Two-dimensional Dyck words (Extended Abstract)

Stefano Crespi Reghizzi[1], Antonio Restivo[2], and Pierluigi San Pietro[1]

[1] Politecnico di Milano - DEIB
[2] Dipartimento di Matematica e Informatica, Università di Palermo
stefano.crespireghizzi@polimi.it antonio.restivo@unipa.it
pierluigi.sanpietro@polimi.it

Introduction and preliminaries The Dyck language is a central concept in formal language theory. It is defined over the alphabet $\{a_1, \ldots, a_k, a'_1, \ldots, a'_k\}$, for any $k \geq 1$, as the set of all words that can be reduced to the empty word by cancellations $a_i a'_i \to \varepsilon$. Motivated by our interest for the theory of two-dimensional (2D) or picture languages, we are investigating the possibility to transport the Dyck concept from one dimension to 2D. When moving from 1D to 2D, most formal concepts and relationships drastically change. In particular, in 2D the Chomsky's language hierarchy is blurred because the notions of regularity and context-freeness cannot be formulated for pictures without giving up some characteristic properties that hold for words. In fact, it is known [6] that the three equivalent definitions of regular languages by means of finite-state recognizer, by regular expressions, and by homomorphism of local languages, produce in 2D three distinct language families. The third one gives the family of *tiling system recognizable languages* (REC) [6], that many think to be the best fit for regularity in 2D.
The situation is less satisfactory for context-free (CF) languages where a transposition in 2D remains problematic. None of the existing proposals of "CF" picture grammars ([12, 7, 8, 10, 3, 4], a survey is [2]) match the expressiveness and richness of formal properties of CF 1D grammars. We make the first step towards a new definition of CF 2D languages via the 2D reformulation of Chomsky-Schützenberger theorem (as in [1, 9]): a CF 2D language is the homomorphic letter-to-letter image of the intersection of a 2D Dyck language and a 2D local language. Although there may exist no definition which generalizes all interesting properties of 1D Dyck languages, it is worth formalizing and comparing several possible choices; this is our contribution, while the study of the resulting 2D CF languages is still under way and not reported here.
We show four definitions of 2D "Dyck" languages based on various approaches, an initial study of their properties and their respective inclusions.
Picture Languages. A *picture* is a rectangular array of letters over a finite alphabet. The set of all non-empty pictures over Σ is denoted by Σ^{++}. A pixel is the letter in a given position of the array. Given a picture p, $|p|_{row}$ and $|p|_{col}$ denote the number of rows and columns, respectively; $|p| = (|p|_{row}, |p|_{col})$ denotes the *picture size*. We refer the reader to standard definitions of 2D languages, as given for instance in [6], in particular for the concepts of horizontal $⦶$ and vertical $\ominus$ concatenations and their closure, and for the *Simplot closure* [11] operation L^{**} defined for any 2D language L.
Dyck languages are basic concepts in formal language theory. For a Dyck language $D_k \subseteq \Gamma_k^*$, the alphabet has size $|\Gamma_k| = 2k$ and is partitioned into two sets of cardinality $k \geq 1$, denoted $\{a_i \mid 1 \leq i \leq k\} \cup \{a'_i \mid 1 \leq i \leq k\}$. D_k has several, equivalent,

definitions, such as the cancellation rule or a *nesting accretion* rule: given a word $x \in \Gamma_k^*$, a nesting accretion of x is a word of the form $a_i x a_i'$; define D_k as the smallest set including the empty word and closed under concatenation and nesting accretion. An equivalent definition can be given by a *neutralization* rule: given $N \notin \Gamma_k$, for each word in $(\Gamma_k \cup \{N\})^*$ define the congruence $\approx$, for all $i \leq i \leq k$, and for all $m \geq 0$ as: $a_i N^m a_i' \approx N^{m+2}$. A word $x \in \Gamma_k^*$ is in D_k if it is ε or it is $\approx$-congruent to $N^{|x|}$.

Box-based choices of 2D Dyck languages We present two simple choices, called well-nested and neutralizable, each one conserving one of the characteristic properties of Dyck words. To make the analogy more evident, we represent in 2D the parentheses pair [,] by a quadruple of corners $\ulcorner$, $\urcorner$, $\llcorner$, $\lrcorner$ (for simplicity often denoted as a, b, c, d). Inside a picture such a quadruple matches if it is laid on the four vertexes of a rectangle (i.e., a subpicture), as in the picture $\begin{array}{cccc} \ulcorner & \ulcorner & \urcorner & \urcorner \\ & & & \\ \llcorner & \llcorner & \lrcorner & \lrcorner \end{array}$ for each quadruple identified by a color.

