A Word Reconstruction Problem for Polynomial Regular Languages

Mehdi Golafshan, Annika Huch[1][0009−0005−1145−5806], and Michel Rigo[2][0000−0001−7463−8507]

[1] Department of Mathematics, University of Liège, Belgium,
{mgolafshan,m.rigo}@uliege.be

[2] Department of Computer Science, Kiel University, Germany,
ahu@informatik.uni-kiel.de

Abstract. The reconstruction problem concerns the ability to uniquely determine an unknown word from querying information on the number of occurrences of chosen subwords. In this work, we focus on the reconstruction problem when the unknown word belongs to a known polynomial regular language, i.e., its growth function is bounded by a polynomial. Exploiting the combinatorial and structural properties of these languages, we are able to translate queries into polynomial equations and transfer the problem of unique reconstruction to finding those sets of queries such that their polynomial equations have a unique integer solution. Alongside combinatorial properties of the equations and polynomials we completely characterize which queries are necessary to uniquely reconstruct the word in the case of two loops. Further, we integrate techniques from real algebraic geometry and Tarski–Seidenberg Theorem on quantifier elimination to show that we can decide for a constant number of queries whether they suffice for a unique reconstruction.

Keywords: Regular languages ; Binomial coefficients of words ; Reconstruction problem ; Tarski–Seidenberg theorem ; Polynomial equations

1 Introduction

The binomial coefficient $\binom{u}{v}$ of two words $u = a_1 \cdots a_n$ and v, where the a_i's are letters, counts the number of times v appears as a subword of u,

$$\binom{u}{v} = \#\{i_1 < \cdots < i_\ell \mid a_{i_1} \cdots a_{i_\ell} = v\}.$$

The word *reconstruction problem* is usually expressed as follows [12, 16]. Let Σ be a finite alphabet and g be a word of length n (to guess). With the knowledge of the numerical values of the binomial coefficients $\binom{g}{q_1}, \ldots, \binom{g}{q_t}$ for some queries $q_1, \ldots, q_t$, one has to uniquely recover g. In the classical setting, one considers all words q_i of some length ℓ and the question is therefore to find the least value $\ell = f(n)$ that allows one to uniquely determine — or, reconstruct — any word g of length n. For results about bounding $f(n)$, see [6, 13, 15]. Variants of this problem exist: for instance, when the sequence of queries $q_1, \dots, q_t$ is not predetermined, the choice of next query q_{i+1} depends on the previous values $\binom{g}{q_1}, \dots, \binom{g}{q_i}$ provided to the guesser. In such a dynamical setting, one tries to determine a strategy and therefore the maximal length $t(n)$ of an optimal sequence of queries, in the worst case scenario, for words $g \in \Sigma^n$. For results in that direction, see [21]. It is usual to assume that the length n of the word to guess is given. However, this is not a strong requirement if one is allowed to ask $\binom{g}{a}$ for all letters $a \in \Sigma$: $n = \sum_{a\in\Sigma} \binom{g}{a}$.

In this article, we consider another variant of this reconstruction problem in a setting where the word g to guess is chosen in a known language but not in the whole set Σ^n. This question is inspired by [8, 9] where words are restricted to lie in what the authors call some *code-book*.

As an introductory example, take a Sturmian word $\mathbf{w} \in \{0,1\}^\omega$ (i.e., an infinite binary word with factor complexity equal to $n+1$). Assume that the word g to determine is a factor of $\mathbf{w}$, i.e., g belongs to the language $L(\mathbf{w})$ of $\mathbf{w}$. It is a well-known result [22] that, for all factors g and g' with the same length $n \geq 2$ of a Sturmian word, we have $g = g'$ if and only if $\binom{g}{u} = \binom{g'}{u}$ for all $u \in \{0,1\}^2$. The knowledge of $\binom{g}{0}$, $\binom{g}{1}$ and $\binom{g}{01}$ is therefore enough to characterize g. With these three values, we know the length $|g| = |g|_0 + |g|_1$ and we easily recover the four coefficients $\binom{g}{u}$ for $u \in \{0,1\}^2$. One can therefore build the list of factors of length $|g|$ of $\mathbf{w}$ and compute the corresponding values of the binomial coefficients for subwords of length 2. Such a table, e.g., Table 1, suffices to determine g. As a conclusion, three queries are enough to reconstruct a factor of arbitrary length picked in a known Sturmian word. Unlike the general case, where the number of queries is a function of the length of the word to be guessed, here we are dealing with a constant number of queries, even without knowing the length of the word to guess. Of course, knowing the input language is a crucial information: we have $n+1$ candidates of length n in $L(\mathbf{w})$ instead of 2^n in the full language $\{0,1\}^n$. Similar reconstruction results exist for the Tribonacci word and hypercubic billiard words [14, 28]. See also the recent paper [29] for more

g	$\lvert g\rvert_0$	$\lvert g\rvert_1$	$\binom{g}{01}$
001001	4	2	6
001010	4	2	5
010010	4	2	4
010100	4	2	3
100100	4	2	2
100101	3	3	5
101001	3	3	4

Table 1. The 7 factors of length 6 in the Fibonacci word.

families of words. Indeed, for these words their 2-binomial complexity coincides with their factor complexity. This means that binomial coefficients for subwords of length 2 suffice to distinguish factors of the same length.

In this article, our aim is to consider words from a regular language with polynomial growth, i.e., the number of words of length n is in $\mathcal{O}(n^k)$ for some integer $k \geq 0$. These languages have a well-known structure, see [25]. For more, see also [23, 24]. They are finite union of languages of the form

$$L = u_1 v_1^* u_2 \cdots u_t v_t^* u_{t+1} \tag{1}$$

where $t \geq 1$ and $v_1, \ldots, v_t$ are non-empty. It is enough to consider a single such component: If necessary, our reconstruction method may be applied separately to each component.

As an example, consider $t = 2$ and the language $L = 1(01)^*10(100)^*0$. Its minimal automaton is depicted in Figure 1 (sink state and dead transitions have not been represented). It exhibits the general structure of polynomial regular languages: (layered) distinct closed paths that never intersect. We have a t-layered automaton: when visiting a cycle, it is never possible to return to a previously traversed cycle.

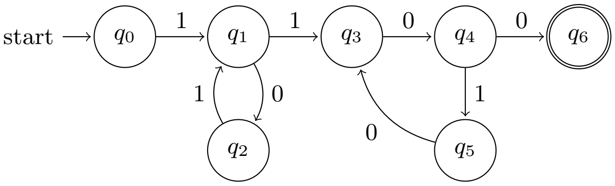

Fig. 1. A trim DFA accepting $1(01)^*10(100)^*0$.

A motivation for considering this type of polynomial language arises from the study of k-automatic sets, i.e., subsets S of $\mathbb{N}$ whose k-ary expansions form a regular language. Indeed, the counting function $\pi_S(n) = \#(S \cap \{0, \ldots, n\})$ of such a set is either of the form $\pi_S(n) = \mathcal{O}((\log n)^d)$ for some $d \geq 1$ or, $\pi_S(n) > n^\alpha$ for some $\alpha > 0$ and large enough n, see [27]. In the first case, the set S is said to be *sparse* and is known to be a finite union of sets

$$[u_1 v_1^* u_2 \cdots u_t v_t^* u_{t+1}]_k \text{ for some } u_i, v_j \in \{0, \ldots, k-1\}^*$$

where $[\cdot]_k$ denote the k-ary valuation, [10]. Such languages, for instance, play a role in refinement of Christol's theorem [1].

Problem 1 (Reconstruction problem – static form). Consider the language (1). Randomly pick a word to guess $g = u_1 v_1^{\gamma_1} u_2 \cdots u_t v_t^{\gamma_t} u_{t+1} \in L$, that is, pick t random non-negative integers $\gamma_1, \ldots, \gamma_t$. The task is to uniquely characterize g from the knowledge of $\binom{g}{q}$ for some selected words q's in Σ^+. How many queries are sufficient? Does the number of queries depends on $|g|$ or $\gamma_1 + \cdots + \gamma_t$?

1.1 Our contributions

First, we show that the information provided by subword queries $\binom{g}{q}$ can be encoded into a system of polynomial equations. In Section 3, with Proposition 2, we show that each query translates into a polynomial identity whose unknowns are precisely the loop exponents in the underlying regular expression (1). The reconstruction task is thus transferred, from a problem in combinatorics on words, to the algebraic domain of Diophantine systems. Moreover, using a multivariate Newton expansion, we give an alternative combinatorial interpretation of the polynomials in Section 4.

Second, in Section 6, we solve the case of languages with two loops, i.e., Problem 1 with $t = 2$. By carefully analyzing the polynomial equations that arise in this setting, we exactly determine which queries are necessary and sufficient to guarantee unique reconstruction: two well-chosen queries are enough. This shows the feasibility of our method on a non-trivial but tractable class of languages. Indeed, this apparently quite simple case leads to a surprisingly non-trivial decision process in three steps (see Fig. 4). In particular, our analysis shows that combinatorial relations among the elements v_1, u_2, v_2 constituting (1) play a predominant role, e.g., commutation relation for k-binomial equivalence of words $v_1^\alpha u_2 \sim_k u_2 v_2^\beta$ for some integers α, β. Characterizing the solutions of such commutation relations is known to be challenging [30].

Third, in Section 7, we provide a certification mechanism for reconstruction in the general case $t \geq 3$. More specifically, with Proposition 7, we show how to decide whether a given set of queries always leads to a unique solution for all admissible words in (1). This certificate is independent of any particular instance and thus yields a structural guarantee of reconstructibility.

Finally, we address the algorithmic complexity of the approach. The certification step relies on techniques from real algebraic geometry [3, 4, 18], in particular quantifier elimination in the theory of real closed fields and Tarski–Seidenberg theorem [26]. While this procedure is in general exponential in the number of variables, on randomly generated examples, we were able to effectively check the reconstruction property with `Mathematica` for 3 loops and, in some cases, for 4 loops. This places the reconstruction problem within a decidable, though computationally demanding, framework. We have developed a `Mathematica` notebook[3] to generate equations and run the various procedures presented in this article. It also contains all the examples.

