ECO-generation for some restricted classes of compositions

Jean-Luc Baril and Phan-Thuan Do[1]

LE2I UMR-CNRS 5158, Université de Bourgogne
B.P. 47 870, 21078 DIJON-Cedex France
e-mail: barjl@u-bourgogne.fr
[1]Hanoi University of Science and Technology, Vietnam
Department of Computer Science
e-mail: thuandp@soict.hut.edu.vn

July 14, 2012

Abstract

We study several restricted classes of compositions by giving one-to-one maps between them and different classes of restricted binary strings or pattern avoiding permutations. Inspired by the ECO method [8], new succession rules for these classes are presented. Finally, we obtain generating algorithms in Constant Amortized Time (CAT) for theses classes.

Keywords : Composition of an integer, ECO method, succession rule, generating tree, pattern avoiding permutation.

1 Introduction

A *composition* of an integer n is an ordered collection of one or more positive integers whose sum is n. So, a composition c of n can be written $c = (c_1, c_2, \dots, c_k)$ with $c_1+c_2+\cdots+c_k = n$ and $c_i \geq 1$ for all $i \leq k$. The integer k corresponds to the number of *parts* of the composition. Let $C(n)$ be the set of compositions of n. It is well known that the cardinality of $C(n)$ is 2^{n-1} and there is a one-to-one correspondence between $C(n)$ and binary strings of length $n-1$ (see Definition 1). There are many studies about enumeration of compositions and their restrictions: $(1, p)$-compositions, *i.e.,* compositions whose parts are 1 or p have been introduced in [12, 15, 16]; compositions with no occurrence of part p have been studied in [17]; see also [1, 14, 24, 25, 27, 28, 30, 32]. However, a very few articles deal with their exhaustive generations. Some Gray codes are given for compositions of a positive integer n in

[31, 37]; for compositions with parts of size smaller than p in [36]; or for $(1,p)$-compositions in [12, 15]. These papers mostly study the classes of compositions in terms of binary strings. On the other hand, in [30], some results are provided using restricted permutations for a few classes of compositions, but they cannot be considered as avoidance patterns. More recently, a generalization of the Simion-Schmidt injection [35] gave a bijection between binary strings and pattern avoiding permutations [29] which creates a natural link between compositions and pattern avoiding permutations. For example, the class of compositions is in one-to-one correspondence with the class of permutations avoiding 321 and 312 [4, 26]; the set of compositions of n with all parts of sizes smaller than $(p+1)$ is enumerated by the p-generalized Fibonacci numbers, see [4, 11, 27, 28] and there is a bijection between this set and permutations avoiding the patterns 321, 312 and $234\cdots(p+1)1$.

In this paper, we use the ECO method [8] (Enumeration Combinatorial Object method) in order to generate some restricted classes of compositions represented as binary strings or pattern avoiding permutations. The *ECO method* is a recursive description of a combinatorial object class which explains how an object of size n can be reached from one and only one object of smaller size (see for example [2, 3, 5, 6, 7, 13, 18, 19, 21, 22, 23]). It consists to define a system of *succession rules* for a combinatorial object class which induces a generating tree such that each node is labeled by the number of its successors. In fact, the set of successions rules describes for each node the label of its successors. More formally, the root of the generating tree is labeled (b), $b \in \mathbb{N}^+$, and we define the rules Ω:

$$\{(k) \rightsquigarrow (e_1(k))(e_2(k))\cdots(e_k(k)), k \in \mathbb{N}\},$$

where $e_i : \mathbb{N}^+ \longrightarrow \mathbb{N}^+$. This means that each node labeled (k) has k successors labeled $(e_1(k))$, $(e_2(k))$,..., $(e_k(k))$. For $\ell \geq 1$, the symbol $\overset{\ell}{\rightsquigarrow}$ means that the succession rule transforms an element of size n into another of size $n+\ell$. For $\ell = 1$ we frequently omit the superscript ℓ over $\rightsquigarrow$.

By coding each node of the generating tree with either a binary string or a permutation, we deduce new bijections between classes of restricted compositions, pattern avoiding permutations and restricted binary strings.

This paper is organized as follows. Section 2 recalls the definition of pattern avoiding permutations, and gives existing links between compositions, binary strings and pattern avoiding permutations. Sections from 3 to 7 present succession rules for compositions with a given number of parts, compositions with at most p parts, $(1,p)$-compositions and compositions without parts of a given size. Moreover, each induced generating tree will be encoded by binary strings and pattern avoiding permutations. Finally, we deduce efficient algorithms (Constant Amortized Time algorithms) for generating all these classes (Constant Amortized Time means that the total amount of computation divided by the number of objects is bounded by a constant independent of the size of objects).

2 Definitions and notations

Let $\mathfrak{S}_n$ be the set of permutations on $[n] = \{1, 2, \ldots, n\}$. We represent a permutation $\pi \in \mathfrak{S}_n$ in one line notation: *i.e.,* $\pi = \pi_1\pi_2\cdots\pi_n$ where $\pi_i = \pi(i)$ for all $i \leq n$. A permutation $\pi \in \mathfrak{S}_n$ contains the pattern $\tau \in \mathfrak{S}_k$ if and only if a sequence of indices $1 \leq i_1 < i_2 < \cdots < i_k \leq n$ exists such that $\pi(i_1)\pi(i_2)\cdots\pi(i_k)$ is order-isomorphic to τ. We denote by $\mathfrak{S}_n(\tau)$ the set of n-length permutations avoiding the pattern τ, *i.e.*, permutations that do not contain τ. For instance, the permutation $\mathbf{5}\mathbf{2}3\mathbf{1}64$ contains the pattern 321 while $314265 \in \mathfrak{S}_6(321)$. Moreover, we consider a *barred pattern* $\bar{\tau}$, *i.e.,* a permutation in $\mathfrak{S}_k$ having a bar over one or several consecutive entries (see [34]). Let r, $1 \leq r \leq k-1$, be the number of barred elements in $\bar{\tau}$; τ be the permutation on $[k]$ identical to $\bar{\tau}$ but unbarred; and $\hat{\tau}$ be the permutation on $[k-r]$ made up of the $k-r$ unbarred elements of $\bar{\tau}$ rewritten to be a permutation on $[k-r]$. Let $b = b_1\cdots b_k \in \{0,1\}^k$ such that $b_i = 1$ if and only if the i-th entry of $\bar{\tau}$ is barred. Then $\pi \in \mathfrak{S}_n$ avoids the pattern $\bar{\tau}$ if and only if each pattern $\hat{\tau}$ in π can be expanded into a pattern τ in π such that the positions of the extended entries correspond to the positions of 1s in b. For example, if $\bar{\tau} = 21\overline{34}$ then $b = 0011$ and $2134 \in \mathfrak{S}_4(\bar{\tau})$ and $21435 \notin \mathfrak{S}_5(\bar{\tau})$ since 43 can not be expanded into a pattern $43ab$ where $4 < a < b$. Now we define a special pattern denoted $\dot{\tau} = 23\cdots(p-1)p\dot{1}$ (see [10]). A permutation π avoids $\dot{\tau}$ if and only if each pattern $23\cdots(p-1)1$ can be extended to a pattern $23\cdots(p-1)p1$ such that the positions of the extended values do not matter, *i.e.*, each pattern $23\cdots(p-1)1$ is contained in a pattern $23\cdots(p-1)p1$ of π. For instance, if $\dot{\tau} = 23\dot{1}$ then $2341 \in \mathfrak{S}_4(\dot{\tau})$ and $21 \notin \mathfrak{S}_2(\dot{\tau})$ while $2341 \notin \mathfrak{S}_4(23\bar{4}1)$.

