

Guessing Linear Recurrence Relations of Sequence Tuples and P-recursive Sequences with Linear Algebra

Jérémy Berthomieu, Jean-Charles Faugère

► **To cite this version:**

Jérémy Berthomieu, Jean-Charles Faugère. Guessing Linear Recurrence Relations of Sequence Tuples and P-recursive Sequences with Linear Algebra. 41st International Symposium on Symbolic and Algebraic Computation, Jul 2016, Waterloo, ON, Canada. pp.95-102, 10.1145/2930889.2930926 . hal-01314266

HAL Id: hal-01314266

<https://hal.inria.fr/hal-01314266>

Submitted on 11 May 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Guessing Linear Recurrence Relations of Sequence Tuples and P-recursive Sequences with Linear Algebra

Jérémy Berthomieu
jeremy.berthomieu@lip6.fr

Jean-Charles Faugère
jean-charles.faugere@inria.fr

Sorbonne Universités, UPMC Univ Paris 06, CNRS, INRIA,
Laboratoire d'Informatique de Paris 6 (LIP6), Équipe POLSYS, 4 place Jussieu, 75252 Paris Cedex 05, France

ABSTRACT

Given several n -dimensional sequences, we first present an algorithm for computing the Gröbner basis of their module of linear recurrence relations.

A P-recursive sequence $(u_i)_{i \in \mathbb{N}^n}$ satisfies linear recurrence relations with polynomial coefficients in \mathbf{i} , as defined by Stanley in 1980. Calling directly the aforementioned algorithm on the tuple of sequences $((\mathbf{i}^j u_i)_{i \in \mathbb{N}^n})_j$ for retrieving the relations yields redundant relations. Since the module of relations of a P-recursive sequence also has an extra structure of a 0-dimensional right ideal of an Ore algebra, we design a more efficient algorithm that takes advantage of this extra structure for computing the relations.

Finally, we show how to incorporate Gröbner bases computations in an Ore algebra $\mathbb{K}\langle t_1, \dots, t_n, x_1, \dots, x_n \rangle$, with commutators $x_k x_\ell - x_\ell x_k = t_k t_\ell - t_\ell t_k = t_k x_\ell - x_\ell t_k = 0$ for $k \neq \ell$ and $t_k x_k - x_k t_k = x_k$, into the algorithm designed for P-recursive sequences. This allows us to compute faster the Gröbner basis of the ideal spanned by the first relations, such as in 2D/3D-space walks examples.

CCS Concepts

• **Comput. method.** → Symbolic calculus algorithms;

Keywords

Gröbner bases, P-recursive sequences, FGLM algorithm

1. INTRODUCTION

Computing linear recurrence relations of multi-dimensional sequences is a fundamental problem in Computer Science. As this is the core of the paper, we start with an example. The binomial sequence $\mathbf{b} = (i_1! / (i_2! (i_1 - i_2)!))_{(i_1, i_2) \in \mathbb{N}^2}$ satisfies Pascal's rule, a linear recurrence relation:

$$\forall (i_1, i_2) \in \mathbb{N}^2, \quad b_{i_1+1, i_2+1} = b_{i_1, i_2+1} + b_{i_1, i_2}. \quad (1)$$

It has constant coefficients. But this sequence also satisfies relations with polynomial coefficients in (i_1, i_2) :

$$\forall (i_1, i_2) \in \mathbb{N}^2, \quad (i_1 - i_2 + 1) b_{i_1+1, i_2} = (i_1 + 1) b_{i_1, i_2}, \quad (2a)$$

$$(i_2 + 1) b_{i_1, i_2+1} = (i_1 - i_2) b_{i_1, i_2}. \quad (2b)$$

ISSAC '16 July 19–22, 2016, Waterloo, ON, Canada

ACM ISBN .

DOI: <http://dx.doi.org/10.1145/2930889.2930926>

Thanks to equations (2a) and (2b), we can compute any term of \mathbf{b} starting with $b_{0,0} = 1$. This makes \mathbf{b} P-recursive.

Deciding if a sequence is P-recursive, as defined by Stanley [28] finds applications for instance in Combinatorics and in Computer algebra [27]. We refer the reader to the series of papers on planar [10, 11, 12] and 3D-space walks [7, 8].

The generating series $\sum_{i_1, \dots, i_n \in \mathbb{N}} u_{i_1, \dots, i_n} x_1^{i_1} \cdots x_n^{i_n}$ of a P-recursive sequence $(u_{i_1, \dots, i_n})_{(i_1, \dots, i_n) \in \mathbb{N}^n}$ is a D-finite function in x_1, \dots, x_n , see [19, Theorem 3.7]. A D-finite function is such that the dimension of the vector space spanned by all its derivatives is finite over $\mathbb{K}(x_1, \dots, x_n)$, this allows one to characterize such functions with initial conditions and differential equations. For instance, the DYNAMICAL DICTIONARY OF MATHEMATICAL FUNCTIONS [3] generates dynamically and automatically web pages on elementary and special functions thanks to the equations they satisfy. Likewise, a P-recursive sequence is uniquely defined by the linear recurrence relations with polynomial coefficients it satisfies and a finite number of initial terms.

1.1 Related work

For one-dimensional sequences, *C-relations* (linear recurrence relations with constant coefficients) can be guessed with Berlekamp – Massey algorithm [4, 20], while for multi-dimensional sequences, they can be guessed with Berlekamp – Massey – Sakata algorithm [24, 26] or SCALAR-FGLM [5, 6]. All these algorithms only *guess* and check that the relations are valid for a finite number of terms of the sequence. Then, usually, one need to *prove* that they are satisfied for all the terms of the sequence. In this paper, a *table* shall denote a finite subset of terms of a sequence: it is one of the input parameter of the algorithms since one cannot handle an infinite sequence.

Likewise, for one-dimensional sequences, *P-relations* (linear recurrence relations with polynomial coefficients) can be guessed with Beckermann – Labahn algorithm [2] for computing Hermite-Padé approximants.

When computing parameterized sums of multivariate sequences, or integrals of multivariate functions, in general, it is hopeless to obtain a closed form. Indeed, even for elementary univariate function we may not have closed forms when computing their integral on some domain. This motivated Zeilberger in 1990 [29] to introduce the so-called *creative telescoping* method. It is now one of the most efficient ways to do so and has been a tremendously active research topic. We refer the reader to the up-to-date nice survey on the state of the art in creative telescoping in [13] and the references therein.

Although sequences can satisfy some linear recurrence re-

lations with constant or polynomial coefficients, they still may be neither C-recursive nor P-recursive. Given a subset \mathcal{S} of steps in $\{0, \pm 1\}^2$, one can define the sequence $\mathbf{u} = (u_{n,i,j})_{(n,i,j) \in \mathbb{N}^3}$ of the number of planar walks of length n ending at (i, j) with steps in \mathcal{S} . Determining if \mathbf{u} is P-recursive has applications in lattice path enumeration [1, 11, 12]. Likewise, now 3D-space walks are investigated. In [7], the authors use several different methods to prove that some sequences of walks are P-recursive among which: the algebraic kernel method, the reduction to smaller dimensions and the guessing and proving method.

1.2 Contributions

We start by recalling the definitions of *C-recursive* (linear recursive with constant coefficients) sequences and define C-recursive tuples of sequences. This framework allows us to transform a P-recursive sequence into a tuple of sequences.

In Section 3, we extend SCALAR-FGLM [5, 6] for guessing the module of C-relations satisfied by a tuple of tables up to a degree bound d . The algorithm returns a $(d+1)$ -truncated Gröbner basis of their module of relations. The main idea of this algorithm is the same as SCALAR-FGLM's: extracting a full-rank submatrix of maximal rank.

P-relations with coefficients of degree at most δ satisfied by a sequence $\mathbf{u} = (u_{i_1, \dots, i_n})_{(i_1, \dots, i_n) \in \mathbb{N}^n}$ are C-relations satisfied by the tuple of sequences $(i_1^{j_1} \cdots i_n^{j_n} u_{i_1, \dots, i_n})_{(i_1, \dots, i_n) \in \mathbb{N}^n}$, with $j_1, \dots, j_n \geq 0$ and $j_1 + \cdots + j_n \leq \delta$. Hence, a first idea is to compute them with the extended SCALAR-FGLM for instance. However, this strategy is not optimal because the sequences are not independent: the module of relations has an additional structure, see also [21].

This is why, in Section 4, we design algorithm P-RECURSIVE-FGLM to take advantage of this particular structure. This algorithm input is a P-recursive table, a degree bound on the order of the relations and a degree bound on the polynomial coefficients. It returns a Gröbner basis of the right ideal of relations in a skew-polynomial ring $\mathbb{K}\langle t_1, \dots, t_n, x_1, \dots, x_n \rangle$ with commutation rules $x_k x_\ell - x_\ell x_k = t_k t_\ell - t_\ell t_k = t_k x_\ell - x_\ell t_k = 0$ for $k \neq \ell$ and $t_k x_k - x_k t_k = x_k$. P-RECURSIVE-FGLM uses an extra criterion *à la* FGLM: if relation (2b) is only known to be valid for a given pair (i_1, i_2) , then we still know that for any $j_1, j_2 \geq 0$, $i_1^{j_1} i_2^{j_2} (i_2 + 1) b_{i_1, i_2+1} = i_1^{j_1} i_2^{j_2} (i_1 - i_2) b_{i_1, i_2}$. This makes the extracted matrix smaller.

