ON THE EQUIVALENCE PROBLEM FOR SUCCESSION RULES

SREČKO BRLEK*, ENRICA DUCHI, ELISA PERGOLA, AND SIMONE RINALDI

Abstract. The notion of *succession rule* (*system* for short) provides a powerful tool for the enumeration of many classes of combinatorial objects. Often, different systems exist for a given class of combinatorial objects, and a number of problems arise naturally. An important one is the equivalence problem between two different systems. In this paper, we show how to solve this problem in the case of systems having a particular form. More precisely, using a bijective proof, we show that the classical system defining the sequence of Catalan numbers is equivalent to a system obtained by linear combinations of labels of the first one.

Résumé Les *systèmes de réécriture* constituent des outils puissants pour l'énumération de nombreuses classes d'objets combinatoires. Souvent, il existe plusieurs systèmes pour une classe donnée et cela soulève le problème de l'équivalence des systèmes. Dans cet article, nous donnons une solution pour des systèmes ayant une forme particulière. Plus précisément, nous montrons que le système classique définissant les nombres de Catalan est équivalent à une infinité de systèmes obtenus par combinaisons linéaires des étiquettes de celui-ci.

1. Introduction

The notion of *succession rule* was introduced by Chung (et all.) in [6] as a compact notation for *generating trees*, and flourished later as a powerful tool for the enumeration of combinatorial objects (see for instance [1, 3, 8, 16]). More precisely, a *succession rule* Ω is a *system* $((b), \mathcal{R})$, consisting of an axiom and a set of *productions* or *rewriting rules* denoted

$$(1) \qquad \Omega = \begin{cases} (b) & b \in \mathbb{N}^+, \\ (k) \rightsquigarrow (e_1(k))(e_2(k))\dots(e_k(k)), & e_i : \mathbb{N}^+ \to \mathbb{N}^+, k \in M \subseteq \mathbb{N}^+. \end{cases}$$

A system Ω is suitably represented by means of a *generating tree*, a rooted labelled where the root is labelled by the axiom (b), and a node labelled (k) produces k sons labelled by $e_1(k), \dots, e_k(k)$ respectively. Ω defines a non-decreasing sequence of positive integers $\{f_n\}_{n\geq 0}$, the number of nodes at level n (by convention, the root is at level 0), and the generating function of Ω is

$$f_\Omega(x) = \sum_{n\geq 0} f_n x^n \ .$$

The structure of the rewriting rules in a system is closely related to the sequence $\{f_n\}_n$, and this relationship has been studied in [1], for rational, algebraic and transcendental generating functions.

A well-known system is the one defining Schröder numbers [4], $(1, 2, 6, 22, 90, 394, \dots)$, (sequence M2898 in [15]):

$$(2) \qquad \Omega_{\mathcal{S}} = \begin{cases} (2) \rightsquigarrow (3)(3) \\ (k) \rightsquigarrow (3)(4)\dots(k)(k+1)^2, & k \geq 3, \end{cases}$$

* On leave from LaCIM, Montréal, with the support of NSERC (Canada) and G.N.C.S., Istituto Nazionale dell'Alta Matematica, Italy.

where the power $(k+1)^2$ stands for the repetition $(k+1)(k+1)$. Often, the production of the axiom is omitted, when no confusion arises. In Fig.1 the first levels of the generating tree of (2) are shown. We refer to [3] for more details and examples. A system Ω is *finite* if

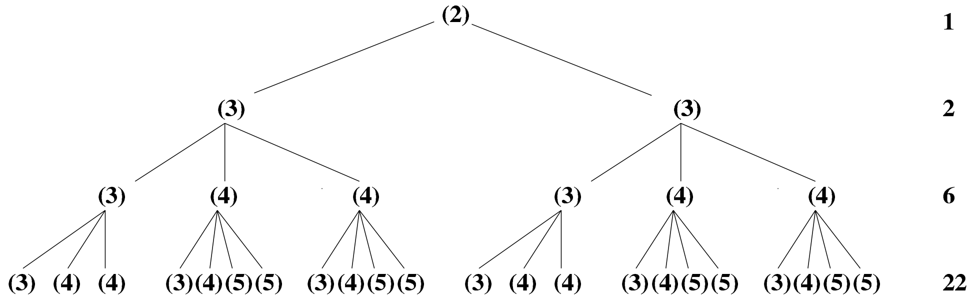

FIGURE 1. The first levels of the generating tree of (2), and its numerical sequence.

the number of labels in the productions is finite, that is, when $|M| < \infty$. In this particular case, the generating function is rational [1], and sometimes has an interpretation as a regular language or other combinatorial structure [8, 12].

A classical example of finite system is the one defining Fibonacci numbers, (M0692 in [15]):

$$\Omega_{\mathcal{F}} = \begin{cases} (1); (1) \rightsquigarrow (2) \\ (2) \rightsquigarrow (1)(2). \end{cases} \tag{3}$$

A succession rule has a *factorial form*, if a finite modification of the set $\{1, 2, \ldots, k\}$ is reachable from k. More formally, a factorial succession rule has the form:

$$\Omega = \begin{cases} (b) \\ (k) \rightsquigarrow (r_0)(r_0+1)\ldots(k-c-1)(k+d_1)(k+d_2)\ldots(k+d_m), & k \geq r_0 \geq 1, \end{cases} \tag{4}$$

with $c \geq 0$, $-c < d_1 \leq d_2 \leq \ldots \leq d_m > 0$, where the consistency principle of succession rules is satisfied imposing that $r_0 + c = m$; the rule in (2) is factorial.

