An Efficient Algorithm for the Generation of Z-Convex Polyominoes*

Giusi Castiglione[1] and Paolo Massazza[2]

[1] Università degli Studi di Palermo, Dipartimento di Matematica ed Informatica, Via Archirafi 34, 90123 Palermo, Italy
giusi@math.unipa.it

[2] Università degli Studi dell'Insubria, Dipartimento di Scienze Teoriche e Applicate - Sezione Informatica, Via Mazzini 5, 21100 Varese, Italy
paolo.massazza@uninsubria.it

Abstract. We present a characterization of Z-convex polyominoes in terms of pairs of suitable integer vectors. This lets us design an algorithm which generates all Z-convex polyominoes of size n in constant amortized time.

Keywords: Polyomino, Convex Polyomino, Z-convex polyomino, Tiling, Complexity.

1 Introduction

A polyomino P [17] is a finite connected set of edge-to-edge adjacent square unit cells in the Cartesian two-dimensional plane, defined up to translations. We call size of the polyomino the number of cells that compose it. Several classes of polyominoes have been considered and extensively studied in enumerative or bijective combinatorics [7,15,3,1]. They have been studied in two-dimensional language theory where tiling systems for various classes of polyominoes have been provided [10,9]. In [2,19] the problem of reconstructing polyominoes from partial information has been addressed. Polyominoes have also received a particular attention in tiling theory see [5,25,8,23], for example.

Due to the difficulty of solving problems for the whole class of polyominoes, many subclasses have been studied, in particular by introducing some convexity constraints on the cells. The most studied polyominoes are convex polyominoes, i.e., polyominoes whose rows and columns are connected sets of cells (there are no holes between the lower cell and the upper cell of a column, and also between the leftmost cell and the rightmost cell of a row, respectively). They have been enumerated with respect to the semi-perimeter (one half the length of the boundary) but only asymptotic results are known with respect to the size [6,18]. In [14] the authors observed that convex polyominoes have the property

* Partially supported by Project M.I.U.R. PRIN 2010-2011: Automi e linguaggi formali: aspetti matematici e applicativi.

R.P. Barneva, V.E. Brimkov, and J. Šlapal (Eds.): IWCIA 2014, LNCS 8466, pp. 51–61, 2014.

that every pair of cells is connected by a monotone path (made of steps in only two directions: North and East, North and West, South and East or South and West) inside the polyomino itself. In this way, one can associate with each convex polyomino P the minimal number k of changes of direction which are necessary to ensure the existence of a monotone path between any two cells of P.

More precisely, a convex polyomino is called *k-convex* if, for every pair of its cells, there is at least one monotone path with at most k changes of direction that connects them. Very recently, an asymptotic estimate of a lower bound for the number of k-convex polyominoes with perimeter p, for any k, has been given [24]. Such a hierarchical approach provides a way to handle the general class of convex polyominoes. When the value of k is 1 we have the so-called L-convex polyominoes, where this terminology is motivated by the L-shape of the paths that connect any two cells which are not in the same column or in the same row. L-convex polyominoes have been studied from a combinatorial point of view: in [11,13] the authors enumerate them with respect to the semiperimeter and size; furthermore in [12] a reconstruction algorithm from partial information is given.

Here we are interested in the second level of such a hierarchy, that is, the class of *Z-convex polyominoes* (or, equivalently, *2-convex* polyominoes). They have been enumerated with respect to semiperimeter in [16] and studied in [26] from the tomographic point of view, but many problems are still open. Our main motivation is to give a methodology and a characterization that allow to handle Z-convex polyominoes and at the same time represent an idea that could be generalized to k-convex polyominoes, for each k. In fact, we show a characterization of Z-convex polyominoes of size n in terms of pairs of suitable integer vectors, and we see that such a characterization directly leads to a CAT (constant amortized time) algorithm for the exhaustive generation of Z-convex polyominoes of size n. We recall that CAT algorithms for the exhaustive generation (with respect to the size) of parallelogram polyominoes [20], L-convex polyominoes [21] and for the whole class of convex polyominoes [22], have been recently proposed.

2 Preliminaries

A polyomino P is called *convex* if the intersection of P with an infinite row or column of connected cells is always connected. We indicate by $\text{CPol}(n)$ the set of convex polyominoes of size n. The *size* and the *width* of P are the number $s(P)$ of its cells and the number $w(P)$ of its columns, respectively. A *path* in P is a sequence $c_1, c_2, \ldots, c_k$ of cells of P such that for all i, with $1 \leq i < k$, c_i shares one edge with c_{i+1}.

