_S is full rank. Then we select the minimal $t \in \mathscr{T} \backslash S$. If $\operatorname{rank} H_{S \cup \{t\}} = \operatorname{rank} H_S + 1$ then we update $S := S \cup \{t\}$; else we have $\operatorname{rank} H_{S \cup \{t\}} = \operatorname{rank} H_S$ and we consider $t' \in \mathscr{T} \backslash (S \cup \{t\})$.

Given a useful staircase S, it is important that $2S = \{st, s, t \in S\}$ be not too big compared to S, when counting the number of table queries. We will see how to bound the cardinality of $2S$ in Sec. 5.1.

4.3 An FGLM-like Algorithm

Since the input table $\mathbf{u}$ is infinite we need a bound given by the user: $d \geq 0$ and T will be the set of all monomials of degree less than d. Accordingly, we will assume that the monomial ordering $\prec$ is an admissible ordering refined by the total degree. Since the output of the BMS algorithm is a (truncated) Gröbner basis, it is a natural idea to try to adapt existing Gröbner basis algorithms to obtain the same result. To this end, we can try to slightly modify the FGLM algorithm. However, in the scalar case, a fundamental difference is that the structure of the quotient ring (and in particular the staircase) is not known. Hence, to clarify our intention we will split the algorithm in two parts: first we derive the staircase wrt. the monomial ordering and the given bound; in a second step we compute a truncated Gröbner basis. In a real implementation, the two steps can be combined to increase the efficiency of the algorithm.

ALGORITHM 3. FGLM *for scalars.*
Input: $\prec$ *a monomial ordering,* $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i} \in \mathbb{N}^n}$ *a sequence with coefficients in* $\mathbb{K}$*,* d *a given bound.*
Output: *a reduced* $(d+1)$*-truncated Gröbner basis wrt.* $\prec$ *of* $\mathbf{u}$

Build the matrix $H_{\mathscr{T}_d}$.
Find S' *the useful staircase s.t.* $\operatorname{rank} H_{\mathscr{T}_d} = \operatorname{rank} H_{S'}$. *// as in Sec. 4.2*
$S :=$ *Stabilize*(S') *// the staircase (stable under division)*
$L := \mathscr{T}_{d+1} \backslash S$ *// list of next terms to study*
$G := []$ *// the future Gröbner basis*
While $L \neq \emptyset$ **do**
  $t := \min_{\prec}(L)$ *and remove* t *from* L
  Find $\mathbf{x}$ *s.t.* $H_{S'}\mathbf{x} + H_{S',\{t\}} = 0$
  $G := G \cup \left[t + \sum_{i=1}^{\#S'} x_i \cdot S'_i \right]$
  Sort L *by increasing order (wrt.* $\prec$*) and remove multiples of* $\mathrm{LT}(G)$.
Return G

EXAMPLE 5 (CONT. OF EX. 2.C). *We consider the table* $\mathbf{u}$ *generated by* $u_{i,j} = \binom{i}{j}$ *and we fix* $d = 2$*. We consider the matrix*

$$H = \begin{array}{c} \\ 1 \\ y \\ x \\ y^2 \\ xy \\ x^2 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & y & x & y^2 & xy & x^2 \end{array} \\ \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 2 \\ 1 & 1 & 1 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 2 & 0 & 1 & 3 \\ 1 & 2 & 1 & 1 & 3 & 1 \end{pmatrix} \end{array}.$$

It is easy to check that the column (resp. row) xy *is the sum of the first two columns (resp. rows). Hence, the useful staircase is* $S' = \{1, y, x, y^2, x^2\}$*. Since* S' *is stable under division,* $S = S'$*. We initialize* $L = [xy, x^3, x^2y, xy^2, y^3]$ *so that* $t = xy$*. The next step is to solve the system* $H_S\mathbf{x} + [1,0,2,0,3]^{\mathrm{T}} = 0$ *and we find* $\mathbf{x} = [-1,-1,0,0,0]^{\mathrm{T}}$ *so that* $G = [xy - y - 1]$*. We can update* $L = [x^3, y^3]$ *and by solving two other linear systems, we find* $G = [xy - y - 1, x^3 - 3x^2 + 3x - 1 = (x-1)^3, y^3]$*. Clearly this is not a full Gröbner basis but a* 3*-truncated Gröbner basis.*

THEOREM 8. *The output of Alg. 3 is a* $(d+1)$*-truncated Gröbner basis. Moreover, if* $\mathbf{u}$ *is a recursive linear sequence of order* D*, taking* $d = D$ *suffices to recover a full Gröbner basis of the sequence.*

PROOF. Alg. 3 clearly terminates since the size of L decreases.