Determining the generating function of a given system is not always an easy task [1]. Therefore, some recent papers focused on the development of some algebraic tools in order to study enumerative properties of succession rules, without computing the corresponding generating functions, by using a linear operator approach [8], or production matrices [7].

The study of these systems has been systematized by the italian school [3] in the so called ECO-method, from which we briefly recall some of the basics. Given a class $\mathcal{O}$ of combinatorial objects, we consider a fixed parameter $p : \mathcal{O} \rightarrow \mathbb{N}$, such that for all $n \in \mathbb{N}, \mathcal{O}_n = p^{-1}(n)$ is finite. If it is possible to define an operator

$$\vartheta : \mathcal{O}_n \longrightarrow 2^{\mathcal{O}_{n+1}},$$

performing "local expansions" on objects of size n (i.e. $\mathcal{O}_n = \{O \in \mathcal{O} : p(O) = n\}$), such that

(i) *for each $O' \in \mathcal{O}_{n+1}$, there exists $O \in \mathcal{O}_n$ such that $O' \in \vartheta(O)$,*
(ii) *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 $\{\vartheta(O) : O \in \mathcal{O}_n\}$ is a partition of $\mathcal{O}_{n+1}$.
We refer to [3] for further details, proofs, definitions and examples. The parameter p being fixed, the recursive construction determined by ϑ is described by a *generating tree* [6], whose vertices are objects of $\mathcal{O}$; the objects having the same parameter value lie on the same level,

and the siblings of an object are the objects produced by ϑ: if $|\vartheta(P)| = k$ then the object P blossoms often according to a *system* (of the form (1)).

2. The equivalence problem

Two rules Ω_1 and Ω_2 are said to be *equivalent*, if they define the same number sequence,

$$\Omega_1 \cong \Omega_2 \iff f_{\Omega_1}(x) = f_{\Omega_2}(x) \ .$$

For instance, the reader can easily verify that the following rules are equivalent to (2), and define the Schröder numbers [4, 5]:

$$\Omega'_S = \begin{cases} (2) \\ (2k) \rightsquigarrow (2)(4)^2 \dots (2k)^2(2k+2), \end{cases}$$

$$\Omega''_S = \begin{cases} (2) \rightsquigarrow (3)(3) \\ (2k-1) \rightsquigarrow (3)^2(5)^2 \dots (2k-1)^2(2k+1), \end{cases}$$

$$\Omega'''_S = \begin{cases} (2) \\ (2^k) \rightsquigarrow (2)^{2^{k-1}}(4)^{2^{k-2}}(8)^{2^{k-3}} \dots (2^{k-1})^2(2^k)(2^{k+1}). \end{cases}$$

The *equivalence problem* consists in determining if two different systems are equivalent. In general, as mentioned recently by M. Robson [11], this problem is not decidable. However, there are classes of systems for which the answer is positive. The easy case of finite systems stems out from formal language theory. Indeed, A PD0L system is a triple [14]:

$$G = (\Sigma, h, w_0),$$

where $\Sigma = \{a_1, \dots, a_k\}$ is a k-letter alphabet, h is an endomorphism defined on the set Σ^+ of non-empty words, and $w_0 \in \Sigma^+$ is called the *axiom*. The length of a word $w \in \Sigma$ is denoted $|w|$. The *language* of G is defined by:

$$L(G) = \{h^i(w_0) : i \geq 0\}.$$

The function $f_G(n) = |h^n(w_0)|$, $n \geq 0$ is the *growth function* of G, and the sequence $|h^n(w_0)|$, $n \geq 0$ is its *growth sequence*. A *growth matrix* M associated to G is defined by,

$$M_G[i, j] = |h(a_j)|_{a_i},$$

where $|h(a_j)|_{a_i}$ is the number of occurrences of the letter a_i in $h(a_j)$. The growth sequence is then obtained by the generating function

$$f_G(x) = \frac{[10^{k-1}] \cdot \chi(M) \cdot (I - Mx)^{-1} \cdot [1^k]^t}{x^k \cdot \chi(M)} \tag{5}$$

where $\chi(M)$ is the characteristic polynomial of M, $[1^k]$ is the k-length vector with all entries 1, and $[10^{k-1}]$ has all entries 0 except the first which is 1. (see [13] for details).

Remark that any finite system Ω can be viewed as a particular PD0L system where the alphabet Σ is the set of labels of Ω, and h is defined by the productions of Ω, and $w_0 \in \Sigma$.

For instance, the rule (3) defines a PD0L system F, where $\Sigma = \{1, 2\}$, $w_0 = 1$, and

$$\begin{array}{rcl} h(1) & = & 2 \\ h(2) & = & 12 \end{array} \quad ; \quad M = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}.$$

The words in the language of F are $1, 2, 12, 212, 12212, 21212212, 1221221212212, \dots$, and its growth sequence is obtained from the matrix associated to G by computing the generating function (5)

$$f_G(x) = (1 + x + 2\,x^2 + 3\,x^3 + 5\,x^4 + 8\,x^5 + 13\,x^6 + 21\,x^7 + 34\,x^8 + 55\,x^9 + O\left(x^{10}\right)).$$

	1	2	3	4	5	6	7	...
1	1							...
2	1	1						...
3	2	2	1					...
4	5	5	3	1				...
5	14	14	9	4	1			...
6	42	42	28	14	5	1		...
7	132	132	90	48	20	6	1	...
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱

TABLE 1. The Catalan triangle.

