Available online at www.sciencedirect.com
SCIENCE DIRECT®

Theoretical Computer Science 340 (2005) 408–431

Theoretical Computer Science

www.elsevier.com/locate/tcs

New operations and regular expressions for two-dimensional languages over one-letter alphabet☆

Marcella Anselmo[a,*], Dora Giammarresi[b], Maria Madonia[c]

[a] *Dipartimento Informatica ed Applicazioni, Università di Salerno, 84081 Baronissi (Sa), Italy*
[b] *Dipartimento Matematica, Università di Roma "Tor Vergata", via Ricerca Scientifica, 00133 Roma, Italy*
[c] *Dipartimento Matematica e Informatica, Università di Catania, Viale Andrea Doria 6/a, 95125 Catania, Italy*

Abstract

We consider the problem of defining regular expressions to characterize the class of recognizable picture languages in the case of a one-letter alphabet. We define a *diagonal concatenation* and its *star* and consider two different families, $L(D)$ and $L(CRD)$, of languages denoted by regular expressions involving such operations plus classical operations. $L(D)$ is characterized both in terms of rational relations and in terms of two-dimensional automata moving only right and down. $L(CRD)$ is included in REC and contains languages defined by three-way automata while languages in $L(CRD)$ necessarily satisfy some regularity conditions. Finally, we introduce new definitions of *advanced stars* expressing the necessity of conceptually different definitions for iteration.

Keywords: Two-dimensional language; Regular expression

1. Introduction

The theory of one-dimensional string languages is well founded and investigated since 1950. From several years, the increasing interest for pattern recognition and image

---

☆ Work partially supported by Miur Cofin Linguaggi Formali e Automi: Metodi, Modelli e Applicazioni.

* Corresponding author.
E-mail addresses: anselmo@dia.unisa.it (M. Anselmo), giammarr@mat.uniroma2.it (D. Giammarresi), madonia@dmi.unict.it (M. Madonia).

0304-3975/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2005.03.031

processing has also motivated the research on *two-dimensional string languages*, and nowadays this is a field of big investigation. The scope is the generalization, or possibly the extension, of the richness of the theory of one-dimensional languages to two dimensions. A first attempt has been devoted to the study of two-dimensional languages defined by finite state devices, with the aim of finding a counterpart of what regular languages are in one dimension. Many approaches have been presented in the literature considering all ways to define regular languages: finite automata, grammars, logic and regular expressions. In 1991, an unifying point of view was presented by A. Restivo and D. Giammarresi who defined the family REC of *recognizable picture languages* (see [7]). This class seems to be the candidate as “the” generalization of the class of regular one-dimensional languages. Indeed REC is well characterized from very different points of view and thus inherits several properties from the class of regular string (one-dimensional) languages. It is characterized in terms of projections of local languages (*tiling systems*), of some finite-state automata, of logic formulas and of regular expressions with alphabetic mapping. The approach by regular expressions is indeed not completely satisfactory: the concatenation and star operations involved there are partial functions and moreover an external operation of alphabetic mapping is needed. Then, in [7], the problem of stating a Kleene-like theorem for the theory of recognizable picture languages remains open.

Several papers were recently devoted to find a better formulation for regular expressions for two-dimensional languages. In [15], O. Matz affords the problem of finding some more powerful expressions to represent recognizable picture languages and suggests some regular expressions where the iteration is over combinations of operators, rather than over languages. The author shows that the power of these expressions does not exceed the family REC, but it remains open whether or not it exhausts it. In [18] some tiling operation is introduced as extension of the Kleene star to pictures and a characterization of REC that involves some morphism and the intersection is given. The paper [19] compares star-free picture expressions with first-order logic.

The aim of this paper is to look for a homogenous notion of regular expression that could extend more naturally the concept of regular expression of one-dimensional languages. In this framework, we propose some new operations on pictures and picture languages and study the families of languages that can be generated using classical and new operations.

The paper focuses on one-letter alphabets. This is a particular case of the more general case of several letter alphabets. However this is not only a simpler case to handle, but it is a necessary and meaningful case to start. Indeed studying two-dimensional languages on one-letter alphabets means to study the “shapes” of pictures: if a picture language is in REC then necessarily the language of its shapes is in REC. Such approach allows us to separate the twofold nature of a picture: its shape and its content.

Classical concatenation operations on pictures and picture languages are the row and column concatenations and their closure. Regular expressions that use only Boolean operations and this kind of concatenations and closure however cannot define a large number of two-dimensional languages in REC. As an example, take the simple language of “squares” (that is pictures with number of rows equal to the number of columns). The major problem with this kind of regular expressions is that they cannot describe any relationship existing between the two dimensions of the pictures. Such operations are useful to express some regularity either on the number of rows or on the number of columns but not between them. This is the reason we introduce, in the one-letter case, a new concatenation operation between pictures: the *diagonal concatenation*. The diagonal concatenation introduces the possibility of constructing new pictures forcing some dependence between their dimensions. Moreover an important aspect of the diagonal concatenation is that it is a total function between pictures. This allows to find a quite clean double characterization of *D-regular languages*, the picture languages denoted by regular expressions containing union, diagonal concatenation and its closure: they are exactly those picture languages in which the dimensions are related by a rational relation and also exactly those picture languages recognizable by particular two-dimensional automata moving only right and down.

Unfortunately, an analogous situation does not hold anymore, when we also introduce row and column concatenations in regular expressions, essentially because they are partial functions. The class of *CRD-regular languages*, the languages denoted by regular expressions with union, column, row and diagonal concatenations and their closures, strictly lies between the class of languages recognized by three-way deterministic automata and REC. The main result for *CRD*-regular languages is a necessary condition regarding a sort of "regularity" in the possible "extensions" of a picture in a given language to another bigger picture in the language. In a *CRD*-regular language: if a picture is sufficiently "long" then we can concatenate to it some picture infinitely often by columns; if a picture is sufficiently "high" then we can concatenate to it some picture infinitely often by rows; if a picture is sufficiently "big" then we can concatenate to it some picture infinitely often in diagonal. This result generalizes in some sense what regularity implies in one-dimensional languages over a one-letter alphabet.

We also provide a collection of examples classically considered in the literature, specifying for each of them whether they belong or not to the classes of two-dimensional languages considered throughout the paper.

Examining some examples of languages not captured by *CRD* formalism, we find out that the "extensions'' of a picture cannot be obtained by iterating the concatenation of the same picture, and this independently from the picture to what we concatenate. On the contrary, for some languages, a kind of iteration that generate pictures in a "non-uniform" way is needed, indeed depending from the picture just obtained. This is a new situation with respect to the one-dimensional case. Such considerations show the necessity of a more complex definition for regular expressions in order to denote a wider class of two-dimensional languages in REC. We introduce the definitions of some *advanced stars*. They allow to capture a wider class of languages, that still remains inside the class REC. All definitions are given in such a way to synchronize the steps of iteration on a picture with the picture just constructed. Observe that in this case we exploit the partial nature of column and row concatenation operations. We conclude by discussing some ideas for extending all those definitions to the general alphabet case.

The paper is organized as follows. In Section 2 we recall some preliminary definitions and results later used in the paper. Section 3 contains the main results: it presents our proposals for possible classes of regular expressions. Moreover, it contains a table summarizing a wide collection of examples. In Section 4 we define new star operations that allow to describe many more languages (over one-letter alphabet) in REC, while in Section 5 we draw some conclusions and proposals to extend results of this paper to two-dimensional languages over general alphabets.

A preliminary version of this paper appeared in [1].

2. Preliminaries

In this section we recall terminology for two-dimensional languages. Then, we briefly describe some machine-based model for recognizing two-dimensional languages and summarize all major results concerning the class *REC* of *recognizable two-dimensional languages*, that is the one that seems to generalize better the family of regular string languages to two dimensions.

We assume the reader is familiar with the basic terminology and properties of the theory of one-dimensional languages as can be found for example in [8]. We will first introduce some definitions about two-dimensional languages by borrowing and extending notation from the theory of one-dimensional languages. Next, we will give formal definitions of the classical concatenation operations between two-dimensional strings (pictures) and two-dimensional languages. The notations used can be mainly found in [7].

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^{**}. A *two-dimensional language* over Σ is a subset of Σ^{**}.

Given a picture $p \in \Sigma^{**}$, let $\ell_1(p)$ denote the number of rows of p and $\ell_2(p)$ denote the number of columns of p. The pair $(\ell_1(p), \ell_2(p))$ is called the *size* of the picture p. 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 $m, n \geqslant 0$, that we call *empty columns* and *empty rows*, and denote by $\lambda_{0,m}$ and $\lambda_{n,0}$ respectively. The *empty picture* is the only picture of size $(0, 0)$ and it will be denoted by $\lambda_{0,0}$. We indicate by Λ_{col} and Λ_{row} the language of all empty columns and of all empty rows, respectively.

We give first some simple examples of two-dimensional languages.

Example 1. Let $\Sigma = \{a\}$ be a one-letter alphabet. The set of pictures of a's with three columns is a two-dimensional language over Σ. It can be formally described as $L = \{p \mid \ell_2(p) = 3\} \subseteq \Sigma^{**}$. As another example let L be the subset of Σ^{**} that contains all the pictures with a shape of "squares". More formally, $L = \{p \mid \ell_1(p) = \ell_2(p)\} \subseteq \Sigma^{**}$.

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 (n, m) and (m', n'), $m, n, m', n' \geqslant 0$, respectively.

Definition 2. The *column concatenation* of p and q (denoted by $p \mathbin{⦶} q$) and the *row concatenation* of p and q (denoted by $p \ominus q$) are partial operations, defined only if $n = n'$ and if $m = m'$, respectively and are given by

$$p \mathbin{⦶} q = \boxed{\begin{array}{c|c} p & q \end{array}}, \qquad p \ominus q = \boxed{\begin{array}{c} p \\ \hline q \end{array}}.$$

Moreover we set $p \oplus \lambda_{n,0} = p$ and $p \ominus \lambda_{0,m} = p$ that is, the empty columns and the empty rows are the neutral elements for the column and the row concatenation operations, respectively.

As in the string language theory, these definitions of pictures concatenation can be extended to define concatenations between set of pictures. Let $L_1, L_2 \subseteq \Sigma^{**}$, the *column concatenation* and the *row concatenation* of L_1 and L_2 are defined respectively by

$$L_1 \oplus L_2 = \{p \oplus q \mid p \in L_1, q \in L_2\} \quad \text{and} \quad L_1 \ominus L_2 = \{p \ominus q \mid p \in L_1, q \in L_2\}.$$

By iterating the concatenation operations, we can define the columns and rows *transitive closures*, which are somehow *"two-dimensional Kleene star"*. Let L be a picture language.

Definition 3. The *column closure* (column star) and the *row closure* (row star) of L are defined as

$$L^{*\oplus} = \bigcup_{i \geqslant 0} L^{i\,\oplus}, \qquad L^{*\ominus} = \bigcup_{i \geqslant 0} L^{i\,\ominus},$$

where $L^{0\oplus} = \Lambda_{\mathrm{col}},\ L^{1\oplus} = L,\ L^{n\oplus} = L \oplus L^{(n-1)\oplus}$ and $L^{0\ominus} = \Lambda_{\mathrm{row}},\ L^{1\ominus} = L,\ L^{n\ominus} = L \ominus L^{(n-1)\ominus}$.

