Theoretical Computer Science 658 (2017) 4–17

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

# Two-dimensional comma-free and cylindric codes ☆

Marcella Anselmo [a], Maria Madonia [b]

[a] Dip. di Informatica, V. G. Paolo II, 132, Università di Salerno, I-84084 Fisciano (SA), Italy
[b] Dip. Matematica e Informatica, Università di Catania, Viale Andrea Doria 6/a, 95125 Catania, Italy

ARTICLE INFO

*Article history:*
Received 7 November 2015
Received in revised form 19 May 2016
Accepted 10 June 2016
Available online 18 June 2016

*Keywords:*
Two-dimensional languages
Codes
Comma-free codes
Circular codes

ABSTRACT

A two-dimensional code of pictures is defined as a set $X \subseteq \Sigma^{**}$ such that any picture over $\Sigma$ is tilable in at most one way with pictures in $X$. It has been proved that it is undecidable whether a finite set of pictures is a code. Here we introduce two classes of picture codes: the comma-free codes and the cylindric codes, with the aim of generalizing the definitions of comma-free (or self-synchronizing) code and circular code of strings. The properties of these classes are studied and compared, in particular in relation to maximality and completeness. As a byproduct, we introduce self-covering pictures and study their periodicity issues.



## 1. Introduction

The theory of variable-length codes is an important part of computer science both in its theoretical and practical issues. The theory is strongly related to combinatorics on words, formal languages, automata and semigroup theory. The aim is to find structural properties of codes to be exploited for their construction. We refer to [10] for complete references.

During the last fifty years, some significant work has been done to transfer the formalisms and the results from the string language theory into the two-dimensional (2D) world (see for example [8,11,17,18,29]). The extension of the classical notion of string (or word) to two dimensions leads to the definition of polyomino, in its different declinations – labeled polyominoes, directed polyominoes, as well as rectangular labeled polyominoes, that we will refer to as *pictures*. In the literature, one can find different attempts to generalize the notion of code to 2D objects. A set $C$ of polyominoes is a *code* if every polyomino is tilable in at most one way with (copies of) elements of $C$. The results show that in the 2D context most important properties are lost. A major result, due to D. Beauquier and M. Nivat, states that the problem whether a finite set of polyominoes is a code is undecidable, and the same result holds for dominoes, too ([9]). Related particular cases were studied in [1]. In [21], the codes of directed polyominoes equipped with catenation operations are considered, and a few special decidable cases are detected. The codes of labeled polyominoes, also called bricks, are studied in [26], and some further undecidability results are proved. In relation to the operations of row and column concatenations and in connection with doubly-ranked monoids, a definition of code of pictures (rectangular arrays of symbols) is given in [13] with the main goal of extending syntactic properties to two dimensions. Unfortunately most of the results are negative.

In this paper we consider the definition of code of pictures introduced in [4], which refers to the operation of tiling star. The *tiling star* of a language $X$, as defined in [29], is the set $X^{**}$ of all pictures that are tilable (in the polyominoes style)

☆ Partially supported by MIUR Project 2010LYA9RH *"Automata and Formal Languages: Mathematical and Applicative Aspects"*, MIUR Project *"PRISMA PON04a2 A/F"*, University of Salerno FARB Project ORSA138754, and University of Catania PRA Project 2012.

*E-mail addresses:* anselmo@dia.unisa.it (M. Anselmo), madonia@dmi.unict.it (M. Madonia).

http://dx.doi.org/10.1016/j.tcs.2016.06.016
0304-3975/

by elements of $X$. Then, $X$ is a code if any picture in $X^{**}$ is tilable in a unique way. In [4], it has been proved that it is undecidable whether a given set of pictures is a code. This is actually not surprising, because it is coherent with the result that it is undecidable whether a picture language can be (tiling system) recognized in an unambiguous way ([7]).

The search for decidable classes of picture codes has followed the footprints of the well established theory of string codes. Two generalizations of the notion of prefix code have been proposed and investigated in [4,6], the prefix codes and the strong prefix codes. Subsequently, the codes with finite deciphering delay have been introduced ([3]). All these classes have good decidability properties; it is decidable whether a finite language is a prefix code, or a strong prefix code, or a code with a given finite deciphering delay. It is meaningful to note that in these definitions the pictures are considered with a preferred scanning direction from a corner to the opposite one. For example, if $X$ is a (top-left to bottom-right) prefix code, the decoding process of a picture $p$ starts on its top-left corner, and proceeds towards the bottom right corner, in such a way that one can decide without ambiguity which is the next element in $X$.

This paper continues this line of investigation and considers two meaningful classes of picture codes, namely comma-free and cylindric codes, which extend comma-free and circular codes of strings. The novelty lies in the fact that the definitions are "non-oriented", in the sense that they do not follow a specific decoding direction.

Let us recall the origin of comma-free codes of strings. Taking inspiration from some biological mechanisms in the information transmission via DNA, in 1958, Golomb, Gordon, and Welch [19] introduced the general concept of comma-free code and studied the quantitative aspects of comma-free codes. As a matter of fact, the authors of the early period of investigation, have considered largely the comma-free codes of constant length and under the need and influence of biology. But, it appeared several years later that the biological code is not a comma-free code, not even a code in the general sense. Nevertheless, the comma-free code has deservedly entered into coding theory. In the first years, the main line of study was concerned with the maximal size of comma-free codes of strings of a fixed length and on alphabets with a fixed number of letters. The research on variable-length comma-free codes was initiated later in 1969 [28]. The problems of completion and of finite completion of comma-free codes have been solved by N.H. Lam [22–24]. The comma-free codes of strings are also called self-synchronizing codes, due to the following property. They have an easy deciphering: if $X$ is comma-free and in a word $w \in X^*$, some factor can be identified to be in $X$, then this factor is one term of the unique factorization of $w$ on $X$.

The generalization of comma-free codes of strings to higher dimensions has been considered in [14]. A $q$-dimensional comma-free code is a set $D$ of $q$-dimensional words or arrays of letters of fixed size, with the property that for any arrangement of the arrays in $D$ on a plane, no "proper" factor is in $D$. The author shows a (not tight) bound on the cardinality of $q$-dimensional comma-free codes (note that comma-free codes are necessarily finite with this definition).

In this paper we introduce variable-length two-dimensional *comma-free sets*. Hence, a comma-free set can be infinite. A set of pictures is comma-free if no picture in $X$ can be covered by the pictures in $X$. Any comma-free set is a code, that we call a *comma-free code*. The definition preserves the main property of self-synchronizability. If $X \subseteq \Sigma^{**}$ is a comma-free code and in a picture $p \in X^{**}$, some factor can be identified to be in $X$, then this factor is one term of the unique tiling decomposition of $p$ on $X$. In a simple way, it is decidable whether a finite set is a comma-free set. Therefore finite comma-free codes are a first example of a decidable class of codes of pictures with a "non-oriented" definition.

In the theory of string codes, comma-free codes are studied inside the class of circular codes. The translation of the definition of circular code of strings into the two-dimensional world leads to some new situations to deal with. The role of a circle can be played in 2D by a cylinder, either horizontally or vertically placed. Now, a set $X$ of pictures is a vertical *cylindric code* if the pictures of $X$ cannot tile the lateral surface of any cylinder (for any height and radius) in two different ways. A main result is that it is undecidable whether a (finite) set of pictures is a cylindric code. This result shows a crucial difference with the comma-free codes. This is not really surprising, since the definition of comma-free code is based on a "local" property (that can be tested on pictures), while the definition of cylindric code is founded on a more "global" and broader property. Subsequently, the maximality of such notions of code is investigated and related to a new notion of completeness. Finally, the relationships among all the introduced classes are shown.

The investigation on comma-free sets has led us to consider *(non-) self-covering pictures*. A picture $p$ is self-covering if it can be completely covered with some juxtaposed copies of itself. This notion deserves a separate consideration, since it represents a sort of periodicity on pictures. Note that matrix periodicity plays a fundamental role in two-dimensional pattern matching. The notion of periodicity in matrices and pictures has been investigated in [2,15,27], while two-dimensional quasi-periodicity was very recently studied in [16]. The definition starts from the consideration that in 1D a string is periodic if it self-overlaps across the middle point. Hence, in 2D a picture has been defined as periodic if it self-overlaps and this overlapping includes the center. The repetition of the self-overlapping produces some periodicity in the picture along one or more directions. Note that such periodicity may leave out some border positions (what is called a "free zone" in [27]) which may present no relation with the content in the rest of the picture. In [25], the well-known Fine and Wilf's Theorem on strings is generalized to two-dimensional words on a large class of convex domains through a geometric approach.

The definition of self-covering pictures, as considered here, gives rise to a different notion of periodicity. A picture $p$ is self-covering if it can be completely covered with some juxtaposed copies of itself. Then, the repetition of such overlapping gives raise to a periodicity in all positions of $p$, without "free zones". Differently said, a self-covering picture can be cut out from an infinite periodic configuration with a "small" period. Note that in 1D, a self-covering string is the power of some of its prefixes (without final extra symbols); hence the self-covering strings correspond to a stronger kind of periodic strings. The notion of non-self-coverability corresponds to a 2D notion of aperiodicity. Non-self-covering pictures are special

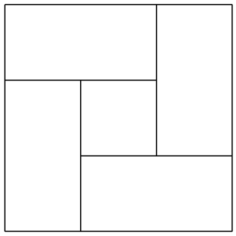


**Fig. 1.** A picture obtained using the tiling star.

unbordered pictures as introduced and studied in [5]. In the same paper the construction of the sets of unbordered pictures of given size is shown.