Now, two D0L systems are *growth equivalent* if they have the same generating function, which amounts to check if two polynomials are equal, and, consequently, the equivalence problem is decidable for the class of finite systems. However, the computation of the generating functions can be avoided, by checking the equality of the first few terms of the two sequences as stated below.

Theorem 1. *The equivalence problem is decidable for the class of finite systems.*

Proof. Let Ω_1 and Ω_2 be two finite succession rules having k_1 and k_2 labels respectively. In view of Theorem 3.3 [14] it is necessary and sufficient to check if the first $k_1 + k_2$ terms of the two sequences defined by Ω_1 and Ω_2 coincide. □

For example, the finite rules:

$$\Omega_1 = \begin{cases} (2) \\ (2) \rightsquigarrow (2)(3) \\ (3) \rightsquigarrow (2)(3)(3), \end{cases} \qquad \Omega_2 = \begin{cases} (2) \\ (1) \rightsquigarrow (2) \\ (2) \rightsquigarrow (1)(4) \\ (4) \rightsquigarrow (1)(2)(4)(4), \end{cases} \tag{6}$$

both define odd index Fibonacci numbers, $1, 2, 5, 13, 34, 89, \ldots$ (sequence M1439 in [15]). Their equivalence can be verified by comparing the first 5 terms of the defined sequences.

In [1] the authors formalize and then apply a method, called the *kernel* method, in order to find a solution to the functional equation arising from a factorial system(4). The main result states that a factorial system has an algebraic generating function.

Theorem 2. *The equivalence problem is decidable for the rules having a factorial form.*

Proof. A classical result on the equality of algebraic generating functions in several commutative variables, shows that the equality is decidable (see [14] Theorem IV. 5.1) □

3. An infinite set of rules for the Ballot numbers

For $k, n \in \mathbb{N}$, let $a_{n,k}$, be the set of Ballot numbers, defined by the recurrence,

$$\begin{aligned} &a_{1,1} = 1, \\ &a_{n+1,1} = \textstyle\sum_{j\geq 1} a_{n,j}, \\ &a_{n+1,k} = \textstyle\sum_{j\geq k-1} a_{n,j}, \qquad k \geq 2. \end{aligned}$$

They can conveniently be displayed in tabular form, as below, in a triangular array, sometimes known as the *Catalan triangle* shown in Table 1. For any positive integer h, a rule defining the sequence in the h-th column is known in literature:

$$\Omega^h = \begin{cases} (h) \\ (k) \rightsquigarrow (2)(3)\dots(k)(k+1). \end{cases} \tag{7}$$

Remark that for $h = 1$, we have the rules defining the Catalan numbers.
Let $h, \alpha \in \mathbb{N}^+$, and $\beta \in \mathbb{N}$. We first define the following rule:
(8)

$$\Omega^h_{\alpha,\beta} = \begin{cases} (h) \\ (1) \rightsquigarrow (2) \\ (2) \rightsquigarrow (2)(3) \\ \dots \qquad \dots \\ (\alpha+\beta-1) \rightsquigarrow (2)(3)\dots(\alpha+\beta) \\ (\alpha k+\beta) \rightsquigarrow (1)^k \dots (\alpha-1)^k(\alpha+1)\dots(\alpha+\beta)(2\alpha+\beta)\dots((k+1)\alpha+\beta), \ k \geq 1. \end{cases}$$

In the sequel we prove that, for $h \leq \alpha + \beta$, the system (8) is equivalent to the system (7), so the first can be viewed as a generalization of the second, where the labels have been linearly combined according to the positive coefficients α and β. Moreover, the first $\alpha+\beta$ levels of the two generating trees coincide. As a consequence, we obtain that (8) defines the numbers $\{a_{n,h} : n \geq 0\}$, for any α and β such that $h \leq \alpha+\beta$. In particular, for $h = 1$ we have an infinite set of succession rules defining Catalan numbers:
(9)

$$\Omega^1_{\alpha,\beta} = \begin{cases} (1) \\ (1) \rightsquigarrow (2) \\ (2) \rightsquigarrow (2)(3) \\ \dots \qquad \dots \\ (\alpha+\beta-1) \rightsquigarrow (2)(3)\dots(\alpha+\beta) \\ (\alpha k+\beta) \rightsquigarrow (1)^k \dots (\alpha-1)^k(\alpha+1)\dots(\alpha+\beta)(2\alpha+\beta)\dots((k+1)\alpha+\beta), \ k \geq 1. \end{cases}$$

Instead of using generating functions as in [1], we provide a bijective proof by the application of the ECO method.

3.1. **Dyck paths.** We consider lattice paths in the plane $\mathbb{Z} \times \mathbb{Z}$, starting from the origin $(0, 0)$, and using *rise steps* $\mathbf{x} = (1, 1)$ and *fall steps* $\overline{\mathbf{x}} = (1, -1)$. The set $\mathcal{D}$ of Dyck paths is the subset of $\Sigma^* = \{\mathbf{x}, \overline{\mathbf{x}}\}^*$ generated by the grammar

$$D := \epsilon + \mathbf{x}D\overline{\mathbf{x}}D, \tag{10}$$

and we refer to paths as words, in which the notions of prefix, suffix have the usual meaning.