2.1. Automata for two-dimensional languages

In this section we briefly review different kinds of automata that read two-dimensional tapes. All models reduce to conventional automata when restricted to operate on one-row pictures.

One of the first attempts at formalizing the concept of "recognizable picture language" was made by M. Blum and C. Hewitt who in 1967 introduced a model of finite automaton that reads a two-dimensional tape (cf. [3]). A deterministic (non-deterministic) four-way automaton, denoted by 4DFA (4NFA), is defined as an extension of the two-way automaton that recognizes strings (cf. [8]) by allowing it to move in four directions: *Left, Right, Up, Down*. For example, a 4DFA can recognize squares by starting its computation from top-left corner of a given picture and going alternatively one step right one step down (i.e. following the diagonal) till it reaches the bottom-right corner.

The families of picture languages recognized by some 4DFA and 4NFA are denoted by L(4DFA) and L(4NFA), respectively. An important result (cf. [3]) states that, unlike in the one-dimensional case, the family L(4DFA) is strictly included in the family L(4NFA). Both families L(4DFA) and L(4NFA) are closed under Boolean union and intersection operations. The family L(4DFA) is also closed under complement, while for L(4NFA) this is not known.

From several points of view, four-way automata could appear as a reasonable model of computation for two-dimensional tapes and they were widely studied, but they have a major bug. In fact, it can be proved that both L(4DFA) and L(4NFA) are not closed under row and column concatenation and closure operations [12].

In [14], a weaker model called three-way automaton is also considered in the two versions non-deterministic and deterministic (referred to as 3NFA and 3DFA, respectively) that is allowed to move right, left and down only. The family L(3NFA) is strictly included in L(4NFA).

Another interesting model of two-dimensional automaton is the *two-dimensional on-line tessellation acceptor* (denoted by 2-OTA) introduced in [9]. In a sense the 2-OTA 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 2-OTA. Despite the fact that this model is quite different in principle from four-way automaton, also in this case the family of languages corresponding to a determinist 2-OTA is strictly included in the one corresponding to the non-deterministic model.

In [9] it is proved that the family of two-dimensional languages recognized by a 2-OTA, L(2-OTA) is closed under union and intersection and also under row/column concatenation and row/column star while it is not closed under complement. Moreover L(2-OTA) properly includes family L(4NFA). The only trouble with this 2-OTA model is that it is quite difficult to manage.

2.2. *Tiling systems and the class REC*

A different way to define (recognize) picture languages was introduced by A. Restivo and D. Giammarresi in [6]. It generalizes the characterization of regular languages by means of local strings language and alphabetic mapping to two dimensions (the local set together with the mapping is an alternative description of the state graph of an automaton).

We recall that a local language of strings is defined by means of a finite set of strings of length 2. As natural generalization, a local picture language L over an alphabet Γ is defined by means of a finite set Θ of pictures of size $(2, 2)$ (called *tiles*) that represent all allowed sub-pictures for the pictures in L. To be more precise, such set Θ is defined over $\Gamma \cup \{\#\}$ where # is a border symbol that we assume always to surround all the pictures. A *tiling system* for a language L over Σ is a pair of a local language over an alphabet Γ and an alphabetic mapping $\pi : \Gamma \to \Sigma$. The mapping π can be extended in the obvious way from the alphabet Γ to pictures over Γ and to picture languages over Γ. Then, we say that a language $L \subseteq \Sigma^{**}$ is *recognizable by tiling systems* if there exists a local language L' over Γ and a mapping $\pi : \Gamma \to \Sigma$ such that $L = \pi(L')$. The family of two-dimensional languages recognizable by tiling systems is denoted by REC.

As an example, consider again the language L of squares over a one-letter alphabet $\Sigma = \{a\}$. Then L is in REC since it can be obtained as $\pi(L')$, where L' is the language of squares over $\Gamma = \{0, 1\}$ that have 1 in the diagonal positions and 0 elsewhere and $\pi(0) = \pi(1) = a$.

The family REC is closed under Boolean union and intersection but not under complement. It is also closed under all row and column concatenations and stars. Moreover, by definition, it is closed under alphabetic mappings. This notion of recognizability by tiling systems turns out to be very *robust*: in [11], it is proved that REC $= L$(2-OTA). Moreover finite tiling systems have also a natural logic meaning: in [7] 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, as specified in Section 2.3.

2.3. Regular expressions and the class REC

The characterizations of the family REC show that the family REC captures in some sense the idea of unification of the concept of recognizability from the two different points of view of descriptive and computational models, that is one of the main properties of the class of recognizable string languages. It seems thus natural to ask whether one can prove also a sort of two-dimensional Kleene's Theorem. Using row and column concatenations and closure operations, it is possible to express two-dimensional languages by means of simpler languages. Nevertheless it can be observed that such classical operations are useful to express some regularity either on the number of rows or on the number of columns, but they cannot describe any relationship existing between the two dimensions of the pictures. As an example, already in the case of a one-letter alphabet, we have that languages such as the language of "squares'' (see Example 1) cannot be described using only classical operations. More precisely, O. Matz [15] has characterized the class of languages that can be obtained starting from finite languages and applying Boolean operations, column and row concatenations and stars, as the class of languages that are a finite union of Cartesian products of ultimately periodic string languages.

Furthermore, it can be shown (cf. [7]) that to describe the whole class REC we need to allow also the alphabet mapping between the regular operations. This characterization of REC in terms of regular expressions seems not completely satisfactory, because it is not purely constructive and involves some external operations. Therefore the problem of proving a sort of two-dimensional Kleene's Theorem, is still under investigation. Furthermore such considerations are a clear sign that, going from one to two dimensions we find a very rich family of languages that need a non-straightforward generalization of the one-dimensional definitions and techniques.

In the next section we are going to define a new operation on picture languages and consider the class of languages that can be thereby denoted.

3. The diagonal concatenation and related regular expressions

In this section we introduce a new operation on picture languages over a one-letter alphabet. We propose some different types of regular expressions involving the new operation, comparing the resulting classes of languages obtained with known families of picture languages. Through all the section, we assume to be in the case of languages over one-letter alphabet $\Sigma = \{a\}$.

Remark 4. When a one-letter alphabet Σ is considered, any picture $p \in \Sigma^{**}$ is characterized only by its size. Therefore it can be equivalently represented either by a pair of words in Σ^*, where the first one is equal to the first column of p and the second one to the first row of p, i.e. $(a^{\ell_1(p)}, a^{\ell_2(p)})$, or simpler by its size, i.e. $(\ell_1(p), \ell_2(p))$.

Remark 5. The one-letter alphabet case means to consider the "shapes" of pictures. Indeed if $L \subseteq \Sigma^{**}$, with $|\Sigma| \geqslant 2$, is in REC then the language obtained by mapping Σ into a one-letter alphabet $\{a\}$, is still in REC, since REC is closed under alphabetic mappings.

Therefore for a language in REC, it is a necessary condition that the language of its shapes is in REC.

Let us denote by $CR = \{\cup, \obar, \ominus, *\obar, *\ominus\}$ the set of classical operations on picture languages (C for "columns" and R for "rows"), and by $L(CR)$ the class of languages (over a one-letter alphabet) that can be denoted by a regular expression involving only operations in CR and starting from finite languages. O. Matz [15] has characterized $L(CR)$ as the class of languages that are a finite union of Cartesian products of ultimately periodic string languages and he has shown that $L(CR)$ is closed under intersection.

3.1. D-regular expressions

We introduce a new simple definition of concatenation of two pictures in the particular case of one-letter alphabet. The definition is motivated by the necessity of an operation between pictures that could express some relationship existing between the dimensions of the pictures. We use this new concatenation to construct some regular expressions and to define a class of languages. This class is characterized in terms of the relations between the dimensions of the pictures and in terms of the four-way automata recognizing them.

Let p and q be two pictures of size (n, m) and (n', m') respectively over a one-letter alphabet.

Definition 6. The *diagonal concatenation* of p and q (denoted by $p \oslash q$) is a picture over Σ of size $(n + n', m + m')$. It can be represented by

$$p \oslash q = \begin{array}{|c|c|}\hline p & \\ \hline & q \\ \hline\end{array}\,.$$

Observe that, unlike the classical row and column concatenation, the diagonal concatenation is a total operation. As usual, it can be extended to define the diagonal concatenation between languages. Moreover the Kleene closure of $\oslash$ can be defined as follows. Let L be a picture language over a one-letter alphabet.

Definition 7. The *diagonal closure* or *diagonal star* of L (denoted by $L^{*\oslash}$) is defined as

$$L^{*\oslash} = \bigcup_{i \geqslant 0} L^{i\,\oslash},$$

where $L^{0\oslash} = \{\lambda_{0,0}\},\ L^{1\oslash} = L,\ L^{n\oslash} = L \oslash L^{(n-1)\oslash}$.

Example 8. Let $L_{n,n}$ be the language of squares (see Example 1) that is $L_{n,n} = \{p \mid \ell_1(p) = \ell_2(p) \geqslant 0\}$. It can be easily shown that $L_{n,n} = \{(1, 1)\}^{*\oslash} = \{\lambda_{0,1} \oslash \lambda_{1,0}\}^{*\oslash}$, observing that $\lambda_{0,1} \oslash \lambda_{1,0}$ is the picture $(1, 1)$.

Example 9. Let $L_{2n,2m}$ be the language of rectangular pictures with even dimensions, that is $L_{2n,2m} = \{p \mid l_1(p) = 2n, l_2(p) = 2m,\ n, m \geqslant 0\}$. We have that $L_{2n,2m} = \{\{(2, 2)\}^{*\ominus}\}^{*\obar}$, and also $L_{2n,2m} = \{\lambda_{0,2}\}^{*\oslash} \oslash \{\lambda_{2,0}\}^{*\oslash}$, using the diagonal concatenation.

Proposition 10. *The family REC is closed under diagonal concatenation and diagonal star.*

Proof. The proof uses similar techniques to the one for the closure of REC under row (or column) concatenation and star (see [6] for more details). A tiling system for $L = L_1 \obslash L_2$ can be defined as follows. Let the local languages for L_1 and L_2 be given by a set of tiles Θ_1 over an alphabet Γ_1 and a set of tiles Θ_2 over an alphabet Γ_2, respectively. Moreover we can always assume that Γ_1 and Γ_2 are disjoint. Then, the local language for L is defined over the alphabet $\Gamma_1 \cup \Gamma_2 \cup \{x\}$, where x is a new symbol not in $\Gamma_1 \cup \Gamma_2$. The set of tiles Θ is defined, using Θ_1 and Θ_2 in a way to represent pictures $\begin{array}{|c|c|}\hline p & s \\ \hline s' & q \\ \hline\end{array}$, where p and q belongs to the local languages of L_1 and L_2, respectively, and s, s' are pictures containing all x. For example, the non-border tiles of Θ consist of all non-border tiles in Θ_1 and Θ_2 plus the tile containing all x plus tiles obtained by replacing by x all border symbols in all right-border and bottom-border tiles in Θ_1 and all left-border and top-border tiles in Θ_2, plus tiles like $\begin{array}{|c|c|}\hline a & x \\ \hline x & b \\ \hline\end{array}$, where a and b are symbols in bottom-right corner tiles of Θ_1 and top-left corner tiles of Θ_2, respectively. Observe that the last mentioned tiles are those that "glue" bottom-right corners of pictures in L_1 to top-left corners of pictures in L_2. Finally, the projection from Γ to Σ maps all symbols to the unique symbol in Σ.