Three classical families of convex polyominoes are Ferrer diagrams (see Fig. 1(a)), stack polyominoes (see Fig. 1(b)) and parallelogram polyominoes (see Fig. 1(c)). These particular convex polyominoes are characterized with respect to the number of vertices of the minimal bounding rectangle of the polyomino which belong to the polyomino itself: three for Ferrer diagrams, two (adjacent) for stacks polyominoes and two (opposite) for parallelogram polyominoes. A polyomino is *h-centered* [4] if it contains at least one row touching both the left and the right side of the minimal bounding rectangle (see Fig. 1(d)).

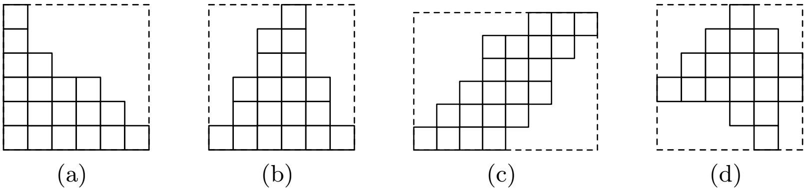

Fig. 1. (a) A Ferrer diagram; (b) a stack polyomino; (c) a parallelogram polyomino and (d) an h-centered convex polyomino

We denote by (i, j) a cell in column i and row j. Without loss of generality, we suppose that for any polyomino P one has $\min\{i|\exists j\ (i,j) \in P\} = 1$ and $\min\{j|(1,j) \in P\} = 0$.

The *vertical projection* of P is the integer vector $\pi(P) = [p_1, \ldots, p_l]$ where $l = w(P)$ and for all i, with $1 \le i \le l$, p_i is the number of cells of column i, $p_i = \sharp\{j|(i,j) \in P\}$. Moreover, the *position vector* of P is defined as the integer vector $\sigma(P) = [s_1, \ldots, s_l]$ where for all i, with $1 \le i \le l$, s_i is the y-coordinate of the bottom cell of column i, $s_i = \min\{j|(i,j) \in P\}$.
Given a class of polyominoes $A \subseteq \mathrm{CPol}(n)$, notice that any $P \in A$ is univocally described by the pair of integer vectors (p, s) where $p = \pi(P)$ and $s = \sigma(P)$.

The following notations are useful when dealing with integer vectors. Let $t = [t_1, \ldots, t_l]$. We denote by $x^{[k]}$ the vector $[\underbrace{x, \ldots, x}_{k}]$, by $\cdot$ the *catenation product* of vectors and we let $t_{<i} = [t_1, \ldots, t_{i-1}]$ $(t_{\le i} = [t_1, \ldots, t_i])$ and $t_{>i} = [t_{i+1}, \ldots, t_l]$ $(t_{\ge i} = [t_i, \ldots, t_l])$.

We say that t is *unimodal* if it can be written as $t = t_{\le i} \cdot t_{>i}$, $1 \le i \le l$, with $t_j \le t_{j+1}$ for each $1 \le j < i$, $t_i > t_{i+1}$ and $t_j \ge t_{j+1}$ for each $i < j < l$. We say that t is *concave* if it can be written as $t = t_{\le i} \cdot t_{>i}$, $1 \le i \le l$, with $t_j \ge t_{j+1}$ for each $1 \le j < i$, $t_i < t_{i+1}$ and $t_j \le t_{j+1}$ for each $i < j < l$.

If P is a convex polyomino one can easily observe that the vector of y-coordinates of the top cells of the columns of P is concave and the vector of y-coordinates of the bottom cells of the columns is unimodal, or vice versa. Note that a constant vector (for example, when P is a stack polyomino) can be, equivalently, considered both unimodal and concave.