Taking the basis monomials in increasing order ensures that the set S' contains monomials of smallest degrees. Let us first show that the staircase of the ideal of u contains the useful staircase. In the following, we shall see polynomials as linear combinations of columns of $H_{\mathscr{T}_d}$. Any polynomial with leading term outside the staircase of the ideal of $\mathbf{u}$ reduces to polynomials in the staircase. Suppose that one element e of the useful staircase lies outside the staircase. Seen as linear combinations of columns of $H_{\mathscr{T}_d}$, it is then a linear combination of elements in the staircase, some of which are not in the useful staircase by linear independence. Let f be one of these particular elements: f is a linear combination of smaller elements in the useful staircase. Then e is actually a linear combination of elements in the useful staircase, which is contradictory.

Conversely, if S' does not contain a maximal (for the natural order on the table) element of the staircase for $\mathbf{u}$, then this element can be written as a linear combination of smaller terms, which contradicts the fact it belongs to the staircase. The stabilization of these maximal elements is therefore the full staircase of the ideal of $\mathbf{u}$.

The set G contains elements with leading terms that do not divide each other. Let us consider f and g in G (with leading terms in $\mathscr{T}_{d+1}$) and their S-polynomials $S(f,g)$. Then either the leading term of $S(f,g)$ is in $\mathscr{T}_{d+1} \setminus S'$ or it is in S'. In the latter case, it means there is a relation in $H_{S'}$, so it cannot be a new relation. In the former case, the relation was already found by the main loop. So no $S(f,g)$ produces a new relation. □

5. ADAPTIVE ALGORITHM

So far we have seen two algorithms to recover the relations from a table: these algorithms are efficient when the degree of the elements in the Gröbner basis G is small compared with the order of the recurrence: $\max \deg G \approx (\text{order of the recurrence})^{1/n}$.

Unfortunately, this is not always the case, especially if the monomial ordering is a lexicographical order. Using the previous algorithms on these examples would increase too much the complexity (by complexity we mean the number of accesses to the table $\mathbf{u}$). The goal of this section is to describe an adaptive algorithm to take into account the shape of the final Gröbner basis.

The main difference between Alg. 3 and the original FGLM is the following: with polynomials, when we discover a relation $f = t + \sum_{s\in S} \alpha_s s$ we know that mf is still a valid relation for any $m \in \mathscr{T}$. In contrast, when we find a relation

$$[f]_{\mathbf{u}} = [t]_{\mathbf{u}} + \sum_{s\in S} \alpha_s [s]_{\mathbf{u}} = 0 \tag{1}$$

it is not true in general that $[mf]_{\mathbf{u}} = 0$. However we know from Sec. 2.2 that any n-dimensional linear recurrence can be written as $u_{\mathbf{i}} = \langle \mathbf{r}, T_1^{i_1} \cdots T_n^{i_n} \cdot \mathbf{1} \rangle$ where $\mathbf{r}$ is a vector depending on the initial conditions and T_i are multiplication matrices associated to the Gröbner basis G. Therefore we can write $\langle \mathbf{r}, \text{NormalForm}(f,G) \rangle = 0$ for relation (1). Hence, if $\mathbf{r}$ is sufficiently random we know that $\text{NF}(f,G) = 0$ so that $\text{NF}(mf,G) = 0$ for all $m \in \mathscr{T}$ which implies that $[mf]_{\mathbf{u}} = 0$. Note that in some applications (see for instance Sec. 5.2) it is possible to check afterwards that the relation is correct. Accordingly, we propose an FGLM algorithm to take advantage of this property. We proceed term by term and we try to discover the new staircase which is equivalent to increasing the rank of the multi-Hankel matrix by 1. We do not give any bounds on the degree of the output polynomial but we assume that an estimate of the size of order of the recurrence is given by the user. We will see later that for many applications the complexity can be reduced drastically; depending on the shape (for instance the convexity) of the final staircase, the number of queries to the table can often be linear in the order of the recurrence, similarly to the one-dimensional case. In the following algorithm, for any list of terms G, $\text{MinGBasis}(G)$ is the corresponding minimal Gröbner basis.

