In-depth comparison of the Berlekamp–Massey–Sakata and the Scalar-FGLM algorithms: the adaptive variants

Jérémy Berthomieu, Jean-Charles Faugère

In-depth comparison of the Berlekamp–Massey–Sakata and the Scalar-FGLM algorithms: the adaptive variants

Jérémy Berthomieu[a,*], Jean-Charles Faugère[b,a]

[a] *Sorbonne Université,* CNRS*,* INRIA*,* LIP6*, Équipe* PolSys*, F-75005 Paris, France*
[b] *CryptoNext Security*

Abstract

The Berlekamp–Massey–Sakata algorithm and the Scalar-FGLM algorithm both compute the ideal of relations of a multidimensional linear recurrent sequence.

Whenever quering a single sequence element is prohibitive, the bottleneck of these algorithms becomes the computation of all the needed sequence terms. As such, having adaptive variants of these algorithms, reducing the number of sequence queries, becomes mandatory.

A native adaptive variant of the Scalar-FGLM algorithm was presented by its authors, the so-called Adaptive Scalar-FGLM algorithm.

In this paper, our first contribution is to make the Berlekamp–Massey–Sakata algorithm more efficient by making it adaptive to avoid some useless relation testings. This variant allows us to divide by four in dimension 2 and by seven in dimension 3 the number of basic operations performed on some sequence family.

Then, we compare the two adaptive algorithms. We show that their behaviors differ in a way that it is not possible to tweak one of the algorithms in order to mimic exactly the behavior of the other. We detail precisely the differences and the similarities of both algorithms and conclude that in general the Adaptive Scalar-FGLM algorithm needs fewer queries and performs fewer basic operations than the Adaptive Berlekamp–Massey–Sakata algorithm.

We also show that these variants are always more efficient than the original algorithms.

Keywords: The BMS algorithm, the Scalar-FGLM algorithm, Gröbner basis

*LIP6, Sorbonne Université, Campus Pierre-et-Marie-Curie, boîte courrier 169, 4 place Jussieu, F-75252 Paris Cedex 05, France.

Email addresses: jeremy.berthomieu@lip6.fr (Jérémy Berthomieu), jean-charles.faugere@inria.fr (Jean-Charles Faugère)

computation, multidimensional linear recurrent sequence, algorithms comparison

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 with constant coefficients satisfied by S or the length of the shortest linear feedback shift register (LFSR) which generates it.

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 $-\sum_{k=1}^{d} \alpha_k\, u_{i-k}$. Least-square sense means that the solution minimizes the sum of the squares of the errors.

This yields a linear system whose matrix is Hankel. This problem has also been extensively used in Digital Signal Processing theory and applications. Numerically, Levinson–Durbin recursion method can be used to solve this problem. Hence, to some extent, the original Levinson–Durbin problem in Norbert Wiener's Ph.D. thesis, Levinson (1947); Wiener (1964), predates the Hankel interpretation of the Berlekamp–Massey algorithm, see for instance Jonckheere and Ma (1989).

The Berlekamp–Massey algorithm (BM, Berlekamp (1968); Massey (1969)) is a famous algorithm guessing a solution of this problem for a one-dimensional sequence. This algorithm has been tremendously studied and many variants were designed. We refer the reader to Kaltofen and Pan (1991); Kaltofen and Yuhasz (2013a,b) for a very nice classification of the BM algorithms for solving this problem, and for its generalization to matrix sequences.

A generalization of the BM algorithm to 2 dimensions was first designed in Sakata (1988). It was then further generalized to n dimensions in Sakata (1990, 2009). The so-called Berlekamp–Massey–Sakata algorithm (BMS) guesses a Gröbner basis of the ideal of relations satisfied by the first terms of the input sequence, (Sakata, 1990, Lemma 5).

In Berthomieu et al. (2015, 2017), the authors designed the Scalar-FGLM algorithm. It also guesses a reduced Gröbner basis of the ideal of relations of a sequence. While the BM algorithm can be seen as the computation of the kernel of a Hankel matrix, the Scalar-FGLM algorithm computes the kernel of a *multi-Hankel* matrix, its multivariate generalization.

In some applications, computing even a term of the input sequence is costly or even the bottleneck of the Scalar-FGLM algorithm. An adaptive variant of the algorithm, called the Adaptive Scalar-FGLM algorithm was designed in Berthomieu et al. (2015, 2017) in order to minimize the number of sequence queries.

More recently, the authors proposed a new algorithm, Polynomial Scalar-FGLM, in Berthomieu and Faugère (2018) for computing the linear recurrence relations of a sequence based on multivariate polynomial arithmetic. It extends the BMS algorithm through the use of polynomial divisions and is a complete revision of the Scalar-FGLM algorithm without any linear algebra operations. Yet, in this paper the algorithms are treated as high-level ones, with linear algebra operations. We do not try to improve them using polynomial arithmetic as in Berthomieu and Faugère (2018).

Finally, let us recall that as it is not possible to store the whole input sequence, all these algorithms take a bound as an input and only handle sequence terms up to this index bound. This is why they can only *guess* the ideal of relations.

1.1. Related works

Computing linear recurrence relations of multi-dimensional sequences finds applications in Coding Theory, Computer Algebra and Combinatorics.

Historically, the BM algorithm was designed to decode cyclic codes, like the BCH codes, Bose and Ray-Chaudhuri (1960); Hocquenghem (1959). Therefore, decoding n-dimensional cyclic codes, a generalization of Reed Solomon codes, was Sakata's motivation for designing the BMS algorithm in Sakata (1991). Let us also mention that extensions of cyclic codes on a projective curve can be fastly decoding using a modified version of the vector-BM and vector-BMS algorithms, see for instance Lee (2016); Sakata and Fujisawa (2018).

On the other hand, as the output of the BMS and the Scalar-FGLM algorithms is a Gröbner basis, a natural application in Computer Algebra is the computation of a Gröbner basis of an ideal for another order, typically from a total degree ordering to an elimination ordering. In fact the latest versions of the Sparse-FGLM algorithm rely heavily on the BM and BMS algorithms, see Faugère and Mou (2011, 2017). These notions are recalled in a concise way in Section 2, see also (Berthomieu and Faugère, 2017, Section 2).

Finally, computing linear recurrence relations with *polynomial* coefficients finds applications in Computer Algebra for computing properties of univariate and multivariate Special Functions. The Dynamic Dictionary of Mathematical Functions (DDMF, Benoit et al. (2010)) generates automatically web-pages on univariate special functions through the differential equations they satisfy. Equivalently, they could be generated through the linear recurrence relations satisfied by their Taylor series sequence of coefficients. Deciding whether 2D/3D-space walks are D-finite or not finds applications in Combinatorics, see Banderier and Flajolet (2002); Bostan et al. (2014); Bousquet-Mélou and Mishna (2010); Bousquet-Mélou and Petkovšek (2003). This motivated the authors to extend the Scalar-FGLM algo-

rithm to handle relations with polynomial coefficients in Berthomieu and Faugère (2016).

1.2. Contributions

Following the open question in Berthomieu and Faugère (2017) whether an adaptive variant of the BMS algorithm, reducing the number of sequence queries, exists or not, first we answer positively. Then, the goal of this paper is to compare this adaptive variant and the Adaptive Scalar-FGLM algorithm.

In Section 3, we design an adaptive variant of the BMS algorithm, namely the Adaptive BMS algorithm, reducing the number of sequence queries. To our knowledge some early termination criteria were proposed for the BMS algorithm, see Sakata (2009). However, these criteria did not allow one to skip some relation testings. Here, the Adaptive BMS algorithm can skip some relation testings and still test some further relations. In practice, this variant is more efficient than the BMS algorithm thanks to these skippings. To do so, it uses an a priori upper bound on the staircase size to prevent some useless relation testings. In some favorable cases, this can even allow us to require fewer sequence elements than when calling the BMS algorithm. The presentation of this variant follows the linear algebra description of the BMS algorithm introduced in (Berthomieu and Faugère, 2017, Section 3.2), see also Section 2.4.2.

In Section 4, we deal with the Adaptive Scalar-FGLM algorithm, first presented in Berthomieu et al. (2015). Compared to the BMS algorithm, we iteratively increase the size of the staircase. Although it can drastically decrease the number of sequence queries, one of its drawback is that it can fail to compute the true ideal of relations of a sequence.

Therefore, it is essential to investigate when these algorithms output a Gröbner basis of the ideal of relations. To do so, we focus on their similarities and differences of behaviors. We report here simplified and synthetic versions of the results obtained in Section 5.

A first similarity is that they both output a zero-dimensional ideal of relations.

Theorem 1 (Theorem 13)**.** *Let* $\mathbf{u} = (u_{i,j})_{(i,j)\in\mathbb{N}^2}$ *be a sequence, let* $\prec$ *be a degree monomial ordering and* d *be the size of the staircase.*

Calling each algorithm on $\mathbf{u}$*,* $\prec$*,* d *yields a truncated Gröbner basis of a zero-dimensional ideal.*

In the Gröbner basis change of ordering application, like the Sparse-FGLM algorithm, one needs to use the lexicographical ordering. Although the BMS algorithm is not designed to handle such an ordering, the Adaptive BMS can perfectly be called with this ordering. Indeed, if the ideal is in *shape position*, then, as a second similarity, both algorithms output correctly the ideal.

Theorem 2 (Theorem 16)**.** *Let* $\mathbf{u} = (u_{i,j})_{(i,j)\in\mathbb{N}^2}$ *be a linear recurrent sequence whose ideal of relations* $I = \langle g(y), x - f(y)\rangle$ *is in* shape position *for the* $\text{LEX}(y \prec x)$ *ordering, with* $\deg f < \deg g = d$ *and* g *squarefree.*

Assuming no error is thrown in the execution of the Adaptive Scalar-FGLM *algorithm called on* $\mathbf{u}$*,* d *and* $\text{LEX}(y \prec x)$ *ordering, then the ouput ideal is* I*.*

Likewise, calling the Adaptive BMS *algorithm on* $\mathbf{u}$*,* d *and* $\text{LEX}(y \prec x)$ *yields ideal* I*.*

Although, the previous two theorems seem to show that both algorithms have very similar outputs, their outputs can still differ.

Indeed, as neither algorithm can test if their output relations are valid on the whole sequence, they intrinsically return the *shifts* of the relations: that is the set of translation monomials for which the relations are valid. Thus, the larger the shift, the more the relation has been tested. Therefore, it reinforces the confidence one can have in the guessed output ideal. Even if both algorithms output the same ideal, they usually do so while outputting different shifts.

Theorem 3 (Theorem 15)**.** *Let* $\mathbf{u} = (u_{i,j})_{(i,j)\in\mathbb{N}^2}$ *be a sequence,* $\prec$ *be a monomial ordering and* d *be the size of the output staircase* S*. Let us assume that both algorithms return a common relation* g *when called on* $\mathbf{u}$*,* $\prec$*,* d *and some stopping monomial* M *for the* Adaptive BMS *algorithm.*

Then, the shift associated to g *the* Adaptive BMS *algorithm yields is the monomial set* $\{m,\ m \ \text{LM}(g) \preceq M\}$*. In other words, the smaller* $\text{LM}(g)$*, the larger its shift.*

The shift associated to g *the* Adaptive Scalar-FGLM *algorithm returns is either* S *if* $\text{LM}(g) \succ \max_{\prec}(S)$ *or* $\{m \in S,\ m \prec \text{LM}(g)\} \cup \{\text{LM}(g)\}$ *otherwise. In other words, the larger* $\text{LM}(g)$*, the larger its shift.*

As a consequence of these differences of behavior, it is not possible to tweak one of the algorithms in order to mimic exactly the behavior of the other.

Finally, in Section 6, we compare both algorithms based on the number of sequence queries they perform and their number of basic operations. We show that the Adaptive BMS algorithm is able to perform four (resp. seven) times fewer operations than the BMS algorithm to ouput the ideal of relations of a family of bidimensional (resp. tridimensional) sequences.

We also show that the Adaptive Scalar-FGLM needs fewer queries and fewer basic operations to recover the whole ideal of relations of several families of sequences. However, it seems that asymptotically the ratios between the number of basic operations and the number of sequence queries made by both algorithm could be the same.

1.3. Conclusion and Perspectives

We now understand better the advantages of each algorithm.

On the one hand, the Adaptive Scalar-FGLM algorithm can fail to return the right answer, yet, on the other hand, we can tweak it to test the computed relations further, allowing us to discard wrong relations. Furthermore, generally it returns the right ideal of relations and it usually does so faster than the Adaptive BMS algorithm.

However, the Adaptive BMS algorithm seems to be the safer one. If the upper bounds on the staircase size is correct, it will always return the right ideal of relations. Though, its performance speedup relies on the number of skipped relation testings and thus on the sharpness of this bound. Moreover, it seems hard to predict in advance which monomials will be totally skipped during the execution of the algorithm.

Combining the design of the Polynomial Scalar-FGLM algorithm, based on polynomial arithmetic in Berthomieu and Faugère (2018), and the comparison of the Adaptive BMS and Adaptive Scalar-FGLM algorithms in this paper could lead to the design of an hybrid algorithm taking advantage of all these algorithms. In particular, this algorithm could replace the linear algebra arithmetic by a polynomial one.

Indeed, the goal would be to mix the efficiency of the polynomial arithmetic in the Polynomial Scalar-FGLM algorithm and the small number of queries performed by the Adaptive BMS and the Adaptive Scalar-FGLM algorithms to compute the relations.

2. Preliminaries

In this section, we give a brief description of classical notation used all along the paper. We refer the reader to (Berthomieu and Faugère, 2017, Section 2) for a more detailed presentation.

2.1. Sequences and relations

For $n \geq 1$, we let $\mathbf{i} = (i_1, \ldots, i_n) \in \mathbb{N}^n$, where $\mathbb{N}$ is the set of natural numbers including 0. Classically, we write $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{x}^{\mathbf{i}} = x_1^{i_1} \cdots x_n^{i_n}$. An n-dimensional sequence $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i} \in \mathbb{N}^n}$ over a field $\mathbb{K}$ satisfies the (linear recurrence) relation induced by $\boldsymbol{\alpha} = (\alpha_{\mathbf{k}})_{\mathbf{k} \in \mathcal{K}} \in \mathbb{K}^{|\mathcal{K}|}$, with $\mathcal{K} \subset \mathbb{N}^n$ finite if

$$\forall \mathbf{i} \in \mathbb{N}^n, \sum_{\mathbf{k} \in \mathcal{K}} \alpha_{\mathbf{k}} \, u_{\mathbf{k}+\mathbf{i}} = 0. \tag{1}$$

Example 1. *Let* $\mathbf{b}$ *be the* 2*-dimensional sequence of the binomial coefficients,* $\mathbf{b} = \left(\binom{i}{j}\right)_{(i,j)\in\mathbb{N}^2}$. *Then the Pascal's rule:*

$$\forall (i,j) \in \mathbb{N}^2,\ \mathbf{b}_{i+1,j+1} - \mathbf{b}_{i,j+1} - \mathbf{b}_{i,j} = 0$$

is a linear recurrence relation for the sequence $\mathbf{b}$*.*

As we can only work with a finite number of terms of a sequence, in this paper, a *table* shall denote a finite subset of terms of a sequence: it is one of the input parameters of the algorithms.

Given a finite table extracted from the sequence $\mathbf{u}$, the main purpose of the BMS and the Scalar-FGLM algorithms is to, loosely speaking, determine a minimal set of relations that will allow us to generate this finite table using only the values of $\mathbf{u}$ on their supports.

Relations satisfied by a sequence can be added and shifted, therefore it is natural to associate them with multivariate polynomials in $\mathbb{K}[\mathbf{x}]$.

Definition 1. *Let* $f = \sum_{\mathbf{k}\in\mathcal{K}} \alpha_{\mathbf{k}}\, \mathbf{x}^{\mathbf{k}} \in \mathbb{K}[\mathbf{x}]$. *We will denote by* $[f]_{\mathbf{u}}$*, or* $[f]$ *when no ambiguity arises, the linear combination* $\sum_{\mathbf{k}\in\mathcal{K}} \alpha_{\mathbf{k}}\, u_{\mathbf{k}}$. *Moreover, if* $\boldsymbol{\alpha}$ *defines a relation for* $\mathbf{u}$*, that is for all* $\mathbf{i} \in \mathbb{N}^n$*,* $\left[\mathbf{x}^{\mathbf{i}}\, f\right] = 0$*, then we say that* f *is the polynomial of this relation.*

The main benefit of the [ ] notation resides in the immediate fact that for all index $\mathbf{i}$, $\left[\mathbf{x}^{\mathbf{i}}\, f\right] = \sum_{\mathbf{k}\in\mathcal{K}} \alpha_{\mathbf{k}}\, u_{\mathbf{k}+\mathbf{i}}$.

In the previous example, the Pascal's rule relation is associated with polynomial $P = x\,y - y - 1$, so that

$$\forall (i,j) \in \mathbb{N}^2,\ [x^i\, y^j\, P] = 0.$$

Definition 2 (Fitzpatrick and Norton (1990); Sakata (1988))**.** *Let* $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ *be an* n*-dimensional sequence with coefficients in* $\mathbb{K}$*. The sequence* $\mathbf{u}$ *is* linear recurrent *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.*

Equivalently, $\mathbf{u}$ *is linear recurrent if its ideal of relations* $\{f,\ \forall\, m \in \mathbb{K}[\mathbf{x}], [m\, f] = 0\}$ *is* zero-dimensional*.*

2.2. Gröbner bases

Let $\mathcal{T} = \{\mathbf{x}^{\mathbf{i}},\ \mathbf{i} \in \mathbb{N}^n\}$ be the set of all monomials in $\mathbb{K}[\mathbf{x}]$. A monomial ordering $\prec$ on $\mathbb{K}[\mathbf{x}]$ is an order relation satisfying the following three classical properties:

1. for all $m \in \mathcal{T}$, $1 \preceq m$;

2. for all $m, m', s \in \mathcal{T}$, $m \prec m' \Rightarrow m\, s \prec m'\, s$;
3. every subset of $\mathcal{T}$ has a least element for $\prec$.