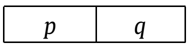
## 2. Preliminaries

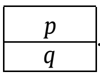
We recall some definitions about two-dimensional languages (see [17,18]). A *picture* over a finite alphabet $\Sigma$ is a two-dimensional rectangular array of elements of $\Sigma$. Given a picture $p$, $|p|_{row}$ and $|p|_{col}$ denote the number of rows and columns, respectively, while $size(p) = (|p|_{row}, |p|_{col})$ denotes the picture *size*. The set of all pictures over $\Sigma$ of fixed size $(m, n)$ is denoted by $\Sigma^{m,n}$, while $\Sigma^{m*}$ and $\Sigma^{*n}$ denote the set of all pictures over $\Sigma$ with a fixed number $m$ of rows and $n$ of columns, respectively. The set of all pictures over $\Sigma$ is denoted by $\Sigma^{**}$. A *two-dimensional language* (or *picture language*) over $\Sigma$ is a subset of $\Sigma^{**}$.

In order to locate a position in a picture, it is necessary to put the picture in a reference system. The set of coordinates $dom(p) = \{1, 2, \ldots, |p|_{row}\} \times \{1, 2, \ldots, |p|_{col}\}$ is referred to as the *domain* of a picture $p$. We let $p(i, j)$ denote the symbol in $p$ at coordinates $(i, j)$. Moreover, to easily detect the border positions of pictures, we use the initials of the words "top", "bottom", "left" and "right"; for example the *tl-corner* of $p$ refers to position $(1, 1)$.

A *subdomain* of $dom(p)$ is a set $d$ of the form $\{i, i+1, \ldots, i'\} \times \{j, j+1, \ldots, j'\}$, where $1 \le i \le i' \le |p|_{row}$, $1 \le j \le j' \le |p|_{col}$, also specified by the pair $[(i, j), (i', j')]$. The portion of $p$ corresponding to the positions in the subdomain $[(i, j), (i', j')]$ is denoted by $p[(i, j), (i', j')]$. Then, a non-empty picture $x$ is *subpicture of* $p$ if $x = p[(i, j), (i', j')]$, for some $1 \le i \le i' \le m$, $1 \le j \le j' \le n$; it will be referred to as the subpicture *associated* with $[(i, j), (i', j')]$ and we will say that $x$ *occurs* at position $(i, j)$ (its tl-corner). Prefixes of pictures are special subpictures. Given pictures $x, p$, with $|x|_{row} \le |p|_{row}$ and $|x|_{col} \le |p|_{col}$, picture $x$ is a *prefix* of $p$ if $x$ is the subpicture of $p$ occurring at its top-left corner, i.e. if $x = p[(1, 1), (|x|_{row}, |x|_{col})]$.

Dealing with pictures, two "classical" concatenation products are defined. Let $p, q \in \Sigma^{**}$ be pictures of size $(m, n)$ and $(m', n')$, respectively. The *column concatenation* of $p$ and $q$ (denoted by $p \obar q$) and the *row concatenation* of $p$ and $q$ (denoted by $p \ominus q$) are partial operations, defined only if $m = m'$ and if $n = n'$, respectively, as:

$p \obar q =$ 


$p \ominus q =$ 

.

These definitions can be extended to define row- and column-concatenations, and *row-* and *column-stars* of two-dimensional languages.

We also consider another interesting star operation for picture languages introduced by D. Simplot in [29], the tiling star. The idea is to compose pictures in a way to cover a rectangular area as, for example, in Fig. 1.

**Definition 1.** The *tiling star* of $X$, denoted by $X^{**}$, is the set of pictures $p$ whose domain can be partitioned in disjoint subdomains $\{d_1, d_2, \ldots, d_k\}$ such that any subpicture $p_h$ of $p$ associated with the subdomain $d_h$ belongs to $X$, for all $h = 1, \ldots, k$.

Language $X^{**}$ is called the set of all tilings by $X$ in [29]. In the sequel, if $p \in X^{**}$, the partition $\{d_1, d_2, \ldots, d_k\}$ of $dom(p)$, together with the associated pictures $\{p_1, p_2, \ldots, p_k\}$, is called a *tiling decomposition* of $p$ in $X$. Moreover if picture $p \in X^{**}$, we say that $p$ is *tilable* in $X$.

A picture $p$ is bordered when one can find the same rectangular portion at two opposite corners. Remark that there are two different kinds of borders depending on the pair of opposite corners that hold the border. More formally we state the following definition, as in [5].

**Definition 2.** Given pictures $p \in \Sigma^{m,n}$ and $x \in \Sigma^{m',n'}$, with $1 \le m' \le m$ and $1 \le n' \le n$, the picture $x$ is a *tl-border* of $p$, if $x$ is a subpicture of $p$ occurring at position $(1, 1)$ and at position $(m - m' + 1, n - n' + 1)$; picture $x$ is a *bl-border* of $p$, if $x$ is a subpicture of $p$ occurring at position $(m - m' + 1, 1)$ and at position $(1, n - n' + 1)$. Moreover $x$ is a *border* of $p$ if it is either a tl- or a bl-border.

Note that, given a picture $p \in \Sigma^{m,n}$, any tl-border of $p$, with size $(m', n)$ or $(m, n')$, is also a bl-border of $p$.

As special cases, $p$ is a *trivial border* of itself, and $x$ is a *proper border* of $p$ if it is not trivial. A picture $p \in \Sigma^{m,n}$ is *bordered* if there exists a picture $x$ that is a proper border of $p$. Picture $p$ is *unbordered* (or *border-free*) if it is not bordered.

Following the terminology in [2], the borders will be identified by some vectors (symmetry vectors in [2]). Given a picture $p$ of size $size(p) = (m, n)$, the tl-border of $p$ of size $(m', n')$ will be identified by the vector $(n - n', m - m')$; it will be called the border of vector $(n - n', m - m')$. In a similar way, the bl-border of $p$ of size $(m', n')$ will be identified by the vector $(n - n', m' - m)$ and it will be called the border of vector $(n - n', m' - m)$.

In the sequel, we will need to manage also non-rectangular "portions" of pictures composed by elements of $X$; those are actually labeled polyominoes, that we will call polyominoes, for the sake of simplicity.

The notations and definitions relative to the pictures can be extended to polyominoes by simply defining the *domain of a (labeled) polyomino* as the set of pairs $(i, j)$ corresponding to all positions occupied inside its minimal bounding box, $(1, 1)$ being the tl-corner position of the bounding box (as in [4]). In this way, we can use the notion of *subpicture* or *prefix of a polyomino c*. Observe that the notion of prefix of polyomino makes sense only if the domain of the polyomino contains $(1, 1)$.

Moreover, we can extend to polyominoes the notion of tiling decomposition in a set of pictures $X$. We define a sort of tiling star that, applied to a set of pictures $X$, produces the set of all polyominoes that have a tiling decomposition in $X$. If a polyomino $p$ belongs to the polyomino star of $X$, we say that $p$ is *tilable* by $X$.

## 3. Two-dimensional codes

Let us recall the definition of code of pictures given in [4], together with some examples. Let $\Sigma$ be a finite alphabet. $X \subseteq \Sigma^{**}$ is a *code* if any $p \in \Sigma^{**}$ has at most one tiling decomposition in $X$.

**Example 3.** Let $\Sigma = \{a, b\}$ be the alphabet and let $X = \left\{ \boxed{a\ b}, \boxed{\begin{array}{c} a \\ b \end{array}}, \boxed{\begin{array}{cc} a & a \\ a & a \end{array}} \right\}$. It is easy to see that $X$ is a code. Any picture $p \in X^{**}$ can be decomposed starting at its tl-corner and checking the subpicture $p[(1, 1), (2, 2)]$; it can be univocally decomposed in $X$. Then, proceed similarly with the next contiguous subpictures of size $(2, 2)$.

**Example 4.** Let $X = \left\{ \boxed{a\ b}, \boxed{b\ a}, \boxed{\begin{array}{c} a \\ a \end{array}} \right\}$. Notice that no picture in $X$ is a prefix of another picture in $X$. Nevertheless, $X$ is not a code. Indeed picture $\boxed{\begin{array}{ccc} a & b & a \\ a & b & a \end{array}}$ has two different tiling decompositions in $X$, $t_1 = \begin{array}{|cc|c|} \hline a & b & a \\ \hline a & b & a \\ \hline \end{array}$ and $t_2 = \begin{array}{|c|cc|} \hline a & b & a \\ \hline a & b & a \\ \hline \end{array}$.

Although this notion of code seems to generalize naturally from strings to pictures, the problem whether a given set of pictures is a code is undecidable, even for finite sets [4]. A challenging aim is then to find decidable subclasses of codes. Recently, some decidable families of two-dimensional codes have been introduced and investigated [3,4,6]. They extend to two dimensions the notions of prefix code of strings and of code with finite deciphering delay. Note that these definitions are all given following a privileged corner-to-corner direction of decoding. On the other hand, the families of codes we are going to introduce are non-oriented, in the sense that they are not related to a leading direction.