Regarding the closure under diagonal star, the tiling system for $L^{\obslash *}$ can be defined as above using two different local languages (i.e. over disjoint local alphabets) for L. $\square$

The diagonal concatenation can be used to generate families of picture languages, starting from atomic languages. Formally, let us denote $D = \{\cup, \obslash, *\obslash\}$; the elements of D are called *diagonal-regular operations*, briefly *D-regular operations*.

Definition 11. A *diagonal-regular expression* (*D-RE*) is defined recursively as follows:
(1) $\emptyset$, $(\lambda_{0,0})$, $(\lambda_{0,1})$, $(\lambda_{1,0})$ are *D-RE*.
(2) if α, β are *D-RE* then $(\alpha) \cup (\beta)$, $(\alpha) \obslash (\beta)$, $(\alpha)^{*\obslash}$ are *D-RE*.

Every D-RE denotes a language using the standard notation. Languages denoted by D-RE are called *diagonal-regular languages*, briefly *D-regular languages*. The class of D-regular languages is denoted by $L(D)$. Observe that languages containing a single picture (n, m) can be denoted by the D-RE $E_{n,m} = (\lambda_{1,0}^{n\obslash}) \obslash (\lambda_{0,1}^{m\obslash})$.

We will now characterize D-regular languages in terms of rational relations and in terms of some 4FA. For this, let us recall that (see [2]) a *rational relation* over alphabets Σ and Δ is a rational subset of the monoid $(\Sigma^* \times \Delta^*, ., (\lambda, \lambda))$, where the operation . is the componentwise product defined by $(u_1, v_1)\,(u_2, v_2) = (u_1u_2, v_1v_2)$ for any $(u_1, v_1), (u_2, v_2) \in \Sigma^* \times \Delta^*$. When the alphabet is $\Sigma = \Delta = \{a\}$, there is a natural correspondence between pictures over Σ and relations over $\Sigma \times \Sigma$. For any relation $T \subseteq \Sigma^* \times \Sigma^*$ we define the picture language:

$$L(T) = \{p \mid \ell_1(p) = |r_1| \text{ and } \ell_2(p) = |r_2| \text{ for some } (r_1, r_2) \in T\}.$$

Vice versa for any picture language $L \subseteq \Sigma^{**}$ we define the relation:

$$R(L) = \{(r_1, r_2) \in \Sigma^* \times \Sigma^* \mid |r_1| = \ell_1(p) \text{ and } |r_2| = \ell_2(p) \text{ for some } p \in L\}.$$

Remark 12. We recall that a 4NFA M over a one-letter alphabet is equivalent to a two-way two-tape automaton M_1 (cf. [10]). In fact, let H_1 and H_2 be the first and the second heads of M_1 respectively, then M_1 simulates M as follows. If the input head H of M moves down (up) one square, M_1 moves H_1 right (left) one square without moving H_2, and if H moves right (left) one square, M_1 moves H_2 right (left) without moving H_1.

Proposition 13. *Let Σ be a one-letter alphabet and let $L \subseteq \Sigma^{**}$. Then L is a D-regular language if and only if $L = L(T)$ for some rational relation $T \subseteq \Sigma^* \times \Sigma^*$ if and only if $L = L(A)$ for some 4NFA A that moves only right and down.*

Proof. In light of Remark 4, the componentwise concatenation in $M = \Sigma^* \times \Sigma^*$ exactly corresponds to the diagonal concatenation in Σ^{**}. It is well-known that a rational subset of any monoid M is either empty or can be expressed, starting with singleton, by a finite number of the (rational) operations $\cup$, . (product) and .-closure (star). Thus L is a D-regular language if and only if $L = L(T)$ for some rational relation $T \subseteq \Sigma^* \times \Sigma^*$. On the other hand, it is well-known that $T \subseteq \Sigma^* \times \Sigma^*$ is a rational relation iff it is accepted by a (finite) transducer, that is a (finite) automaton over $\Sigma^* \times \Sigma^*$. Further such an automaton can be viewed as a (finite) one-way automaton with two tapes (cf. [17]). Then, in analogy to Remark 12, one-way two-tape automata are equivalent to 4NFA that move only right and down. $\square$

Example 14. Let $L_{n,n}$ be the language of squares, as in Example 8. We have $L_{n,n} \in L(D)$. Indeed it can be easily shown that $L_{n,n}$ is denoted by the following D-RE: $E_{n,n} = (\lambda_{0,1} \oslash \lambda_{1,0})^{*\oslash}$. We have $L_{n,n} = L(T)$, where T is the rational relation $T = \{(a^n, a^n) \mid n \geqslant 0\}$. Further $L = L(A)$ where A is the 4NFA that, starting in the top-left corner, moves along the main diagonal until it eventually reaches the bottom-right corner and accepts. More generally, the languages $L_{n,n+i} = \{p \mid l_1(p) = n, l_2(p) = n + i,\ n \geqslant 1\}$, for some $i \geqslant 0$, are denoted by the D-RE: $E_{n,n+i} = E_{n,n} \obar ((E_{1,i})^{*\ominus})$, where $E_{1,i} = (\lambda_{0,1}^{i\oslash} \oslash \lambda_{1,0})$ denotes the language $\{(1, i)\}$.

Example 15. Let $L_{2n,2m}$ be the language of even sides pictures, as in Example 9, that is $L_{2n,2m} = \{p \in \Sigma^{**} \mid l_1(p) = 2n, l_2(p) = 2m,\ n, m \geqslant 0\}$. We have that $L_{2n,2m} = L(T)$, where T is the rational relation $T = \{(a^{2n}, a^{2m}) \mid n, m \geqslant 0\}$. Further $L = L(A)$, where A is the 4NFA that, starting in top-left corner, moves down checking the parity of the number of rows and then to the right checking the parity of the number of columns, eventually accepting in the bottom-right corner. Therefore $L \in L(D)$. Indeed $L \in L(CR)$ because of the characterization of $L(CR)$ given in [15].

Corollary 16. $L(CR) \subset L(D)$.

Proof. Following [15], $L(CR)$ is the class of languages that are a finite union of Cartesian products of ultimately periodic string languages. Let $L \in L(CR)$ and suppose

$L = \bigcup_{i=1,\ldots,k} A_i \times B_i$. Then L can be recognized by a 4NFA (that moves only right and down) that non-deterministically checks whether a picture belongs to some $A_i \times B_i$ checking first the belonging of the first row to A_i and then the belonging of the last column to B_i. Hence $L \in L(D)$ by Proposition 13.

Moreover the inclusion is strict since for example the language of squares is in $L(D)$ (see Example 14) and it is not in $L(CR)$ since it is not a finite union of Cartesian products of ultimately periodic string languages. □

In the same way that $L(D)$ corresponds to the class of rational relations, $L(CR)$ corresponds to its subclass of recognizable relations.

Corollary 17. *$L(D)$ is closed under intersection and complement.*

Proof. The result follows from the characterization of $L(D)$ in terms of rational relations in Proposition 13, and from the closure under intersection and complement of rational relations over a one-letter alphabet [2]. □

The following example shows that, also in the case of a one-letter alphabet, four-way automata that move only right and down are strictly less powerful than 3DFA.

Example 18. Let L be the following picture language over a one-letter alphabet: $L = \{(kn, n) \mid k, n \geqslant 0\}$. Language L can be easily recognized by a 3DFA that, starting in the top-left corner moves along the main diagonal until it reaches the right boundary and then moves along the secondary diagonal until it reaches the left boundary and so on until it eventually reaches the bottom-right corner and accepts. By Proposition 13, language L cannot be recognized by a four-way automaton that moves only right and down, since $\{(a^{kn}, a^n) \mid k, n \geqslant 0\}$ is not a rational relation (see [4]).

3.2. CRD-regular expressions

In this section we consider regular expressions that involve columns, rows and diagonal concatenations and stars defined in previous sections. We refer to them as *CRD*-regular expression. We show that, in the case of one-letter alphabet, the class $L(CRD)$ of corresponding languages is strictly included in the family REC, and strictly contains L(3DFA). Further we show that there are languages accepted by a four-way automaton that do not belong to $L(CRD)$. The main result is a necessary condition for languages in $L(CRD)$ that expresses a sort of "regularity" on the possible "extensions" of a picture (pictures containing the given one as a subpicture) inside the language.

Let us denote $CRD = \{\cup, \obar, \ominus, \obslash, *\obar, *\ominus, *\obslash\}$, where *C*, *R*, *D* stand for "column", "row" and "diagonal". The elements of *CRD* are called *CRD-regular operations*.

Definition 19. A *CRD-regular expression* (*CRD-RE*), is defined recursively as follows:
(1) $\emptyset$, $(\lambda_{0,0})$, $(\lambda_{0,1})$, $(\lambda_{1,0})$ are *CRD-RE*.
(2) if α, β are *CRD-RE* then $(\alpha) \cup (\beta)$, $(\alpha) \obar (\beta)$, $(\alpha)^{*\obar}$, $(\alpha) \ominus (\beta)$, $(\alpha)^{*\ominus}$, $(\alpha) \obslash (\beta)$, $(\alpha)^{*\obslash}$ are *CRD-RE*.

Every *CRD*-RE denotes a language using the standard notation. Languages denoted by *CRD*-RE are called *CRD-regular languages*. The family of *CRD*-regular two-dimensional languages (over one-letter alphabets) will be denoted by $L(CRD)$. Observe that $L(CRD)$ is contained in REC, since REC is closed under operations in *CRD*. A more precise positioning of $L(CRD)$ inside REC is established in Proposition 30 below.

Example 20. Let $L = \{(n, k_1(n+1)+k_2(n+2)+k_3(n+3)) \mid n, k_1, k_2, k_3 \geqslant 0\}$. Consider the languages $L_{n,n+i}$ denoted by the *D*-RE: $E_{n,n+i} = E_{n,n} \oplus ((E_{1,i})^{*\ominus})$, as in Example 14. Language $L \in L(CRD)$ since it can be denoted by the following *CRD*-RE: $E = E^{*\oplus}_{n,n+1} \oplus E^{*\oplus}_{n,n+2} \oplus E^{*\oplus}_{n,n+3}$.

Example 21. Let $L = \{(hn, hkn + n) \mid n, h, k \geqslant 0\}$. Language L belongs to $L(CRD)$. Indeed $L = L_1 \oplus L_2$, where $L_1 = \{(n, kn) \mid n, k \geqslant 0\}$ and $L_2 = \{(hm, m) \mid m, h \geqslant 0\}$. If $E_{n,n}$ is a *D*-RE for the languages of squares (see Example 14), a *CRD*-RE for L is $E = (E^{*\oplus}_{n,n}) \oplus (E^{*\ominus}_{n,n})$.

We now present some "regularity" conditions necessarily satisfied by *CRD*-regular languages. They generalize in some sense what regularity means for one-dimensional languages in what concerns the possible extensions of a picture inside a regular language. Indeed it is well-known that a string language over a one-letter alphabet $\Sigma = \{a\}$ is regular if and only if it is ultimately periodic. In particular if $L \subseteq \{a\}^*$ is a regular language and $a^n \in L$ is a sufficient long string then there exists a string a^m such that $a^n(a^m)^* \subseteq L$. We show that a generalization of this necessary condition holds for two-dimensional languages in $L(CRD)$: if a picture is sufficient "long" then we can concatenate to it some picture infinitely often by columns; if a picture is sufficient "high" then we can concatenate to it some picture infinitely often by rows; if a picture is sufficient "big" then we can concatenate to it some picture infinitely often in diagonal.