For a monomial ordering $\prec$ on $\mathbb{K}[\mathbf{x}]$, the *leading monomial* of f, denoted $\textsc{lm}(f)$, is the greatest monomial in the support of f for $\prec$. The *leading coefficient* of f, denoted $\textsc{lc}(f)$, is the nonzero coefficient of $\textsc{lm}(f)$. The *leading term* of f, $\textsc{lt}(f)$, is defined as $\textsc{lt}(f) = \textsc{lc}(f)\ \textsc{lm}(f)$. For an ideal I, we denote, classically, $\textsc{lm}(I) = \{\textsc{lm}(f),\ f \in I\}$.

We recall briefly the definition of a Gröbner basis and a staircase.

Definition 3. *Let I be a nonzero ideal of $\mathbb{K}[\mathbf{x}]$ and let $\prec$ be a monomial ordering. A set $\mathcal{G} \subseteq I$ is a* Gröbner basis *of I if for all $f \in I$, there exists $g \in \mathcal{G}$ such that $\textsc{lm}(g) | \textsc{lm}(f)$.*

The set $\mathcal{G}$ is a minimal *Gröbner basis of I if for any $g \in \mathcal{G}$, $\mathcal{G} \setminus \{g\}$ does not span I.*

Furthermore, $\mathcal{G}$ is (minimal) reduced *if for any $g, g' \in \mathcal{G}$, $g \neq g'$ and any monomial $m \in \operatorname{supp} g'$, $\textsc{lt}(g) \nmid m$.*

Let $\mathcal{G}$ be a reduced truncated Gröbner basis, the staircase *of $\mathcal{G}$ is*

$$S = \text{Staircase}(\mathcal{G}) = \{s \in \mathcal{T},\ \forall\, g \in \mathcal{G}, \textsc{lm}(g) \nmid s\}.$$

It is also the canonical basis of $\mathbb{K}[\mathbf{x}]/I$.

Gröbner basis theory allows us to choose any monomial ordering $\prec$. Among all the monomial ordering, we will mainly use the

- $\textsc{lex}(x_n \prec \cdots \prec x_1)$ ordering which compares monomials as follows $\mathbf{x}^{\mathbf{i}} \prec \mathbf{x}^{\mathbf{i}'}$ if, and only if, there exists k, $1 \leq k \leq n$ such that for all $\ell < k$, $i_\ell = i'_\ell$ and $i_k < i'_k$, see (Cox et al., 2015, Chapter 2, Definition 3);

- $\textsc{drl}(x_n \prec \cdots \prec x_1)$ order which compares monomials as follows $\mathbf{x}^{\mathbf{i}} \prec \mathbf{x}^{\mathbf{i}'}$ if, and only if, $i_1 + \cdots + i_n < i'_1 + \cdots + i'_n$ or $i_1 + \cdots + i_n = i'_1 + \cdots + i'_n$ and there exists k, $2 \leq k \leq n$ such that for all $\ell > k$, $i_\ell = i'_\ell$ and $i_k > i'_k$. Equivalently, there exists k, $1 \leq k \leq n$ such that for all $\ell > k$, $i_1 + \cdots + i_\ell = i'_1 + \cdots + i'_\ell$ and $i_1 + \cdots + i_k < i'_1 + \cdots + i'_k$, see (Cox et al., 2015, Chapter 2, Definition 6).

However, in the BMS algorithm, we need to be able to enumerate all the monomials up to a bound monomial. This forces the user to take an ordering $\prec$ such that for all $M \in \mathcal{T}$, the set $\{m \prec M,\ m \in \mathcal{T}\}$ is finite. Such an ordering $\prec$ makes $(\mathbb{N}^n, \prec)$ isomorphic to $(\mathbb{N}, <)$, thus it makes sense to speak about the next monomial for $\prec$.

This request excludes for instance the $\textsc{lex}$ ordering, and more generally any elimination ordering. In other words, only weighted degree ordering, or *weight ordering*, should be used.

2.3. Multi-Hankel matrices

A matrix $H \in \mathbb{K}^{m\times n}$ is *Hankel*, if there exists a sequence $\mathbf{u} = (u_i)_{i\in\mathbb{N}}$ such that for all $(i, i') \in \{1, \ldots, m\} \times \{1, \ldots, n\}$, the coefficient $h_{i,i'}$ lying on the ith row and i'th column of H satisfies $h_{i,i'} = u_{i+i'}$.

In a multivariate setting, we can extend this Hankel matrices notion to *multi-Hankel* matrices. Indexing the rows and columns with monomials $\mathbf{x}^{\mathbf{i}} = x_1^{i_1} \cdots x_n^{i_n}$ and $\mathbf{x}^{\mathbf{i}'} = x_1^{i'_1} \cdots x_n^{i'_n}$, the coefficient of H lying on the row labeled with $\mathbf{x}^{\mathbf{i}}$ and column labeled with $\mathbf{x}^{\mathbf{i}'}$ is $u_{\mathbf{i}+\mathbf{i}'}$. Given two sets of monomials U and T, we let $H_{U,T}$ be the multi-Hankel matrix with rows (resp. columns) indexed with monomials in U (resp. T).

Example 2. *Let* $\mathbf{u} = (u_{i,j})_{(i,j)\in\mathbb{N}^2}$ *be a sequence.*

1. *Let* $U = \{1, y, y^2, x, xy, xy^2, x^2, x^2y, x^2y^2\}$ *and* $T = \{1, y, x, xy, x^2, x^2y, x^3, x^3y\}$, *then*

$$H_{U,T} = \begin{array}{c} \\ 1 \\ y \\ y^2 \\ x \\ xy \\ xy^2 \\ x^2 \\ x^2y \\ x^2y^2 \end{array} \begin{array}{c} \begin{array}{cc|cc|cc|cc} 1 & y & x & xy & x^2 & x^2y & x^3 & x^3y \end{array} \\ \left(\begin{array}{cc|cc|cc|cc} u_{0,0} & u_{0,1} & u_{1,0} & u_{1,1} & u_{2,0} & u_{2,1} & u_{3,0} & u_{3,1} \\ u_{0,1} & u_{0,2} & u_{1,1} & u_{1,2} & u_{2,1} & u_{2,2} & u_{3,1} & u_{3,2} \\ u_{0,2} & u_{0,3} & u_{1,2} & u_{1,3} & u_{2,2} & u_{2,3} & u_{3,2} & u_{3,3} \\ \hline u_{1,0} & u_{1,1} & u_{2,0} & u_{2,1} & u_{3,0} & u_{3,1} & u_{4,0} & u_{4,1} \\ u_{1,1} & u_{1,2} & u_{2,1} & u_{2,2} & u_{3,1} & u_{3,2} & u_{4,1} & u_{4,2} \\ u_{1,2} & u_{1,3} & u_{2,2} & u_{2,3} & u_{3,2} & u_{3,3} & u_{4,2} & u_{4,3} \\ \hline u_{2,0} & u_{2,1} & u_{3,0} & u_{3,1} & u_{4,0} & u_{4,1} & u_{5,0} & u_{5,1} \\ u_{2,1} & u_{2,2} & u_{3,1} & u_{3,2} & u_{4,1} & u_{4,2} & u_{5,1} & u_{5,2} \\ u_{2,2} & u_{2,3} & u_{3,2} & u_{3,3} & u_{4,2} & u_{4,3} & u_{5,2} & u_{5,3} \end{array}\right) \end{array}.$$

   We can see that this matrix is a 3×4 block-Hankel *matrix with Hankel blocks of size* 3×2.

2. *Let* $T = \{1, y, x, y^2, xy, x^2, y^3, xy^2, x^2y, x^3\}$, *then the following matrix has a less obvious structure:*

$$H_{T,T} = \begin{array}{c} \\ 1 \\ y \\ x \\ y^2 \\ xy \\ x^2 \\ y^3 \\ xy^2 \\ x^2y \\ x^3 \end{array} \begin{array}{c} \begin{array}{cccccccccc} 1 & y & x & y^2 & xy & x^2 & y^3 & xy^2 & x^2y & x^3 \end{array} \\ \left(\begin{array}{cccccccccc} u_{0,0} & u_{0,1} & u_{1,0} & u_{0,2} & u_{1,1} & u_{2,0} & u_{0,3} & u_{1,2} & u_{2,1} & u_{3,0} \\ u_{0,1} & u_{0,2} & u_{1,1} & u_{0,3} & u_{1,2} & u_{2,1} & u_{0,4} & u_{1,3} & u_{2,2} & u_{3,1} \\ u_{1,0} & u_{1,1} & u_{2,0} & u_{1,2} & u_{2,1} & u_{3,0} & u_{1,3} & u_{2,2} & u_{3,1} & u_{4,0} \\ u_{0,2} & u_{0,3} & u_{1,2} & u_{0,4} & u_{1,3} & u_{2,2} & u_{0,5} & u_{1,4} & u_{2,3} & u_{3,2} \\ u_{1,1} & u_{1,2} & u_{2,1} & u_{1,3} & u_{2,2} & u_{3,1} & u_{1,4} & u_{2,3} & u_{3,2} & u_{4,1} \\ u_{2,0} & u_{2,1} & u_{3,0} & u_{2,2} & u_{3,1} & u_{4,0} & u_{2,3} & u_{3,2} & u_{4,1} & u_{5,0} \\ u_{0,3} & u_{0,4} & u_{1,3} & u_{0,5} & u_{1,4} & u_{2,3} & u_{0,6} & u_{1,5} & u_{2,4} & u_{3,3} \\ u_{1,2} & u_{1,3} & u_{2,2} & u_{1,4} & u_{2,3} & u_{3,2} & u_{1,5} & u_{2,4} & u_{3,3} & u_{4,2} \\ u_{2,1} & u_{2,2} & u_{3,1} & u_{2,3} & u_{3,2} & u_{4,1} & u_{2,4} & u_{3,3} & u_{4,2} & u_{5,1} \\ u_{3,0} & u_{3,1} & u_{4,0} & u_{3,2} & u_{4,1} & u_{5,0} & u_{3,3} & u_{4,2} & u_{5,1} & u_{6,0} \end{array}\right) \end{array}.$$

2.4. *The* BMS *algorithm*

As in Guisse (2016), we specialize to $\mathbb{K}[\mathbf{x}]$ the presentation of the BMS algorithm given in Bras-Amorós and O'Sullivan (2006), Cox et al. (2005) and Sakata (2009) in the more general case of ordered domains.

2.4.1. *A Polynomial interpretation of the* BMS *algorithm*

Given a table $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ and a weight ordering $\prec$ for $\mathbf{x}$. We let $\mathcal{T}_0 = \{0\} \cup \{\mathbf{x}^{\mathbf{i}},\ \mathbf{i} \in \mathbb{N}^n\}$ and extend $\prec$ (still denoted by $\prec$) to $\mathcal{T}_0$ with the convention that $0 \prec 1$.

The goal is to iterate on a monomial m, by only considering, at each step, the table $(u_{\mathbf{i}})_{\mathbf{i}\in\{\mathbf{k},\ \mathbf{x}^{\mathbf{k}}\preceq m\}}$. As we only know partially the table $\mathbf{u}$, we need to define some notions according to this partial knowledge at step m.

Definition 4. *Let $m \in \mathcal{T}_0$. Let $f \in \mathbb{K}[\mathbf{x}]$, we say that the relation f is* valid up to *m, whenever*

$$\forall t \in \mathcal{T}_0,\ \mathrm{LM}(t\,f) \preceq m \Rightarrow [t\,f] = 0.$$

We thus define the shift *of f as* $\mathrm{shift}(f) = \frac{m}{\mathrm{LM}(f)}$.

We say that the relation f fails *at m whenever*

$$\forall t \in \mathcal{T}_0,\ \mathrm{LM}(t\,f) \prec m \Rightarrow [t\,f] = 0,\quad \left[\frac{m}{\mathrm{LM}(f)}\,f\right] \neq 0.$$

We define the fail *of f as* $\mathrm{fail}(f) = m$. *If the relation f never fails, that is for all $t \in \mathcal{T}_0$, $[t\,f] = 0$, then by convention* $\mathrm{fail}(f) = \mathrm{shift}(f) = +\infty$.

Proposition 4. *Let $\mathbf{u}$ be a table and $f \in \mathbb{K}[x]$ such that* $\mathrm{fail}(f) \succ m$. *For all $g \in \mathbb{K}[\mathbf{x}]$, if* $\mathrm{LM}(g\,f) \preceq m$, *then $[g\,f] = 0$.*

The following proposition show how to combine two failing relations with the same shift in order to obtain a new relation valid with a bigger shift.

Proposition 5. *Let f_1 and f_2 be two relations such that $v = \frac{\mathrm{fail}(f_1)}{\mathrm{LM}(f_1)} = \frac{\mathrm{fail}(f_2)}{\mathrm{LM}(f_2)}$ and $e_1 = [v\,f_1]$, $e_2 = [v\,f_2]$. Let f be the nonzero polynomial $f_1 - \frac{e_1}{e_2}\,f_2$. Then, for $i \in \{1, 2\}$,* $\mathrm{fail}(f) \succ \mathrm{fail}(f_i)$, *i.e.* $\frac{\mathrm{fail}(f)}{\mathrm{LM}(f)} \succ v$.

Proof. For any $c \in \mathbb{K}$ and any $\mu \in \mathbb{K}[\mathbf{x}]$ such that $\mathrm{LM}(g) \prec v$, we have $[\mu\,(f_1 + c\,f_2)] = [\mu\,f_1] + c\,[\mu\,f_2] = 0$, hence $\mathrm{fail}(f_1 + c\,f_2) \succeq \mathrm{fail}(f_i)$.

It remains to prove that for a good choice of c, we have a strict inequality: as, $[v\,(f_1 + c\,f_2)] = [v\,f_1] + c\,[v\,f_2] = e_1 + c\,e_2$, it is clear that $[v\,f] = [v\,(f_1 - \frac{e_1}{e_2}\,f_2)] = 0$, so that $\mathrm{fail}(f) \succ v\ \mathrm{LM}(f) \succeq \mathrm{fail}(f_i)$. □

Definition 5. *Using the same notation as in Definition 3, we let*

$$I_m = \{f \in \mathbb{K}[\mathbf{x}],\ \mathrm{fail}\,(f) \succ m\},$$

and $\mathcal{G}_m$ be the least elements for $\prec$ of I_m, it is a truncated Gröbner basis of I_m:

$$\mathcal{G}_m = \min_{\prec}\{g,\ g \in I_m\},$$
$$S_m = \mathrm{Staircase}(\mathcal{G}_m).$$

Example 3. *Let us go back to Example 1 with sequence* $\mathbf{b} = \left(\binom{i}{j}\right)_{(i,j)\in\mathbb{N}^2}$. *Consider* $\mathbb{K}[x, y]$ *with the* DRL$(y \prec x)$ *ordering, and* $m = x^2$.

y^2	0		
y	0	1	
1	1	1	1
	1	x	x^2

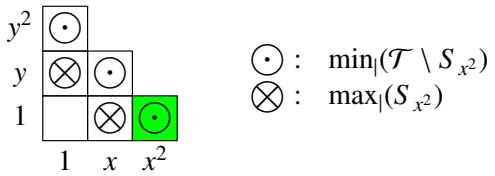
From this table, on the one hand, we can deduce that

- *since it is not identically* 0*, there is no relation with leading monomial* 1 *valid up to x^2, hence $1 \in S_{x^2}$;*
- *since $[y + \alpha] = \alpha$ and $[x(y + \alpha)] = 1 + \alpha$, there is no relation with leading monomial y valid up to xy and thus x^2, hence $y \in S_{x^2}$;*
- *since $[y(x + \beta y + \alpha)] = 1$, there is no relation with leading monomial x valid up to xy and thus x^2, hence $x \in S_{x^2}$.*

On the other hand, we can check that

- *since $[y^2] = 0$, relation y^2 is valid up to y^2 and thus x^2, hence $y^2 \in \mathcal{T} \setminus S_{x^2}$;*
- *since $[xy - 1] = 0$, relation $xy - 1$ is valid up to xy and thus x^2, hence $xy \in \mathcal{T} \setminus S_{x^2}$;*
- *since $[x^2 - x] = 0$, relation $x^2 - x$ is valid up to x^2, hence $x^2 \in \mathcal{T} \setminus S_{x^2}$.*

Therefore, $S_{x^2} = \{1, y, x\}$, $\max_{|}(S_{x^2}) = \{y, x\}$ and $\min_{|}(\mathcal{T} \setminus S_{x^2}) = \{y^2, xy, x^2\}$. This is summed up in the following diagram.



Let us notice that many relations with respective leading monomials $y^2, x\,y, x^2$ suit actually. These would be $y^2 - \alpha_1\,x + \alpha_y\,y + \alpha_1, x\,y - (1+\alpha_1)\,x + \alpha_y\,y + \alpha_1$ and $x^2 - (1+\alpha_1)\,x + \alpha_y\,y + \alpha_1$. Furthermore, I_{x^2} is not stable by addition: $(x^2 - x), (x^2 - 2\,x + 1) \in I_{x^2}$ but $x^2 - x - (x^2 - 2\,x + 1) = (x - 1) \notin I_{x^2}$ since $\mathrm{fail}\,(x-1) = x\,y$. Hence, I_{x^2} is not an ideal of $\mathbb{K}[x, y]$.