Sometimes an estimate of the Gröbner basis staircase size is known. To optimize the number of table queries, we present an output sensitive probabilistic algorithm in Section 4.2.

The relations computed by P-RECURSIVE-FGLM can be output one after the other. When several relations are found, it seems profitable not to look for any relation in the ideal they span. In Section 5, we use a mixed approach by computing truncated Gröbner bases in the aforementioned skew-polynomial ring to determine "needed redundant relations". On the one hand, they are redundant for they are deduced from previously found relations. On the other hand, they are computed faster since this approach does not require any extra queries to the table. Furthermore, these relations can help us to close the staircase in order to guess prematurely the P-recursive-ness of the table for a given degree. We illustrate this with Examples 6 and 7

In Section 6, we show that the matrices handled by MULTI-SCALAR-FGLM and P-RECURSIVE-FGLM are heavily structured. When the monomial ordering is LEX, we can make use of the quasi-Hankel structure to improve the complex-

ity of the linear system solving [9]. We also investigate the number of table queries performed by the algorithms.

2. DEFINITIONS AND NOTATION

We recall the classical definitions of a C-recursive sequence and of a P-recursive sequence. We shall also see how to deal with C-relations between terms of multiple sequences and see that from a P-recursive sequence, we can create a tuple of sequences satisfying C-relations.

In all the paper, we shall use standard notation with bolded letters corresponding to vectors and sequences: for $n \in \mathbb{N}^*$, $\mathbf{i} = (i_1, \dots, i_n)$ and $\mathbf{x} = (x_1, \dots, x_n)$. As usual, $\mathbf{x}^{\mathbf{i}}$ denotes $x_1^{i_1} \cdots x_n^{i_n}$ and $|\mathbf{i}| = i_1 + \cdots + i_n$. Finally, $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i} \in \mathbb{N}^n}$ is an n -dimensional sequence over a field \mathbb{K} .

For $\mathbf{i} \in \mathbb{N}^n$, we let $[\mathbf{x}^{\mathbf{i}}]_{\mathbf{u}} = u_{\mathbf{i}} = [\mathbf{x}^{\mathbf{i}}]$, if no ambiguity on \mathbf{u} shall arise. We extend linearly this $[\]$ operator to $\mathbb{K}[\mathbf{x}]$.

C-relations are straightforwardly extended to n -dimensional sequences [23, Definition 21]: let \mathcal{S} be a finite subset of \mathbb{N}^n , let $\alpha = (\alpha_{\mathbf{s}})_{\mathbf{s} \in \mathcal{S}}$ be nonzero, then $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i} \in \mathbb{N}^n}$ satisfies the linear relation defined by α if for all $\mathbf{i} \in \mathbb{N}^n$,

$$\left[\sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\mathbf{s}} \mathbf{x}^{\mathbf{i}+\mathbf{s}} \right] = \sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\mathbf{s}} u_{\mathbf{i}+\mathbf{s}} = 0. \quad (3)$$

Operator $[\]$ allows us to make a one-to-one correspondence between C-relations satisfied by \mathbf{u} and polynomials in $\mathbb{K}[\mathbf{x}]$: relation $\sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\mathbf{s}} u_{\mathbf{s}} = 0$ is *associated* to polynomial $\sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\mathbf{s}} \mathbf{x}^{\mathbf{s}}$.

According to Sakata [24] and to Fitzpatrick and Norton [17], a C-recursive sequence is defined as follows.

Definition 1. An n -dimensional sequence $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i} \in \mathbb{N}^n}$ is *C-recursive* if from a nonzero finite number of initial terms $u_{\mathbf{s}}, \mathbf{s} \in \mathcal{S}$, and a finite number of C-relations, without any contradiction, one can compute any term of the sequence.

From the polynomial viewpoint, Definition 1 can be restated as sequence \mathbf{u} is *C-recursive* if and only if set $I = \{P \in \mathbb{K}[\mathbf{x}], \forall \mathbf{i} \in \mathbb{N}^n, [P \mathbf{x}^{\mathbf{i}}] = 0\}$ is a zero-dimensional ideal.

A Gröbner basis of this ideal can be computed using the so-called BMS [25] and SCALAR-FGLM [5, Algorithm 3] algorithms, see also [6] for an extended version of [5].

C-recursive sequences are also exactly the only sequences such that their generating series $\sum_{\mathbf{i} \in \mathbb{N}^n} u_{\mathbf{i}} \mathbf{x}^{\mathbf{i}}$ is a rational fraction in \mathbf{x} whose denominator factors into univariate polynomials, see for instance [6, Proposition 19]

For a sequence \mathbf{u} not C-recursive, both algorithms, called on the table of terms $u_{\mathbf{i}}, |\mathbf{i}| \leq d$, return a $(d+1)$ -truncated Gröbner basis. Sequences such as the binomial one $\mathbf{b} = \left(\binom{i_1}{i_2} \right)_{\mathbf{i} \in \mathbb{N}^2}$ may satisfy a C-relation but not be C-recursive. For instance, \mathbf{b} satisfies equation (1), Pascal's rule, while its generating series is the rational fraction $1/(1-x_1-x_2)$ whose denominator is irreducible.

The greater class of P-recursive sequences satisfy *P-relations* (linear recurrence relations with polynomial coefficients). These relations are analogous to relations as equation (3) but with $\alpha_{\mathbf{s}}$ a polynomial in \mathbf{i} . For instance, both equations (2a) and (2b) satisfied by \mathbf{b} are P-relations.

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

2.1 Tuple of sequences

In this section, we aim to deal with multiple sequences $\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(m)}$ all n -dimensional over a common field \mathbb{K} . In particular, we shall ask them to satisfy C-relations, of type $[P_1]_{\mathbf{u}^{(1)}} + \dots + [P_m]_{\mathbf{u}^{(m)}} = 0$.

We extend the operator $[\]$ to vector of polynomials: for all $\mathbf{P} = (P_1, \dots, P_m) \in \mathbb{K}[\mathbf{x}]^m$, $[\mathbf{P}] = \sum_{\ell=1}^m [P_\ell]_{\mathbf{u}^{(\ell)}}$. This yields a one-to-one correspondence between vectors of polynomials in $\mathbb{K}[\mathbf{x}]^m$ and linear relations. We now aim to find the $\mathbb{K}[\mathbf{x}]$ -submodule $I = \{(P_1, \dots, P_m) \in \mathbb{K}[\mathbf{x}]^m, \forall \mathbf{i} \in \mathbb{N}^n, [P_1 \mathbf{x}^{\mathbf{i}}]_{\mathbf{u}^{(1)}} + \dots + [P_m \mathbf{x}^{\mathbf{i}}]_{\mathbf{u}^{(m)}} = 0\}$ of $\mathbb{K}[\mathbf{x}]^m$. We shall denote (e_1, \dots, e_m) the canonical basis of $\mathbb{K}[\mathbf{x}]^m$.

Let $\mathbf{u}^{(1)} = (3^{i_1})_{i_1 \in \mathbb{N}^2}$ and $\mathbf{u}^{(2)} = (3^{i_1} + 2^{i_2})$. The ideal of relations of $\mathbf{u}^{(1)}$ is $\langle x_2 - 1, x_1 - 3 \rangle$, while this of $\mathbf{u}^{(2)}$ is $\langle x_2^2 - 3x_2 + 2, x_1 + 2x_2 - 5 \rangle$. Therefore, we need to know $u_{0,0}^{(1)}, u_{0,0}^{(2)}, u_{0,1}^{(2)}$ to compute any term of both sequences.

Yet, the module of relations satisfied by $(\mathbf{u}^{(1)}, \mathbf{u}^{(2)})$ is spanned by $\{(x_2 - 1, 0), (x_1 - 3, 0), (1, x_2 - 2), (-2, x_1 - 1)\}$ and we only need to know $u_{0,0}^{(1)}, u_{0,0}^{(2)}$ to compute any term.

2.2 P-recursive sequences

For P-recursive sequences, we can naturally create several auxiliary sequences such that they satisfy C-relations. For instance, with sequence \mathbf{b} , we introduce the three auxiliary sequences: $\mathbf{b}^{(0)} = \mathbf{b}$, $\mathbf{b}^{(1)} = (i_1 \binom{i_1}{i_2})_{i_1 \in \mathbb{N}}$ and $\mathbf{b}^{(2)} = (i_2 \binom{i_1}{i_2})_{i_1 \in \mathbb{N}}$. Equation (1) is still satisfied by $\mathbf{b}^{(0)}$ and both equations (2a) and (2b) can be restated as C-relations satisfied by these 3 auxiliary sequences:

$$\forall \mathbf{i} \in \mathbb{N}^2, \quad b_{i_1+1, i_2}^{(1)} + b_{i_1+1, i_2}^{(2)} = b_{i_1, i_2}^{(1)} + b_{i_1, i_2}^{(0)}, \quad (4a)$$

$$b_{i_1, i_2+1}^{(2)} = b_{i_1, i_2}^{(1)} - b_{i_1, i_2}^{(2)}. \quad (4b)$$

Hence, their module of relations contains $\text{Span}\{(x_1 x_2 - x_2 - 1, 0, 0), (1, 1 - x_1, x_1), (0, -1, x_2 + 1)\} \subseteq \mathbb{K}[\mathbf{x}]^3$. More generally, for any P-recursive sequence, one would have to introduce as many auxiliary sequences as needed depending on the number of indices and the allowed degree for the polynomial coefficients.