Let $\Sigma = \{a\}$ and $L \subseteq \Sigma^{**}$. Let us define for any $n, m \geqslant 0$, the following string languages: $C_n = \{a^m \mid (n, m) \in L\}$ and $R_m = \{a^n \mid (n, m) \in L\}$.

Proposition 22. *Let $L \subseteq \{a\}^{**}$ and $L \in REC$. Then for any $n, m \geqslant 0$, C_n, R_m are regular languages.*

Proof. For any alphabet Σ, and fixed n, the fixed-height-n word language of $L \subseteq \Sigma^{**}$ is the language $L(n)$ over the alphabet $\Sigma^{n,1}$, of all the strings of columns of height n that compose pictures in L. In [16], it is shown that if L is in REC, then $L(n)$ is regular, for any alphabet Σ, and any integer n. In the special case of an alphabet of a single letter, we can identify any column in $\{a\}^{n,1}$ with a and we have that $L(n)$ is regular iff C_n is regular. An analogous reasoning implies the regularity of R_m. $\square$

The proof of the following proposition is only sketched here; a more complete proof is given in Appendix A.

Proposition 23. *Let L be a CRD-regular language. Then there exist $\varphi, \psi, \overline{\varphi}, \overline{\psi} : \mathbb{N} \to \mathbb{N}$, $\rho, \xi: \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ increasing functions and $\overline{n}, \overline{m} \in \mathbb{N}$ such* *that for any* $p = (n, m) \in L$ *we have*

(1) *if* $m > \varphi(n)$ *then* $p \mathbin{\oplus} q^{*\oplus} \subseteq L$ *for* $q = (n, \overline{\varphi}(n))$ *with* $\overline{\varphi}(n) \neq 0$,

(2) *if* $n > \psi(m)$ *then* $p \ominus q^{*\ominus} \subseteq L$ *for* $q = (\overline{\psi}(m), m)$ *with* $\overline{\psi}(n) \neq 0$,

(3) *if* $n \geqslant \overline{n}$, $m \geqslant \overline{m}$ *then* $p \oslash q^{*\oslash} \subseteq L$ *for some* $q = (n_q, m_q)$ *with* $n_q, m_q \neq 0$, $n_q \leqslant \rho(n, m)$, $m_q \leqslant \xi(n, m)$.

Proof (*Sketch*). First we show how to choose the functions φ, ψ, $\overline{\varphi}$, $\overline{\psi}$. From Proposition 22, we know that the sets $C_n = \{a^m \mid (n, m) \in L\}$ and $R_m = \{a^n \mid (n, m) \in L\}$ are regular and therefore ultimately periodic. So we can define φ (ψ, resp.) in relation to the steam of the minimal automaton of C_n (R_m, resp.) and $\overline{\varphi}$ ($\overline{\psi}$, resp.) related to the period of C_n (R_m, resp.), in such a way to ensure that such functions be increasing functions.

Now we sketch how to choose $\overline{n}$, $\overline{m}$, ρ and ξ for a *CRD*-regular language L, by induction on the number of operators in a *CRD*-regular expression r that denotes L.

For the basis, if $L = \emptyset$ then the proposition is vacuously true. If $L = \{\lambda_{0,0}\}$, or $L = \{\lambda_{0,1}\}$, or $L = \{\lambda_{1,0}\}$, then we can choose $\overline{n}$, $\overline{m}$, ρ and ξ in such a way the proposition is always vacuously true, having care to define $\overline{\varphi}(n) = 1$, $\overline{\psi}(m) = 1$, $\rho(n, m) = \xi(n, m) = 1$ so that $q \neq \lambda_{0,0}$.

Suppose now $r > 0$. There are seven different cases depending on the form of r: $r = r_1 \cup r_2$, $r = r_1 \mathbin{\oplus} r_2$, $r = r_1 \ominus r_2$, $r = r_1 \oslash r_2$, $r = r_1^{*\oplus}$, $r = r_1^{*\ominus}$, or $r = r_1^{*\oslash}$. In any of the seven cases, r_1 and r_2 denote some language L_1 and L_2, respectively, that satisfies the conditions. Let φ_i, ψ_i, $\overline{\varphi}_i$, $\overline{\psi}_i$, ρ_i, ξ_i, $\overline{n}_i$, $\overline{m}_i$ be the functions and the values for L_i, with $i = 1, 2$.

The values $\overline{n}$, and $\overline{m}$ for L are chosen in such a way that a "big" picture (i.e. $p = (n, m)$ with $n \geqslant \overline{n}$ and $m \geqslant \overline{m}$) in L always decomposes in some pictures in L_1 and in L_2 that are either "big" or "long" (i.e. $m > \varphi_i(n)$, for $i = 1, 2$) or "high" (i.e. $n > \psi_i(m)$, $i = 1, 2$). Therefore $\overline{n}$ may depend on $\overline{n}_1$, $\overline{n}_2$, but also on the other functions of L_1 and L_2. As an example, when $L = L_1 \oslash L_2$ then any "big" $p = p_1 \oslash p_2$, where $p_1 \in L_1$, $p_2 \in L_2$ can be such that p_1 and p_2 are either both "big", or one of fixed size and the other one "big" or one "high" and the other one "long".

The functions ρ, ξ ensure a limitation on the size of picture q that can be diagonal concatenated infinitely many times to a big p. Such picture q is constructed from some corresponding pictures q_1 for p_1 and q_2 for p_2. The major problem is due to the partiality of column and row concatenations that requires that q_1 and q_2 must have same number of rows or columns. This problem is solved by concatenating q_1 and q_2 with itself as many times as necessary. For example $q_1^{\ominus k_1}$ and $q_2^{\ominus k_2}$ have same number of rows if we choose $k_1 = \ell_1(q_2)$ and $k_2 = \ell_1(q_1)$ (a more refined version could consider a lowest common multiple).

A special care is due to handle also the case where p is an empty column or an empty row. $\square$

The regularity conditions in Proposition 23 are stated in such a way a finite number of pictures that could "disturb" this regularity are put away, by properly defining the limitation on the size (namely $\overline{n}$, $\overline{m}$, φ, ψ). Observe that such "small" pictures may indeed have an infinite number of extensions in some direction (horizontal, vertical, diagonal). This situation is illustrated in the following Example 24.

Example 24. For a language $L_1 = \{(n_0, m_0)\}$ consisting of a single picture, the functions $\varphi_1, \overline{\varphi}_1, \psi_1, \overline{\psi}_1, \rho_1, \xi_1$ and integers $\overline{n}_1, \overline{m}_1$ as in Proposition 23 can be chosen as $\varphi_1(n) = m_0$, $\psi_1(m) = n_0$, $\overline{\varphi}_1(n) = \overline{\psi}_1(m) = \rho_1(n, m) = \xi_1(n, m) = 1$, $\overline{n}_1 = n_0 + 1$, and $\overline{m}_1 = m_0 + 1$. Indeed the conditions (1), (2) and (3) in the proposition will be vacuously true. Consider now the language of squares $L_2 = L_{n,n} = \{p \mid \ell_1(p) = \ell_2(p) \geqslant 0\}$. The functions $\varphi_2, \overline{\varphi}_2, \psi_2, \overline{\psi}_2, \rho_2, \xi_2$ and integers $\overline{n}_2, \overline{m}_2$ as in Proposition 23 for L_2 can be chosen as follows: $\varphi_2(n) = \psi_2(n) = n$, $\overline{\varphi}_2(n) = \overline{\psi}_2(n) = 1$, $\overline{n}_2 = \overline{m}_2 = 0$, and $\rho_2(n, m) = \xi_2(n, m) = 1$. Finally, consider $L = L_1 \cup L_2 = \{(n_0, m_0)\} \cup L_{n,n}$ and suppose $(n_0, m_0) \notin L_2$. According to Proposition 23 (case 1), the functions $\varphi, \overline{\varphi}, \psi, \overline{\psi}, \rho, \xi$ and integers $\overline{n}, \overline{m}$ are the following: $\varphi(n) = max\{n, m_0\}$; $\overline{\varphi}(n) = 1$; $\psi(m) = max\{n_0, m\}$; $\overline{\psi}(n) = 1$; $\overline{n} = max\{\overline{n}_1, \overline{n}_2\} = n_0 + 1$; $\overline{m} = max\{\overline{m}_1, \overline{m}_2\} = m_0 + 1$; $\rho(n, m) = max\{\rho_1(n, m), \rho_2(n, m)\} = 1$ and $\xi(n, m) = max\{\xi_1(n, m), \xi_2(n, m)\} = 1$. Observe that, even if the picture (n_0, m_0) have an infinite number of extensions, we cannot find some picture q that can be concatenate infinitely often in diagonal. The choice of $\overline{n}$ and $\overline{m}$ is made in such a way (n_0, m_0) does not satisfy the conditions $n_0 \geqslant \overline{n}$ and $m_0 \geqslant \overline{m}$, and it does not "disturb" the regularity of L.

Remark 25. In Proposition 23 we state that for any p there exists a picture q that can be concatenate to p as many times as we want. This picture may indeed depend on p as shown in the following Example 26.

Example 26. Let $L = \{(kn, n) \mid k, n \geqslant 0\} = (L_{n,n})^{*\ominus}$, where $L_{n,n}$ is the language of squares. The functions $\varphi, \overline{\varphi}, \psi, \overline{\psi}, \rho, \xi$ and integers $\overline{n}, \overline{m}$ as in Proposition 23 can be chosen as follows: $\varphi(n) = n$, $\psi(m) = 0$, $\overline{\varphi}(n) = 1$, $\overline{\psi}(m) = max\{m, 1\}$, $\overline{n} = \overline{m} = 0$, $\rho(n, m) = max\{m, 1\}$ and $\xi(n, m) = 1$. Remark that the size of picture q in case $p = (n, m)$ with $n > \psi(m)$ or $n \geqslant \overline{n}$, $m \geqslant \overline{m}$, depends on the size of p. This situation is indeed unavoidable. For example, when $p = (k'n', n')$, any $q = (n_q, n')$ such that $p \ominus q^{*\ominus} \subseteq L$ is such that $q = (k''n', n')$, thus depending on the number of columns of p, as pointed out in Remark 25.