For $m = x^3$, with the following table, we find that

y^3	0			
y^2	0	0		
y	0	1	2	
1	1	1	1	1
	1	x	x^2	x^3

- *since $[y^2] = [y\,y^2] = [x\,y^2] = 0$, then y^2 is valid up to $x\,y^2$ and thus x^3;*
- *since $[x\,y - 1] = [y\,(x\,y - 1)] = 0$ and $[x\,(x\,y - y)] = 1$, then $x\,y - 1$ fails at $x^2\,y$. Yet, since $[y] = [y\,y] = 0$ and $[x\,y] = 1$, then by Proposition 5, $[x\,y - y - 1] = [y\,(x\,y - y - 1)] = 0$ and $[x\,(x\,y - y - 1)]$ vanishes as well. Hence, $x\,y - y - 1$ is valid up to $x^2\,y$ and thus x^3;*
- *since $[x^2 - x] = 0$ and $[y\,(x^2 - x)] = 1$, then $x^2 - x$ fails at $x^2\,y$. Likewise, since $[x - 1] = 0$ and $[y\,(x-1)] = 1$, then $[x^2 - 2\,x + 1] = 0$ and $[y\,(x^2 - 2\,x + 1)] = 0$. Furthermore, $[x\,(x^2 - 2\,x + 1)] = 0$, so that $x^2 - 2\,x + 1$ is valid up to x^3.*

Therefore, $S_{x^3} = \{1, y, x\}$, $\max_{|}(S_{x^3}) = \{y, x\}$ and $\min_{|}(\mathcal{T} \setminus S_{x^3}) = \{y^2, x\,y, x^2\}$. We can also check that these relations span the only valid relations with support in $S_{x^3} \cup \{y^2, x\,y, x^2\}$.

y^3				
y^2	⊙			
y	⊗	⊙		
1		⊗	⊙	
	1	x	x^2	x^3

Although I_m is not an ideal in general, we have the following results:

Proposition 6. *Using the notation of Definitions 4 and 5,*

1. *I_m is closed under multiplication by elements of $\mathbb{K}[\mathbf{x}]$,*
2. *for all monomials t, t' such that $t|t'$,*
   (a) *if $t' \in S_m$, then $t \in S_m$.*

(b) *if* $t \in \mathcal{T} \setminus S_m$, *then* $t' \in \mathcal{T} \setminus S_m$,

Moreover, it is clear that the sequence $(I_m)_{m\in\mathcal{T}_0}$ is decreasing and that if $\mathbf{u}$ is linear recurrent then $I = \bigcap_{m\in\mathcal{T}_0} I_m$. Therefore, $(S_m)_{m\in\mathcal{T}_0}$ is increasing and its limit is S the finite target staircase. Hence, for m big enough, S_m will be the target staircase. We will give an upper bound in Proposition 9.

The following result gives an intrinsic characterization of S_m that is key in the iteration of the BMS algorithm.

Proposition 7. *For all monomial* $m \in \mathcal{T}_0$, $S_m = \left\{\frac{\mathrm{fail}(f)}{\mathrm{LM}(f)},\ f \notin I_m\right\}$.

Furthermore, let m^+ *be the successor of* m. *Let* s *be a monomial in the staircase* S_{m^+}. *Then,* s *was added at step* m^+, *i.e.* $s \notin S_m$, *if, and only if,* $s|m^+$ *and* $\frac{m^+}{s} \in S_{m^+} \setminus S_m$.

Proof. We shall prove the first assertion by double inclusion. If $s = \frac{\mathrm{fail}(f)}{\mathrm{LM}(f)}$ then for all $g \in \mathbb{K}[\mathbf{x}]$ such that $\mathrm{LM}(g) = s$, $\mathrm{fail}(g) \preceq m$, hence $s \notin \mathrm{LM}(I_m)$, $s \in S_m$.

The reverse inclusion is proved by induction on m. For $m = 0$, $S_m = \emptyset$ and there is nothing to do. Let us assume the inclusion is satisfied for a monomial m.

Let $s \in S_{m^+}$. On the one hand, if $s \in S_m$, then there exists $f \in \mathbb{K}[\mathbf{x}] \setminus I_m \subseteq \mathbb{K}[\mathbf{x}] \setminus I_{m^+}$ such that $s = \frac{\mathrm{fail}(f)}{\mathrm{LM}(f)}$.

If, on the other hand, $s \in S_{m^+} \setminus S_m$, then there exists a relation $f \in \mathbb{K}[\mathbf{x}]$ such that $\mathrm{LM}(f) = s$, and $m \prec \mathrm{fail}(f) \preceq m^+$, hence $\mathrm{fail}(f) = m^+$ and s divides m^+.

Let us assume that for all $g \in \mathbb{K}[\mathbf{x}]$ with $\mathrm{LM}(g) = \frac{m^+}{s}$, we have $\mathrm{fail}(g) \preceq m \prec m^+$. Therefore, $\frac{m^+}{s} \in S_m$ and there exists $h \notin I_m$ such that $\frac{\mathrm{fail}(h)}{\mathrm{LM}(h)} = \frac{m^+}{s}$. By Proposition 5, there is $\alpha \in \mathbb{K}$ such that $\mathrm{fail}(f - \alpha\, h) \succ m^+$. Since $\mathrm{fail}(h) \preceq m \prec m^+$, then $\mathrm{LM}(h) \preceq s$ and $\mathrm{LM}(f - \alpha\, h) = s$, hence $\frac{\mathrm{fail}(f-\alpha\, h)}{\mathrm{LM}(f-\alpha\, h)} \succ \frac{m^+}{s}$. This contradicts the fact that $\frac{m^+}{s} \in S_m$. Thus there exists a $g \in \mathbb{K}[\mathbf{x}]$ with $\mathrm{LM}(g) = \frac{m^+}{s}$ and $\mathrm{fail}(g) \succeq m^+$.

Let g be such a relation, since $\mathrm{fail}(f) = m^+$, then $[g\ f] \neq 0$ and $\mathrm{fail}(g) = m^+$. Therefore, $\frac{\mathrm{fail}(g)}{\mathrm{LM}(g)} = \frac{m^+}{m^+/s} = s$ so that $s \in \left\{\frac{\mathrm{fail}(f)}{\mathrm{LM}(f)},\ f \notin I_{m^+}\right\}$.

Now, we proved that $s \in S_{m^+} \setminus S_m$ implies $s|m^+$ and $\frac{m^+}{s} \in S_{m^+} \setminus S_m$. This implication is clearly an equivalence. □

From this proposition it follows that if $m \in \mathcal{T}_0$, and if m^+ is its successor:

$$\max_{|}(S_{m^+}) = \max_{|}\left(\max_{|}(S_m) \cup \left\{\frac{m^+}{s},\ s \in \min_{|}(\mathcal{T} \setminus S_m) \cap S_{m^+}\right\}\right) \tag{2}$$

Relation 2 allows us to construct, iterating on the monomial m, the set of relations G_m representing the truncated Gröbner basis of I_m. Relations $g \in G_m$ are indexed by their leading monomials, describing $\mathcal{T} \setminus S_m$.

Remark 8. *We can also construct another set, describing the edge of S_m, still denoted S_m, as there is a one-to-one correspondence between a staircase and its edge. The relations $h \in S_m$ are indexed by their ratio $\frac{\text{fail}(h)}{\text{LM}(h)}$ between their fail and their leading monomial, describing the full staircase of I_m.*

When two relations h and h' in S_m are such that $\frac{\text{fail}(h)}{\text{LM}(h)} = \frac{\text{fail}(h')}{\text{LM}(h')}$, then we only need to keep one. Since the goal is to combine a relation of S_m with a relation failing at m^+ to make a new one with a bigger shift, as in Proposition 5, it is best to handle smaller polynomials.

This yields Algorithm 1.

Algorithm 1: The BMS algorithm.

Input: A table $\mathbf{u} = (u_\mathbf{i})_{\mathbf{i}\in\mathbb{N}^n}$ with coefficients in $\mathbb{K}$, a monomial ordering $\prec$ and a monomial M as the stopping condition.

Output: A set G of relations generating I_M.

$T := \{m \in \mathbb{K}[\mathbf{x}],\ m \preceq M\}$. `// ordered for` $\prec$

$G := \{1\}$. `// the future Gröbner basis`

$S := \emptyset$. `// staircase edge, elements will be` $[h, \text{fail}(h)/\text{LM}(h)]$

For all $m \in T$ **do**

  $S' := S$.

  For $g \in G$ **do**

    If $\text{LM}(g) | m$ **then**

      $e := \left[\frac{m}{\text{LM}(g)}\, g\right]_\mathbf{u}$.

      If $e \neq 0$ **then**

        $S' := S' \cup \left\{\left[\frac{g}{e}, \frac{m}{\text{LM}(g)}\right]\right\}$.

  $S' := \min_|\{[h, \text{fail}(h)/\text{LM}(h)]\}$. `// see Remark 8`

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

  For $g' \in G'$ **do**

    Let $g \in G$ such that $\text{LM}(g) | \text{LM}(g')$.

    If $\text{LM}(g) \nmid m$ **then**

      $g' := \frac{\text{LM}(g')}{\text{LM}(g)}\, g$. `// translates the relation`

    Else if $\exists\, h \in S, \frac{m}{\text{LM}(g')} | \text{fail}(h)$ **then**

      $g' := \frac{\text{LM}(g')}{\text{LM}(g)}\, g - \left[\frac{m}{\text{LM}(g)}\, g\right]_\mathbf{u} \frac{\text{LM}(g')\ \text{fail}(h)}{m}\, h$. `// see Proposition 5`

    Else $g' := g$.

  $G := G'$.

  $S := S'$.

Return G.

We saw that for m big enough, S_m will be the target staircase. We now give an upper bound that can be found in (Bras-Amorós and O'Sullivan, 2006, Proposition 10)

Proposition 9. *Let* $\mathbf{u}$ *be a linear recurrent sequence and I be its ideal of relations.*

Let S be the staircase of I for $\prec$*. Let* $s_{\max}$ *be the largest monomial in S. Then, for* $m \geq (s_{\max})^2$*,* $S_m = S$*.*

Let $\mathcal{G}$ *be a minimal Gröbner basis of I for* $\prec$ *and let* $g_{\max}$ *be the largest leading monomial of* $\mathcal{G}$*. Then, for* $m \geq s_{\max} \cdot \max_{\prec}(g_{\max}, s_{\max})$*, the* BMS *algorithm returns a minimal Gröbner basis of I for* $\prec$*.*

Example 4. *For the* $\textsc{drl}(y \prec x)$ *ordering,* $I = \langle x^p, y^q \rangle$ *and* $q > p \geq 1$*, we have,* $s_{\max} = x^{p-1}\, y^{q-1}$ *and* $g_{\max} = y^q$*. Therefore, the right staircase is found at most at step* $m = x^{2\,p-2}\, y^{2\,q-2}$*, while the Gröbner basis is found at most at step* $x^{p-1}\, y^{q-1}\ \max_{\prec}(x^{p-1}\, y^{q-1}, y^q)$*, i.e.* $y^{2\,q-1}$ *if* $p = 1$ *and* $x^{2\,p-2}\, y^{2\,q-2}$ *otherwise.*

From Propositions 7 and 9, we can deduce that $S = \left\{ \frac{\text{fail}(f)}{\text{LM}(f)},\ f \notin I \right\}$.

Example 5. *We give the trace of the algorithm called on the binomial sequence* $\mathbf{b}$ *for the* $\textsc{drl}(y \prec x)$ *ordering up to monomial* x^3 *(hence visiting all the monomials of degree at most* 3*).*

To simplify the reading, whenever a relation succeeds in m or cannot be tested in m, we skip the updating part as this relation remains the same.

We start with the empty staircase S and the relation $G = \{1\}$*.*

For the monomial 1

The relation $g_1 = 1$ *fails since* $[\mathbf{b}_{0,0}] = 1$*. Thus* $S' = \{[1, 1]\}$*.*
S′ is updated to $\{[1, 1]\}$ *and* $G' = \{y, x\}$*.*
For the relation $g'_1 = y$*,* $y \nmid 1$ *thus* $g'_1 = y$*.*
For the relation $g'_2 = x$*,* $x \nmid 1$ *thus* $g'_2 = x$*.*
We update $G := G' = \{y, x\}$ *and* $S := S' = \{[1, 1]\}$*.*

For the monomial y

The relation $g_1 = y$ *succeeds since* $[\mathbf{b}_{0,1}] = 0$*.*
Nothing must be done for the relation $g_2 = x$*.*
S′ is set to $\{[1, 1]\}$ *and* $G' = \{y, x\}$*.*
We set $g'_1 = y$ *and* $g'_2 = x$*.*
We update $G := G' = \{y, x\}$ *and* $S := S' = \{[1, 1]\}$*.*

For the monomial x

Nothing must be done for the relation $g_1 = y$*.*
The relation $g_2 = x$ *fails since* $[\mathbf{b}_{1,0}] = 1$*. Thus* $S' = \{[1, 1], [x, 1]\}$*.*
S′ is set to $\{[1, 1]\}$ *and* $G' = \{y, x\}$*.*
We set $g'_1 = y$*.*
For the relation $g'_2 = x$*,* $x | x$ *and* $\frac{x}{x} | \,\text{fail}(1)$*, hence* $g'_2 = x - 1$*.*
We update $G := G' = \{y, x - 1\}$ *and* $S := S' = \{[1, 1]\}$*.*

For the monomial y^2

The relation $g_1 = y$ *succeeds since* $[\mathbf{b}_{0,2}] = 0$.
Nothing must be done for the relation $g_2 = x - 1$.
S' *is set to* $\{[1, 1]\}$ *and* $G' = \{y, x\}$.
We set $g'_1 = y$ *and* $g'_2 = x - 1$.
We update $G := G' = \{y, x - 1\}$ *and* $S := S' = \{[1, 1]\}$.

For the monomial $x\,y$

The relation $g_1 = y$ *fails since* $[\mathbf{b}_{1,1}] = 1$. *Thus* $S' = \{[1, 1], [y, x]\}$.
The relation $g_2 = x - 1$ *fails since* $[\mathbf{b}_{1,1} - \mathbf{b}_{0,1}] = 1$. *Thus* $S' = \{[1, 1], [y, x], [x - 1, y]\}$.
S' *is set to* $\{[y, x], [x - 1, y]\}$ *and* $G' = \{y^2, x\,y, x^2\}$.
For the relation $g'_1 = y^2$, $y^2 \nmid x\,y$ *thus* $g'_1 = y^2$.
For the relation $g'_2 = x\,y$, $x\,y | x\,y$ *and* $\frac{x\,y}{x\,y} | \,\mathrm{fail}(y)$, *hence* $g'_2 = x\,y - 1$.
For the relation $g'_3 = x^2$, $x^2 \nmid x\,y$ *thus* $g'_3 = x^2 - x$.
We update $G := G' = \{y^2, x\,y - 1, x^2 - x\}$ *and* $S := S' = \{[y, x], [x - 1, y]\}$.

For the monomial x^2

Nothing must be done for the relation $g_1 = y^2$.
Nothing must be done for the relation $g_2 = x\,y - 1$.
The relation $g_3 = x^2 - x$ *succeeds since* $[\mathbf{b}_{2,0} - \mathbf{b}_{1,0}] = 0$.
S' *is set to* $\{[y, x], [x - 1, y]\}$ *and* $G' = \{y^2, x\,y, x^2\}$.
We set $g'_1 = y^2$, $g'_2 = x\,y - 1$ *and* $g'_3 = x^2 - x$.
We update $G := G' = \{y^2, x\,y - 1, x^2 - x\}$ *and* $S := S' = \{[y, x], [x - 1, y]\}$.

For the monomial y^3

The relation $g_1 = y^2$ *succeeds since* $[\mathbf{b}_{0,3}] = 0$.
Nothing must be done for the relation $g_2 = x\,y - 1$.
Nothing must be done for the relation $g_3 = x^2 - x$.
S' *is set to* $\{[y, x], [x - 1, y]\}$ *and* $G' = \{y^2, x\,y, x^2\}$.
We set $g'_1 = y^2$, $g'_2 = x\,y - 1$ *and* $g_3 = x^2 - x$.
We update $G := G' = \{y^2, x\,y - 1, x^2 - x\}$ *and* $S := S' = \{[y, x], [x - 1, y]\}$.

For the monomial $x\,y^2$

The relation $g_1 = y^2$ *succeeds since* $[\mathbf{b}_{1,2}] = 0$.
The relation $g_2 = x\,y - 1$ *succeeds since* $[\mathbf{b}_{1,2} - \mathbf{b}_{0,1}] = 0$.
Nothing must be done for the relation $g_3 = x^2 - x$.
S' *is set to* $\{[y, x], [x - 1, y]\}$ *and* $G' = \{y^2, x\,y, x^2\}$.
We set $g'_1 = y^2$, $g'_2 = x\,y - 1$ *and* $g_3 = x^2 - x$.
We update $G := G' = \{y^2, x\,y - 1, x^2 - x\}$ *and* $S := S' = \{[x, y], [y, x - 1]\}$.

For the monomial $x^2\,y$

Nothing must be done for the relation $g_1 = y^2$.
The relation $g_2 = x\,y - 1$ *fails since* $[\mathbf{b}_{2,1} - \mathbf{b}_{1,0}] = 1$. *Thus* $S' = \{[y, x], [x - 1, y], [x\,y - 1, x]\}$.

The relation $g_3 = x^2 - x$ *fails since* $[\mathbf{b}_{2,1} - \mathbf{b}_{1,1}] = 1$. *Thus* $S' = \{[y, x], [x - 1, y], [x\,y - 1, x], [x^2 - x, y]\}$.
S' is set to $\{[y, x], [x - 1, y]\}$ *and* $G' = \{y^2, x\,y, x^2\}$.
We set $g'_1 = y^2$.

For the relation $g'_2 = x\,y$, $x\,y | x^2\,y$ *and* $\frac{x^2\,y}{x\,y} | \operatorname{fail}(y)$, *hence* $g'_3 = x\,y - y - 1$.

For the relation $g'_3 = x^2$, $x^2 | x^2\,y$ *and* $\frac{x^2\,y}{x^2} | \operatorname{fail}(x - 1)$, *hence* $g'_3 = x^2 - 2\,x + 1$.
We update $G := G' = \{y^2, x\,y - y - 1, x^2 - 2\,x + 1\}$ *and* $S := S' = \{[y, x], [x - 1, y]\}$.

For the monomial x^3

Nothing must be done for the relation $g_1 = y^2$.
Nothing must be done for the relation $g_2 = x\,y - y - 1$.
The relation $g_3 = x^2 - 2\,x + 1$ *succeeds since* $[\mathbf{b}_{3,0} - 2\,\mathbf{b}_{2,0} + \mathbf{b}_{1,0}] = 0$.
S' is set to $\{[y, x], [x - 1, y]\}$ *and* $G' = \{y^2, x\,y, x^2\}$.
We set $g'_1 = y^2$, $g'_2 = x\,y - y - 1$ *and* $g_3 = x^2 - 2\,x + 1$.
We update $G := G' = \{y^2, x\,y - y - 1, x^2 - 2\,x + 1\}$ *and* $S := S' = \{[y, x], [x - 1, y]\}$.

For the monomial y^4

The relation $g_1 = y^2$ *succeeds since* $[\mathbf{b}_{0,4}] = 0$.
Nothing must be done for the relation $g_2 = x\,y - y - 1$.
Nothing must be done for the relation $g_3 = x^2 - 2\,x + 1$.
S' is set to $\{[y, x], [x - 1, y]\}$ *and* $G' = \{y^2, x\,y, x^2\}$.
We set $g'_1 = y^2$, $g'_2 = x\,y - y - 1$ *and* $g_3 = x^2 - 2\,x + 1$.
We update $G := G' = \{y^2, x\,y - y - 1, x^2 - 2\,x + 1\}$ *and* $S := S' = \{[y, x], [x - 1, y]\}$.