## 4. Comma-free codes and self-covering pictures

In this section we introduce the comma-free codes of pictures, as a generalization of the comma-free, or self-synchronizing, codes of strings. The aim is to propose a decidable class of codes with a definition which does not need a privileged decoding direction. This definition highlights the notion of (non-) self-covering picture, a kind of pictures which deserves to be studied separately.

Let us specify the following definition, illustrated in Fig. 2, where picture $p$ is drawn with dashed borders.

**Definition 5.** A picture $p$ is *covered* by a set of pictures $X$, if $p$ is a subpicture of a polyomino $c$ whose domain can be partitioned into rectangular subdomains $\{d_1, ..., d_h\}$, such that, for any $i = 1, \ldots, h$, $d_i$ corresponds to a picture in $X$, and $d_i \cap d \neq \emptyset$, where $d$ is the subdomain that corresponds to $p$.

Moreover, if $h = 1$ then the picture associated with $d_1$ is not $p$.

The partition $\{d_1, ..., d_h\}$ together with the associated subpictures will be called a *covering* of $p$.

**Definition 6.** A language $X \subseteq \Sigma^{**}$ is comma-free if no picture $p \in X$ is covered by $X$.

**Example 7.** Let $X$ be the following language $X = \left\{ \boxed{\begin{array}{ccc} a & b & b \\ a & a & b \\ b & a & a \end{array}}, \boxed{\begin{array}{ccc} b & b & b \\ b & a & a \end{array}}, \boxed{b\ a\ b\ a\ b\ b} \right\}$.

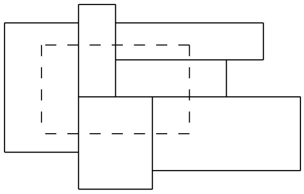


**Fig. 2.** A picture covered by a set of pictures.

The language $X$ is comma-free. This can be checked by considering all possible attempts to cover by $X$ each picture in $X$, as follows.

Denote $x_1 = \boxed{\begin{array}{ccc} a & b & b \\ a & a & b \\ b & a & a \end{array}}$, $x_2 = \boxed{\begin{array}{ccc} b & b & b \\ b & a & a \end{array}}$, and $x_3 = \boxed{\begin{array}{cccccc} b & a & b & a & b & b \end{array}}$. Consider first $x_1$ and look for all the ways to cover a prefix of $x_1$ with a picture in $X$ (maybe, $x_1$ itself). There are the following possibilities. A first attempt is to cover $x_1[(1,1),(1,1)]$ with the br-corner of $x_1$ or $x_2$. After this, no picture in $X$ can be hereby placed in a way to cover the position $(1,2)$ of $x_1$. So, this attempt fails. A second attempt is to cover $x_1[(1,1),(1,3)]$ with the br-portion of $x_3$. After this, no picture in $X$ can be placed in a way to cover the position $(2,1)$ of $x_1$. A third attempt is to cover $x_1[(1,1),(2,2)]$ with the br-portion of $x_1$. After this, it is possible to cover, in many different ways, $x_1[(1,3),(2,3)]$ (for example with the tl-portion of $x_2$ or with two copies of the tl-corner of $x_3$, or with the bl-corner of $x_1$ and the tl-corner of $x_3$ and so on) and it is also possible to cover, in many different ways, $x_1[(3,1),(3,1)]$. But no picture in $X$ can be placed in a way to cover the position $(3,2)$ of $x_1$. Hence picture $x_1$ cannot be covered by $X$.

Consider now picture $x_2$. The attempt to cover $x_2[(1,1),(1,1)]$ with the br-corner of $x_3$ fails since then there is no way to place a picture in $X$ on position $(1,2)$ of $x_2$. In a similar way, also the attempt to cover $x_2[(1,1),(2,1)]$ with $x_1$ fails. Finally, the attempt to cover $x_2[(1,1),(1,2)]$ with the br-corner of $x_3$ fails since, even if it is possible to cover $x_2[(1,3),(1,3)]$ with the bl-corner of either $x_1$ or $x_3$ and to cover $x_2[(2,1),(2,1)]$ with the tr-corner of either $x_1$ or $x_3$, there is no way to place a picture in $X$ on position $(2,2)$ of $x_2$.

Finally, consider picture $x_3$. The prefix $x_3[(1,1),(1,1)]$ can be covered with $x_1$, $x_2$, or $x_3$. After this, no picture in $X$ can be placed in a way to cover the position $(2,1)$ of $x_3$.

No other way exists to cover any picture in $X$. Hence, the language $X$ is comma-free.

**Remark 8.** It is immediate to observe that any comma-free set is a code, that we will call a comma-free code.

**Remark 9.** Comma-free codes of strings are also called self-synchronizing codes, due to the following property. Comma-free codes have an easy deciphering: if $X$ is comma-free and in a word $w \in X^*$, some factor can be identified to be in $X$, then this factor is one term of the unique factorization of $w$ on $X$.

An analogous property holds for comma-free codes of pictures, as here introduced. If $X \subseteq \Sigma^{**}$ is comma-free and in a picture $p \in X^{**}$, some factor can be identified to be in $X$, then this factor is one term of the unique tiling decomposition of $p$ in $X$.

Note that a picture in a comma-free code in particular cannot be covered with copies of itself. This definition deserves a special attention.

### *4.1. Self-covering pictures and periodicity*

A picture $p$ is self-covering when it can be completely covered with some juxtaposed copies of itself. Any picture in a comma-free code is not self-covering. The property of being self-covering implies a sort of periodicity in a picture. We are going to investigate this correspondence.

**Definition 10.** A picture $p \in X^{**}$ is *self-covering* if it is covered by $\{p\}$.

**Example 11.** A simple example of a self-covering picture is $\boxed{\begin{array}{cc} a & b \\ b & a \end{array}}$.

The definition of self-coverability corresponds to a notion of periodicity in pictures, as shown in Proposition 12. It says that a picture $p$ is self-covering if and only if it can be cut out from some particular periodic configuration. Let us introduce some terminology (see [20]).

A two-dimensional infinite array of symbols in $\Sigma$, or a *configuration*, is any $c \in \Sigma^{\mathbb{Z}^2}$. As usual, we denote by $c_v \in \Sigma$ the symbol in $c$ in position $v \in \mathbb{Z}^2$. For $u \in \mathbb{Z}^2$, we say that $c$ is *u-periodic* if $c_v = c_{v+u}$ holds for all $v \in \mathbb{Z}^2$, and $c$ is *periodic* if it is $u$-periodic for some $u \in \mathbb{Z}^2$, with $u \neq (0,0)$.

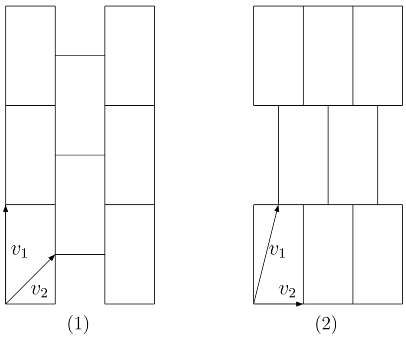



**Fig. 3.** A picture that tiles a configuration: two cases.

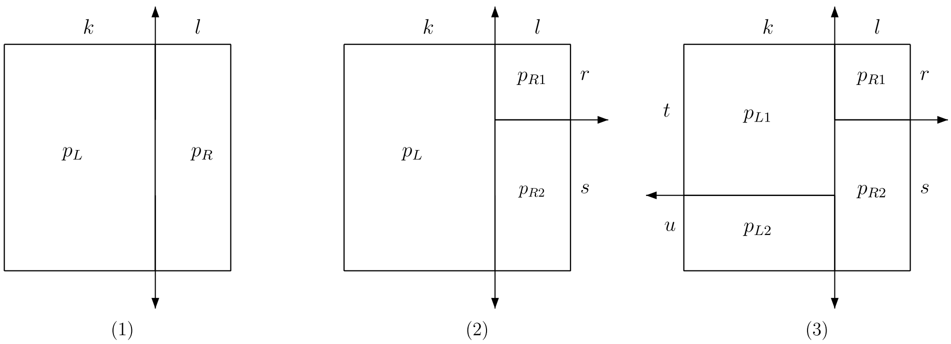



**Fig. 4.** The three main types of a self-covering.

We say that a picture $p$ *tiles* the configuration $c$, if $c$ can be obtained as a juxtaposition of copies of $p$ without holes. For example, $p$ can appear in $c$ with bl-corner in some position $(h, k)$ and then repeat itself on the top and on the right of any of its occurrences. More formally, for $p \in \Sigma^{**}$, with $size(p) = (m, n)$, we say that $p$ *tiles* the configuration $c$ with vectors $v_1$ and $v_2$ if:

- $c_v = p(m - j + k, i - h + 1)$, for $v = (i, j)$ and $i = h, h+1, \cdots, h+n-1$, $j = k, k+1, \cdots, k+m-1$
- $c$ is $v_1$- and $v_2$-periodic, for some vectors $v_1$, $v_2 \in \mathbb{Z}^2$ with either $v_1 = (0, m)$ and $v_2 = (n, m')$, $0 \leq m' \leq m$ (see Fig. 3(1)), or $v_1 = (n', m)$ and $v_2 = (n, 0)$, $0 \leq n' \leq n$ (see Fig. 3(2)).