Definition 1 (well-nested 2D Dyck language). *Let* $\Delta_k = \{a_i, b_i, c_i, d_i \mid 1 \leq i \leq k\}$. *Define two bijections:* $h_r : \{a_i, b_i\} \to \{c_i, d_i\}$, $h_c : \{a_i, c_i\} \to \{b_i, d_i\}$ *with* $h_r(a_i) = c_i$, $h_r(b_i) = d_i$ *and* $h_c(a_i) = b_i$, $h_c(c_i) = d_i$.
For every picture $p \in \Delta_k^{++}$, *for all rows* w_r *in the (word) Dyck language over the parentheses* (a_i, b_i), *and for all columns* w_c *in the Dyck language over parentheses* (a_i, c_i), *such that* $|w_r| = |p|_{col}$, $|w_c| = |p|_{row}$, *the* nesting accretion *of* p *within* w_r, w_c *is the picture:* $(a_i \oplus w_r \oplus b_i) \ominus (w_c \oplus p \oplus h_c(w_c)) \ominus (c_i \oplus h_r(w_r) \oplus d_i)$.
The language DW_k *is the smallest set including the empty picture and closed under nesting accretion and Simplot closure.*

Fig. 1 (right) illustrates accretion and (left) shows a picture in DW_1 (when $k = 1$, $\Delta_k = \{a, b, c, d\}$). The definition can be explained intuitively by considering two distinct occurrences of a quadruple of corners: the subpictures delimited by each quadruple (i.e., their bounding boxes) are either disjoint, or included one into the other; or they overlap and a third box exists that "minimally" bounds both boxes. The third case is illustrated in Fig. 1, left, by the overlapping blue and green boxes.
It is immediate to see that for any size $(2m, 2n)$, $m, n \geq 1$, there is a picture in DW_k; moreover, DW_k is not (tiling system) recognizable (see Th. 2).
We now investigate a definition of 2D Dyck languages, called DN_k, by means of a neutralization rule analogous to the congruence of Dyck word languages: a DN_k picture is transformed into a picture in N^{**}, where N is a new symbol, by a series of neutralizations. Let $N^{m,n}$ be the homogeneous picture of size (m, n) in N^{**}.

Definition 2 (neutralizable Dyck language). *Let* N *be a symbol not in* Δ_k. *The* neutralization relation $\overset{\nu}{\to} \subseteq (\{N\} \cup \Delta_k)^{++} \times (\{N\} \cup \Delta_k)^{++}$, *is the smallest relation such that for all pictures* p, p' *in* $(\{N\} \cup \Delta_k)^{++}$, $p \overset{\nu}{\to} p'$ *if there are* $m, n \geq 2$ *and* $1 \leq i \leq k$, *such that* p' *is obtained from* p *by replacing a subpicture of* p *of the form:* $(a_i \ominus N^{m-2,1} \ominus c_i) \oplus N^{m,n-2} \oplus (b_i \ominus N^{m-2,1} \ominus d_i)$ *with the isometric picture* $N^{m,n}$.
The 2D neutralizable Dyck language, *denoted with* $DN_k \subseteq \Delta_k^{++}$, *is the set of pictures* p *such that there exists* $p' \in N^{++}$ *with* $p \overset{\nu}{\to}^{+} p'$.

Example 1 (neutralizations). The following picture p_1 on the alphabet Δ_1 is in DN_1 since it reduces to the neutral one by means of a sequence of six neutralization steps:

$$p_1 = \begin{array}{cccccc} \ulcorner & \ulcorner & \urcorner & \ulcorner & \urcorner & \urcorner \\ \ulcorner & \ulcorner & \urcorner & \llcorner & \lrcorner & \urcorner \\ \llcorner & \llcorner & \lrcorner & \ulcorner & \urcorner & \lrcorner \\ \llcorner & \llcorner & \lrcorner & \llcorner & \lrcorner & \lrcorner \end{array} \overset{\nu}{\to} \begin{array}{cccccc} \ulcorner & \ulcorner & \urcorner & \ulcorner & \urcorner & \urcorner \\ \ulcorner & N & N & \llcorner & \lrcorner & \urcorner \\ \llcorner & N & N & \ulcorner & \urcorner & \lrcorner \\ \llcorner & \llcorner & \lrcorner & \llcorner & \lrcorner & \lrcorner \end{array} \overset{\nu}{\to} \begin{array}{cccccc} \ulcorner & N & N & \ulcorner & \urcorner & \urcorner \\ \ulcorner & N & N & \llcorner & \lrcorner & \urcorner \\ \llcorner & N & N & \ulcorner & \urcorner & \lrcorner \\ \llcorner & N & N & \llcorner & \lrcorner & \lrcorner \end{array} \overset{\nu}{\to} \begin{array}{cccccc} \ulcorner & N & N & N & N & \urcorner \\ \ulcorner & N & N & N & N & \urcorner \\ \llcorner & N & N & \ulcorner & \urcorner & \lrcorner \\ \llcorner & N & N & \llcorner & \lrcorner & \lrcorner \end{array}$$