For the monomial $x\,y^3$

The relation $g_1 = y^2$ *succeeds since* $[\mathbf{b}_{1,3}] = 0$.
The relation $g_2 = x\,y - y - 1$ *succeeds since* $[\mathbf{b}_{1,3} - \mathbf{b}_{0,3} - \mathbf{b}_{0,2}] = 0$.
Nothing must be done for the relation $g_3 = x^2 - 2\,x + 1$.
S' is set to $\{[y, x], [x - 1, y]\}$ *and* $G' = \{y^2, x\,y, x^2\}$.
We set $g'_1 = y^2$, $g'_2 = x\,y - y - 1$ *and* $g_3 = x^2 - 2\,x + 1$.
We update $G := G' = \{y^2, x\,y - y - 1, x^2 - 2\,x + 1\}$ *and* $S := S' = \{[y, x], [x - 1, y]\}$.

For the monomial $x^2\,y^2$

The relation $g_1 = y^2$ *fails since* $[\mathbf{b}_{2,2}] = 1$. *Thus* $S' = \{[y, x], [x - 1, y], [y^2, x^2]\}$.
The relation $g_2 = x\,y - y - 1$ *succeeds since* $[\mathbf{b}_{2,2} - \mathbf{b}_{1,2} - \mathbf{b}_{1,1}] = 0$.
The relation $g_3 = x^2 - 2\,x + 1$ *fails since* $[\mathbf{b}_{2,2} - 2\,\mathbf{b}_{1,2} + \mathbf{b}_{0,2}] = 1$. *Thus* $S' = \{[y, x], [x - 1, y], [y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$.

S' is set to $\{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$ and $G' = \{x\,y, y^3, x^3\}$.
We set $g'_1 = x\,y - y - 1$.
For the relation $g'_2 = y^3$, $y^3 \nmid x^2\,y$ thus $g'_2 = y^3$.
For the relation $g'_3 = x^3$, $x^3 \nmid x^2\,y$ thus $g'_3 = x^3 - 2\,x^2 + x$.
We update $G := G' = \{x\,y - y - 1, y^3, x^3 - 2\,x^2 + x\}$ and $S := S' = \{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$.

For the monomial $x^3\,y$

The relation $g_1 = x\,y - y - 1$ succeeds since $[\mathbf{b}_{3,1} - \mathbf{b}_{2,1} - \mathbf{b}_{2,0}] = 0$.
Nothing must be done for the relation $g_2 = y^3$.
The relation $g_3 = x^3 - 2\,x^2 + 1$ succeeds since $[\mathbf{b}_{3,1} - 2\,\mathbf{b}_{2,1} + \mathbf{b}_{1,1}] = 0$.
S' is set to $\{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$ and $G' = \{x\,y, y^3, x^3\}$.
We set $g'_1 = x\,y - y - 1$, $g'_2 = y^3$ and $g'_3 = x^3 - 2\,x^2 + x$.
We update $G := G' = \{x\,y - y - 1, y^3, x^3 - 2\,x^2 + x\}$ and $S := S' = \{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$.

For the monomial x^4

Nothing must be done for the relation $g_1 = x\,y - y - 1$.
Nothing must be done for the relation $g_2 = y^3$.
The relation $g_3 = x^3 - 2\,x^2 + 1$ succeeds since $[\mathbf{b}_{4,0} - 2\,\mathbf{b}_{3,0} + \mathbf{b}_{2,0}] = 0$.
S' is set to $\{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$ and $G' = \{x\,y, y^3, x^3\}$.
We set $g'_1 = x\,y - y - 1$, $g'_2 = y^3$ and $g'_3 = x^3 - 2\,x^2 + x$.
We update $G := G' = \{x\,y - y - 1, y^3, x^3 - 2\,x^2 + x\}$ and $S := S' = \{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$.

The algorithm returns relations $x\,y - y - 1, y^3, x^3 - 2\,x^2 + x$, all three with a shift x.

2.4.2. A Linear Algebra interpretation of the BMS *algorithm*

In order to make the presentation of the BMS algorithm closer to that of the Scalar-FGLM algorithm, we propose to replace every evaluation using the [ ] operator with a matrix-vector product.

As stated above, given a monic relation $f = \text{LM}(f) + \sum_{s \in S} \alpha_s\,s$, testing the shift of this relation by a monomial m is done with the bracket operator, i.e. testing whether $[m\,f] = 0$ or not. Denoting $\vec{f}$, the vector

$$\vec{f} = \begin{array}{c} \\ \vdots \\ s \in S \\ \vdots \\ \text{LM}(f) \end{array} \begin{array}{c} 1 \\ \left(\begin{array}{c} \vdots \\ \alpha_s \\ \vdots \\ 1 \end{array} \right) \end{array},$$

this can also be done through testing if the following matrix-vector product

$$H_{m,S\cup\{\mathrm{LM}(f)\}}\,\vec{f} = m \begin{pmatrix} & \overset{\cdots}{\cdots} & \overset{s\in S}{[m\ s]} & \overset{\cdots}{\cdots} & \overset{\mathrm{LM}(f)}{[m\ \mathrm{LM}(f)]} \end{pmatrix} \begin{pmatrix} \vdots \\ \alpha_s \\ \vdots \\ 1 \end{pmatrix} = 0$$

or not. In this setting, the definitions of the *shift* and the *fail* of a relation, i.e. Definition 4, become as follows.

Definition 6. *Let $f = \mathrm{LT}(f) + \sum_{s\in S} \alpha_s\, s$ be a polynomial.*

The monomial m is a shift of f *if*

$$H_{\{1,\ldots,m\},S\cup\{\mathrm{LM}(f)\}}\,\vec{f} = \begin{matrix} 1 \\ \vdots \\ m \end{matrix} \begin{pmatrix} \overset{\cdots}{\cdots} & \overset{s\in S}{[s]} & \overset{\cdots}{\cdots} & \overset{\mathrm{LM}(f)}{[\mathrm{LM}(f)]} \\ & \vdots & & \vdots \\ \cdots & [m\ s] & \cdots & [m\ \mathrm{LM}(f)] \end{pmatrix} \begin{pmatrix} \vdots \\ \alpha_s \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Let m^+ be the successor of m, $m^+\ \mathrm{LM}(f)$ is the fail of f *if*

$$H_{\{1,\ldots,m,m^+\},S\cup\{\mathrm{LM}(f)\}}\,\vec{f} = \begin{matrix} 1 \\ \vdots \\ m \\ m^+ \end{matrix} \begin{pmatrix} \overset{\cdots}{\cdots} & \overset{s\in S}{[s]} & \overset{\cdots}{\cdots} & \overset{\mathrm{LM}(f)}{[\mathrm{LM}(f)]} \\ & \vdots & & \vdots \\ \cdots & [m\ s] & \cdots & [m\ \mathrm{LM}(f)] \\ \cdots & [m^+\ s] & \cdots & [m^+\ \mathrm{LM}(f)] \end{pmatrix} \begin{pmatrix} \vdots \\ \alpha_s \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ e \end{pmatrix},$$

with $e \neq 0$.

We can also write another proof of Proposition 5 with a matrix viewpoint.

Proof of Proposition 5. Let $f_1 = \mathrm{LM}(f_1) + \sum_{s\in S} \alpha_s\, s$ and $f_2 = \mathrm{LM}(f_2) + \sum_{s\in S'} \beta_s\, s$ be monic. Let v^- be the predecessor of v. Let $\tilde{S} = S \cup S' \setminus \{\mathrm{LM}(f_2), \mathrm{LM}(f_1)\}$, assuming

$\text{LM}(f_2) \neq \text{LM}(f_1)$, then we have

$$H_{\{1,\ldots,v^-,v\},\tilde{S}\cup\{\text{LM}(f_2),\text{LM}(f_1)\}}\,(\vec{f_1} + c\,\vec{f_2}) = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ e_1 + c\,e_2 \end{pmatrix}$$

$$\begin{array}{c} \\ 1 \\ \vdots \\ v^- \\ v \end{array} \begin{array}{c} \begin{array}{cccccc} & \cdots & s\in\tilde{S} & \cdots & \text{LM}(f_2) & \text{LM}(f_1) \end{array} \\ \begin{pmatrix} \cdots & [s] & \cdots & [\text{LM}(f_2)] & [\text{LM}(f_1)] \\ & \vdots & & \vdots & \vdots \\ \cdots & [v^-\,s] & \cdots & [v^-\,\text{LM}(f_2)] & [v^-\,\text{LM}(f_1)] \\ \cdots & [v\,s] & \cdots & [v\,\text{LM}(f_2)] & [v\,\text{LM}(f_1)] \end{pmatrix} \end{array} \begin{pmatrix} \vdots \\ \alpha_s + c\,\beta_s \\ \vdots \\ c \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ e_1 + c\,e_2 \end{pmatrix}.$$

It is now clear that vector $\vec{f_1} - \frac{e_1}{e_2}\,\vec{f_2}$ is in the kernel of this matrix. That is, polynomial $f_1 - \frac{e_1}{e_2}\,f_2$ has a shift v. □

Changing every evaluation into a matrix-vector product in the BMS algorithm yields the following presentation of the BMS algorithm, namely Algorithm 2.

Algorithm 2: Linear Algebra variant of the BMS algorithm.

Input: A table $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ with coefficients in $\mathbb{K}$, a monomial ordering $\prec$ and a monomial M as the stopping condition.

Output: A set G of relations generating I_M.

$T := \{m \in \mathbb{K}[\mathbf{x}], m \preceq M\}$. `// ordered for` $\prec$

$G := \{1\}$. `// the future Gröbner basis`

$S := \emptyset$. `// staircase edge, elements will be` $[h, \mathrm{fail}(h)/\,\mathrm{LM}(h)]$

For all $m \in T$ **do**

  $S' := S$.

  For $g \in G$ **do**

    If $\mathrm{LM}(g) | m$ **then**

      $e := H_{\left\{\frac{m}{\mathrm{LM}(g)}\right\},\mathrm{supp}(g)}\,\vec{g}$.

      If $e \neq 0$ **then**

        $S' := S' \cup \left\{\left[\frac{g}{e}, \frac{m}{\mathrm{LM}(g)}\right]\right\}$.

  $S' := \min_{\mathrm{fail}(h)\in S'} \{[h, \mathrm{fail}(h)/\,\mathrm{LM}(h)]\}$. `// see Remark 8`

  $G' := \mathrm{Border}(S')$.

  For $g' \in G'$ **do**

    Let $g \in G$ such that $\mathrm{LM}(g) | \mathrm{LM}(g')$.

    If $\mathrm{LM}(g) \nmid m$ **then**

      $g' := \frac{\mathrm{LM}(g')}{\mathrm{LM}(g)}\, g$. `// shifts the relation`

    Else if $\exists\, h \in S, \frac{m}{\mathrm{LM}(g')} | \,\mathrm{fail}(h)$ **then**

      $g' := \frac{\mathrm{LM}(g')}{\mathrm{LM}(g)}\, g - \left(H_{\left\{\frac{m}{\mathrm{LM}(g)}\right\},\mathrm{supp}(g)}\,\vec{g}\right) \frac{\mathrm{LM}(g')\ \mathrm{fail}(h)}{m}\, h$. `// see Proposition 5`

    Else $g' := g$.

  $G := G'$

  $S := S'$

Return G.

3. An Adaptive version of the BMS algorithm

The BMS algorithm was presented first in Sakata (1988) for the dimension 2 case and then was extended to dimension n in Sakata (1990, 2009). The BMS algorithm is an iterative algorithm, visiting each term $u_{\mathbf{i}} = [\mathbf{x}^{\mathbf{i}}]$ of the input sequence in increasing order for the input monomial order. At each step, it has a truncated Gröbner basis of the ideal of relations and test them in the visited monomial. If some of them fail, the algorithm updates the Gröbner basis with new valid relations.

When a relation g fails at monomial m, two situations arise: either $\frac{m}{\mathrm{LM}(g)}$ was already in the staircase and then a new relation g' with $\mathrm{LM}(g') = \mathrm{LM}(g)$ is computed or it was not and both $\mathrm{LM}(g)$ and $\frac{m}{\mathrm{LM}(g)}$ are added to the staircase. New relations are then computed depending on the possible new leading monomials.

The problem is now to understand when the Gröbner basis of the ideal of relations has actually been computed. Assuming the sequence is linear recurrent, Proposition 9 provides an answer to this question.

Remark 10. *Although this bound is sharp generically, in some favourable cases, one can guess the right relations early on. In Example 4, for $p = 1$ and $q = 2$, the right staircase is found at step y. In fact, the right Gröbner basis is already guessed as well, while Proposition 9 only ensures that it will be correctly guessed at step y^3.*

It could therefore be very fruitful to have an heuristic helping us determining if the current Gröbner basis is the right one when the size of the staircase is known in advance. Indeed, it could allow us to end earlier the running of the BMS algorithm. Unfortunately, it is not rare that an interrupted BMS algorithm does not return the correct Gröbner basis, in fact such an interrupted BMS algorithm will never return the right Gröbner basis for any of the four families of sequences used in Section 6. The goal is thus to reduce the number of testings differently.

Let us recall that at step m, whenever a relation g such that $\mathrm{LM}(g)|m$ fails, if $\frac{m}{\mathrm{LM}(g)}$ is not in the staircase, then the algorithm adds both $\mathrm{LM}(g)$ and $\frac{m}{\mathrm{LM}(g)}$ in the new staircase. Assuming we know in advance the size of the staircase of the output Gröbner basis, during the execution of the algorithm, we can detect that testing the relation g in m is useless if the staircase becomes too big after adding the two monomials.

Let us show in the following example how we can take advantage of this strategy.

Example 6. *Let us resume Example 5 with the assumption that the staircase has a size at most* 5 *and that we want to run the algorithm up to monomial x^5.*

We start with the non empty staircase $S = \{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$ *and the relations* $G = \{x\,y - y - 1, y^3, x^3 - 2\,x^2 + x\}$. *This means that on the one hand the relations in G have been tested up to all their multiples less than* y^5 *while relation* y^2 *(resp.* $x^2 - 2\,x + 1$*) in S fails when multiplied by* x^2 *(resp.* y^2*) but does not fail when multiplied by a lesser monomial.*

For the monomial y^5

Nothing must be done for the relation $g_1 = x\,y - y - 1$.
The relation $g_2 = y^3$ *succeeds since* $[\mathbf{b}_{0,5}] = 0$.
Nothing must be done for the relation $g_3 = x^3 - 2\,x^2 + x$.

For the monomial $x\,y^4$

Should the relation $g_1 = x\,y - y - 1$ *fail in* $x\,y^4$*, we would have to add* $x\,y$ *and* y^3 *in the staircase, raising its size to* 7. *We skip testing* g_1.
Should the relation $g_2 = y^3$ *fail in* $x\,y^4$*, we would have to add* y^3 *and* $x\,y$ *in the staircase, raising its size to* 7. *We skip testing* g_2.
Nothing must be done for the relation $g_3 = x^3 - 2\,x^2 + x$.

For the monomial $x^2\,y^3$

Should the relation $g_1 = x\,y - y - 1$ *fail in* $x^2\,y^3$*, we would have to add* $x\,y$ *and* $x\,y^2$ *in the staircase, raising its size to* 7. *We skip testing* g_1.
The relation $g_2 = y^3$ *succeeds since* $[\mathbf{b}_{2,3}] = 0$.
Nothing must be done for the relation $g_3 = x^3 - 2\,x^2 + x$.

For the monomial $x^3\,y^2$

Should the relation $g_1 = x\,y - y - 1$ *fail in* $x^3\,y^2$*, we would have to add* $x\,y$ *and* $x^2\,y$ *in the staircase, raising its size to* 7. *We skip testing* g_1.
Nothing must be done for the relation $g_2 = y^3$.
The relation $g_3 = x^3 - 2\,x^2 + x$ *fails since* $[\mathbf{b}_{3,2} - 2\,\mathbf{b}_{2,2} + \mathbf{b}_{1,2}] = 1$. *Thus* $S' = \{[y^2, x^2], [x^2 - 2\,x + 1, y^2], [x^3 - 2\,x + 1, y^2]\}$.
S' *is set to* $\{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$ *and* $G' = \{y^3, x\,y, x^3\}$.
We set $g'_1 = x\,y - y - 1$ *and* $g'_2 = y^3$.
For the relation $g'_3 = x^3$, $x^3 | x^3\,y^2$ *and* $\frac{x^3\,y^2}{x^3} | \,\mathrm{fail}(x^2 - 2\,x + 1)$, *hence* $g'_3 = x^3 - 3\,x^2 + 3\,x - 1$.
We update $G := G' = \{y^3, x\,y - y - 1, x^3 - 3\,x^2 + 3\,x - 1\}$ *and* $S := S' = \{[y^2, x^2], [x^2 - 2\,x + 1, y^2]\}$.

For the monomial $x^4\,y$

Should the relation $g_1 = x\,y - y - 1$ *fail in* $x^4\,y$*, we would have to add* $x\,y$ *and* x^3 *in the staircase, raising its size to* 7. *We skip testing* g_1.
Nothing must be done for the relation $g_2 = y^3$.
Should the relation $g_3 = x^3 - 3\,x^2 + 3\,x - 1$ *fail in* $x^4\,y$*, we would have to add* x^3 *and* $x\,y$ *in the staircase, raising its size to* 7. *We skip testing* g_3.

For the monomial x^5

Nothing must be done for the relation $g_1 = x\,y - y - 1$.
Nothing must be done for the relation $g_2 = y^3$.
The relation $g_3 = x^3 - 3\,x^2 + 3\,x - 1$ *succeeds since* $[\mathbf{b}_{5,0} - 3\,\mathbf{b}_{4,0} + 3\,\mathbf{b}_{3,0} - \mathbf{b}_{2,0}] = 0$.

The algorithm returns relations $x\,y - y - 1, y^3, x^3 - 3\,x^2 + 3\,x - 1$*, the first one with a shift* x^3 *and the other two with a shift* x^2.

In this example, skipping some relation testings allowed us to skip all the testings in a loop, namely loops $x\,y^4$ and $x^4\,y$. As a byproduct, we also reduced the number of table queries.

