RAIRO-Inf. Theor. Appl. **40** (2006) 277-293
DOI: 10.1051/ita:2006008

UNAMBIGUOUS RECOGNIZABLE TWO-DIMENSIONAL LANGUAGES *

Marcella Anselmo[1], Dora Giammarresi[2], Maria Madonia[3] and Antonio Restivo[4]

Abstract. We consider the family UREC of unambiguous recognizable two-dimensional languages. We prove that there are recognizable languages that are inherently ambiguous, that is UREC family is a proper subclass of REC family. The result is obtained by showing a necessary condition for unambiguous recognizable languages. Further UREC family coincides with the class of picture languages defined by unambiguous 2OTA and it strictly contains its deterministic counterpart. Some closure and non-closure properties of UREC are presented. Finally we show that it is undecidable whether a given tiling system is unambiguous.

Mathematics Subject Classification. 68Q45, 68Q10.

1. Introduction

The theory of one-dimensional string languages is well founded and investigated since fifties. From several years, the increasing interest for pattern recognition and image processing has also motivated the research on *two-dimensional* or *picture*

Keywords and phrases. Automata and formal languages, unambiguity, determinism, two-dimensional languages.

* *This work was partially supported by MIUR project Linguaggi formali e automi: metodi, modelli e applicazioni.*

[1] Dipartimento di Informatica ed Applicazioni, Università di Salerno, 84081 Baronissi, (Salerno), Italy; anselmo@dia.unisa.it

[2] Dipartimento di Matematica, Università di Roma "Tor Vergata", via Ricerca Scientifica, 00133 Roma, Italy; giammarr@mat.uniroma2.it

[3] Dipartimento di Matematica e Informatica, Università di Catania, Viale Andrea Doria 6/a, 95125 Catania, Italy; madonia@dmi.unict.it

[4] Dipartimento di Matematica e Applicazioni, Università di Palermo, via Archirafi 34, 90123 Palermo, Italy; restivo@dipmat.math.unipa.it

languages, and nowadays this is a research field of great interest. A first attempt to formalize the concept of finite state recognizability for two-dimensional languages can be attributed to Blum and Hewitt [5] who started in 1967 the study of finite state devices that can define two-dimensional languages, with the aim of finding a counterpart of what regular languages are in one dimension. Since then, many approaches have been presented in the literature following all classical ways to define regular languages: finite automata, grammars, logics and regular expressions.

In 1991, a unifying point of view was presented by Restivo and Giammarresi who defined the family REC of *recognizable picture languages* (see [10, 11]). The definition of recognizable picture language takes as starting point a well known characterization of recognizable string languages in terms of local languages and projections. Namely, any recognizable string language can be obtained as projection of a local string language defined over a larger alphabet (*cf.* Th. 6.1 in [9]). Such notion can be extended in a natural way to the two-dimensional case: more precisely, local picture languages are defined by means of a set of square arrays of side-length two (called *tiles*) that represents the only allowed blocks of that size in the pictures of the language. Then, we say that a two-dimensional language is (tiling system) recognizable if it can be obtained as a projection of a local picture language. REC is the family of all recognizable two-dimensional languages. Remark that, when we consider strings as particular pictures (that is pictures in which one side has length one), this definition of recognizability coincides with the one for the strings, *i.e.* the definition given in terms of finite automata. Further the definition turns out to be robust from different points of view: REC family is characterized in terms of some kind of finite-state automata called 2-dimensional on-line tessellation automata (2OTA for short), of logic formulas and of regular expressions with alphabetic mapping and it inherits several properties from the class of regular string languages.

Finally, the approach of recognizability in terms of tiling systems is very close to that one proposed by W. Thomas in the more general context of graphs (*cf.* [24]). From the results in [12], one derives that both approaches are indeed equivalent for particular graphs corresponding to two-dimensional languages, referred to as rectangular grid graphs.

A crucial difference between the recognizability of string languages and the one of picture languages in REC arises directly from its definition. The definition of recognizability in terms of local languages and projections is implicitly non-deterministic. This can be easily understood if we refer to the one-dimensional case: if no particular constraints are given for the tiling system, this corresponds in general to a non-deterministic automaton. On the contrary to the one-dimensional case, determinism and non-determinism are no more equivalent in power of recognition for two-dimensional languages. Several definitions of deterministic models to recognize two-dimensional languages have been considered in the literature. In particular there is a deterministic on-line tessellation automaton, (see [14]) and there are various versions of deterministic graph acceptors (see [24]). All approaches lead to the result that the deterministic models recognize a smaller class of languages. This result is strengthened by another one stating a major difference

with the one-dimensional case: REC family is not closed under complementation. As a consequence of such results, we infer that it is not possible to eliminate the non-determinism from this model without losing in power of recognition (as long as deterministic versions allow complementation).

In formal language theory, an intermediate notion between determinism and non-determinism is the notion of *unambiguity*. In an unambiguous model, we require that there is at most one way to recognize an object. Both determinism and unambiguity correspond to the existence of a *unique* process of computation, but while determinism is a "local" notion, unambiguity is a "global" one. As far as regular string languages are concerned, these three notions of determinism, non-determinism and unambiguity coincide. The notion of unambiguity is widely studied also in structures more general than strings. For example, in the case of trace languages, A. Bertoni *et al.* proved that in general there is no equivalence between the family of rational trace languages and its unambiguous counterpart (*cf.* [3], see also [23]). We also remark that there are many other more general frameworks (such as grammars, for example) where the unambiguity concept naturally plays a very important role. This is especially true when the involved structures do not have a privileged direction of computation, like with pictures. Also note that this divergence between determinism and non-determinism can be overcome for tree languages by reversing the computation from top-down to bottom-up; but this cannot be avoided for grid graphs [22].