If $p$ tiles $c$ then $c$ is $v_1$- and $v_2$-periodic, by definition. The case when $p$ is self-covering is characterized by the fact that $c$ is also $u$-periodic for some $u$ “smaller” than $v_1$ and $v_2$.

Before going on, let us look in detail at a self-covering picture and classify the different cases whereby a picture is self-covering. Recall that a picture $p$ is self-covering if it is covered by $\{p\}$, that is $p$ is a subpicture of a special polyomino $c$ whose domain can be partitioned in rectangular subdomains $\{d_1, ..., d_h\}$, such that each $d_i$ corresponds to $p$ (cf. Definition 5 for all details). The partition together with the associated subpictures will be called a *self-covering* of $p$. Note that in this case $h = 2, 3$, or 4. A self-covering of a picture $p$ of size $(m, n)$ can be of several different types. Among these types, we choose three main ones, in such a way that the other ones can be obtained by a rotation. In the sequel, all the proofs and considerations are explicitely given for these main types, and they can be simply adapted to the other cases. Let us classify the three main types of a self-covering, as follows (see Fig. 4).

(1) $(h = 2)$; $p = p_L \oplus p_R$ where $p_L$ is a tl-border of vector $(l, 0)$ with $l = |p_R|_{col}$, and $p_R$ is a tl-border of vector $(k, 0)$ with $k = |p_L|_{col}$, and $k + l = n$
(2) $(h = 3)$; $p = p_L \oplus (p_{R1} \ominus p_{R2})$ where $p_L$ is a tl-border of vector $(l, 0)$, $l = |p_{R1}|_{col}$, $p_{R1}$ is a bl-border of $p$ of vector $(k, -s)$, with $k = |p_L|_{col}$, $s = |p_{R2}|_{row}$, and $p_{R2}$ is a tl-border of $p$ of vector $(k, r)$, with $r = |p_{R1}|_{row}$.
(3) $(h = 4)$; $p = (p_{L1} \ominus p_{L2}) \oplus (p_{R1} \ominus p_{R2})$ where $p_{L1}$ is a tl-border of $p$ of vector $(l, u)$, with $l = |p_{R1}|_{col}$, $u = |p_{L2}|_{row}$, $p_{L2}$ is a bl-border of $p$ of vector $(l, -t)$, with $t = |p_{L1}|_{row}$, $p_{R1}$ is a bl-border of $p$ of vector $(k, -s)$, with $k = |p_{L1}|_{col}$, $s = |p_{R2}|_{row}$, and $p_{R2}$ is a tl-border of $p$ of vector $(k, r)$, with $r = |p_{R1}|_{row}$.

**Proposition 12.** *Let $p \in \Sigma^{**}$ be a picture. Then $p$ is self-covering if and only if $p$ tiles a configuration $c$ with period $v = (x, y)$ with $x < |p|_{col}$ and $y < |p|_{row}$.*

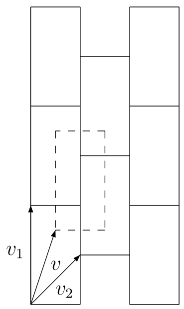



**Fig. 5.** The self-covering of picture $p$ considered in the proof of Proposition 12.

**Proof.** Let $p$ be a self-covering picture of size $(m, n)$. The proof is given for the cases (1), (2), and (3) of the above classification and can be simply adapted to the other cases.

If $p$ has a self-covering of type (1) then consider the configuration $c$ that is tiled by $p$ with periodicities $v_1 = (0, m)$ and $v_2 = (n, 0)$. The configuration $c$ has period $(k, 0)$ and $k < n$.

If $p$ has a self-covering of type (2) then consider the configuration $c$ that is tiled by $p$ with periodicities $v_1 = (0, m)$ and $v_2 = (n, s)$. The configuration $c$ has period $(k, s)$ with $k < n$ and $s < m$.

If $p$ has a self-covering of type (3) then consider the configuration $c$ that is tiled by $p$ with periodicities $v_1 = (0, m)$ and $v_2 = (n, s - u)$. The configuration $c$ has period $(k, s - u)$ with $k < n$ and $s - u < m$.

Vice versa, consider the configuration $c$ which is tiled by $p$ with vectors $v_1$, $v_2$, and which has period $v = (x, y)$ with $x < |p|_{col}$ and $y < |p|_{row}$. The idea is that there is an occurrence of $p$ with its bl-corner falling in the internal part of another occurrence of $p$ that can be replicated all around it (as sketched in Fig. 5). Due to length reasons, $p$ is covered by some other copies of $p$ juxtaposed each other.

More precisely, suppose without loss of generality that $v_1 = (0, m)$ and $v_2 = (n, m')$, $0 \leq m' < m$.

Consider first the case $m' = 0$. If $y = 0$ then $p$ has a self-covering of type (1) with $k = x$. If $y \neq 0$ then $p$ has a self-covering of type (3) with $t = r = y$ and $l = x$.

Let now $m' \neq 0$. If $y = 0$ then $p$ has a self-covering of type (2) with $l = x$ and $s = m'$. If $y \neq 0$ then $p$ has a self-covering of type (3) with $t = y$, $l = x$, and $r = y - m'$. □

Let us conclude the section with some considerations on the combinatorial structure of self-covering pictures.

Let $p$ be a self-covering picture of size $(m, n)$.

If $p$ has a self-covering of type (1) then $p = p_L \oplus p_R$ where $p_L$ is a tl-border of vector $(l, 0)$ with $l = |p_R|_{col}$, and $p_R$ is a tl-border of vector $(k, 0)$ with $k = |p_L|_{col}$, and $k + l = n$. Consider $p$, $p_L$, and $p_R$ as strings on the alphabet $\Sigma^{m,1}$. We have that $p = p_L p_R = p_R p_L$. Applying a well-known result in the theory of languages of strings (see [10]), one can state that $p = z^d$, for some integer $d \neq 1$ that divides $n$. More precisely, applying Fine and Wilf's Theorem, $p$ has period $gcd(k, l)$.

In a similar way, if $p$ has a self-covering of type (2), then the existence of the tl-border of vector $(l, 0)$ implies that $p = z \oplus \cdots \oplus z \oplus z'$, for some $z, z' \in \Sigma^{m,*}$ and $z'$ prefix of $z$, with $|z|_{col} < n$. The existence of the other two borders may also produce or not a periodicity on the rows. According to a result in [2], any linear combination of two vectors which identify two different borders of a picture, and which fall inside the picture, is again a vector that identifies a border of the picture. The two borders may yield a periodicity on the rows depending on relative divisibility properties among the size of $p$ and of its borders. Similar reasonings apply to self-covering pictures of type (3).

Let us illustrate these ideas in the following example.

**Example 13.** Consider a picture $p \in \Sigma^{**}$ of size $(4, 5)$.

Suppose that $p$ has a self-covering of type (2) with $p = p_L \oplus (p_{R1} \ominus p_{R2})$ where $size(p_L) = (4, 3)$, $size(p_{R1}) = (1, 2)$, and $size(p_{R2}) = (3, 2)$. The vectors which specify the three borders are: $(2, 0)$, $(3, 1)$, and $(3, -3)$. Applying the above cited result in [2], the vector $(0, 2) = 8(3, 1) + 2(3, -3) - 15(2, 0)$ gives a border of $p$, too. The existence of the borders of vectors $(2, 0)$ and $(0, 2)$ implies that for some $a, b, c, d \in \Sigma$, the picture $p$ is $p = \begin{array}{|ccccc|} a & b & a & b & a \\ c & d & c & d & c \\ a & b & a & b & a \\ c & d & c & d & c \end{array}$. Actually, $p = \begin{array}{|ccccc|} a & b & a & b & a \\ b & a & b & a & b \\ a & b & a & b & a \\ b & a & b & a & b \end{array}$, because of the border of vector $(3, -3)$ that implies $ba = cd$. Note that $p$ can be obtained as a repetition of its prefix $\begin{array}{|cc|} a & b \\ b & a \end{array}$. The picture has even more self-coverings, among which, there is the self-covering of type (3) defined by the vectors $(3, 1)$, $(3, -3)$, $(2, 2)$, and $(2, -2)$.

Suppose now that $p$ has a self-covering of type (2) with $p = p_L \oplus (p_{R1} \ominus p_{R2})$ where $size(p_L) = (4, 3)$, $size(p_{R1}) = size(p_{R2}) = (2, 2)$. The vectors which identify the three borders are $(2, 0)$, $(3, 2)$, and $(3, -2)$. The existence of the border

**Fig. 6.** Two different cylindric tiling decompositions of $p$.

of vector $(2,0)$, implies that $p = z \oplus z \oplus z'$, where $z$ is the prefix of $p$ of size $(4,2)$, and $z'$ is the prefix of $p$ of size $(4,1)$ (the first column). In this case, there is no linear combination of the three vectors which gives a vector $(0,x)$, for some $1 \le x \le 3$, and hence, no periodicity on the rows is necessarily implied by this self-covering. In fact, there exist no integers $\alpha$, $\beta$, $\gamma$ such that $\alpha(3,2) + \beta(3,-2) + \gamma(2,0) = (0,x)$ for some $1 \le x \le 3$. Otherwise, we have that $3(\alpha+\beta) = -2\gamma$ and $2(\alpha-\beta) = x$. These conditions imply that $\alpha+\beta$ and $x$ must be both even. It follows that $x = 2$, $\alpha-\beta = 1$, $\alpha = \beta+1$, and finally $\alpha+\beta = 2\beta+1$ which cannot be even.