It is well-known that the set $\mathcal{C}(n+1)$ of compositions of $n+1$ is in one-to-one correspondence with the set $\mathcal{B}(n)$ of binary strings of length n. The following bijection φ shows this correspondence.

Definition 1 *Let $c = (c_1, c_2, \ldots, c_k)$ be a composition of $n+1$. The bijection φ between $\mathcal{C}(n+1)$ and $\mathcal{B}(n)$ is defined by: $\varphi(c) = 1^{c_1-1}01^{c_2-1}0\cdots1^{c_{k-1}-1}01^{c_k-1}$.*

For instance, if $c = (1, 3, 2, 3)$ then $\varphi(c) = 0\mathbf{11}0\mathbf{1}0\mathbf{11}$.

On the other hand, Juarna and Vajnovszki in [29] gave a bijection ϕ between the binary strings in $\mathcal{B}(n)$ and the permutations in $\mathfrak{S}_{n+1}(321, 312)$. This bijection is considered as a generalization of the Simion-Schmidt injection [35].

Definition 2 *Let $b = b_1b_2\cdots b_n \in \mathcal{B}(n)$. The bijection ϕ between $\mathcal{B}(n)$ and $\mathfrak{S}_{n+1}(321, 312)$ is defined by: $\pi = \phi(b) \in \mathfrak{S}_{n+1}$ which has its i-th value π_i given by the following rule: if $X_i = [n+1] \setminus \{\pi_1, \pi_2, \ldots, \pi_{i-1}\}$, then*

$$\pi_i = \begin{cases} \text{the minimum value in } X_i, \text{ if } b_i = 0, \text{ or } i = n+1 \\ \text{the second minimum value in } X_i, \text{ if } b_i = 1. \end{cases}$$

For instance, if $b = \mathbf{0}11\mathbf{0}1\mathbf{0}11$ then $\phi(b) = \mathbf{1}34\mathbf{2}6\mathbf{5}89\mathbf{7}$.

3 Compositions of n with parts of size at most p

The set $\mathcal{C}_{\leq p}(n)$ of compositions of n with parts of size at most p is enumerated by the $(n+1)$-th p-generalized Fibonacci number (see [11]). The map φ (Section 2) induces a bijection between $\mathcal{C}_{\leq p}(n)$ and the set $\mathcal{B}_{<p}(n-1)$ of binary strings of size $n-1$ without p consecutive ones. It is proved [4, 11] that there are also one-to-one correspondences with the two classes of permutations $\mathfrak{S}_n(321, 231, (p+1)12\cdots p)$ and $\mathfrak{S}_n(321, 312, 23\cdots(p+1)1)$. These permutation classes admit known succession rules (see [4, 26] and Table 1) and they can be generated in constant amortized time (see also [11, 9, 33, 36] for Gray code listing).

4 Compositions of n with exactly p parts

The set $\mathcal{C}_p(n)$ of compositions of n with exactly p parts is enumerated by the binomial coefficient $\binom{n-1}{p-1}$. Also, $\mathcal{C}_p(n)$ is in one-to-one correspondence with the set $\mathcal{B}_{p-1}(n-1)$ of binary strings of length $n-1$ and having exactly $p-1$ zeros. The function φ (see Section 2) shows such a bijection. The following theorem gives a system of succession rules in order to generate the sets $\mathcal{B}_{p-1}(n-1)$ and $\mathfrak{S}_n(132, 312, (p+1)\cdots 21, 12\cdots(n-p+1)(n-p+2))$. In this part, we say that the level of a node in the generating tree is the length of the unique path between the root and this node, plus p (thus the root is on the level p).

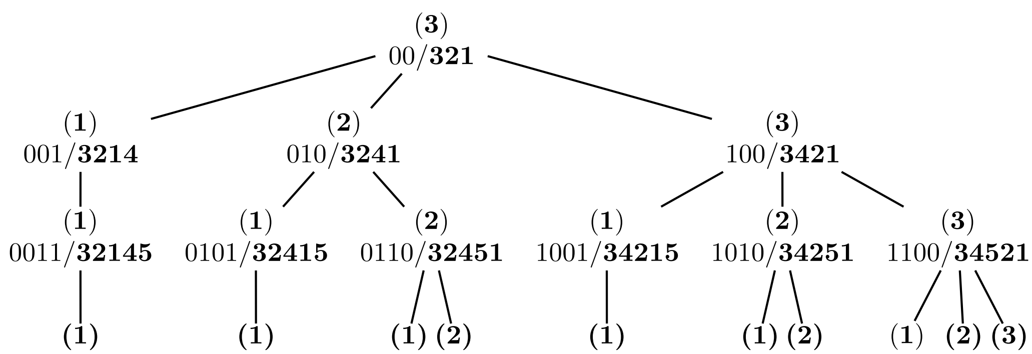

Figure 1: The first levels of the generating tree (Ω_3) (the level of the root is 3). Each node on the level n is coded by one binary string in $\mathcal{B}_2(n-1)$ or by one permutation in $\mathfrak{S}_n(132, 312, 4321, 12\cdots(n-2)(n-1))$.