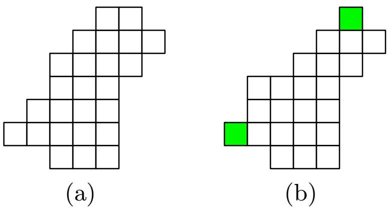

Fig. 2. (a) A Z-convex polyomino, (b) a 3-convex polyomino

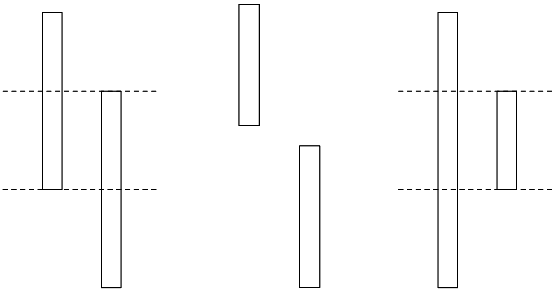

Fig. 3. (a) Columns with a horizontal intersection, (b) horizontally disjoint columns, (c) column horizontally included

3 Z-Convex Polyominoes

A convex polyomino P is said *Z-convex* if any two cells of P are connected by a path in P with at most two changes of direction. Figure 2 shows a Z-convex polyomino together with a polyomino which is 3-convex but not Z-convex (every path between the highlighted cells has at least three changes of direction).

We denote by ZPol(n) the set of Z-convex polyominoes of size n. In this section we give a characterization of Z-convex polyominoes in terms of position vectors and vertical projections and we give some basic properties which are fundamental for the generation of ZPol(n). Firstly, we give a basic definition.

Definition 1. *Let P be a convex polyomino with $\sigma(P) = [s_1, \ldots, s_l]$ and $\pi(P) = [p_1, \ldots, p_l]$. Let $1 \leq i, j \leq l$, we say that*

- *columns i and j of P have a horizontal intersection (see Fig. 3(a)) iff $s_j < s_i \leq s_j + p_j - 1 < s_i + p_i - 1$ or $s_i < s_j \leq s_i + p_i - 1 < s_j + p_j - 1$;*
- *columns i and j of P are horizontally disjoint (see Fig. 3(b)) iff $s_i > s_j + p_j - 1$ or $s_j > s_i + p_i - 1$;*
- *column i horizontally includes column j of P (see Fig. 3(c)) iff $s_i \leq s_j$ and $s_i + p_i - 1 \geq s_j + p_j - 1$.*

The previous definition can be used to characterize Z-convex polyominoes, as stated in the following theorem.

Theorem 1. *Let P be a convex polyomino. Then P is Z-convex if and only if for all i, j, with $1 \leq i < j \leq l$, if columns i and j are horizontally disjoint then there exists k, with $i < k < j$, such that column k horizontally includes both columns i and j.*

Proof. Le P be a Z-convex polyomino and i and j two horizontally disjoint columns of P. Let us consider the top cell $x = (i, s_i + p_i - 1)$ of column i and the bottom cell $y = (j, s_j)$ of column j. If $s_i > s_j + p_j - 1$ (see Fig. 3(b)) the only way to connect x to y by a path with at most two changes of direction is a path that starts with a horizontal right step and changes direction in cells $(k, s_i + p_i - 1)$ and (k, s_j), for a certain k. Then there exists in P a column k with $s_k \leq s_j$ and $p_k \geq p_j + p_i$, and so column k includes both columns i and j. Vice versa, for any pair of cells $x = (i, h)$ and $y = (j, g)$ if columns i and j have a horizontal intersection with $s_i \leq s_j + p_j - 1$ (see Fig. 3(a)) then there exists a row in P, formed by the cells $(i, m), (i + 1, m), \ldots, (j, m)$, with $h \leq m \leq g$, that crosses both columns i and j. Then, there is a path which connects x to y and has exactly two changes of direction, in cells (i, m) and (j, m). Otherwise, if i and j are horizontally disjoint, by hypothesis we can consider the path that starts with an horizontal right step and changes direction in cells (k, h) and (k, g). □

Let $\sigma(P) = [s_1, \ldots, s_l]$ and define $\textsc{top}(P) = [s_1 + p_1 - 1, \ldots, s_l + p_l - 1]$. As observed in the previous section, if P is a convex polyomino then $\sigma(P)$ is concave and $\textsc{top}(P)$ unimodal, or vice versa. Without loss of generality, from here on we consider convex polyominoes of the first kind, i.e. $\sigma(P)$ is concave and $\textsc{top}(P)$ is unimodal; we call *descending* this kind of polyominoes.

Thus, let h be the first index such that $s_{h-1} < s_h$ or column h is horizontally disjoint from one of the first $h - 1$ columns and let k be the smallest index such that column k is not included in column $k + 1$. Trivially, one has $k < h$.

We write $\sigma(P) = s_{\leq h} \cdot s_{>h}$ and $\textsc{top}(P) = t_{\leq k} \cdot t_{>k}$. It follows a decomposition of a Z-convex polyomino as stated in the following proposition.