2 Background

For some general references on combinatorics on words and formal language theory, we refer the reader to [15, 24].

Let Σ be a finite alphabet, i.e., a finite set of elements called *letters*. We let Σ^* denote the free monoid generated by Σ with concatenation as product operation.

[3] available at `https://orbi.uliege.be/handle/2268/336473`, you may also download a standalone Wolfram Player.

The *empty word* is denoted by ε and $\Sigma^+ = \Sigma^* \setminus \{\varepsilon\}$. By $|w|$ we denote the *length* of w, i.e., the number of its letters and $|w|_a$ denotes the number of occurrences of the letter $a \in \Sigma$ in w. Further, $w[i]$ for $1 \leq i \leq |w|$ denotes the letter at the i^{th} index in w. If $w = uv$ for $u, v \in \Sigma^*$ then we write $u = wv^{-1}$ and $v = u^{-1}w$. A word u is a *subword* of a word $w \in \Sigma^*$ if there exist $v_1, \cdots, v_{|u|+1} \in \Sigma^*$ such that $w = v_1\, u[1]\, v_2\, u[2] \cdots v_{|u|}\, u[|u|]\, v_{|u|+1}$. For $\Sigma = \{a_1, \ldots, a_n\}$ for some $n \in \mathbb{N}$, the *Parikh vector* of a word $w \in \Sigma^*$ is given by $\Psi(w) = (|w|_{a_1}, \ldots, |w|_{a_n})$. For basics about combinatorics on words, we refer the reader to [15].

Recall that the *binomial coefficient* of the words g and q is defined as

$$\binom{g}{q} = \#\{i_1 < \cdots < i_{|q|} \mid g[i_1] \cdots g[i_{|q|}] = q\}$$

and counts the number of times q appears as a subword (understood as a subsequence, also called scattered subword) of g.

Let $k \geq 1$ be an integer. Two words x, y are *k-binomially equivalent* and we write $x \sim_k y$, whenever

$$\binom{x}{w} = \binom{y}{w}, \forall w \in \Sigma^{\leq k},$$

i.e., they share the same subwords of length at most k with the same multiplicities. For $k = 1$, this is exactly the abelian equivalence, i.e., the same Parikh vectors. We refer the reader to [22, 14] for details. As a consequence of [16], if x, y have length at least k, then $x \sim_k y$ if and only if $\binom{x}{w} = \binom{y}{w}$ for all $w \in \Sigma^k$. As an example, we have $0110 \sim_2 1001$ but $0110 \not\sim_3 1001$. It is easy to see that $\sim_k$ is a congruence: if $x \sim_k x'$ and $y \sim_k'$, then $xy \sim_k x'y'$. It is also cancellative: if $xy \sim_k xz$, then $y \sim_k z$.

3 A system of polynomial equations

When a word g belongs to a language of the form (1), binomial coefficients $\binom{g}{q}$ have a special form. Our aim is thus to introduce some useful polynomial mainly obtained from the Chu–Vandermonde identity. But first, let us consider some preliminary constructions. For $v \in \Sigma^+$, $u \in \Sigma^*$ and $i \in \mathbb{N} \setminus \{0\}$, we let

$$\alpha_i^{(v,u)} := \sum_{\substack{u = u_1 \cdots u_i \\ u_1, \ldots, u_i \in \Sigma^+}} \binom{v}{u_1} \cdots \binom{v}{u_i}.$$

These integers serve as coefficients of some univariate polynomials $B_{v,u}$. In particular, if $u = a_1 \cdots a_{|u|}$ with $a_k \in \Sigma$, for all k,

$$\alpha_{|u|}^{(v,u)} = \binom{v}{a_1} \cdots \binom{v}{a_{|u|}} \quad \text{and} \quad \alpha_1^{(v,u)} = \binom{v}{u}.$$

Note that $\alpha_i^{(v,u)} = 0$ when $i > |u|$.

Let x be a variable. Let $i \in \mathbb{N} \setminus \{0\}$. The expression $\binom{x}{i}$ can be seen as a polynomial in x of degree i with rational coefficients:

$$\binom{x}{i} = \frac{1}{i!} x(x-1)\cdots(x-i+1) = \frac{x^{\underline{i}}}{i!}$$

where we use the *falling factorial* notation [11].

Definition 1. *Let $v \in \Sigma^+$. We set $B_{v,\varepsilon}(x) = 1$ and, for $u \neq \varepsilon$, we define a polynomial in $\mathbb{Q}[x]$ of degree at most $|u|$ by*

$$B_{v,u}(x) := \sum_{i=1}^{|u|} \frac{\alpha_i^{(v,u)}}{i!}\, x^{\underline{i}}.$$

If $\alpha_{|u|}^{(v,u)} \neq 0$, then $\deg(B_{v,u}) = |u|$.

Remark 1. Note that $B_{v,u}(0) = 0$ for all $v, u \in \Sigma^+$.

Lemma 1. *For $v, u \in \Sigma^+$ and $n \in \mathbb{N}$, we have*

$$\binom{v^n}{u} = B_{v,u}(n).$$

Proof. To count the number of occurrences of u as a subword of v^n, we first choose j copies (with j between 1 and $\min\{|u|, n\}$) of the word v such that u is obtained by concatenating j non-empty subwords contained in the selected copies. There are exactly $\binom{n}{j}$ ways to choose the j copies of v in v^n. Hence

$$\binom{v^n}{u} = \sum_{j=1}^{\min\{|u|,n\}} \binom{n}{j} \sum_{\substack{u=u_1\cdots u_j \\ u_1,\ldots,u_j \in \Sigma^+}} \binom{v}{u_1} \cdots \binom{v}{u_j} = \sum_{j=1}^{|u|} \binom{n}{j} \alpha_j^{(v,u)}.$$

For the last equality, observe that if $n < |u|$, then $\binom{n}{j} = 0$ when $j > n$. □

Any polynomial relationship among binomial coefficients of words can be exported to these polynomials.

Lemma 2. *Let $u_1, \ldots, u_k \in \Sigma^+$ and $Q[x_1, \ldots, x_k]$ be a polynomial. For all $v \in \Sigma^+$, we have*

$$\left[\forall w \in \Sigma^+, Q\left(\binom{w}{u_1}, \ldots, \binom{w}{u_k}\right) = 0\right] \Rightarrow Q(B_{v,u_1}(x), \ldots, B_{v,u_k}(x)) = 0.$$

Proof. If the words u_j's have length $\leq d$ and Q has total degree n, then

$$Q(B_{v,u_1}(x), \ldots, B_{v,u_k}(x))$$

is a univariate polynomial of degree at most nd. Using Lemma 1, evaluation at $x = 1, 2, \ldots, nd+1$ gives $Q(x) = Q\left(\binom{v^x}{u_1}, \ldots, \binom{v^x}{u_k}\right) = 0$ by assumption. As a corollary of the fundamental theorem of algebra, $Q = 0$ identically. □

As an example, $\binom{w}{0}\binom{w}{1} = \binom{w}{01} + \binom{w}{10}$. Hence, $B_{v,10} = B_{v,0}B_{v,1} - B_{v,01}$. More relations of this form can be found in [19].

3.1 Building polynomial equations

Since the language (1) has a special form, we obtain a system of polynomial equations built from the polynomials B's discussed above, with the aim to uniquely determine the word to guess. The number of unknowns is equal to t and the degree of a polynomial equation is equal to the length of the corresponding query. Recall that the unknown word picked in the language is g and we may ask an "oracle" the value of $\binom{g}{q}$ for a query q of our choice.

Let us recall the Chu–Vandermonde identity [15, Cor. 6.3.7].

Proposition 1. *For all $w, z, u \in \Sigma^*$, we have*

$$\binom{wz}{u} = \sum_{\substack{e,f \in A^* \\ u=ef}} \binom{w}{e}\binom{z}{f}.$$

Using Chu–Vandermonde identity, a query with $q \in \Sigma^+$ yields

$$\begin{aligned}\binom{g}{q} &= \sum_{\substack{q=e_1f_1e_2\cdots e_tf_te_{t+1} \\ e_i,f_i \in \Sigma^*}} \binom{u_1}{e_1}\binom{v_1^{x_1}}{f_1}\binom{u_2}{e_2}\cdots\binom{u_t}{e_t}\binom{v_t^{x_t}}{f_t}\binom{u_{t+1}}{e_{t+1}} \\ &= \sum_{\substack{q=e_1f_1e_2\cdots e_tf_te_{t+1} \\ e_i,f_i \in \Sigma^*}} \prod_{j=1}^{t+1}\binom{u_j}{e_j}\prod_{j=1}^{t} B_{v_j,f_j}(x_j). \qquad (2)\end{aligned}$$

where the r.h.s. (2) is a multivariate polynomial in $x_1, \ldots, x_t$. The degree of a monomial (i.e., the sum of the exponents of all the variables within that monomial) in $\prod_{j=1}^{t} B_{v_j,f_j}(x_j)$ is at most $|f_1| + \cdots + |f_t| \leq |q|$. Note that the r.h.s. does not depend on g but only on the query q. In particular, the constant term in the r.h.s. of (2) is given by

$$\sum_{\substack{q=e_1e_2\cdots e_{t+1} \\ e_i \in \Sigma^*}} \prod_{j=1}^{t+1}\binom{u_j}{e_j}.$$

Indeed, if f_j is non-empty, then $B_{v_j,f_j}(x_j)$ either vanishes or, is a polynomial in x_j with a zero constant term (see Remark 1). To emphasize the fact that the r.h.s. depends only on the language and the query q, it deserves a notation

$$\mathsf{P}_q(x_1, \ldots, x_t) := \sum_{\substack{q=e_1f_1e_2\cdots e_tf_te_{t+1} \\ e_i,f_i \in \Sigma^*}} \prod_{j=1}^{t+1}\binom{u_j}{e_j}\prod_{j=1}^{t} B_{v_j,f_j}(x_j). \qquad (3)$$

When the query is a single letter $a \in \Sigma$, we have

$$\mathsf{P}_a(x_1, \ldots, x_t) = \sum_{i=1}^{t} |v_i|_a\, x_i + |u_1u_2\cdots u_{t+1}|_a. \qquad (4)$$

Remark 2. Let $M_{q,w}$ be the generalized Parikh matrix associated with $q = q_1 \cdots q_\ell$ of the word $w = w_1 \cdots w_n$, where q_i's and w_j's are letters. By definition, for $a \in \Sigma$, $M_{q,a}$ is a unitriangular square matrix of size $|q|+1$ (with 1's on the diagonal) such that $[M_{q,a}]_{i,i+1} = 1$ if and only if $q_i = a$. Hence $M_{q,w}$ is defined as $M_{q,w_1} \cdots M_{q,w_n}$ and the upper-right corner of this matrix contains $\binom{w}{q}$. Then, the polynomial P_q can be obtained as

$$\left[M_{q,u_1}(M_{q,v_1})^{x_1} M_{q,u_2} \cdots (M_{q,v_t})^{x_t} M_{q,u_{t+1}}\right]_{1,|q|+1}.$$

For details, see [5].