ALGORITHM 4. *Adaptive* FGLM *for scalars (simple version)*
Input: *$\prec$ a monomial ordering, $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ a sequence with coefficients in $\mathbb{K}$, d a given bound.*
$L := [1]$ *// list of next terms to study*
$S := []$ *// the useful staircase wrt. the new ordering $\prec$*
$G' := []$ *// leading terms of the final Gröbner basis*
While $L \neq \emptyset$ **do**
  $t := first(L)$
  If $H_{S\cup\{t\}}$ *is full rank* **then**
    $S := S \cup [t]$ *and* $L := L \cup [x_i t \mid i = 1,\dots,n] \setminus [t]$
    Sort L (wrt. $\prec$) and remove duplicates and multiples of G'
    If $\#S \geq d$ **then** *// early termination*
      $G := []$ *and* $G' := \text{MinGBasis}(G' \cup L \cup \mathscr{T}_{\deg(t)+1})$
      For $t \in G'$ **do**
        $G := G \cup [t + \sum_{i=1}^{\#S} x_i \cdot S_i]$ *where* $\mathbf{x}$ *s.t.* $H_S \mathbf{x} + H_{S,\{t\}} = 0$
      Return *S and G*
  Else $G' := G' \cup [t]$ *and remove multiples of t in L*
Error *"Run Alg. 3"*

REMARK 9. *In some applications (see e.g. Sec. 5.3 on error correcting codes), the input table is bounded: $u_{i_1,\dots,i_n}$ cannot be computed or has no meaning when $i_j > B$ for some bound B. One can easily modify the algorithm to take this constraint into consideration.*

PROPOSITION 10. *Let S and G be the output of Alg. 4. Then S is a staircase of size $\geq d$ and G is a list of valid relations, that is to say $\text{NF}(f,\mathbf{u},S) = 0$ for all $f \in G$.*

EXAMPLE 6. *As explained in Sec. 5.2, we consider the ideal of the vanishing ideal of the points $\{[0,0],[0,1],[1,1]\}$. We compute a total degree Gröbner basis in $\mathbb{F}_{11}[x_1,x_2]$ and we apply the Sparse-*FGLM *algorithm with a random vector $\mathbf{r} = [10,3,5]$. Hence, we run Alg. 4 with $d = 3$ on the table $\mathbf{u} = \begin{pmatrix} 10 & 3 & 3 & \cdots \\ 5 & 5 & 5 & \\ 5 & 5 & 5 & \\ \vdots & & & \ddots \end{pmatrix}$.*
Step 1: *$t = 1$; we check $H_{[1]} = (10)$ has rank 1 so that $L = [x_2, x_1]$.*
Step 2: *$t = x_2$; we compute the rank of $H_{[1,x_2]} = \left(\begin{smallmatrix} 10 & 3 \\ 3 & 3 \end{smallmatrix}\right)$ which is equal to 2. We update $L = [x_2^2, x_1, x_1x_2]$ and $S = [1, x_2]$.*
Step 3: *$t = x_2^2$ but obviously the rank of $H_{[1,x_2,x_2^2]} = \left(\begin{smallmatrix} 10 & 3 & 3 \\ 3 & 3 & 3 \\ 3 & 3 & 3 \end{smallmatrix}\right)$ is always 2. We set $G' = [x_2^2]$.*
Step 4: *$t = x_1$; we compute the rank of $H_3 = H_{[1,x_2,x_1]} = \left(\begin{smallmatrix} 10 & 3 & 5 \\ 3 & 3 & 5 \\ 5 & 5 & 5 \end{smallmatrix}\right)$ which is equal to 3. Now $S = [1, x_2, x_1]$ and $L = [x_1x_2, x_1^2]$. Since $\#S \geq 3$ we can stop the algorithm so that $G' = \text{MinGBasis}([x_2^2, x_1x_2, x_1^2, x_2^3, \dots]) = [x_2^2, x_1x_2, x_1^2]$. Lastly, we solve the 3 linear systems $H_3\mathbf{x} = \mathbf{u}$ with*

$$\mathbf{u} = H_{[x_2^2],S} = \begin{pmatrix} 3 \\ 3 \\ 5 \end{pmatrix} \text{ or } \mathbf{u} = H_{[x_1x_2],S} = \begin{pmatrix} 5 \\ 5 \\ 5 \end{pmatrix} \text{ or } \mathbf{u} = H_{[x_1^2],S} = \begin{pmatrix} 5 \\ 5 \\ 5 \end{pmatrix}$$

and we obtain $G_{\text{LEX}} = [{x_2}^2 - x_2, x_1x_2 - x_1, {x_1}^2 - x_1]$. It is easy to compute the solutions $\{(0,0),(0,1),(1,1)\}$.