Proposition 1. *Let P be a Z-convex polyomino with $\sigma(P) = s_{\leq h} \cdot s_{>h}$ and $\textsc{top}(P) = t_{\leq k} \cdot t_{>k}$. Then P can be decomposed into three (possibly null) parts: a* left *stack polyomino L (the first k columns) and a* right *stack polyomino R (the last $w(P) - h + 1$ columns) joined by a central* parallelogram *polyomino C (from column $k + 1$ to column $h - 1$) with the following properties:*

- *C is an h-centered convex polyomino (see Fig. 4(a)).*
- *L and C form an h-centered polyomino (see Fig. 4(b)).*
- *C and R form an h-centered polyomino (see Fig. 4(c)).*
- *if there exist in P two horizontally disjoint columns then they belong to L and to R, respectively (see the highlighted columns in Fig. 4(d)).*

Proof. Consider the fourth property. Let i, j be two horizontally disjoint columns, with $i < j$, and suppose that $s_i > s_j + p_j - 1$. By Theorem 1 there exists a column k, with $i < k < j$, that includes both columns i and j. Moreover, for convexity reasons the y-coordinate of the top cell of a column h to the right of column j must satisfy the relation $\textsc{top}[h] \leq \textsc{top}[j]$. So, columns h and i are horizontally disjoint and then there is g, with $i < g < j < h$, such that column g includes both columns i and h. If one had $s_h < s_j$ then it would be impossible to have $s_g \leq s_h$ due to the convexity constraint. Therefore, any column h to the right of column j is horizontally included in column j, and column j belongs to the right stack R. In the same way one can prove that column i belongs to the left stack L. The others properties are proved as a consequence. □

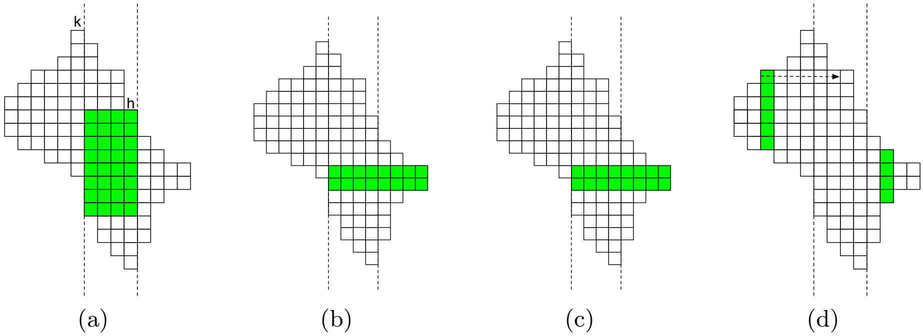

Fig. 4. Properties of the decomposition of a Z-convex polyomino

Let P be a Z-convex polyomino with decomposition $P = L \cdot C \cdot R$ provided by Proposition 1. Note that the left stack L is never null and consists of the first k columns of P, $k \geq 1$. Then we can give the following definition.

Definition 2. *Let $s = \sigma(P) = [s_1, \ldots, s_l]$ and $s_{\leq k} = 0^{[j_1]} \cdot x_2^{[j_2]} \cdots x_s^{[j_s]}$, with $0 > x_2 \cdots > x_s$. We define the integer vector* BOTTOMCELL *as* BOTTOMCELL$[e] = \sum_{i \leq e} j_i$, *with $1 \leq e \leq s$.*

In other words, if $B = \{0, x_2, \ldots, x_s\}$ then the eth entry of BOTTOMCELL is the index of the rightmost column of the left stack L such that the y-coordinate of its bottom cell is the eth value in B.

Furthermore if $C \neq \emptyset$ we can give the following definition.

Definition 3. *Let m be an integer, with $k < m < h$, such that the first m columns of P form an h-centered polyomino with horizontal intersection given by a rectangle of height p_1 (the height of the first column of L). We set* DOM$[m] = e$, *if column m horizontally includes column* BOTTOMCELL$[e]$ *but does not horizontally include column* BOTTOMCELL$[e+1]$. *We define also the integer vector* ISDOM *such that* ISDOM$[e] = m$, *with $1 \leq e \leq s$, if column* DOM$[m] = e$ *In such a case we say that column m* dominates *column* BOTTOMCELL$[e]$.

Example 1 illustrates the vectors BOTTOMCELL and ISDOM associated with the polyomino of Fig. 4. The importance of dominating columns is pointed out by the following lemma.

