Discrete Mathematics and Theoretical Computer Science **AB(DMCS)**, 2003, 103–116

Enumeration of convex polyominoes using the ECO method

A. Del Lungo[1] and E. Duchi[2] and A. Frosini[1] and S. Rinaldi[1]

[1] *Università di Siena, Dipartimento di Scienze Matematiche ed Informatiche, via del Capitano, 15, 53100, Siena, Italy*
[2] *Università di Firenze, Dipartimento di Sistemi e Informatica, via Lombroso, 6/17, 50134, Firenze*

ECO is a method for the enumeration of classes of combinatorial objects based on recursive constructions of such classes. In the first part of this paper we present a construction for the class of convex polyominoes based on the ECO method. Then we translate this construction into a succession rule. The final goal of the paper is to determine the generating function of convex polyominoes according to the semi-perimeter, and it is achieved by applying an idea introduced in [11].

Keywords: convex polyominoes, ECO method, succession rules, kernel method, generating functions.

1 Introduction

A *polyomino* is a finite union of elementary cells of the lattice $Z \times Z$, whose interior is connected. The term *polyomino* is commonly attributable to Golomb [12]. A series of problems is related to these objects such as, for example, the decidability problems concerning the tiling of the plane, or of a rectangle, using polyominoes [3, 8] and, on the other hand, the covering problems of a polyomino by rectangles [6]. The general enumeration problem of polyominoes is difficult to solve and still open.

In order to simplify enumeration problems of polyominoes, several subclasses were defined and studied. In this paper we consider classes of polyominoes that verify particular properties, concerning the geometrical notion of convexity. A polyomino is said to be *column-convex* [*row-convex*] when its intersection with any vertical [horizontal] line is convex. A polyomino is *convex* if it is both column and row convex.
The number f_n of convex polyominoes with semi-perimeter $n+4$ was determined by Delest and Viennot in [9] and it is equal to:

$$(2n+11)4^n - 4(2n+1)\binom{2n}{n}, \tag{1}$$

where the semi-perimeter of a convex polyomino is defined as the sum of the numbers of its rows and columns. The first terms of the sequence are:

$$1, 2, 7, 28, 120, 528, \ldots$$

1365–8050 © 2003 Discrete Mathematics and Theoretical Computer Science (DMTCS), Nancy, France

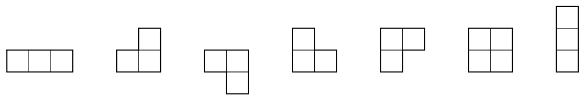

Fig. 1: The seven convex polyominoes with semi-perimeter 4.

sequence M1778 in [14]. Figure 1 depicts the 7 convex polyominoes with semi-perimeter 4.
In [9] the authors obtained (1) after determining the generating function for the class of convex polyominoes,

$$f(x)=\sum_{n\geq 0} f_n x^n = \frac{1-6x+11x^2-4x^3-4x^2\sqrt{1-4x}}{(1-4x)^2}. \tag{2}$$

In the recent years the result in (2) has been re-established using different analytical strategies [4], [7], [13], or bijective proofs [5].

In this paper we present a new proof of (2) which consists in a sequence of simple steps, and it is principally based on the ECO method. ECO is a method for the enumeration and the generation of combinatorial structures introduced by some researchers of the University of Florence (see [2] for an interesting survey). Basically, it consists in determining a recursive construction for the considered class based on *local expansions* performed on the objects themselves: each object is produced exactly once by an object of immediately lower size. In some fortunate case the construction determined by the ECO method can be translated into a formal system called *succession rule*.

Our proof makes use of three steps:

1. We determine an ECO construction ϑ for the set of convex polyominoes.

2. We easily pass from the ECO construction to the associated succession rule Ω. The generating function of Ω is equal to the generating function $f(x)$ for the class of convex polyominoes.

3. Instead of using standard analytic techniques, we apply an idea by Fédou and Garcia proposed in [11] to determine the generating function of some algebraic succession rules. In practice, due to the ECO construction ϑ, each convex polyomino with semi-perimeter $n+2$ can be easily represented as a word of length n of a noncommutative formal power series S_Ω over an infinite alphabet. We determine a recursive decomposition of S_Ω using a set of auxiliary formal power series, and then obtain an algebraic system of equations by taking the commutative image S_Ω. Finally, the solution of this system produces the desired generating function $f(x)$.

We wish to point out that this approach may be used for solving functional equations arising from the application of ECO method to combinatorial structures, and for which the standard methods fail.

2 An ECO operator for the class of convex polyominoes

ECO (Enumerating Combinatorial Objects) [2] is a method for the enumeration and the recursive construction of a class of combinatorial objects, O, by means of an operator ϑ which performs “local expansions” on the objects of O. More precisely, let p be a parameter on O, such that $|O_n| =$

$|\{O \in \mathcal{O} : p(O) = n\}|$ is finite. An operator ϑ on the class $\mathcal{O}$ is a function from $\mathcal{O}_n$ to $2^{\mathcal{O}_{n+1}}$, where $2^{\mathcal{O}_{n+1}}$ is the power set of $\mathcal{O}_{n+1}$.