5.1 Relation between the number of table queries and the geometry of the final basis

To estimate the complexity of the algorithms we have to bound the number of table queries and the complexity of the linear algebra part (this issue is addressed in Sec. 6). Indeed, in some applications computing *one* element $u_{i_1,i_2,\dots}$ of the table is very costly (see for instance Sec. 5.2) and it is important to minimize the number of queries. Estimating this number is equivalent to counting the number of distinct elements in H_S where S can be any value of the variable in Alg. 4. We denote by S the value at the end of the algorithm. Similarly to the original FGLM algorithm we can bound the number of monomials t that we have to consider using $\#L \leq n\#S$. Hence it is crucial to bound the number of elements in H_S where S is the final staircase. Restating Th. 2, the necessary number of queries to $\mathbf{u}$ to build H_S is the cardinal of $2S = \{uv \mid (u,v) \in S^2\}$ the dilated set of S.

It is clear that $\#(2S) \leq \#S(\#S-1)/2 \leq (\#S)^2/2$ in the worst case; in many applications we have $\#(2S) \leq c\#S$ for some constant c. According to [16, Th. 1.1], sets S verifying this condition must be included in a bigger set whose elements are in arithmetical progression of dimension d and of size $C\#S$ for some constant C. In other words, S must be included in a d-dimensional parallelotope whose number of points is $C\#S$.

PROPOSITION 11. *Depending on the shape of the final Gröbner basis G, we estimate $\#(2S)$ when $d \to \infty$:*
a. *(1-dimensional case –* BM*) $n = 1$, $S_d = \{1, x, \dots, x^{d-1}\}$ then $\#(2S_d) = 2d-1$ and $\#(2S_d)/\#S_d \approx 2$;*
b. *(Worst 1-dimensional case) $n = 1$, $S_d = \{1, x^2, x^4, \dots, x^{2^{d-2}}\}$ then $\#(2S_d) = \binom{d-2}{2} + d + 1$ and $\#(2S_d)/S_d \approx \frac{d}{2}$;*
c. *(Alg. 3) $S_d = \{t \in T \mid \deg(t) \leq d\}$ then $\#(2S_d) = \binom{d+2}{2}$ and $\#(2S_d)/\#S_d \approx 2^n$;*
d. *(Shape position) When $G = [x_i - h_i(x_n) \mid i = 1,\dots,n]$ then $S_s = \{1, x_n, \dots, x_n^{d-1}\}$ where $d = \deg(h_n)$. Again $\#(2S_d)/\#S_d \approx 2$;*

e. *(Worst case in dimension n)* $S_d = \bigcup_{i=1}^{n} \{1, x_i, x_i^2, \ldots, x_i^{d/n}\}$ *then* $\frac{\#(2S_d)}{\#S_d} \approx \frac{1}{2}\frac{n-1}{n}\#S_d$.

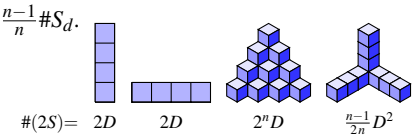

Figure 1: Behavior of $\#(2S)$ wrt. $D = \#S$ (the area in blue)

PROOF. *(a)* Clearly $2S_d = S_{2d-1}$.
(b) $S_{2d} = S_d \cup \{x^{2^i+2^j} \mid i \neq j\} \cup \{x^{2^{d-1}}\}$. Note that S_d is *not stable* under division in that case.
(c) Noticing $2S_d = S_{2d}$ and $\#S_d = \binom{d+n}{n}$, we have $\frac{\#S_{2d}}{\#S_d} = 2^n - 2^{n-1}\binom{n+1}{2}\frac{1}{d} + O(\frac{1}{d^2})$.
(d) Same as item a.
(e) We define $S'(n,d) = \bigcup_{i=1}^{n}\{x_i^j \mid j = 0, \ldots, d\}$ and it easy to show that $\#(2S'(n,d)) = n(n-1)d^2 + 2nd + 1$. Hence $S_d = S'(n, d/n)$ and $\#(2S_d)/\#S_d = (\frac{n-1}{2n}d^2 + 2d + 1)/(d+1) \approx \frac{1}{2}\frac{n-1}{n}d$. □

5.2 Application to the Sparse-FGLM algorithm

The Sparse-FGLM [5] is a natural application of the previous algorithm: for a 0-dimensional polynomial system we compute a first Gröbner basis (most of the time wrt. a total degree ordering). Then, we compute the $D \times D$ multiplication matrices T_i wrt. the variable x_i for all $i \in \{1, \ldots, n\}$. We consider the table $u_{\mathbf{i}} = \langle \mathbf{r}, T_1^{i_1} \cdots T_n^{i_n} \cdot \mathbf{1} \rangle$ where $\mathbf{r}$ is a random vector and $\mathbf{1} = [1, 0, \ldots]^{\mathrm{T}}$. The computation of one element of the table from the previous ones can be reduced to one matrix-vector multiplication.