Usually a new set of variables $\mathbf{t} = (t_1, \dots, t_n)$ is introduced, such that $[t_k \mathbf{x}^{\mathbf{i}}] = i_k u_i$, and thus $[\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}] = \mathbf{j}^{\mathbf{i}} u_i$. This yields commutation rules $t_k x_k = x_k (t_k + 1)$ and $t_k x_\ell = x_\ell t_k$ for $k \neq \ell$. We denote by $\mathbb{K}(\mathbf{t}, \mathbf{x})$ the skew-polynomial ring with these commutation rules. We also let $\mathbb{K}(\mathbf{t})(\mathbf{x})$ be the ring of skew-polynomials in \mathbf{x} whose coefficients are rational fractions in \mathbf{t} .

In this setting, P-relations are in a one-to-one correspondence with polynomials in $\mathbb{K}(\mathbf{t}, \mathbf{x})$ and a P-recursive sequence is a sequence such that the set of relations $I = \{P \in \mathbb{K}(\mathbf{t})(\mathbf{x}), \forall \mathbf{i}, \mathbf{j} \in \mathbb{N}^n, [P \mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}] = 0\}$ is a 0-dimensional right ideal, i.e. for all $k, 1 \leq k \leq n, I \cap \mathbb{K}(\mathbf{t})(x_k) \neq \{0\}$. For \mathbf{b} , this yields the right ideal spanned by $\langle (t_1 - t_2)x_1 - (t_1 + 1), t_2 x_2 + t_2 - t_1 \rangle \subseteq \mathbb{K}(\mathbf{t})(\mathbf{x})$.

In general, if one considers the right ideal $I' = \{P \in \mathbb{K}(\mathbf{t}, \mathbf{x}), \forall \mathbf{i}, \mathbf{j} \in \mathbb{N}^n, [P \mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}] = 0\}$, then for all k , there exists $d > 0$ and \mathbf{j} such that $\mathbf{t}^{\mathbf{j}} x_k^d$ is the leading monomial of a polynomial in the ideal.

2.3 Lattice-based relations

We now deal with sequences satisfying C-relations when evaluated in points $\mathbf{i} \in \Lambda$, a sublattice of \mathbb{N}^n .

The one-dimensional sequence $\mathbf{u} = (u_i)_{i \in \mathbb{N}}$ defined by for all $i \in \mathbb{N}$, $u_{2i} = u_i^{(0)} = i!$ and $u_{2i+1} = u_i^{(1)} = i! + 1$ satisfies for all $i \in \mathbb{N}$

$$u_{2i+3} - u_{2i+2} - u_{2i+1} + u_{2i} = u_{i+1}^{(1)} - u_{i+1}^{(0)} - u_i^{(1)} + u_i^{(0)} = 0. \quad (5)$$

but not $u_{i+3} - u_{i+2} - u_{i+1} + u_i = 0$ for odd i . Nonetheless, computing equation 5 is of interest since with $(u_{2i})_{i \in \mathbb{N}}$ and u_1 , it allows us to compute all the odd-indexed terms of \mathbf{u} .

As in Section 2.1, we can write equation 5:

$$u_{i+1}^{(1)} - u_{i+1}^{(0)} - u_i^{(1)} + u_i^{(0)} = [(x+1)(e_2 - e_1)x^i] = 0.$$

More generally, let $(\lambda_1, \dots, \lambda_n)$ be a basis of Λ , \mathcal{S} be a finite subset of \mathbb{N}^n such that sequence $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i} \in \mathbb{N}^n}$ satisfies

$$\sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\mathbf{s}} u_{\mathbf{i}+\mathbf{s}} = 0, \quad \forall \mathbf{i} \in \Lambda. \quad (6)$$

Then, one can create auxiliary sequences $\mathbf{u}^{(\mathbf{f})} = (u_{\mathbf{i}+\mathbf{f}})_{\mathbf{i} \in \Lambda}$ for all $\mathbf{f} \in \mathcal{F}$, the fundamental domain of Λ to find the relation of equation 6.

Mimicking the P-recursive case, we introduce a new variable t such that for all $\mathbf{i} \in \mathbb{N}^n$, $t \mathbf{x}^{\mathbf{i}} = \mathbf{x}^{i_1 \lambda_1} \dots \mathbf{x}^{i_n \lambda_n} t$ and $[t \mathbf{x}^{\mathbf{i}}] = u_{i_1 \lambda_1 + \dots + i_n \lambda_n}$, we can extend these relations to $\mathbb{K}(t, \mathbf{x})$ by linearity. For $\mathbf{i} \in \Lambda$, we let $\mathbf{i} = c_1 \lambda_1 + \dots + c_n \lambda_n$, therefore $\mathbf{x}^{\mathbf{i}} = t x_1^{c_1} \dots x_n^{c_n}$. Then, one obtains relation

$$\left[\sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\mathbf{s}} \mathbf{x}^{\mathbf{i}+\mathbf{s}} \right] = \left[\sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\mathbf{s}} \mathbf{x}^{\mathbf{s}} t x_1^{c_1} \dots x_n^{c_n} \right] = 0.$$

3. GUESSING RELATIONS FOR TUPLES OF SEQUENCES

This section is devoted to the design of an FGLM-like algorithm in the module $\mathbb{K}[\mathbf{x}]^m$ to compute the Gröbner basis of the submodule of relations of n -dimensional tables $\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(m)}$ defined over a common field \mathbb{K} .

For the paper to be self-contained, we recall the notation and definitions of [5]. We fix an admissible order \prec on \mathbf{x} , \mathcal{T} is the ordered set of terms made from \mathbf{x} . We extend \prec to monomial vectors of type $\mathbf{x}^{\mathbf{i}} e_\ell, \mathbf{x}^{\mathbf{i}'} e_{\ell'}$ with $\mathbf{x}^{\mathbf{i}} e_\ell \preceq \mathbf{x}^{\mathbf{i}'} e_{\ell'}$ if $\mathbf{x}^{\mathbf{i}} \prec \mathbf{x}^{\mathbf{i}'}$ or if $\mathbf{x}^{\mathbf{i}} = \mathbf{x}^{\mathbf{i}'}$ and $\ell \leq \ell'$.

For $\mathbf{P} \in \mathbb{K}[\mathbf{x}]^m$, we denote by $\mathcal{T}(\mathbf{P})$ the set of terms appearing in \mathbf{P} and by $\text{LT}(\mathbf{P})$ the maximum of $\mathcal{T}(\mathbf{P})$ for \prec .

As we try to design an algorithm, we cannot expect to check that a relation is satisfied for all indices $\mathbf{i} \in \mathbb{N}^n$. During the execution of the algorithm, we shall deal with a finite subset of indices $\mathcal{U} \subset \mathbb{N}^n$, and we aim to find relations valid for those indices: $\forall \mathbf{i} \in \mathcal{U}, \sum_{\ell=1}^m \sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\ell, \mathbf{s}} u_{\mathbf{i}+\mathbf{s}}^{(\ell)} = 0$.

From the polynomial viewpoint, this translates to $\forall \mathbf{i} \in \mathcal{U}$,

$$\left[\sum_{\ell=1}^m \sum_{\mathbf{s} \in \mathcal{S}} \alpha_{\ell, \mathbf{s}} \mathbf{x}^{\mathbf{i}+\mathbf{s}} e_\ell \right] = 0.$$

Definition 3. Let \mathcal{U} be a finite subset of \mathbb{N}^n . We say that a vector of polynomials $\mathbf{P} = (P_1, \dots, P_m) \in \mathbb{K}[\mathbf{x}]^m$ is valid up to \mathcal{U} if $[\mathbf{P} \mathbf{x}^{\mathbf{i}}] = \sum_{\ell=1}^m [P_\ell \mathbf{x}^{\mathbf{i}}]_{\mathbf{u}^{(\ell)}} = 0$ for all $\mathbf{i} \in \mathcal{U}$.

3.1 Staircases

Since we want to generate minimal relations, it is mandatory to determine a set of terms in \mathcal{T} for which there is no C-relations. *Useful staircases* were defined for C-recursive tables in [5, Definition 6]. The following definitions extends it for multiple tables.

Definition 4. Let T_1, \dots, T_m, U be finite subsets of \mathcal{T} such that $T_1, \dots, T_m \subseteq U$ and $\#U = \sum_{\ell=1}^m \#T_\ell$. We say that a set $S = S_1 e_1 \cup \dots \cup S_m e_m$, with $S_\ell \subseteq T_1, \dots, S_m \subseteq T_m$, is a *useful staircase* with $V \subseteq U$ wrt. $\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(m)}$, $T = T_1 e_1 \cup \dots \cup T_m e_m, U$, and \prec if $\#V = \#S$ and

$$\sum_{1 \leq \ell \leq m} \sum_{s \in S_\ell} \beta_{\ell,s} [s v e_\ell] = 0, \forall v \in V$$

implies that $\beta_{\ell,s} = 0$ for all ℓ, s , sets S_1, \dots, S_m, V are 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.

As we are in a more general case, useful staircases still fail in general to be true staircases in $\mathbb{K}[\mathbf{x}]^m$, see [5, Example 3]. However, this problem is easily solved by stabilizing the staircase, i.e. adding to each set of the tuple all the divisors of its elements.

3.2 Algorithm for finding the relations

We design an algorithm for checking that a finite subset $S \subseteq T$ is a useful staircase with $V \subseteq U$ wrt. $\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(m)}$, T, U and \prec . In the following example, we compute the useful staircase for two univariate tables.