Integrating this strategy in the BMS algorithm yields an adaptive variant, Algorithm 3, reducing the number of relation testings and table queries. This algorithm uses the Stabilize(S) procedure, returning the set of all the divisors of elements of S, i.e. the whole smallest staircase containing S.

This version was motivated by a remark in Sakata (2009) where the author announced that in applications where an approximate size of the staircase is known, one can stop early the execution of the BMS algorithm. Yet, we do not know if such a strategy is classical and if it is exactly the one described in Algorithm 3.

Predicting how many monomials will be completely skipped in order to reduce the number of table queries can be a hard task. Indeed, it is clear that if relation g can be skipped at monomial m, it will also be skipped at any multiple of m. Yet, even if m is completely skipped, a relation that cannot be tested in m might need to be tested in $m\,x_i$ for some i.

Therefore, even if m is completely skipped, $m\,x_i$ might not be. We illustrate this phenomon with the following example.

Example 7. *Let* $\mathbf{u} = (u_{i,j})_{(i,j)\in\mathbb{N}^2}$ *be the sequence defined by* $u_{4,1} = 1$ *and* $u_{i,j} = 0$ *if* $(i, j) \neq (4, 1)$. *Running the* BMS *algorithm on these arguments yields relations* y^2, x^5 *so that the staircase has size* 10. *We assume though that the only known upper bound on the staircase size is* 14.

We give a short trace of the algorithm called on $\mathbf{u}$ *for the* DRL($y \prec x$) *ordering up to monomial* x^9. *Therefore, we also input* 14 *as the upper bound on the size of the output staircase to the* Adaptive BMS *algorithm.*

For all the monomials from 1 *to* $x^3\,y^2$

The relation $g_1 = 1$ *succeeds.*

For the monomial $x^4\,y$

The relation $g_1 = 1$ *fails since* $[u_{4,1}] = 1$. *Thus* $S' = \{[1, x^4\,y]\}$.
S' *is set to* $\{[1, x^4\,y]\}$ *and* $G' = \{y^2, x^5\}$.
We set $g'_1 = y^2$ *and* $g'_2 = x^5$.

Algorithm 3: Adaptive BMS (Linear Algebra variant).

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

Output: A set G of relations generating I_M.

$T := \{m \in \mathbb{K}[\mathbf{x}], m \preceq M\}$.
$G := \{1\}$.
$S := \emptyset$.
For all $m \in T$ **do**
  $S' := S$.
  For $g \in G$ **do**
    If $\mathrm{LM}(g) | m$ **then**
      If $\frac{m}{\mathrm{LM}(g)} \notin \mathrm{Stabilize}(S)$ **and** $\#\,\mathrm{Stabilize}\left(S \cup \left\{\mathrm{LM}(g), \frac{m}{\mathrm{LM}(g)}\right\}\right) > d$ **then**
        next. `// skip this relation testing`
      $e := \left[\frac{m}{\mathrm{LM}(g)}\, g\right]_{\mathbf{u}}$
      If $e \neq 0$ **then**
        $S' := S' \cup \left\{\left[\frac{g}{e}, \frac{m}{\mathrm{LM}(g)}\right]\right\}$.
  $S' := \min_{\mathrm{fail}(h)\in S'} \{[h, \mathrm{fail}(h)/\,\mathrm{LM}(h)]\}$.
  $G' := \mathrm{Border}(S')$.
  For $g' \in G'$ **do**
    Let $g \in G$ such that $\mathrm{LM}(g) | \,\mathrm{LM}(g')$.
    If $\mathrm{LM}(g) \nmid m$ **then**
      $g' := \frac{\mathrm{LM}(g')}{\mathrm{LM}(g)}\, g$.
    Else if $\exists\, h \in S, \frac{m}{\mathrm{LM}(g')} | \,\mathrm{fail}(h)$ **then**
      $g' := \frac{\mathrm{LM}(g')}{\mathrm{LM}(g)}\, g - \left[\frac{m}{\mathrm{LM}(h)}\, h\right]_{\mathbf{u}} \frac{\mathrm{LM}(g')\ \mathrm{fail}(h)}{m}\, h$.
    Else $g' := g$.
  $G := G'$.
  $S := S'$.
Return G.

For the relation $g_1' = y^2$, $y^2 \nmid x^4\,y$ *thus* $g_1' = y^2$.
For the relation $g_2' = x^5$, $x^5 \nmid x^4\,y$ *thus* $g_2' = x^5$.
We update $G := G' = \{y^2, x^5\}$ *and* $S := S' = \{[1, x^4\,y]\}$.

For the monomial x^5

The relation $g_2 = x^5$ *succeeds.*

For the monomial y^6

The relation $g_1 = y^2$ *succeeds.*

For the monomial $x\,y^5$

The relation $g_1 = y^2$ *succeeds.*

For the monomial $x^2\,y^4$

The relation $g_1 = y^2$ *succeeds.*

For the monomial $x^3\,y^3$

The relation $g_1 = y^2$ *succeeds.*

For the monomial $x^4\,y^2$

The relation $g_1 = y^2$ *succeeds.*

For the monomial $x^5\,y$

The relation $g_2 = x^5$ *succeeds.*

For the monomial x^6

The relation $g_2 = x^5$ *succeeds.*

For the monomial y^7

The relation $g_1 = y^2$ *succeeds.*

For the monomial $x\,y^6$

Should the relation $g_1 = y^2$ *fail, we would have to add* y^2 *and* $x\,y^4$ *to the staircase, raising its size to* 16. *We skip testing* g_1.

For the monomial $x^2\,y^5$

Should the relation $g_1 = y^2$ *fail, we would have to add* y^2 *and* $x^2\,y^3$ *to the staircase, raising its size to* 16. *We skip testing* g_1.

For the monomial $x^3\,y^4$

The relation $g_1 = y^2$ *succeeds.*

For the monomial $x^4\,y^3$

The relation $g_1 = y^2$ *succeeds.*

For the monomial $x^5\,y^2$

The relation $g_1 = y^2$ *succeeds.*
The relation $g_2 = x^5$ *succeeds.*

For the monomial $x^6\,y$

The relation $g_2 = x^5$ *succeeds.*

For the monomial x^7

The relation $g_2 = x^5$ *succeeds.*

For the monomial y^8

Should the relation $g_1 = y^2$ *fail, we would have to add* y^2 *and* y^6 *to the staircase, raising its size to* 16*. We skip testing* g_1*.*

For the monomial $x\,y^7$

We did not test g_1 *in* $x\,y^6$*. We skip testing* g_1*.*

For the monomial $x^2\,y^6$

We did not test g_1 *in* $x\,y^6$ *and* $x^2\,y^5$*. We skip testing* g_1*.*

For the monomial $x^3\,y^5$

We did not test g_1 *in* $x^2\,y^5$*. We skip testing* g_1*.*

For the monomial $x^4\,y^4$

Should the relation $g_1 = y^2$ *fail, we would have to add* y^2 *and* $x^4\,y^2$ *to the staircase, raising its size to* 15*. We skip testing* g_1*.*

For the monomial $x^5\,y^3$

The relation $g_1 = y^2$ *succeeds.*

The relation $g_2 = x^5$ *succeeds.*

For the monomial $x^6\,y^2$

The relation $g_1 = y^2$ *succeeds.*

The relation $g_2 = x^5$ *succeeds.*

For the monomial $x^7\,y$

The relation $g_2 = x^5$ *succeeds.*

For the monomial x^8

The relation $g_2 = x^5$ *succeeds.*

For the monomial y^9

We did not test g_1 *in* y^8*. We skip testing* g_1*.*

For the monomial $x\,y^8$

We did not test g_1 *in* y^8 *and* $x\,y^7$*. We skip testing* g_1*.*

For the monomial $x^2\,y^7$

We did not test g_1 *in* $x\,y^7$ *and* $x^2\,y^6$*. We skip testing* g_1*.*

For the monomial $x^3\,y^6$

We did not test g_1 *in* $x^2\,y^6$ *and* $x^3\,y^5$*. We skip testing* g_1*.*

For the monomial $x^4\,y^5$

We did not test g_1 *in* $x^3\,y^5$ *and* $x^4\,y^4$*. We skip testing* g_1*.*

For the monomial $x^5\,y^4$

We did not test g_1 *in* $x^4\,y^4$*. We skip testing* g_1*.*

The relation $g_2 = x^5$ *succeeds.*

For the monomial $x^6\,y^3$

Should the relation $g_1 = y^2$ fail, we would have to add y^2 and $x^6\,y$ to the staircase, raising its size to 15. *We skip testing g_1.*

Should the relation $g_2 = x^5$ fail, we would have to add x^5 and $x\,y^3$ to the staircase, raising its size to 15. *We skip testing g_2.*

For the monomial $x^7\,y^2$

The relation $g_1 = y^2$ succeeds.

The relation $g_2 = x^5$ succeeds.

For the monomial $x^8\,y$

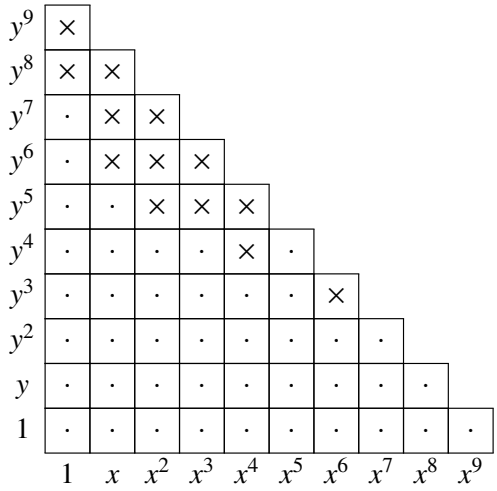
The relation $g_2 = x^5$ succeeds.

For the monomial x^9

The relation $g_2 = x^5$ succeeds.

The algorithm returns relations y^2, x^5, the first one with a shift x^7 and the other one with a shift x^4.

The following figure shows the visited monomials where at least one relation was tested ($\cdot$) and those completely skipped ($\times$).



Although the monomial $x^4\,y^4$ was completely skipped, $x^5\,y^4$ is not thanks to the relation x^5 that must be tested.

Remark 11. *If d is too big, then the* Adaptive BMS *algorithm will behave exactly like the* BMS *algorithm. For instance, if d is at least the number of monomials less or equal to the stopping monomial, then no sequence term can be skipped.*

Yet, even though the bound in Proposition 9 is sharp, running the Adaptive BMS *algorithm with $d = \#S$ on a sequence $\mathbf{u}$ up to $s_{\max} \cdot \max(g_{\max}, s_{\max})$ may allow us to skip some sequence terms.*

In some applications, like the Sparse-FGLM *one, the size of the staircase is already known and thus can be used as an upper bound on d, Faugère and Mou (2011, 2017). In the correcting codes application, the choice for d can be a sensible bound on the number of errors.*

Let us also note that guessing d beforehand can be more difficult for the Adaptive BMS *algorithm than for the* Adaptive Scalar-FGLM *algorithm. Indeed, the efficiency on the adaptive behaviour of the* Adaptive BMS *algorithm depends on the sharpness of d, while, in the* Adaptive Scalar-FGLM *algorithm, d merely serves as an early termination condition, see also Algorithm 4.*

4. The Adaptive version of the Scalar-FGLM algorithm

While the BMS and Adaptive BMS algorithms are iterative algorithms, the Scalar-FGLM algorithm is global, see Berthomieu et al. (2015, 2017) and (Berthomieu and Faugère, 2017, Section 4). It finds the Gröbner basis of the ideal of relations by computing the column rank profile of a big multi-Hankel matrix indexed by a set of monomials T. In practice, this set T must contain all the monomials less than the monomials in the Gröbner basis of relations.

To circumvent the inherent complexity of computing the rank profile of a big multi-Hankel matrix, the authors proposed an adaptive algorithm behaving more closely to the FGLM algorithm, see Faugère et al. (1993).

The goal is to iterate on a monomial t and compute, for a set S such that $H_{S,S}$ is full rank, if $H_{S\cup\{t\},S\cup\{t\}}$ is also full rank. If it is, then t is added to S, otherwise a relation with support in $S \cup \{t\}$ has been found. No further relation with leading term a multiple of t will be computed. When a given lower-bound on the size of the staircase is reached, the algorithm stops and computes the remaining relations from the leading terms lying on the border of the staircase.

This yields the Adaptive Scalar-FGLM algorithm: Algorithm 4.

Example 8. *We give the trace of the algorithm on the sequence* $\mathbf{u} = (2^i 3^j (i+1))_{(i,j)\in\mathbb{N}^2}$ *with the* drl$(y \prec x)$ *ordering with a lower bound* 2 *on the staircase size.*

We set $L = \{1\}$, $S = \emptyset$, $G' = \emptyset$.
We set $t = 1$ *and build the matrix* $H_{S\cup\{1\},S\cup\{1\}} = \begin{pmatrix} 1 \end{pmatrix}$ *that is full rank. Hence* $S = \{1\}$ *and* $L = \{y, x\}$.
We set $t = y$ *and build the matrix* $H_{S\cup\{y\},S\cup\{y\}} = \begin{pmatrix} 1 & 3 \\ 3 & 9 \end{pmatrix}$ *that is not full rank. Solving* $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{y\}} = 0$ *yields relation* $y - 3$, *so* $G = \{y-3\}$, $G' = \{y\}$ *and* L *is updated to* $\{x\}$.
We set $t = x$ *and build the matrix* $H_{S\cup\{x\},S\cup\{x\}} = \begin{pmatrix} 1 & 4 \\ 4 & 12 \end{pmatrix}$ *that is full rank. Hence* $S = \{1, x\}$ *and* $L = \{x^2\}$.

Algorithm 4: Adaptive Scalar-FGLM (simple version).

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

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

$L := \{1\}$. `// set of next terms to study`
$S := \emptyset$. `// the useful staircase with respect to` $\prec$
$G := \emptyset, G' := \emptyset$.
While $L \neq \emptyset$ **do**
  $t := \min_\prec(L)$.
  If $H_{S\cup\{t\},S\cup\{t\}}$ *is full rank* **then** `// as in Section 2.3`
    $S := S \cup \{t\}$ and $L := L \cup \{x_i\, t, i = 1, \ldots, n\} \setminus \{t\}$.
    Remove multiples of elements of G' in L.
    If $\#S \geq d$ **then** `// early termination`
      While $L \neq \emptyset$ **do**
        $t' := \min_\prec(L)$.
        Find $\boldsymbol{\alpha}$ such that $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{t'\}} = 0$.
        $G := G \cup \{t' + \sum_{s\in S} \alpha_s\, s\}$.
        Remove multiples of elements of t' in L.
      Return G.
  Else
    Find $\boldsymbol{\alpha}$ such that $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{t\}} = 0$.
    $G' := G' \cup \{t\}$.
    $G := G \cup \{t + \sum_{s\in S} \alpha_s\, s\}$.
    Remove multiples of t in L and sort L by increasing order.
Error "Run Scalar-FGLM".

Now $\#S$ is greater or equal to the bound 2. Solving $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{x^2\}} = 0$ yields relation $x^2 - 4\,x + 4$, so $G = \{y - 3, x^2 - 4\,x + 4\}$ and L is updated to $\emptyset$. Furthermore, the relation $y - 3$ has been tested with a shift $\{1, y\}$ while the relation $x^2 - 4\,x + 4$ has been tested with a shift $\{1, x\}$.

Remark 12. *If no lower bound on the size of S were given, then an infinite loop might occur on a non linear recurrent sequence. For instance, on the factorial sequence $(i!)_{i\in\mathbb{N}}$, all the monomials x^i would be found in the staircase.*

If we know the sequence is linear recurrent, then we can remove this bound. In that case, the last step of Example 8 becomes:

We set $t = x^2$ and build the matrix $H_{S\cup\{x^2\},S\cup\{x^2\}} = \begin{pmatrix} 1 & 4 & 12 \\ 4 & 12 & 32 \\ 12 & 32 & 80 \end{pmatrix}$ that is not full rank. Solving $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{x^2\}} = 0$ yields relation $x^2 - 4\,x + 4$, so $G = \{y - 3, x^2 - 4\,x + 4\}, G' = \{y, x^2\}$ and L is updated to $\emptyset$.

Furthermore, the relation $y - 3$ has been tested with shift $\{1, y\}$ while the relation $x^2 - 4\,x + 4$ has been tested with a shift $\{1, x, x^2\}$.

For a generic sequence, the algorithm computes the ideal of relations of the sequence. However, it is easy to make a sequence such that the algorithm fails. It suffices to have a sequence whose staircase S has a subset S' such that the matrix $H_{S',S'}$ has a rank defect.

This motivated the authors to extend the algorithm to bypass this issue in Berthomieu et al. (2017).

We give an example of what can happen when the wrong relations are computed and describe their shifts.

Example 9. *We consider the ideal $I = \langle y^2 - y, x^2\,y - x\,y, x^4 - 6\,x^3 + 11\,x^2 - 6\,x\rangle \subseteq \mathbb{F}_{11}[x, y]$ and a sequence $\mathbf{u} = (u_{i,j})_{(i,j)\in\mathbb{N}^2}$ over $\mathbb{F}_{11}$ made from this ideal and some initial conditions. The first terms of the sequence are $\begin{pmatrix} 1 & 2 & 2 & 2 & 2 & \cdots \\ 3 & 4 & 4 & 4 & 4 & \cdots \\ 3 & 4 & 4 & 4 & 4 & \cdots \\ -1 & 4 & 4 & 4 & 4 & \cdots \\ 1 & 4 & 4 & 4 & 4 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}$. We also call the algorithm on the* $\textsc{drl}(y \prec x)$ *ordering.*

We set $L = \{1\}$, $S = \emptyset$, $G' = \emptyset$.
We set $t = 1$ and build the matrix $H_{S\cup\{1\},S\cup\{1\}} = (\,1\,)$ that is full rank. Hence $S = \{1\}$ and $L = \{y, x\}$.
We set $t = y$ and build the matrix $H_{S\cup\{y\},S\cup\{y\}} = \begin{pmatrix} 1 & 2 \\ 2 & 2 \end{pmatrix}$ that is full rank. Hence $S = \{1, y\}$ and $L = \{x, y^2, x\,y\}$.
We set $t = x$ and build the matrix $H_{S\cup\{x\},S\cup\{x\}} = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 2 & 4 \\ 3 & 4 & 3 \end{pmatrix}$ that is full rank. Hence $S = \{1, y, x\}$ and $L = \{y^2, x\,y, x^2\}$.