REMARK 12. *Assuming that we store the vectors* $\mathbf{V_i} = T_1^{i_1} \cdots T_n^{i_n} \cdot \mathbf{1}$ *for the visited indices* $\mathbf{i}$*, any relation* $g = \sum_{s \in S} \alpha_s s \in G$ *computed by the algorithm can be easily checked:if* $\sum_{s \in S} \alpha_s \mathbf{V}_s = 0$ *then we have a proof that* $g \in I$*. Note, that in addition, we know precisely the bound d since it is the number of solutions (with multiplicities). Hence it is always possible to check the correctness of Alg. 4.*

Even if the sparsity of the multiplication matrices can be used to speed up the computation, it is important not to precompute *all* the elements of the table in advance. Hence a *black-box* representation is recommended. As shown in [5], when the lexicographical basis is in shape position, the Gröbner basis can be computed very efficiently; in particular, the number of table queries is $2D$, in this situation we can also use the change of variables designed in Sec. 3 to compute the Gröbner basis. This is why, in the experiments of the following paragraphs, we consider examples which are far from the shape position and we compute the LEX basis.

Cyclic-n problem. This is a well known benchmark; there are n equations in n variables, the ith equation is of degree i and is invariant by the action of the nth Cyclic group; since there is a linear equation, the actual number of variables is $n-1$. We report in Tab. 5.2, the number of rank computations and the normalized number of table queries (the number divided by the number of solutions). This number is always less than 2^{n-1}.

Example	n	D	Nb Ranks	#Queries/D
Cyclic 5	5	70	76	7.4
Cyclic 6	6	156	167	9.4
Cyclic 7	7	924	953	21.7

Ideal of points. Given a set $P \subset \mathbb{K}^n$ of t distinct points, we define the ideal $I_P = \{f \in \mathbb{K}[x_1, \ldots, x_n] \mid f(\mathbf{p}) = 0 \ \forall \mathbf{p} \in P\}$. We consider two such sets.

a. (Random) For any integer B, we generate exactly t points in $P_B \subset \mathbb{K}^n$ with coordinates randomly chosen in $\{0, \ldots, B-1\}$. Since B is a bound on the degree of the univariate polynomial in the LEX Gröbner basis, this basis is far from the shape position when $t \gg B$.

b. (Worst Case) $P_t = \{ie_j,\ 1 \leq i \leq n, 1 \leq j \leq t/n\}$. In both cases we report the ratio between the number of queries and the number of points. As expected in the first case, this ratio is a constant $c \in [2, 2^n]$ depending on the value of B. In the second case, we expect a linear behavior, from Prop. 11. In Fig. 5.2 the points below the thick dashed black line correspond to Gröbner bases in shape positions.

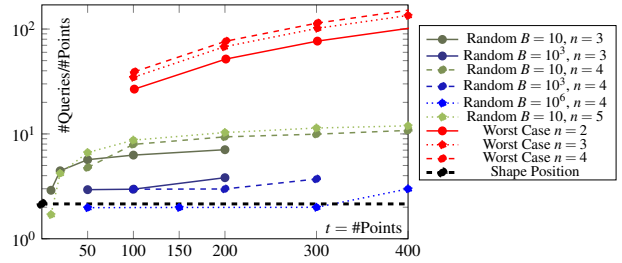

Figure 2: Number of table queries divided by number of points.

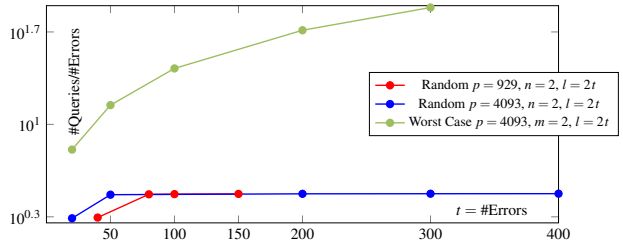

Figure 3: Number of table queries divided by number of points.