Example 1. Let us consider $\mathbf{u}^{(1)} = (3^{i_1})_{i \in \mathbb{N}^2}$ and $\mathbf{u}^{(2)} = (3^{i_1+2^{i_2}})_{i \in \mathbb{N}^2}$. Let $T_1 = T_2 = \{1, x_2\}$ and $U = \{1, x_2, x_1, x_2^2\}$, then we have the following potential relations

$$\begin{matrix} & e_1 & e_2 & x_2 e_1 & x_2 e_2 \\ \begin{matrix} 1 \\ x_2 \\ x_1 \\ x_2^2 \end{matrix} & \begin{pmatrix} 1 & 2 & 1 & 3 \\ 1 & 3 & 1 & 5 \\ 3 & 4 & 3 & 5 \\ 1 & 5 & 1 & 9 \end{pmatrix} & \begin{pmatrix} \alpha_1 \\ \beta_1 \\ \alpha_{x_2} \\ \beta_{x_2} \end{pmatrix} & = 0. \end{matrix}$$

From the matrix viewpoint, a useful staircase is a set of independent columns with as many low-labeled columns as possible wrt. independent rows with as many low-labeled rows as possible. This matrix has rank 2: the first two columns (resp. the first two rows) are independent. Therefore, $S = \{e_1, e_2\}$ is the useful staircase with $V = \{1, x_2\}$.

Thus, we can find relations $(-\alpha_1 + x_2)e_1 - \beta_1 e_2$ and $-\alpha'_1 e_1 + (-\beta'_1 + x_2)e_2$ with a linear system:

$$\begin{pmatrix} 1 & 2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} \alpha_1 & \alpha'_1 \\ \beta_1 & \beta'_1 \end{pmatrix} = \begin{pmatrix} 1 & 3 \\ 1 & 5 \end{pmatrix}.$$

This yields $(x_2 - 1)e_1 = e_1 + (x_2 - 2)e_2 = 0$.

We are now in a position to define the structured matrix associated to lists of terms $T_1, \dots, T_m \subseteq U \subseteq \mathcal{T}$.

Definition 5. Let T_1, \dots, T_m, U be finite lists of terms such that $T_1, \dots, T_m \subseteq U \subseteq \mathcal{T}$ and $\#U = \sum_{\ell=1}^m \#T_\ell$. The *multi-Hankel* matrix $H_{U,T}$ has the same columns and rows as the concatenation of matrices $H_{U,T_1}, \dots, H_{U,T_m}$, where each H_{U,T_ℓ} is a multi-Hankel matrix for table $\mathbf{u}^{(\ell)}$ as defined in [5, Definition 7]:

$$v \in U \begin{pmatrix} \dots & s \in T_1 & \dots & \dots & \dots & s \in T_m & \dots \\ \vdots & \vdots & \ddots & \dots & \vdots & \vdots & \ddots \\ \dots & [s v]_{\mathbf{u}^{(1)}} & \dots & \dots & [s v]_{\mathbf{u}^{(m)}} & \dots & \dots \\ \vdots & \vdots & \ddots & \dots & \vdots & \vdots & \ddots \end{pmatrix}.$$

If each $S_\ell \subseteq T_\ell, V \subseteq U$, $\#V = \sum_{\ell=1}^m \#S_\ell$ and $H_{V,S}$ has full rank, then $\bigcup_{\ell=1}^m S_\ell e_\ell$ is a useful staircase with V .

Furthermore, finding a *monic* polynomial vector \mathbf{P} of a given support such that $[\mathbf{P} \mathbf{x}^i] = 0$, for all $\mathbf{x}^i \in U$ is equivalent to solving a linear system $H_{U,T} \boldsymbol{\alpha} + H_{U,LT(\mathbf{P})} = 0$, where T is the support of \mathbf{P} without its leading term.

PROPOSITION 1. Let T_1, \dots, T_m, U be finite subsets of \mathcal{T} . If the finite sets S_1, \dots, S_m, V are such that $S = \bigcup_{\ell=1}^m S_\ell e_\ell$ is a useful staircase with V wrt. $\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(m)}, T, U$ and \prec , then $\det H_{V,S} \neq 0$, $\text{rank } H_{V,S} = \text{rank } H_{U,T}$.

If $S'_1 \subseteq S_1, \dots, S'_m \subseteq S_m, V' \subseteq V$ are finite sets of terms, then $\text{rank } H_{V',S'} \leq \text{rank } H_{V,S}$.

3.3 Extension of SCALAR-FGLM

As presented in [5, 6], SCALAR-FGLM is an algorithm whose input is a table \mathbf{u} , an integer d and a monomial order \prec refined by the total degree. If \mathbf{u} is C-recursive of order less than d , then the algorithm returns a Gröbner basis of the ideal of relations of \mathbf{u} . Otherwise it returns a $(d+1)$ -truncated Gröbner basis of an ideal such that the relations induced by the Gröbner basis are only valid up to degree d . This extension is natural: the input shall be a vector of n -dimensional tables $(\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(m)})$, a vector of integers (d_1, \dots, d_m) and a monomial order \prec on $\mathbb{K}[\mathbf{x}]$ which we extend to $\mathbb{K}[\mathbf{x}]^m$ as specified earlier.

Algorithm 1, MULTI-SCALAR-FGLM, computes the module of relations of a list of tables $\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(m)}$ up to degrees d_1, \dots, d_m .

Algorithm 1: MULTI-SCALAR-FGLM.

Input: Tables $\mathbf{u}^{(1)}, \dots, \mathbf{u}^{(m)}$ over \mathbb{K} of dimension n , (d_1, \dots, d_m) given bounds and \prec a monomial order.
Output: A reduced $(d_1 + 1, \dots, d_m + 1)$ -truncated Gröbner basis wrt. \prec of their module of relations.

Let $T = \mathcal{T}_{d_1} e_1 \cup \dots \cup \mathcal{T}_{d_m} e_m$ where \mathcal{T}_d is the ordered list of monomials of degree at most d .

Let U be the ordered list of the first $\#T$ monomials in \mathbf{x} .

Build the matrix $H_{U,T}$.

Extract a submatrix of maximal rank by finding

$S = S_1 e_1 \cup \dots \cup S_m e_m$ and V such that S is the useful staircase for V , i.e. s.t. $\text{rank } H_{U,T} = \text{rank } H_{V,S}$.

$S' := \text{Stabilize}(S)$.

$L := \{\mathcal{T}_{d_1+1} \setminus S'_1, \dots, \mathcal{T}_{d_m+1} \setminus S'_m\}$. // list of next terms
 $G := \{\}$. // the future Gröbner basis

For ℓ from 1 to m **do**

While $L[\ell] \neq \emptyset$ **do**

$\tau := \min_{\prec}(L[\ell])$ and remove τ from $L[\ell]$.

 Find $\boldsymbol{\alpha}$ s.t. $H_{V,S} \boldsymbol{\alpha} + H_{V,\{\tau e_\ell\}} = 0$.

$G := G \cup \{\tau e_\ell + \sum_{s \in S} \alpha_s s\}$.

 Sort $L[\ell]$ by increasing order (wrt. \prec) and remove multiples of $\text{LT}(G)$.

Return G .

Example 2. Let us unroll the algorithm on a modified version of Example 1, with tables $\mathbf{u}^{(1)} = (3^{i_1})_{i \in \mathbb{N}^2}$, $\mathbf{u}^{(2)} = (3^{i_1+2^{i_2}})_{i \in \mathbb{N}^2}$ and $(d_1, d_2) = (1, 1)$.

First, $T = \{1, x_2, x_1\} e_1 \cup \{1, x_2, x_1\} e_2$, so that $U = \{1, x_2, x_1, x_2^2, x_1 x_2, x_1^2\}$ and matrix $H_{U,T}$ is

$$\begin{matrix} & e_1 & e_2 & x_2 e_1 & x_2 e_2 & x_1 e_1 & x_1 e_2 \\ \begin{matrix} 1 \\ x_2 \\ x_1 \\ x_2^2 \\ x_1 x_2 \\ x_1^2 \end{matrix} & \begin{pmatrix} 1 & 2 & 1 & 3 & 3 & 4 \\ 1 & 3 & 1 & 5 & 3 & 5 \\ 3 & 4 & 3 & 5 & 9 & 10 \\ 1 & 5 & 1 & 9 & 3 & 7 \\ 3 & 5 & 3 & 7 & 9 & 11 \\ 9 & 10 & 9 & 11 & 27 & 20 \end{pmatrix} & & & & & \end{matrix}$$

The first two independent columns are e_1, e_2 while the first two independent rows are $1, x_2$. This means that the useful

staircase is $S = \{e_1, e_2\}$ with $V = \{1, x_2\}$. The true staircase is then $S' = S$ and we initialize L with $\{x_2, x_1, x_2^2, x_1 x_2, x_1^2\} e_1 \cup \{x_2, x_1, x_2^2, x_1 x_2, x_1^2\} e_2$.

For $\ell = 1$, we set $\tau = x_2$, solve the system $H_{V,S} \alpha + \binom{1}{1}$ yielding $\mathbf{P} = (x_2 - 1) e_1$ and update L to $\{x_1, x_1^2\} e_1 \cup \{x_2, x_1, x_2^2, x_1 x_2, x_1^2\} e_2$.

Then, we set $\tau = x_1$, solve the system $H_{V,S} \alpha + \binom{3}{3}$ yielding $\mathbf{P}' = (x_1 - 3) e_1$ and update L to $\{x_2, x_1, x_2^2, x_1 x_2, x_1^2\} e_2$.