Definition 2. *Let $q \in \Sigma^+$ and g in (1). Consider the polynomial in $\mathbb{Q}[x_1, \ldots, x_t]$*

$$\mathsf{Eq}_{g,q} := \mathsf{P}_q(x_1, \ldots, x_t) - \binom{g}{q}.$$

We say that $\mathsf{Eq}_{g,q}(x_1, \ldots, x_t) = 0$ is the equation associated with the query q.

The following proposition should be obvious.

Proposition 2. *Let $g = u_1 v_1^{\gamma_1} u_2 \cdots u_t v_t^{\gamma_t} u_{t+1}$ be in (1). For all $q \in \Sigma^+$, the t-uple $(\gamma_1, \ldots, \gamma_t)$ of non-negative integers is a solution of the equation $\mathsf{Eq}_{g,q} = 0$.*

Example 1. We emphasize here that the polynomial P_q only depends on q and the equation $\mathsf{Eq}_{g,q}$ also depends on g. Consider the language $0(001)^*110(1001)^*0$. For convenience, we write x, y instead of x_1, x_2. With the query $q = 01$, we get the polynomial $\mathsf{P}_{01}(x,y) = x^2 + 4xy + 6x + 2y^2 + 4y + 2$. Now if the word g is of the form $0(001)^{\gamma_1}110(1001)^{\gamma_2}0$, the corresponding curves of equation $\mathsf{Eq}_{g,01} = 0$ are depicted in Figure 2 for (γ_1, γ_2) equal to $(3,0)$, $(2,2)$, $(3,2)$ and $(0,3)$.

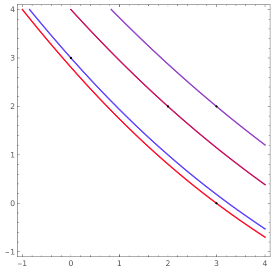

Fig. 2. Curves $\mathsf{Eq}_{g,10} = 0$ for $(\gamma_1, \gamma_2) \in \{(3,0), (2,2), (3,2), (0,3)\}$.

Example 2. We present a first resolution of a system. Consider $t = 2$, $u_1 = 0$, $v_1 = 10010$, $u_2 = 010$, $v_2 = 010$ and $u_3 = \varepsilon$. With the word $q_1 = 0$, we get the polynomial $\mathsf{P}_0(x,y) = 3x + 2y + 3$. With $q_2 = 10$, we get

$$\mathsf{P}_{10}(x,y) = 3x^2 + 4xy + 5x + y^2 + 2y + 1 \tag{5}$$

Finally, with $q_3 = 011$, we get

$$\mathsf{P}_{011}(x,y) = 2x^3 + 3x^2y + \frac{5x^2}{2} + \frac{3xy^2}{2} + \frac{5xy}{2} + \frac{x}{2} + \frac{y^3}{3} + y^2 + \frac{2y}{3}.$$

Now assume that the word to guess is $g = 0(10010)^4010(010)^3$. The three queries gives respectively,

$$\binom{g}{q_1} = 21, \binom{g}{q_2} = 132, \text{ and } \binom{g}{q_3} = 418.$$

The system of two polynomial equations

$$\begin{cases} \mathsf{Eq}_{g,0} \ : & 3x + 2y - 18 = 0 \\ \mathsf{Eq}_{g,10} : & 3x^2 + 4xy + 5x + y^2 + 2y - 131 = 0 \end{cases}$$

has two solutions $(x, y) = (4, 3)$ or $(32/7, -7)$ but only one has non-negative integer entries. Adding the third query provides a system with the unique solution $(4, 3)$ as can be seen in Figure 3.

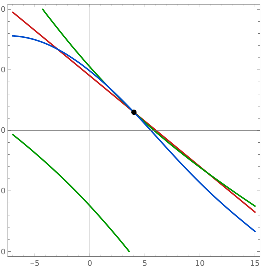

Fig. 3. Curves $\mathsf{Eq}_{g,q} = 0$ for $q_1 = 0$ (red), $q_2 = 10$ (green) and $q_3 = 011$ (blue).

Remark 3. The example given above could be misleading, the coefficients of P_q are not necessarily non-negative rational numbers. All of this depends on the cancellations that can occur when we expand the falling factorials occurring in the B_{v_j,f_j}'s. Continuing Example 2, one can check that $\mathsf{P}_{0111}(x, y)$ contains several negative coefficients $-\frac{x^2}{2}$, $-\frac{xy}{2}$, $-\frac{x}{6}$, $-\frac{y^2}{12}$ and $-\frac{y}{6}$.

3.2 Chen–Fox–Lyndon relations

There exist some dependencies among the polynomials P_q's. This result is related to the famous Chen–Fox–Lyndon relation presented for binomial coefficients of words in [15, Prop. 6.3.25]. For $h, r, s \in \Sigma^*$ we have

$$\binom{h}{r}\binom{h}{s} = \sum_{w \in \Sigma^*} \langle r \uparrow s, w \rangle \binom{h}{w}$$

where $u \uparrow v$ is the infiltration product of the words u and v and $\langle u \uparrow v, w\rangle$ denotes the coefficient of the word w in this formal polynomial. As an example, since $01 \uparrow 0 = 2 \cdot 001 + 010 + 01$, using the next proposition, we get

$$\binom{h}{01}\binom{h}{0} = 2\binom{h}{001} + \binom{h}{010} + \binom{h}{01} \quad \text{and} \quad \mathsf{P}_{01}.\mathsf{P}_0 = 2\mathsf{P}_{001} + \mathsf{P}_{010} + \mathsf{P}_{01}.$$

Proposition 3. *Let $L = u_1 v_1^* u_2 \cdots u_t v_t^* u_{t+1}$ be a polynomial regular language. For all $r, s \in \Sigma^+$, with notation* (3)*, we have*

$$\mathsf{P}_r \cdot \mathsf{P}_s = \sum_{w \in \Sigma^*} \langle r \uparrow s, w\rangle \mathsf{P}_w.$$

Proof. Let $(m_1, \dots, m_t) \in \mathbb{N}^t$. For all q, we have

$$\mathsf{P}_q(m_1, \dots, m_t) = \binom{u_1 v_1^{m_1} \cdots u_t v_t^{m_t} u_{t+1}}{q}.$$

We may apply Chen–Fox–Lyndon relation and get

$$\binom{u_1 v_1^{m_1} \cdots v_t^{m_t} u_{t+1}}{r}\binom{u_1 v_1^{m_1} \cdots v_t^{m_t} u_{t+1}}{s} = \sum_{w \in \Sigma^*} \langle r \uparrow s, w\rangle \binom{u_1 v_1^{m_1} \cdots v_t^{m_t} u_{t+1}}{w}.$$

This means that, for all $(m_1, \dots, m_t) \in \mathbb{N}^t$,

$$(\mathsf{P}_r \cdot \mathsf{P}_s)(m_1, \dots, m_t) = \sum_{w \in \Sigma^*} \langle r \uparrow s, w\rangle \mathsf{P}_w(m_1, \dots, m_t).$$

We conclude using the following result[4] (Schwartz–Zippel Lemma, also referred to as *polynomial principle*). Let $\mathbb{F}$ be an infinite field, $d \geq 0$, and $A_1, \dots, A_n \subset \mathbb{F}$ be sets such that $\#A_i = d + 1$ for all $i \in \{1, \dots, n\}$. If a polynomial $P \in \mathbb{F}[x_1, \dots, x_n]$ has total degree $\leq d$ and satisfies

$$P(a_1, \dots, a_n) = 0 \quad \text{for every } (a_1, \dots, a_n) \in A_1 \times \cdots \times A_n,$$

then P is identically 0. □

As a consequence of this result, it is thus enough to consider Lyndon words as queries. Any polynomial equation can be obtained as a combination of equations associated with Lyndon words. See [20, Thm. 6.4].

4 An alternative expression for the polynomials

Since the polynomials P_q play a central role in our developments, we present an alternative way to obtain them. We take into account Remark 3 and get an

[4] For a multivariate polynomial P having infinitely many zeros is not enough to imply $P = 0$ identically. As an example, the polynomial xy has infinitely many zeros.

expression with a combinatorial explanation. Let $f(x_1,\dots,x_t)\in\mathbb{Q}[x_1,\dots,x_t]$. For each multi-index $\mathbf{k}=(k_1,\dots,k_t)\in\mathbb{N}^t$, define

$$\binom{\mathbf{x}}{\mathbf{k}} = \binom{x_1}{k_1}\binom{x_2}{k_2}\cdots\binom{x_t}{k_t}$$

and the *forward-difference operators*

$$\Delta_{x_i} f(\mathbf{x}) = f(x_1,\dots,x_i+1,\dots,x_t) - f(x_1,\dots,x_i,\dots,x_t),$$

with the iterated difference $\Delta^{\mathbf{k}} = \Delta_{x_1}^{k_1}\circ\cdots\circ\Delta_{x_t}^{k_t}$.