Unambiguous recognizable two-dimensional languages have been introduced in [10], and their family referred to as UREC. Informally, a picture language belongs to UREC when it admits an unambiguous tiling system, where a tiling system is unambiguous if every picture has a unique counter-image in its corresponding local language. In [10], the emptiness problem is proved undecidable for UREC and the proper inclusion of UREC in REC is conjectured. In the same years unambiguous graph automata are considered in [22], and they indeed exactly match unambiguous tiling systems. The authors also provided an example of a set of grid graphs that is unambiguous but not deterministically recognizable and stated some questions about the relations of unambiguity to recognizability. Also note that the questions about unambiguity are related to quantitative properties of languages. In one-dimension this approach is formalized considering formal power series instead of formal languages, and unambiguity corresponds to recognition by means of some device whose behavior is a characteristic series. In this framework we mention some recent papers where new notions of weighted recognizability for picture languages are considered. In [4] weighted picture automata (WPA) and weighted picture series are defined. The family of behaviors of such WPA coincides with projections of certain rational picture series and can be also characterized by using tiling and domino systems [20]. Very recently, weighted on-line tessellation automata (W2OTA) have been introduced in [21], where the author considers unambiguous recognizable picture languages in order to obtain some results relating picture series recognized by W2OTA to series definable by some weighted logic. Some other recent results on two-dimensional languages are in [1, 6, 8, 17].

The definition of unambiguous tiling system yields several questions, as the ones above mentioned. This paper affords some of them. First of all we study some properties of UREC family. We state a necessary condition for UREC family that gives an upper bound to the rank of a set of matrices associated to a language in UREC. The result is achieved considering pictures of fixed height as strings over the alphabet of columns. Then, by using this necessary condition, we answer the question whether all tiling recognizable languages admit unambiguous tiling systems. Our main result states that this is not the case: UREC is strictly included in REC. In other words, in REC there exist languages that are inherently ambiguous. We also show that UREC family is closed under rotation and intersection, but not under the classical operations of row and column concatenations and their iterations. Furthermore, we consider the problem of detecting the counterpart of UREC inside the family of languages accepted by 2OTA (recall the characterization of REC in terms of 2OTA). So we consider *unambiguous on-line tessellation automata* (2UOTA for short) and show that they accept exactly UREC family. Moreover we prove that 2UOTA are able to recognize languages that cannot be recognized by a 2DOTA. Finally we show that it is undecidable whether a given tiling system is unambiguous.

The paper is organized as follows. We start, in Section 2, by introducing some basic two-dimensional languages terminology and definitions and recalling the definitions of recognizable two-dimensional languages. Then in Section 3 we introduce the definition of unambiguous recognizable two-dimensional languages and prove our main results. Finally, Section 4 lists some concluding remarks.

2. Preliminaries

We introduce some definitions about two-dimensional languages by borrowing and extending notations from the one-dimensional theory and give formal definitions of concatenation operations between two-dimensional strings and two-dimensional languages. The notations used can be mainly found in [11].

Let Σ be a finite alphabet. A *two-dimensional string* (or a *picture*) over Σ is a two-dimensional rectangular array of elements of Σ. The set of all two-dimensional strings over Σ is denoted by Σ^{**} and a *two-dimensional language* over Σ is a subset of Σ^{**}. Given a picture $p \in \Sigma^{**}$, let $p_{i,j}$ denote the symbol in p with coordinates (i, j), let $\ell_1(p)$ denote the number of rows of p and $\ell_2(p)$ denote the number of columns of p. The pair $(\ell_1(p), \ell_2(p))$ of dimensions of p is called the *size* of the picture p. The set of all two-dimensional strings over Σ of size (m, n) is denoted by $\Sigma^{m,n}$. Note that unlike the one-dimensional case, we can define an infinite number of empty pictures namely all the pictures of size $(n, 0)$ and of size $(0, m)$, for all $m, n \geq 0$, that we call *empty columns* and *empty rows*, and denote by $\lambda_{n,0}$ and $\lambda_{0,m}$ respectively.

We now recall the classical concatenation operations between pictures and picture languages. Let p and q be two pictures over an alphabet Σ, of size (m, n) and

(m', n'), $m, n, m', n' \geq 0$, respectively. The *column concatenation* of p and q (denoted by $p \oplus q$) 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 and are given by:

$p \oplus q =$ | p | q |

$p \ominus q =$ | p |
|---|
| q |

As in the string language theory, these definitions of pictures concatenation can be extended to define concatenations between set of pictures. By iterating the concatenation operations, we obtain the columns and rows *closure* or *star*.

In order to describe scanning or recognizing strategies for pictures, it is often needed to identify the symbols on the boundary. Then, for any picture p of size (m, n), we consider picture $\widehat{p}$ of size $(m+2, n+2)$ obtained by surrounding p with a special *boundary symbol* $\# \notin \Sigma$.

2.1. Recognizable two-dimensional languages

We now recall definition and properties of (tiling) recognizable two-dimensional languages. For more details see [11].

We call *tile* a square picture of dimension $(2, 2)$ and given a picture x we denote by $B_{2,2}(x)$ the set of all blocks of x of size $(2, 2)$. A two-dimensional language $L \subseteq \Gamma^{**}$ is *local* if there exists a finite set Θ of tiles over the alphabet $\Gamma \cup \{\#\}$ (the set of *allowed blocks*) such that $L = \{x \in \Gamma^{**} | B_{2,2}(\widehat{x}) \subseteq \Theta\}$ and we will write $L = L(\Theta)$.

Let Γ and Σ be two finite alphabets. A mapping $\pi : \Gamma \to \Sigma$ will be in the sequel called *projection*. The projection $\pi(x)$ of $x \in \Gamma^{**}$ is the picture $x' \in \Sigma^{**}$ such that $x'_{ij} = \pi(x_{ij})$ for all $1 \leq i \leq \ell_1(x)$, $1 \leq j \leq \ell_2(x)$. As usual, the projection of an empty row (column, resp.) is itself. Similarly, if $L \subseteq \Gamma^{**}$ is a picture language over Γ, we indicate by $\pi(L)$ the projection of language L, *i.e.* $\pi(L) = \{x' | x' = \pi(x) \, \forall x \in L\} \subseteq \Sigma^{**}$.