Proposition 1 *Let ϑ be an operator on $\mathcal{O}$. If ϑ satisfies the following conditions:*

1. *for each $O' \in \mathcal{O}_{n+1}$, there exists $O \in \mathcal{O}_n$ such that $O' \in \vartheta(O)$,*

2. *for each $O, O' \in \mathcal{O}_n$ such that $O \neq O'$, then $\vartheta(O) \cap \vartheta(O') = \emptyset$,*

then the family of sets $\mathcal{F}_{n+1} = \{\vartheta(O) : O \in \mathcal{O}_n\}$ is a partition of $\mathcal{O}_{n+1}$.

This method was successfully applied to the enumeration of various classes of walks, permutations, and polyominoes. We refer to [2] for further details, proofs, and definitions. The recursive construction determined by ϑ can be suitably described through a *generating tree* [2], i.e. a rooted tree whose vertices are objects of $\mathcal{O}$. The objects having the same value of the parameter p lie at the same level, and the sons of an object are the objects it produces through ϑ.

In this section we define an ECO operator describing a recursive construction of the set of convex polyominoes. We first partition the set of convex polyominoes $\mathcal{C}$ into four classes, denoted by $\mathcal{C}_b$, $\mathcal{C}_a$, $\mathcal{C}_r$, and $\mathcal{C}_g$:

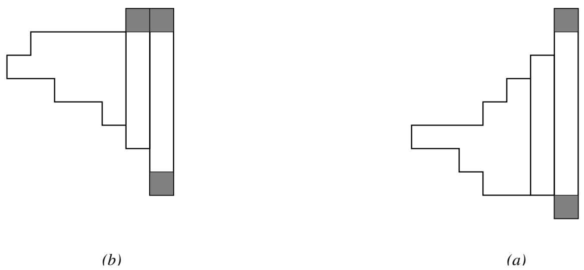

Fig. 2: Convex polyominoes in $\mathcal{C}_b$, (b), $\mathcal{C}_a$, (a).

i) $\mathcal{C}_b$ is the set of convex polyominoes having at least two columns and such that (Fig. 2, (b)):

1. The uppermost cell of the rightmost column has the maximal ordinate among all the cells of the polyomino, and it is the same ordinate as the uppermost cell of the column on its left.
2. The lowest cell of the rightmost column has the minimal ordinate among all the cells of the polyomino.

ii) $\mathcal{C}_a$ is the set of convex polyominoes not in $\mathcal{C}_b$, and such that (see Fig. 2, (a)):

1. The uppermost cell of the rightmost column has the maximal ordinate among all the cells of the polyomino.
2. The lowest cell of the rightmost column has the minimal ordinate among all the cells of the polyomino.

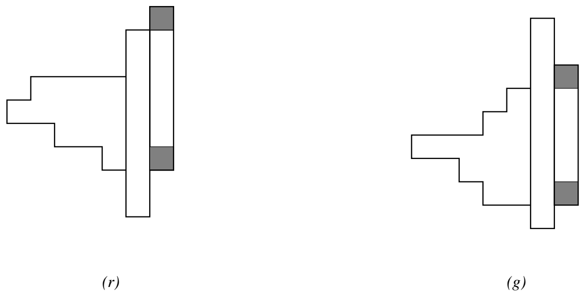

Fig. 3: Convex polyominoes in $\mathcal{C}_r$, (r), and $\mathcal{C}_g$, (g).

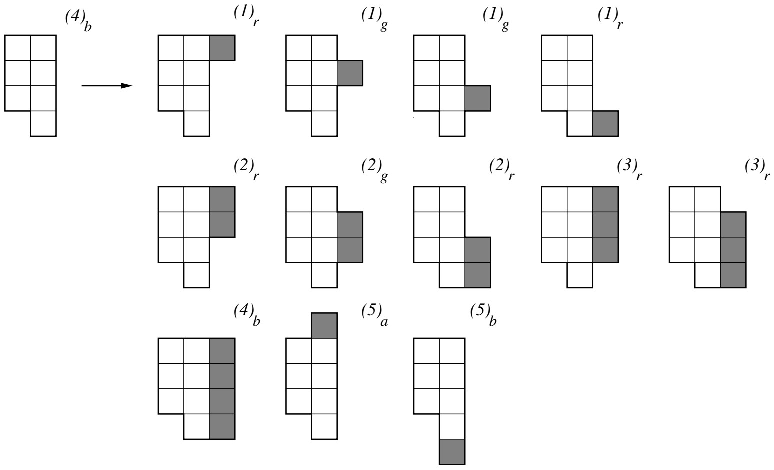

Fig. 4: The ECO operator for a polyomino in the class $\mathcal{C}_b$.

Let us remark that, according to such definition, all convex polyominoes made only of one column lie in the class $\mathcal{C}_a$.

iii) $\mathcal{C}_r$ is the set of convex polyominoes where only one among the lowest and the highest cells have minimal (resp. maximal) ordinate among all the cells of the polyomino (see Fig. 3, (r)).

iv) $\mathcal{C}_g$ is the set of remaining convex polyominoes (see Fig. 3, (g)).