Lemma 1. *If two columns i, j of a Z-convex polyomino, with $i \leq k$ and $j > h$, are horizontally disjoint then column j is horizontally included in the column that dominates column i.*

Proof. Recall that if columns i and j are horizontally disjoint then there exists a column that includes them. The column m which dominates column i is the rightmost that includes i and so it is the column with the lowest bottom cell among the columns which include column i. Thus m includes also column j. $\square$

Example 1. Let us consider the Z-convex polyomino P in Fig. 4(d). We have

$$\sigma(P) = [0, -1, -1, -2, -2, -3, -6, -8, -9, -10, -8, -5, -4, -4]$$

and

$$\textsc{top}(P) = [2, 3, 4, 4, 5, 7, 6, 4, 4, 1, -1, -2, -3, -3].$$

Furthermore, the left stack has width $k = 6$, with BOTTOMCELL $= [1, 3, 5, 6]$. Moreover, one has ISDOM $= [9, 9, 7, 6]$.

Let us observe that column 12 and column BOTTOMCELL[2] (highlighted in Fig. 4(d)) are horizontally disjoint and that column ISDOM[2] (pointed by the arrow) dominates column BOTTOMCELL[2], i.e. column 9 dominates column 3 and horizontally includes column 12.

4 The Exhaustive Generation of ZPol(n)

In this section we give an outline of an algorithm which works column by column and generates all Z-convex polyominoes of size n. The algorithm is based on an inductive approach: at step i it assumes that a Z-convex polyomino P_{i-1} with $i-1$ columns, vertical projection $[p_1, \ldots, p_{i-1}]$, position vector $[s_1, \ldots, s_{i-1}]$ and size $n-r$ has been generated and it determines all and only those ith columns (of size at most r) that can extend it. More precisely, it computes all the integer pairs (a, b) such that $[p_1, \ldots, p_{i-1}, a]$ and $[s_1, \ldots, s_{i-1}, b]$ denote a Z-convex polyomino. Obviously, by convexity reasons, the values a, b must satisfy the relations

$$b \leq s_{i-1} + p_{i-1} - 1, \quad b + a - 1 \geq s_{i-1} \; .$$

Let us call H-CENTERED(i, r, d) a recursive procedure that, given a Z-convex polyomino P_{i-1} with size $n-r$ and horizontal intersection of height d, generates all Z-convex polyominoes with prefix P_{i-1} by computing all the possible ith columns (identified by the size a and the position b) and making a recursive call for each of them. The crucial points of the algorithm are described below, where all the cases which may arise when adding column i, of size a and position b, are considered (recall the decomposition $P = L \cdot C \cdot R$).

1. Column i horizontally contains all the previous columns. In this case P_i is a prefix of the left stack L and $d = p_1$. So, the procedure updates the vector BOTTOMCELL and makes a recursive call H-CENTERED$(i + 1, r - a, p_1)$;
2. P_{i-1} is a prefix of L, P_i is h-centered $(b + a > 0)$ and column i does not horizontally includes column $i-1$. In this case column i is the first column of C and the vectors DOM and ISDOM are possibly updated before the recursive call H-CENTERED$(i + 1, r - a, \min(d, b + a))$;
3. P_{i-1} is a prefix of L and $b > s_{i-1}$ or $b = s_{i-1}$ and P_i is not h-centered $(b + a < 0)$. In this case C is null and column i is the first column of R. We then proceed by calling a simple procedure R$(i + 1, r - a, a)$ to generate all right stacks of height at most a and size $r - a$;

4. P_{i-1} is not a prefix of L, is h-centered ($d > 0$) and $b + a > 0$. In this case column i belongs either to C (if $b \leq s_{i-1}$ – the vectors DOM and ISDOM are possibly updated) or to R (if $b > s_{i-1}$), and the recursive call H-CENTERED$(i+1, r-a, \min(d, b+a))$ occurs;
5. P_{i-1} is not a prefix of L, is h-centered ($d > 0$) and $b + a \leq 0$. In this case column i is the first column of R (there must be a column $j < i$ including i and any column h horizontally disjoint from i), the vectors DOM and ISDOM are not updated and the recursive call H-CENTERED$(i+1, r-a, 0)$ occurs.
6. P_{i-1} is not h-centered ($d = 0$), that is, at least one column of R has been already generated. In this case we have to generate a suffix of R. The position b of column i has to be determined by exploiting the vectors ISDOM and DOM and the recursive call H-CENTERED$(i+1, r-a, 0)$ occurs.