A quadruple $(\Sigma, \Gamma, \Theta, \pi)$ is called a *tiling system* if Σ and Γ are finite alphabets, Θ is a finite set of tiles over $\Gamma \cup \{\#\}$ and $\pi : \Gamma \to \Sigma$ is a projection. Therefore, a tiling system is composed by a local language over Γ (defined by the set Θ) and a projection $\pi : \Gamma \longrightarrow \Sigma$. A two-dimensional language $L \subseteq \Sigma^{**}$ is *tiling recognizable* if there exists a tiling system $(\Sigma, \Gamma, \Theta, \pi)$ such that $L = \pi(L(\Theta))$. Moreover, we will refer to $L' = L(\Theta)$ as an *underlying local language* for L and to Γ as a *local alphabet* for L. Let $x \in L$, if $x' \in L'$ is such that $\pi(x') = x$, we refer to x' as a *counter-image* of x in the local language L'.

The family of all two-dimensional languages that are *tiling recognizable* is denoted by REC. We give here an example to which we will refer in the next section. More examples can be found in [11].

Example 1. Let L_{col-1n} be the language of pictures p whose first column is equal to the last one. We have that $L_{col-1n} \in REC$. Indeed we can define a tiling system where the information on each letter of the first column of p is brought along horizontal direction, using some subscripts, to the last column of p. More

precisely, we use a local alphabet $\Gamma = \{x_y\}$ with $x, y \in \Sigma$ (the subscripts y are used to recall the symbols in the first column of a picture), the projection $\pi(x_y) = x$ and a set of tiles such that if $p'_{i,j} = x_y$ then $p_{i,j} = x$ and $p_{i,1} = y$. The tiles of the left border must be of the form $\begin{array}{|c|c|}\hline \# & z_z \\ \hline \# & t_t \\ \hline \end{array}$, the tiles of the right border must be of the form $\begin{array}{|c|c|}\hline z_z & \# \\ \hline t_t & \# \\ \hline \end{array}$, whereas the "middle tiles" must be of the form $\begin{array}{|c|c|}\hline z_z & s_z \\ \hline t_t & r_t \\ \hline \end{array}$. Here below it is given an example of a picture $p \in L_{col-1n} \subseteq \{a, b\}^*$ together with a corresponding local picture p'.

$$p = \begin{array}{|c|c|c|c|c|}\hline b & b & a & b & b \\ \hline a & a & b & a & a \\ \hline b & a & a & a & b \\ \hline a & b & b & b & a \\ \hline a & b & b & b & a \\ \hline \end{array} \qquad p' = \begin{array}{|c|c|c|c|c|}\hline b_b & b_b & a_b & b_b & b_b \\ \hline a_a & a_a & b_a & a_a & a_a \\ \hline b_b & a_b & a_b & a_b & b_b \\ \hline a_a & b_a & b_a & b_a & a_a \\ \hline a_a & b_a & b_a & b_a & a_a \\ \hline \end{array}$$

We remark that a tiling system $(\Sigma, \Gamma, \Theta, \pi)$ for a picture language is in some sense a generalization to the two-dimensional case of an automaton that recognizes a string language. Indeed, in one-dimensional case, the quadruple $(\Sigma, \Gamma, \Theta, \pi)$ corresponds exactly to the state-graph of the automaton: the alphabet Γ is in a one-to-one correspondence with the edges, the set Θ describes the edges adjacency, the mapping π gives the labelling of the edges in the automaton. Then, the set of words of the underlying local language defined by set Θ corresponds to all accepting paths in the state-graph and its projection by π gives the language recognized by the automaton (*cf.* [9]). As consequence, when rectangles degenerate in strings the definition of recognizability coincides with the classical one for strings.

We also remark that this approach is very close to that one proposed by W. Thomas in the more general context of graphs (*cf.* [24]). From the results in [12], one derives that both approaches are indeed equivalent for particular graphs, referred there as rectangular grid-graphs.

The family REC is closed with respect to different types of operations (see [11] for all the proofs). Firstly, REC is closed under row and column concatenation operations and under row and column closure operations. As immediate application of this closure properties we consider the following example to which we will refer later.

Example 2. Let L_{col-ij} be the language of pictures p of size (m, n) with the property "$\exists 1 \leq i < j \leq n$ such that the ith column of p is equal to the jth column of p". Observe that $L_{col-ij} = \Sigma^{**} \Phi L_{col-1n} \Phi \Sigma^{**}$. Therefore, since $L_{col-1n} \in REC$ (see Ex. 1) and since REC is closed under column concatenation, we can conclude that $L_{col-ij} \in REC$.

The family REC is also closed under union, intersection and rotation. All those closure properties confirm the close analogy with the one-dimensional case. The big difference regards the complement operation. In [11] and, in a different set-up, in [15], it is shown that the family REC is *not* closed under complement.

Moreover, notice that this definition of recognizability in terms of local languages and projections is implicitly non-deterministic. This can be easily understood if we refer to the one-dimensional case: if no particular constraints are given for the set Θ, the quadruple $(\Sigma, \Gamma, \Theta, \pi)$ corresponds in general to a non-deterministic automaton.

An interesting model of two-dimensional automaton to recognize picture languages is the *two-dimensional on-line tessellation acceptor* (denoted by 2OTA) introduced in [14]. In a sense the 2OTA is an infinite array of identical finite-state automata in a two dimensional space. The computation goes by diagonals starting from top-left towards bottom-right corner of the picture. Depending on the corresponding kinds of automata we can have a deterministic or a non-deterministic version of 2OTA. The family of languages corresponding to a deterministic 2OTA (2DOTA for short) is strictly included in the one corresponding to the non-deterministic model (*cf.* [14]). We will denote $\mathcal{L}$(2OTA) the family of languages accepted by 2OTA; and similarly for the other types of 2OTA.

The definition of REC in terms of tiling systems turns out to be very robust: in [16], it is proved that $REC = \mathcal{L}$(2OTA). Moreover finite tiling systems have also a natural logic meaning: in [11,12] it is shown that the family REC and the family of languages defined by existential monadic second order formulas coincide. And this is actually the generalization of Büchi's theorem for strings to two-dimensional languages. The class REC can also be characterized in terms of regular expressions with projection.

2.2. Unambiguous string languages

We conclude this preliminary section by recalling the definition of unambiguous regular language in the one-dimensional case together with some properties (see [2] for more details) and in particular a necessary condition on unambiguous finite automata for strings we will need in the sequel (*cf.* [13]).