For $\ell = 2$, we set $\tau = x_2$, solve the system $H_{V,S} \alpha + \binom{3}{5}$ yielding $\mathbf{P}'' = e_1 + (x_2 - 2) e_2$ and update L to $\{x_1, x_1^2\} e_2$.

Then, we set $\tau = x_1$, solve the system $H_{V,S} \alpha + \binom{4}{5}$ yielding $\mathbf{P}''' = -2e_1 + (x_1 - 1) e_2$ and update L to \emptyset .

The algorithm returns $\{\mathbf{P}, \mathbf{P}', \mathbf{P}'', \mathbf{P}'''\}$.

4. GUESSING RELATIONS FOR P-RECURSIVE SEQUENCES

As stated in Section 2.2, one could use Algorithm 1 to deal with P-recursive tables. For instance, if one suspects a sequence \mathbf{u} to satisfy relations of type $\sum_{s \in S} \alpha_s (\mathbf{i} + \mathbf{s}) u_{\mathbf{i} + \mathbf{s}} = 0$ with $\alpha_s \in \mathbb{K}[\mathbf{i}]$ of degree at most δ , then one would just need to call the algorithm with all the $\mathbf{u}^{(j)} = (\mathbf{i}^j u_i)_{i \in \mathbb{N}}$, s.t. $|\mathbf{j}| \leq \delta$, as input tables.

4.1 A criterion for P-recursive sequences

The main drawback of this strategy is the output of the algorithm: there are polynomials representing redundant relations if the sequence is P-recursive but not C-recursive. For instance, Algorithm 1 called on $\mathbf{b}^{(0)}, \mathbf{b}^{(1)}, \mathbf{b}^{(2)}$, of Section 2.2, with degree bounds $(1, 1, 1)$ in \mathbf{x} returns both relations (4a) and (4b) together with six extra relations.

Yet, the first two are enough to compute every term of the sequence starting from $\binom{0}{0} = 1$.

These superfluous relations come from the fact that we forget the extra structure of the module of relations, i.e. it is also a right ideal of $\mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$ for $\mathbf{t} = (t_1, \dots, t_n)$, $t_k x_\ell = x_\ell t_k$ if $k \neq \ell$ and $t_k x_k = x_k (t_k + 1)$.

Therefore, it is possible to extend MULTI-SCALAR-FGLM to take into account this structure. Whenever a relation

$$\left[\mathbf{t}^{\mathbf{j} + \delta} \mathbf{x}^{\mathbf{i} + \mathbf{d}} \right] = \left[\sum_{r \in \mathcal{R}, s \in S} \alpha_{r,s} \mathbf{t}^{\mathbf{j} + \mathbf{r}} \mathbf{x}^{\mathbf{i} + \mathbf{s}} \right]$$

$$(\mathbf{i} + \mathbf{d})^{\mathbf{j} + \delta} u_{\mathbf{i} + \mathbf{d}} = \sum_{r \in \mathcal{R}, s \in S} \alpha_{r,s} (\mathbf{i} + \mathbf{s})^{\mathbf{j} + \mathbf{r}} u_{\mathbf{i} + \mathbf{s}}$$

is true up to a certain threshold for \mathbf{i} , then its multiples by $i_k + d_k$ are also true up to the same threshold.

PROPOSITION 2. *Let \mathbf{u} be a P-recursive table over \mathbb{K} . Let T (resp. U) be a finite subset of monomials in $\mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$ (resp. in $\mathbb{K}[\mathbf{x}]$). If $\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}$ is not in the useful staircase of matrix $\left(\left[\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i} + \mathbf{i}'} \right] \right)_{\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}} \in T, \mathbf{x}^{\mathbf{i}'} \in U}$ then neither is any multiple $\mathbf{t}^{\mathbf{j} + \mathbf{j}'} \mathbf{x}^{\mathbf{i}}$.*

PROOF. If $\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}$ is not in the useful staircase, a relation starting with $u_{\mathbf{i} + \mathbf{k}}$ exists. Multiplying it by $i_k + d_k$ yields a new relation whose associated polynomial has a leading term $t_k \mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}$, hence column $t_k \mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}$ can also be removed. \square

Let us notice that this is analogous to the behavior of FGLM for polynomial ideals in $\mathbb{K}[\mathbf{x}]$ [16]: whenever a relation $\mathbf{x}^{\mathbf{d}} = \sum_{s \prec \mathbf{d}} \alpha_s \mathbf{x}^s$ is found, then automatically for all $\mathbf{i} \in \mathbb{N}^n$, $\mathbf{x}^{\mathbf{i} + \mathbf{d}} = \sum_{s \prec \mathbf{d}} \alpha_s \mathbf{x}^{\mathbf{i} + \mathbf{s}}$. This is also analogous to Buchberger – Möller algorithms wherein the finitely many points

can be interpreted as the finitely many table terms we have. We illustrate this criterion.

Example 3. We consider $\mathbf{u} = (i!)_{i \in \mathbb{N}}$ and skew-polynomial ring $\mathbb{K}\langle t, x \rangle$. Columns will be labeled by $T = \{1, t, x, tx\}$ and rows by $U = \{1, x, x^2, x^3\}$. This yield the rank 3 matrix

$$H_{U,T} = \begin{matrix} & 1 & t & x & tx \\ \begin{matrix} 1 \\ x \\ x^2 \\ x^3 \end{matrix} & \begin{pmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 2 & 4 \\ 2 & 4 & 6 & 18 \\ 6 & 18 & 24 & 96 \end{pmatrix} \end{matrix}$$

Column labeled x is the first dependent one meaning we find relations $[x^{i+1} - (t+1)x^i] = u_{i+1} - (i+1)u_i = 0$ for all $i \in \{0, \dots, 3\}$. This leads us to determine that $(i+1)u_{i+1} + (i^2 + 2i + 1)u_i = 0 = [tx^{i+1} + (t^2 + 2t + 1)x^i]$ for all $i \in \{0, \dots, 3\}$ without computing $[t^2], [t^2x], [t^2x^2], [t^2x^3]$, i.e. column t^2 . Therefore, column tx is independent from the previous ones but is not in the useful staircase.

In this context, a *useful staircase* is the maximal subset $S \subseteq T$ such that columns of $H_{U,S}$ are all linearly independent and for all $\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}} \in T \setminus S$ and $\mathbf{j}' \in \mathbb{N}^n$, $\mathbf{t}^{\mathbf{j} + \mathbf{j}'} \mathbf{x}^{\mathbf{i}} \notin S$.

To compare monomials in $\mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$ in order to build our matrix, we proceed as follows. We let \prec be a monomial ordering refined by the total degree on \mathbf{t} and \mathbf{x} with the convention that for any k, ℓ , $t_k \prec x_\ell$.

The following algorithm, P-RECURSIVE-FGLM, computes the linear recurrence relations with polynomial coefficients of an input P-recursive table. Thanks to the criterion given in Proposition 2, the linear systems that we solve at the end are smaller than in Algorithm 1.

Algorithm 2: P-RECURSIVE-FGLM.

Input: A table $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i} \in \mathbb{N}^n}$ with coefficients in \mathbb{K} , (δ, d) given bounds and \prec a monomial ordering.

Output: A reduced (δ, d) -truncated Gröbner basis wrt. \prec of the ideal of relations of \mathbf{u} in $\mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$.

Let T be the ordered list of monomials in \mathbf{t}, \mathbf{x} of bidegree at most (δ, d) .

Let U be the ordered list of the first $\#T$ monomials in \mathbf{x} .

Build the matrix $H_{U,T}$.

Find S the useful staircase.

Find V the set of first linearly independent rows of $H_{U,S}$.

$S' := \text{Stabilize}(S)$.

$L := T \setminus S'$.

// list of next terms to study

$G := \{\}$.

// the future Gröbner basis

While $L \neq \emptyset$ **do**

$\tau := \min_{\prec}(L)$.

 Find α s.t. $H_{V,S} \alpha + H_{V,\{\tau\}} = 0$.

$G := G \cup \{\tau + \sum_{s \in S} \alpha_s s\}$.

 Sort L by increasing order (wrt. \prec) and remove multiples of $\text{LT}(G)$.

Return G .

Example 4. Let us unroll Algorithm 2 on table $(i_1! / i_2!)_{i \in \mathbb{N}^2}$ with $\delta = d = 1$ and ordering $\text{DRL}(t_2 \prec t_1 \prec x_2 \prec x_1)$.

We have $T = \{1, t_2, t_1, x_2, x_1, t_2 x_2, t_2 x_1, t_1 x_2, t_1 x_1\}$ and we let $U = \{1, x_2, x_1, x_2^2, x_1 x_2, x_1^2, x_2^3, x_1 x_2^2, x_1^2 x_2\}$. This yields matrix $H_{U,T}$ described in equation 7.