The ECO operator, namely ϑ, performs local expansions on the rightmost column of any polyomino of semi-perimeter $n+2$, producing a set of polyominoes of semi-perimeter $n+3$. More precisely, the operator ϑ performs the following set of expansions on any convex polyomino P, with semi-perimeter $n+2$ and k cells in the rightmost column:

- for any $i = 1, \ldots, k$ the operator ϑ glues a column of length i to the rightmost column of P; this can be done in $k-i+1$ possible ways. Therefore this operation produces $1+2+\ldots+k$ polyominoes with semi-perimeter $n+3$.

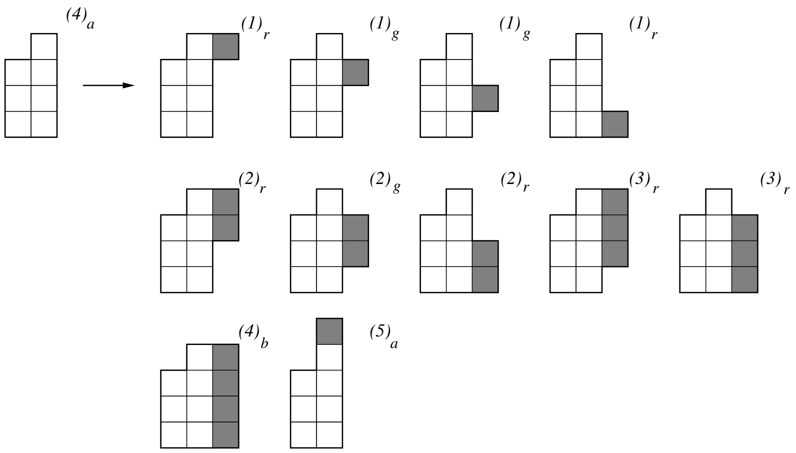

Fig. 5: The ECO operator for a polyomino in the class $\mathcal{C}_a$.

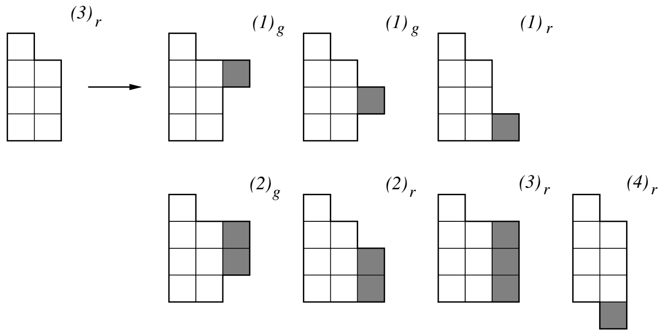

Fig. 6: The ECO operator for a polyomino in the class $\mathcal{C}_r$.

Moreover, the operator performs some other transformations on convex polyominoes, according to the belonging class:

- if $P \in \mathcal{C}_b$, then the operator ϑ produces two more polyominoes, one by gluing a cell onto the top of the rightmost column of P, and another by gluing a cell on the bottom of the rightmost column of P (Fig. 4).

- if $P \in \mathcal{C}_a$, then the operator ϑ produces one more polyomino by gluing a cell onto the top of the rightmost column of P (Fig. 5).

- if $P \in \mathcal{C}_r$, then:

    i) if the uppermost cell of the rightmost column of P has the maximal ordinate, the operator ϑ glues a cell onto the top of that column;

    ii) else, the operator ϑ glues a cell on the bottom of the rightmost column of P (Fig. 6).

The construction for polyominoes in $\mathcal{C}_g$ requires no addictive expansions, and it is graphically explained in Fig. 7. The reader can easily check that the operator ϑ satisfies conditions 1. and 2. of Proposition 1.

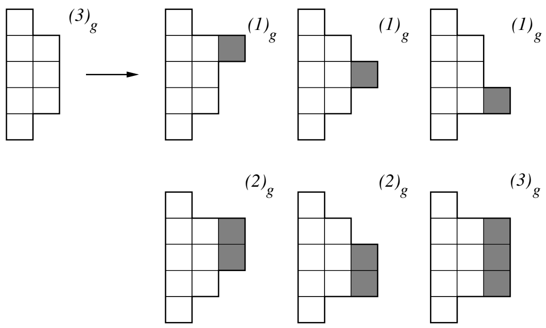

Fig. 7: The ECO operator for a polyomino in the class $\mathcal{C}_g$.

2.1 The succession rule associated with ϑ

The next step consists in translating the previous construction into a set of equations whose solution is the generating function for convex polyominoes. To achieve this purpose, we must introduce the second ingredient of our work, the concept of *succession rule*.