Let $L \subseteq \Sigma^*$ be a regular language. A trim automaton for L is *unambiguous* if it has a *unique* accepting path for each word $w \in L$. Language L is *unambiguous* if it is accepted by an unambiguous automaton. Observe that any deterministic automaton is unambiguous but not *vice versa*. Moreover from classical results on string languages we know that determinism, non-determinism and unambiguity coincide when string languages are concerned. Indeed any regular language is unambiguous, since any (non-deterministic) automaton that accepts it can be determinized and any deterministic automaton is unambiguous.

We also recall the following theorem (see [23]).

Theorem 2.1. *It is decidable whether a given automaton is unambiguous.*

Observe that the problem of deciding whether a regular language is unambiguous is meaningless, because of the just mentioned equivalence results.

Let us now denote by $uns(L)$ the size of a minimal unambiguous non-deterministic finite automaton accepting L. For every regular (string) language $L \subseteq \Sigma^*$, we define the infinite boolean matrix $M_L = \|a_{\alpha\beta}\|_{\alpha\in\Sigma^*,\beta\in\Sigma^*}$ where $a_{\alpha\beta} = 1$ if and only if $\alpha\beta \in L$. Observe that, since every regular language has a finite index (Myhill-Nerode Theorem), the number of different rows of M_L is finite. Moreover, given a matrix M, we denote by $Rank_Q(M)$, the rank of M over the field of rational numbers Q. The following theorem gives a lower bound for the number of states of an unambiguous automaton recognizing a language L.

Theorem 2.2 (Hromkovic *et al.*)**.** *For every regular language $L \subseteq \Sigma^*$, $uns(L) \geq Rank_Q(M_L)$.*

3. Unambiguous recognizable languages

In this section we consider a notion of unambiguity for two-dimensional languages. Informally, a tiling system is unambiguous if every picture has a unique counter-image in its corresponding local language. As in one-dimension, the notion of unambiguity lies between non-determinism and determinism. The definition of *unambiguous recognizable two-dimensional language* was first given in [10].

Definition 3.1. A quadruple $(\Sigma, \Gamma, \Theta, \pi)$ is an *unambiguous tiling system* for a two-dimensional language $L \subseteq \Sigma^{**}$ if and only if for any picture $x \in L$ there exists a *unique* local picture $y \in L(\Theta)$ such that $x = \pi(y)$.

An alternative definition for *unambiguous tiling system* is that function π extended to $\Gamma^{**} \to \Sigma^{**}$ is injective on $L(\Theta)$.

Observe that an unambiguous tiling system can be viewed as a generalization in two dimensions of the definition of unambiguous automaton that recognizes a string language. Moreover if we consider the definition of recognizability on graphs by Thomas [24] an unambiguous tiling system matches exactly the unambiguous finite-state acceptor (that has at most one accepting run for each graph).

Definition 3.2. A two-dimensional tiling recognizable language $L \subseteq \Sigma^{**}$ is *unambiguous* if and only if it admits an unambiguous tiling system $(\Sigma, \Gamma, \Theta, \pi)$.

We denote by UREC the family of all unambiguous recognizable two-dimensional languages. Obviously it holds true that UREC $\subseteq$ REC.

Example 3. The language L_{col-1n} (see Ex. 1) of pictures p whose first column is equal to the last one, is in UREC. Indeed, the tiling system, considered in Example 1, is unambiguous: each picture in L_{col-1n} has a unique counter-image in the local language denoted by that set of tiles. This because there is only one possible counter-image for the first column of a picture p and there is a unique way to build, from this, the counter-image for the second column of p and so on up to the last column of p.

Given the definition of unambiguous tiling system, several questions naturally arise. First of all, in Section 3.1, we give a necessary condition for a language in

UREC. In Section 3.2, we consider the problem of whether all tiling recognizable languages admit unambiguous tiling systems (*i.e.* whether family UREC coincides with REC or it is strictly included in it). We prove that UREC is closed under rotation and intersection operations while it is not closed under row and column concatenations and star operations. Then, in Section 3.3 we investigate on the counterpart of UREC inside family $L(2OTA)$ of two-dimensional languages accepted by on-line tessellation acceptors. Finally, in Section 3.4 we study the problem of deciding whether a given tiling system is unambiguous.

3.1. A necessary condition for unambiguity

In this section we state a necessary condition for the family UREC. For this we use a technique introduced in [18] that reduces two-dimensional languages to string languages over the alphabet of the columns.

More precisely, let $L \subseteq \Sigma^{**}$ be a picture language. For any $m \geq 1$, we consider the subset $L(m) \subseteq L$ containing all pictures in L with exactly m rows. The language $L(m)$ can be viewed as a string language over the alphabet $\Sigma^{m,1}$ of the columns, *i.e.* words in $L(m)$ have a "fixed height m". For example, if $p = \begin{array}{|cccc|}\hline a & b & b & a & a \\ a & a & b & b & a \\ b & b & a & b & a \\ a & a & a & a & b \\ \hline \end{array} \in L$ then the word $w = \begin{bmatrix} a \\ a \\ b \\ a \end{bmatrix}\begin{bmatrix} b \\ a \\ b \\ a \end{bmatrix}\begin{bmatrix} b \\ b \\ a \\ a \end{bmatrix}\begin{bmatrix} a \\ b \\ b \\ a \end{bmatrix}\begin{bmatrix} a \\ a \\ a \\ b \end{bmatrix}$ belongs to the string language $L(4)$ over the alphabet $\Sigma^{4,1} = \left\{ \begin{bmatrix} x \\ y \\ s \\ t \end{bmatrix} \mid x, y, s, t \in \Sigma \right\}$.

Furthermore, recall from Section 2.2 that for any string language L, one can define the infinite boolean matrix $M_L = \|a_{\alpha\beta}\|_{\alpha\in\Sigma^*,\beta\in\Sigma^*}$ where $a_{\alpha\beta} = 1$ if and only if $\alpha\beta \in L$.

We now state the theorem.

Theorem 3.1. *Let $L \subseteq \Sigma^{**}$. If $L \in UREC$, then there is a $k \in N$ such that, for all $m \geq 1$, $Rank_Q(M_{L(m)}) \leq k^m$.*

Proof. Let $L \subseteq \Sigma^{**}$ be recognizable and let $(\Sigma, \Gamma, \Theta, \pi)$ be a tiling system for L. In [18, 19] it is shown that there is a k such that for all $m \geq 1$ there is a finite (string) automaton with k^m states that accepts $L(m)$. This is easily understood from the following observations.