We set $t = y^2$ *and build the matrix* $H_{S\cup\{y^2\},S\cup\{y^2\}} = \begin{pmatrix} 1 & 2 & 3 & 2 \\ 2 & 2 & 4 & 2 \\ 3 & 4 & 3 & 4 \\ 2 & 2 & 4 & 2 \end{pmatrix}$ *that is not full rank. Solving* $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{y^2\}} = 0$ *yields relation* $y^2 - y$ *with a shift* $\{1, y, x, y^2\}$*, so* $G = \{y^2 - y\}, G' = \{y^2\}$ *and* L *is updated to* $\{x\,y, x^2\}$*.*

We set $t = x\,y$ *and build the matrix* $H_{S\cup\{x\,y\},S\cup\{x\,y\}} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 2 & 4 & 4 \\ 3 & 4 & 3 & 4 \\ 4 & 4 & 4 & 4 \end{pmatrix}$ *that is not full rank. Solving* $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{x\,y\}} = 0$ *yields relation* $x\,y - x - y + 1$ *with a shift* $\{1, y, x, x\,y\}$*, so* $G = \{y^2 - y, x\,y - x - y + 1\}, G' = \{y^2, x\,y\}$ *and* L *is updated to* $\{x^2\}$*.*

We set $t = x^2$ *and build the matrix* $H_{S\cup\{x^2\},S\cup\{x^2\}} = \begin{pmatrix} 1 & 2 & 3 & 3 \\ 2 & 2 & 4 & 4 \\ 3 & 4 & 3 & -1 \\ 3 & 4 & -1 & 1 \end{pmatrix}$ *that is full rank. Hence* $S = \{1, y, x, x^2\}$ *and* $L = \{x^3\}$*.*

We set $t = x^3$ *and build the matrix* $H_{S\cup\{x^3\},S\cup\{x^3\}} = \begin{pmatrix} 1 & 2 & 3 & 3 & -1 \\ 2 & 2 & 4 & 4 & 4 \\ 3 & 4 & 3 & -1 & 1 \\ 3 & 4 & -1 & 1 & 2 \\ -1 & 4 & 1 & 2 & 6 \end{pmatrix}$ *that is not full rank. Solving* $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{x^3\}} = 0$ *yields relation* $g_3 = x^3 + 3\,x^2 + 10\,x + y + 4$ *with a shift* $\{1, y, x, x^2\}$*, so* $G = \{y^2 - y, x\,y - x - y + 1, x^3 + 3\,x^2 + 10\,x + y + 4\}, G' = \{y^2, x\,y, x^3\}$ *and* L *is updated to* $\emptyset$*.*

We can notice that

- *the first relation,* $y^2 - y$ *is really a relation of* $\mathbf{u}$ *but has only, a priori, a shift* $\{1, y, x, y^2\}$*, i.e. its shift is* y^2*.*
- *the second relation,* $x\,y - x - y + 1$*, is not a real relation of* $\mathbf{u}$ *and is known to have a shift* $\{1, y, x, x\,y\}$*. Actually we can check that* $[y^2\,(x\,y - x - y + 1)] = 0$ *and* $[x^2\,(x\,y - x - y + 1)] = 4$*, so that the relation has a shift* $\{1, y, x, y^2, x\,y\}$*, i.e. its shift is* $x\,y$ *and its fail is* $x^3\,y$*.*
- *the third relation,* $x^3 + 3\,x^2 + 10\,x + y + 4$*, is not a true relation of* $\mathbf{u}$ *and is known to have a shift* $\{1, y, x, x^2, x^3\}$*. Actually we can check that* $[y^2\,(x\,y - x - y + 1)] = 0$ *and* $[x\,y\,(x^3 + 3\,x^2 + 10\,x + y + 4)] = -1$*, i.e. its shift is* y^2 *and its fail is* $x^4\,y$*.*

All in all, we computed the relation $x^3 + 3\,x^2 + 10\,x + y + 4$ *assuming it should be valid when multiplied by* x^2 *or* x^3*, while it cannot be valid when multiplied by* $x\,y \prec x^2 \prec x^3$*.*

5. Analogies and differences of the adaptive variants

We now compare theoretically the Adaptive BMS and the Adaptive Scalar-FGLM algorithms. As the Adaptive BMS algorithm differs from the BMS algorithm just in the execution: mainly some testings are skipped, results from (Berthomieu

and Faugère, 2017, Section 6) are still valid for the Adaptive BMS algorithm. On the other hand, the Adaptive Scalar-FGLM algorithm does not necessarily provide the same output as the Scalar-FGLM algorithm.

5.1. Closed staircase

In (Berthomieu and Faugère, 2017, Section 5.1, Theorem 7), we show that the BMS algorithm always returns a zero-dimensional ideal while the Scalar-FGLM algorithm can return a zero-dimensional or a positive-dimensional ideal. This is in fact one of the main differences between these two algorithms.

In the following theorem, we prove that the Adaptive BMS algorithm and the Adaptive Scalar-FGLM algorithm are closer on that matter assuming one knows the size of the output staircase in advance.

Theorem 13. *Let* $\mathbf{u}$ *be a sequence,* $\prec$ *be a monomial ordering and* d *be the size of the staircase.*

Calling the Adaptive BMS *algorithm on* $\mathbf{u}$*,* $\prec$*,* d *and a stopping monomial* M *yields a truncated Gröbner basis of a zero-dimensional ideal.*

Calling the Adaptive Scalar-FGLM *algorithms on* $\mathbf{u}$*,* $\prec$ *and* d *yields a truncated Gröbner basis of a zero-dimensional ideal.*

Proof. The first part of the result comes directly from the line $G' := \text{Border}(S')$ in the description of the Adaptive BMS algorithm, Algorithm 3.

The second part of the result comes from the fact that the leading terms of the relations are lying in the border of the staircase and are minimal for both $\prec$ and $|$. Thus, for any variable x_i, there always exists a relation with leading term a pure power of x_i. □

It is possible to change this early termination procedure so that the Adaptive Scalar-FGLM algorithm is closer to the Scalar-FGLM algorithm, yielding a potential positive-dimensional algorithm. If we still want to try to close as much as possible the staircase with degenerate square matrices, it suffices to check that the relation $t' + \sum_{s \in S} \alpha_s \, s$ is valid with a shift $S \cup \{t'\}$. This yields Algorithm 5.

5.2. Reduction of relations

The Adaptive Scalar-FGLM algorithm computes a staircase and then relations with support in the staircase except their leading terms that lie on the border. On the other hand, although the Adaptive BMS algorithm may compute the same ideal of relations as the Adaptive Scalar-FGLM algorithm, their Gröbner basis can be different.

Algorithm 5: Tweaked Adaptive Scalar-FGLM.

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

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

$L := \{1\}$. `// set of next terms to study`
$S := \emptyset$. `// the useful staircase with respect to` $\prec$
$G := \emptyset, G' := \emptyset$.
While $L \neq \emptyset$ **do**
  $t := \min_\prec(L)$.
  If $H_{S\cup\{t\},S\cup\{t\}}$ *is full rank* **then** `// as in Section 2.3`
    $S := S \cup \{t\}$ and $L := L \cup \{x_i\, t, i = 1, \ldots, n\} \setminus \{t\}$.
    Remove multiples of elements of G' in L.
    If $\#S \geq d$ **then** `// early termination`
      While $L \neq \emptyset$ **do**
        $t' := \min_\prec(L)$.
        Find $\boldsymbol{\alpha}$ such that $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{t'\}} = 0$.
        If $H_{\{t'\},S}\,\boldsymbol{\alpha} + H_{\{t'\},\{t'\}} = 0$ **then**
          $G := G \cup \{t' + \sum_{s\in S} \alpha_s\, s\}$.
        Remove multiples of elements of t' in L.
      Return G.
  Else
    Find $\boldsymbol{\alpha}$ such that $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{t\}} = 0$.
    $G' := G' \cup \{t\}$.
    $G := G \cup \{t + \sum_{s\in S} \alpha_s\, s\}$.
    Remove multiples of t in L and sort L by increasing order.
Error "Run Scalar-FGLM".

Theorem 14. *Let* $\mathbf{u}$ *be a sequence,* $\prec$ *be a monomial ordering and* d *be the size of the staircase.*

Calling the Adaptive Scalar-FGLM *algorithms on* $\mathbf{u}$*,* $\prec$*, and* d *yields a truncated reduced Gröbner basis of an ideal.*

Calling the Adaptive BMS *algorithm on* $\mathbf{u}$*,* $\prec$*,* d *and a stopping monomial* M *yields a truncated minimal Gröbner basis of an ideal, which is not necessarily reduced.*

Furthermore, even if $\mathbf{u}$ *is linear recurrent and the* Adaptive Scalar-FGLM *algorithm computes the ideal of relations of* $\mathbf{u}$*, then there is no reason for the output of the* Adaptive BMS *algorithm to be reduced.*

Proof. For two distinct polynomials g, g' in the Gröbner basis returned by Adaptive Scalar-FGLM algorithm, $\text{LT}(g)$ does not divide any monomial in the support of g'. Hence the Gröbner basis is reduced.

For two distinct polynomials g, g' in the Gröbner basis returned by Adaptive BMS algorithm, $\text{LT}(g)$ does not divide $\text{LT}(g')$. Hence the Gröbner basis is minimal. However, there is no reason for $\text{LT}(g)$ not to divide any monomial in the support of g'. □

Example 10. *We let* $\mathbf{u} = \left(i^2 + j^2 - 1\right)_{(i,j)\in\mathbb{N}^2}$ *be a sequence and consider the* $\text{DRL}(y \prec x)$ *ordering. The ideal of relations of* $\mathbf{u}$ *is* $I = \langle x\,y - x - y + 1, x^2 - y^2 - 2\,x + 2\,y, y^3 - 3\,y^2 + 3\,y - 1\rangle$.

The Adaptive BMS *algorithm called on* $\mathbf{u}$ *and the stopping monomial* y^5 *returns* $g_1 = x\,y - x - y + 1$*, with shift* x^2*,* $g_2 = x^2 - \frac{1}{3}\,x\,y - y^2 - \frac{5}{3}\,x + \frac{7}{3}\,y - \frac{1}{3}$*, with shift* x^2 *and* $g_3 = y^3 - \frac{1}{2}\,x\,y - 3\,y^2 + \frac{1}{2}\,x + \frac{7}{2}\,y - \frac{3}{2}$*, with shift* y^2*. We can notice that* $\{g_1, g_2, g_3\}$ *is a Gröbner basis but not a reduced Gröbner basis of* I*.*

The Adaptive Scalar-FGLM *algorithm called on* $\mathbf{u}$ *and the set of all the monomials of degree at most* 3 *yields relations* $g'_1 = x\,y - x - y + 1, g'_2 = x^2 - y^2 - 2\,x + 2\,y, g'_3 = y^3 - 3\,y^2 + 3\,y - 1$*. We can notice that* $\{g'_1, g'_2, g'_3\} = \{g_1, g_2 + \frac{1}{3}\,g_1, g_3 + \frac{1}{2}\,g_1\}$ *is a reduced Gröbner basis of* I*.*

As for the BMS algorithm, it is not hard to tweak the Adaptive BMS algorithm so that it returns a reduced Gröbner basis. It suffices to perform an inter-reduction of the relations either at the end of each step of the main **For** loop or just before returning the Gröbner basis, see Algorithm 6.

5.3. Validity of relations

One of the main differences between the BMS and the Scalar-FGLM algorithms is the validity of the relations they return. Given a Gröbner basis returned by both algorithms. Loosely speaking, the Scalar-FGLM algorithm will only ensure that all the relations in the Gröbner basis have the same shifts while for the

Algorithm 6: Tweaked Adaptive BMS algorithm.

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

Output: A set G of relations generating I_M.

$T := \{m \in \mathbb{K}[\mathbf{x}], m \preceq M\}$.
$G := \{1\}$.
$S := \emptyset$.
For all $m \in T$ **do**
  $S' := S$.
  For $g \in G$ **do**
    If $\mathrm{LM}(g) | m$ **then**
      If $\frac{m}{\mathrm{LM}(g)} \notin \mathrm{Stabilize}(S)$ **and** $\#\,\mathrm{Stabilize}\left(S \cup \left\{\mathrm{LM}(g), \frac{m}{\mathrm{LM}(g)}\right\}\right) > d$ **then**
        next. `// skip this relation testing`
      $e := \left[\frac{m}{\mathrm{LM}(g)}\, g\right]_{\mathbf{u}}$
      If $e \neq 0$ **then**
        $S' := S' \cup \left\{\left[\frac{g}{e}, \frac{m}{\mathrm{LM}(g)}\right]\right\}$.
  $S' := \min_{\mathrm{fail}(h)\in S'} \{[h, \mathrm{fail}(h)/\,\mathrm{LM}(h)]\}$.
  $G' := \mathrm{Border}(S')$.
  For $g' \in G'$ **do**
    Let $g \in G$ such that $\mathrm{LM}(g) | \,\mathrm{LM}(g')$.
    If $\mathrm{LM}(g) \nmid m$ **then**
      $g' := \frac{\mathrm{LM}(g')}{\mathrm{LM}(g)}\, g$.
    Else if $\exists\, h \in S, \frac{m}{\mathrm{LM}(g')} | \,\mathrm{fail}(h)$ **then**
      $g' := \frac{\mathrm{LM}(g')}{\mathrm{LM}(g)}\, g - \left[\frac{m}{\mathrm{LM}(h)}\, h\right]_{\mathbf{u}} \frac{\mathrm{LM}(g')\ \mathrm{fail}(h)}{m}\, h$.
    Else $g' := g$.
  $G := \mathrm{InterReduce}(G')$
  $S := S'$.
Return G.

BMS algorithm, the smaller the leading term of a relation is, the larger its shift is computed. See (Berthomieu and Faugère, 2017, Theorem 19).

Naturally, if the given upper bound on the size of the staircase to the Adaptive BMS algorithm is correct, then the shifts computed by the Adaptive BMS algorithm are the same as those computed by the BMS algorithm.

In Examples 8 and 9, we can see that the shifts computed by the Adaptive Scalar-FGLM algorithm are not all the same. This is the main difference between the Scalar-FGLM and the Adaptive Scalar-FGLM algorithms.

In fact, we prove in the following Theorem 15 that the larger the leading term of a computed relation, the larger its shift.

Theorem 15. *Let* $\mathbf{u}$ *be a sequence,* $\prec$ *be a monomial ordering and* d *be the size of the output staircase* S*. Let* $S_M = \{m \in S,\ m \prec M\}$*.*

Calling the Adaptive BMS *algorithm on* $\mathbf{u}$*,* $\prec$*,* d *and a stopping monomial* M *yields relations* $g_1, \ldots, g_r$ *and shifts* $v_1, \ldots, v_r$ *such that*

$$\forall\, i, 1 \leq i \leq r, \quad v_i \operatorname{LM}(g_i) \leq M$$

and g_i *is valid with a shift* v_i*, potentially* 0*.*

Calling the Adaptive Scalar-FGLM *algorithm on* $\mathbf{u}$*,* $\prec$ *and* d *yields relations* $g'_1, \ldots, g'_{r'}$ *such that*

$$\forall\, i, 1 \leq i \leq r', \quad \deg \operatorname{LM}(g'_i) \leq d$$

and g'_i *has a shift* $S_{\operatorname{LM}(g'_i)} \cup \{\operatorname{LM}(g'_i)\}$ *if* $\operatorname{LM}(g'_i) \succ \max_{\prec}(S)$ *and* S *otherwise.*

Proof. The first part is clear from the behavior of both the BMS and the Adaptive BMS algorithms.

The second part comes from the fact that if g'_i, with $\operatorname{LM}(g'_i) = t$ is found before S is completed, then it was because the matrix $H_{S^t \cup \{t\}, S^t \cup \{t\}}$ had a rank default, where S^t is the state for S at loop t. Furthermore, $S^t = S_{\operatorname{LM}(g'_i)} = S_t$.

Otherwise, it is computed by solving $H_{S,S}\, \boldsymbol{\alpha} + H_{S,\{t'\}} = 0$ so that the relation has only been tested with a shift S. □

In a way, the behavior of the Adaptive Scalar-FGLM algorithm is the opposite of the behaviors of the BMS and the Adaptive BMS algorithms.

Furthermore, if one uses Algorithm 5 instead of the Adaptive Scalar-FGLM algorithm, then each returned relation g'_i has a shift $S_{\operatorname{LM}(g'_i)} \cup \{\operatorname{LM}(g'_i)\}$.

Example 11. *Let us consider the sequence* $\mathbf{u} = (F_{i+1})_{(i,j)\in\mathbb{N}^2}$*, where* $(F_i)_{i\in\mathbb{N}}$ *is the Fibonacci sequence. Its ideal of relation is* $\langle y - 1, x^2 - x - 1\rangle$ *so that its staircase has size* 2*.*

Calling the Adaptive Scalar-FGLM *algorithm on this sequence with this bound of the staircase makes us creating the matrices*

$H_{\{1\},\{1\}}$, *which is full rank, hence* $1 \in S$*;*
$H_{\{1,y\},\{1,y\}}$, *which is not full rank, hence the relation* $y - 1$ *is found with a shift* $\{1, y\}$*;*
$H_{\{1,x\},\{1,x\}}$, *which is full rank, hence* $x \in S$.

Now, the staircase is found so it remains to solve $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{x^2\}} = 0$ *yielding the relation* $x^2 - x - 1$ *with a shift* S.

5.4. Monomial ordering and Set of Terms

In this section, we study how both algorithms handle a monomial ordering that is not a weighted degree ordering. The classical specification of the BMS algorithm are that the ordering must be a weighted ordering. However, when running the Adaptive BMS algorithm, the upper bound on the staircase size makes us never visit monomials of degree more than twice this size. Therefore, we can now use any monomial ordering with the Adaptive BMS algorithm by just enumerating, in increasing order, all the monomials of degree less than twice the upper bound.

This allows us to deal with ideal in shape position with both the Adaptive BMS and the Adaptive Scalar-FGLM algorithms.