Theorem 1 (Multivariate Newton Expansion). *Let $f(\mathbf{x})\in\mathbb{Q}[x_1,\dots,x_t]$. There is a unique expansion*

$$f(\mathbf{x}) = \sum_{\mathbf{k}\in\mathbb{N}^t} a_{\mathbf{k}}\binom{\mathbf{x}}{\mathbf{k}},\quad a_{\mathbf{k}}\in\mathbb{Q},$$

given by

$$a_{\mathbf{k}} = \big(\Delta^{\mathbf{k}} f\big)(0,\dots,0).$$

Moreover, $f(\mathbb{N}^t)\subseteq\mathbb{N}$ if and only if $a_{\mathbf{k}}\in\mathbb{N}$ for all $\mathbf{k}$.

We may apply this theorem to (2).

Proposition 4. *A query with $q\in\Sigma^+$ yields*

$$\mathsf{P}_q(x_1,\dots,x_t) = \sum_{\substack{\mathbf{k}\in\mathbb{N}^t\\ k_1+\cdots+k_t\le|q|}} a_{\mathbf{k}}\binom{\mathbf{x}}{\mathbf{k}}$$

where $a_{\mathbf{k}}$ is equal to

$$\sum_{\substack{q=e_1f_{1,1}\cdots f_{1,k_1}e_2\cdots e_tf_{t,1}\cdots f_{t,k_t}e_{t+1}\\ e_i,f_{i,j}\in\Sigma^*\\ f_{i,j}\neq\varepsilon \text{ if } k_i>0}} \binom{u_1}{e_1}\binom{v_1}{f_{1,1}}\cdots\binom{v_1}{f_{1,k_1}}\binom{u_2}{e_2} \cdots\binom{u_t}{e_t}\binom{v_t}{f_{t,1}}\cdots\binom{v_t}{f_{t,k_t}}\binom{u_{t+1}}{e_{t+1}}.$$

Proof. The combinatorial interpretation is the following one, the result follows from the uniqueness of the expansion. Fix $\mathbf{k}=(k_1,\dots,k_t)\in\mathbb{N}^t$ be such that $k_1+\cdots+k_t\le|q|$. With this choice, we may build particular occurrences of q appearing as a subword of g. Pick k_j non-empty subwords in k_j distinct copies of v_j. These subwords are denoted by $f_{j,1},\dots,f_{j,k_j}$. This correspond to the factorization

$$q = e_1f_{1,1}\cdots f_{1,k_1}e_2\cdots e_tf_{t,1}\cdots f_{t,k_t}e_{t+1}$$

where e_j is a (possibly empty) subword of u_j. The word g contains x_j copies of v_j, we thus have to choose k_j copies among v_j. Hence the number of this particular type of subwords occurring in g is given by $a_{\mathbf{k}}\binom{\mathbf{x}}{\mathbf{k}}$. To get $\binom{g}{q}$, one has to sum over all possible t-uples $\mathbf{k}$. □

Example 3. We continue Example 2. The polynomial (5) can be expressed as

$$8\binom{x}{1}+6\binom{x}{2}+3\binom{y}{1}+4\binom{x}{1}\binom{y}{1}+2\binom{y}{2}+1.$$

5 Unique reconstruction versus unique solution

Several regular expressions may describe a language. We can always assume that if u_{i+1} is non-empty, then v_i and u_{i+1} start with distinct letters. Indeed, if a is the first letter of both v_i and u_{i+1}, then

$$v_i^* u_{i+1} = a(a^{-1}v_i a)^* a^{-1} u_{i+1}.$$

One can iterate this transformation until the first letters in v_i and u_{i+1} differ or, u_{i+1} is empty. In particular, if u_{i+1} is a prefix of v_i, then $v_i^* u_{i+1} = u_{i+1}(u_{i+1}^{-1} v_i u_{i+1})^*$. Also, the number t of loops occurring in (1) can be assumed to be minimal. One has to built the minimal automaton of (1) which is unique up to isomorphism, to obtain a convenient regular expression that we said to be *reduced.* As an example, the language $(01)^*(0101)^*$ is actually replaced by $(01)^*$.

Remark 4. When the word to reconstruct does not use all the loops, the situation can be more complicated. As an example, take the language

$$L = (011)^*(1001)^*011(011)^*.$$

Its minimal automaton has 3 loops. Let $g = (011)^{\gamma_1}(1001)^{\gamma_2}011(011)^{\gamma_3} \in L$. If $\gamma_2 = 0$, the word g belongs to $(011)^*011(011)^* = 011(011)^*$. So any equation $\mathsf{Eq}_{g,q}$ having $(\gamma_1, 0, \gamma_3)$ as solution also has $(i, 0, \gamma_1+\gamma_3-i)$ for $i = 0, \ldots, \gamma_1+\gamma_3$ as solution. However, all these solutions correspond to the same word $(001)^{\gamma_1+\gamma_3+1}$. So there is a distinction between searching for a unique word to reconstruct and a system of equations having a unique solution.

When a word to reconstruct is generated, there are 2^t ways to choose when the γ_i's are either zero or positive. If for all $z \in \{0, \ldots, t-1\}$ and for all indices $1 \leq j_1 < \cdots < j_z \leq t$, the minimal automaton of the language (1) where $v_{j_1}, \ldots, v_{j_z}$ are set to the empty word has $t-z$ loops, then L is said to be *full.* For instance, the language in Remark 4 is not full.

6 The surprisingly interesting case of two loops

Assume $L = u_1 v_1^* u_2 v_2^* u_3$ is in reduced form with $v_1, v_2 \in \Sigma^+$, so $t = 2$ in Problem 1. The word to guess is of the form $g = u_1 v_1^x u_2 v_2^y u_3$. Since we only have two variables, for the sake of notation, we use x, y instead of x_1, x_2. The polynomial equation (2) becomes

$$\binom{g}{q} = \sum_{\substack{q=e_1 f_1 e_2 \cdots e_t f_t e_{t+1} \\ e_i, f_i \in \Sigma^*}} \left(\prod_{j=1}^{3} \binom{u_j}{e_j} \right) B_{v_1,f_1}(x) B_{v_2,f_2}(y). \qquad (6)$$

As already observed in (4), if $|q| = 1$, i.e., $q = a \in \Sigma$, then the above equation is simply

$$\binom{g}{a} = |v_1|_a x + |v_2|_a y + |u_1 u_2 u_3|_a \tag{7}$$

which is a polynomial of degree at most 1.

We let $\Psi(w)$ denote the Parikh vector of the word w where the alphabet Σ is assumed to be totally ordered. Figure 4 gives the structure (or, even a decision diagram) of the results.

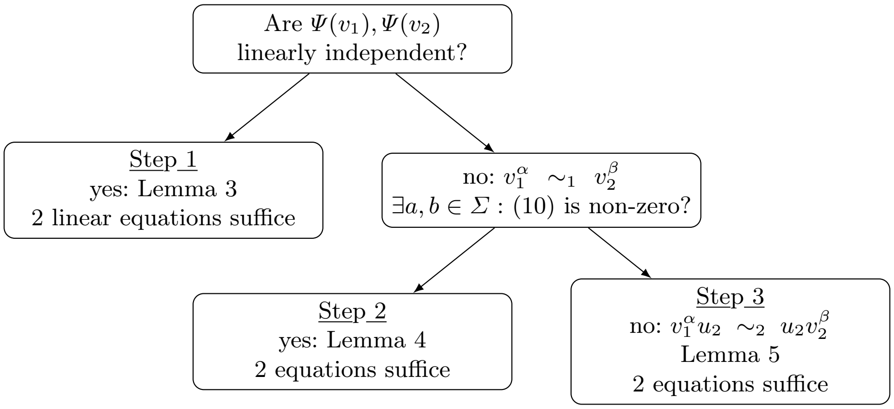
Are $\Psi(v_1), \Psi(v_2)$ linearly independent?
Step 1
yes: Lemma 3
2 linear equations suffice
no: $v_1^\alpha \sim_1 v_2^\beta$
$\exists a, b \in \Sigma$: (10) is non-zero?
Step 2
yes: Lemma 4
2 equations suffice
Step 3
no: $v_1^\alpha u_2 \sim_2 u_2 v_2^\beta$
Lemma 5
2 equations suffice

Fig. 4. Synopsis

Step 1. We start with an easy situation: we begin by considering the case of two independent Parikh vectors. This allows us, in the subsequent steps, to assume the alternative situation.

Lemma 3. *Let $L = u_1 v_1^* u_2 v_2^* u_3$ be a language where $v_1, v_2 \in \Sigma^+$ and $u_1, u_2, u_3 \in \Sigma^*$ are given. If the Parikh vectors $\Psi(v_1), \Psi(v_2)$ are linearly independent over $\mathbb{Q}$, then a word in L (of unknown length) can be uniquely reconstructed with two queries (of length 1).*

Proof. Since $\Psi(v_1), \Psi(v_2)$ are linearly independent over $\mathbb{Q}$, there exist two letters $a, b \in \Sigma$ such that

$$\det \begin{pmatrix} |v_1|_a & |v_2|_a \\ |v_1|_b & |v_2|_b \end{pmatrix} \neq 0.$$

Hence, two queries for $q = a$ and $q = b$ gives for (7) a system of two linear equations having a unique solution (by Cramer's rule):

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} |v_1|_a & |v_2|_a \\ |v_1|_b & |v_2|_b \end{pmatrix}^{-1} \begin{pmatrix} |g|_a - |u_1 u_2 u_3|_a \\ |g|_b - |u_1 u_2 u_3|_b \end{pmatrix}$$

Geometrically, the two equations are describing in the plane $\mathbb{R}^2$, two straight lines of distinct slope $-|v_1|_a/|v_2|_a$ and $-|v_1|_b/|v_2|_b$ respectively. By usual convention, when $|v_2|_a = 0$ (resp. $|v_2|_b = 0$), this corresponds to a vertical line. □

We can mention here the direct extension to more loops.

Proposition 5. *Let $L = u_1 v_1^* u_2 \cdots u_t v_t^* u_{t+1}$ with $v_1, \ldots, v_t \in \Sigma^+$ be given. If $\Psi(v_1), \ldots, \Psi(v_t)$ are linearly independent, then any word in L (of unknown length) can be uniquely reconstructed with t queries.*

Remark 5. The same reasoning as in the previous proof applies. In this case, the assumption implies that $\#\Sigma \geq t$. Then t queries with well-chosen letters give equations of t hyperplanes of $\mathbb{R}^t$ intersecting in exactly one point and the word to guess can be uniquely reconstructed.