## 5. Comma-free codes and cylindric codes

In the theory of string codes, comma-free codes are studied inside the class of circular codes.

Recal that $X \subseteq \Sigma^*$ is a circular code of strings if for all $m, n \ge 1$ and $x_1, x_2, \ldots, x_n \in X$, $y_1, y_2, \ldots, y_m \in X$, $t \in \Sigma^*$ and $s \in \Sigma^+$, the equalities $sx_2 \ldots x_n t = y_1 y_2 \ldots y_m$, $x_1 = ts$ imply that $t$ is the empty string, $m = n$ and $x_i = y_i$ for $1 \le i \le n$.

The translation of the definition of circular code of strings into the two-dimensional world leads to some new situations. The role of a circle can be played in 2D by a cylinder, either horizontally or vertically placed. Then, a set $X$ of pictures is a cylindric code if the pictures of $X$ cannot tile the lateral surface of any cylinder (for any height and radius) in two different ways. The definitions that follow can be thought of as referring to a cylinder with its base placed on a horizontal plane. In an analogous way, one could give the definitions in the case of a cylinder with its base placed on a vertical plane.

Let us introduce the following notations.

Given a picture $p$ of size $(m,n)$ an *across subdomain* of $dom(p)$ is a set $d$ of the form $\{i, i+1, \ldots, i'\} \times \{j, j+1, \ldots, n, 1, 2, \ldots, j'\}$, where $1 \le i \le i' \le |p|_{row}$, $1 \le j' < j \le |p|_{col}$; $d$ will be denoted by the pair $[(i,j),(i',j')]$. In order to stress the difference between an across subdomain and a subdomain as defined before (in Section 2), the latter will be sometimes called an *internal subdomain*. The portion of $p$ corresponding to the positions in an across subdomain $[(i,j),(i',j')]$ is denoted by $p[(i,j),(i',j')]$. It is a union of two (internal) subdomains.

**Definition 14.** The *cylindric tiling star* of $X$, denoted by $X^{cyl}$, is the set of pictures $p$ whose domain can be partitioned into disjoint internal and/or across subdomains $\{d_1, d_2, \ldots, d_k\}$ such that any subpicture $p_h$ of $p$ associated with the subdomain $d_h$ belongs to $X$, for all $h = 1, \ldots, k$.

In the sequel, if $p \in X^{cyl}$, the partition $\{d_1, d_2, \ldots, d_k\}$ of $dom(p)$, together with the corresponding pictures $\{p_1, p_2, \ldots, p_k\}$, is called a *cylindric tiling decomposition* of $p$ in $X$. Moreover if picture $p \in X^{cyl}$, we will say that $p$ is *cylindrically tilable* in $X$.

**Example 15.** Consider the language

$$X = \left\{ \boxed{a\ b},\ \boxed{a\ a\ b\ a},\ \boxed{\begin{array}{ccc} a & a & a \\ b & a & b \end{array}},\ \boxed{\begin{array}{ccc} a & a & a \\ b & b & a \end{array}},\ \boxed{\begin{array}{cccc} a & a & a & b \\ b & a & b & a \end{array}} \right\}.$$

The picture $p = \boxed{\begin{array}{ccccccc} a & a & a & a & a & a & b \\ b & b & a & b & a & b & a \end{array}}$ of size $(2,7)$ has two different cylindric tiling decompositions in $X$ (see Fig. 6).

A first cylindric tiling decomposition of $p$ in $X$ is the one given by the partition of $dom(p)$, $t_1 = \{d_1, d_2\}$, with $d_1 = \{1,2\} \times \{1,2,3\}$ and $d_2 = \{1,2\} \times \{4,5,6,7\}$. Note that the subdomains $d_1$ and $d_2$ are both internal subdomains. The pictures $p[(1,1),(2,3)]$ corresponding to $d_1$ and $p[(1,4),(2,7)]$ corresponding to $d_2$ both belong to $X$.

A second cylindric tiling decomposition of $p$ in $X$ is the one given by the partition of $dom(p)$, $t_2 = \{s_1, s_2, s_3, s_4\}$, with $s_1 = \{1,2\} \times \{2,3,4\}$, $s_2 = \{2\} \times \{5,6\}$, $s_3 = \{1\} \times \{5,6,7,1\}$, and $s_4 = \{2\} \times \{7,1\}$. Note that the subdomains $s_1$ and $s_2$ are both internal subdomains, while $s_3$ and $s_4$ are across subdomains. The pictures $p[(1,2),(2,4)]$ corresponding to $s_1$, $p[(2,5),(2,6)]$ corresponding to $s_2$, $p[(1,5),(1,1)]$ corresponding to $s_3$, and $p[(2,7),(2,1)]$ corresponding to $s_4$, all belong to $X$.

We are now ready to introduce the definition of cylindric code.

**Definition 16.** A language $X \subseteq \Sigma^{**}$ is a *cylindric code* if any picture in $\Sigma^{**}$ has at most one cylindric tiling decomposition in $X$.

**Remark 17.** Any comma-free code of pictures is a cylindric code of pictures.

The converse does not hold, as it is shown in the following example.

**Example 18.** Consider the language $X_1 = \left\{ \boxed{\begin{matrix} a & b \\ b & a \end{matrix}} \right\}$. It is a cylindric code. Actually, the picture $x_1 = \boxed{\begin{matrix} a & b \\ b & a \end{matrix}}$ can be covered by four copies of itself, but this cannot yield two different cylindric tiling decompositions in $X_1$, due to size motivations. Note that $X_1$ is not comma-free since $x_1$ is a self-covering picture.

As another example consider the language $X_2 = \left\{ \boxed{\begin{matrix} b & a \\ b & b \end{matrix}}, \boxed{\begin{matrix} a & a & a \\ b & b & b \\ a & b & b \end{matrix}} \right\}$. $X_2$ is not comma-free since the first picture can be covered by two copies of the second one. Let us show that $X_2$ is a cylindric code. Denote $x_2 = \boxed{\begin{matrix} b & a \\ b & b \end{matrix}}$ and $x_3 = \boxed{\begin{matrix} a & a & a \\ b & b & b \\ a & b & b \end{matrix}}$. First, note that $x_3$ cannot be covered by $X_2$. On the other hand, $x_2$ can be covered by $X_2$ either by two copies of $x_3$ or by two copies of $x_3$ and one copy of itself. None of these coverings of $x_2$ can yield two different cylindric tiling decompositions of some picture in $\Sigma^{**}$, because they should imply a covering of $x_3$ (that does not exist).

Let us now consider some decidability issues. In a simple way, it is decidable whether a finite set of pictures $X$ is a comma-free code. On the other hand, the situation is more involved in the case of cylindric codes.

**Proposition 19.** *It is decidable whether a finite set of pictures $X$ is a comma-free code.*

**Proof.** A set $X$ is not a comma-free code if and only if there exists $x \in X$ such that $x$ is covered by $X$. Hence, one has to check that each $x \in X$ is not covered by $X$. If $X$ is finite there exists an algorithm that checks the property. The algorithm follows the steps sketched in Example 7. It starts on the top-left corner of each picture in $X$ and tries to cover a prefix of the picture with a picture in $X$. Then, it looks for any possibility to place thereby a picture of $X$ in a way to cover a next uncovered position; and so on, until it succeeds or fails and backtracks. □

The *Cylindric-Codicity Problem* is the problem to decide whether a given (finite) set $X \subseteq \Sigma^{**}$ of pictures is a cylindric code or not. In the next proposition, we will show that the Cylindric-Codicity Problem is undecidable by using the well-known undecidability of the Thue System Word Problem. Let us recall the problem (see also [12]).

Consider a finite alphabet $\Sigma$ and a finite subset $S \subseteq \Sigma^* \times \Sigma^*$. The elements of $S$ are called relations. A *Thue System* is the pair $(\Sigma, S)$. For $u, v \in \Sigma^*$, we write $u \sim_S v$ if there exists a relation $(s,t) \in S$ or $(t,s) \in S$ such that $u = xsy$ and $v = xty$ for some $x, y \in \Sigma^*$. By $\equiv_S$ we denote the transitive closure of $\sim_S$. The *Thue System Word Problem* is: given a Thue System $(\Sigma, S)$, and $u, v \in \Sigma^*$, does $u \equiv_S v$ hold?

**Proposition 20.** *The* Cylindric-Codicity Problem *is undecidable.*

**Proof.** Let us show that the Thue System Word Problem reduces to the *Cylindric-Codicity Problem*. The well-known undecidability of the Thue System Word Problem (see e.g. [12]) implies the undecidability of the *Cylindric-Codicity Problem*.