Basically, a polyomino in $\mathcal{C}_i$, $i \in \{a,b,g,r\}$ with k cells in the rightmost column is labelled $(k)_i$. Let us take as an example, the polyomino on the left side of Fig. 6, with label $(3)_r$; according to Fig.6, the performance of the ECO operator on the polyomino can be sketched by the production

$$(3)_r \rightsquigarrow (1)_g\ (1)_g\ (1)_r\ (2)_g\ (2)_r\ (3)_r\ (4)_r.$$

In a similar fashion, the performance of the ECO operator on a polyomino can be sketched by the following set of *productions*:

$$\begin{cases} (k)_g \rightsquigarrow \prod_{j=1}^{k} (j)_g^{k-j+1} \\ (k)_r \rightsquigarrow \prod_{j=1}^{k-1} (j)_g^{k-j}\ \prod_{j=1}^{k+1} (j)_r \\ (k)_a \rightsquigarrow \prod_{j=1}^{k-2} (j)_g^{k-j-1}\ \prod_{j=1}^{k-1} (j)_r^2\ (k)_b\ (k+1)_a \\ (k)_b \rightsquigarrow \prod_{j=1}^{k-2} (j)_g^{k-j-1}\ \prod_{j=1}^{k-1} (j)_r^2\ (k)_b\ (k+1)_a\ (k+1)_b, \end{cases} \tag{3}$$

where k can assume all positive integer values, and the power notation is used to express repetitions, that is $(i)^j$ stands for $(i)(i)\dots(i)$, where (i) is repeated j times. The system constituted by:

1. the label $(1)_a$ (the *axiom*) of the polyomino with semi-perimeter 2;

2. the sets of productions in (3),

forms a *succession rule*. In the sequel we will refer to this succession rule as Ω. Succession rules are commonly known in literature [2, 10], and strictly related to the ECO method.

As an example, for $k = 1, 2, 3$ we have the following productions of Ω:

$$\begin{array}{ll}
(1)_g \rightsquigarrow (1)_g & (1)_r \rightsquigarrow (1)_r\,(2)_r \\
(2)_g \rightsquigarrow (1)_g\,(1)_g\,(2)_g & (2)_r \rightsquigarrow (1)_g\,(1)_r\,(2)_r\,(3)_r \\
(3)_g \rightsquigarrow (1)_g\,(1)_g\,(1)_g\,(2)_g\,(2)_g\,(3)_g & (3)_r \rightsquigarrow (1)_g\,(1)_g\,(1)_r\,(2)_g\,(2)_r\,(3)_r\,(4)_r \\
 & \\
(1)_a \rightsquigarrow (1)_b\,(2)_a & (1)_b \rightsquigarrow (1)_b\,(2)_a\,(2)_b \\
(2)_a \rightsquigarrow (1)_r\,(1)_r\,(2)_b\,(3)_a & (2)_b \rightsquigarrow (1)_r\,(1)_r\,(2)_b\,(3)_a\,(3)_b \\
(3)_a \rightsquigarrow (1)_g\,(1)_r\,(1)_r\,(2)_r\,(2)_r\,(3)_b\,(4)_a & (3)_b \rightsquigarrow (1)_g\,(1)_r\,(1)_r\,(2)_r\,(2)_r\,(3)_b\,(4)_a\,(4)_b.
\end{array}$$

Every succession rule can be graphically represented by means of a *generating tree*, i.e. a rooted labelled tree whose vertices are labelled with the labels of the rule. In practice:

1) the root is labelled with the axiom;

2) each node with label $(k)_t$ produces a set of *sons* whose labels are determined by the production of $(k)_t$ in the succession rule.

Figure 8 (b) depicts the first levels of the generating tree of the rule Ω. A non-decreasing sequence $\{u_n\}_{n\geq 0}$ of positive integers is defined by any succession rule Γ, where u_n is the number of nodes at level n in the generating tree defined by Γ. By convention, the root is at level 0, so $u_0 = 1$. In the general setting of succession rules we also consider the generating function $u_\Gamma(x)$ of the sequence $\{u_n\}_{n\geq 0}$.

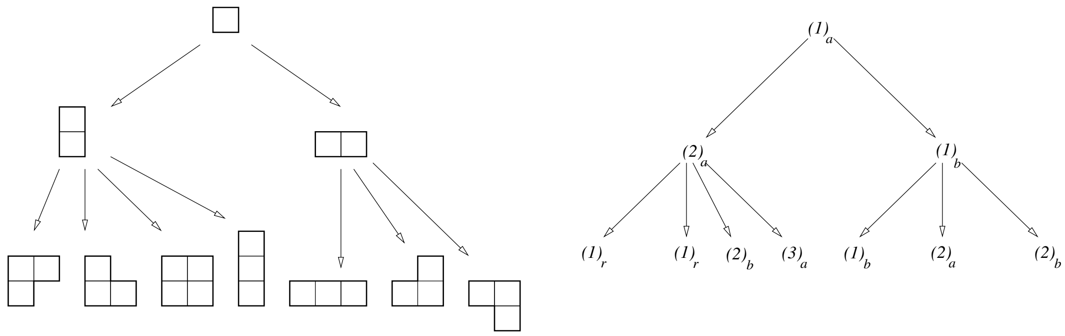

Fig. 8: (a) The first levels of the generating tree of the ECO operator ϑ; (b) the first levels of the generating tree of Ω.

3 Determining the generating function of Ω

In this section, we propose a new approach that allows us to determine the generating function of Ω, i.e. the generating function of convex polyominoes according to the semi-perimeter, and does not use the standard analytical method (the so called *kernel method* [1]).