Step 2. From now on, we assume that there exist coprime integers $\alpha, \beta \geq 1$ such that $\alpha \Psi(v_1) = \beta \Psi(v_2)$. This can be written as

$$v_1^\alpha \sim_1 v_2^\beta \tag{8}$$

i.e., these powers of v_1 and v_2 are abelian equivalent.

In the situation we are now dealing with, for any two letters a, b appearing in v_1 (and thus in v_2), (7) yields the same equation $\mathsf{Eq}_{g,a}(x, y) = 0$ which is

$$|v_1|_a x + |v_2|_a y + |u_1 u_2 u_3|_a = \binom{g}{a}$$

and can be rewritten as

$$y = -\frac{\beta}{\alpha} x + \frac{|v_1^{\gamma_1} v_2^{\gamma_2}|_a}{|v_2|_a}. \tag{9}$$

Geometrically, the two straight lines $\mathsf{Eq}_{g,a}(x, y) = 0$ and $\mathsf{Eq}_{g,b}(x, y) = 0$ have the same slope

$$-|v_1|_a/|v_2|_a = -|v_1|_b/|v_2|_b = -\beta/\alpha$$

and thanks to Proposition 2, we know that they share a common point: they coincide. So all queries with a single letter lead to the same linear equation (or, to an empty equation when the letter does not appear in v_1).

Lemma 4. *Let $L = u_1 v_1^* u_2 v_2^* u_3$ be a language where $v_1, v_2 \in \Sigma^+$ and $u_1, u_2, u_3 \in \Sigma^*$ are given and $v_1^\alpha \sim_1 v_2^\beta$ for some coprime integers $\alpha, \beta \geq 1$. If there exist $a, b \in \Sigma$ such that the quantity*

$$|v_1^\alpha|_b |u_2|_a - |v_1^\alpha|_a |u_2|_b + \binom{v_2^\beta}{ab} - \binom{v_1^\alpha}{ab} \tag{10}$$

is non-zero, then a word in L (of unknown length) can be uniquely reconstructed with two queries (of length 1 and 2 respectively).

Proof. Our aim is to show that an extra equation with a query of length 2 is indeed enough.

• Assume first that two distinct letters a, b appear in v_1 (and thus also in v_2 because the Parikh vectors are linearly dependent). Again (6) with $q = ab$ becomes here

$$\begin{aligned}\binom{g}{ab} &= B_{v_1,ab}(x) + B_{v_2,ab}(y) + B_{v_1,a}(x)B_{v_2,b}(y)\\ &+ B_{v_1,a}(x)\binom{u_2u_3}{b} + \binom{u_1}{a}B_{v_1,b}(x)\\ &+ B_{v_2,a}(y)\binom{u_3}{b} + \binom{u_1u_2}{a}B_{v_2,b}(y) + \binom{u_1u_2u_3}{ab}. \qquad (11)\end{aligned}$$

A term of degree 2 may come, for $i = 1, 2$ and $z = x, y$, from

$$B_{v_i,ab}(z) = |v_i|_a|v_i|_b\binom{z}{2} + \binom{v_i}{ab}z = \frac{|v_i|_a|v_i|_b}{2}z^2 + \left[\binom{v_i}{ab} - \frac{|v_i|_a|v_i|_b}{2}\right]z$$

or, from $B_{v_1,a}(x)B_{v_2,b}(y) = |v_1|_a|v_2|_b xy$. We can thus write

$$\begin{aligned}0 = \begin{pmatrix}x \ y\end{pmatrix} M \begin{pmatrix}x\\ y\end{pmatrix} + \underbrace{\left(|v_1|_a\binom{u_2u_3}{b} + \binom{u_1}{a}|v_1|_b + \binom{v_1}{ab} - \frac{|v_1|_a|v_1|_b}{2}\right)}_{=:2D} x\\ + \underbrace{\binom{u_1u_2u_3}{ab} - \binom{g}{ab}}_{=:F} + \underbrace{\left(|v_2|_a\binom{u_3}{b} + \binom{u_1u_2}{a}|v_2|_b + \binom{v_2}{ab} - \frac{|v_2|_a|v_2|_b}{2}\right)}_{=:2E} y\end{aligned} \qquad (12)$$

where, setting $\mu := \alpha/\beta$, the matrix for the homogeneous quadratic part is

$$M = \frac{1}{2}\begin{pmatrix}|v_1|_a|v_1|_b & |v_1|_a|v_2|_b\\ |v_1|_a|v_2|_b & |v_2|_a|v_2|_b\end{pmatrix} = \frac{|v_1|_a|v_1|_b}{2}\begin{pmatrix}1 & \mu\\ \mu & \mu^2\end{pmatrix}$$

and using the fact that $|v_2|_c = \mu|v_1|_c$ for $c \in \{a, b\}$. The determinant of M is zero. We conclude that (11) is the equation of a parabola. See, for instance, [17]. We still have to determine whether or not this conic is degenerate.

Consider the 3×3 matrix Q whose upper-left corner is M and whose last row and column consist of $\begin{pmatrix}D \ E \ F\end{pmatrix}$. Because of the relation existing among the rows (or equivalently, columns) of M, the determinant of Q is equal to

$$-\frac{|v_1|_a|v_1|_b}{2}(E - \mu D)^2 = -\frac{|v_1|_a|v_1|_b}{2\beta^2}(\beta E - \alpha D)^2.$$

For instance, subtract μ times the first row from the second row. Since we are interested to know when $\det Q = 0$, we only have to look at the factor

$$2(\beta E - \alpha D) = |v_1^\alpha|_b|u_2|_a - |v_1^\alpha|_a|u_2|_b + \binom{v_2^\beta}{ab} - \binom{v_1^\alpha}{ab}.$$

The above equality comes from

$$\binom{v_1^\alpha}{ab} = \alpha\binom{v_1}{ab} + \binom{\alpha}{2}|v_1|_a|v_1|_b$$

indeed one can find ab within one of the α blocks v_1 or, choose two blocks v_1 among α and pick a in the first block and b in the other one. Replacing $\binom{\alpha}{2}$ with $\alpha(\alpha-1)/2$, we get

$$\alpha\left(\binom{v_1}{ab} - \frac{|v_1|_a|v_1|_b}{2}\right) = \binom{v_1^\alpha}{ab} - \frac{\alpha^2}{2}|v_1|_a|v_1|_b$$

and a similar computation is done for $\binom{v_2^\beta}{ab}$. We have a degenerate parabola if and only if $\det Q = 0$. This is equivalent to the fact that (10) is zero. Since a degenerate parabola is made of two parallel lines, by Proposition 2, one of these lines has equation $\mathsf{Eq}_{g,a}(x,y) = 0$. This means that $\mathsf{Eq}_{g,ab}(x,y)$ has $\mathsf{Eq}_{g,a}(x,y)$ as a factor.

With the assumption that there exist a, b such that (10) is non-zero, we are in the non-degenerate case. The eigenvalues of M are 0 and μ^2+1 with respective orthogonal eigenvectors $\begin{pmatrix}1 & -1/\mu\end{pmatrix}^\top$ and $\begin{pmatrix}1 & \mu\end{pmatrix}^\top$. Recall that the parabola extends in the direction of the eigenvector corresponding to the zero eigenvalue. Hence a straight line with slope $-1/\mu$ intersects the parabola exactly once. Consequently the queries $q = a$ and $q = ab$ are enough to uniquely determine g.

• We have also to deal with the case where $v_1 = a^r$ and $v_2 = a^s$ for some $r, s > 0$. If $u_2 \in \{a\}^*$, then $\binom{g}{a}$ gives the total number of a's in the word g and this is enough for unique reconstruction (even though the regular expression $(a^r)^*u_2(a^s)^*$ is not reduced). The system may have several solutions but they all provide the same word. Now assume that u_2 contains a letter b distinct from a. Hence, the quantity (10) reduces to $-|v_1^\alpha|_a|v_2|_b|$ and is non-zero. □

Example 4. If we consider the same language as in Example 2, $\Psi(v_1)$ and $\Psi(v_2)$ are linearly independent. The queries with 0 and 1 are enough: the two linear equations $3x + 2y - 18 = 0$ and $2x + y - 11 = 0$ are depicted in red and yellow resp. on the left of Figure 5.

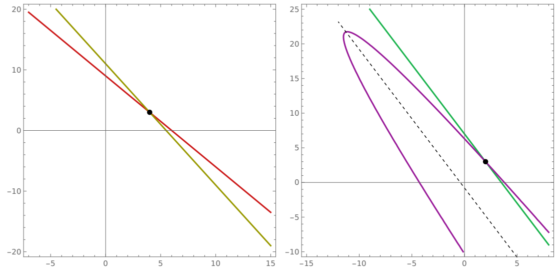

Fig. 5. The cases where the Parikh vectors are linearly independent or dependent.

If we consider instead $v_1 = 110000$ and $v_2 = 001$. The queries with 0 and 1 give the same equation $2x + y - 7 = 0$. The query with 01 gives the parabola of equation $4x^2 + 4xy + 2x + y^2 + 4y - 65 = 0$. Both are depicted on the right of Figure 5.

Remark 6 (Combinatorial interpretation of the condition). Observe that when (10) is zero for all $a, b \in \Sigma$, two cases may occur: the two main terms contributing to (10) are both zero or, they are opposite and compensate each other.

So, either, for all $a, b \in \Sigma$,

$$\binom{v_1^\alpha}{ab} = \binom{v_2^\beta}{ab} \quad \text{and} \quad |v_1^\alpha|_b |u_2|_a = |v_1^\alpha|_a |u_2|_b.$$

In particular, the l.h.s. equality means that $v_1^\alpha \sim_2 v_2^\beta$ and the r.h.s. equality means that $\Psi(v_1^\alpha)$ and $\Psi(u_2)$ are linearly dependent. So, in that case, $v_1^\alpha \sim_2 v_2^\beta$ if and only if $\Psi(v_1^\alpha)$ and $\Psi(u_2)$ are linearly dependent.