A *Dyck path* of length $2n$ is a sequence remaining weakly above the x-axis. The *height* of a point $P = (P_x, P_y)$ is defined by $h(P) = P_y$. A Dyck path is said *elevated* or *primitive* if it can be written as $D = \mathbf{x}D'\overline{\mathbf{x}}$ with $D' \in \mathcal{D}$, and we denote the stripping operation by

$$D' = \mathrm{Top}(D).$$

Given two points P', P of a Dyck path D, the *factor* starting at P' and ending at P of the corresponding Dyck word is denoted $D[P'_x, P_x]$. By convention, $D[i, j] = \epsilon$ if $i \geq j$. The *insertion* of a word w in D at position i is defined by

$$\mathrm{insert}(D, w, i) = D[0, i] \cdot w \cdot D[i + 1, 2n] \ .$$

The last sequence of fall steps $\ell_d(D)$, or *last descent*, of D satisfies $\ell_d(D) = \overline{\mathbf{x}}^k$ for some $k \geq 1$, and $\mathbf{P}(D)$ is the set of its points. Finally, $|D|$ denotes the length of the word (number of steps). From the grammar (10), one can easily deduce the properties summarized in the next statement.

Proposition 1. *Every non empty Dyck* $D = u\mathbf{x}\overline{\mathbf{x}}^k, k \geq 1$, *satisfies the conditions*
(a) $D = D_1 D_2 \ldots D_m$ *where* $\forall i, D_i$ *is primitive;*
(b) $\exists D'' = u\mathbf{x}^{k-1} \in \mathcal{D}$ *such that* $D = \mathrm{insert}(D'', \mathbf{x}\overline{\mathbf{x}}, |D''| - (k-1))$*;*
(c) $\forall$ *suffix* v *of* D, $\exists D'' \in \mathcal{D}, \exists u' \in \Sigma^*$, *such that* $u'v \in \mathcal{D}$ *and* $D = \mathrm{insert}(u'v, D'', |u'|) = u'D''v$.

We provide now a valuation on the set of Dyck paths,

$$\mathrm{Val} : \mathcal{D} \longrightarrow \{0, 1, 2\} \times \mathbb{N},$$

where the first term yields a partition

$$\mathcal{D} = \mathcal{D}^0 \cup \mathcal{D}^1 \cup \mathcal{D}^2, \quad \text{with } \mathcal{D}^i \cap \mathcal{D}^j = \emptyset, \ \forall i \neq j \ .$$

A valuation for Dyck paths. For $l, i, n \in \mathbb{N}$, a *Dyck path at level* i is the image of an ordinary Dyck path (at level 0) under the translation $(0,0) \mapsto (l, i)$, running from (l, i) to $(l + 2n, i)$ and above the line $y = i$.

Let $\mathcal{D}(i)$ be the set of Dyck paths at level i, and $\mathbb{D} = \{\bigcup \mathcal{D}(i) : i \in \mathbb{N}\}$. By Proposition 1 (a), each path $D(i) \in \mathcal{D}(i)$ admits a unique decomposition in terms of *primitive* paths at level i,

$$D(i) = D(i,1)D(i,2)\ldots D(i,m),$$

where $D(i,j)$ is the j-th *component*, and $\#(D(i)) = m$ is the number of components. In

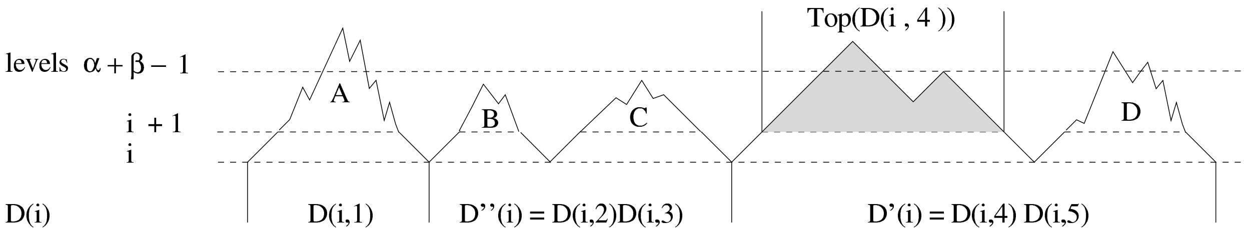

FIGURE 2. Decomposition of a Dyck path at level i and notations.

this decomposition, $D''(i)$ denotes the rightmost factor having height less than $\alpha + \beta - i - 1$, while $D'(i)$ is the factor on the right of $D''(i)$.

The valuation is defined by $\mathrm{Val}(D) = \overline{\mathrm{Val}}(D, 0)$, where $\overline{\mathrm{Val}}(D, i)$ is defined by the following algorithm, where the variables are defined by the notations above.