The basic idea, introduced by Fédou and Garcia in [11], is that of working on succession rules by means of noncommutative formal power series. Each convex polyomino is univocally identified by a node N of the generating tree of the rule Ω, and this node can be encoded by a word in the infinite alphabet $\Sigma = \{(i)_a, (j)_b, (h)_g, (l)_r : i, j, h, l \in \mathbb{N}^+\}$. Such a word is naturally defined by the path of labels of the nodes starting from the root and ending at N. Naturally, due to the form of the productions of the rule Ω, some convex polyominoes have the same word representation. For example the word $(1)_a(2)_a(1)_r$ represents two polyominoes of size 4, as the reader can easily verify in Fig. 8.

Let L_Ω be the set of words, over Σ, beginning by $(1)_a$ and satisfying the productions of Ω. Each word w of L_Ω corresponds to at least one path in the generating tree of Ω. We denote by S_Ω the noncommutative formal power series:

$$S_\Omega = \sum_{w \in L_\Omega} m(w)w,$$

where $m(w)$ is the number of paths corresponding to w in the generating tree of Ω. By construction, the succession rule Ω and the noncommutative formal power series S_Ω have the same generating function,

$$f_\Omega(x) = \sum_{n \geq 0} f_n x^n, \quad \text{where} \quad f_n = \sum_{w \in L_\Omega \, |w|=n} m(w).$$

For example, we have

$$S_\Omega = (1)_a + (1)_a(1)_b + (1)_a(2)_a + (1)_a(1)_b(1)_b + (1)_a(1)_b(2)_a + (1)_a(1)_b(2)_b +$$

$$2 \cdot (1)_a(2)_a(1)_r + (1)_a(2)_a(2)_b + (1)_a(2)_a(3)_a + \dots$$

$$f_\Omega(x) = 1 + 2x + 7x^2 + 28x^3 + 122x^4 + 558x^5 + \dots$$

We work on the series S_Ω using the standard operations on noncommutative formal power series; in particular, for any positive integer n, and $(i)_j \in \Sigma$:

$$nS_\Omega = \sum_{w \in L_\Omega} (nm(w))w, \qquad (i)_j S_\Omega = \sum_{w \in L_\Omega} m(w)(i)_j w.$$

Referring to [11], we introduce the operation $\oplus$: for any word $u = (i_1)_{j_1}(i_2)_{j_2} \dots (i_k)_{j_k}$ of L_Ω, $u^\oplus$ will denote the word $(i_1+1)_{j_1}(i_2+1)_{j_2} \dots (i_k+1)_{j_k}$. For example $((1)_a(2)_a(1)_r)^\oplus = (2)_a(3)_a(2)_r$. Moreover

$$L_\Omega^\oplus = \sum_{w \in L_\Omega} w^\oplus, \quad \text{and} \quad S_\Omega^\oplus = \sum_{w \in L_\Omega} (m(w))w^\oplus.$$

It is a neat consequence that $S_\Omega^\oplus$ and S_Ω have the same generating function.

Generally speaking, a noncommutative formal power series S_Γ can be associated with any succession rule Γ in a completely analogous way.

Catalan succession rule. To fully understand the heart of the matter, we start presenting an example already given in [11]. Let us consider the well-known succession rule defining Catalan numbers [2]:

$$\Gamma : \quad \begin{cases} (1) \\ (k) \rightsquigarrow (1)(2)\dots(k+1), \end{cases} \tag{4}$$

Let $C = S_\Gamma$ be the noncommutative formal power series associated with the words of Γ. In practice:

$$C = (1)+(1)(1)+(1)(2)+(1)(1)(1)+(1)(1)(2)+(1)(2)(1)+(1)(2)(2)+(1)(2)(3)+\dots$$

Easily, we prove that:

$$C = (1)+(1)\,C+(1)\,C^{\oplus}+(1)\,C^{\oplus}\,C. \tag{5}$$

In fact, let w be a term of C. If $|w| \neq 1$, then $w = (1)v$, with $|v| \geq 1$. So we have these possibilities:

1) v begins with (1). Then w is a term of the sum $(1)C$.

2) $v = (2)z$, with $z = (u_1)\dots(u_k)$, and $u_i > 1$, for $i \in \{1,\dots,k\}$. In this case $(2)z$ is a term of $C^{\oplus}$, and then w is a term of $(1)C^{\oplus}$.

3) $v = (2)(u_1)\dots(u_k)w_2$, where $u_i > 1$, for $i \in \{1,\dots,k\}$, and w_2 begins with (1). Then, w is a term of $(1)C^{\oplus}C$.

From (5) we immediately derive a functional equation satisfied by the generating function $C(x)$ of C:

$$C(x) = x + x\,C(x) + x\,C(x) + x\,C^2(x). \tag{6}$$

The basic idea to deal with the succession rule Ω for convex polyominoes is substantially the same, but it requires some more precautions.

A succession rule for central binomial coefficients. In order to ensure a precise comprehension of all the steps of our methodology, hereafter we present a detailed description of the calculus of the generating function for a succession rule more complex than (4).

Let us consider the succession rule Ω' with axiom $(1)_r$, and whose productions are the same as Ω, i.e. those defined in (3). In practice:

$$\Omega' : \quad \begin{cases} (1)_r \\ (k)_g \rightsquigarrow \prod_{j=1}^{k} (j)_g^{k-j+1} \\ (k)_r \rightsquigarrow \prod_{j=1}^{k-1} (j)_g^{k-j} \; \prod_{j=1}^{k+1} (j)_r. \end{cases} \tag{7}$$

In this paragraph our aim is to give a proof that the succession rule Ω' defines central binomial coefficients, $\left\{\binom{2n}{n}\right\}_{n\geq 0}$. We advise the reader that to determine the generating function of Ω', rather than being a mere exercise, will remarkably simplify the complete calculus of $f_\Omega(x)$.