Or, there exist letters $a, b \in \Sigma$ such that

$$|v_1^\alpha|_b |u_2|_a - |v_1^\alpha|_a |u_2|_b = \binom{v_1^\alpha}{ab} - \binom{v_2^\beta}{ab} \neq 0.$$

Proposition 6. *Let $L = u_1 v_1^* u_2 v_2^* u_3$ be a language where $v_1, v_2 \in \Sigma^+$ and $u_1, u_2, u_3 \in \Sigma^*$ are given and $v_1^\alpha \sim_1 v_2^\beta$ for some coprime integers $\alpha, \beta \geq 1$.*

Let $k \geq 2$. Assume that the word $g = u_1 v_1^{\gamma_1} u_2 v_2^{\gamma_2} u_3$ to guess is long enough to guarantee that $\lfloor \gamma_1/\alpha \rfloor + \lfloor \gamma_2/\beta \rfloor \geq k$. All queries q such that $2 \leq |q| \leq k$ do not provide any more constraints on the set of solutions than the linear equation provided by letter a occurring in v_1, i.e., the two systems

$$\{\mathsf{Eq}_{g,a} = 0\} \text{ and } \{\mathsf{Eq}_{g,q} = 0 \mid 1 \leq |q| \leq k\}$$

have the same sets of solutions, if and only if $v_1^\alpha u_2 \sim_k u_2 v_2^\beta$.

Proof. $\Leftarrow$ The condition is sufficient. Assume that $v_1^\alpha u_2 \sim_k u_2 v_2^\beta$. Let $N \geq 0$. With the assumption about k-binomial equivalence, we directly have

$$(v_1^\alpha)^i u_2 (v_2^\beta)^{N-i} \sim_k u_2 (v_2^\beta)^N, \quad \forall i \in \{0, \ldots, N\}.$$

Let $m_1, m_2 \geq 0$, $r_1 < \alpha$ and $r_2 < \beta$ be integers such that $m_1 + m_2 \geq k$. The latter condition will become clear in the next few lines. Since k-binomial equivalence is a congruence and using the above relation with $N = m_1 + m_2$, we have

$$W_i := u_1 v_1^{r_1} (v_1^\alpha)^i u_2 (v_2^\beta)^{m_1+m_2-i} v_2^{r_2} u_3 \sim_k u_1 v_1^{r_1} u_2 (v_2^\beta)^{m_1+m_2} v_2^{r_2} u_3 =: W_0$$

for all $i \in \{0, \ldots, m_1 + m_2\}$.

Let q be a word of length at most k. By (3), we have

$$\mathsf{P}_q(i\,\alpha + r_1, (m_1 + m_2 - i)\,\beta + r_2) = \binom{W_i}{q} = \binom{W_0}{q}.$$

This means that the polynomial $\mathsf{P}_q(x,y)$ takes the same value on m_1+m_2+1 points belonging to the line of equation

$$y=-\frac{\beta}{\alpha}x+\beta\left(m_1+m_2+\frac{r_1}{\alpha}\right)+r_2. \tag{13}$$

The reader may notice that the word W_0 only depends on m_1,m_2,r_1,r_2. Hence, the univariate polynomial of degree at most k

$$\mathsf{P}_q\left(x,-\frac{\beta}{\alpha}x+\beta\left(m_1+m_2+\frac{r_1}{\alpha}\right)+r_2\right)-\binom{W_0}{q}$$

has $i\,\alpha+r_1$ as a zero for all $i\in\{0,\dots,m_1+m_2\}$. As a corollary of the fundamental theorem of algebra, since $m_1+m_2+1>k$, we conclude that this polynomial is identically zero. Hence, we have a factorization of the form

$$\mathsf{P}_q(x,y)-\binom{W_0}{q}=\left[y+\frac{\beta}{\alpha}x-\beta\left(m_1+m_2+\frac{r_1}{\alpha}\right)-r_2\right]Q(x,y)$$

for any query q such that $|q|\le k$.

Let $a\in\Sigma$. From (8), we get

$$\alpha|v_1|_a=\beta|v_2|_a.$$

Let $g=u_1v_1^{\gamma_1}u_2v_2^{\gamma_2}u_3$ be the word to guess. The above discussion was made for arbitrary m_1,m_2,r_1,r_2. Now, we specialize these values as follows. Consider the Euclidean divisions: $\gamma_1=m_1\alpha+r_1$ and $\gamma_2=m_2\beta+r_2$. Assume that g is long enough to ensure $m_1+m_2\ge k$. In particular, note that the word W_0 above is nothing else but g. The equation $\mathsf{Eq}_{g,a}(x,y)$ is again (9). It is easy to see that the constant term can be rewritten as

$$\frac{1}{|v_2|_a}\left(\binom{g}{a}-|u_1u_2u_3|_a\right)=\frac{|v_1^{\gamma_1}v_2^{\gamma_2}|_a}{|v_2|_a}.$$

and is equal to $\beta(m_1+m_2+r_1/\alpha)+r_2$. So $\mathsf{Eq}_{g,a}(x,y)=0$ exactly matches (13). Hence every solution (x_0,y_0) satisfying $\mathsf{Eq}_{g,a}(x_0,y_0)=0$ is such that

$$\mathsf{P}_q(x_0,y_0)-\binom{g}{q}=0$$

and is thus a solution of $\mathsf{Eq}_{g,q}(x,y)=0$. So queries q such that $2\le|q|\le k$ do not give any extra constraints on the solutions, when added to linear equations provided by letters. Otherwise stated, substituting y from $\mathsf{Eq}_{g,a}(x,y)=0$ into $\mathsf{Eq}_{g,q}(x,y)=0$ leads to a trivial equation.

⇒ Conversely, assume that, for all q such that $1\le|q|\le k$, $\mathsf{Eq}_{g,q}(x,y)$ vanishes when substituting y as above. This means that every solution of the form $(i\alpha+r_1,(m_1+m_2-i)\beta+r_2)$ of $\mathsf{Eq}_{g,a}(x,y)=0$ also satisfies $\mathsf{Eq}_{g,q}(x,y)=0$. Thus, for all $i\in\{0,\dots,m_1+m_2\}$,

$$\mathsf{P}_q(i\,\alpha+r_1,(m_1+m_2-i)\,\beta+r_2)=\binom{g}{q}.$$

Since the r.h.s. is constant, we conclude that

$$u_1 v_1^{r_1}(v_1^\alpha)^i u_2 (v_2^\beta)^{m_1+m_2-i} v_2^{r_2} u_3 \sim_k u_1 v_1^{r_1}(v_1^\alpha)^j u_2 (v_2^\beta)^{m_1+m_2-j} v_2^{r_2} u_3$$

for all $i, j \leq m_1 + m_2$. Take $i = 1$ and $j = 0$. By cancellation, we get

$$v_1^\alpha u_2 \sim_k u_2 v_2^\beta.$$

□

<u>Step 3.</u> From now on, we may assume that (10) is zero, for all $a, b \in \Sigma$.

Remark 7. As observed in the proof of Lemma 4, if (10) is zero, for all $a, b \in \Sigma$, then $\mathsf{Eq}_{g,ab}(x, y)$ has $\mathsf{Eq}_{g,a}(x, y)$ as a factor. Hence, adding the set of quadratic equations $\{\mathsf{Eq}_{g,q}(x, y) = 0 : |q| = 2\}$ does not modify the set of solutions given by the linear equation $\mathsf{Eq}_{g,a}(x, y) = 0$ (assuming a occurs in v_1). Hence, by Proposition 6, in the current situation, we have $v_1^\alpha u_2 \sim_2 u_2 v_2^\beta$.

Note that we cannot have $v_1^\alpha u_2 = u_2 v_2^\beta$. Indeed, if u_2 is empty, then $v_1^\alpha = v_2^\beta$ and v_1, v_2 are power of the same word (thanks to Fine–Wilf theorem). However, the regular expression has been assumed to be reduced. If u_2 is non-empty, this imply that u_2 and v_1 start with the same letter which is also impossible.

So there exists a largest $K \geq 2$ (bounded by $|v_1^\alpha u_2| - 1$) such that

$$v_1^\alpha u_2 \sim_K u_2 v_2^\beta \quad \text{and} \quad v_1^\alpha u_2 \not\sim_{K+1} u_2 v_2^\beta.$$

Lemma 5. *In the situation described above (with the definition of K), a word in L (of unknown length) can be uniquely reconstructed with two queries (of length 1 and $K + 1$ respectively).*

Proof. By Proposition 6, there exists a word q of length $K + 1$ such that when substituting y with (9) in $\mathsf{Eq}_{g,q}$ the equation does not vanish (because otherwise, K would not be maximal). Let

$$\delta = \binom{v_1^\alpha u_2}{q} - \binom{u_2 v_2^\beta}{q}.$$

Since $\mathsf{Eq}_{g,01}(x, y) = 0$ describes two parallel lines of slope $-\beta/\alpha$, we have the following periodicity of the bivariate polynomial

$$\mathsf{Eq}_{g,01}(x + \alpha, y - \beta) = \mathsf{Eq}_{g,01}(x, y). \tag{14}$$

In particular, the homogeneous quadratic part of the equation is thus $(\beta x + \alpha y)^2$.