If we have a representation for a recognizable string language S over an alphabet Δ by means of a local string language (a finite set Z of strings of length 2 over $\Gamma \cup \{\#\}$) and a projection $\pi : \Gamma \longrightarrow \Delta$, we can define a corresponding automaton as follows. The states of the automaton are the letters of the local alphabet Γ plus an initial state (that corresponds to the border symbol #); then for each string $xy \in Z$ we add a transition $x \xrightarrow{\pi(y)} y$.

Then, coming back to languages $L(m)$, they have as local alphabets the corresponding sets of "local columns": if γ is the number of symbols in Γ, then the local alphabet for $L(m)$ has γ^m symbols, at most.

Observe that if the tiling system for L is unambiguous, then the automata defined above for the languages $L(m)$ will result unambiguous too. (Recall that such definition of unambiguous tiling systems generalizes the one of unambiguous automata for strings). Therefore we can conclude that if L is in UREC then there exists a k such that, for all $m \geq 1$ the string language $L(m)$ is accepted by an *unambiguous* (string) automaton with k^m states. Then we combine this result with Theorem 2.2 and conclude the proof. □

3.2. Properties of UREC family

We now immediately use the necessary condition for UREC of Theorem 3.1 to prove that, differently from the one-dimensional case (see Sect. 2.2), there are recognizable languages that are not unambiguous. This answers a question already posed in [10] and mentioned also in [22] directly referred to grid-graphs. Then, we prove some closure and non-closure properties of UREC.

Theorem 3.2. *UREC is strictly included in REC.*

Proof. We exhibit a language L in REC and show that L is not in UREC. The language we consider is $L = L_{col-ij}$, the one in Example 2. L contains all pictures p of size (m, n), over the alphabet Σ, with the property "$\exists 1 \leq i < j \leq n$ such that the ith-column of p is equal to the jth-column of p".

For any $m > 1$, we consider languages $L(m)$ as defined above. Observe that such languages $L(m)$ are languages of strings over the alphabets $\Sigma^{m,1}$ (of all possible columns of height m on Σ) with at least two occurrences of the same symbol. Let σ be the cardinality of the alphabet Σ, then the cardinality of $\Sigma^{m,1}$ is σ^m. The problem is now reduced to a problem on strings: more specifically we need to calculate the rank of all the matrices $M_{L(m)}$. We do all the calculations in the next Lemma 3.1 (see below) and obtain that $Rank_Q(M_{L(m)}) = 2^{\sigma^m} + 1$ and then, by Theorem 3.1, L is not in UREC. □

We now prove the result used in the previous theorem.

Lemma 3.1. *Let Δ be a finite alphabet and let $S \subseteq \Delta^*$ be the set of strings with at least two occurrences of the same symbol. Then $Rank_Q(M_S) = 2^{Card(\Delta)} + 1$.*

Proof. Let $Card(\Delta) = c$ and let us suppose $\Delta = \{a_1, a_2, \ldots, a_c\}$. The maximum number of different rows in M_S is 2^c+1. They are, for example, the rows indexed by a_1a_1, λ and by all possible $\alpha = a_{i_1}a_{i_2}\ldots a_{i_h}$ with $1 \leq h \leq c$ and $i_1 < i_2 < \ldots < i_h$ (note that if α contains two occurrences of the same symbol then its row contains only 1 as the row indexed by a_1a_1). Similarly, the maximum number of different columns in M_S is $2^c + 1$ and, for example, the columns indexed as before are all different. Let us consider, in M_S, the sub-matrix M_c composed by all these

different rows and all these different columns rearranged, for induction on c, in this way: if $c = 1$, then the first row and the first column of M_1 are indexed by a_1a_1, the second ones by λ and the third ones by a_1. The matrix M_c is obtained from the matrix M_{c-1} by adding to it 2^{c-1} rows and 2^{c-1} columns, so indexed: if $\alpha_1, \alpha_2, \ldots, \alpha_{2^{c-1}}$ are the indexes of the rows (and the columns) of M_{c-1} (except the first one), then the rows (and the columns), added in order to obtain M_c, are those ones indexed by $\alpha_1 a_c, \alpha_2 a_c, \ldots, \alpha_{2^{c-1}} a_c$.

For example, if $c = 3$, the matrix M_c will be the following one:

$M_c =$

	a_1a_1	λ	a_1	a_2	a_1a_2	a_3	a_1a_3	a_2a_3	$a_1a_2a_3$
a_1a_1	1	1	1	1	1	1	1	1	1
λ	1	0	0	0	0	0	0	0	0
a_1	1	0	1	0	1	0	1	0	1
a_2	1	0	0	1	1	0	0	1	1
a_1a_2	1	0	1	1	1	0	1	1	1
a_3	1	0	0	0	0	1	1	1	1
a_1a_3	1	0	1	0	1	1	1	1	1
a_2a_3	1	0	0	1	1	1	1	1	1
$a_1a_2a_3$	1	0	1	1	1	1	1	1	1

Now we show that $\forall c \geq 1$, $det(M_c) \neq 0$ and, therefore, since M_c has order $2^c + 1$, we can conclude that $Rank_Q(M_S) = 2^c + 1$. Indeed, if we evaluate $det(M_c)$ along its second column and then along its first row, then we have that $det(M_c) = (-1)det(M'_c)$, where M'_c is the matrix obtained from M_c by removing the first two rows and the first two columns. We will show that $det(M'_c) \neq 0$ and this will conclude this proof.

Let us set $k = 2^c - 1$: note that $M'_c = \|m'_{ij}\|$ is a square matrix of order k such that $m'_{ij} = 1$, $\forall 1 \leq i, j \leq k$ with $i + j \geq k + 1$, and $m'_{ij} = 0$, $\forall 1 \leq i, j \leq k$ with $i + j = k$. In other words, matrix M'_c has 1 in all positions of the counter-diagonal and in all positions below it. Moreover, all positions immediately above the counter-diagonal contain 0.