As usual, let us denote $L_{\Omega'}$ the set of the words produced by Ω', and $R = S_{\Omega'} = \sum_{w \in L_{\Omega'}} m(w)w$.

Theorem 3.1 The noncommutative formal power series R can be decomposed into the following sum:

$$(1)_r + (1)_r R + (1)_r R^{\oplus} + (1)_r C^{\oplus} R + (1)_r P^{\oplus} G + (1)_r Q^{\oplus} G, \tag{8}$$

where

$$\begin{aligned}
C = &\quad (1)_r + (1)_r C + (1)_r C^{\oplus} + (1)_r C^{\oplus} C \\
G = &\quad (1)_g + (1)_g G \\
P = &\quad (1)_r + (1)_r P + (1)_r C^{\oplus} P + (1)_r P^{\oplus} + (1)_r C^{\oplus} \\
Q = &\quad (1)_r Q + (1)_r C^{\oplus} Q + 2(1)_r P^{\oplus} G + 2(1)_r Q^{\oplus} G + \\
&\quad (1)_r Q^{\oplus} \; + \; (1)_r (R-C)^{\oplus}.
\end{aligned} \tag{9}$$

Proof. In order to let the reader have a better comprehension of the role of each term of the sum in (8), we depict in Fig. 9 the first levels of the generating tree of Ω'.

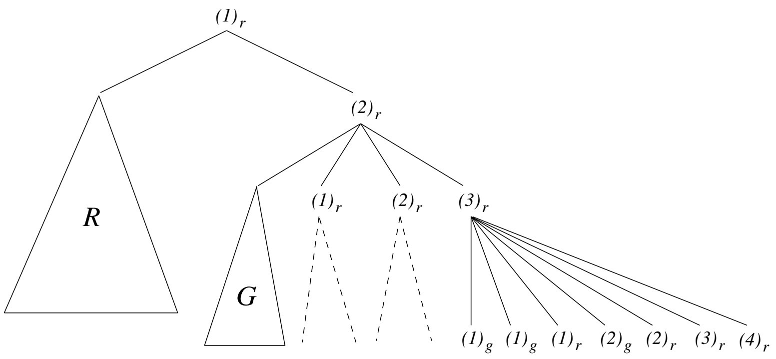

Fig. 9: The first levels of the generating tree of Ω'.

Let w be a term of $L_{\Omega'}$. We have the following cases:

-$|w| = 1$, then $w = (1)_r$.

- $|w| > 1$ then $w = (1)_r v$, and we distinguish the following cases:

1) v begins with $(1)_r$. The set of words in $L_{\Omega'}$ having the form $w = (1)_r v$ is then equal to $(1)_r L_{\Omega'}$. Consequently

$$\sum_{w=(1)_r v} m(w)w = (1)_r R.$$

For example, the word $(1)_r(1)_r(1)_r(2)_r(1)_r(1)_r(2)_r(3)_r(1)_r$ is a term of $(1)_r R$.

2) v begins with $(2)_r$. Four cases are possible:

a) $v \in L^{\oplus}_{\Omega'}$, and

$$\sum_{w=(1)_r v,\, v \in L^{\oplus}_{\Omega'}} m(w)w = (1)_r \sum_{v \in L^{\oplus}_{\Omega'}} m(v)v = (1)_r R^{\oplus}.$$

For example, the word $(1)_r(2)_r(3)_r(4)_r(5)_r(3)_g(3)_g(2)_g$ is a term of $(1)_rR^{\oplus}$.

b) $v=(2)_r(u_2)_r\dots(u_k)_r(1)_rw_2$, with $u_i>1$, and $i\in\{1,\dots k\}$.
The reader can easily verify that the language of such words having the form $(2)_r(u_2)_r\dots$ $\dots(u_k)_r$, $u_i>1$, coincides with $L_\Gamma^{\oplus}$, where L_Γ is the language of Catalan words. Consequently, summing over the set of such words w leads to:

$$\sum_w m(w)w=(1)_r\sum_{v\in L_\Gamma^{\oplus}L_{\Omega'}} m(v)v=(1)_rC^{\oplus}R.$$

The word $(1)_r(2)_r(3)_r(2)_r(1)_r(2)_r(3)_r(2)_g(2)_g(1)_g$ is a term of $(1)_rC^{\oplus}R$.