Theorem 16. *Let* $\mathbf{u}$ *be a linear recurrent sequence whose ideal of relation* I *is in shape position for the* $\text{LEX}(x_n \prec \cdots \prec x_2 \prec x_1)$ *ordering, i.e. there exist* g_n *squarefree and* $f_{n-1}, \ldots, f_1 \in \mathbb{K}[x_n]$ *with* $\deg g_n = d, \deg f_i < d$ *such that* $I = \langle g_n(x_n), x_{n-1} - f_{n-1}(x_n), \ldots, x_1 - f_1(x_n)\rangle$.

Assuming no error is thrown in the execution of the Adaptive Scalar-FGLM *algorithm called on* $\mathbf{u}$*,* d *and* $\text{LEX}(x_n \prec \cdots \prec x_2 \prec x_1)$*, then the ouput is* I.

Calling the Adaptive BMS *algorithm on* $\mathbf{u}$*,* d *and* $\text{LEX}(x_n \prec \cdots \prec x_2 \prec x_1)$ *yields* I.

Proof. Assuming no error is thrown during the execution of the Adaptive Scalar-FGLM algorithm, the staircase is incrementally updated from $\emptyset$ to $\left\{1, x_n, \ldots, x_n^{d-1}\right\}$. Then, the staircase size is reached and the early termination procedure solves the system $H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{t\}} = 0$ for $t \in \left\{x_n^d, x_{n-1}, \ldots, x_1\right\}$ yielding $g_n(x_n), x_{n-1} - f_{n-1}(x_n), \ldots, x_1 - f_1(x_n)$.

For the Adaptive BMS algorithm, we visit every monomial of degree at most $2\,d - 1$. The first relation, $g_n(x_n)$ is computed by the algorithm visiting monomials $1, x_n, \ldots, x_n^{2\,d-1}$ like the BM algorithm. Then, each relation $x_i - f_i(x_n)$ is computed by visiting monomials $x_i, x_i\,x_n, \ldots, x_i\,x_n^{d-1}$, all of degree less than $2\,d - 1$. □

Example 12. *We let* $\mathbf{u} = (F_{4\,i+k+1})_{(i,j,k)\in\mathbb{N}^3}$*, where* $(F_i)_{i\in\mathbb{N}}$ *is the Fibonacci sequence. The ideal of relations of* $\mathbf{u}$ *is* $I = \langle z^2 - z - 1, y - 1, x - 3\,z - 2\rangle$ *with a staircase of size* 2.

For the Adaptive Scalar-FGLM *called on* $\mathbf{u}$*,* $d = 2$ *and the* lex$(z \prec y \prec x)$ *ordering, the algorithm creates the matrices*

$H_{\{1\},\{1\}} = (1)$*, which is full rank, hence* $1 \in S$*;*
$H_{\{1,z\},\{1,z\}} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$*, which is full rank, hence* $z \in S$*.*

Now, the staircase is found so it remains to solve

$H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{z^2\}} = 0$ *yielding the relation* $g_1 = z^2 - z - 1$*;*
$H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{y\}} = 0$ *yielding the relation* $g_2 = y - 1$*;*
$H_{S,S}\,\boldsymbol{\alpha} + H_{S,\{x\}} = 0$ *yielding the relation* $g_3 = x - 3\,z - 2$*.*
The algorithm returns $\langle g_1, g_2, g_3\rangle = I$*.*

Calling the Adaptive BMS *algorithm on* $\mathbf{u}$*,* $d = 2$*, the stopping monomial* $x\,z$ *and* lex$(z \prec y \prec x)$ *ordering makes us visit the set of all monomials of degree at most* $2\,d - 1 = 3$ *less than* $x\,z$*, i.e.* $\{1, z, z^2, z^3, y, y\,z, y\,z^2, y^2, y^2\,z, y^3, x, x\,z\}$*.*

The algorithms tests the relation $g = 1$ *in* $u_{0,0,0} = F_1 = 1$ *where it fails. It has now relations* $g_1 = x$*,* $g_2 = y$ *and* $g_3 = z$*.*
Testing $g_3 = z$ *in* $u_{0,0,2} = F_2 = 1$*, it updates now the relation to* $g_3 = z - 1$*.*
Going on testing $g_3 = z - 1$ *in* $u_{0,0,2} = F_3 = 2$ *and* $u_{0,0,3} = F_4 = 3$*, it is able to guess that* $g_3 = z^2 - z - 1$*. The staircase is now* $\{1, z\}$ *of size* 2 *so it has been found. As anticipated, there is no need to go further in that direction.*
Testing $g_2 = y$ *in* $u_{0,1,0} = F_1 = 1$*, the relation is updated to* $g_2 = y - 1$*.*
Then, it checks that this relation is valid in $u_{0,1,1}$ *but skips* $u_{0,1,2}, u_{0,2,0}, u_{0,2,1}, u_{0,3,0}$ *thanks to its criterion.*
It remains to test $g_3 = x$ *in* $u_{1,0,0} = F_5 = 5$*. It fails and the algorithm updates the relation to* $g_3 = x - 5$*.*
Finally, $g_3 = x - 5$ *is tested in* $u_{1,0,1} = F_6 = 8$ *and the relation is updated to* $g_3 = x - 3\,z - 2$*.*
The algorithm returns $\langle g_1, g_2, g_3\rangle = I$*.*

6. Complexity and Benchmarks of the adaptive variants

In this section, we present some benchmarks to compare how the Adaptive BMS and the Adaptive Scalar-FGLM algorithms behave.

Five families of ideals of relations are used to make the sequences.

- In the first family, the leading monomials of the ideal of relations are $\langle y^{\lfloor d/2\rfloor}, x^d\rangle$. Thus, its staircase is a rectangle of size around $d^2/2$. In three variables, the leading monomials are $\langle z^{\lceil d/3\rceil}, y^{\lfloor d/2\rfloor}, x^d\rangle$, so that the staircase is a rectangular cuboid of size around $d^3/6$. This family will be called *Rectangle*.

- In the second family, the leading monomials of the ideal of relations are $\langle x\,y, y^d, x^d\rangle$. Thus, its staircase looks like a L and has size $2\,d-1$. In three variables, the leading monomials are $\langle y\,z, x\,z, x\,y, z^d, y^d, x^d\rangle$, so that the staircase has size $3\,d-2$. This family will be called L *shape*. It was considered as the worst case in Berthomieu et al. (2015, 2017) for the Adaptive Scalar-FGLM algorithm for the number of queries. It should also be a worst case for the Adaptive BMS algorithm.

- In the third family, the leading monomials of the ideal of relations are all the monomials of degree d. Thus, its staircase is a simplex and has size $\binom{d+1}{2} = \frac{d\,(d+1)}{2}$ in two variables and size $\binom{d+2}{3} = \frac{d\,(d+1)\,(d+2)}{6}$ in three variables. This family will be called *Simplex*. It should be the best case for both the Scalar-FGLM and the BMS algorithms.

- In the fourth family, the leading monomials of the ideal of relations are $\langle y^d, x\rangle$. Thus, its staircase looks like a line and has size d. In three variables, the leading monomials are $\langle z^d, y, x\rangle$, so that the staircase has also size d. This is the generic family for a $\text{lex}(y \prec x)$ or $\text{lex}(z \prec y \prec x)$ basis and this example corresponds to the change of ordering application, see Section 5.4. This family will be called *Shape position*.

- The last family is coming from economic modelings, see (Morgan, 2009, Equation 7.4). Contrary to the others, the number of variables n varies while the degree of the polynomials defining the system is fixed: one equation of degree 1 and $n-1$ equations of degree 2. As expected, this system has degree 2^{n-1} and behaves generically for the $\text{lex}(x_n \prec \cdots \prec x_1)$ ordering. That is, it is in shape position and its Gröbner basis is $\langle P_n(x_n), x_{n-1} - P_{n-1}(x_n), \ldots, x_1 - P_1(x_n)\rangle$, with $\deg P_n = 2^{n-1}$ and for all k, $1 \le k \le n-1$, $\deg P_k < \deg P_n$. This family will be called *Eco*.

For the first three families, we called the algorithms with the $\text{drl}(y \prec x)$ and $\text{drl}(z \prec y \prec x)$ orderings, for the fourth one, we called them with the $\text{lex}(y \prec x)$ and $\text{lex}(z \prec y \prec x)$ orderings. Finally, for the last one, we called the algorithms with the $\text{lex}(x_n \prec \cdots \prec x_1)$ ordering.

For the Adaptive BMS algorithm, we used Proposition 9 to estimate sharply the stopping monomial.

6.1. Counting the number of table queries

The Adaptive Scalar-FGLM algorithm computes all the multi-Hankel matrices whose rows and columns are all the terms that are in the staircase or are a leading monomial in the Gröbner basis.

Likewise, the Adaptive BMS algorithm needs to test each relation, with support in $S \cup \mathrm{LM}(\mathcal{G})$, shifted by as many monomial as in S.

Therefore, we have the following proposition.

Proposition 17. *Let $\mathbf{u} = (u_\mathbf{i})_{\mathbf{i}\in\mathbb{N}^n}$ be a sequence and $\mathcal{G}$ be a reduced Gröbner basis of its ideal of relations for a total degree ordering.*

Let S be the staircase of $\mathcal{G}$, $S^+ = S \cup \mathrm{LM}(\mathcal{G})$. Let $S + T = \{s\,t,\ s \in S, t \in T\}$ and $2\,S = S + S = \{s\,s',\ s, s' \in S\}$.

Let d_S be the greatest degree of the elements in S, $d_\mathcal{G}$ be the greatest degree of the elements in $\mathcal{G}$ and $d_{\max} = \max(d_S, d_\mathcal{G})$.

Let $\mathcal{S}(d)$ be the simplex of all monomials of degree d.

Then, the Adaptive BMS *algorithm needs to perform at least $\#(S + S^+)$ and at most $\#\mathcal{S}(d_S + d_{\max}) = \binom{n+d_S+d_{\max}}{n}$ queries to the sequence.*

The Adaptive Scalar-FGLM *algorithm needs to perform at least $\#(2\ S)$ and fewer than $\#(2\,S^+)$ queries to $\mathbf{u}$. In the worst case, this number grows as $(\#\,S^+)^2$.*

In the experiments of Figures 1, 2 and 3, we can see that the Adaptive Scalar-FGLM algorithm always requires fewer queries than the Adaptive BMS algorithm to recover the relations. For instance, for the Rectangle family, the ratio seems to be around 2.

For the L shape family, the size of the staircase only grows as $O(d)$. Our experiments suggest that the number of required queries grows as $O(d^n)$ for the Adaptive BMS algorithm, while it only grows as $O(d^2)$ for the Adaptive Scalar-FGLM algorithm. This can be a huge advantage in dimension at least 3.

Furthermore, the Adaptive BMS algorithm cannot take profit from the size of the staircase in the L shape family compared to the Simplex family while the Adaptive Scalar-FGLM algorithm is able to reduce the number of queries in this case.

In Figure 3, we can see that the Adaptive BMS and the Adaptive Scalar-FGLM algorithms require roughly the same number of table elements to recover the relations. On the other hand, if we call the BMS or the Scalar-FGLM algorithms on all the monomials up to Proposition 9 bound, then the queries overhead make them unpractical. This is due to the fact that the adaptive algorithms, on this example, behave like a call to the BM algorithm on the subsequence $([x_n^i])_{i\in\mathbb{N}}$, where all the indices are zero but the last one, and then $n-1$ Hankel linear system solving to recover the relations $x_k - P_k(x_n)$ for $1 \le k \le n-1$.

6.2. Counting the number of basic operations

The complexity of the BMS algorithm has been studied in Sakata (2009) yielding the following proposition.

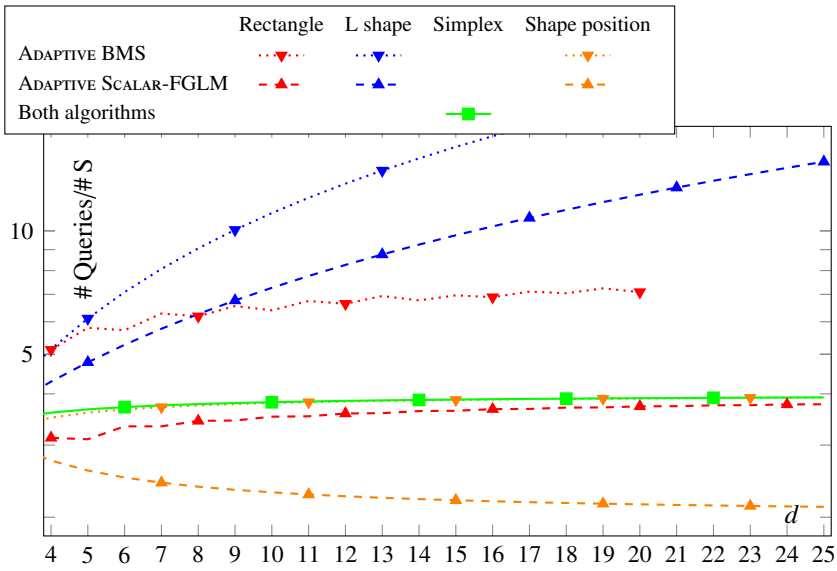


Figure 1: Number of table queries (2D): Adaptive BMS & Adaptive Scalar-FGLM

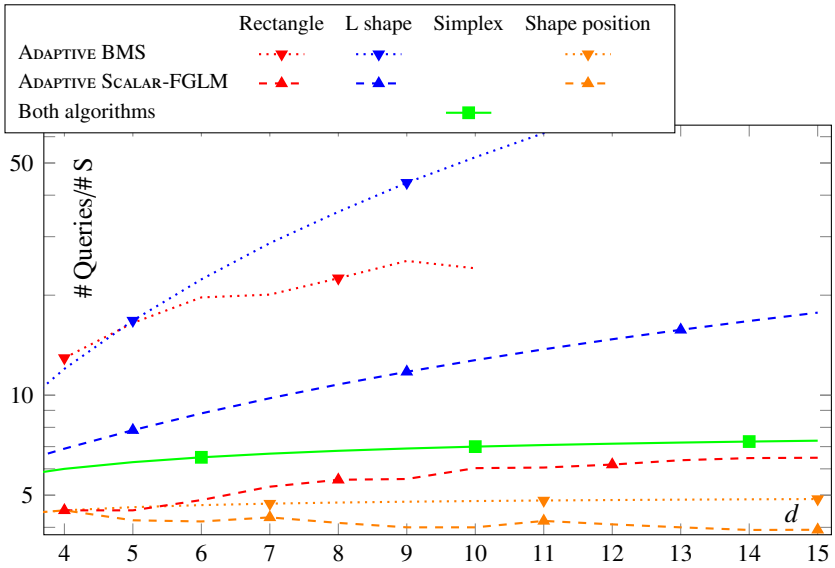


Figure 2: Number of table queries (3D): Adaptive BMS & Adaptive Scalar-FGLM

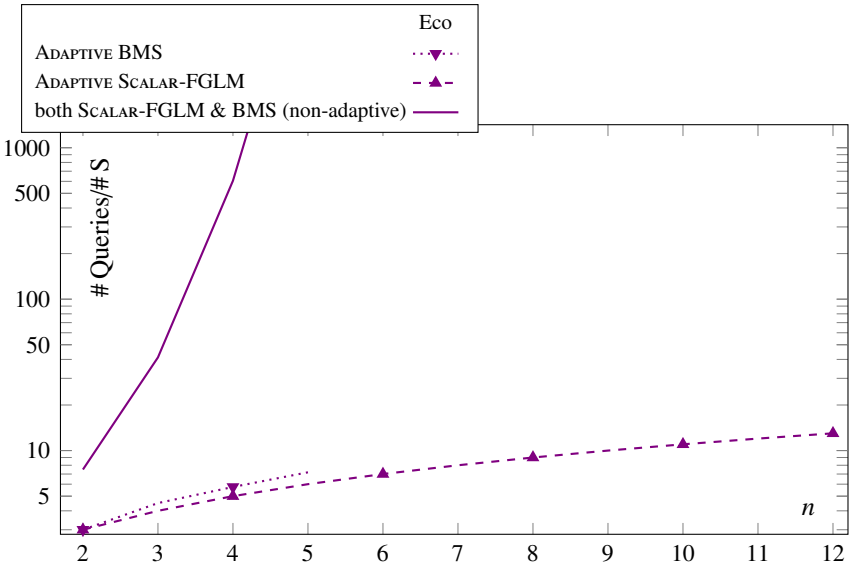


Figure 3: Number of table queries (nD Eco): Adaptive BMS, Adaptive Scalar-FGLM and both BMS and Scalar-FGLM

Proposition 18. *Let* $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ *be a sequence,* $\mathcal{G}$ *be a minimal Gröbner basis of its ideal of relations for a total degree ordering and* S *be the staircase of* $\mathcal{G}$*.*

Then, the BMS *algorithm performs at most* $O\left((\#S)^2\ \mathrm{LM}(\mathcal{G})\right)$ *operations to recover the ideal of relations of* $\mathbf{u}$*.*

Obviously, the bound of Proposition 18 on the number of basic operations applies to the Adaptive BMS algorithm. Yet, since the number of skipped relation testings is hard to predict, it is not clear how to make it sharper for the Adaptive BMS algorithm.

The Adaptive Scalar-FGLM computes the rank of a matrix of size at most $\#S$. Furthermore, it solves as many linear systems with this matrix as there are polynomials in the Gröbner basis. All in all, we have the following result.

Proposition 19. *Let* $\mathbf{u} = (u_{\mathbf{i}})_{\mathbf{i}\in\mathbb{N}^n}$ *be a sequence,* $\mathcal{G}$ *be a reduced Gröbner basis of its ideal of relations for a total degree ordering and* S *be the staircase of* $\mathcal{G}$*.*

Then, the number of operations performed by the Adaptive Scalar-FGLM *algorithm to recover the ideal of relations of* $\mathbf{u}$ *is at most* $O\left((\#S)^2\ (\#S\ +\#\ \mathrm{LM}(\mathcal{G}))\right)$*.*

In the following Figures 4, 5 and 6, we report on the ratio between the number of basic operations and the cube of the size of the staircase.

While, for these sizes the Adaptive Scalar-FGLM algorithm performs fewer operations than the Adaptive BMS algorithm, it is not clear if this remains true

asymptotically, especially in dimension 2.

Concerning the L shape family, although the Adaptive BMS algorithm do not reduce much its number of table queries, it performs in fact much fewer basic operations than the BMS algorithm. For instance, in (Berthomieu and Faugère, 2017, Section 6), we can see that the BMS algorithm performs four times (resp. seven times) as many basic operations as the Adaptive BMS algorithm in dimension 2 (resp. dimension 3).