Theorem 1 *For $p \geq 1$, a system (Ω_p) of succession rules for the set $\mathcal{C}_p(n)$ is:*

$$(\Omega_p)\left\{\begin{array}{l}(p)\\(k)\rightsquigarrow(1)(2)\cdots(k-1)(k).\end{array}\right.$$

Each level $n \geq p$ of the generating tree induced by (Ω_p) can be coded by the binary strings of $\mathcal{B}_{p-1}(n-1)$ or by the permutations in $\mathfrak{S}_n(132, 312, (p+1)\cdots 21, 12\cdots(n-p+1)(n-p+2))$. A node other than the root and labeled (k) is coded by a binary string of the form $b = b'10^{k-1}$ (resp. a permutation $\pi = \pi' n(k-1)(k-2)\cdots 21$) and its successors are obtained from b (resp. π) either by inserting 1 (resp. $n+1$) between two entries of the suffix 10^{k-1} (resp. $n(k-1)(k-2)\cdots 21$) or by appending 1 (resp. $n+1$) on the right (see Figure 1).

Proof. We attach the binary string 0^{p-1} to the root of the generating tree obtained by (Ω_p) and we proceed by induction on the level n of the tree (the root being on the level p by convenience). So we assume that the level $(n-1)$ generates once each binary string of $\mathcal{B}_{p-1}(n-2)$. Let $b \in \mathcal{B}_{p-1}(n-2)$ such that $b = b'0^k$, $k \leq p-1$, where b' is either empty or has 1 on its right. Therefore, by inserting 1 on the right of b, or between two entries of the suffix 0^k, or on the left of 0^k, we produce $k+1$ binary strings of $\mathcal{B}_{p-1}(n-1)$ and each binary string $c = b'10^{k-\ell+1}10^{\ell-1}$ obtained by this process has ℓ successors labeled (1), (2), $\ldots$, (ℓ). Conversely, each binary string of $\mathcal{B}_{p-1}(n-1)$ can be uniquely obtained from an element of $\mathcal{B}_{p-1}(n-2)$ by this construction.

Now, we define a map ϕ' from $\mathcal{B}_{p-1}(n-1)$ to $\mathfrak{S}_n(132, 312)$.

Let $b = b_1 b_2 \cdots b_{n-1} \in \mathcal{B}_{p-1}(n-1)$. If $\pi = \phi'(b_1 b_2 \cdots b_{n-1})$ then $\pi_1 = p$ and for $i \geq 2$,

$$\pi_i = \left\{\begin{array}{ll} p-\ell & \text{if } b_{i-1} \text{ is the } \ell\text{-th } 0 \text{ from the left}\\ p+\ell & \text{if } b_{i-1} \text{ is the } \ell\text{-th } 1 \text{ from the left.}\end{array}\right.$$

For instance, if $p = 5$ and $n = 9$ then $\phi'(1\mathbf{00}11\mathbf{0}1\mathbf{0}) = \mathbf{5}6\mathbf{43}78\mathbf{2}9\mathbf{1}$. In fact, the image by ϕ' of an element in $\mathcal{B}_{p-1}(n-1)$ is a permutation of $\mathfrak{S}_n(132, 312)$ verifying $\pi_1 = p$, or equivalently a permutation of $\mathfrak{S}_n(132, 312)$ that avoids the two patterns $(p+1)\cdots 21$ and $12\cdots(n-p+1)(n-p+2)$. Now let us prove that ϕ' is a bijection from $\mathcal{B}_{p-1}(n-1)$ to $\mathfrak{S}_n(132, 312, (p+1)\cdots 21, 12\cdots(n-p+1)(n-p+2))$. Indeed, a permutation π in $\mathfrak{S}_n(132, 312, (p+1)\cdots 21, 12\cdots(n-p+1)(n-p+2))$ verifies that π_i is either $\max\{\pi_j, j \leq i\}$ or $\min\{\pi_j, j \leq i\}$ which are respectively represented by 1 and 0 in order to obtain a binary string b of length $n-1$ (we do not consider the bit corresponding to π_1). Obviously, if π avoids $(p+1)p\cdots 21$ (resp. $12\cdots(n-p+1)(n-p+2)$) then b does not contain p zeros (resp. $n-p+1$ ones) which means that b contains exactly $p-1$ zeros. Moreover, if $c = b'10^{\ell}10^{k-\ell}$ is obtained from $b = b'10^k \in \mathcal{B}_{p-1}(n-2)$ by inserting 1 then $\phi'(c)$ is obtained from $\phi'(b)$ by inserting n on the same position from the right. $\square$

Notice that the set $\mathfrak{S}_n(132, 312, (p+1)\cdots 21, 12\cdots(n-p+1)(n-p+2))$ depends on a pattern of length $n-p+2$. However, it remains the open question: is it possible to find a finite basis B (independent of n) such that $\mathfrak{S}_n(B)$ is enumerated by $\binom{n-1}{p-1}$?

5 Compositions of n with at most p parts

The set $\mathcal{C}_{\#p}(n)$ of compositions of n with at most p parts is enumerated by $\sum_{k=1}^{p}\binom{n-1}{k-1}$. Moreover, $\mathcal{C}_{\#p}(n)$ is in one-to-one correspondence with the set $\mathcal{B}_{\#(p-1)}(n-1)$ of binary strings of length $n-1$ and having at most $p-1$ zeros. The function φ (see Section 2) shows such a bijection. The following theorem gives succession rules in order to generate $\mathcal{B}_{\#(p-1)}(n-1)$. Since $\mathcal{B}_{\#(p-1)}(n-1)=\bigcup_{i=0}^{p-1}\mathcal{B}_i(n-1)$ these rules are obtained by a simple adaptation of the rules described in the previous section. Here we say that the level of a node in the generating tree is the length of the unique path between the root and this node (the root is on the level 0).