The first linearly independent columns are $1, t_2, t_1, x_2, t_2 x_1, t_1 x_2, t_1 x_1$. As x_1 is not in this set, we need to remove both $t_2 x_1$ and $t_1 x_1$ by Proposition 2. Hence the useful staircase is $S = \{1, t_2, t_1, x_2, t_1 x_2\}$. The set of the first 5 independent rows is $V = \{1, x_2, x_1, x_2^2, x_1 x_2\}$.

$$\begin{array}{c}
1 \\
x_2 \\
x_1 \\
x_2^2 \\
x_1 x_2 \\
x_1^2 \\
x_2^3 \\
x_1 x_2^2 \\
x_1^2 x_2
\end{array}
\begin{pmatrix}
1 & t_2 & t_1 & x_2 & x_1 & t_2 x_2 & t_2 x_1 & t_1 x_2 & t_1 x_1 \\
1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\
1 & 1 & 0 & 1/2 & 1 & 1 & 1 & 0 & 1 \\
1 & 0 & 1 & 1 & 2 & 1 & 0 & 1 & 4 \\
1/2 & 1 & 0 & 1/6 & 1/2 & 1/2 & 1 & 0 & 1/2 \\
1 & 1 & 1 & 1/2 & 2 & 1 & 2 & 1/2 & 4 \\
2 & 0 & 4 & 2 & 6 & 2 & 0 & 4 & 18 \\
1/6 & 1/2 & 0 & 1/24 & 1/6 & 1/6 & 1/2 & 0 & 1/6 \\
1/2 & 1 & 1/2 & 1/6 & 1 & 1/2 & 2 & 1/6 & 2 \\
2 & 2 & 4 & 1 & 6 & 2 & 6 & 2 & 18
\end{pmatrix} \quad (7)$$

We set $L = \{x_1, t_2 x_2, t_2 x_1, t_1 x_1\}$ and let $\tau = x_1$. We then find $G = \{x_1 - t_1 - 1\}$ and L is updated to $\{t_2 x_2\}$.

We set $\tau = t_2 x_2$, find $G = \{x_1 - t_1 - 1, t_2 x_2 - 1\}$ and update L to \emptyset .

REMARK 3. *It is sometimes more convenient to obtain the ideal of relations, not in $\mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$ but in $\mathbb{K}(\mathbf{t})\langle \mathbf{x} \rangle$, the ring of skew-polynomials in \mathbf{x} with rational fractions in \mathbf{t} coefficients. In this setting, we can mimic SCALAR-FGLM algorithm by setting L to the set of all monomials in \mathbf{t}, \mathbf{x} of bidegree at most $(\delta, d+1)$ which are not in S' . In a way, this would ensure that the staircase of the Gröbner basis obtained at the end is closed. Then, we need to change also how to update L : since now each t_k is invertible, whenever $\tau = \mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}$, we remove every monomial $\mathbf{t}^{\mathbf{j}'} \mathbf{x}^{\mathbf{i}+\mathbf{i}'}$.*

Example 5. Algorithm 2 on the binomial table \mathbf{b} returns the three relations (1), (2a) and (2b) as $\{t_2 x_2 - (t_1 - t_2), (t_1 - t_2)x_1 - (t_1 + 1), x_1 x_2 - x_2 - 1\}$ for degree bounds at least (1, 2) in \mathbf{t} and \mathbf{x} and ordering $\text{DRL}(t_2 \prec t_1 \prec x_2 \prec x_1)$. Therefore during the execution of the algorithm, the least monomials of list L were $t_2 x_2, t_1 x_1$ and $x_1 x_2$.

Over $\mathbb{K}(\mathbf{t})\langle \mathbf{x} \rangle$, we would have to consider that $x_1 x_2$ is a multiple of $t_2 x_2$, when updating L . Therefore, Pascal's rule is never computed. This is consistent with the fact that $g_1 = t_2 x_2 - t_1 + t_2$ and $g_2 = (t_1 - t_2)x_1 - (t_1 + 1)$ form a Gröbner basis in $\mathbb{K}(\mathbf{t}, \mathbf{x})$, that $(x_1 x_2 - x_2 - 1) \notin I = \langle g_1, g_2 \rangle$ but that $g_1 + g_2 x_2 = (t_1 - t_2)(x_1 x_2 - x_2 - 1) \in I$. Hence, $x_1 x_2 - x_2 - 1$ is a member of the ideal $I \mathbb{K}(\mathbf{t})\langle \mathbf{x} \rangle$.

4.2 Adaptive algorithm

If one is concerned about the number of queries to table \mathbf{u} , it is necessary not to visit monomials yielding relations which are multiple of already computed relations. Therefore, we need to have an adaptive approach of the algorithm: we shall discover the staircase term by term, see also ADAPTIVE SCALAR-FGLM [5, Section 5]. As in Remark 3, we assume that we look for a Gröbner basis of the ideal in $\mathbb{K}(\mathbf{t})\langle \mathbf{x} \rangle$.

We start from $S = \emptyset, V = \emptyset$ and try to increase them both: if $H_{V \cup \{\mathbf{x}^{\mathbf{i}'}\}, S \cup \{\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}\}}$ has a greater rank than $H_{V, S}$, then S and V are updated accordingly. Otherwise, we have found a new relation whose leading term is $\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}$. Therefore, we do not update S nor V and shall never suggest any multiple of $\mathbf{x}^{\mathbf{i}}$ (resp. $\mathbf{x}^{\mathbf{i}'}$) as a new term in S (resp. V). This yields Algorithm 3.

5. MIXED APPROACH WITH GRÖBNER BASES COMPUTATIONS

When dealing with a sequence $\mathbf{u} = (u_i)_{i \in \mathbb{N}^n}$, not all its ideal of relations I in $\mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$ is necessarily interesting. For instance, whenever one wants to know if \mathbf{u} is P-recursive, it is enough to compute for all $k, 1 \leq k \leq n$, a polynomial

Algorithm 3: ADAPTIVE P-RECURSIVE-FGLM.

Input: A table $\mathbf{u} = (u_i)_{i \in \mathbb{N}^n}$ with coefficients in \mathbb{K} , (δ, d) given bounds and \prec a monomial ordering.

Output: A reduced Gröbner basis of a zero-dimensional ideal of degree $\geq d$.

$L := \{1\}, M := \{1\}$. // list of next terms to study
 $S := \{\}, V := \{\}$. // S is a useful staircase for V
 $G' := \{\}$. // leading terms of the Gröbner basis
 $H' := \{\}$. // deleted rows

While $L \neq \emptyset$ and $M \neq \emptyset$ **do**

$\tau := \min_{\prec} L, \chi := \min_{\prec} M$.

If $H_{V \cup \{\chi\}, S \cup \{\tau\}}$ is full rank **then**

$S := S \cup \{\tau\}, V := V \cup \{\chi\}$;

$L := L \cup \{t_1 \tau, \dots, t_n \tau, x_1 \tau, \dots, x_n \tau \mid \deg_{\mathbf{t}} t_i \tau \leq \delta\}$

$M := M \cup \{x_1 \chi, \dots, x_n \chi\}$

Sort L (resp. M) by increasing order (wrt. \prec) and remove duplicates and multiples of G' (resp. H').

If $\#S \geq d$ **then** // early termination

$G := \{\}$ and $G' := \text{MinGBasis}(G' \cup L \cup T \setminus S)$, with T the set of monomials in \mathbf{t}, \mathbf{x} of bidegree at most $(\deg_{\mathbf{t}} \tau, \deg_{\mathbf{x}} \tau + 1)$.

For all $\tau' \in G'$ **do**

$G := G \cup \{\tau' + \sum_{s \in S} \alpha_s s\}$ with $H_{V, S} \alpha + H_{V, \{\tau'\}} = 0$.

Return S, V and G .

Else

$G' := G' \cup \{\tau\}, H' := H' \cup \{\chi\}$ and remove multiples of τ in L and of χ in M .

Error "Run Algorithm 2".

in I whose leading monomial is $\mathbf{t}^{\mathbf{j}} x_k^d$ with \mathbf{j} any and $d > 0$. We recall that this ensures that $I \mathbb{K}(\mathbf{t})\langle \mathbf{x} \rangle$ is 0-dimensional. Therefore, only a subideal of I may be needed to certify that \mathbf{u} is P-recursive.

In ADAPTIVE P-RECURSIVE-FGLM algorithm, as soon as a polynomial P_1 with leading monomial $\text{LT}(P_1)$ is found, all the $\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}$ $\text{LT}(P_1)$ are discarded so that the multiples of P_1 are not computed. Therefore, whenever polynomials P_2, \dots, P_r are found, we can discard all the right-linear combinations of P_1, \dots, P_r , hence the ideal they span. In this algorithm, if P_1, \dots, P_r are exactly the first output polynomials, then we know that they are in the final Gröbner basis and that their leading monomials are the least first. By computing the Gröbner basis of $J = \langle P_1, \dots, P_r \rangle \subseteq \mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$, we can expect to be able to find polynomials with leading monomials not in $\langle \text{LT}(P_1), \dots, \text{LT}(P_r) \rangle$. On the one hand, these polynomials are redundant since they are deduced from P_1, \dots, P_r , i.e. the relations they represent are deduced by those induced by P_1, \dots, P_r . On the other hand, we may find polynomials ensuring that J is zero-dimensional and thus that I is.

Let us assume that when computing the Gröbner basis of J , we find another polynomial P_{r+1} . Let us assume furthermore that, seen as monomials in \mathbf{x} , $\text{LT}(P_1), \dots, \text{LT}(P_r) \prec \text{LT}(P_{r+1})$ and that P_{r+1} is also in the Gröbner basis of I . Then, we computed P_{r+1} without new queries to the table.

REMARK 4. *The Gröbner basis computation can be done either in $\mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$ or in $\mathbb{K}(\mathbf{t})\langle \mathbf{x} \rangle$. In the latter case, as soon as a relation with leading monomial $\mathbf{x}^{\mathbf{i}}$ is found, no relation with leading monomial $\mathbf{x}^{\mathbf{i}+\mathbf{i}'}$ shall be computed. However, in the former case, it is possible that we find new relations with leading monomials $\mathbf{x}^{\mathbf{i}+\mathbf{i}'}$, hence with higher degrees in \mathbf{x} , but with also lower degrees in \mathbf{t} .*

In the *black-box* model, this approach can be very beneficial, since the computation of even one element of \mathbf{u} can be costly. Therefore, we can modify ADAPTIVE P-RECURSIVE-

FGLM so that it takes these Gröbner bases computations into account: first we compute the polynomials as soon as they are found, instead of just storing their leading monomials in G' . Then, whenever $\#G$ is big enough, we replace G by the true Gröbner basis of the ideal spanned by G and we remove all the multiples of $\text{LT}(g)$, for $g \in G$, in L .