c) $v=(2)_r(u_2)_r\dots(u_k)_rg_1$, where $u_i>1$, for $i\in\{2,\dots,k\}$, and $g_1\in L_G=\{(1)_g,(1)_g(1)_g,$ $(1)_g(1)_g(1)_g,(1)_g(1)_g(1)_g(1)_g,\dots\}$. Using the considerations in step *b*), the sum over all words w of this type leads to:

$$\sum_w m(w)w=(1)_r\sum_{v\in L_\Gamma^{\oplus}L_G} m(v)v. \tag{10}$$

In this case, for any word v, the value $m(v)$ depends on u_k: more precisely, according to the productions of Ω', if $u_k=j$,

$$m(v)=(j-1)\cdot m((2)_r(u_2)_r\dots(j)_r)\cdot m(g_1). \tag{11}$$

Let us denote by $L_{\Gamma(i)}$ the language of the words of L_Γ ending with a label $(i)_t\in\Sigma$ of any color. Using equations (10),(11) the sum over the words w of such form leads to:

$$\sum_w m(w)w=(1)_r\sum_{j\geq 2}\left((j-1)\cdot\sum_{v\in L^{\oplus}_{\Gamma(j-1)}} m(v)v\right)\sum_{v\in L_G} m(v)v.$$

Let

$$C_{(j-1)}=\sum_{v\in L_{\Gamma(j-1)}} m(v)v \text{ and } G=\sum_{v\in L_G} m(v)v,$$

then (10) becomes

$$\sum_w m(w)w=(1)_r\sum_{j\geq 2}(j-1)C^{\oplus}_{(j-1)}G=(1)_rP^{\oplus}G,$$

where $P=\sum_{j\geq 2}(j-1)C_{(j-1)}$. The word $(1)_r(2)_r(3)_r(4)_r(2)_r(3)_r(1)_g(1)_g(1)_g$ is a term of $(1)_rP^{\oplus}G$.

d) $v=(2)_r(u_2)_{j_2}\dots(u_k)_{j_k}g_1$, with $u_i>1$, for $i\in\{2,\dots,k\}$, $j_k=g$, and $g_1\in L_G$. Considerations analogous to those in step *c*) suggest that the sum over all words w of this type leads to:

$$\sum_w m(w)w=(1)_rQ^{\oplus}G,$$

where

$$Q=\sum_{j\geq 2} j\, R_{(j-1)_g} \text{ and } R_{(j-1)_g}=\sum_{v\in L_{\Omega'(j-1)_g}} m(v)v,$$

$L_{\Omega'(j-1)_g}$ being the set of words belonging to $L_{\Omega'}$ and ending with $(j-1)_g$. The word $(1)_r(2)_r(3)_r(4)_r(2)_r(3)_r(2)_g(2)_g(1)_g(1)_g$ is an example of a term of $(1)_r Q^{\oplus} G$.

It is easy to verify that the decomposition (9) takes into account all the words that satisfy the succession rule Ω'. To conclude the proof we must check that the noncommutative formal power series C, G, P, and Q satisfy the system of equations (9). The statement is obvious for C and G. Below, we will prove that P satisfies:

$$P = (1)_r + (1)_r P + (1)_r C^{\oplus} P + (1)_r P^{\oplus} + (1)_r C^{\oplus}. \tag{12}$$

By definition, $P = \sum_{j\geq 2}(j-1)C_{(j-1)}$. From the equation for C in (9) we deduce that:

$$\begin{aligned} C_{(1)} = &\quad (1)_r + (1)_r\, C_{(1)} + (1)_r\, C^{\oplus}\, C_{(1)} \\ C_{(i)} = &\quad (1)_r\, C_{(i)} + (1)_r\, C^{\oplus}_{(i-1)} + (1)_r\, C^{\oplus}\, C_{(i)} \quad \text{for } i > 1. \end{aligned} \tag{13}$$

Consequently

$$\begin{aligned} P &= C_{(1)} + \textstyle\sum_{j\geq 3}(j-1)C_{(j-1)} \\ &= (1)_r + (1)_r\, C_{(1)} + (1)_r\, C^{\oplus}\, C_{(1)} + (1)_r \textstyle\sum_{j\geq 3}(j-1)\, C_{(j-1)} \\ &\quad +(1)_r \textstyle\sum_{j\geq 3}(j-1)\, C^{\oplus}_{(j-2)} + (1)_r \textstyle\sum_{j\geq 3}(j-1)\, C^{\oplus}\, C_{(j-1)}. \end{aligned}$$

By performing simple algebraic manipulations we obtain (12). A similar proof holds for Q. □

The system (9) leads to a system of functional equations,

$$\begin{aligned} R(x) &= x + x\,R(x) + x\,R(x) + x\,C(x)\,R(x) + x\,P(x)\,G(x) + x\,Q(x)\,G(x) \\ C(x) &= x + x\,C(x) + x\,C(x) + x\,C^2(x) \\ G(x) &= x + x\,G(x) \\ P(x) &= x + x\,P(x) + x\,C(x)\,P(x) + x\,P(x) + x\,C(x) \\ Q(x) &= x\,Q(x) + x\,C(x)\,Q(x) + 2x\,P(x)\,G(x) + 2x\,Q(x)\,G(x)\ + xQ(x)\ +\ x(R(x) - C(x)). \end{aligned} \tag{14}$$

One of the solutions of the system (14) is the generating function of Ω', $R(x) = \frac{x}{\sqrt{1-4x}}$.

The generating function of Ω. Let A be the noncommutative formal power series associated with the succession rule Ω, i.e. $A = S_\Omega = \sum_{w\in L_\Omega} m(w)w$. Using the same strategies as in the previous case we manage to determine a decomposition for the series A, and then translate it into a system of equations. For brevity sake, we prefer not to go deep into details, and only present the two main equations involving A and the final solution.