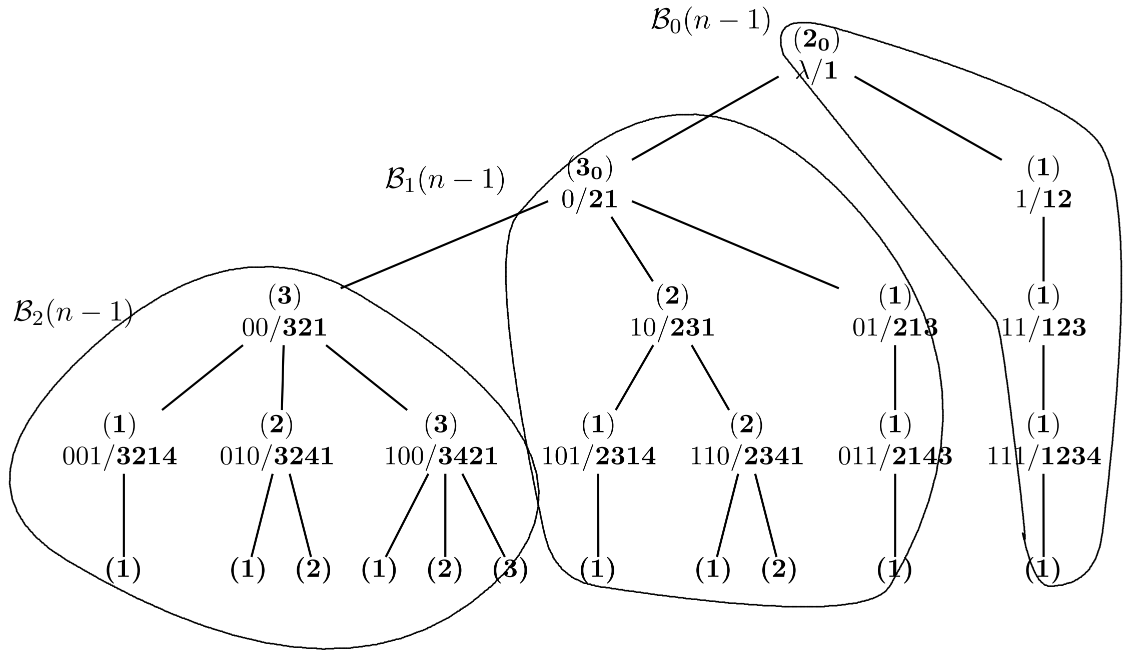

Figure 2: The first levels of the generating tree ($\Omega_{\#3}$). Each node on the level n is coded by one permutation in $\mathfrak{S}_{n+1}(132, 312, 4321)$ or by one binary string in $\mathcal{B}_{\#2}(n)$. Encircled subtrees correspond to the subsets $\mathcal{B}_i(n-1)$ for $0 \le i \le 2$.

Theorem 2 *For $p \ge 1$, a system $(\Omega_{\#p})$ of succession rules for the set $\mathcal{C}_{\#p}(n)$ is:*

$$(\Omega_{\#p})\left\{\begin{array}{lll} (2_0) & & \\ (k_0) & \rightsquigarrow ((k+1)_0)(k-1)\cdots(2)(1), & \text{if } 2 \le k < p \\ (p_0) & \rightsquigarrow (p)(p-1)\cdots(2)(1) & \\ (k) & \rightsquigarrow (1)(2)\cdots(k), & \text{if } 1 \le k \le p. \end{array}\right.$$

Each level $n \geq 0$ of the generating tree induced by $(\Omega_{\#p})$ can be coded by the binary strings of $\mathcal{B}_{\#(p-1)}(n)$ or by the permutations in $\mathfrak{S}_{n+1}(132, 312, (p+1)\cdots 21)$.

- *A node labeled (k_0), $2 \leq k \leq p$, is coded by the binary string 0^{k-2} (resp. the permutation $(k-1)(k-2)\cdots 21$) and its successors are obtained either by appending 0 (resp. k) on the left or by inserting 1 (resp. k) between two zeros, on the right or on the left (resp. on the same position from the right as for binary strings).*
- *All other nodes obey to the rules described in Theorem 1.*

Proof. The proof is directly deduced from Theorem 1. Indeed, a node labeled (k_0), $2 \leq k \leq p$, produces $k-1$ nodes labeled $(1), (2), \ldots, (k-1)$ which have the same succession rules as those of Theorem 1, and either one node labeled $((k+1)_0)$ if $k \neq p$ or one node labeled (k) otherwise. This means that the subtree T rooted by a node labeled (k_0), $k \neq p$, has one subtree T_1 rooted by a node labeled $((k+1)_0)$ that generates the sets $\mathcal{B}_k(n-1)$ for $n-1 \geq k$ (see Theorem 1). Now let $T_2 = T \backslash T_1$ be the subtree of T obtained by deleting all nodes of T_1. So, T_2 generates the set $\mathcal{B}_{k-1}(n-1)$ for $n-1 \geq k-1$. Finally, the complete generating tree of $(\Omega_{\#p})$ is exactly the union of subtrees T_i for $0 \leq i \leq p-1$ where T_i generates the set $\mathcal{B}_i(n-1)$ where $n-1 \geq i$. This proves that $(\Omega_{\#p})$ generates $\mathcal{B}_{\#(p-1)}(n)$. By duality and with the same argument, it also generates all permutations in $\mathfrak{S}_{n+1}(132, 312, (p+1)\cdots 21)$ (see Figure 2). $\square$

6 Compositions of n with parts 1 and p

Let $\mathcal{C}_{1,p}(n)$ be the set of compositions of n with parts 1 and p. The following bijection φ' (see for example [12]) gives a bijection between $\mathcal{C}_{1,p}(n)$ and the set $\mathcal{B}_{\geq p-1}(n-p+1)$ of binary strings of length $n-p+1$ with at least $p-1$ zeros between two ones.

Definition 3 *Let $c = (c_1, c_2, \ldots, c_\ell)$ be a composition of n such that $c_i \in \{1, p\}$ for all $i \leq \ell$. We define the bijection φ' between $\mathcal{C}_{1,p}(n)$ and $\mathcal{B}_{\geq p-1}(n-p+1)$ by the following algorithmical process. We initialize $b = \lambda$ (the empty string). For each i from 1 to ℓ, if $c_i = 1$ then we modify b by appending 0 on its right; otherwise (i.e., $c_i = p$), we modify b by appending 10^{p-1} on its right. Finally, we delete $p-1$ zeros on the right of b which defines a binary string b of length $n-p+1$ with at least $p-1$ zeros between two ones.*

For instance, if $n = 12$, $p = 3$ and $c = (1, 3, 1, 3, 3, 1)$ then $\varphi'(c) = 0\mathbf{100}0\mathbf{100}\mathbf{10}$.