Computation of Gröbner bases in an Ore algebra can be done with the adaptations of Buchberger and F4 [14] algorithms in computer algebra systems [18], such as MAPLE or PLURAL the noncommutative extension of SINGULAR.

Let us illustrate this behavior with an example.

Example 6. We consider Gessel's walks and sequence $\mathbf{g} = (g_{n,i,j})_{(n,i,j) \in \mathbb{N}^3}$ denoting the number of walks of length n ending at (i, j) , with steps in $\{(-1, 0), (-1, -1), (1, 0), (1, 1)\}$. The kernel method, see [7] for instance, ensures that \mathbf{g} is P-recursive if and only if so are both $\mathbf{g}' = (g_{n,0,j})_{(n,j) \in \mathbb{N}^2}$ and $\mathbf{g}'' = (g_{n,i,0})_{(n,i) \in \mathbb{N}^2}$.

We apply this mixed approach to find a 0-dimensional subideal of relations of table \mathbf{g}' up to degrees $(5, 5)$. Variables t_0, x_0 shall be associated with n , while t_2, x_2 shall be associated with j . The first computed polynomial has leading monomial $t_0 t_2^3 x_0^3 x_2^3$. Then, four polynomials with leading monomials $t_0^4 t_2 x_0^4 x_2, t_0^3 t_2^2 x_0^4 x_2, t_0^3 t_2^2 x_0^3 x_2^3, t_0^4 x_0^4 x_2$.

Calling a Gröbner basis computation on these polynomials in $\mathbb{K}\langle \mathbf{t}, \mathbf{x} \rangle$ allows us to determine polynomials with leading monomials $t_0^5 t_2^2 x_0^4, t_0^5 x_0^6, t_0^3 t_2^4 x_2^4, t_0^3 t_2^3 x_2^5$. We can notice that the first (resp. last) two are purely in \mathbf{t} and x_0 (resp. x_2). Hence we can suspect that \mathbf{g}' is P-recursive.

Given a 3D-space walk sequence $\mathbf{w} = (w_{n,i,j,k})_{(n,i,j,k) \in \mathbb{N}^4}$ counting the number of walks of length n ending at (i, j, k) , the kernel method still applies. Thanks to the C-relation satisfied by said walk, one can prove, see [7, Section 4.3] that the generating series $W(\mathbf{x}) = \sum_{n,i,j,k \geq 0} w_{n,i,j,k} x_0^n x_1^i x_2^j x_3^k$ is a $\mathbb{K}\langle \mathbf{x} \rangle$ -linear combination of $W_1 = W(x_0, 0, x_2, x_3), W_2, W_3, W_{12} = W(x_0, 0, 0, x_3), W_{13}, W_{23}, W_{123} = W(x_0, 0, 0, 0)$ and 1. That is, there is a C-relation between $\mathbf{w}, (w_{n,0,j,k})_{(n,j,k) \in \mathbb{N}^3}, \dots, (w_{n,0,0,k})_{(n,k) \in \mathbb{N}^2}, \dots, (w_{n,0,0,0})_{n \in \mathbb{N}}$.

In the next example, we deal with the 3D-space walk \mathbf{w} whose steps are in $\{(-1, -1, -1), (-1, -1, 1), (-1, -1, 0), (1, 0, 0)\}$, see [7, Proposition 6].

Example 7. We use the same notation as in Example 6 with the extra convention that t_1, x_1 (resp. t_3, x_3) are associated with i (resp. k).

Running MULTI-SCALAR-FGLM on \mathbf{w} and all seven tables obtained from it when possibly setting parameters i, j, k to 0 yields the kernel equation

$$\begin{aligned} & (x_0 x_1^2 x_2 x_3 + x_0 x_2^2 x_3 + x_0 x_3^2 - x_1 x_2 x_3 + x_0) W \\ &= (x_0 x_2^2 x_3 + x_0 x_3^2 + x_0) W_1 + (x_0 x_3^2 + x_0) W_2 + x_0 W_3 \\ & \quad - (x_0 x_3^2 + x_0) W_{12} - x_0 (W_{13} + W_{23} - W_{123}) - x_1 x_2 x_3. \end{aligned}$$

Now, let us see why we can guess that W_3 is D-finite, i.e. $(w_{n,i,j,0})_{(n,i,j) \in \mathbb{N}^3}$ is P-recursive [19, Theorem 3.7]. The least relation for this table, $t_0 t_1 x_0 x_1 + t_1^2 x_0 x_1 + 2 t_1 x_0 x_1 - 2 t_0 t_1 - 4 t_0 - 2 t_1 - 4$, is found in degree $(\delta, d) = (2, 2)$.

Increasing gradually to column degree $(2, 4)$ and row degree 30, we obtain a 1448×350 matrix and 8 leading terms. A Gröbner basis computation yields 25 leading terms including $t_0 t_1 t_2 x_1^2$. This closes the staircase on the x_1 side.

We increase now the column degree to $(3, 4)$ and the row degree to 40, obtain a 2959×600 matrix and find an extra

25 relations. Unfortunately this does not close the staircase on either side.

A Gröbner basis computation, ideally truncated in total degree 9, returns 127 relations. Among which, there are leading monomials $t_0^3 t_2 t_3^2 x_0^4$ and $t_0 t_1^2 t_2 x_2^4$. In total, 19 monomials are purely in \mathbf{t} and one of the x_i . Hence, the staircase is closed in $\mathbb{K}\langle \mathbf{t} \rangle \langle \mathbf{x} \rangle$.

In Table 1, we detail the linear algebra part for W_3 . These computations have been integrated in and done with FGB [15].

Degree	Matrix		Rank	
(2, 2)	270×100	0.57 s	98	0.00 s
(2, 3)	766×200	0.60 s	181	0.00 s
(3, 3)	766×368	0.64 s	98	0.01 s

Table 1: Results and timings of Example 7.

6. COMPLEXITY ANALYSIS

This section is devoted to the study of the complexity of both MULTI-SCALAR-FGLM and P-RECURSIVE-FGLM algorithms. This comes down to the study of the structure of the multi-Hankel matrix.

6.1 Algorithm for tuples of tables

For the MULTI-SCALAR-FGLM algorithm, the multi-Hankel matrix $H_{U,T}$ is the concatenation of m multi-Hankel matrices H_{U,T_ℓ} . From [6, Section 6], we know that for an n -dimensional table and LEX($x_1 \prec \dots \prec x_n$) ordering, if the leading monomials are $x_1^{d_1}, \dots, x_2^{d_2}, \dots, x_n^{d_n}$, then the multi-Hankel matrix created by SCALAR-FGLM is a submatrix of a *multilevel block Hankel matrix of depth n* , or *n -multiblock Hankel matrix* which is heavily structured.

Definition 6. A *multilevel block Hankel matrix of depth 0* is a scalar. For any $n \in \mathbb{N}$, a *multilevel block Hankel matrix of depth $n + 1$* is a block Hankel matrix whose blocks are multilevel block Hankel matrices of depth n .

By construction in Algorithm 1, each T_ℓ of $T = T_1 e_1 \cup \dots \cup T_n e_n$ is a subset of U , therefore $H_{T_\ell, T_\ell e_\ell}$ is a submatrix of the generalized multi-Hankel matrix handled by the algorithm. Let $S = S_1 e_1 \cup \dots \cup S_m e_m$ and V be such that S is a useful staircase with V . A necessary condition for a monomial s to be in S_ℓ is that column $s e_\ell$ is linearly independent from the previous ones in $H_{T_\ell, T_\ell e_\ell}$. Thus, row $s e_\ell$ is also independent from the previous ones in $H_{U,T}$. Hence, $s \in S_\ell$ implies that $s \in V$ and each S_ℓ is a subset of V .