Let Ω'' be the succession rule having the same productions as Ω and axiom $(1)_b$, and $L_{\Omega''}$ the language associated with Ω. Moreover, let $B = \sum_{w \in L_{\Omega''}} m(w)w$. An approach similar to that used for Ω' leads to the system of equations:

$$\begin{aligned}
A = \quad & (1)_a + (1)_a B + (1)_a A^{\oplus} + 2(1)_a A^{\oplus}_{a,b} R + (1)_a A^{\oplus}_r R + \\
& (1)_a \textstyle\sum_{j\geq 2} (j-1) A^{\oplus}_{(j)_{a,b}} G + (1)_a \sum_{j\geq 2}(j-1) A^{\oplus}_{(j-1)_r} G + \\
& (1)_a \textstyle\sum_{j\geq 2} j A^{\oplus}_{(j-1)_g} G \\
B = \quad & \xi(A) + (1)_b B^{\oplus} + 2(1)_b B^{\oplus}_{a,b} R + (1)_b B^{\oplus}_r R + \\
& (1)_b \textstyle\sum_{j\geq 2}(j-1) B^{\oplus}_{(j)_{a,b}} G + (1)_b \sum_{j\geq 2}(j-1) B^{\oplus}_{(j-1)_r} G + \\
& (1)_b \textstyle\sum_{j\geq 2}(j) B^{\oplus}_{(j-1)_g} G
\end{aligned} \tag{15}$$

where R, G are already known, and $\xi(A)$ is the formal power series obtained from A by replacing, in each term, the first occurrence of $(1)_a$ with $(1)_b$. Moreover:

i) for any label $(i)_t \in \Sigma$, $A_{(i)_t}$ (resp. $B_{(i)_t}$) denotes the formal power series of words in A (resp. B) ending with the label $(i)_t$;

ii) for any $q \in \{a,b,r\}$, A_q (resp. B_q) denotes the formal power series of words in A (resp. B) ending a label of type q, i.e. of the form $(x)_q$.

The solution of the entire system produces the desired function:

$$A(x) = xf(x) = x\sum_{n\geq 0} f_n x^n = \frac{x - 6x^2 + 11x^3 - 4x^4 - 4x^3\sqrt{1-4x}}{(1-4x)^2}. \tag{16}$$

Final remarks. The problem of enumeration of convex polyominoes according to the semi-perimeter is commonly considered a difficult problem. Our approach solves this problem in a sequence of simple steps that finally produce a system of equations satisfied by the desired generating function. This approach can be naturally applied to the enumeration of some other structures (for instance column-convex polyominoes) for which an ECO construction is already known, or can be determined.

References

[1] C. Banderier, M. Bousquet-Mélou, A. Denise, P. Flajolet, D. Gardy, and D. Gouyou-Beauchamps. On generating functions of generating trees, *Discrete Math.* 246(13) (2002) 29-55.

[2] E. Barcucci, A. Del Lungo, E. Pergola, and R. Pinzani ECO: a methodology for the Enumeration of Combinatorial Objects, *Journal of Difference Equations and Applications,* Vol.5 (1999) 435-490.

[3] D. Beauquier and M. Nivat. On translating one polyomino to tile the plane, *Discrete Comput. Geom.*, 6 (1991) 575–592.

[4] M. Bousquet-Mèlou. A method for the enumeration of various classes of column-convex polygons, *Discrete Math.*, 154 (1996) 1-25.

[5] M. Bousquet-Mèlou, and A. J. Guttmann. Enumeration of three dimensional convex polygons, *Annals of Combinatorics*, 1 No.1 (1997) 27-53.

[6] S. Chaiken, D. J. Kleitman, M. Saks, and J. Shearer. Covering regions by rectangles, *SIAM J. Discr. and Alg. Meth.*, 2 (1981) 394–410.

[7] S. J. Chang, and K. Y. Lin. Rigorous results for the number of convex polygons on the square and honeycomb lattices, *J. Phys. A: Math. Gen.*, 21 (1988) 2635-2642.

[8] J. H. Conway, and J. C. Lagarias. Tiling with polyominoes and combinatorial group theory, *Journal of Combinatorial Theory A*, 53 (1990) 183–208.

[9] M. Delest, and X. Viennot. Algebraic languages and polyominoes enumeration, *Theor. Comp. Sci.*, 34 (1984) 169-206.

[10] L. Ferrari, E. Pergola, R. Pinzani, and S. Rinaldi. An algebraic characterization of the set of succession rules, *Theor. Comp. Sci.* 281 (2002) 351-367.

[11] J. M. Fedou, and C. Garcia. Algebraic succession rules, *Proceedings of 14th FPSAC* (2002) n.27.

[12] S. W. Golomb. Checker boards and polyominoes, *Amer. Math. Monthly*, vol. 61, n. 10 (1954) 675-682.

[13] A. J. Guttmann, and I. G. Enting. The number of convex polygons on the square and honeycomb lattices, *J. Phys.* A 21 (1988), L467-L474.

[14] N. J. A. Sloane, and S. Plouffe. *The Encyclopedia of Integer Sequences,* Academic Press, New York (1995).