$$\overset{\nu}{\to} \begin{array}{cccccc} \ulcorner & N & N & N & N & \urcorner \\ \ulcorner & N & N & N & N & \urcorner \\ \llcorner & N & N & N & N & \lrcorner \\ \llcorner & N & N & N & N & \lrcorner \end{array} \overset{\nu}{\to} \begin{array}{cccccc} \ulcorner & N & N & N & N & \urcorner \\ N & N & N & N & N & N \\ N & N & N & N & N & N \\ \llcorner & N & N & N & N & \lrcorner \end{array} \overset{\nu}{\to} \begin{array}{cccccc} N & N & N & N & N & N \\ N & N & N & N & N & N \\ N & N & N & N & N & N \\ N & N & N & N & N & N \end{array}$$

Neutralizations have been arbitrarily applied in top to bottom, left to right order, since the order of application of the neutralization steps is irrelevant.

Although DW_k is defined by a diverse mechanism, DW_k is included in DN_k (Th. 2); the inclusion is strict since the picture of Fig. 2 is in $DN_1 \setminus DW_1$.

Row-column combination of Dyck languages We consider the pictures, called Dyck crosswords, such that their rows and columns are Dyck word languages over the same alphabet but with different pairing of terminal characters. They may be viewed as analogous of Dyck word languages. Following [6] we introduce the row-column combination operation that takes two word languages and produces a 2D language.

Definition 3 (row-column combination a.k.a. crossword). *Let $S', S'' \subseteq \Sigma^*$ be two word languages, called* component languages*. The* row-column combination *or* crossword *of S' and S'' is the 2D language L such that a picture $p \in \Sigma^{++}$ belongs to L if and only if the words corresponding to each row (in left-to-right order) and to each column (in top-down order) of p belong to S' and S'', respectively.*

Row-column combinations of regular languages are are called "regex crosswords" in [5] where some complexity issues are studied; they are important since their alphabetic projection coincide with the REC family [6]. We investigate the properties of the row-column combination of Dyck languages. The alphabet is still $\Delta_k = \{a_i, b_i, c_i, d_i \mid 1 \le i \le k\}$. We interpret Δ_k as two different Dyck alphabets, the *Dyck row alphabet* Δ_k^{Row} and the *Dyck column alphabet* Δ_k^{Col} as follows, allowing the definition of the corresponding ε-free Dyck languages: $D_k^{Row} \subset (\Delta_k^{Row})^+$ and $D_k^{Col} \subset (\Delta_k^{Col})^+$.

Definition 4 (Dyck crossword alphabet and language).
$\Delta_k^{Row} = \{a_i, b_i \mid i \le 1 \le k\} \cup \{c_i, d_i \mid 1 \le i \le k\}$*, where all a_i, c_i are open parenthesis and b_i, d_i are their respective closed parenthesis.*
$\Delta_k^{Col} = \{a_i, c_i \mid 1 \le i \le k\} \cup \{b_i, d_i \mid 1 \le i \le k\}$*, where all a_i, b_i are open parenthesis and c_i, d_i are their respective closed parenthesis.*
The Dyck crossword *language DC_k is the row-column combination of D_k^{Row} and D_k^{Col}.*

It is easy to notice that $DN_k \subseteq DC_k$: for instance, when neutralizing a subpicture, the neutralization of its two corners (a_i, b_i) acts in that row as the neutralization rule for words in D_k^{row}, and similarly for the other corners. The inclusion is proper (see Th. 2).

Any picture p that is partitioned into DC_k subpictures is also in DC_k. This is obvious since each row of p is the concatenation of Dyck words, and similarly for columns. An analogous result holds for each language DN_k (for DW_k this holds by definition).