Example 27. Let $L = \{(hn, hn + n) \mid n, h \geqslant 0\}$. We have $L = L_{n,n} \oplus (L_{n,n})^{*\ominus}$, where $L_1 = L_{n,n}$ is the language of squares. The functions $\varphi_1, \overline{\varphi}_1, \psi_1, \overline{\psi}_1, \rho_1, \xi_1$ and integers $\overline{n}_1, \overline{m}_1$ as in Proposition 23 for L_1 can be chosen as follows: $\varphi_1(n) = \psi_1(n) = n$, $\overline{\varphi}_1(n) = \overline{\psi}_1(n) = 1$, $\overline{n}_1 = \overline{m}_1 = 0$, and $\rho_1(n, m) = \xi_1(n, m) = 1$. The functions $\varphi_2, \overline{\varphi}_2, \psi_2, \overline{\psi}_2, \rho_2, \xi_2$ and integers $\overline{n}_2, \overline{m}_2$ as in Proposition 23 for $L_2 = (L_{n,n})^{*\ominus}$ can be chosen as follows: $\varphi_2(n) = n$, $\psi_2(m) = 0$, $\overline{\varphi}_2(n) = 1$, $\overline{\psi}_2(m) = max\{m, 1\}$, $\overline{n}_2 = \overline{m}_2 = 0$, $\rho_2(n, m) = max\{m, 1\}$ and $\xi_2(n, m) = 1$. According to Proposition 23 (case 2), the functions $\varphi, \overline{\varphi}, \psi, \overline{\psi}, \rho, \xi$ and integers $\overline{n}, \overline{m}$ are the following ones: $\varphi(n) = 2n$; $\overline{\varphi}(n) = 1$; $\psi(m) = m - 1$; $\overline{\psi}(n) = 1$; $\overline{n} = 1$; $\overline{m} = 0$; $\rho(n, m) = max\{m, 1\}$ and finally $\xi(n, m) = max\{m, 1\} + 1$. Observe that Proposition 23 ensures that, for any picture $p = (n', m') = (hn, hn+n)$ with $n' \geqslant 1$, $m' \geqslant 0$, there exists $q = (n_q, m_q)$ with $n_q, m_q \neq 0$, $n_q \leqslant max\{m', 1\}$, $m_q \leqslant max\{m', 1\} + 1$ such that $p \oslash q^{*\oslash} \subseteq L$. Indeed such a picture can be chosen as $q = (n, n)$, that really satisfies $n \leqslant m' = hn + n$ and $n \leqslant m' + 1 = hn + n + 1$. For example, in Fig. 1, given $p_1 = (6, 6) \in L_1$ we choose $q_1 = (1, 1)$ and $p_1 \oslash q_1^{*\oslash} \subseteq L_1$.

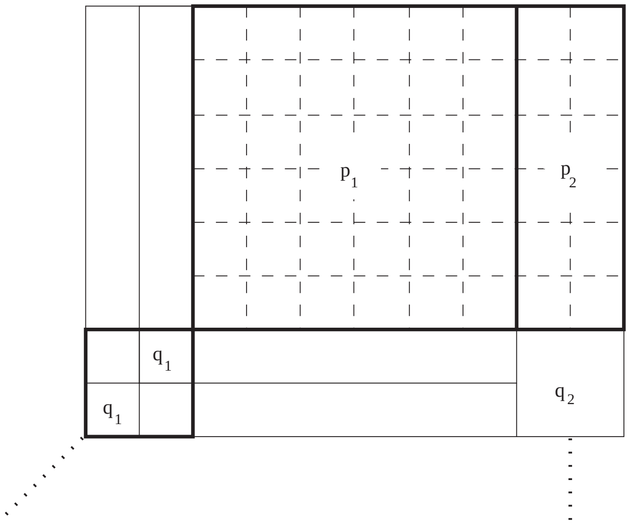

Fig. 1. Extensions of p_1, p_2 and p as in Example 27.

Given $p_2 = (6, 2) \in L_2$ we choose $q_2 = (2, 2)$ and we have $p_2 \ominus q_2^{*\ominus} \subseteq L_2$.Then, if we consider $p = p_1 \obar p_2$ we can choose $q = (2, 2)$ according to Proposition 23 (case 2) and we have $p \obslash q^{*\obslash} \subseteq L$.

Proposition 23 can be used to prove that some picture languages are not *CRD*-regular languages, as shown in the following examples.

Example 28. Let $L = \{(n, n^2) \mid n \geqslant 0\}$. We show that $L \notin L(CRD)$, proving that it does not satisfy the condition (3) in Proposition 23. Indeed suppose on the contrary that there exist $\overline{n}, \overline{m} \in \mathbb{N}$, $\rho, \xi : \mathbb{N} \to \mathbb{N}$ as in the proposition. Observe that in L, for any $n \geqslant 0$, there is only one picture with n rows and one picture with n^2 columns. Hence the pictures of L with number of rows less than or equal to $\overline{n}$ or number of columns less than or equal to $\overline{m}$ are in a finite number. Since L is infinite, then there exists a picture $p = (n, n^2) \in L$ such that $n > \overline{n}$ and $n^2 > \overline{m}$. Therefore there exists $q = (n_q, m_q)$ with $n_q, m_q \neq 0$ such that $p \obslash q^{*\obslash} \subseteq L$. Consider $p_1 = p \obslash q = (n + n_q, n^2 + m_q)$. We must have that $n^2 + m_q = (n + n_q)^2$ and thus $m_q = (n + n_q)^2 - n^2$. Consider now $p_2 = p \obslash q \obslash q = (n + 2n_q, n^2 + 2m_q)$; we have that $n^2 + 2m_q = n^2 + 2(n + n_q)^2 - 2n^2 = n^2 + 4nn_q + 2n_q^2 \neq (n + 2n_q)^2$ (since $n_q \neq 0$) against $p_2 \in L$.

Example 29. Let $L = \{(2^n, 2^n) \mid n \geqslant 0\}$. We show that $L \notin L(CRD)$, proving that it does not satisfy the condition (3) in Proposition 23. Indeed, suppose on the contrary that there exist $\overline{n}, \overline{m} \in \mathbb{N}$, $\rho, \xi : \mathbb{N} \to \mathbb{N}$ as in the proposition. Observe that in L, for any $n \geqslant 0$, there is only one picture with 2^n rows and one picture with 2^n columns. Hence the pictures of L with number of rows less than or equal to $\overline{n}$ or number of columns less than or equal to $\overline{m}$

$\overline{m}$ are in finite number. Since L is infinite, then there exists a picture $p = (2^n, 2^n) \in L$ such that $2^n > \overline{n}$ and $2^n > \overline{m}$. Therefore there exists $q = (n_q, m_q)$ with $n_q, m_q \neq 0$ such that $p \oslash q^{*\oslash} \subseteq L$. Consider $p_1 = p \oslash q = (2^n + n_q, 2^n + m_q)$. Since $p_1 \in L$, we have that $2^n + n_q = 2^n + m_q = 2^{n+k}$ for some $k \neq 0$ (since $n_q \neq 0$) and thus $n_q = m_q = 2^{n+k} - 2^n$. Consider now $p_2 = p \oslash q \oslash q = (2^n + 2n_q, 2^n + 2m_q)$; we have that $2^n + 2m_q = 2^n + 2(2^{n+k} - 2^n) = 2^n(1 + 2^{k+1} - 2) = 2^n(2^{k+1} - 1)$. Therefore $2^n + 2m_q$ is the product of a power of 2 times an odd number different from 1 and it cannot be a power of 2, against $p_2 \in L$.

We now show that the family of *CRD*-regular languages lies between the class L(3DFA) and REC. On the other hand, there are languages that belong to L(4NFA) and that are not *CRD*-regular.

Proposition 30. $L(3DFA) \subset L(CRD) \subset REC$.

Proof. Let $L \in L$(3DFA). Following [14] we have that L is a finite union of languages R whose elements are (f, g), where $f = a_0 + a_1 n$ and $g = h(b_0 + b_1 n) + b_2 n + b_3 k + b_4$ with $a_0, a_1, b_0, b_1, b_2, b_3, b_4$ positive integers and n, h, k positive integers variables. We show that any language R of this form is in $L(CRD)$. Let $E_{n,n}$ a *CRD*-RE for the language of squares (see Example 14). The language $L_{a_0,a_1,b_0,b_1} = \{(a_0 + a_1 n, b_0 + b_1 n) \mid a_0, a_1, b_0, b_1, n \in N\}$ can be denoted by the *CRD*-RE $E_{a_0,a_1,b_0,b_1} = ((a_0, b_0) \oslash ((E_{n,n})^{a_1 \ominus})^{b_1 \obar})$. Then $E = (E_{a_0,a_1,b_0,b_1})^{*\obar} \obar ((a_0, 1)^{*\obar} \ominus E_{0,a_1,0,b_2}) \obar (((1, b_3)^{*\ominus})^{*\obar}) \obar ((1, b_4)^{*\ominus})$ is a *CRD*-RE for language *R*. Moreover the inclusion L(3DFA)$\subset L(CRD)$ is strict: in fact the language $L = \{n, k_1(n+1) + k_2(n+2) + k_3(n+3)\}$ in Example 20 is in $L(CRD)$, but it cannot be recognized by a 3DFA (cf. [14]). *CRD*-regular languages are contained in REC because REC is closed under union, column and row concatenations and stars (cf. [6]) and under diagonal concatenation and star (cf. Proposition 10). Moreover there can be found examples of languages in REC that are not *CRD*-regular languages as language $L = \{(2^n, 2^n) \mid n \geqslant 0\}$ (see Example 29) or $L = \{(n, n^2) \mid n \geqslant 0\}$. □

Regarding the comparison with the class of languages recognized by four-way automata, consider language $L = \{(2^n, 2^n) \mid n \geqslant 0\}$. As shown in Example 29, L is not a *CRD*-regular language, but J. Kari and C. Moore [13] showed that L is recognized by a 4DFA. On the other hand, the class L(4DFA) seems not closed under concatenation and star operations (despite the case of one-letter alphabet is still open, it seems that for example the column closure of language in Example 21 cannot be recognized by a 4NFA).

3.3. A collection of examples

In this section, we give a collection of examples of two-dimensional languages and classify them with respect to their machine-type and regular expression-type. Languages are given by their representative element, where $n, m, h, k \geqslant 1$ are integer variables and $c \geqslant 1$ is an integer constant. Moreover $f_1(n) = a_1 + \cdots + a_n$, where $a_1, \ldots, a_n$ are all chosen in a finite subset of $\mathbb{N}$, and $f_2(n) = k_1(n+1) + k_2(n+2) + k_3(n+3)$, where $k_1, k_2, k_3 \geqslant 1$ are integer variables.

Element	2DFA	2NFA	3DFA	3NFA	4DFA	4NFA	*D*-RE	*CRD*-RE	REC
(n, n)	Y	Y	Y	Y	Y	Y	Y	Y	Y
$(2, 2n)$	Y	Y	Y	Y	Y	Y	Y	Y	Y
$(2n, 2n)$	Y	Y	Y	Y	Y	Y	Y	Y	Y
$(2n, 2m)$	Y	Y	Y	Y	Y	Y	Y	Y	Y
(n, cn)	Y	Y	Y	Y	Y	Y	Y	Y	Y
$(n, f_1(n))$	N	Y	Y	Y	Y	Y	Y	Y	Y
(kn, n)	N	N	Y	Y	Y	Y	N	Y	Y
$(n, f_2(n))$	N	N	N	Y	Y	Y	N	Y	Y
(n, kn)	N	N	N	N	Y	Y	N	Y	Y
$(2^n, 2^n)$	N	N	N	N	Y	Y	N	N	Y
$(hn, hkn + n)$	N	N	N	N	N	Y	N	Y	Y
(n, n^2)	N	N	N	N	N	N	N	N	Y
(n^2, n)	N	N	N	N	N	N	N	N	Y
(n^2, n^2)	N	N	N	N	N	N	N	N	Y
$(n, 2^n)$	N	N	N	N	N	N	N	N	Y
$(n, n!)$	N	N	N	N	N	N	N	N	N

4. Advanced star operations

Using the three types of concatenation operations (row, column and diagonal) and the three corresponding stars we get regular expressions describing a quite large family of two-dimensional languages over one-letter alphabet. Unfortunately, all those operations together are not enough to describe the whole family REC because in REC there are very "complex" languages even in the case of one-letter alphabet. For example, REC contains languages of the form $L = \{(n, f(n)) \mid n > 0\}$, as well as $L = \{(f(n), g(n)) \mid n > 0\}$, where $f(n), g(n)$ are polynomial or exponential functions in n (see [5] for details).