Algorithm $\overline{\mathrm{Val}}(D, i)$**:** ;
$D - D''$ is the path obtained by removing D'' from D; $\overline{\mathrm{Val}}(D, i)[1]$ refers to the first component of the valuation.
if $D = \epsilon$ **then:** return $(0, i)$
elseif $\#(D') = 0$ **then :** there are three cases
  if $\#(D'') > 1$ **then:** return $(0, i)$
  elseif: $\overline{\mathrm{Val}}(D - D'', i)[1] = 1$ **then** return $(2, i)$
  else: return $(0, i)$
else: let $D' = \prod_{j=1}^{m} D'(j)$; we have four cases:
  if: $\overline{\mathrm{Val}}(\prod_{j=1}^{m-1} D'(j), i)[1] = 1$ **then** return $(2, i)$
  elseif $i < \alpha - 2$ **then:** return $\overline{\mathrm{Val}}(\mathrm{Top}(D'_m), i + 1))$
  elseif $|\mathbf{P}(D'_m)| < \alpha + \beta - i$ **then:** return $(0, i)$
  else : return $(1, i)$.
end $\overline{\mathrm{Val}}(D, i)$**.:**

3.2. **An ECO-system for $\mathcal{D}$.** We define now an ECO-system for the generation of Dyck paths, according to the rule $\Omega^1_{\alpha,\beta}$, that classifies the paths according to the valuation Val.

The ECO operator $\vartheta : \mathcal{D}_n \to 2^{D_{n+1}}$, is defined inductively by setting $\epsilon \in \mathcal{D}^0$, and each path produced is classified in some $\mathcal{D}^i$, with an extra labelling of those in $\mathcal{D}^2$. So, in case of

$[D \in \mathcal{D}^0]$**:** /*See Figure 3 for an example with $\alpha = 3, \beta = 2$, and $|\mathbf{P}(D)| = 4$.*/
- for each point $P \in \mathbf{P}(D)$ do
  $D_{h(P)} \leftarrow \text{insert}(D, \mathbf{x}\overline{\mathbf{x}}, P_x)$;
  if $h(P) < \alpha + \beta - 2$ then $D_{h(P)} \in \mathcal{D}^0$ else $D_{h(P)} \in \mathcal{D}^1$. /* classifying */

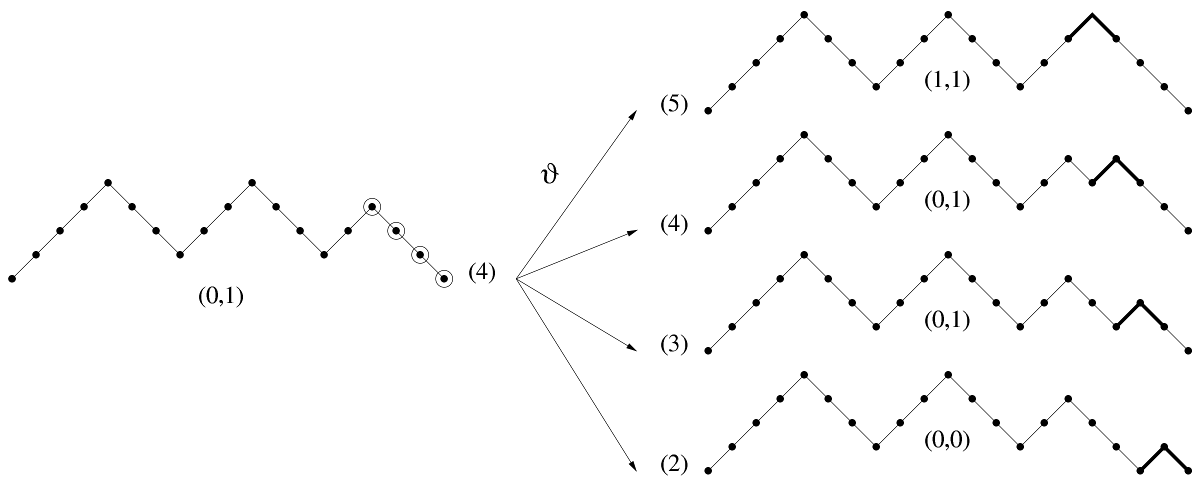

FIGURE 3. The operator ϑ applied to a path in $\mathcal{D}^0$.

Remark. For each path in the class $\mathcal{D}^0$ we have $D \in \mathcal{D}^0 \Longrightarrow h(\ell_d(D)) < \alpha + \beta - 1$.

$[D \in \mathcal{D}^1]$**:** /* See Figure 4 for an example with $\alpha = 3, \beta = 2$.*/
- for each point $P \in \mathbf{P}(D)$ such that $h(P) \geq \alpha + \beta - 1$ do
  $D_{h(P)} \leftarrow \text{insert}(D, \mathbf{x\overline{x}}, P_x)$; $D_{h(P)} \in \mathcal{D}^1$;
  let P' be the leftmost point of D such that $P'P \in \mathcal{D}$;
  /* then $D = uD[P'_x, P_x]v$ with $uv \in \mathcal{D}$; this decomposition exists by Proposition 1(c) */
  for each point $Q \in \mathbf{P}(D)$ such that $0 \leq h(Q) \leq \alpha - 2$ do
  $D_{h(P),h(Q)} \leftarrow \text{insert}(uv, \mathbf{x}D[P'_x, P_x]\overline{\mathbf{x}}, Q_x)$; $D_{h(P),h(Q)} \in \mathcal{D}^2$;
  $\text{label}(D_{h(P),h(Q)}) \leftarrow h(Q)$; /* labelling*/
- for each point $P \in \mathbf{P}(D)$ such that $\alpha - 1 \leq h(P) \leq \alpha + \beta - 2$ do
  $D_{h(P)} \leftarrow \text{insert}(D, \mathbf{x\overline{x}}, P_x)$;
  if $h(P) = \alpha + \beta - 2$ then $D_{h(P)} \in \mathcal{D}^1$ else $D_{h(P)} \in \mathcal{D}^0$.