Assume first $y > \beta$ and that $x, y \in \mathbb{N}$. Now observe that

$$\begin{aligned} \mathsf{Eq}_{g,q}(x + \alpha, y - \beta) - \mathsf{Eq}_{g,q}(x, y) &= \mathsf{P}_q(x + \alpha, y) - \mathsf{P}_q(x, y + \beta) \\ &= \binom{u_1 v_1^{x+\alpha} u_2 v_2^{y-\beta} u_3}{q} - \binom{u_1 v_1^x u_2 v_2^y u_3}{q} = \delta. \end{aligned}$$

Indeed, let $w = u_1 v_1^x$ and $z = v_2^{y-\beta} z$, by Chu–Vandermonde identity

$$\binom{wv_1^\alpha u_2 z}{q} = \binom{w}{q} + \binom{v_1^\alpha u_2 z}{q} + \sum_{\substack{q=q_1 q_2 \\ q_1,q_2\in\Sigma^+}} \binom{w}{q_1}\binom{v_1^\alpha u_2 z}{q_2}$$

and proceed similarly with $wu_2 v_2^\beta z$. Since $v_1^\alpha u_2 z \sim_K u_2 v_2^\beta z$ and $|q_2| \le K$, we conclude that

$$\binom{wv_1^\alpha u_2 z}{q} - \binom{wu_2 v_2^\beta z}{q} = \binom{v_1^\alpha u_2 z}{q} - \binom{u_2 v_2^\beta z}{q} = \binom{v_1^\alpha u_2}{q} - \binom{u_2 v_2^\beta}{q} = \delta.$$

To get rid of z, apply Chu–Vandermonde again. Since the equality

$$\mathsf{Eq}_{g,q}(x+\alpha, y-\beta) - \mathsf{Eq}_{g,q}(x,y) = \delta$$

holds on an arbitrary large grid of integer points, the polynomial identity holds for all $x, y \in \mathbb{R}$ (applying Schwartz–Zippel Lemma as in the proof of Proposition 3).

Proceed to the division of $\mathsf{Eq}_{g,q}(x,y)$ by $\mathsf{Eq}_{g,01}(x,y)$ assuming the monomial order $x > y$. Since $\mathsf{Eq}_{g,01}(x,y)$ contains a leading term in x^2, we get an expression of the form

$$\mathsf{Eq}_{g,q}(x,y) = Q(x,y)\,\mathsf{Eq}_{g,01}(x,y) + R(x,y)$$

where

$$R(x,y) = x\,A(y) + B(y)$$

for some $A, B \in \mathbb{Q}[y]$. In general, nothing can be expected about the degree of A and B. We only know that no term of the remainder R is divisible by x^2. Our aim is to show that $A(y)$ is a constant and $B(y)$ has degree at most 1.

Shifting $(x,y) \mapsto (x+\alpha, y-\beta)$ and using the periodicity of (14), we also have

$$\mathsf{Eq}_{g,q}(x+\alpha, y-\beta) = Q(x+\alpha, y-\beta)\,\mathsf{Eq}_{g,01}(x,y) + R(x+\alpha, y-\beta)$$

Subtracting the last two relations, we get

$$\delta = [Q(x+\alpha, y-\beta) - Q(x,y)]\,\mathsf{Eq}_{g,01}(x,y) + R(x+\alpha, y-\beta) - R(x,y)$$

We now argue on the degree of x appearing in the different terms. The l.h.s. is a constant, $\deg_x \mathsf{Eq}_{g,01}(x,y) = 2$ and $\deg_x(R(x+\alpha, y-\beta) - R(x,y)) \le 1$. Hence, $Q(x+\alpha, y-\beta) - Q(x,y)$ must be zero. Now, we have

$$\delta = [A(y-\beta) - A(y)]x + A(y-\beta)\alpha + B(y-\beta) - B(y).$$

Again, the l.h.s. is constant. Hence $A(y-\beta) - A(y) = 0$ for all y. This means that the polynomial A is constant, say λ. Thus,

$$\delta - \lambda\alpha = B(y-\beta) - B(y).$$

So, we get $B(y) = \frac{\lambda\alpha-\delta}{\beta}y + \kappa$ for some constant κ. Hence, we conclude that the remainder is linear:

$$\mathsf{Eq}_{g,q}(x,y) = Q(x,y).\mathsf{Eq}_{g,01}(x,y) + \lambda x + \frac{\lambda\alpha-\delta}{\beta}y + \kappa.$$

Now substituting y with (9) provides a linear equation in x. Indeed, when substituting y with (9) in $\mathsf{Eq}_{g,01}(x,y)$, this factor vanishes because $v_1^\alpha u_2 \sim_2 u_2 v_2^\beta$. Moreover, we have chosen q such that $\mathsf{Eq}_{g,q}(x,y)$ does not vanish when substituting y with (9). So the linear equation is not identically zero and we can uniquely determine x, then y from (9). □

Remark 8. In [30] the commutation relation $uv \sim_k vu$ is studied. In particular, if $|u| = |v|$ then, it is shown that $uv \sim_k vu$ if and only if $u \sim_{k-1} v$. However, the question about the conjugacy relation $uv \sim_k vw$ is far from being solved. It is thus an open problem to describe the exact combinatorial conditions under which the relation $v_1^\alpha u_2 \sim_K u_2 v_2^\beta$ holds.

Under the extra assumptions that $v_1^\alpha \sim_K v_2^\beta$ and $|v_1^\alpha| = |u_2|$, then more can be said. We must have $u_2 \sim_{K-1} v_1^\alpha$. It is a consequence of the next two results from Whiteland [30].

Lemma 6. *Let $y, z \in \Sigma^*$ and $k, n \geq 1$ be integers. If $y^n \sim_k z^n$, then $y \sim_k z$.*

Theorem 2. *Let $k \geq 2$ be an integer and let $u, v \in \Sigma^*$ be words such that $|u| = |v|$. Then $uv \sim_k vu$ if and only if $u \sim_{k-1} v$.*

Corollary 1. *Let u, v, w be words of the same length such that $v \sim_k w$. Then $uv \sim_k wu$ if and only if $u \sim_{k-1} v$ (and also $u \sim_{k-1} w$).*

Proof. Assume that $uv \sim_k wu$. Since $v \sim_k w$, we have $uw \sim_k uv \sim_k wu$. By the above theorem, we conclude that $u \sim_{k-1} v$. Since $v \sim_{k-1} w$, we also have $u \sim_{k-1} w$. The converse is a direct application of Lemma 6. □

7 Investigating more loops

We now consider the general case. Let $m \geq 2$ and $q_1, \ldots, q_m \in \Sigma^+$. With these words, we can associate a system of m polynomial equations $\mathsf{Eq}_{g,q_i} = 0$ for $i = 1, \ldots, m$ (recall Definition 2). If $g = u_1 v_1^{\gamma_1} \cdots u_t v_t^{\gamma_t} u_{t+1}$, we known that this system has at least one non-negative integer solution $(\gamma_1, \ldots, \gamma_t)$. It would be enough to show that the map

$$F : \mathbb{N}^t \to \mathbb{N}^m,\ \mathbf{x} = (x_1, \ldots, x_t) \mapsto (\mathsf{P}_{q_1}(\mathbf{x}), \ldots, \mathsf{P}_{q_m}(\mathbf{x}))$$

is injective. For fixed $q_1, \ldots, q_m \in \Sigma^+$, if the map F is injective, there exists a unique t-uple $\mathbf{x} = (x_1, \ldots, x_t)$ such that

$$(\mathsf{P}_{q_1}(\mathbf{x}), \ldots, \mathsf{P}_{q_m}(\mathbf{x})) = \left(\binom{g}{q_1}, \ldots, \binom{g}{q_m}\right)$$

and g can thus be uniquely reconstructed.

We extend the domain to the non-negative real numbers and consider the map

$$G : \mathbb{R}^t_{\geq 0} \to \mathbb{R}^m_{\geq 0},\ \mathbf{x} = (x_1, \dots, x_t) \mapsto (\mathsf{P}_{q_1}(\mathbf{x}), \dots, \mathsf{P}_{q_m}(\mathbf{x})).$$

Indeed, as described below, using quantifier elimination, the theory of the real numbers is decidable whereas Matiyasevich's theorem implies that the corresponding theory over the integers is not. For the question of decidability over the purely existential fragment, more efficient algorithms have been devised: their complexity is simply exponential, instead of doubly exponential in the number of variables. See, for instance, [18]. For some general reference on real algebraic geometry, see [2].

Injectivity is trivially defined by the sentence in the existential theory of the reals,

$$\neg \left(\exists x_1, \dots, x_t, y_1, \dots, y_t \geq 0 : \bigvee_{j=1}^{t} x_j \neq y_j \wedge G(\mathbf{x}) = G(\mathbf{y}) \right). \tag{15}$$

Tarski–Seidenberg theorem [26] states that the projection on a linear subspace of any semialgebraic set in $\mathbb{R}^{n+1}$ (i.e., a finite union of sets defined by polynomial equalities and polynomial inequalities) is again a semialgebraic set in $\mathbb{R}^n$. Since existential quantification corresponds to a projection, this result thus allows to replace a formula $(\exists x_{n+1})\Psi(x_1, \dots, x_{n+1})$ by some equivalent quantifier-free formula $\varphi(x_1, \dots, x_n)$, i.e.,

$$(\exists x_{n+1})\Psi(x_1, \dots, x_{n+1}) \Leftrightarrow \varphi(x_1, \dots, x_n).$$

By induction, it follows that any quantified formula may be replaced by a quantifier-free one. In the case of the sentence (15), since there is no free variable, this process results in `True` or `False`. As a summary, the result is often stated as follows.

Theorem 3 (Tarski–Seidenberg Theorem). *For every first-order formula over the real field, there exists an equivalent quantifier-free formula. Furthermore, there is an explicit algorithm to compute this quantifier-free formula.*

`Mathematica` can handle quantifier elimination: QEPCAD-B [3] and versions of cylindrical algebraic decomposition [4] have been implemented [18] in the software. The command `Resolve[expr]` can, in principle, always eliminate quantifiers if `expr` contains only polynomial equations and inequalities over the real numbers.

Example 5. We have randomly generated a thousand instances of non-empty words $u_1, \dots, u_4, v_1, v_2, v_3 \in \{0,1\}^*$ of length at most 11 and considered the queries $q_1 = 0$, $q_2 = 1$, $q_3 = 01$, $q_4 = 001$ and $q_5 = 011$. Considering the first three (resp. four, five) polynomials, injectivity of the corresponding map G was certified for 695 (resp. 962, 994) languages of the sample. Five cases where it was not possible to prove injectivity, are trivial situations, e.g., $v_1 = v_2 = 00$ and $u_2 = 0$. In such a case, the minimal number of loops is not 3 but 2. The only non-trivial case where injectivity is not proved is

$$L = (10)^*01(01100111)^*1000(10)^*.$$

In that particular case, we get a polynomial

$$\mathsf{P}_{001}(x,1,z) = \frac{x^3}{6} + \frac{z^3}{6} + \frac{x^2z}{2} + \frac{xz^2}{2} + 3x^2 + 3z^2 + 6xz + \frac{107x}{6} + \frac{107z}{6} + 26$$

satisfying

$$\mathsf{P}_{001}(x,1,z) = \mathsf{P}_{001}(x+1,1,z-1) = \mathsf{P}_{001}(x-1,1,z+1).$$

Similar relations hold for P_{011}. Hence these queries are useless to prove injectivity. However, the queries q_1, q_2, q_3 and $q_6 = 0111$ are enough to get injectivity.