Theorem 3 *For $p \geq 2$, a system $(\Omega_{1,p})$ of succession rules for the $(1,p)$-compositions is given by:*

$$(\Omega_{1,p}) \begin{cases} (2) \\ (2) \rightsquigarrow (2)(1_0) \\ (1_i) \rightsquigarrow (1_{i+1}), \quad \textit{for } 0 \leq i < p-2 \\ (1_{p-2}) \rightsquigarrow (2). \end{cases}$$

Each level n of the generating tree induced by $(\Omega_{1,p})$ can be coded by the binary strings in $\mathcal{B}_{\geq p-1}(n)$ (the root is on the level 0). A binary string of length n can be obtained from a string of length $n-1$ by inserting 0 or 1 on the last position (see Figure 3).

Proof. We will prove by induction that the nodes on the level k can be coded by the binary strings of the set $\mathcal{B}_{\geq p-1}(k)$ for all $k \geq 0$. Remark that this is true for the root which is coded by the empty string λ (by convenience the level of the root will be 0). So let us assume that each level $j \leq k$ is coded by the elements of $\mathcal{B}_{\geq p-1}(j)$.

Let α be a binary string of length $k+1$ on the level $k+1$ and let $\beta \in \mathcal{B}_{\geq p-1}(k)$ be its predecessor on the level k.

If α is obtained from β by inserting 0 on its right then α also belongs to $\mathcal{B}_{\geq p-1}(k+1)$. If α is obtained by inserting 1 on the right of β then β has two sons, so its label is (2), and its predecessor γ is labeled by (2) or (1_{p-2}), then $\beta = \gamma 0$.

(i) If γ is labeled (1_{p-2}), its predecessor γ_1 is labeled (1_{p-3}). We repeat this process until γ_{p-2}, *i.e.,* until we reach the label (1_0). Then γ_{p-2} is obtained from a binary string β' labeled (2) by inserting 1 on the right of β'.

Thus α is of the form $\alpha = \beta 1 = \beta' 10^{p-1}1$. Moreover, $\beta' \in \mathcal{B}_{\geq p-1}(k-p)$ by the recurrence hypothesis. We conclude that $\alpha \in \mathcal{B}_{\geq p-1}(k+1)$.

(ii) If γ is labeled (2), its predecessor γ_1 is labeled (1_{p-2}) or (2). If γ_1 is labeled (1_{p-2}), we return to the case (i) just above, so α has at least $p-1$ consecutive zeros between two ones. If γ_1 is labeled (2), we repeat the process by replacing γ_1 with γ and it will finish when: either we reach the label (1_{p-2}) which corresponds to the case (i), or we reach the root labeled (2). In any case, α contains $p-1$ consecutive zeros between two ones. Then $\alpha \in \mathcal{B}_{\geq p-1}(k+1)$.

Conversely, we consider $\alpha \in \mathcal{B}_{\geq p-1}(k+1)$ and we construct a path on the generating tree $(\Omega_{1,p})$ which generates this string. We distinguish two cases:

- $\alpha = \alpha' 10^j$, where $j \leq p-1$. So $\alpha' 1 \in \mathcal{B}_{\geq p-1}(k+1-j)$. Therefore, α' is labeled (2) and α is obtained from α' with either the path $(2)/\alpha' \rightsquigarrow (1_0)/\alpha' 1 \rightsquigarrow (1_1)/\alpha' 10 \rightsquigarrow \cdots \rightsquigarrow (1_j)/\alpha' 0^j$ or $(2)/\alpha' \rightsquigarrow (1_0)/\alpha' 1 \rightsquigarrow (1_1)/\alpha' 10 \rightsquigarrow \cdots \rightsquigarrow (1_{p-2})/\alpha' 10^{p-2} \rightsquigarrow (2)/\alpha' 10^{p-1}$;
- $\alpha = \alpha'' 10^j$, where $j \geq p$. Then $\alpha' = \alpha'' 10^{p-1} \in \mathcal{B}_{\geq p-1}(k+p-j)$. So, α is obtained from α' with a path of nodes all labeled (2) in the generating tree.

We repeat the same process by replacing α with α' and we will find the path from the root of the generating tree $(\Omega_{1,p})$ to reach α. For instance, $\alpha = 010010001$, the path to reach α from the root in the generating tree $(\Omega_{1,3})$ is:

$(2)/\lambda \rightsquigarrow (2)/0 \rightsquigarrow (1_0)/01 \rightsquigarrow (1_1)/010 \rightsquigarrow (2)/0100 \rightsquigarrow (1_0)/01001 \rightsquigarrow (1_1)/010010 \rightsquigarrow (2)/0100100 \rightsquigarrow (2)/01001000 \rightsquigarrow (1_0)/010010001$.

We finally conclude that the generating tree induced by $(\Omega_{1,p})$ is coded by the set $\mathcal{B}_{\geq p-1}(n)$ of binary strings of length n with at least $p-1$ zeros between two ones. $\square$

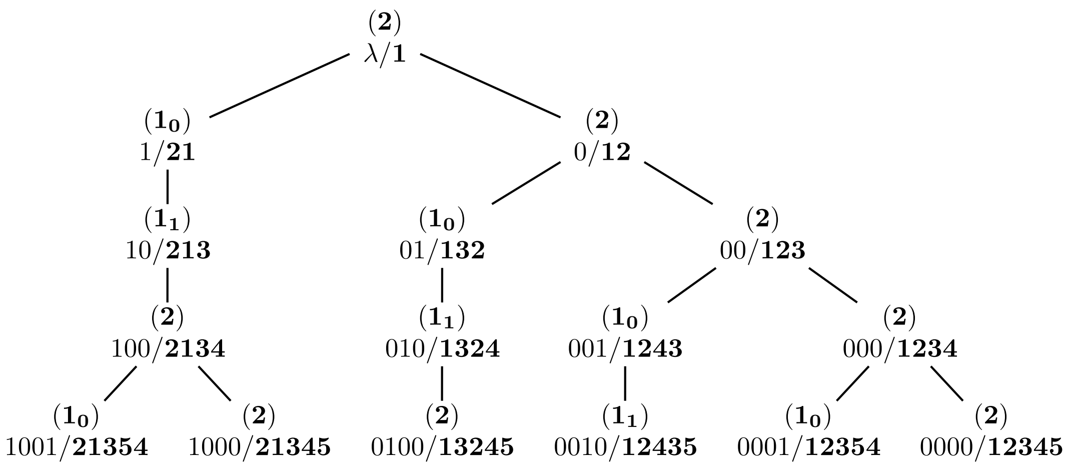

Figure 3: The first five levels of the generating tree $(\Omega_{1,3})$. Each node on the level n is coded by one permutation in $\mathfrak{S}_n(231, 312, 321, 21\overline{34}65)$ or by one binary string in $\mathcal{B}_{\geq 2}(n-1)$.