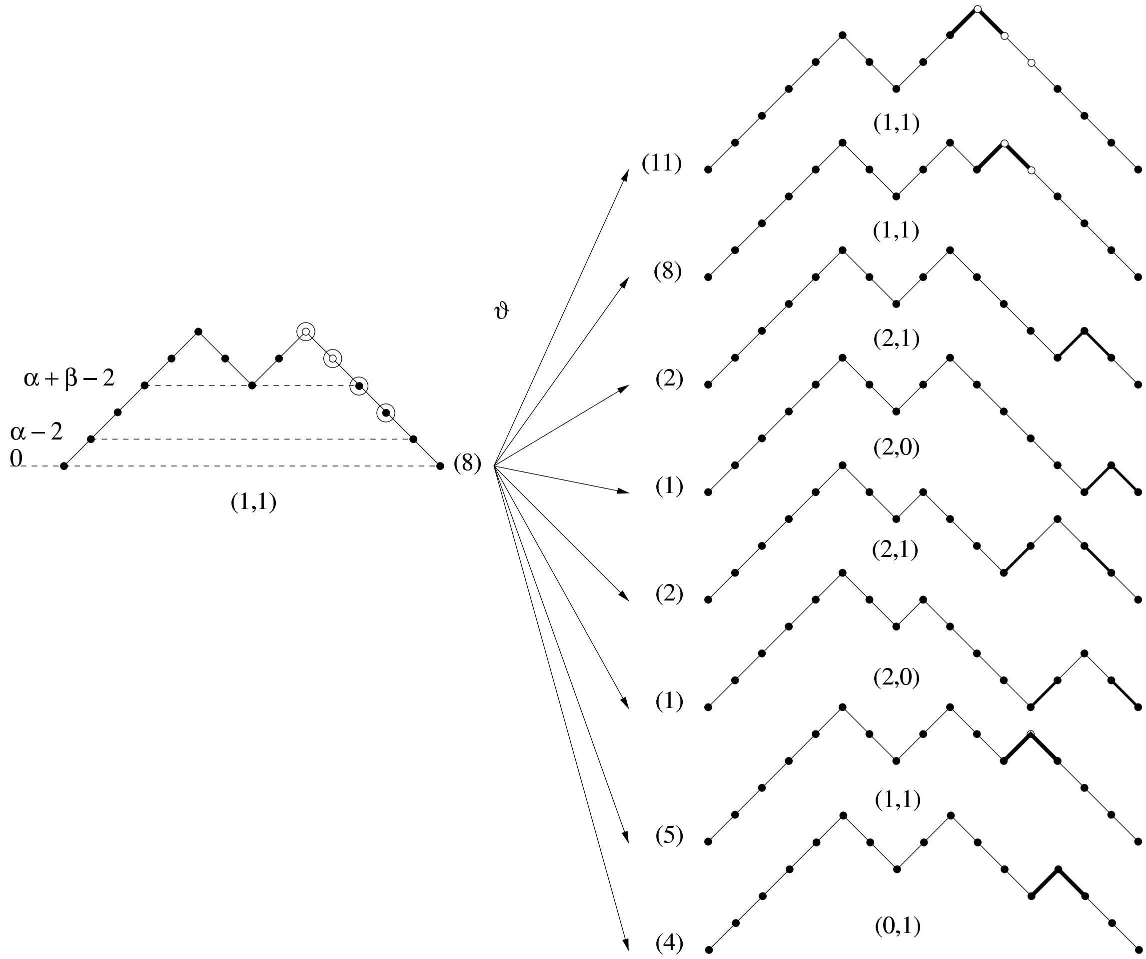

FIGURE 4. The operator ϑ applied to a path in $\mathcal{D}^1$.

Remark. For algorithmic efficiency, the statements of the form $D_{h(P)} \in \mathcal{D}^i$ should be viewed as an assignment of a label, which avoids a call of the valuation function. Moreover, observe that only the paths in $\mathcal{D}^2$ receive a label which is equal to the height of the insertion point.

$[D \in \mathcal{D}^2]$**:** See Figure 5 for an example with $\alpha = 3, \beta = 0$. These paths are labelled.

- for each point $P \in \mathbf{P}(D)$ such that $h(P) \leq \text{label}(D)$ do
  $D_{h(P)} \leftarrow \text{insert}(D, \mathbf{x}\overline{\mathbf{x}}, P_x)$;
  if $\text{label}(D) < \alpha + \beta - 2$ then $D_{h(P)} \in \mathcal{D}^0$ else $D_{h(P)} \in \mathcal{D}^1$.