Taking inspiration from a construction in [26], for a given Thue System $(\Sigma, S)$, where $S \subseteq \Sigma^* \times \Sigma^*$ is the set of relations, and words $u, v \in \Sigma^*$, construct the set $X_{S,u,v}$ of square pictures over the alphabet $\Sigma_1 = \Sigma \cup \{\star, \rhd, \odot\}$, as follows.

For each relation $(s_1 s_2 \ldots s_n, t_1 t_2 \ldots t_m) \in S$ insert in $X_{S,u,v}$ the two following (square) pictures. Note that if one of the words is shorter, it is left-padded with $\star$ symbol (as in the following figure where $n < m$).

$$\boxed{\begin{matrix} \odot & \star & \ldots & \star & s_1 & \star & s_2 & \star & \ldots & \star & s_n \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \star & t_1 & \star & t_2 & \star & \ldots & \ldots & \ldots & \ldots & \star & t_m \end{matrix}} \quad \boxed{\begin{matrix} \odot & t_1 & \star & t_2 & \star & \ldots & \ldots & \ldots & \ldots & \star & t_m \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \star & \ldots & \ldots & \star & s_1 & \star & s_2 & \star & \ldots & \star & s_n \end{matrix}}$$

Then, insert in $X_{S,u,v}$ the following (square) pictures associated with $u = u_1 u_2 \ldots u_i$ and $v = v_1 v_2 \ldots v_j$.

$$\boxed{\begin{matrix} \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \star & \rhd & \star & u_1 & \star & u_2 & \star & \ldots & \star & u_i \end{matrix}} \quad \boxed{\begin{matrix} \star & \rhd & \star & v_1 & \star & v_2 & \star & \ldots & \star & v_j \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \star & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \star \end{matrix}}$$

For each symbol $a \in \Sigma \cup \{\rhd\}$ insert in $X_{S,u,v}$ the pictures

$$\boxed{\begin{matrix} \star & \star & \star & a \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & a & \star & \star \end{matrix}} \text{ and } \boxed{\begin{matrix} \star & a \\ \star & a \end{matrix}}$$

and for each symbol $a \in \Sigma$ add the picture

$$\begin{array}{|cc|} \hline \star & a \\ \odot & a \\ \hline \end{array}.$$

Finally insert in $X_{S,u,v}$ the pictures $\begin{array}{|c|} \hline \star \\ \hline \end{array}$ and $\begin{array}{|c|} \hline \odot \\ \hline \end{array}$. Using considerations analogous to those in [26], one can show that $u$ and $v$ are equivalent in $S$ if and only if $X_{S,u,v}$ is not a cylindric code.

Indeed, suppose that $u$ and $v$ are equivalent in $S$. Then, a successful derivation in the Thue System corresponds to a series of bricks, stacked vertically in layers: a first layer for one of the input words, some bricks corresponding to the first derivation step, some bricks corresponding to the second step, and so on, and at the end a layer for the other input word. Rewriting bricks are used to shift symbols to the left and to rewrite symbols that are not changed at each step. Now, note that a tiling described as above can also be constructed using only the symbol-rewriting bricks and the fillers (see also Example 21 and Fig. 7). This implies non-codicity of the set.

Suppose now that $u$ and $v$ are not equivalent in $S$. In this case, it is not possible to complete the construction of two different tiling decompositions of a picture $p$ on $X_{S,u,v}$, as sketched above (see Fig. 7), and one can see (similarly to [26]) that $X_{S,u,v}$ is a picture code. Moreover, it is possible to show that there is no picture with two different cylindric tiling decompositions on $X_{S,u,v}$, i.e. $X_{S,u,v}$ is a cylindric code. Then, observe that a picture has two different cylindric tiling decompositions (which are not tiling decompositions) on a language if and only if there exist two pictures $p$ and $q$ of the language that properly overlap with their first rows aligned. In the case of the language $X_{S,u,v}$, this can happen only if $p = \begin{array}{|ccccccccc|} \hline \star & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \star & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \star & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \star & \triangleright & \star & u_1 & \star & u_2 & \star \ \cdots & \star & u_i \\ \hline \end{array}$ and $q$ is either $\begin{array}{|ccccccccc|} \hline \star & \triangleright & \star & v_1 & \star & v_2 & \star \ \cdots & \star & v_j \\ \star & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \star & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \star & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \hline \end{array}$ with $j < i$, or $\begin{array}{|cccc|} \hline \star & \star & \star & a \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & a & \star & \star \\ \hline \end{array}$ or $\begin{array}{|cc|} \hline \star & a \\ \star & a \\ \hline \end{array}$ with $a \in \Sigma \cup \{\triangleright\}$.

In each case, there is no possibility to cover with pictures in $X_{S,u,v}$ the remaining part of $q$:

$$\begin{array}{|cccccccc|} \hline \triangleright & \star & v_1 & \star & v_2 & \star \ \cdots & \star & v_j \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \star \\ \hline \end{array} \text{ or } \begin{array}{|ccc|} \hline \star & \star & a \\ \star & \star & \star \\ \star & \star & \star \\ a & \star & \star \\ \hline \end{array} \text{ or } \begin{array}{|c|} \hline a \\ a \\ \hline \end{array}. \qquad \square$$

To clarify the idea of the reduction of the Thue System Word Problem to the Cylindric-Codicity Problem, let us show an example.

**Example 21.** Let $S = \{(ab, b), (aa, bb), (aa, a)\}$, $u = bab$, and $v = a$. The set $X_{S,u,v}$ consists of the following pictures.

Bricks corresponding to the relations in $S$:

$$\begin{array}{|cccc|} \hline \odot & a & \star & b \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & \star & \star & b \\ \hline \end{array}, \begin{array}{|cccc|} \hline \odot & \star & \star & b \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & a & \star & b \\ \hline \end{array}, \begin{array}{|cccc|} \hline \odot & a & \star & a \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & b & \star & b \\ \hline \end{array}, \begin{array}{|cccc|} \hline \odot & b & \star & b \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & a & \star & a \\ \hline \end{array}, \begin{array}{|cccc|} \hline \odot & a & \star & a \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & \star & \star & a \\ \hline \end{array}, \begin{array}{|cccc|} \hline \odot & \star & \star & a \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & a & \star & a \\ \hline \end{array}$$

Bricks corresponding to the input words:

$$\begin{array}{|cccccccc|} \hline \star & \star & \star & \star & \star & \star & \star & \star \\ \star & \star & \star & \star & \star & \star & \star & \star \\ \star & \star & \star & \star & \star & \star & \star & \star \\ \star & \star & \star & \star & \star & \star & \star & \star \\ \star & \star & \star & \star & \star & \star & \star & \star \\ \star & \star & \star & \star & \star & \star & \star & \star \\ \star & \star & \star & \star & \star & \star & \star & \star \\ \star & \triangleright & \star & b & \star & a & \star & b \\ \hline \end{array}, \begin{array}{|cccc|} \hline \star & \triangleright & \star & a \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \hline \end{array}$$

Rewriting bricks and the fillers:

$$\begin{array}{|cccc|} \hline \star & \star & \star & a \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & a & \star & \star \\ \hline \end{array} \begin{array}{|cccc|} \hline \star & \star & \star & b \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & b & \star & \star \\ \hline \end{array} \begin{array}{|cccc|} \hline \star & \star & \star & \triangleright \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ \star & \triangleright & \star & \star \\ \hline \end{array} \begin{array}{|cc|} \hline \star & a \\ \star & a \\ \hline \end{array} \begin{array}{|cc|} \hline \star & b \\ \star & b \\ \hline \end{array} \begin{array}{|cc|} \hline \star & a \\ \odot & a \\ \hline \end{array} \begin{array}{|cc|} \hline \star & b \\ \odot & b \\ \hline \end{array} \begin{array}{|cc|} \hline \star & \triangleright \\ \star & \triangleright \\ \hline \end{array} \begin{array}{|c|} \hline \star \\ \hline \end{array} \begin{array}{|c|} \hline \odot \\ \hline \end{array}.$$

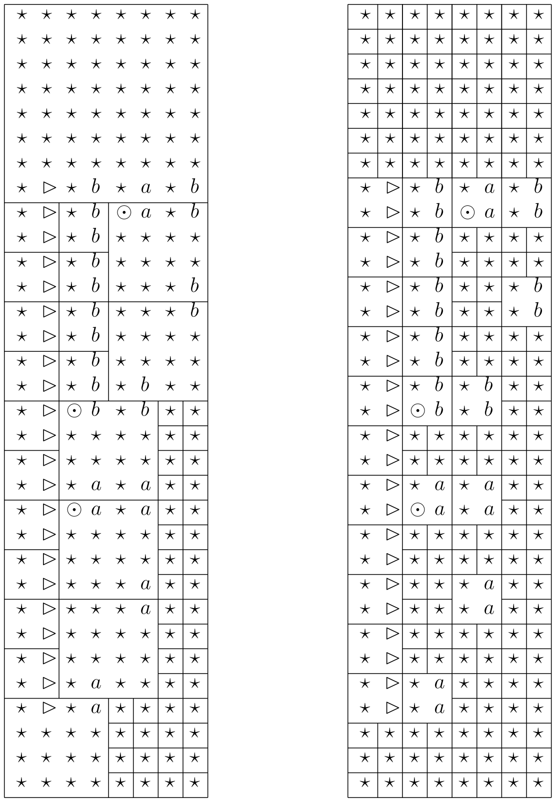


**Fig. 7.** A picture with two different (cylindric) tiling decompositions on the set $X_{S,u,v}$ in Example 21.

We ask whether $bab \equiv_S a$. The answer is yes, since there is the following derivation $b\underline{ab} \sim_S \underline{bb} \sim_S \underline{aa} \sim_S a$, where the underlined subwords are the ones being substituted at each step. In correspondence to this derivation one can find a picture with two different (cylindric) tiling decompositions, as shown in Fig. 7. Hence, the set of pictures $X_{S,u,v}$ is not a cylindric code.