Observe that the peculiarities of the "classical" star operations (along which such column, row or diagonal stars are defined) are mainly the following: (a) they are a simple iteration of one kind (row- or column- or diagonal-) of concatenation between pictures; (b) they correspond to an iterative process that at each step adds (concatenates) always the same set. We can say that they correspond to the idea of the iteration for some recursive H defined as $H(1) = S$ and $H(n+1) = H(n)S$, where S is a given set.

In this section we define new types of iteration operations, to which we will refer as *advanced stars*, that result much more powerful than the "classical" ones. We will use subscripts "r"and "d" with the meaning of "right" and "down", respectively.

Definition 31. Let $L, L_\mathrm{r}, L_\mathrm{d}$ be two-dimensional languages. The *star of L with respect to* $(L_\mathrm{r}, L_\mathrm{d})$ is defined as

$$L^{(L_\mathrm{r}, L_\mathrm{d})*} = \bigcup_{i \geqslant 0} L^{(L_\mathrm{r}, L_\mathrm{d})i},$$

where $L^{(L_r,L_d)0} = \{\lambda_{0,0}\}$, $L^{(L_r,L_d)1} = L$ and

$$L^{(L_r,L_d)i+1} = \left\{ p' = \begin{array}{|c|c|}\hline p & p_r \\ \hline p_d & q \\ \hline\end{array} \mid p \in L^{(L_r,L_d)i},\ p_r \in L_r, p_d \in L_d, q \in \Sigma^{**} \right\}.$$

Remark that the operation we defined cannot be simulated by a sequence of $\oplus$ and $\ominus$ operations because to get p' we first concatenate $p \oplus p_r$ and $p \ominus p_d$, then we overlay them and finally we fill the "hole" with a picture $q \in \Sigma^{**}$. For this reason this definition is conceptually different from the one given by O. Matz in [15]. Moreover, observe that such advanced star is based on a reverse principle with respect to the diagonal star: we "decide" what to concatenate to the right and down to the given picture and then fill the hole in the bottom-right corner. This implies that, at $(i+1)$th step of the iteration, we are forced to select pictures $p_r \in L_r$ and $p_d \in L_d$ that have the same number of rows and the same number of columns, respectively, of pictures generated at the ith step. Therefore, we actually exploit the fact that column and row concatenations are partial operations to somehow *synchronize* each step of the iteration with the *choice* of pictures in L_r and L_d.

We now state the following proposition.

Proposition 32. *If L, L_r, L_d are languages in REC, then $L^{(L_r,L_d)*}$ is in REC.*

Proof. We give only few hints for the proof because it can be carried over using the techniques shown in the proof of Proposition 10. The idea is to assume that the tiling systems for L, L_r, L_d are over disjoints local alphabets Γ, Γ_r, Γ_d and define a local language M' over an alphabet Γ' equal to the union of the three ones together with a new different symbol $\{x\}$. Language M' contains pictures like $\begin{array}{|c|c|}\hline p' & p'_r \\ \hline p'_d & s \\ \hline\end{array}$, where p', p'_r and p'_d belong to the local languages for L, L_r and L_d, respectively and s is any picture filled with symbol x. Then the set of tiles for $L' = L^{(L_r,L_d)*}$ can be defined by taking two "different copies" (i.e., over disjoint local alphabets) of languages M' and different local languages for L_r and L_d and define tiles according to the definition of pictures in L'. $\Box$

As immediate application, consider the language $L = \{(n, n^2) \mid n \geqslant 0\}$ of Example 28. Then L can be defined as advanced star of $M = \{(1, 1)\}$ with respect to $M_r = \{(n, 2n + 1)\ n \geqslant 0\}$ and $M_d = \{(1, n) \mid n \geqslant 0\}$ (at $(i+1)$th step of the iteration we "add" $(2i+1)$ columns to the current i^2 ones and 1 row to the current i ones). Using the same principle, namely exchanging languages M_r and M_d, it is easy to define also the rotation of this language, i.e. language $L' = \{(n^2, n) \mid n \geqslant 0\}$. Then also the language $L'' = \{(n^2, n^2) \mid n \geqslant 0\}$ can be defined as advanced star of $M = \{(1, 1)\}$ with respect to $N_r = \{(n^2, 2n+1)\ \ n \geqslant 0\}$ and $N_d = \{(2n+1, n^2) \mid n \geqslant 0\}$, where N_r (N_d) can be obtained by column-concatenation (row-concatenation) of two copies of L' (L) and 1-row (1-column) pictures.

Remark that, even using the above defined advanced star, it seems still not possible to define the language of Example 29 of pictures of size $(2^n, 2^n)$ or the language of pictures of size $(n, 2^n)$ and similar ones. In fact, for this kind of languages (recall that they are all in REC), it would be needed a definition that allows to use as L_r and/or L_d the language itself.

We give the following definition.

Definition 33. Let L, L_d be two-dimensional languages. The *bi-iteration along the columns* of L with respect to L_d is defined as

$$L^{(*,L_d)*} = \bigcup_{i \geqslant 0} L^{(*,L_d)i},$$

where $L^{(*,L_d)0} = \{\lambda_{0,0}\}, \quad L^{(*,L_d)1} = L$ and

$$L^{(*,L_d)i+1} = \left\{ p' = \begin{array}{|c|c|}\hline p_1 & p_2 \\ \hline p_d & q \\ \hline \end{array} \mid p_1, p_2 \in L^{(*,L_d)i}, \ p_d \in L_d, \ q \in \Sigma^{**} \right\}.$$

Similarly we define the bi-iteration along the rows of L with respect to a language L_r, denoted by $L^{(L_r,*)*}$, where the $(i+1)$th step of the iteration is given by

$$L^{(L_r,*)i+1} = \left\{ p' = \begin{array}{|c|c|}\hline p_1 & p_r \\ \hline p_2 & q \\ \hline \end{array} \mid p_1, p_2 \in L^{(L_r,*)i}, \ p_r \in L_r, \ q \in \Sigma^{**} \right\}.$$

These notations naturally bring us to define also the bi-iteration along rows and columns, denoted by $L^{(*,*)*}$, where the $(i+1)$th step of the iteration is given by

$$L^{(*,*)i+1} = \left\{ p' = \begin{array}{|c|c|}\hline p_1 & p_3 \\ \hline p_2 & q \\ \hline \end{array} \mid p_1, p_2, p_3 \in L^{(*,*)i}, \ q \in \Sigma^{**} \right\}.$$

Using same techniques as in the proof of Proposition 32, one can prove that the family REC over one-letter alphabet is closed under all such bi-iteration operations.

It is immediate to verify that the language L of pictures of size $(n, 2^n)$ can be obtained from language $M = \{(1, 1)\}$ and $M_d = \{(1, n) \mid n > 0\}$ as $L = M^{(*,M_d)*}$. We conclude by observing that the language of Example 29 of pictures of size $(2^n, 2^n)$ can be obtained as a bi-iteration both along rows and columns of the same language $M = \{(1, 1)\}$.

5. Towards the general alphabet case

In this paper, we have defined new operations between pictures so that a quite wide class of two-dimensional languages over one-letter alphabet could be described in terms of regular expressions. All these languages belong to REC that is the class of recognizable languages that generalizes better to two dimensions the class of regular string languages. Next step is surely to complete the definitions of some other kind of “advanced” star operations in the aim of proving a two-dimensional Kleene’s Theorem in this simpler case of one-letter alphabet.

We also emphasize that an important goal of further work is to extend all these results to the general case of two-dimensional languages over *any* alphabet Σ (i.e. the case with more than one-letter). Observe that the definitions of diagonal concatenation and star are hard to extend to such general case, even using their characterizations in terms of rational relations or in terms of automata with only two moving directions. The main problem is that, if p, q are two pictures over Σ, to define $q \oslash q$ we need to specify two pictures r, s such that

$$p \oslash q = \begin{array}{|c|c|} \hline p & r \\ \hline s & q \\ \hline \end{array}.$$

On the other hand, the formalism of the advanced stars appears to be a more reasonable approach to the general case. Recall that, in this case, we need always to specify four pictures (or four languages). We will use subscripts r, d and c with the meaning of "right", "down" and "corner", respectively. Then, we can give the following definition that directly extends Definition 31.

Definition 34. Let $L, L_{\mathrm{r}}, L_{\mathrm{d}}, L_{\mathrm{c}}$ be two-dimensional languages over Σ. The *star of L with respect to* $(L_{\mathrm{r}}, L_{\mathrm{d}}, L_{\mathrm{c}})$ is defined as

$$L^{(L_{\mathrm{r}}, L_{\mathrm{d}}, L_{\mathrm{c}})*} = \bigcup_{i \geqslant 0} L^{(L_{\mathrm{r}}, L_{\mathrm{d}}, L_{\mathrm{c}})i},$$

where $L^{(L_{\mathrm{r}}, L_{\mathrm{d}}, L_{\mathrm{c}})0} = \{\lambda_{0,0}\}$, $\quad L^{(L_{\mathrm{r}}, L_{\mathrm{d}}, L_{\mathrm{c}})1} = L$ and

$$L^{(L_{\mathrm{r}}, L_{\mathrm{d}}, L_{\mathrm{c}})i+1} = \left\{ p' = \begin{array}{|c|c|} \hline p & p_{\mathrm{r}} \\ \hline p_{\mathrm{d}} & p_{\mathrm{c}} \\ \hline \end{array} \mid p \in L^{(L_{\mathrm{r}}, L_{\mathrm{d}}, L_{\mathrm{c}})i},\ p_{\mathrm{r}} \in L_{\mathrm{r}},\ p_{\mathrm{d}} \in L_{\mathrm{d}},\ p_{\mathrm{c}} \in L_{\mathrm{c}} \right\}.$$

Remark that this kind of star operation is not the iteration of a "classical" concatenation operation. These operations seem to be able to describe several languages in REC, despite the "regular expressions" for the two-dimensional languages in the general case will result very complex.

Appendix A.