5.3 Application to error correcting codes

In Coding Theory, n-dimensional cyclic codes with $n > 1$ are generalizations of Reed Solomon codes. We give a simplified description of such codes. Let l be an integer and $a \in \mathbb{F}_p$ such that $a^j \neq 1$ for $0 < j < p-1$. We work with polynomials in $R = \mathbb{F}_p[\mathbf{x}]/\langle x_1^{p-1} - 1, \ldots, x_n^{p-1} - 1\rangle$. Then we define the generating polynomials $g_i(\mathbf{x}) = \prod_{j=0}^{l-1}(x_i - a^j)$. When we send a message $\mathscr{M}$ we split this message into n blocks $\mathscr{M}^{(k)} = (c_1^{(k)}, c_2^{(k)}, \ldots)$ where $c_i \in \mathbb{F}_p$ and we generate n multivariate polynomials $U_k(\mathbf{x}) = c_1^{(k)} + c_2^{(k)}x_1 + c_3^{(k)}x_2 + \cdots$. The transmitter sends the encoded message $M(\mathbf{x}) = \sum_{k=1}^{n} g_k(\mathbf{x})U_k(\mathbf{x})$. The receiver interprets the received word as a multivariate polynomial $N(\mathbf{x}) = M(\mathbf{x}) + e(\mathbf{x})$ where $e(\mathbf{x}) \in R$ is the error polynomial. If the length of $e(\mathbf{x})$ is less than $t = \frac{l}{2}$ the goal is to recover it. To this end, we build the table $u_{i_1,\ldots,i_n} := N(a^{i_1}, \ldots, a^{i_n}) \equiv e(a^{i_1}, \ldots, a^{i_n})$ in R for $0 \leq i_j < t$ and we apply Alg. 4 to obtain a LEX Gröbner basis G. It is easy to recover all the solutions in the finite field $\mathbb{F}_q$; next, by computing the discrete logarithm wrt. a of all the components we recover the position of the nonzero monomials in $e(\mathbf{x})$. Lastly, we solve a linear system to find the coefficients of $e(\mathbf{x})$.

In the experiments of Tab. 5.3, we consider two cases: (random case) we randomly generate the support and the coefficients of the error polynomial $e(\mathbf{x})$; (worst case) we take $e(\mathbf{x}) = \sum_{i=1}^{n}\sum_{j=0}^{t/n} c_{i,j}x_i^j$.

6. MULTIBLOCK HANKEL ARITHMETIC

In Alg. 3 and 4, linear systems must be solved. In this Section, we show that in fact, if $\prec$ is a LEX order, then the matrices are heavily structured as they are Hankel matrices.

Let's recall that a Gröbner basis of a 0-dimensional ideal for LEX order on $x_1, \ldots, x_n$ with $x_1 \prec \cdots \prec x_n$ is given in terms of non-constant polynomials $P_{1,1}(x_1)$ and $P_{i,j}(x_1, \ldots, x_i)$ for $i > 1$, with $\deg_{x_i} P_{i,j} \leq d_i$.

When computing $P_{1,1}$, one will only consider sets of monomials $S = \{1, x_1, \ldots, x_1^{d_1-1}\}$. Therefore, the matrix H_S is Hankel. When looking for $P_{2,1}, \ldots, P_{2,m_2}$, one needs to consider sets of monomials $S' = S \cup x_2 S_1 \cup \cdots \cup x_2^{d_2-1} S_{d_2-1}$ with $S_1, \ldots, S_{d_2-1} \subseteq S$. This yields

the following matrix

$$H_{S'} = \begin{pmatrix} H_S & H_{x_2 S, S_1} & \cdots & H_{x_2^{d_2-1} S, S_{d_2-1}} \\ H_{x_2 S_1, S} & H_{x_2^2 S_1, S_1} & \cdots & H_{x_2^{d_2} S_1, S_{d_2-1}} \\ \vdots & \vdots & \ddots & \vdots \\ H_{x_2^{d_2-1} S_{d_2-1}, S} & H_{x_2^{d_2} S_{d_2-1}, S_1} & \cdots & H_{x_2^{2d_2-2} S_{d_2-1}, S_{d_2-1}} \end{pmatrix}$$