We show that for every square boolean matrix $M = \|a_{ij}\|$ of order k such that $a_{ij} = 1$, $\forall 1 \leq i, j \leq k$ with $i + j \geq k + 1$, and $a_{ij} = 0$, $\forall 1 \leq i, j \leq k$ with $i + j = k$, it holds $detM \neq 0$. The proof is by induction on k. For the basis, $k = 2$, it is obvious. Suppose that it is true for square matrix of order $k - 1$ and consider a square matrix M of order k. Note that the last column of M contains only 1 and the second-last column contains 0 in its first position and 1 in the other positions. Therefore, if we evaluate $det(M)$ along its first row, we have that $\forall j = 1, \ldots, k-2$, $det(M_{1j}) = 0$ (because every M_{1j} has two identical columns, the last one and the second-last one). Moreover $a_{1k-1} = 0$ and therefore $a_{1k-1}det(M_{1k-1}) = 0$. So we have that $detM = (-1)^{1+k}a_{1k}detM_{1k} = (-1)^{1+k}detM_{1k}$. But M_{1k} is a matrix of order $k - 1$ that satisfies the hypothesis and, therefore, by inductive hypothesis, $detM_{1k} \neq 0$. □

Given a recognizable two-dimensional language L, we say that L is *inherently ambiguous* if there are no unambiguous tiling systems for L. Then, Theorem 3.2 can be restated as:

Theorem 3.3. *There exist recognizable two-dimensional languages that are inherently ambiguous.*

We now prove some closure and some non-closure properties of UREC under classical operations on REC family. In [21] it is proved that UREC is closed under projection and disjoint union. We now state that it is also closed under rotation and intersection. So UREC family shares several important properties of closure with REC family. Recall that REC is closed under the boolean operations, rotation and projection, but not under complementation. The closure of UREC under complementation is still an open question. This problem is also stated in [22] in the more general context of the graphs.

Proposition 3.1. *UREC is closed under intersection and rotation operations.*

Proof. If $L_1, L_2 \in REC$ then one can construct a tiling system for $L_1 \cap L_2$ following the proof that REC is closed under intersection (see [11]). It is easy to show that if the tiling systems for L_1 and L_2 were unambiguous then the resulting tiling system is so. The closure under rotation comes directly from the definition of UREC. □

As important consequence of the argument in the proof of Theorem 3.2, we obtain that UREC does not inherit all closure properties of REC.

Proposition 3.2. *UREC is not closed under row and column concatenation and under row and column closure.*

Proof. Let us consider language L_{col-1n} of pictures with leftmost column equal to the rightmost one. In Example 3 it is shown that it belongs to UREC. Now consider language L_{col-ij} of pictures such that there exist $1 \leq i < j \leq n$ and the i-th column is equal to the j-th one. In Example 2 we observed that $L_{col-ij} = \Sigma^{**} \circledcirc L_{col-1n} \circledcirc \Sigma^{**}$ but it is not in UREC as shown in the proof of Theorem 3.2.

Using similar techniques (for example, languages obtained applying some rotations to the above ones) one can prove also that family UREC is not closed under column concatenation and under row and column closure. □

3.3. Unambiguous Tessellation Acceptors

In this section we consider the relation between UREC and the automata models. Recall that family REC is exactly the family $\mathcal{L}(2OTA)$ of languages accepted by two-dimensional on-line tessellation acceptors (*cf.* Sect. 2). Then it is natural to ask what is the counterpart of UREC inside $\mathcal{L}(2OTA)$. We can give the following definition (see [21]).

Definition 3.3. A 2-dimensional on-line tessellation acceptor is an *unambiguous on-line tessellation acceptor* (2UOTA, for short), if any picture has at most one accepting computation.

We can immediately state the following proposition whose proof can be given by following step-by-step the proof that $REC = \mathcal{L}(2OTA)$ (see [16]) (the result was independently observed in [21]).

Proposition 3.3. $\mathcal{L}(2UOTA) = UREC$.

It is easy to see that any deterministic 2OTA (2DOTA for short) is in particular a 2UOTA. We recall that family $\mathcal{L}(2DOTA)$ is strictly included in $\mathcal{L}(2OTA)$ (see Sect. 2). The following theorem proves that family $\mathcal{L}(2UOTA)$ is still different both with respect to $\mathcal{L}(2OTA)$ and $\mathcal{L}(2DOTA)$.

Theorem 3.4. *The inclusions* $\mathcal{L}(2DOTA) \subset \mathcal{L}(2UOTA) \subset \mathcal{L}(2OTA)$ *are all strict.*

Proof. The second strict inclusion (*i.e.*, $\mathcal{L}(2UOTA) \subset \mathcal{L}(2OTA)$) can be obtained directly from Theorem 3.2 by using the result that $REC = \mathcal{L}(2OTA)$ (see [16]) together with Proposition 3.3.

Let us focus on the first strict inclusion, *i.e.* that $\mathcal{L}(2DOTA) \subset \mathcal{L}(2UOTA)$. Let L be the language of squares over a two-letters alphabet $\Sigma = \{a, b\}$ with last row equal to the last column. We have that $L \in UREC$ because we can define a tiling system where the information on each letter of the last row is brought up till the diagonal and then right towards the last column. More precisely, we use a local alphabet of 12 symbols (apart #) $\Gamma = \{0_x^y, 1_x^y, 2_x^y\}$ with $x, y \in \{a, b\}$ and let symbols 0_x^y occur only below the diagonal, symbols 1_x^y occur only in the diagonal and symbols 2_x^y only above the diagonal. Moreover the superscript symbols correspond to the "real value" of the symbol (*i.e.* $\pi(0_x^y) = \pi(1_x^y) = \pi(2_x^y) = y$) while the subscript symbols correspond to information we are bringing from the last row to the last column (making a turn at the diagonal). For example, if $p'_{ij} = 0_b^a$ this means that position (i, j) is below the diagonal and that in the corresponding picture p, $p(i, j) = a$, while going down to the last row, $p(m, j) = b$. Moreover, if $p'_{ij} = 2_b^a$ this means that position (i, j) is above the diagonal and that in the corresponding picture p, $p(i, j) = a$, while going right to the last column, $p(i, n) = b$. It is not too difficult to guess what are all the allowed tiles for the local language L': tiles with 0 (or 0 and 1) should have same subscript symbols along vertical direction while tiles with 2 (or 2 and 1) should have same subscript symbols along horizontal direction while tiles for the diagonal (with 0, 1 and 2) should have same subscript symbols in 0 and 1 along vertical direction and in 1 and 2 along horizontal one. Moreover tiles in the bottom border contain only 0_a^a or 0_b^b while tiles in the right border contain only 2_a^a or 2_b^b with the exception of the bottom-right corner that can hold only 1_a^a or 1_b^b.