## 6. Maximality and completeness

In theory of string languages, the notion of maximality of codes is related to the one of completeness. A string language $S \subseteq \Sigma^*$ is defined complete if the set of all factors of $S^*$ is equal to $\Sigma^*$. Then a thin code is maximal if and only if it is complete. We refer the reader to [10] for all the definitions and results.

In the framework of pictures, completeness has been defined along a corner-to-corner direction in [4]; br-complete sets of pictures are compared to maximal prefixes codes. It holds that the br-completeness of a prefix set of pictures implies its prefix maximality, but it is not its characterization.

In this paper, we consider comma-free and cylindric codes which are based on a non-oriented definition of codicity. Hence, let us introduce a (new) notion of completeness for the 2D languages and investigate the relationship of completeness with maximality for these new families of codes.

**Definition 22.** A comma-free code $X \subseteq \Sigma^{**}$ is *maximal comma-free* over $\Sigma$ if it is not properly contained in any other comma-free code over $\Sigma$; that is, if $X \subseteq Y \subseteq \Sigma^{**}$ and $Y$ is a comma-free code, then $X = Y$.

A cylindric code $X \subseteq \Sigma^{**}$ is *maximal cylindric* over $\Sigma$ if it is not properly contained in any other cylindric code over $\Sigma$; that is, if $X \subseteq Y \subseteq \Sigma^{**}$ and $Y$ is a cylindric code, then $X = Y$.

**Definition 23.** A set $X \subseteq \Sigma^{**}$ is *complete* if any picture in $\Sigma^{**} \setminus X$ is covered by $X$.

**Proposition 24.** *Let $X \subseteq \Sigma^{**}$ be a cylindric code. If $X$ is maximal cylindric then $X$ is complete.*

**Proof.** Let $X$ be a maximal cylindric code and suppose by contradiction that there exists $p \in \Sigma^{**} \setminus X$ which is not covered by $X$ (i.e. $X$ is not complete).

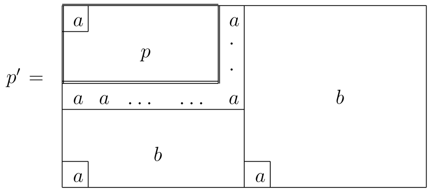



**Fig. 8.** The unbordered picture $p'$ in the proof of Proposition 24.

If $p$ is unbordered then $X \cup \{p\}$ is a cylindric code, contradicting the maximality of $X$. Indeed, the existence of two distinct cylindric tilings with pictures in $X \cup \{p\}$ including $p$, implies that $p$ is covered by $X \cup \{p\}$. This is a contradiction because $p$ is not covered by $X$ and it is unbordered.

Consider now the case that $p$ is bordered. Suppose $size(p) = (m, n)$, let $p(1, 1) = a$ and let $b \in \Sigma \setminus \{a\}$. Define the picture $p'$ (see Fig. 8) obtained from $p$ by the following operations: surround $p$ with a row and a column of $a$; then row concatenate a picture of size $(m, n + 1)$ filled by $b$, except for its bl-corner; finally column concatenate a picture of size $(2m + 1, n + 1)$ filled by $b$, except for its bl-corner. The resulting picture is a picture $p'$ of size $(2m + 1, 2n + 2)$ having $p$ as a prefix. It is possible to claim that $p'$ is unbordered, by focusing on the possible positions of the subpicture $p[(1, n + 1), (m + 1, n + 1)]$ when $p$ hypothetically self overlaps.

We claim that $X \cup \{p'\}$ is a cylindric code, contradicting the maximality of $X$.

Assuming the contrary, if $X \cup \{p'\}$ is not a cylindric code then $p'$ would be covered by $X \cup \{p'\}$. Furthermore, since $p'$ is unbordered, $p'$ would be covered by $X$. Finally, also $p$ which is a prefix of $p$, would be covered by $X$, contradicting the hypothesis. □

**Proposition 25.** *Let $X \subseteq \Sigma^{**}$ be a comma-free code. If $X$ is complete then $X$ is maximal comma-free.*

**Proof.** Suppose that $X$ is complete but not maximal comma-free. Then there exists a picture $p \in \Sigma^{**} \setminus X$, such that $X \cup \{p\}$ is comma-free. In particular $p$ is not covered by $X$. This contradicts the completeness of $X$. □

Let us now state some properties of complete sets. We will simply denote by $a^{m,n}$ the picture of size $(m, n)$ over $\{a\}$.

**Proposition 26.** *Let $X \subseteq \Sigma^{**}$ be a finite set of pictures. If $X$ is complete then for any $a \in \Sigma$ there exist $m, n > 0$ such that $a^{m,n} \in X$.*

**Proof.** If $X$ is finite then denote by $r_X$ ($c_X$, respectively) the maximum number of rows (columns, respectively) of a picture in $X$. Consider the picture $p = a^{2r_X, 2c_X}$. Since $X$ is complete, $p$ is covered by $X$. Therefore there exist $m, n > 0$ such that $a^{m,n} \in X$. □

**Proposition 27.** *Let $X \subseteq \Sigma^{**}$ be a cylindric code which is complete. Then either $X = \Sigma$ or $X$ is infinite.*

**Proof.** According to Proposition 26, if $X$ is finite then, for any $a \in \Sigma$, there exist $m, n > 0$ such that $a^{m,n} \in X$. Since $X$ is a cylindric code, we must have $m = n = 1$, that is $X = \Sigma$. □

**Remark 28.** Any string $w \in \Sigma^*$ can be viewed as a one-row picture $pict(w)$ that has $w$ in its (first) row. Denote by $pict(X)$, the set $pict(X) \subseteq \Sigma^{1*}$ of the one-row pictures corresponding to the strings in $X$. The properties of being a comma-free or a cylindric code are preserved in this translation, in the sense that one can easily show that $X$ is comma-free if and only if $pict(X)$ is comma-free, $X$ is circular if and only if $pict(X)$ is cylindric. Furthermore, $X$ is complete if and only if $pict(X)$ is complete.

**Remark 29.** Let $pict(X)$ be the language of one-row pictures corresponding to a set $X$ of strings. If $pict(X)$ is maximal cylindric then $X$ is maximal circular. If $pict(X)$ is maximal comma-free then $X$ is maximal comma-free, but the converse does not hold. Indeed, consider the set $X \subseteq \Sigma^*$, $X = \{bab\} \cup \{a^2b^ia^jb : i, j \geq 2\} \cup \{a^2ba^ib : i \geq 3\}$. In [22], it is shown that $X$ is a maximal comma-free code of strings. Consider now the picture $p = pict(r_1) \ominus pict(r_2)$ with $r_1 = aabaaaab$ and $r_2 = abaabaab$. Note that $r_1 \in X$ whereas $r_2 \notin X$ and $r_2$ cannot be covered by $X$. It is easy to prove that $pict(X) \cup \{p\}$ is a comma-free code and therefore $pict(X)$ is not maximal comma-free.

## 7. Comparing the families of codes

We are going to compare all the classes of two-dimensional codes considered in the previous sections. Let us denote them as follows.

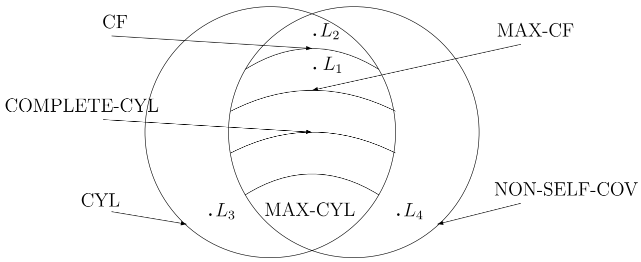



**Fig. 9.** Relations among the families of two-dimensional codes considered in the paper.

MAX-CYL is the class of all maximal cylindric codes.
COMPLETE-CYL is the class of all cylindric codes that are also complete sets.
MAX-CF is the class of all maximal comma-free codes.
CF is the class of all comma-free codes.
CYL is the class of all cylindric codes.
NON-SELF-COV is the class of languages which contain only non-self-covering pictures.

**Proposition 30.** *MAX-CYL ⊆ COMPLETE-CYL ⊆ MAX-CF ⊂ CF ⊂ CYL.*

**Proof.** The inclusions are shown in Propositions 24, 25, and Remark 17.

The proper inclusions come from the following examples of languages.

An example of a language in CF\MAX-CF can be obtained from the language $X$ in Example 7, as $X \setminus \{p\}$, for any $p \in X$.