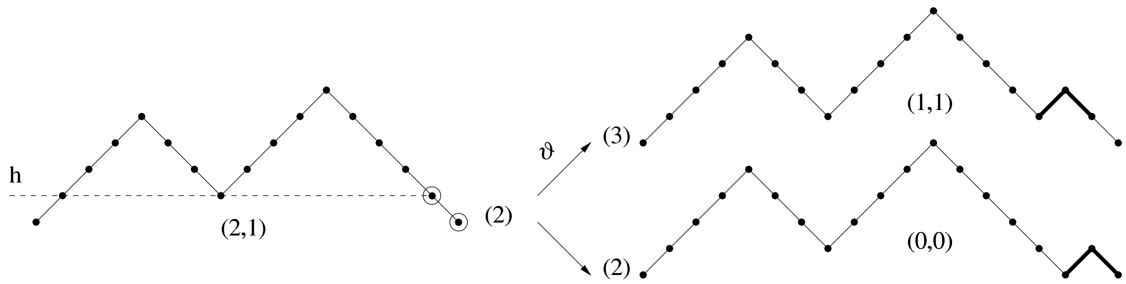

FIGURE 5. The operator ϑ applied to a path in $\mathcal{D}^2$.

It remains now to prove that the described construction generates all the Dyck paths (i) and that we have a partition (ii). This is achieved for both conditions by induction, that follows the inductive definition of ϑ.

(i) Let $D' \in \mathcal{D}_{n+1}$; then there exists $D \in \mathcal{D}_n$, such that $D' \in \vartheta(D)$:
- if $D' \in \mathcal{D}^0 \cup \mathcal{D}^1$, then $D' = u\mathbf{x}\overline{\mathbf{x}}^k$ and Proposition 1(b) identifies the last peak of $D^{'}$ to be removed, i.e.
$$D = u\overline{\mathbf{x}}^{k-1} \in \mathcal{D}_n \ ,$$
such that $D' = \text{insert}(D, \mathbf{x}\overline{\mathbf{x}}, |D| - (k-1))$.
- if $D' \in \mathcal{D}^2$, from the valuation 3.1, we have
$$\text{Val}(D') = (2, |v|), \quad \text{and} \quad D' = uD''v \ ,$$
where $D'' \neq \epsilon$, and $\#(D'') \geq 2$. Then, Proposition 1(a) provides the factorization
$$D'' = D''_1 \ldots D''_{m-1} D''_m \ ,$$
where $D''_m = \mathbf{x}\text{Top}(D''_m)\overline{\mathbf{x}}$. Let P_x be the position of the last peak of D''_{m-1}. Then,
$$D = \text{insert}(uD''_1 \ldots D''_{m-1}v, \text{Top}(D''_m), P_x) \ .$$

(ii) Let D and $D^{'} \in \mathcal{D}_n$, then $\vartheta(D) \cap \vartheta(D^{'}) = \emptyset$; when D and D' are such that ϑ performs the insertion of $\mathbf{x}\overline{\mathbf{x}}$ in their last descent, the result follows from the fact that, for each $P \in \mathbf{P}(D)$ and for each $P' \in \mathbf{P}(D')$, we have
$$\text{insert}(D, \mathbf{x}\overline{\mathbf{x}}, P_x) = \text{insert}(D', \mathbf{x}\overline{\mathbf{x}}, P'_x) \Longrightarrow D = D'.$$
When $\vartheta(D), \vartheta(D') \in \mathcal{D}^2$, we have two cases.
- $\text{Val}(\vartheta(D))[2] \neq \text{Val}(\vartheta(D'))[2] \Longrightarrow \vartheta(D) \neq \vartheta(D')$.
- $\text{Val}(\vartheta(D))[2] = \text{Val}(\vartheta(D'))[2]$; if $\vartheta(D) = \vartheta(D')$ then the construction (i) above shows that $D = D'$.

We are now in a position to state our main result.

Proposition 2. *Let $\alpha \in \mathbb{N}^+$ and $\beta \in \mathbb{N}$, then $\Omega^1_{\alpha,\beta} \cong \Omega^1$.*

Corollary 1. *Let $h, \alpha \in \mathbb{N}^+$, $\beta \in \mathbb{N}$ and $h \leq \alpha + \beta$. We have $\Omega^h_{\alpha,\beta} \cong \Omega^h$.*

Proof. It is a direct consequence of Proposition 2. Indeed, the rules $\Omega^h_{\alpha,\beta}$ and Ω^h both enumerate the class of Dyck paths beginning with h rise steps. □

Remark 1. *We then have an infinite set of systems defining Ballot numbers. In particular, the following define the Catalan numbers:*

$$\Omega_{2,0} = \begin{cases} (1) \\ (1) \rightsquigarrow (2) \\ (2k) \rightsquigarrow (1)^k(4)(6)\dots(2k)(2k+2); \end{cases} \tag{11}$$

$$\Omega_{3,1} = \begin{cases} (1) \\ (1) \rightsquigarrow (2) \\ (2) \rightsquigarrow (2)(3) \\ (3) \rightsquigarrow (2)(3)(4) \\ (3k+1) \rightsquigarrow (1)^k(2)^k(4)(7)\dots(3k+1)(3k+4). \end{cases} \tag{12}$$

3.3. **Computation of the valuation for Dyck paths.** For sake of completenes we provide a computation of the valuation described in section 3.1 applied to the path in Figure 6.