Here below it is given an example of a picture $p \in L$ together with the corresponding local picture p'.

$$p = \begin{array}{|c|c|c|c|c|}\hline b & b & a & a & a \\ \hline a & a & a & b & b \\ \hline b & a & a & b & b \\ \hline a & b & b & b & a \\ \hline a & b & b & a & a \\ \hline \end{array} \qquad p' = \begin{array}{|c|c|c|c|c|}\hline 1_a^b & 2_a^b & 2_a^a & 2_a^a & 2_a^a \\ \hline 0_a^a & 1_b^a & 2_b^a & 2_b^b & 2_b^b \\ \hline 0_a^b & 0_b^a & 1_b^a & 2_b^b & 2_b^b \\ \hline 0_a^a & 0_b^b & 0_b^b & 1_a^b & 2_a^a \\ \hline 0_a^a & 0_b^b & 0_b^b & 0_a^a & 1_a^a \\ \hline \end{array}$$

Observe that each picture in L has a unique counter-image in the local language denoted by this set of tiles. This because there is only one possible counter-image for the last row and last column and then, fixed those ones, one can "built" the whole counter-image in a unique way. Then the theorem holds because $L \notin \mathcal{L}$(2DOTA) (see [14]). □

3.4. An undecidable problem

We now consider the problem of deciding whether a given tiling system $(\Sigma, \Gamma, \Theta, \pi)$ is unambiguous. We prove that this problem is undecidable. Recall that on the contrary, when string languages are dealt with, it is decidable whether a given automaton is unambiguous (see Th. 2.1). The proof of the undecidability will be obtained by reduction from the *2-dimensional Unique Decipherability Problem* that we briefly recall below (see [7] for all details).

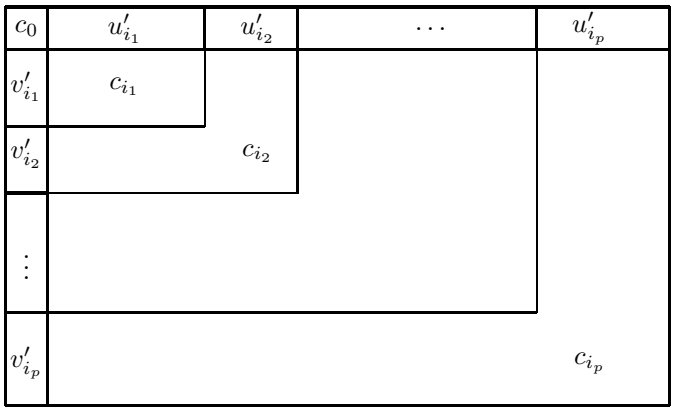
Let Σ be a finite alphabet and $S = \{(u_1, v_1), \ldots, (u_k, v_k)\}$ be a system where $u_i, v_i \in \Sigma^*$ for all $i = 1, \ldots, k$. The system S is *uniquely decipherable* if and only if, for all $i_1, \ldots, i_p, j_1, \ldots, j_p \in \{1, \ldots, k\}$, the equalities $u_{i_1} \ldots u_{i_p} = u_{j_1} \ldots u_{j_q}$ and $v_{i_1} \ldots v_{i_p} = v_{j_1} \ldots v_{j_q}$ imply together that $p = q$ and $(i_1, \ldots, i_p) = (j_1, \ldots, j_p)$. The *2-dimensional Unique Decipherability Problem* is the one of deciding whether a given system is uniquely decipherable. In [7] it is shown that this problem is undecidable using a reduction from the *Post Correspondence Problem*.

Theorem 3.5. *Given a tiling system $(\Sigma, \Gamma, \Theta, \pi)$, it is undecidable whether it is unambiguous.*

Proof. We define a reduction from the 2-dimensional Unique Decipherability Problem. Let $S = \{(u_1, v_1), \ldots, (u_k, v_k)\}$ be a system where $u_i, v_i \in \Sigma^*$ for all $i = 1, \ldots, k$ and let $c \notin \Sigma$. We define a picture language $L(S)$ over the alphabet $\Sigma \cup \{c\}$ as a set of rectangles defined as follows. For any combination $(i_1, \ldots, i_p)$ of $\{1, \ldots, k\}$ there is a rectangle in $L(S)$ such that the first row is the string $c\, u_{i_1} \ldots u_{i_p}$ and the first column is the string $c\, v_{i_1} \ldots v_{i_p}$ and all the other letters are c's.

It is not difficult to prove that $L(S) \in REC$. In fact, we can construct a tiling system $(\Sigma, \Gamma, \Theta, \pi)$ for $L(S)$ in a similar way as in the proof of Theorem 3.4 for the language of squares with last row equal to the last column. The local alphabet is

$\Gamma = \Sigma' \cup \{c_0, c_1, \ldots, c_k\}$ where Σ' is a local alphabet for the strings u'_i, v'_j for all $i, j = 1, \ldots, k$. The pictures of the set $L(\Theta)$ are like the following. The information that in the first row there is a factor u_{i_j} is brought up by a stripe of c_{i_j} of width $|u_{i_j}|$ downwards and then left towards the first column in a stripe of length $|v_{i_j}|$. The symbol c_0 is needed for the position $(1,1)$. The projection π maps each string u'_i, v'_j to u_i, v_j respectively and symbol c_i to c.