The previous description provides us with the high level behaviour of the algorithm. Nevertheless, a detailed analysis of the steps associated with Case 2 (the generation of the first column of C) is needed to point out the special role played by the vectors ISDOM and DOM, in particular how they can be updated in time $O(1)$. So, let us see how H-CENTERED$(k+1, r, p_1)$ works when P_k is a (left) stack.

First, we point out that procedure H-CENTERED examines the integer pairs (column-size, bottomcell-position) according to the following order:

$$(a, b) < (a', b') \quad \text{iff} \quad a < a' \vee a = a' \wedge b > b'.$$

This corresponds to two nested loops, the outer associated with the size a (increasing form 1 to r) and the inner associated with the position b (decreasing from $p_k + s_k - a$ to $s_k - a + 1$).

So, the first column that can belong to C corresponds to the pair $(-s_k+1, s_k)$ (a column associated with a pair $(a', b') < (-s_k+1, s_k)$ is necessarily the first column of R). Such a column may possibly dominate only column BOTTOMCELL[1]: this happens if $p_j = 1$ with $j =$ BOTTOMCELL[1], that is, if $1^{[j]}$ is a prefix of $p_{\leq k}$. In this case we set DOM$[k+1] = 1$ and ISDOM$[1] = k+1$. More generally, an inductive approach is used to determine which column is dominated by the $(k+1)$-th column associated with (a, b): it is sufficient to remember the entries $e' =$ DOM$[k+1]$ and $e'' =$ DOM$[k+1]$ associated with the pair (a', b') which immediately precedes (a, b) and with the pair $(a-1, p_k+s_k-a+1)$ (the column of size $a-1$ in the highest position), respectively. Indeed, if $b+a-1 < s_k + b_k$ a simple check let us to see whether to set either DOM$[k+1] = e'$ or DOM$[k+1] = e'+1$ or DOM$[k+1] = e'-1$, whereas if $b+a-1 = s_k+b_k$ one has either DOM$[k+1] = e''$ or DOM$[k+1] = e''+1$ (the vector ISDOM is updated accordingly).

By the same method we can operate in Case 4, when all the subsequent columns j which belong to the parallelogram C, with $j > k+1$, are placed at a position for which P_j is h-centered.

Lastly, we illustrate how the call H-CENTERED(i, r, d) works in Case 5. First, since $b+a \leq 0$, the size a of the first column of R is at most $-s_{i-1}$ (the pairs (a, b) are still generated according to the order $<$ previously defined). By Theorem 1, the highest position for a column i of size a and such that P_i is not h-centered (but still Z-Convex) is the value b which satisfies $b + a = 0$, provided that there is $j < i$ such that column j horizontally includes both column BOTTOMCEL$l[1]$ and column i. This can be checked by accessing column j, where $j =$ ISDOM$[1]$ and testing whether $s_j \leq -a$. If $s_]$ $> -a$ then the position $-a + 1$ (analysed in the previous iteration over b) is the lowest admissible for a column of size a. Thus, $b = -a$ can be considered as the basis of our inductive approach for generating valid positions. More generally, in order to check whether $-a - q$ can be a valid position for a column of size a we need only to remember the index e' such that at the previous iteration (the position $-a - q + 1$) column BOTTOMCELL$[e']$ was the rightmost column horizontally disjoint from column i. If column BOTTOMCELL$[e' + 1]$ is now the rightmost column horizontally disjoint from column i then we read ISDOM$[e' + 1]$ to get the index j and make the test $s_j \leq -a - q$. Otherwise the index $j =$ BOTTOMCELL$[e']$ is used also for testing the current position.

We point out that procedure H-CENTERED works similarly also in Case 6.