Proposition 7. *Let $q_1, \dots, q_m$ be queries and $G(\mathbf{x}) = (\mathsf{P}_{q_1}(\mathbf{x}), \dots, \mathsf{P}_{q_m}(\mathbf{x}))$ be the corresponding polynomial map. If the quantifier elimination of* (15) *returns* `True`*, then any word in the language* (1) *can be reconstructed in a constant number of m queries.*

Proof. Since G is injective on $\mathbb{R}^t_{\geq 0}$, the same holds for F. Otherwise stated, the system $(\mathsf{Eq}_{g,q_1}(\mathbf{x}), \dots, \mathsf{Eq}_{g,q_m}(\mathbf{x}))$ has a unique solution. We may assume that among the m queries, we have queries $\binom{g}{a}$ for all the letters of the alphabet. This information provides us with an obvious upper bound for each x_i:

$$x_i \leq \min_{\substack{a \in \Sigma \\ |v_i|_a \neq 0}} \left\lfloor \frac{\binom{g}{a} - |u_1 u_2 \cdots u_{t+1}|_a}{|v_i|_a} \right\rfloor .$$

Hence, there is only a finite number of t-uples of candidates to test and only one is valid. □

Example 6. With four loops, an example such a

$$L = 0(10010)^*010(010)^*10(001)^*110(1001)^*010$$

is certified to be reconstructed using the queries 0, 1, 01 and 011. Without the last query, injectivity fails. However, simply modifying a bit the language (e.g., $v_3 = 0011$ instead of 001) leads `Resolve` to not terminate. Hence, we face the limit of the exponential decision procedure.

With four loops but over a 3-letter alphabet, the situation is once again under control. We have randomly generated several thousand of languages, as in Example 5 with components of length at most 10, `Resolve` yields a certification with the queries 0, 1, 2, 01, 02 and 21. Dealing with quadratic polynomial surely helps.

Remark 9. In the literature, we also found other techniques to ensure injectivity of polynomial maps over the positive orthant. In [7], the authors characterize injectivity of classes of maps (on cosets of a linear subspace) by injectivity of classes of matrices. Their method avoids quantifier elimination and only uses linear algebra but it is unfortunately not suited for our case study. For the interested reader, we have considered the language $00(0100)^*10(10)^*(101)^*1$. With notation from [7], considering the queries 0, 1 and 01 give three polynomial maps encoded by two matrices (one for the coefficients, the other for the corresponding exponents):

$$A = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 & 3 & 0 & 0 & 0 \\ 2 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 7 & 1 & \frac{7}{2} & 2 & \frac{1}{2} & \frac{15}{2} & 6 & 3 & \frac{3}{2} \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 2 \\ 0 & 0 & 1 & 1 & 2 & 0 & 0 & 1 & 0 \\ 1 & 2 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}^{\top} .$$

Then, one has to look at the kernel of

$$A.\mathrm{diag}(\kappa_1, \dots, \kappa_9).B.\mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)$$

if all the parameters κ_i and λ_j are set to 1 except $\kappa_6 = 2$, then the determinant of the matrix is equal to zero. Thus the kernel is not restricted to $\{0\}$ which does not permit us to apply the method from [7].

References

1. Albayrak, S., Bell, J.P.: A refinement of Christol's theorem for algebraic power series. Math. Z. **300**(3), 2265–2288 (2022). https://doi.org/10.1007/s00209-021-02868-7
2. Basu, S., Pollack, R., Roy, M.F.: Algorithms in real algebraic geometry, Algorithms and Computation in Mathematics, vol. 10. Springer-Verlag, Berlin, second edn. (2006)
3. Brown, C.W.: QEPCAD B: a program for computing with semi-algebraic sets using cads. SIGSAM Bull. **37**(4), 97–108 (Dec 2003). https://doi.org/10.1145/968708.968710
4. Collins, G.E.: Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In: Quantifier elimination and cylindrical algebraic decomposition (Linz, 1993), pp. 85–121. Texts Monogr. Symbol. Comput., Springer, Vienna (1998). https://doi.org/10.1007/978-3-7091-9459-1_4
5. Şerbănuţă, T.F.: Extending Parikh matrices. Theoret. Comput. Sci. **310**(1-3), 233–246 (2004). https://doi.org/10.1016/S0304-3975(03)00396-7
6. Dudík, M., Schulman, L.J.: Reconstruction from subsequences. J. Comb. Theory, Ser. A **103**(2), 337–348 (2003). https://doi.org/10.1016/S0097-3165(03)00103-1
7. Feliu, E., Müller, S., Regensburger, G.: Characterizing injectivity of classes of maps via classes of matrices. Linear Algebra Appl. **580**, 236–261 (2019). https://doi.org/10.1016/j.laa.2019.06.015
8. Gabrys, R., Milenkovic, O.: The hybrid k-deck problem: Reconstructing sequences from short and long traces. IEEE Trans. Inform. Theory pp. 1206–1310 (2017). https://doi.org/10.1109/ISIT.2017.8006740

9. Gabrys, R., Milenkovic, O.: Unique reconstruction of coded strings from multiset substring spectra. IEEE Trans. Inform. Theory **65**(12), 7682–7696 (2019). https://doi.org/10.1109/TIT.2019.2935973
10. Ginsburg, S., Spanier, E.H.: Bounded regular sets. Proc. Am. Math. Soc. **17**, 1043–1049 (1966). https://doi.org/10.2307/2036087
11. Graham, R.L., Knuth, D.E., Patashnik, O.: Concrete mathematics. Addison-Wesley Publishing Company, Reading, MA, second edn. (1994), a foundation for computer science
12. Kalašnik, L.I.: The reconstruction of a word from fragments. In: Numerical mathematics and computer technology, No. IV (Russian), pp. 56–57, 137. Akad. Nauk Ukrain. SSR, Fiz.-Tehn. Inst. Nizkih Temperatur, Khar'kov (1973)
13. Krasikov, I., Roditty, Y.: On a reconstruction problem for sequences. J. Combin. Theory Ser. A **77**(2), 344–348 (1997). https://doi.org/10.1006/jcta.1997.2732
14. Lejeune, M., Rigo, M., Rosenfeld, M.: Templates for the k-binomial complexity of the Tribonacci word. Adv. in Appl. Math. **112**, 101947, 26 (2020). https://doi.org/10.1016/j.aam.2019.101947
15. Lothaire, M.: Combinatorics on Words. Cambridge Mathematical Library, Cambridge University Press (1997)
16. Manvel, B., Meyerowitz, A., Schwenk, A., Smith, K., Stockmeyer, P.: Reconstruction of sequences. Discrete Math. **94**(3), 209–219 (1991). https://doi.org/10.1016/0012-365X(91)90026-X
17. Osgood, W.F., Graustein, W.C.: Plane and solid analytic geometry. Macmillan (1922)
18. Passmore, G.O., Jackson, P.B.: Combined decision techniques for the existential theory of the reals. In: Carette, J., Dixon, L., Coen, C.S., Watt, S.M. (eds.) Intelligent Computer Mathematics. pp. 122–137. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)
19. Renard, A., Rigo, M., Whiteland, M.A.: q-Parikh matrices and q-deformed binomial coefficients of words. Discrete Math. **348**(5), Paper No. 114381, 21 (2025). https://doi.org/10.1016/j.disc.2024.114381
20. Reutenauer, C.: Free Lie algebras, London Mathematical Society Monographs. New Series, vol. 7. The Clarendon Press, Oxford University Press, New York (1993), oxford Science Publications
21. Richomme, G., Rosenfeld, M.: Reconstructing words using queries on subwords or factors. In: 40th International Symposium on Theoretical Aspects of Computer Science, LIPIcs. Leibniz Int. Proc. Inform., vol. 254, pp. Art. No. 52, 15. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern (2023). https://doi.org/10.4230/lipics.stacs.2023.52
22. Rigo, M., Salimov, P.: Another generalization of abelian equivalence: binomial complexity of infinite words. Theoret. Comput. Sci. **601**, 47–57 (2015). https://doi.org/10.1016/j.tcs.2015.07.025
23. Rigo, M.: Construction of regular languages and recognizability of polynomials. Discrete Math. **254**(1-3), 485–496 (2002). https://doi.org/10.1016/S0012-365X(01)00377-6
24. Rigo, M.: Formal languages, automata and numeration systems. 2. Networks and Telecommunications Series, ISTE, London; John Wiley & Sons, Inc., Hoboken, NJ (2014), applications to recognizability and decidability, With a foreword by Valérie Berthé
25. Szilard, A., Yu, S., Zhang, K., Shallit, J.: Characterizing regular languages with polynomial densities. In: Mathematical foundations of computer science 1992

(Prague, 1992). Lecture Notes in Comput. Sci., vol. 629, pp. 494–503. Springer, Berlin (1992). https://doi.org/10.1007/3-540-55808-X_48

26. Tarski, A.: A Decision Method for Elementary Algebra and Geometry. The Rand Corporation, Santa Monica, CA (1948)
27. Trofimov, V.I.: Growth functions of some classes of languages. Kibernetika **6**, 9–12 (1981), (in Russian)
28. Vivion, L.: On the k-binomial complexity of hypercubic billiard words. In: Mons Theoretical Computer Science Days (2024)
29. Vivion, L.: New examples of words for which the binomial complexities and the subword complexity coincide. arXiv:2509.11172 (2025), https://arxiv.org/abs/2509.11172
30. Whiteland, M.A.: Equations over the k-binomial monoids. In: Combinatorics on words, Lecture Notes in Comput. Sci., vol. 12847, pp. 185–197. Springer, Cham ([2021] ©2021)