Theorem 4 *Each level $n \geq 0$ of the generating tree of $(\Omega_{1,p})$ can be coded by the permutations in $\mathfrak{S}_{n+1}(231, 312, 321, 21\overline{34\cdots(p+1)}(p+3)(p+2))$. A permutation of length n is obtained from a permutation π of length $(n-1)$ by inserting n either on the right of π or just before its last entry (see Figure 3).*

Proof. In [12], Baril and Moreira showed there is a bijection f between $\mathcal{B}_{\geq p-1}(n)$ and $\mathfrak{S}_{n+1}(231, 312, 321, 21\overline{34\cdots(p+1)}(p+3)(p+2))$. Moreover, the insertion of 0 (resp. 1) on the right of $b \in \mathcal{B}_{\geq p-1}(n)$ is equivalent to the insertion of $(n+1)$ on the right of $\pi = f(b)$ (resp. just before the last entry). This means that $\mathfrak{S}_{n+1}(231, 312, 321, 21\overline{34\cdots(p+1)}(p+3)(p+2))$ also codes the generating tree $(\Omega_{1,p})$. $\square$

7 Compositions of n without part of size p

Let $\mathcal{C}_{\widehat{p}}(n)$ be the set of compositions of n without part of size p. The bijection φ (see Section 2) transforms $\mathcal{C}_{\widehat{p}}(n)$ into the set $\mathcal{B}_{\widehat{p-1}}(n-1)$ of the binary strings of length $(n-1)$ without run of ones of length $p-1$ knowing that, a run of ones is a maximal substring of consecutive ones. For instance, the binary string $b = 0\mathbf{11}00\mathbf{111}0\mathbf{1}$ contains three runs of ones illustrated in boldface. In this section, we consider the concept of jumping succession rules introduced in [23]. This allows the construction of an element of size greater than $n+1$ from an element of size n (see Section 1).

Theorem 5 *For $p \geq 2$, a system of jumping succession rules $(\Omega_{\widehat{p}})$ for the compositions without part of size p is given by:*

$$(\Omega_{\widehat{p}}) \begin{cases} (2_0) \\ (2_i) \rightsquigarrow (2_0)(2_{i+1}), \qquad \textit{for } 0 \leq i \leq p-3 \\ (2_{p-2}) \overset{1}{\rightsquigarrow} (2_0) \\ \qquad\quad \overset{2}{\rightsquigarrow} (2_{p-1}) \\ (2_{p-1}) \rightsquigarrow (2_0)(2_{p-1}). \end{cases}$$

Each level n of the generating tree of $(\Omega_{\widehat{p}})$ is coded by the set $\mathcal{B}_{\widehat{p-1}}(n)$ (the root is on the level 0). Let b be a binary string in $\mathcal{B}_{\widehat{p-1}}(n)$ corresponding to a node of level n. Then b has two successors:
- if the two successors of b are on the level $n+1$, they are obtained by inserting 0 or 1 on the right of b.
- if one successor of b is on the level $n+1$ and the other on the level $n+2$, we insert 0 on the right of b in order to obtain the successor on the level $n+1$ and we insert 11 on the right of b in order to obtain that on the level $n+2$ (see Figure 4).

Proof. We proceed by induction. The root of the tree is coded by the empty string λ (by convenience the level of the root is 0). We assume that each level $k \leq n$ is coded by the elements of $\mathcal{B}_{\widehat{p-1}}(k)$. Let α be the binary string of length $(n+1)$ corresponding to a node on the level $n+1$ and let β be its predecessor on the level n or $n-1$, then β belongs to $\mathcal{B}_{\widehat{p-1}}(n)$ or $\mathcal{B}_{\widehat{p-1}}(n-1)$.

(i) If β is on the level $n-1$, then β is labeled (2_{p-2}), α is labeled (2_{p-1}) and α is obtained from β by inserting 11 on its right. Thus β is obtained from its predecessor β_1 labeled (2_{p-3}) (if $p-3>0$) by inserting 1 on its right. We repeat this process until we create β_{p-2}, *i.e.,* until we reach a node labeled (2_0). Necessarily β_{p-2} is obtained from its predecessor β' by inserting 0 on its right. Thus $\alpha = \beta 11 = \beta' 01^p$, and α does not contain any run of ones of length $p-1$, which implies that $\alpha \in \mathcal{B}_{\widehat{p-1}}(n+1)$.

(ii) If β is on the level n, then α is obtained from β by inserting 0 or 1 on its right.

- if α is obtained from β by inserting 0 on its right, then α obviously belongs to $\mathcal{B}_{\widehat{p-1}}(n+1)$.

- if α is obtained from β by inserting 1 on its right, then β is labeled (2) or (2_i), with $i < p-2$:

(a) if β is labeled (2), it is obtained from its predecessor (also labeled (2)) by inserting 1 on its right. We repeat this process until we reach the label (2_{p-2}). So we retrieve the case (i) above. Then, $\alpha = \alpha' 1^\ell = \beta' 01^p 1^\ell$, with $\ell > 0$, and $\beta' \in \mathcal{B}_{\widehat{p-1}}(n-p-l)$. Therefore $\alpha \in \mathcal{B}_{\widehat{p-1}}(n+1)$.

(b) if β is labeled (2_i), with the same process of the case (i), we have $\alpha = \beta' 1^{i+1}$, with $\beta' \in \mathcal{B}_{\widehat{p-1}}(n-i)$ with $i+1 < p-1$. Thus $\alpha \in \mathcal{B}_{\widehat{p-1}}(n+1)$.