Proposition 23. *Let L be a CRD-regular language. Then there exist $\varphi, \psi, \overline{\varphi}, \overline{\psi} : \mathbb{N} \to \mathbb{N}$, $\rho, \xi: \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ increasing functions and $\overline{n}, \overline{m} \in \mathbb{N}$ such that for any $p = (n, m) \in L$ we have*

(1) *if $m > \varphi(n)$ then $p \oplus q^{*\oplus} \subseteq L$ for $q = (n, \overline{\varphi}(n))$ with $\overline{\varphi}(n) \neq 0$,*
(2) *if $n > \psi(m)$ then $p \ominus q^{*\ominus} \subseteq L$ for $q = (\overline{\psi}(m), m)$ with $\overline{\psi}(n) \neq 0$,*
(3) *if $n \geqslant \overline{n}$, $m \geqslant \overline{m}$ then $p \oslash q^{*\oslash} \subseteq L$ for some $q = (n_q, m_q)$ with $n_q, m_q \neq 0$, $n_q \leqslant \rho(n, m)$, $m_q \leqslant \xi(n, m)$.*

Proof. First let us see how to choose $\varphi, \psi, \overline{\varphi}, \overline{\psi}$ in all these cases. From Proposition 22, we know that the sets $C_n = \{a^m \mid (n, m) \in L\}$ and $R_m = \{a^n \mid (n, m) \in L\}$ are regular and therefore ultimately periodic. So there exist $h_C, k_C, h_R, k_R \in \mathbb{N}$ such that $a^j \in C_n \Leftrightarrow a^{j+k_C} \in C_n$, for every $j \geqslant h_C$, and $a^j \in R_m \Leftrightarrow a^{j+k_R} \in R_m$, for every $j \geqslant h_R$. If we do not take care to the fact that $\varphi, \psi, \overline{\varphi}, \overline{\psi}$ have to be increasing and that $\overline{\varphi}, \overline{\psi}$ have to be $\neq 0$, then it would be sufficient to set $\varphi(n) = h_C$, $\psi(m) = h_R$, $\overline{\varphi}(n) = k_C$ and

$\overline{\psi}(m) = k_R$. But, to be sure that $\overline{\varphi}(n), \overline{\psi}(m) \neq 0$ and to assure the increase of the functions, we set $\varphi(n) = h_C + s_1 k'_C$, $\psi(n) = h_R + s_2 k'_R$, $\overline{\varphi}(n) = k_C + s_3 k'_C$ and $\overline{\psi}(m) = k_R + s_4 k'_R$, where $k'_C = max\{1, k_C\}$, $k'_R = max\{1, k_R\}$ and $s_1, s_2, s_3, s_4 \geqslant 0$ are the minimal integer such that $\varphi(n) \geqslant \varphi(n-1)$, $\psi(n) \geqslant \psi(n-1)$, $\overline{\varphi}(n) \geqslant \overline{\varphi}(n-1)$ and $\overline{\psi}(n) \geqslant \overline{\psi}(n-1)$.

Let us now show how to choose $\overline{n}$, $\overline{m}$, ρ and ξ for a *CRD*-regular language L. Let r be a *CRD*-regular expression denoting L. The proof is by induction on the number of operators in r.

For the basis, if $L = \emptyset$ then the proposition is vacuously true. If $L = \{\lambda_{0,0}\}$, then we can set $\overline{n} = 1$, $\overline{m} = 1$, $\varphi(n) = 0$, $\psi(m) = 0$. If $L = \{\lambda_{1,0}\}$ (resp. $L = \{\lambda_{0,1}\}$), then we can set $\overline{n} = 2$ (resp. $\overline{n} = 1$), $\overline{m} = 1$ (resp. $\overline{m} = 2$), $\varphi(n) = 0$ (resp. $\varphi(n) = 1$), $\psi(m) = 1$ (resp. $\psi(m) = 0$). In all these cases we can set $\overline{\varphi}(n) = 1$, $\overline{\psi}(m) = 1$, $\rho(n,m) = \xi(n,m) = 1$.

Assume now that the proposition is true for languages denoted by *CRD*-regular expression with less than i operators, $i \geqslant 1$, and let r have i operators. There are seven cases depending on the form of r: (1) $r = r_1 \cup r_2$, (2) $r = r_1 \oplus r_2$, (3) $r = r_1 \ominus r_2$, (4) $r = r_1 \oslash r_2$, (5) $r = r_1^{*\oplus}$, (6) $r = r_1^{*\ominus}$, or (7) $r = r_1^{*\oslash}$. In any of the seven cases, r_1 and r_2 denote some language L_1 and L_2, respectively, that satisfies the condition. Let $\varphi_1, \psi_1, \overline{\varphi}_1, \overline{\psi}_1, \rho_1, \xi_1, \overline{n}_1, \overline{m}_1$ be the functions and the values for L_1 and let $\varphi_2, \psi_2, \overline{\varphi}_2, \overline{\psi}_2, \rho_2, \xi_2, \overline{n}_2, \overline{m}_2$ be the functions and the values for L_2.

Case 1: We have $L = L_1 \cup L_2$. We set $\rho(n,m) = max\{\rho_1(n,m), \rho_2(n,m)\}$, $\xi(n,m) = max\{\xi_1(n,m), \xi_2(n,m)\}$, $\overline{n} = max\{\overline{n}_1, \overline{n}_2\}$, $\overline{m} = max\{\overline{m}_1, \overline{m}_2\}$.

Case 2: We have $L = L_1 \oplus L_2$. We set:

$\rho(n,m) = max\{\rho_1(n,m)\rho_2(n,m), \overline{\psi}_1(m)\rho_2(n,m), \overline{\psi}_2(m)\rho_1(n,m)\}$,

$\xi(n,m) = max\{\rho_1(n,m)\xi_2(n,m) + \rho_2(n,m)\xi_1(n,m), \overline{\psi}_1(m)\xi_2(n,m), \overline{\psi}_2(m)\xi_1(n,m)\}$,

$\overline{n} = max\{\overline{n}_1, \overline{n}_2, \psi_1(\overline{m}_1), \psi_2(\overline{m}_2)\}$,

$\overline{m} = \overline{m}_1 + \overline{m}_2$.

Now, let $p = (n,m) \in L$, with $n \geqslant \overline{n}$, $m \geqslant \overline{m}$. Clearly, $p = p_1 \oplus p_2$ for some $p_1 = (n_{p_1}, m_{p_1}) = (n, m_{p_1}) \in L_1$ and $p_2 = (n_{p_2}, m_{p_2}) = (n, m_{p_2}) \in L_2$. We have to consider three different cases:

(2a) $m_{p_1} \geqslant \overline{m}_1$ and $m_{p_2} \geqslant \overline{m}_2$, (2b) $m_{p_1} < \overline{m}_1$, (2c) $m_{p_2} < \overline{m}_2$.

(2a) Since $n_{p_1} \geqslant \overline{n}_1$, $m_{p_1} \geqslant \overline{m}_1$, $n_{p_2} \geqslant \overline{n}_2$ and $m_{p_2} \geqslant \overline{m}_2$, from the hypothesis on L_1 and L_2, we have that $p_1 \oslash q_1^{*\oslash} \subseteq L_1$ for some $q_1 = (n_{q_1}, m_{q_1})$ with $n_{q_1}, m_{q_1} \neq 0$, $n_{q_1} \leqslant \rho_1(n, m_{p_1})$, $m_{q_1} \leqslant \xi_1(n, m_{p_1})$ and that $p_2 \oslash q_2^{*\oslash} \subseteq L_2$ for some $q_2 = (n_{q_2}, m_{q_2})$ with $n_{q_2}, m_{q_2} \neq 0$, $n_{q_2} \leqslant \rho_2(n, m_{p_2})$, $m_{q_2} \leqslant \xi_2(n, m_{p_2})$.

Now let us set $q = (n_{q_1}n_{q_2}, n_{q_1}m_{q_2} + n_{q_2}m_{q_1}) = (n_q, m_q)$. Then $p \oslash q^{*\oslash} \subseteq L$ with $n_q, m_q \neq 0$, $n_q = n_{q_1}n_{q_2} \leqslant \rho_1(n, m_{p_1})\rho_2(n, m_{p_2}) \leqslant \rho_1(n,m)\rho_2(n,m)$ and $m_q = n_{q_1}m_{q_2} + n_{q_2}m_{q_1} \leqslant \rho_1(n,m)\xi_2(n,m) + \rho_2(n,m)\xi_1(n,m)$.

(2b) Since $m_{p_1} < \overline{m}_1$, then $m_{p_2} \geqslant \overline{m}_2$ (recall that $m_{p_1} + m_{p_2} = m \geqslant \overline{m} = \overline{m}_1 + \overline{m}_2$) and therefore $p_2 \oslash q_2^{*\oslash} \subseteq L_2$ for some $q_2 = (n_{q_2}, m_{q_2})$ with $n_{q_2}, m_{q_2} \neq 0$, $n_{q_2} \leqslant \rho_2(n, m_{p_2})$, $m_{q_2} \leqslant \xi_2(n, m_{p_2})$. Moreover $n_{q_1} = n \geqslant \overline{n} \geqslant \psi_1(\overline{m}_1) > \psi_1(m_{p_1})$: therefore $p_1 \ominus q_1^{*\ominus} \subseteq L_1$ for $q_1 = (n_{q_1}, m_{q_1}) = (\overline{\psi}_1(m_{p_1}), m_{p_1})$. Note that we have $n_{q_1} \neq 0$. Let us set $q = (n_{q_1}n_{q_2}, n_{q_1}m_{q_2}) = (n_q, m_q)$. Then we have $p \oslash q^{*\oslash} \subseteq L$ with $n_q, m_q \neq 0$, $n_q = n_{q_1}n_{q_2} \leqslant \overline{\psi}_1(m_{p_1})\rho_2(n, m_{p_2}) \leqslant \overline{\psi}_1(m)\rho_2(n,m)$ and $m_q = n_{q_1}m_{q_2} \leqslant \overline{\psi}_1(m_{p_1})\xi_2(n, m_{p_2}) \leqslant \overline{\psi}_1(m)\xi_2(n,m)$.

(2c) It is analogous to the previous case.

Case 3: We have $L = L_1 \ominus L_2$ and the proof is similar to that one of the previous case.

Case 4: We have $L = L_1 \oslash L_2$. We set:

$\rho(n,m) = max\{\rho_1(n,m), \rho_2(n,m), \overline{\psi}_1(m), \overline{\psi}_2(m)\}$,

$\xi(n,m) = max\{\xi_1(n,m), \xi_2(n,m), \overline{\varphi}_2(n), \overline{\varphi}_1(n)\}$,

$\overline{n} = max\{\overline{n}_1 + \overline{n}_2, \psi_1(\overline{m}_1) + \overline{n}_2, \psi_2(\overline{m}_2) + \overline{n}_1\}$,

$\overline{m} = max\{\overline{m}_1 + \overline{m}_2, \varphi_2(\overline{n}_2) + \overline{m}_1, \varphi_1(\overline{n}_1) + \overline{m}_2\}$.

Now, let $p = (n,m) \in L = L_1 \oslash L_2$, with $n \geqslant \overline{n}$, $m \geqslant \overline{m}$. Clearly, $p = p_1 \oslash p_2$ for some $p_1 = (n_{p_1}, m_{p_1}) \in L_1$ and $p_2 = (n_{p_2}, m_{p_2}) \in L_2$. We have to consider two different cases 4(a) and (b): with $n_q, m_q \neq 0$

(4a) At least one of the following conditions (1) and (2) is verified

$$(1) \begin{cases} n_{p_1} \geqslant \overline{n}_1, \\ m_{p_1} \geqslant \overline{m}_1. \end{cases} \qquad (2) \begin{cases} n_{p_2} \geqslant \overline{n}_2, \\ m_{p_2} \geqslant \overline{m}_2. \end{cases}$$

If condition (1) is verified, then $p_1 \oslash q_1^{*\oslash} \subseteq L_1$ for some $q_1 = (n_{q_1}, m_{q_1})$ with $n_{q_1}, m_{q_1} \neq 0$, $n_{q_1} \leqslant \rho_1(n,m)$, $m_{q_1} \leqslant \xi_1(n,m)$ and it suffices to set $q = q_1$. If, instead, condition (2) is verified, then $p_2 \oslash q_2^{*\oslash} \subseteq L_2$ for some $q_2 = (n_{q_2}, m_{q_2})$ with $n_{q_2}, m_{q_2} \neq 0$, $n_{q_2} \leqslant \rho_2(n,m)$, $m_{q_2} \leqslant \xi_2(n,m)$ and it suffices to set $q = q_2$.