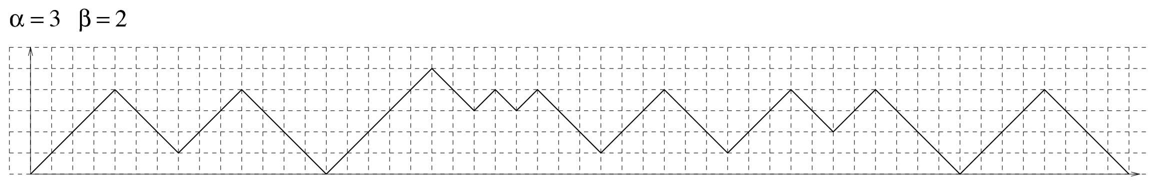

FIGURE 6. A Dyck D path with $Val(D) = (2, 0)$.

The path D factors as $D = D_1D_2D_3$, and it can be easily checked that

$\mathrm{Val}(D_1) = \overline{\mathrm{Val}}(D_1, 0) = (2, 1),$

$\mathrm{Val}(D_1D_2) = \overline{\mathrm{Val}}(D_1D_2, 0) = (1, 1),$

$\mathrm{Val}(D_1D_2D_3) = \overline{\mathrm{Val}}(D_1D_2D_3, 0) = (2, 0).$

4. Concluding remarks and open questions.

The equivalence relation $\cong$ partitions a set $\mathcal{R} \subset \mathcal{S}$ of systems into equivalence classes, identified by the corresponding number sequence. For instance, if $\mathcal{R}$ is the class of *rational systems*, those having a rational generating function, we already know that finite systems are in it. On the other hand, there exists rational generating functions that are not the growth sequence of a D0L system ([13] Theorem III.4.11). Therefore, many problems arise naturally concerning

- finiteness of the equivalence classes: is $|[\Omega]_{\cong}| < \infty$ when Ω is finite? More generally, for a given rational generating function, is its class finite?
- the characterization of the rules in a given equivalence class;
- the extension of the decidability of equivalence for finite systems to a larger class, by using the same decision procedure;
- operations on rules (or trees) that provide equivalent systems.

By Theorem 1, the class of finite systems in included in the class of rational systems. On the other hand, Theorem III.4.11 of [13] characterizes the rational functions with integers coefficients that are the generating functions of D0L systems. Therefore, the following problem seems natural.

Conjecture 1: *Each rational system is equivalent to a finite one.*

Actually, a weaker statement is the following.

Conjecture 2: *A system counting a regular language is equivalent to a finite system.*

In some recent discussions with Cyril Banderier, of INRIA, we were led to speculate about *algebraic* systems.

Conjecture 3: *A system with algebraic generating function is equivalent to a factorial system.*

References

[1] C. Banderier, M. Bousquet-Mélou, A. Denise, P. Flajolet, D. Gardy and D. Gouyou-Beauchamps, On generating functions of generating trees, *Proceedings of 11th FPSAC, Barcelona* (1999), 40–52.

[2] E. Barcucci, A. Del Lungo, A. Frosini, S. Rinaldi, From rational functions to regular languages, *Proceedings of 12th FPSAC, Moscow* (2000), 633–644.

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

[4] E. Barcucci, A. Del Lungo, E. Pergola, R. Pinzani, Some combinatorial interpretations of q-analogs of Schröder numbers, *Annals of Combinatorics* **3** (1999), 173–192.

[5] E. Barcucci, E. Pergola, R. Pinzani, S. Rinaldi, Eco method and hill-free paths, *Seminaire Lotharingien de Combinatorie*, B46b (2001) 14 pp.

[6] F. R. K. Chung, R. L. Graham, V. E. Hoggatt, M. Kleimann, The number of Baxter permutations, *J. Combin. Theory Ser. A* **24** (1978), 382–394.

[7] E. Deutsch, Operations on production matrices, (Personal Communication).

[8] L. Ferrari, E. Pergola, R. Pinzani, S. Rinaldi, An algebraic characterization of the set of succession rules, *Theoret. Comp. Sci.* (to appear).

[9] O. Guibert, *Combinatoire des permutations à motifs exclus en liason avec mots, cartes planaries et tableaux de Young*, Thèse de l'Université de Bordeaux I, (1996).

[10] E. Pergola, R. Pinzani, S. Rinaldi, Approximating algebraic functions by means of rational ones, *Theoret. Comp. Sci.* **270** (2002), 643–657.

[11] M. Robson (Personal communication).

[12] S. Rinaldi, *Succession rules: the whole story*, Phd thesis, University of Florence, (2001).

[13] G. Rozenberg, A. Salomaa, *The mathematical theory of L systems*, Academic Press, London (1980).

[14] A. Salomaa and M. Soittola, *Automata-theoretic aspects of formal power series*, Springer-Verlag, New York (1978).

[15] N. J. A. Sloane and S. Plouffe, *The encyclopedia of integer sequences*, Academic Press, New York (1996).

[16] J. West, Generating trees and the Catalan and Schröder numbers, *Discrete Mathematics* **146** (1995), 247–262.

LaCIM, Université du Québec à Montréal, CP 8888, Succ. Centre-Ville, Montréal (QC), Canada H3C 3P8
E-mail address: brlek@lacim.uqam.ca

Dipartimento di Sistemi e Informatica, Via Lombroso 6/17, 50134 Firenze, Italy
E-mail address: {duchi, elisa}@dsi.unifi.it

Dipartimento di Matematica, Via del Capitano, 15, 53100 Siena, Italy
E-mail address: rinaldi@unisi.it