Conversely, each string α in $\mathcal{B}_{\widehat{p-1}}(n+1)$ can be constructed on the level $n+1$ of the generating tree (Ω_p). Indeed, if $\beta = \alpha 0$ then $\beta \in \mathcal{B}_{\widehat{p-1}}(n+2)$. So β can be decomposed as $1^{c_1-1}01^{c_2-1}0 \cdots 1^{c_\ell-1}0$ such that $c = (c_1, c_2, \ldots, c_\ell) \in \mathcal{C}_{\widehat{p-1}}(n+3)$ (see the bijection $\varphi : \mathcal{C}_{\widehat{p}}(n+1) \rightsquigarrow \mathcal{B}_{\widehat{p-1}}(n)$). Let $\beta = \beta' 1^{c_\ell-1}0$, then $\beta' \in \mathcal{B}_{\widehat{p-1}}(n+2-c_\ell)$. We distinguish two cases:

- if $c_\ell < p$ then β is obtained from β' on the generating tree (Ω_p) by the path $(2_1) \rightsquigarrow (2_2) \rightsquigarrow \cdots \rightsquigarrow (2_{c_\ell-1}) \rightsquigarrow (2_0)$,

- if $c_\ell > p$ then β is obtained from β' on the generating tree (Ω_p) by the path $(2_1) \rightsquigarrow (2_2) \rightsquigarrow \cdots \rightsquigarrow (2_{p-2}) \overset{2}{\rightsquigarrow} (2_{p-1}) \rightsquigarrow (2_{p-1})^{c_\ell-p-1} \rightsquigarrow (2_0)$.

We repeat this process by replacing β with β' and we obtain a path from the root to β on the generating tree $(\Omega_{\widehat{p}})$. For instance on the generating tree $(\Omega_{\widehat{3}})$, if $\alpha = 1110100$ then the path for reaching α from the root is:
$(2_0)/\lambda \rightsquigarrow (2_1)/1 \overset{2}{\rightsquigarrow} (2_2)/111 \rightsquigarrow (2_0)/1110 \rightsquigarrow (2_1)/11101 \rightsquigarrow (2_0)/111010 \rightsquigarrow (2_0)/1110100$. □

In order to code the generating tree by permutations avoiding patterns, we use the new pattern $23\cdots n\dot{1}$ presented in Section 2.

Theorem 6 *Each level n of the generating tree induced by $(\Omega_{\widehat{p}})$ can be coded by the permutations $\pi \in \mathfrak{S}_n(312, 321, 23\cdots(p+1)\dot{1})$ as follows:*
• *If the two successors of π belong to the level $(n+1)$, they are obtained by inserting $n+1$ on the right or just before the last entry of π.*
• *If a successor of π is on the level $(n+1)$ and the other on the level $(n+2)$, we insert $n+1$ on the right of π in order to obtain the successor on the level $n+1$, and we insert $(n+1)(n+2)$ just before the last entry of π in order to obtain that of the level $(n+2)$ (see Figure 4).*

In order to prove this theorem, we present the following proposition.

Proposition 1 *The map ϕ defined in Section 2 is a bijection from $\mathcal{B}_{\widehat{p}}(n)$ to $\mathfrak{S}_{n+1}(312, 321, 23\cdots(p+2)\dot{1})$.*

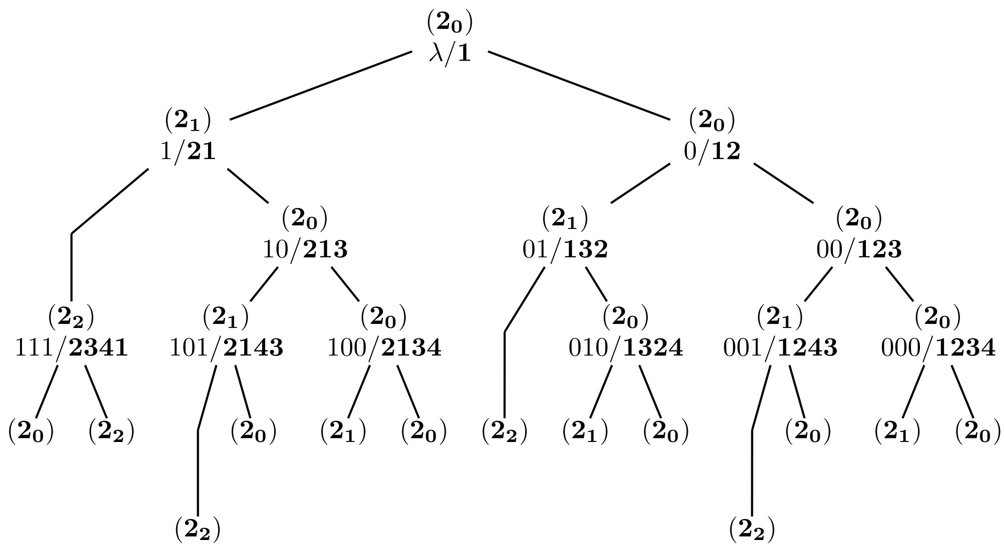

Figure 4: The first levels of the generating tree ($\Omega_{\widehat{3}}$). Each node on the level $n \geq 0$ is coded by a permutation in $\mathfrak{S}_{n+1}(312, 321, 234\dot{1})$ or by a binary string in $\mathcal{B}_{\widehat{2}}(n)$.

Proof. Recall that ϕ is a bijection from the set of binary strings of length n to the set of $(n+1)$-length permutations avoiding 321 and 312. So we will prove that $\phi(\mathcal{B}_{\widehat{p}}(n)) = \mathfrak{S}_{n+1}(321, 312, 23\cdots(p+2)\dot{1})$.

Let b be a binary string in $\mathcal{B}_{\widehat{p}}(n)$ and π be its image by ϕ in $\mathfrak{S}_{n+1}(321, 312)$. As $\phi(b) \in \mathfrak{S}_{n+1}(321, 312)$, there exist (see [36]) some indices $0 = k_0 < k_1 < \cdots < k_r < \cdots < k_m = n$ such that π is divided into m blocks

$$\pi = \boxed{\pi_1\pi_2\cdots\pi_{k_1}} \cdots \boxed{\pi_{k_{r-1}+1}\pi_{k_{r-1}+2}\cdots\pi_{k_r}} \cdots \boxed{\pi_{k_{m-1}+1}\pi_{k_{m-1}+2}\cdots\pi_{k_m}}$$

satisfying the two following conditions:

(i) the rightmost elements of each block are in increasing order (*i.e.*, $1 = \pi_{k_1} < \pi_{k_2} < \cdots < \pi_{k_m}$);

(ii) each block $\pi_{k_{r-1}+1}\pi_{k_{r-1}+2}\cdots\pi_{k_r}$ having at least two elements is of the form $(a+1)(a+2)\cdots(a+k_r-k_{r-1}-1)a$ with $a = \pi_{k_r}$.