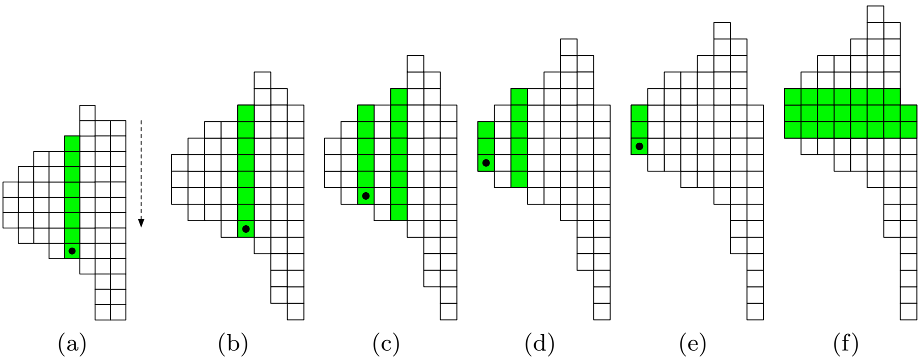

Fig. 5. Some iterations of H-CENTERED$(8, 13, 3)$

Example 2. In Fig. 5 we show the first 6 iterations for b when $a = 13$ of the call H-CENTERED$(8, 13, 3)$. In (a), since $p_8 = p_7$ and $s_8 = s_7$ one has DOM$[8] =$ DOM$[7] = e = 3$ and the current value (7) of ISDOM$[3]$ is replaced by 8. Nothing changes at the 2nd iteration (b). In (c) since TOP$[8] <$ TOP$[$BOTTOMCELL$[e]]$ then $e = e - 1 = 3 - 1 = 2$, DOM$[8] = 2$ and ISDOM$[2] = 8$ and the old value ISDOM$[3] = 7$ is restored. In (d) since TOP$[8] <$ TOP$[$BOTTOMCELL$[e]]$ then $e = e - 1 = 2 - 1 = 1$, DOM$[8] = 1$ and ISDOM$[1] = 8$, Nothing changes at the 5th iteration (e). At the 6th iteration (f) column 8 reduces the horizontal intersection then it does not dominate any column of the left stack L.

5 Complexity

We provide in this section some remarks that let us understand why the proposed algorithm is CAT. The algorithm has a recursive structure. Its execution can be described by a tree where each node at level i corresponds to a call of a procedure which works on column $i+1$ (the root is associated with the call H-CENTERED$(1, n, n)$). Moreover, the outdegree of each internal node is at least 2. This implies that the number of internal nodes is $O(N)$ where N is the number of leaves, that is, the numbers of polyominoes in ZPol(n).

Hence, it follows that the algorithm is CAT if we show that each call (working on a given column, say i) has a running time $O(K)$, where

$$K = \sharp\{(a,b) | P_i \text{ identified by } [p_1, \ldots, p_{i-1}, a], [s_1, \ldots, s_{i-1}, b] \text{ is Z-convex}\}.$$

Indeed, the strategy that procedure H-CENTERED adopts for determining all the admissible pairs (a, b) (representing the size and the position of the column to be generated, respectively) guarantees that.

As shown in the previous section the structure of the procedure consists of two nested cycles, the outer one associated with the size a of the column, the inner one associated with the position b of the bottom cell. For each size a (analysed in increasing order) the admissible positions are generated downwards, with a constant number of operations associated with each value. As soon as a value b' is reached such that the two integer vectors $[p_1, \ldots, p_{i-1}, a]$ and $[s_1, \ldots, s_{i-1}, b']$ do not denote a Z-convex polyomino, a break in the inner loop occurs and a new iteration of the outer loop starts (for the size $a+1$). The key observation is that if b' is not a valid position for the current size then, for any integer $q > 0$, also $b' - q$ is not valid. Note also that if we can not find a valid position for a column of size a, then a valid position for a column of size $a+j$, for all $j > 0$, does not exist. So Procedure H-CENTERED has to return the control to the caller as soon as such a size is reached.

6 Conclusions

In this paper we have shown that, by representing Z-convex polyominoes as pairs of suitable integer vectors, we can easily obtain a CAT algorithm for the exhaustive generation of ZPol(n). It is also straightforward to see that our approach can be exploited for solving the membership problem for ZPol(n). More precisely, having as input a convex polyomino $P \in$ CPol(n) individuated by $\pi(P)$ and $\sigma(P)$, we test whether $P \in$ ZPol(n) in time $O(l)$ where $l = w(P)$. It is also natural to ask whether such an approach can be used to test k-convexity for a generic k, and more generally to design a CAT generation algorithm for k-convex polyominoes. So, a first step in this direction is to find a characterization of k-convex polyominoes in terms of vertical projections and position vectors.