where each $H_{x_2^k S_i, S_j}$ is Hankel rectangular. Completing $H_{S'}$ so that each block is square, i.e. replacing each $H_{x_2^k S_i, S_j}$ by $H_{x_2^k S, S}$ makes it block Hankel with Hankel blocks. We shall say that $H_{S'}$ is *multiblock Hankel of depth* 2. Then, for $P_{3,1}, \ldots, P_{3,m_3}$ of degree at most d_3 in x_3, one will consider the matrix $H_{S''}$ that is block Hankel with blocks $H_{x_3^k S'_i, S'_j}$, $0 \le i, j \le d_3 - 1$ such that $S'_1, \ldots, S'_{d_3-1} \subseteq S'_0 = S'$. That is, they will have the same shape as $H_{S'}$ and thus can be embedded in a multiblock Hankel matrix of depth 2. The matrix $H_{S''}$ shall be called *multiblock Hankel of depth* 3. This definition extends to all $n \in \mathbb{N}^*$, with depth 1 being classical Hankel matrices.

6.1 Displacement rank

We recall that for a matrix H, a *displacement operator* φ is an operator s.t. $\varphi(H)$ has small rank. One can classically take, for H Hankel, $\varphi(H) = H - ZHZ$ with $Z = (\delta_{i-1,j})_{1 \le i,j \le d}$, where $\delta_{i,j}$ denote Kronecker's delta function. Indeed, $\varphi(H)$ has rank at most 2. This nice structure allows us to solve a linear system with a Hankel-like matrix H, i.e. a small sum of Hankel matrices, in $O(\alpha^{\omega-1}\mathsf{M}(d)\log d)$ operations if $\alpha = \operatorname{rank}\varphi(H)$ and if d is the size of H, see [2].

On block Hankel matrices, one can take the deflated operator, in which each 1 of Z is replaced by an Identity matrix of the right size. However, the expected matrix should have rank twice as much as the size of the blocks. Because our blocks are themselves Hankel, we can once again apply the displacement operator of Hankel matrices on all remaining blocks. If H is Hankel block Hankel with d_2 blocks by row or column of size d_1, then the obtained matrix has rank at most $2\min(d_1, d_2)$.

Consequently, with multiblock Hankel matrices of depth n and embedding blocks of sizes $d_1, \ldots, d_n$, one can find a displacement operator s.t. the displacement rank is at most $2\prod_{i=1}^n d_i / \max_{i=1}^n d_i$. These displacement ranks are not too small, unless e.g. all the d_i's stay constant but one that grows to infinity.

6.2 Polynomial interpretation

Fast algorithms on solving Hankel linear systems are coming from the fact that multiplying a Hankel matrix of size d with a vector can be seen as computing the middle product of univariate polynomials of sizes $2d$ and d. Solving such a system comes down to dividing a polynomial of sizes $2d$ by a polynomial of size d, which can be done in $\tilde{O}(\mathsf{M}(d))$ operations in the base field. For Hankel block Hankel linear systems with d_2 blocks of size d_1, the matrix-vector product can be seen as a generalization of the middle product of two bivariate polynomials, both of degree $d_1 - 1$ in the first variable and one of degree $d_2 - 1$ and the other $2d_2 - 1$ in the second variable. By Kronecker's trick, the complexity of solving such a system is $\tilde{O}(\mathsf{M}(2d_1 d_2))$. For multiblock Hankel of depth n system, this strategy yields a complexity in $\tilde{O}(\mathsf{M}(2^{n-1} d_1 \cdots d_n))$.

6.3 Complexity comparisons

Let us recall that d is the order of the recursive sequence $\mathbf{u}$: it is the size of the staircase of any Gröbner basis of its ideal of relations. Let also μ denote the size of the computed Gröbner basis. In [21], the complexity of BMS is given as $O(\mu d^2)$ and estimated as $O(d^3)$ with the approximation $\mu \in O(d)$. Let us remark that the only proven bound is $\mu \le nd$ [4, Cor. 2.1] making the complexity of BMS in $O(nd^3)$ operations in $\mathbb{K}$.

In the shape position situation, $d_1 = d, d_2 = \cdots = d_n = 1$; in the worst-case scenario, the staircase is a simplex with $d_1 = \cdots = d_n$ and $d_1 \cdots d_n = n!\, d$. Our complexity estimate becomes resp. $\tilde{O}(\mathsf{M}(2^{n-1} d))$ or $\tilde{O}(\mathsf{M}(2^{n-1} n!\, d))$. Both are quasi-linear in d if n is fixed.