Then the tiling system $(\Sigma, \Gamma, \Theta, \pi)$ is unambiguous if and only if the system S is uniquely decipherable. Indeed the equalities $u_{i_1} \ldots u_{i_p} = u_{j_1} \ldots u_{j_q}$ and $v_{i_1} \ldots v_{i_p} = v_{j_1} \ldots v_{j_q}$ hold with $(i_1, \ldots, i_p) \neq (j_1, \ldots, j_p)$ if and only if the picture in $L(S)$ whose first row is the string $c\, u_{i_1} \ldots u_{i_p}$ and first column is the string $c\, v_{i_1} \ldots v_{i_p}$ has two counter-images, one with stripes carrying in order $c_{i_1}, \ldots c_{i_p}$ and the other one with stripes carrying in order $c_{j_1}, \ldots, c_{j_q}$. □

Recall that in the framework of one-dimensional string languages, it holds the opposite result with respect to Theorem 3.5: in fact it is decidable whether a given automaton is unambiguous (see Sect. 2.2). Furthermore the problem of deciding whether a regular language is unambiguous is meaningless. In the two dimensional case, the problem of deciding whether a given picture language (given by its tiling system) is unambiguous is an open problem.

4. Concluding remarks and future works

In this paper we have studied the family UREC of unambiguous recognizable two-dimensional languages. The notion of unambiguity seems to be more suitable in this framework, since while expressing the uniqueness of a computation, it is independent from a privileged direction of computation. In fact we have shown that definition of UREC is stable with respect to rotations. Remark that in the case of deterministic 2OTA, this is no longer true. Moreover UREC family is closed with respect to intersection, as required in a deterministic model. The main

open problem with this definition is whether UREC family is also closed under complementation. Moreover one can ask other related questions. For example, we can show that the complement of language L_{col-ij} of Example 2 is not recognizable (recall that in Th. 3.2 we proved that L_{col-ij} is inherently ambiguous). Then one could ask whether this holds for any language in REC but not in UREC, or whether languages in REC whose complement is in REC are all unambiguous.

Furthermore, in Section 3.4 we have proved that it is undecidable whether a given tiling system is unambiguous. Another open problem is to decide whether a given picture language (given by its tiling system) is unambiguous.

References

[1] M. Anselmo, D. Giammarresi and M. Madonia, New Operations and Regular Expressions for two-dimensional languages over one-letter alphabet. *Theoret. Comput. Sci.* **340** (2005) 408–431.

[2] J. Berstel and D. Perrin, *Theory of Codes*. Academic Press (1985).

[3] A. Bertoni, G. Mauri and N. Sabadini, Unambiguous regular trace languages, in *Algebra, Combinatorics and Logic in Computer Science*, edited by J. Demetrovics, G. Katona and A. Salomaa, North Holland. *Math. Soc. Janos Bolyay* **42** (1985).

[4] S. Bozapalidis and A. Grammatikopoulou, Recognizable picture series, in Special Issue on Weighted Automata. *Journal of Automata, Languages and Combinatorics*, Vol. 10, No. 2 (2005).

[5] M. Blum and C. Hewitt, Automata on a two-dimensional tape, in *IEEE Symposium on Switching and Automata Theory* (1967) 155–160.

[6] C. Choffrut and B. Durak, Collage of two-dimensional words. *Theoret. Comput. Sci.* **340** (2005) 364–380.

[7] M. Chrobak and W. Rytter, Unique decipherability for partially commutative alphabets. *Fundamenta Informatica* **X** (1987) 323–336.

[8] S. Crespi Reghizzi and M. Pradella, Tile rewriting grammars and picture languages. *Theoret. Comput. Sci.* **340** (2005) 257–272.

[9] S. Eilenberg, *Automata, Languages and Machines*. Vol. A, Academic Press (1974).

[10] D. Giammarresi and A. Restivo, Recognizable picture languages. *International Journal Pattern Recognition and Artificial Intelligence* **6** (1992) 241–256.

[11] D. Giammarresi and A. Restivo, Two-dimensional languages, in *Handbook of Formal Languages*, edited by G. Rozenberg and A. Salomaa. Springer-Verlag, Berlin **III** (1997) 215–268.

[12] D. Giammarresi, A. Restivo, S. Seibert and W. Thomas, Monadic second order logic over pictures and recognizability by tiling systems. *Inform. Computat.* **125** (1996) 32–45.

[13] J. Hromkovic, J. Karumäki, H. Klauck, G. Schnitger and S. Seibert, Communication Complexity Method for Measuring Nondeterminism in Finite Automata. *Inform. Comput.* **172** (2002) 202–217.

[14] K. Inoue and A. Nakamura, Some properties of two-dimensional on-line tessellation acceptors. *Information Sciences* **13** (1977) 95–121.

[15] K. Inoue and A. Nakamura, Nonclosure properties of two-dimensional on-line tessellation acceptors and one-way parallel/sequential array acceptors. *Trans. IECE Japan* **6** (1977) 475–476.

[16] K. Inoue and I. Takanami, A Characterization of recognizable picture languages, in *Proc. Second International Colloquium on Parallel Image Processing*, edited by A. Nakamura *et al. Lect. Notes Comput. Sci.* **654** (1993).

[17] J. Kari and C. Moore, Rectangles and squares recognized by two-dimensional automata, in *Theory is Forever*, edited by Karhumaki *et al. Lect. Notes Comput. Sci.* **3113** (2004) 134–144.

[18] O. Matz, On piecewise testable, starfree, and recognizable picture languages, in *Foundations of Software Science and Computation Structures*, edited by M. Nivat, Springer-Verlag, Berlin **1378** (1998).
[19] O. Matz and W. Thomas, The Monadic Quantifier Alternation Hierarchy over Graphs is Infinite, in *IEEE Symposium on Logic in Computer Science, LICS. IEEE* (1997) 236–244.
[20] I. Mäurer, Recognizable and Rational Picture Series, in *Procs. Conf. on Algebraic Informatics*, Thessaloniki (2005), Aristotte University of Thessaloniki.
[21] I. Mäurer, Weighted Picture Automata and Weighted Logics, in *Procs. STACS 2006*, Springer Belin. *Lect. Notes Comput. Sci.* **3884** (2006) 313–324.
[22] A. Potthoff, S. Seibert and W. Thomas, Nondeterminism versus determinism of finite automata over directed acyclic graphs. *Bulletin Belgian Math. Soc.* **1** (1994) 285–298.
[23] J. Sakarovitch, *Eléments de théorie des automates*. Vuibert, Paris (2003).
[24] W. Thomas, On Logics, Tilings, and Automata, in *Proc. 18th ICALP*, Springer-Verlag, Berlin. *Lect. Notes Comput. Sci.* **510** (1991) 441–453.