Some examples of languages in CYL\CF are the sets $X_1$ and $X_2$ in Example 18. □

**Proposition 31.** *CF ⊂ CYL ∩ NON-SELF-COV.*

**Proof.** The inclusion CF ⊂ CYL is stated in Remark 17 and, moreover, a picture in a comma-free code cannot be self-covering. The strict inclusion is proved by the following language

$$X = \left\{ \boxed{\begin{array}{cc} a & b \end{array}}, \ \boxed{\begin{array}{cccc} a & a & a & b \\ b & a & b & a \end{array}} \right\}. \qquad \square$$

We are now able to show the following relations among all the classes of two-dimensional sets considered in this paper, as summarized in Fig. 9.

The languages $L_1$, $L_2$, $L_3$, and $L_4$, which witness the strict inclusions, in Fig. 9, can be chosen as follows.

$$L_1 = \left\{ \boxed{\begin{array}{ccc} a & b & b \\ a & a & b \\ b & a & a \end{array}}, \ \boxed{\begin{array}{cccccc} b & a & b & a & b & b \end{array}} \right\}$$

$$L_2 = \left\{ \boxed{\begin{array}{cc} a & b \end{array}}, \ \boxed{\begin{array}{cccc} a & a & a & b \\ b & a & b & a \end{array}} \right\}$$

$$L_3 = \left\{ \boxed{\begin{array}{cccc} a & a & a & b \\ a & a & a & b \end{array}} \right\}$$

$$L_4 = \left\{ \boxed{\begin{array}{cc} a & b \end{array}}, \ \boxed{\begin{array}{cccc} a & a & b & a \end{array}}, \ \boxed{\begin{array}{ccc} a & a & a \\ b & a & b \end{array}}, \ \boxed{\begin{array}{ccc} a & a & a \\ b & b & a \end{array}}, \ \boxed{\begin{array}{cccc} a & a & a & b \\ b & a & b & a \end{array}} \right\}.$$

Note that the language $L_1$ can be obtained from the language $X$ in Example 7, by removing a picture.

The language $L_2$ is the language in the proof of Proposition 31.

It is easy to see that $L_3 \in$ CYL\NON-SELF-COV.

Finally, the language $L_4 \in$ NON-SELF-COV, and $L_4 \notin$CYL since the picture $p = \boxed{\begin{array}{ccccccc} a & a & a & a & a & a & b \\ b & b & a & b & a & b & a \end{array}}$ has two different cylindric tiling decompositions in $L_4$ (see Example 15 noting that, in that example, $X = L_4$).

## 8. Conclusions and open questions

In this paper we have introduced two families of two-dimensional codes of pictures – comma-free and cylindric codes. A main feature is that their definition is "non-oriented", in the sense that they are not related to a leading direction of decoding. Subsequently, we have studied decidability and maximality issues. At the end, we have collected all the considered families and shown their relations with respect to inclusion and intersection. Most of the classes are separated by counter-examples. We leave open whether the class of complete comma-free codes coincides with the class of maximal comma-free or maximal cylindric codes.

Another family of codes to compare with is the class of prefix codes introduced in [4]. Any comma-free code is a prefix code; moreover all its rotations are prefix codes, too. This suggests the investigation of such a special class of codes which are prefix along the four corner-to-corner directions; say "quadrifix" codes, as a generalization of the bifix codes of strings.

A remarkable issue is that it is decidable whether a finite language is a comma-free code, whereas it is undecidable for (finite) cylindric codes. An interesting question is the decidability of the maximality of a (finite) comma-free code. Recall that the question is solved for the languages of strings, although the completion of (finite and regular) comma-free codes of strings is not a simple process (see [22–24]).

Finally, cylindric codes are a two-dimensional generalization of circular codes of strings. A further generalization is to consider not only the tiling of the surface of a cylinder, but also the tiling of the surface of a torus. But then, a major effort of imagination would be needed!

## Acknowledgements

We are grateful to Antonio Restivo for having suggested us to investigate this topic, and for the plenty of suggestions, hints and advice that he always heartily gave us.

## References

[1] P. Aigrain, D. Beauquier, Polyomino tilings, cellular automata and codicity, Theoret. Comput. Sci. 147 (1995) 165–180.
[2] A. Amir, G. Benson, Two-dimensional periodicity in rectangular arrays, SIAM J. Comput. 27 (1) (1998) 90–106.
[3] M. Anselmo, D. Giammarresi, M. Madonia, Picture codes and deciphering delay, Inform. and Comput. (2016), in press.
[4] M. Anselmo, D. Giammarresi, M. Madonia, Prefix picture codes: a decidable class of two-dimensional codes, Int. J. Found. Comput. 25 (8) (2014) 1017–1031, World Scientific Publishing Company.
[5] M. Anselmo, D. Giammarresi, M. Madonia, Unbordered pictures: properties and construction, in: A. Maletti (Ed.), Procs. CAI 2015, in: LNCS, vol. 9270, Springer-Verlag, Berlin, Heidelberg, 2015, pp. 45–57.
[6] M. Anselmo, D. Giammarresi, M. Madonia, Structure and properties of strong prefix codes of pictures, Math. Struct. Comput. Sci., http://dx.doi.org/10.1017/S0960129515000043, available on CJO 2015.
[7] M. Anselmo, D. Giammarresi, M. Madonia, A. Restivo, Unambiguous recognizable two-dimensional languages, RAIRO Theor. Inform. Appl. 40 (2) (2006) 227–293, EDP Sciences.
[8] M. Anselmo, N. Jonoska, M. Madonia, Framed versus unframed two-dimensional languages, in: M. Nielsen, et al. (Eds.), Procs. SOFSEM 09, in: LNCS, vol. 5404, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 79–92.
[9] D. Beauquier, M. Nivat, A codicity undecidable problem in the plane, Theoret. Comput. Sci. 303 (2003) 417–430.
[10] J. Berstel, D. Perrin, C. Reutenauer, Codes and Automata, Cambridge University Press, 2009.
[11] M. Blum, C. Hewitt, Automata on a two-dimensional tape, IEEE Symposium on Switching and Automata Theory (1967) 155–160.
[12] R.V. Book, F. Otto, String-Rewriting Systems, Springer-Verlag, 1993.
[13] S. Bozapalidis, A. Grammatikopoulou, Picture codes, RAIRO Theor. Inform. Appl. 40 (4) (2006) 537–550.
[14] A.L. Churchill, Restrictions and generalizations on comma-free codes, Electron. J. Combin. 16 (1) (2009).
[15] Z. Galil, K. Park, Truly alphabet independent two-dimensional pattern matching, in: Proc. 33rd IEEE Symposium on Foundations of Computer Science, IEEE Computer Society Press, Los Alamitos, CA, 1992, pp. 247–256.
[16] G. Gamard, G. Richomme, Coverability in two dimensions, in: A.-H. Dediu, et al. (Eds.), Procs. LATA 2015, in: LNCS, vol. 8977, Springer, 2015, pp. 402–413.
[17] D. Giammarresi, A. Restivo, Recognizable picture languages, Int. J. Pattern Recognit. 6 (2–3) (1992) 241–256.
[18] D. Giammarresi, A. Restivo, Two-dimensional languages, in: G. Rozenberg, et al. (Eds.), Handbook of Formal Languages, vol. III, Springer Verlag, 1997, pp. 215–268.
[19] S.W. Golomb, B. Gordon, L.R. Welch, Comma-free codes, Canad. J. Math. 10 (1958) 202–209.
[20] J. Kari, M. Szabados, An algebraic geometric approach to Nivat's conjecture, ICALP 2 (2015) 273–285.
[21] M. Kolarz, W. Moczurad, Multiset, set and numerically decipherable codes over directed figures, in: Combinatorial Algorithms, 23rd International Workshop IWOCA, in: LNCS, vol. 7643, Springer, 2012, pp. 224–235.
[22] N.H. Lam, Completing comma-free codes, Theoret. Comput. Sci. 301 (1–3) (2003) 399–415.
[23] N.H. Lam, Finite completion of comma-free codes. Part 1, RAIRO Theor. Inform. Appl. 38 (2) (2004) 91–115.
[24] N.H. Lam, Finite completion of comma-free codes. Part 2, RAIRO Theor. Inform. Appl. 38 (2) (2004) 117–136.
[25] F. Mignosi, A. Restivo, P.V. Silva, On Fine and Wilf's theorem for bidimensional words, Theoret. Comput. Sci. 292 (1) (2003) 245–262.
[26] M. Moczurad, W. Moczurad, Some open problems in decidability of brick (labeled polyomino) codes, in: K.-Y. Chwa, J.I. Munro (Eds.), COCOON 2004, in: LNCS, vol. 3106, Springer-Verlag, Berlin, Heidelberg, 2004, pp. 72–81.
[27] M. Regnier, L. Rostami, A unifying look at d-dimensional periodicities and space coverings, CPM 1993 (1993) 215–227.
[28] R.A. Scholtz, Maximal and variable word-length comma-free codes, IEEE Trans. Inform. Theory IT-15 (1969) 555–559.
[29] D. Simplot, A characterization of recognizable picture languages by tilings by finite sets, Theoret. Comput. Sci. 218 (2) (1991) 297–323.