Another question for any of the Dyck-like 2D languages introduced is whether its row and column languages saturate the horizontal and vertical Dyck word languages. This is the case for DN_k and DC_k, but not for DW_k.

DC_k pictures may contain a rich variety of patterns; we present some and state a formal property on the valid patterns. The simplest patterns are in pictures partitioned into rectangular circuits connecting four elements, e.g., Fig. 2, right, where an edge connects two symbols on the same row (or column) which match in the row (column) Dyck word. Notice that the graph made by the edges contains four disjoint circuits of length four, called *rectangles* for brevity. Three of the circuits are nested inside the outermost one.

A picture in DC_k may also include *circuits* longer than four. In Fig. 3 (left) we see a circuit of length 12, labeled by the word $(abdc)^3$, and on the right a circuit of length 36. The pixels of every DC_k picture p can be seen as the nodes of a graph, called *matching graph* of p. The graph is partitioned into disjoint simple circuits, i.e. each DC_k picture p consists of a set of such circuits positioned on the picture. Therefore, there is a horizontal edge connecting two matching letters a_i, b_i or c_i, d_i that occur in the same row: e.g., the edge $(2,1) \leftrightarrow (2,4)$ of Figure 3, left. Analogously, there is a vertical edge connecting two matching letters a_i, c_i or b_i, d_i, that occur in the same column: e.g., the edge $(2,2) \leftrightarrow (3,2)$ of Figure 3, left. When a picture is represented by its matching graph, the node labels are redundant since they are uniquely determined on each circuit of the graph: the clockwise visit of any such circuit, starting from one of its nodes with label a_j, yields a word in the language $(a_j b_j d_j c_j)^+$.

Theorem 1 (Unbounded circuit length). *For all $h \geq 0$ there exist a picture in DC_h that contains a circuit of length $4 + 8h$.*

Another series of pictures that can be enlarged indefinitely is the one in Fig. 3, where the first two terms of the series are shown. The next definition forbids any cycle longer than 4 and keeps, e.g., the pictures in Fig. 2 and 5.

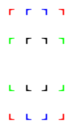
Definition 5 (Quaternate DC_k). *A Dyck crossword picture such that all its circuits are of length 4 is called* quaternate*; their language DQ_k is the* quaternate Dyck language*.*

Since DC_k pictures may contain circuits of length > 4, (e.g., in Fig. 3) quaternate Dyck languages are strictly included in Dyck crosswords.

The following theorem summarizes some results for the various 2D Dyck languages.

Theorem 2 (Hierarchy). *The 2D Dyck languages form a strict linear hierarchy: $DW_k \subsetneq DN_k \subsetneq DQ_k \subsetneq DC_k$ and they are not included in the REC family.*

Conclusion By introducing some definitions of 2D Dyck languages we have made the first step towards a new characterization of 2D CF languages by means of a 2D Chomsky-Schützenberger theorem. But the mathematical study of 2D Dyck languages has independent interest, and much remains to be understood, especially for the richer case of Dyck crosswords. Very diverse patterns may occur in DC_k pictures, that currently we are unable to classify. The variety of patterns is related to the length of the circuits and to the number of intersection points in a circuit or between different circuits.



a_i	w_r	b_i
w_c	p	$h_c(w_c)$
c_i	$h_r(w_r)$	d_i

Fig. 1: (Left) An example of picture in DW_1 and (Right) Scheme of nesting accretion.

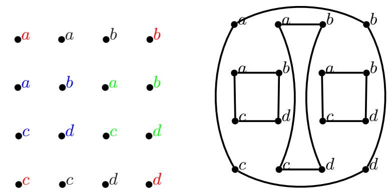


Fig. 2: (Left) A DC_1 picture with 4 quadruples of matching symbols, alternatively (Right) visualized by circuits.

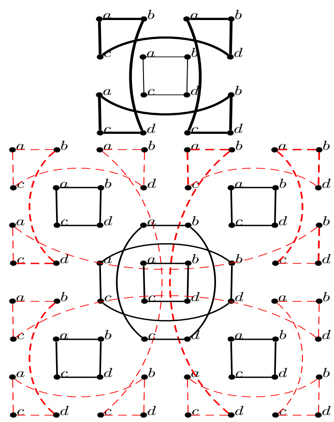


Fig. 3: Two pictures in DC_1. (Left) The picture has two circuits of length 12 and 4. (Right) The picture includes a circuit of length 36 (and 7 rectangular circuits). Its pattern embeds four partial copies (direct or rotated) of the left picture; in the NW copy the "triangle" bdc has been changed to aaa. The transformation can be reiterated to grow a series of pictures.