(4b) If neither condition (1) nor condition (2) is verified, then, again, we have to consider two different subcases either $n_{p_1} \geqslant \overline{n}_1$, $m_{p_1} < \overline{m}_1$, $n_{p_2} < \overline{n}_2$, $m_{p_2} \geqslant \overline{m}_2$ or $n_{p_1} < \overline{n}_1$, $m_{p_1} \geqslant \overline{m}_1$, $n_{p_2} \geqslant \overline{n}_2$, $m_{p_2} < \overline{m}_2$. We give the details only for the first subcase, since the other one can be handled in a similar way. So, in the first subcase, we have $n_{p_1} = n - n_{p_2} \geqslant \overline{n} - n_{p_2} > \overline{n} - \overline{n}_2 \geqslant \psi_1(\overline{m}_1) + \overline{n}_2 - \overline{n}_2 = \psi_1(\overline{m}_1) > \psi_1(m_{p_1})$ i.e., $n_{p_1} > \psi_1(m_{p_1})$ and $m_{p_2} = m - m_{p_1} \geqslant \overline{m} - m_{p_1} > \overline{m} - \overline{m}_1 \geqslant \varphi_2(\overline{n}_2) + \overline{m}_1 - \overline{m}_1 = \varphi_2(\overline{n}_2) > \varphi_2(n_{p_2})$ i.e., $m_{p_2} > \varphi_2(n_{p_2})$. Therefore, $p_1 \ominus q_1^{*\ominus} \subseteq L_1$ for $q_1 = (\overline{\psi}_1(m_{p_1}), m_{p_1})$ and $p_2 \oplus q_2^{*\oplus} \subseteq L_2$ for $q_2 = (n_{p_2}, \overline{\varphi}_2(n_{p_2}))$. We set $q = (n_q, m_q) = (n_{q_1}, m_{q_2}) = (\overline{\psi}_1(m_{p_1}), \overline{\varphi}_2(n_{p_2}))$ and we will have $p \oslash q^{*\oslash} \subseteq L$ with $n_q, m_q \neq 0$, $n_q = \overline{\psi}_1(m_{p_1}) \leqslant \overline{\psi}_1(m)$, $m_q = \overline{\varphi}_2(n_{p_2}) \leqslant \overline{\varphi}_2(n)$.

Case 5: We have $L = L_1^{*\oplus}$. We set $\rho(n,m) = max\{\rho_1(n,m), \rho_1^m(n,m)\overline{\psi}_1^m(m)\}$, $\xi(n,m) = max\{m\xi_1(n,m)\rho_1^{m-1}(n,m)\overline{\psi}_1^m(m), \xi_1(n,m)\}$, $\overline{n} = max\{\overline{n}_1, \psi_1(\overline{m}_1)\}$ and $\overline{m} = \overline{m}_1$.

Now, let $p = (n,m) \in L$, with $n \geqslant \overline{n}$, $m \geqslant \overline{m}$. If $m = 0$, then $p \in L_1$ and we can apply the inductive hypothesis. If instead $m \neq 0$, then we have $p = p_1 \oplus \cdots \oplus p_k$ with $p_i = (n_{p_i}, m_{p_i}) = (n, m_{p_i}) \in L_1$. Let us consider two different subcases 5(a) and (b).

(5a) There exists some $\bar{\imath} \in \{1, \ldots, k\}$ such that $m_{p_i} \geqslant \overline{m}_1$ for every $i = 1, \ldots, \bar{\imath}$ and $m_{p_i} < \overline{m}_1$ for every $i = \bar{\imath}+1, \ldots, k$. Therefore, for every $i = 1, \ldots, \bar{\imath}$, there exists $q_i = (n_{q_i}, m_{q_i})$ with $n_{q_i}, m_{q_i} \neq 0$, $n_{q_i} \leqslant \rho_1(n_{p_i}, m_{p_i})$, $m_{q_i} \leqslant \xi_1(n_{p_i}, m_{p_i})$, such that $p_i \oslash q_i^{*\oslash} \subseteq L_1$. Note that for $i = 1, \ldots, \bar{\imath}$, we have $n_{q_i} \leqslant \rho_1(n_{p_i}, m_{p_i}) = \rho_1(n, m_{p_i}) \leqslant \rho_1(n,m)$, $m_{q_i} \leqslant \xi_1(n_{p_i}, m_{p_i}) = \xi_1(n, m_{p_i}) \leqslant \xi_1(n,m)$. Moreover, since for every $i = \bar{\imath}+1, \ldots, k$, we have $m_{p_i} < \overline{m}_1$, it follows that $\psi_1(m_{p_i}) < \psi_1(\overline{m}_1) \leqslant \overline{n} \leqslant n = n_{p_i}$. So for every $i = \bar{\imath}+1, \ldots, k$, there exists $q_i = (n_{q_i}, m_{q_i}) = (\overline{\psi}_1(m_{q_i}), m_{q_i})$ such that $p_i \ominus q_i^{*\ominus} \subseteq L_1$. We set $q = (n_q, m_q) = (\prod_{i=1}^k n_{q_i}, \sum_{i=1}^{\bar{\imath}} (m_{q_i} \prod_{j=1, j\neq i}^k n_{q_j}))$. Then $p \oslash q^{*\oslash} \subseteq L$, where $n_q, m_q \neq 0$, with $n_q \leqslant \rho_1^{\bar{\imath}}(n,m)\, \overline{\psi}_1(m_{q_{\bar{\imath}+1}}) \ldots \overline{\psi}_1(m_{q_k}) \leqslant \rho_1^m(n,m)\overline{\psi}_1^m(m)$ and $m_q =$

$m_{q_1}\,\rho_1^{m-1}(n,m)\,\overline{\psi}_1^{\,m}(m)+\cdots+m_{q_{\bar{\imath}}}\rho_1^{m-1}(n,m)\overline{\psi}_1^{\,m}(m)\leqslant\xi_1(n,m)\bar{\imath}\rho_1^{m-1}(n,m)\overline{\psi}_1^{\,m}(m)\leqslant m\xi_1(n,m)\rho_1^{m-1}(n,m)\overline{\psi}_1^{\,m}(m)$.

(5b) In this subcase, for every $i=1,\ldots,k$, $m_{p_i}<\overline{m}_1$. Therefore, as in case (5a), for every $i=1,\ldots,k$, there exists $q_i=(n_{q_i},m_{q_i})=(\overline{\psi}_1(m_{q_i}),m_{q_i})$ such that $p_i\ominus q_i^{*\ominus}\subseteq L_1$. We set $q=(n_q,m_q)=(\prod_{i=1}^k n_{q_i},m)$. Then n_q, $m_q\neq 0$, $n_q\leqslant\overline{\psi}_1^{\,m}(m)\leqslant\rho_1^m(n,m)\overline{\psi}_1^{\,m}(m)$, $m_q=m\leqslant m\xi_1(n,m)\rho_1^{m-1}(n,m)\overline{\psi}_1^{\,m}(m)$.

Case 6: This case is analogous to the previous one.

Case 7: We have $L=L_1^{*\oslash}$. If there exists $q=(n_q,m_q)\in L$ with $n_q,m_q\neq 0$, we can set $\rho(n,m)=n_q$, $\xi(n,m)=m_q$, $\overline{n}=\overline{m}=0$. Then for every $p=(n,m)\in L=L_1^{*\oslash}$ we have $p\oslash q^{*\oslash}\subseteq L$. If instead $L\subseteq\Lambda_{\text{col}}$ (resp. $L\subseteq\Lambda_{\text{row}}$), then we can set $\overline{n}=0$, $\overline{m}=1$ (resp. $\overline{n}=1$, $\overline{m}=0$) and condition (3) will be vacuously true.

Note that in all the cases the choice of the functions φ, ψ, $\overline{\varphi}$, $\overline{\psi}$, ρ and ξ preserves their increase. □

References

[1] M. Anselmo, D. Giammarresi, M. Madonia, Regular expressions for two-dimensional languages over one-letter alphabet, in: C.S. Calude, E. Calude, M.J. Dinneen (Eds.), Proc. Development in Language Theory (DLT 04), Lecture Notes in Computer Science, Vol. 3340, Springer, Berlin, 2004, pp. 63–75.

[2] J. Berstel, Transductions and Context-free Languages, Teubner, 1979.

[3] M. Blum, C. Hewitt, Automata on a two-dimensional tape, IEEE Symp. on Switching and Automata Theory, 1967, pp. 155–160.

[4] L. De Prophetis, S. Varricchio, Recognizability of rectangular pictures by Wang systems, J. Automat. Languages Combin. 2 (1997) 269–288.

[5] D. Giammarresi, Two-dimensional languages and recognizable functions, in: G. Rozenberg, A. Salomaa (Eds.), Proc. Developments in Language Theory, Finland, 1993, World Scientific Publishing Co., Singapore, 1994.

[6] D. Giammarresi, A. Restivo, Two-dimensional finite state recognizability, Fund. Inform. 25 (3,4) (1996) 399–422.

[7] D. Giammarresi, A. Restivo, Two-dimensional languages, in: G. Rozenberg, et al. (Eds.), Handbook of Formal Languages, Vol. III, Springer, Berlin, 1997, pp. 215–268.

[8] J.E. Hopcroft, J.D. Ullman, Introduction to Automata Theory, Languages and Computation, Addison-Wesley, Reading, MA, 1979.

[9] K. Inoue, A. Nakamura, Some properties of two-dimensional on-line tessellation acceptors, Inform. Sci. 13 (1977) 95–121.

[10] K. Inoue, A. Nakamura, Two-dimensional finite automata and unacceptable functions, Internat. J. Comput. Math. Sec. A 7 (1979) 207–213.

[11] K. Inoue, I. Takanami, A characterization of recognizable picture languages, in: Proc. Second Internat. Colloq. on Parallel Image Processing, Lecture Notes in Computer Science, Vol. 654, Springer, Berlin, 1993.

[12] K. Inoue, I. Takanami, A. Nakamura, A note on two-dimensional finite automata, Inform. Process. Lett. 7 (1) (1978) 49–52.

[13] J. Kari, C. Moore, Rectangles and squares recognized by two-dimensional automata, In: J. Karhumaki, H. Maurer, G. Paun, G. Rozenburg (Eds.), Theory is Forever, Essays dedicated to Arto Salomaa on the occasion of his 70th birthday. LNCS 3113 (2004) 134–144.

[14] E.B. Kinber, Three-way automata on rectangular tapes over a one-letter alphabet, Inform. Sci. 35 (1985) 61–77.

[15] O. Matz, Regular expressions and context-free grammars for picture languages, Proc. STACS'97, Lecture Notes in Computer Science, Vol. 1200, Springer, Berlin, 1997, pp. 283–294.

[16] O. Matz, On piecewise testable, starfree, and recognizable picture languages, in: M. Nivat (Ed.), Foundations of Software Science and Computation Structures, Vol. 1378, Springer, Berlin, 1998.
[17] M. Rabin, T. Scott, Finite automata and their decision problems, IBM J. Res. Dev. 3 (1959) 114–125.
[18] D. Simplot, A characterization of recognizable picture languages by tilings by finite sets, Theoret. Comput. Sci. 218 (2) (1999) 297–323.
[19] T. Wilke, Star-free picture expressions are strictly weaker than first-order logic, in: Proc. ICALP'97, Lecture Notes in Computer Science, Vol. 1256, Springer, Berlin, 1997, pp. 347–357.