References

1. Barcucci, E., Frosini, A., Rinaldi, S.: On directed-convex polyominoes in a rectangle. Discrete Mathematics 298(1-3), 62–78 (2005)
2. Barcucci, E., Lungo, A.D., Nivat, M., Pinzani, R.: Reconstructing Convex Polyominoes from Horizontal and Vertical Projections. Theor. Comput. Sci. 155(2), 321–347 (1996)

3. Barcucci, E., Lungo, A.D., Pergola, E., Pinzani, R.: ECO: a methodology for the Enumeration of Combinatorial Objects. J. of Diff. Eq. and App. 5, 435–490 (1999)
4. Battaglino, D., Fedou, J.M., Frosini, A., Rinaldi, S.: Encoding Centered Polyominoes by Means of a Regular Language. In: Mauri, G., Leporati, A. (eds.) DLT 2011. LNCS, vol. 6795, pp. 464–465. Springer, Heidelberg (2011)
5. Beauquier, D., Nivat, M.: On Translating One Polyomino to Tile the Plane. Discrete & Computational Geometry 6, 575–592 (1991)
6. Bender, E.A.: Convex n-ominoes. Discrete Math. 8, 219–226 (1974)
7. Bousquet-Mélou, M.: A method for the enumeration of various classes of column-convex polygons. Discrete Math. 154(1-3), 1–25 (1996)
8. Brlek, S., Provençal, X., Fedou, J.-M.: On the tiling by translation problem. Discrete Applied Mathematics 157(3), 464–475 (2009)
9. Brocchi, S., Frosini, A., Pinzani, R., Rinaldi, S.: A tiling system for the class of L-convex polyominoes. Theor. Comput. Sci. 475, 73–81 (2013)
10. Carli, F.D., Frosini, A., Rinaldi, S., Vuillon, L.: On the Tiling System Recognizability of Various Classes of Convex Polyominoes. Ann. Comb. 13, 169–191 (2009)
11. Castiglione, G., Frosini, A., Munarini, E., Restivo, A., Rinaldi, S.: Combinatorial aspects of L-convex polyominoes. Eur. J. Comb. 28(6), 1724–1741 (2007)
12. Castiglione, G., Frosini, A., Restivo, A., Rinaldi, S.: A Tomographical Characterization of L-Convex Polyominoes. In: Andrès, É., Damiand, G., Lienhardt, P. (eds.) DGCI 2005. LNCS, vol. 3429, pp. 115–125. Springer, Heidelberg (2005)
13. Castiglione, G., Frosini, A., Restivo, A., Rinaldi, S.: Enumeration of L-convex polyominoes by rows and columns. Theor. Comput. Sci. 347(1-2), 336–352 (2005)
14. Castiglione, G., Restivo, A.: Reconstruction of L-convex Polyominoes. Electronic Notes in Discrete Mathematics 12, 290–301 (2003)
15. Delest, M.P., Viennot, G.: Algebraic Languages and Polyominoes Enumeration. Theor. Comput. Sci. 34, 169–206 (1984)
16. Duchi, E., Rinaldi, S., Schaeffer, G.: The number of Z-convex polyominoes. Advances in Applied Mathematics 40(1), 54–72 (2008)
17. Golomb, W.S.: Checker Boards and Polyominoes. The American Mathematical Monthly 61, 675–682 (1954)
18. Klarner, D.A., Rivest, R.R.: Asymptotic bounds for the number of convex n-ominoes. Discrete Math. 8, 31–40 (1974)
19. Kuba, A., Balogh, E.: Reconstruction of convex $2D$ discrete sets in polynomial time. Theor. Comput. Sci. 283(1), 223–242 (2002)
20. Mantaci, R., Massazza, P.: From Linear Partitions to Parallelogram Polyominoes. In: Mauri, G., Leporati, A. (eds.) DLT 2011. LNCS, vol. 6795, pp. 350–361. Springer, Heidelberg (2011)
21. Massazza, P.: On the generation of L-convex polyominoes. In: Proc. of GASCom 2012, Bordeaux, June 25-27 (2012)
22. Massazza, P.: On the Generation of Convex Polyominoes. Discrete Applied Mathematics (to appear)
23. Massé, A.B., Garon, A., Labbé, S.: Combinatorial properties of double square tiles. Theor. Comput. Sci. 502, 98–117 (2013)
24. Micheli, A., Rossin, D.: Counting k-Convex Polyominoes. Electr. J. Comb. 20(2) (2013)
25. Ollinger, N.: Tiling the Plane with a Fixed Number of Polyominoes. In: Dediu, A.H., Ionescu, A.M., Martín-Vide, C. (eds.) LATA 2009. LNCS, vol. 5457, pp. 638–647. Springer, Heidelberg (2009)
26. Tawbe, K., Vuillon, L.: 2L-convex polyominoes: Geometrical aspects. Contributions to Discrete Mathematics 6(1) (2011)