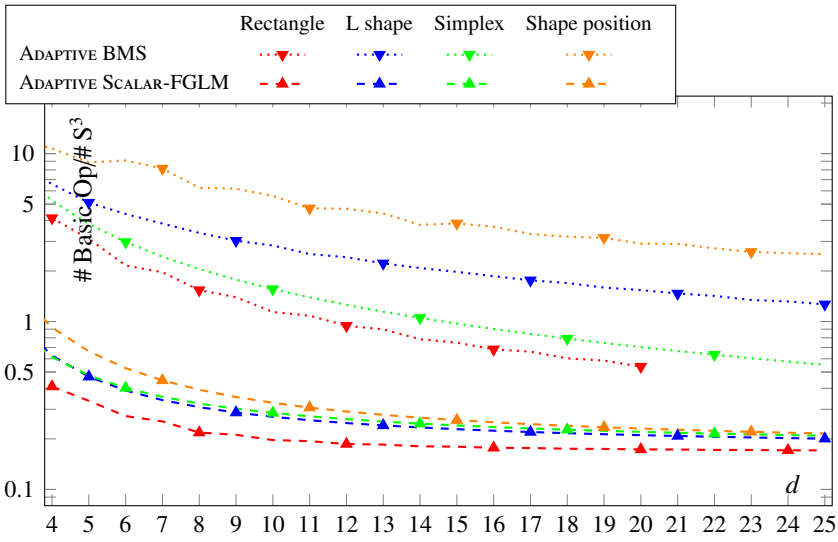


Figure 4: Number of basic operations (2D): Adaptive BMS & Adaptive Scalar-FGLM

It is also possible that the larger number of operations the Adaptive BMS algorithm performs compared to the Scalar-FGLM algorithm is due to the larger number of queries it needs to recover the relations.

Therefore, we now also compare the ratio between their number of basic operations and their number of queries in Figures 7, 8 and 9.

In dimension 2, the Adaptive Scalar-FGLM algorithm seems to have a better ratio between the number of operations and the number of queries than the Adaptive BMS algorithm. Yet, once again, it is possible that this statement is not true for larger d.

In dimension 3, however, our experiments lead us to believe that this ratio will always be larger for the Adaptive BMS algorithm than for the Adaptive Scalar-FGLM algorithm.

Acknowledgements

We thank the anonymous referees for their careful reading and their helpful comments. The authors are partially supported by the PGMO grant GAMMA.

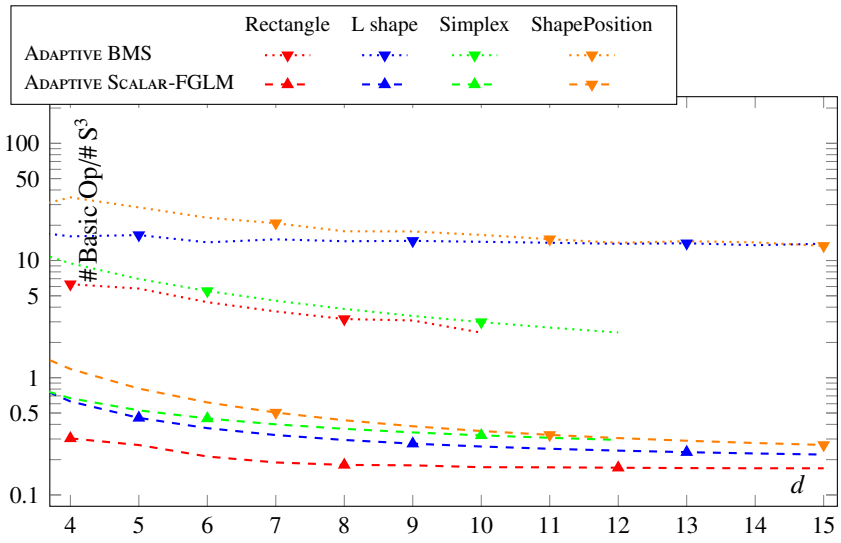


Figure 5: Number of basic operations (3D): Adaptive BMS & Adaptive Scalar-FGLM

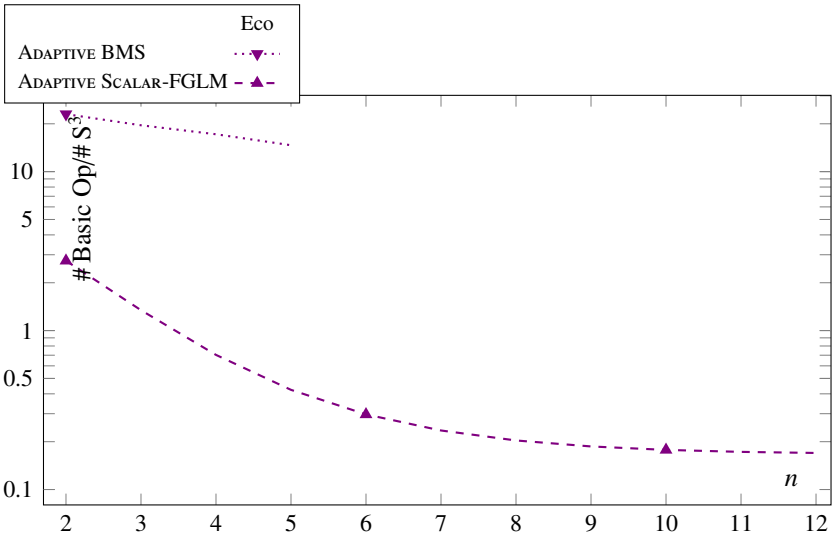


Figure 6: Number of basic operations (nD Eco): Adaptive BMS & Adaptive Scalar-FGLM

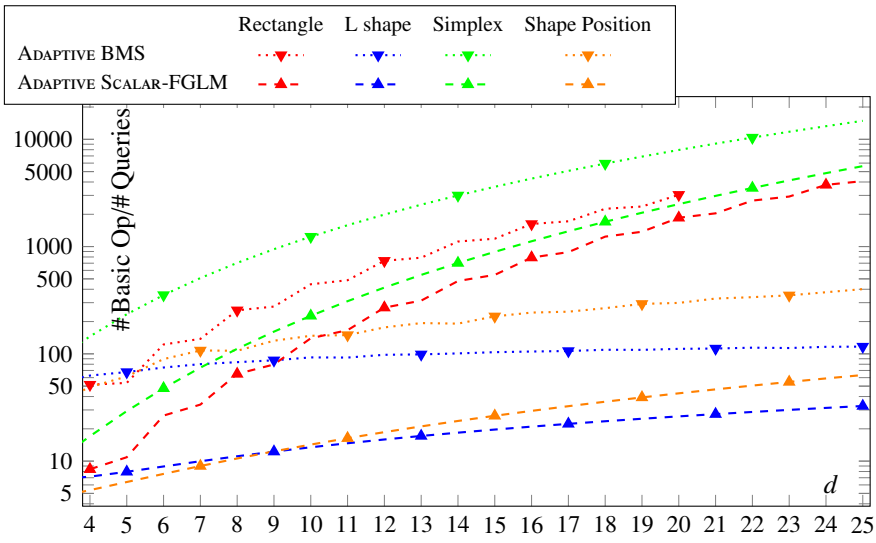


Figure 7: Number of basic operations by queries (2D): Adaptive BMS & Adaptive Scalar-FGLM

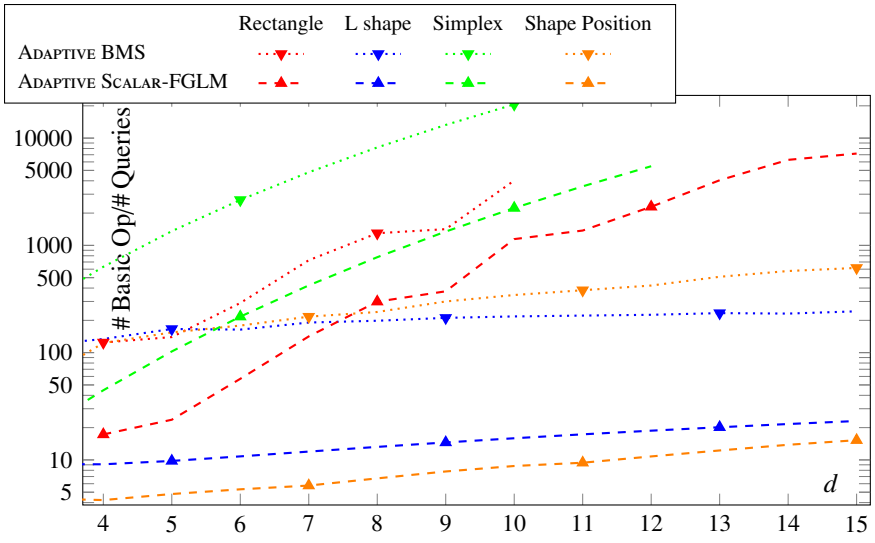


Figure 8: Number of basic operations by queries (3D): Adaptive BMS & Adaptive Scalar-FGLM

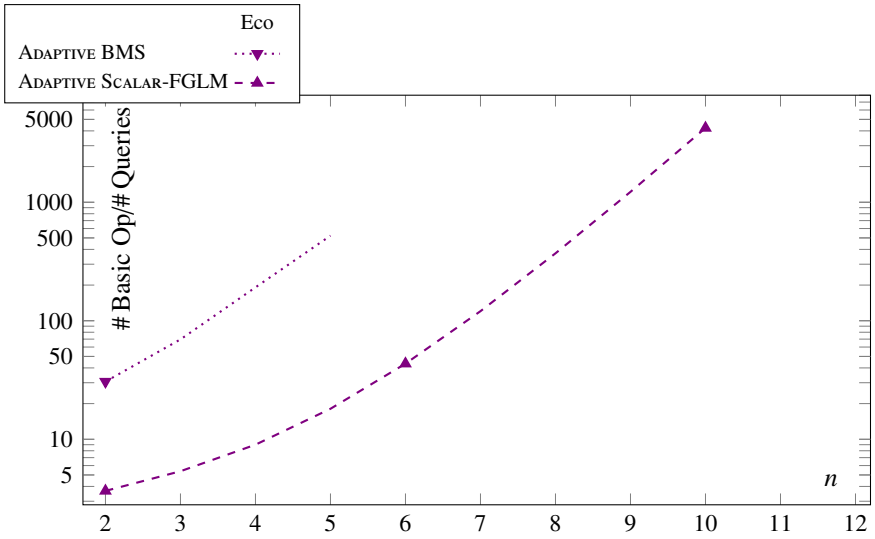


Figure 9: Number of basic operations by queries (nD Eco): Adaptive BMS & Adaptive Scalar-FGLM

Banderier, C., Flajolet, P., 2002. Basic analytic combinatorics of directed lattice paths. Theoret. Comput. Sci. 281 (1–2), 37–80, selected Papers in honour of Maurice Nivat.
URL http://www.sciencedirect.com/science/article/pii/S0304397502000075

Benoit, A., Chyzak, F., Darrasse, A., Gerhold, S., Mezzarobba, M., Salvy, B., 2010. The Dynamic Dictionary of Mathematical Functions (DDMF). In: Fukuda, K., Hoeven, J. v. d., Joswig, M., Takayama, N. (Eds.), Mathematical Software – ICMS 2010. Springer, Berlin, Heidelberg, pp. 35–41.
URL http://dx.doi.org/10.1007/978-3-642-15582-6_7

Berlekamp, E., 1968. Nonbinary BCH decoding. IEEE Trans. Inform. Theory 14 (2), 242–242.

Berthomieu, J., Boyer, B., Faugère, J.-Ch., 2015. Linear Algebra for Computing Gröbner Bases of Linear Recursive Multidimensional Sequences. In: 40th International Symposium on Symbolic and Algebraic Computation. Proceedings of the 40th International Symposium on Symbolic and Algebraic Computation. Bath, United Kingdom, pp. 61–68.

Berthomieu, J., Boyer, B., Faugère, J.-Ch., 2017. Linear Algebra for Computing Gröbner Bases of Linear Recursive Multidimensional Sequences. Journal

of Symbolic Computation 83 (Supplement C), 36–67, special issue on the conference ISSAC 2015: Symbolic computation and computer algebra.
URL https://hal.inria.fr/hal-01253934

Berthomieu, J., Faugère, J.-Ch., 2016. Guessing Linear Recurrence Relations of Sequence Tuples and P-recursive Sequences with Linear Algebra. In: 41st International Symposium on Symbolic and Algebraic Computation. Waterloo, ON, Canada, pp. 95–102.

Berthomieu, J., Faugère, J.-Ch., 2017. In-depth comparison of the Berlekamp – Massey – Sakata and the Scalar-FGLM algorithms: the non adaptive variants, preprint.
URL https://hal.inria.fr/hal-01516708

Berthomieu, J., Faugère, J.-Ch., 2018. A Polynomial-Division-Based Algorithm for Computing Linear Recurrence Relations. In: ISSAC 2018 - 43rd International Symposium on Symbolic and Algebraic Computation. New York, United States, p. 8.
URL https://hal.inria.fr/hal-01784369

Bose, R., Ray-Chaudhuri, D., 1960. On a class of error correcting binary group codes. Information and Control 3 (1), 68 – 79.
URL http://www.sciencedirect.com/science/article/pii/S0019995860902874

Bostan, A., Bousquet-Mélou, M., Kauers, M., Melczer, S., 2014. On 3-dimensional lattice walks confined to the positive octant, to appear in Annals of Combinatorics.

Bousquet-Mélou, M., Mishna, M., 2010. Walks with small steps in the quarter plane. In: Algorithmic probability and combinatorics. Vol. 520 of Contemp. Math. Amer. Math. Soc., Providence, RI, pp. 1–39.
URL http://dx.doi.org/10.1090/conm/520/10252

Bousquet-Mélou, M., Petkovšek, M., 2003. Walks confined in a quadrant are not always d-finite. Theoret. Comput. Sci. 307 (2), 257–276, random Generation of Combinatorial Objects and Bijective Combinatorics.
URL http://www.sciencedirect.com/science/article/pii/S0304397503002196

Bras-Amorós, M., O'Sullivan, M. E., 2006. The correction capability of the Berlekamp–Massey–Sakata algorithm with majority voting. Applicable Alge-

bra in Engineering, Communication and Computing 17 (5), 315–335.
URL http://dx.doi.org/10.1007/s00200-006-0015-8

Cox, D., Little, J., O'Shea, D., 2015. Ideals, Varieties, and Algorithms, 4th Edition. Undergraduate Texts in Mathematics. Springer, New York, an introduction to computational algebraic geometry and commutative algebra.

Cox, D. A., Little, J., O'Shea, D., 2005. Using Algebraic Geometry, 2nd Edition. Vol. 185 of Graduate Texts in Mathematics. Springer, New York.

Faugère, J.-Ch., Gianni, P., Lazard, D., Mora, T., 1993. Efficient Computation of Zero-dimensional Gröbner Bases by Change of Ordering. J. Symbolic Comput. 16 (4), 329–344.

Faugère, J.-Ch., Mou, C., 2011. Fast Algorithm for Change of Ordering of Zero-dimensional Gröbner Bases with Sparse Multiplication Matrices. In: Proc. of the 36th ISSAC. ACM, pp. 115–122.

Faugère, J.-Ch., Mou, C., 2017. Sparse FGLM algorithms. Journal of Symbolic Computation 80 (3), 538 – 569.

Fitzpatrick, P., Norton, G., 1990. Finding a basis for the characteristic ideal of an n-dimensional linear recurring sequence. IEEE Trans. Inform. Theory 36 (6), 1480–1487.

Guisse, V., 2016. Algèbre linéaire dédiée pour les algorithmes SCALAR-FGLM et Berlekamp-Massey-Sakata. Master's thesis, Université Paris-Diderot.

Hocquenghem, A., 1959. Codes correcteurs d'erreurs. Chiffres 2, 147 – 156.

Jonckheere, E., Ma, C., 1989. A simple Hankel interpretation of the Berlekamp-Massey algorithm. Linear Algebra Appl. 125 (0), 65 – 76.
URL http://www.sciencedirect.com/science/article/pii/0024379589900323

Kaltofen, E., Pan, V., 1991. Processor efficient parallel solution of linear systems over an abstract field. In: SPAA '91. ACM Press, New York, N.Y., pp. 180–191.

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

Kaltofen, E., Yuhasz, G., 2013b. On the Matrix Berlekamp-Massey Algorithm. ACM Trans. Algorithms 9 (4), 33:1–33:24.
URL http://doi.acm.org/10.1145/2500122

Lee, K., 2016. Decoding of differential AG codes. Advances in Mathematics of Communications 10 (2), 307–319.
URL http://aimsciences.org//article/id/45525cec-3d10-4bc9-ba47-dfa02b6eb5f9

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

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

Morgan, A., 2009. Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems. Society for Industrial and Applied Mathematics.
URL https://epubs.siam.org/doi/abs/10.1137/1.9780898719031

Sakata, S., 1988. Finding a minimal set of linear recurring relations capable of generating a given finite two-dimensional array. J. Symbolic Comput. 5 (3), 321–337.
URL http://www.sciencedirect.com/science/article/pii/S0747717188800336

Sakata, S., 1990. Extension of the Berlekamp-Massey algorithm to N Dimensions. Inform. and Comput. 84 (2), 207–239.
URL http://dx.doi.org/10.1016/0890-5401(90)90039-K

Sakata, S., 1991. Decoding binary 2-D cyclic codes by the 2-D Berlekamp-Massey algorithm. IEEE Trans. Inform. Theory 37 (4), 1200–1203.
URL http://dx.doi.org/10.1109/18.86974

Sakata, S., 2009. The BMS Algorithm. In: Sala, M., Sakata, S., Mora, T., Traverso, C., Perret, L. (Eds.), Gröbner Bases, Coding, and Cryptography. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 143–163.
URL http://dx.doi.org/10.1007/978-3-540-93806-4_9

Sakata, S., Fujisawa, M., 2018. Fast Decoding of Dual Multipoint Codes From Algebraic Curves Up to the Kirfel–Pellikaan Bound. IEEE Transactions on Information Theory 64 (6), 4452–4466.

Wiener, N., 1964. Extrapolation, Interpolation, and Smoothing of Stationary Time Series. The MIT Press.