We cannot say if one of our two algorithms could be seen as a matrix version of BMS – in which case, we would improve the complexity estimate of BMS. Finding loop invariants in these algorithms could also help make their complexities sharper and find optimal termination criteria, hence reduce the number of table queries.

Acknowledgements

We thank the anonymous referees for their careful reading and their helpful comments and Erich L. Kaltofen for valuable discussions. This work has been partly supported by the French National Research Agency ANR-11-BS02-0013 HPAC project.

7. REFERENCES

[1] E. Berlekamp. Nonbinary BCH decoding. *IEEE Trans. Inform. Theory*, 14(2):242–242, March 1968.

[2] A. Bostan, C.-P. Jeannerod, and É. Schost. Solving Toeplitz- and Vandermonde-like Linear Systems with Large Displacement Rank. In C. W. Brown, editor, *ISSAC'07*, pages 33–40. ACM Press, 2007.

[3] H. Chabanne and G. H. Norton. On the key equation for n-dimensional cyclic codes: applications to decoding. Tech. report INRIA RR-1796, 1992.

[4] 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.

[5] J.-C. Faugère and C. Mou. Fast Algorithm for Change of Ordering of Zero-dimensional Gröbner Bases with Sparse Multiplication Matrices. In *Proc. of the 36th ISSAC*, pages 115–122. ACM, 2011.

[6] P. Gianni and T. Mora. Algebraic solution of systems of polynomial equations using Gröbner bases. In *Proc. of AAECC-5, vol. 356 of LNCS*, pages 247–257. Springer, 1989.

[7] E. Jonckheere and C. Ma. A simple Hankel interpretation of the Berlekamp-Massey algorithm. *Linear Algebra Appl.*, 125(0):65–76, 1989.

[8] E. Kaltofen and V. Pan. Processor efficient parallel solution of linear systems over an abstract field. In *SPAA '91*, pages 180–191, New York, N.Y., 1991. ACM Press.

[9] E. Kaltofen and G. Yuhasz. A fraction free Matrix Berlekamp/Massey algorithm. *Linear Algebra Appl.*, 439(9):2515–2526, 2013.

[10] E. Kaltofen and G. Yuhasz. On the Matrix Berlekamp-Massey Algorithm. *ACM Trans. Algorithms*, 9(4):33:1–33:24, Oct. 2013.

[11] C. Koutschan. Creative Telescoping for Holonomic Functions. In C. Schneider and J. Blümlein, editors, *Computer Algebra in Quantum Field Theory*, pages 171–194. Springer Vienna, 2013.

[12] Y. N. Lakshman. On the Complexity of Computing a Gröbner Basis for the Radical of a Zero Dimensional Ideal. In *Proc. of the 22nd Annual ACM STOC*, pages 555–563. ACM, 1990.

[13] N. Levinson. The Wiener RMS (Root-Mean-Square) error criterion in the filter design and prediction. *J. Math. Phys.*, 25:261–278, 1947.

[14] J. L. Massey. Shift-register synthesis and BCH decoding. *IEEE Trans. Inform. Theory*, IT-15:122–127, 1969.

[15] A. Poteaux and É. Schost. On the complexity of computing with 0-dimensional triangular sets. *J. Symbolic Comput.*, 50(0):110–138, 2013.

[16] I. Z. Ruzsa. Generalized arithmetical progressions and sumsets. *Acta Math. Hungar.*, 65(4):379–388, 1994.

[17] K. Saints and C. Heegard. Algebraic-geometric codes and multi-dimensional cyclic codes: Theory and algorithms for decoding using Gröbner bases. *IEEE Trans. Inform. Theory*, 41(6):1733–1751, 1995.

[18] S. Sakata. Finding a minimal set of linear recurring relations capable of generating a given finite two-dimensional array. *J. Symbolic Comput.*, 5(3):321–337, 1988.

[19] S. Sakata. Extension of the Berlekamp-Massey algorithm to N Dimensions. *Inform. and Comput.*, 84(2):207–239, Feb. 1990.

[20] S. Sakata. Decoding binary 2-D cyclic codes by the 2-D Berlekamp-Massey algorithm. *IEEE Trans. Inform. Theory*, 37(4):1200–1203, 1991.

[21] S. Sakata. The BMS algorithm. In M. Sala, T. Mora, L. Perret, S. Sakata, and C. Traverso, editors, *Gröbner Bases, Coding, and Cryptography*, pages 143–163. Springer Berlin Heidelberg, 2009.

[22] N. Wiener. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series.* New York Wiley, 1949. ISBN 0-262-73005-7.