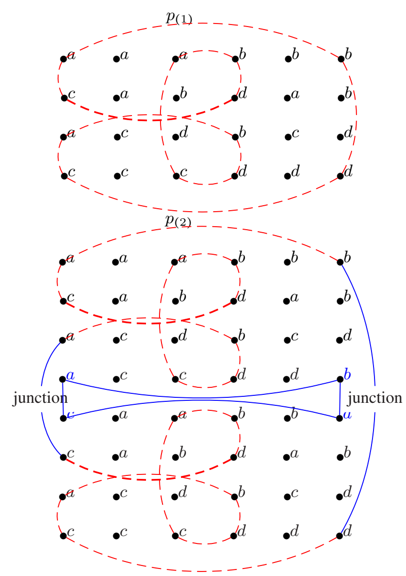


Fig. 4: Two examples of Thm 1. Picture $p_{(1)}$ has a circuit of length $4 + 8 \cdot 1 = 12$, picture $p_{(2)}$ has a circuit of length $4 + 8 \cdot 2$ obtained from $p_{(1)} \ominus p_{(1)}$ by a formal transformation that creates the blue edges.

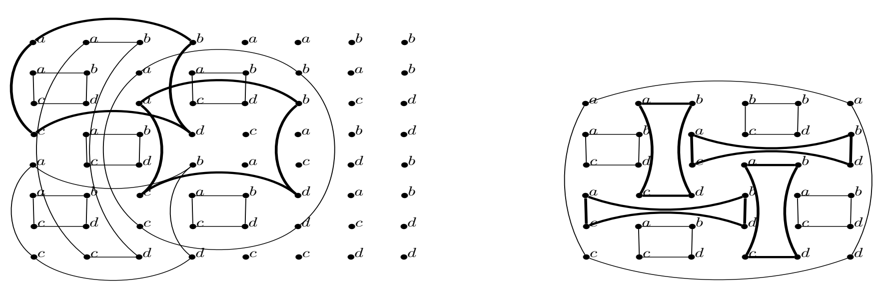


Fig. 5: Two examples of non-neutralizable, quaternate picture.

References

1. J. Berstel and L. Boasson. Towards an algebraic theory of context-free languages. *Fundam. Informaticae*, 25(3):217–239, 1996.
2. S. Crespi Reghizzi, D. Giammarresi, and V. Lonati. Two-dimensional models. In J. Pin, editor, *Handbook of Automata Theory*, pages 303–333. European Mathematical Society Publishing House, 2021.
3. S. Crespi-Reghizzi and M. Pradella. Tile rewriting grammars and picture languages. *Theor. Comput. Sci.*, 340(1):257–272, 2005.
4. F. Drewes. *Grammatical Picture Generation: A Tree-Based Approach*. Springer, 2006.
5. S. A. Fenner, D. Padé, and T. Thierauf. The complexity of regex crosswords. *Inf. Comput.*, 286:104777, 2022.
6. D. Giammarresi and A. Restivo. Two-dimensional languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of formal languages, vol. 3*, pages 215–267. Springer, 1997.
7. O. Matz. Regular expressions and context-free grammars for picture languages. In *14th Annual Symposium on Theoretical Aspects of Computer Science*, volume 1200 of *LNCS*, pages 283–294, 1997.
8. M. Nivat, A. Saoudi, K. G. Subramanian, R. Siromoney, and V. R. Dare. Puzzle grammars and context-free array grammars. *Int. Journ. of Pattern Recognition and Artificial Intelligence*, 5:663–676, 1991.
9. A. Okhotin. Non-erasing variants of the Chomsky—Schützenberger Theorem. In *Proc. of the 16th Intern. Conf. on Developments in Language Theory*, DLT'12, pages 121–129, Berlin, Heidelberg, 2012. Springer-Verlag.
10. D. Průša. *Two-dimensional Languages (PhD Thesis)*. Charles University, Faculty of Mathematics and Physics, Czech Republic, 2004.
11. D. Simplot. A characterization of recognizable picture languages by tilings by finite sets. *Theor. Comput. Sci.*, 218(2):297–323, 1999.
12. R. Siromoney, K. G. Subramanian, V. R. Dare, and D. G. Thomas. Some results on picture languages. *Pattern Recognition*, 32(2):295–304, 1999.