Let $\Sigma = \bigcup_{\ell=1}^m S_\ell$, then $\Sigma \subseteq V$. Let M be the least integer such that $V \subseteq W = \Sigma \cup \bigcup_{k=2}^M w_k \Sigma$. Now, $H_{V,S}$ is a submatrix of $H_{W, \Sigma e_1, \dots, \Sigma e_m}$ and $H_{W, \Sigma e_1, \dots, \Sigma e_m}$ is a block matrix whose blocks $H_{w_k \Sigma, \Sigma}$ are all n -multiblock:

$$H_{W, \Sigma e_1, \dots, \Sigma e_m} = \begin{pmatrix} H_{\Sigma, \Sigma}(\mathbf{u}^{(1)}) & \cdots & H_{\Sigma, \Sigma}(\mathbf{u}^{(m)}) \\ \vdots & & \vdots \\ H_{w_M \Sigma, \Sigma}(\mathbf{u}^{(1)}) & \cdots & H_{w_M \Sigma, \Sigma}(\mathbf{u}^{(m)}) \end{pmatrix}.$$

PROPOSITION 5. *Given a tuple of m tables. Let μ be the number of polynomials in the Gröbner basis output by MULTI-SCALAR-FGLM. For $1 \leq k \leq n$, let d_k be the greatest powers in x_k appearing in the Gröbner basis. Let M be the number of blocks of size $d_1 \cdots d_n$ appearing on each column of the computed matrix by the algorithm. Then the*

number of operations in the base field to compute the Gröbner basis is no more than

$$O\left(\mu(M m d_2 \cdots d_n)^{\omega-1} M(d_1 \cdots d_n) \log(d_1 \cdots d_n)\right).$$

PROOF. Applying Hankel displacement operator makes each block $H_{w_k \Sigma, \Sigma}$ have displacement rank $d_2 \cdots d_n$, see [6, Proposition 17]. Therefore, $H_{W, \Sigma e_1 \cup \dots \cup \Sigma e_m}$ have displacement rank $M m d_2 \cdots d_n$. Thanks to [9], solving a linear system with this matrix can be done in $O((M m d_1 \cdots d_n)^{\omega-1} M(d_1 \cdots d_n) \log(d_1 \cdots d_n))$. Finally, we solve μ systems. \square

6.2 Algorithms for P-recursive tables

For $\mathbf{j} \in \mathbb{N}^n$, let $\Sigma_{\mathbf{j}}$ be the set of all monomials $\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}}$ in the useful staircase. By the column removal criterion of Proposition 2, we know that $\Sigma_{\mathbf{j}+\mathbf{j}'} \subseteq \Sigma_{\mathbf{j}}$ for any $\mathbf{j}' \in \mathbb{N}^n$. Hence for all \mathbf{j} , $\Sigma_{\mathbf{j}} \subseteq \Sigma_0 = \Sigma$.

The remaining part of the analysis is the same as for MULTI-SCALAR-FGLM called on $m = \binom{n+\delta}{\delta} = O(n^\delta)$ tables. This yields the following corollary.

COROLLARY 6. *Let δ be the maximal degree in \mathbf{t} allowed to compute the relations of a table. Let μ be the number of polynomials in the Gröbner basis output by P-RECURSIVE-FGLM. For $1 \leq k \leq n$, let d_k be the greatest powers in x_k appearing in the Gröbner basis. Let M be the number of blocks of size $d_1 \cdots d_n$ appearing on each column of the computed matrix by the algorithm. Then the number of operations in the base field to compute the Gröbner basis is no more than*

$$O\left(\mu(M n^\delta d_2 \cdots d_n)^{\omega-1} M(d_1 \cdots d_n) \log(d_1 \cdots d_n)\right).$$

Concerning the number of table queries, Algorithm 2, P-RECURSIVE-FGLM, can reach elements of \mathbf{u} far away from the useful staircase whenever the degree bound in \mathbf{x} is too big. In Algorithm 3, ADAPTIVE P-RECURSIVE-FGLM, the matrix elements are $[\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}+\mathbf{i}'}] = (\mathbf{i} + \mathbf{i}')^{\mathbf{j}} u_{\mathbf{i}+\mathbf{i}'}$ with $\mathbf{t}^{\mathbf{j}} \mathbf{x}^{\mathbf{i}} \in S$, $\mathbf{x}^{\mathbf{i}} \in \Sigma$ and $\mathbf{x}^{\mathbf{i}'} \in V$. Hence, the number of table queries is the cardinal of $\Sigma V = \{\tau \chi \mid \tau \in \Sigma, \chi \in V\}$.

Embedding V into $\Sigma \cup \bigcup_{k=2}^M w_k \Sigma$ ensures that $\Sigma V \subseteq (2\Sigma) \cup \bigcup_{k=2}^M w_k (2\Sigma)$, where $2\Sigma = \{\tau \tau' \mid \tau, \tau' \in \Sigma\}$. Hence, $\#(\Sigma V) \leq M \#(2\Sigma) \leq M (\#\Sigma)^2 / 2$ in the worst case.

Whenever Σ is included in a parallelotope of size $C \#\Sigma$, then Ruzsa's Theorem [22, Theorem 1.1] ensures that there exists c such that $\#(2\Sigma) \leq c \#\Sigma$, hence $\#(\Sigma V) \leq c M \#\Sigma$.

Acknowledgments

We thank the anonymous referees for their careful reading and their helpful comments, C. Banderier, A. Bostan, F. Chyzak and B. Salvy for valuable discussions. This work has been partly supported by the French National Research Agency ANR-11-BS02-0013 HPAC project.

7. REFERENCES

- [1] C. Banderier and P. Flajolet. Basic analytic combinatorics of directed lattice paths. *Theoret. Comput. Sci.*, 281(1–2):37–80, 2002. Selected Papers in honour of Maurice Nivat.
- [2] B. Beckermann and G. Labahn. A uniform approach for the fast computation of matrix-type pade approximants. *SIAM J. Matrix Anal. Appl.*, 15(3):804–823, 1994.
- [3] A. Benoit, F. Chyzak, A. Darrasse, S. Gerhold, M. Mezzarobba, and B. Salvy. The Dynamic Dictionary of Mathematical Functions (DDMF). In K. Fukuda, J. v. d. Hoeven, M. Joswig, and N. Takayama, editors, *Mathematical Software – ICMS 2010*, pages 35–41, Berlin, Heidelberg, 2010. Springer.
- [4] E. Berlekamp. Nonbinary BCH decoding. *IEEE Trans. Inform. Theory*, 14(2):242–242, 1968.
- [5] J. Berthomieu, B. Boyer, and J.-Ch. Faugère. Linear Algebra for Computing Gröbner Bases of Linear Recursive Multidimensional Sequences. In *Proc. of the 40th ISSAC*, pages 61–68, Bath, UK, 2015.
- [6] J. Berthomieu, B. Boyer, and J.-Ch. Faugère. Linear Algebra for Computing Gröbner Bases of Linear Recursive Multidimensional Sequences. Extended version, preprint available at <https://hal.inria.fr/hal-01253934>, Dec. 2015.
- [7] A. Bostan, M. Bousquet-Mélou, M. Kauers, and S. Melczer. On 3-dimensional lattice walks confined to the positive octant. To appear in *Annals of Combinatorics*, 2014.
- [8] A. Bostan, F. Chyzak, M. van Hoeij, and L. Pech. Explicit formula for the generating series of diagonal 3D rook paths. *Séminaire Lotharingien de Combinatoire*, B66a, 2011.
- [9] A. Bostan, C.-P. Jeannerod, and É. Schost. Solving Toeplitz- and Vandermonde-like Linear Systems with Large Displacement Rank. In C. W. Brown, editor, *Proc. of the 32nd ISSAC*, pages 33–40. ACM Press, 2007.
- [10] A. Bostan, K. Raschel, and B. Salvy. Non-D-finite excursions in the quarter plane. *J. Combin. Theory Ser. A*, 121:45–63, 2014.
- [11] M. Bousquet-Mélou and M. Mishna. Walks with small steps in the quarter plane. In *Algorithmic probability and combinatorics*, volume 520 of *Contemp. Math.*, pages 1–39. Amer. Math. Soc., Providence, RI, 2010.
- [12] M. Bousquet-Mélou and M. Petkovšek. Walks confined in a quadrant are not always d-finite. *Theoret. Comput. Sci.*, 307(2):257–276, 2003. Random Generation of Combinatorial Objects and Bijective Combinatorics.
- [13] F. Chyzak. *The ABC of Creative Telescoping – Algorithms, Bounds, Complexity*. Accreditation to supervise research, École polytechnique X, 2014.
- [14] J.-Ch. Faugère. A new efficient algorithm for computing Gröbner bases (F4). *J. Pure Appl. Algebra*, 139(1–3):61–88, 1999.
- [15] J.-Ch. Faugère. FGb: A Library for Computing Gröbner Bases. In K. Fukuda, J. v. d. Hoeven, M. Joswig, and N. Takayama, editors, *Mathematical Software – ICMS 2010*, pages 84–87, Berlin, Heidelberg, 2010. Springer.
- [16] J.-Ch. 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.
- [17] P. Fitzpatrick and G. Norton. Finding a basis for the characteristic ideal of an n-dimensional linear recurring sequence. *Information Theory, IEEE Transactions on*, 36(6):1480–1487, 1990.
- [18] V. Levandovskyy. *Non-commutative Computer Algebra for polynomial algebras: Gröbner bases, applications and implementation*. PhD thesis, Technische Universität Kaiserslautern, 2005.
- [19] L. Lipshitz. D-finite power series. *Journal of Algebra*, 122(2):353–373, 1989.
- [20] J. L. Massey. Shift-register synthesis and BCH decoding. *IEEE Trans. Inform. Theory*, IT-15:122–127, 1969.
- [21] A. Quadrat and D. Robertz. A constructive study of the module structure of rings of partial differential operators. *Acta Applicandae Mathematicae*, 133(1):187–234, 2014.
- [22] I. Z. Ruzsa. Generalized arithmetical progressions and sunsets. *Acta Math. Hungar.*, 65(4):379–388, 1994.
- [23] 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.
- [24] 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.
- [25] S. Sakata. Extension of the Berlekamp-Massey algorithm to N Dimensions. *Inform. and Comput.*, 84(2):207–239, 1990.
- [26] 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, 2009.
- [27] B. Salvy. D-finiteness: Algorithms and applications. In *Proc. of the 30th ISSAC*, pages 2–3, New York, NY, USA, 2005. ACM.
- [28] R. Stanley. Differentially finite power series. *European J. Combin.*, 1(2):175–188, 1980.
- [29] D. Zeilberger. A holonomic systems approach to special functions identities. *J. Comput. Appl. Math.*, 32(3):321–368, 1990.