For instance if $b = 110111$ then $\pi = \mathbf{231}5674$. It is remarkable from the definition of ϕ that if we add on the last position of b one occurrence of 0 and then divide the obtained binary string into separated blocks where each block contains exactly one occurrence of 0 at its end, then we obtain m blocks corresponding to the m blocks of $\phi(b)$. Furthermore, the number of consecutive 1s in each block of b is also the number, minus one, of elements of the respective block in $\phi(b)$ (see Definition 2). This leads us to the following claim: for $q \geq 2$, if one subsequence of $\phi(b)$ is a pattern $23 \ldots q1$ (of length q) then this sequence must belong to only one block of $\phi(b)$ (since the smallest element of each block of $\phi(b)$ is greater than the largest element of the previous blocks of $\phi(b)$). Moreover, since b does not contain exactly p consecutive 1s, $\phi(b)$ does not contain any block of length $p+1$ exactly. Hence, if the subsequence $23 \ldots p(p+1)1$ appears in $\phi(b)$ then we can extend it (without considering positions of extended entries) into a sequence $23 \ldots (p+1)(p+2)1$ which means that $\phi(b)$ avoids $23 \ldots (p+1)(p+2)\dot{1}$.

Conversely, we take a permutation π of $\mathfrak{S}_{n+1}(321, 312, 23 \ldots (p+1)(p+2)\dot{1})$. We also have the decomposition of π into blocks as above. It needs to show that $\phi^{-1}(\pi)$ belongs to $B_{\widehat{p}}(n)$. This is induced by the fact that all blocks of π considered as subsequences of π are, either of length less than $p+1$ (if they do not contain the reduced pattern $23 \ldots p(p+1)1$) or more than $p+1$ (if they contain the extended pattern $23 \ldots (p+2)1$). Therefore, $\phi^{-1}(\pi)$ is a binary string without p consecutive 1s. □

Now, the proof of Theorem 6 is obtained using the following remark. The insertion of 0 (resp. 1) on the right of $b \in \mathcal{B}_{\widehat{p}}(n)$ is equivalent to the insertion of $n+1$ on the right of $\pi = \phi(b)$ (resp. just before the last entry). And also the insertion of 11 on the right of $b \in \mathcal{B}_{\widehat{p}}(n)$ is equivalent to the insertion of $(n+1)(n+2)$ just before the last entry of $\pi = \phi(b)$. This means that $\mathfrak{S}_n(312, 321, 23 \cdots (p+1)\dot{1})$ also codes the generating tree $(\Omega_{\widehat{p}})$.

8 Algorithmic considerations and conclusion

In this section, we explain how all studied classes can be generated efficiently.

Let $\pi \in \mathfrak{S}_n$; the *sites* of π are the positions between two consecutive entries, before the first and after the last entry. We suppose that the sites are numbered from 1 to $n+1$ and from right to left. For example, the third site of the permutation $\pi = 463512$ is between the entries 5 and 1. Moreover, let τ be an n-length permutation (or equivalently a binary string) on a generating tree defined by a succession rule (Ω). Then, the i-th site of τ is said *active* if the element obtained from τ by the insertion of a value into this site also belongs to the generating tree. The active sites of τ are *right-justified* (see [20]) if all sites to the right of the leftmost active site are also active. If every elements on the generating tree are right-justified we will say that the generated class is *regular*. It is crucial to notice that all classes defined in this paper are *regular*.

An algorithm runs in Constant Amortized Time (CAT) if the amount of computations, after a small amount of preprocessing, is proportional to the number of generated objects. Many CAT algorithms exist in the literature, but we will take that presented in [20, 33]. This recursive algorithm acts on regular classes and enables us to ensure that we can generate all successors of a given node in constant amortized time. Thus the total amount of computations is proportional to the number of recursive calls. Moreover, the number obj of generated objects is at least $\frac{c-c_1}{2}$ where c is the total number of recursive calls, and c_1 is the number of recursive calls of degree one. So, for each generating tree of this study, we will calculate c_1.

- We immediately have $c_1 = 0$ for the generating tree defined in Section 7 and $\frac{c}{obj}$ evolves as $\mathcal{O}(1)$.

- In Section 6, a simple observation gets $c_1 \leq p(c - c_1)$ and $\frac{c}{obj}$ evolves as $\mathcal{O}(p)$.

- The generating tree of Section 5 is constituted of several generating trees of Section 4. So it suffices to compute c_1 for Section 4 (see below).

- In Section 4, each node produces exactly once a node of degree one. Thus the number of nodes of degree one and on levels at most $n-1$ in the generating tree (Ω_p) is equal to

$$c_1 = \sum_{i=p-1}^{n-1} \binom{i}{p-1}.$$

A simple calculation proves that, if $p \geq \lfloor \frac{n}{2} \rfloor$ and $n \geq 4$ then $c_1 \leq 2\binom{n}{p-1}$ which means that the number of nodes of degree one divided by 2 is smaller than the number of generated objects $obj = \binom{n}{p-1}$ for $p \geq \lfloor \frac{n}{2} \rfloor$. In this case, the complexity is $\mathcal{O}(1)$. The case $p \leq \lfloor \frac{n}{2} \rfloor$ is obtained *mutatis mutandis* by interchanging 0 with 1 in the binary strings of the generating tree.

Finally, this means that the total amount of computation divided by the number of objects is bounded by a constant independently to the size of the objects. Therefore all studied classes in this paper can be generated in Constant Amortized Time (CAT) using an algorithm similar to that of [20, 33]. We summarize our results in Table 1 that contains the successions rules of each studied classes and their corresponding pattern avoiding permutation classes.

References

[1] K. Alladi and V.E. Hoggatt. Compositions with ones and twos. *Fibonacci Quarterly*, 13(3):233–239, 1975.

[2] S. Bacchelli, E. Barcucci, E. Grazzini, and E. Pergola. Exhaustive generation of combinatorial objects by ECO. *Acta Informatica*, 40:585–602, 2004.

[3] C. Banderier, P. Flajolet, D. Gardy, M. Bousquet-Melou, A. Denise, and D. Gouyou-Beauchamps. Generating functions for generating trees. *Discrete Mathematics*, 246:29–55, 2002.

[4] E. Barcucci, A. Bernini, and M. Poneti. From Fibonacci to Catalan permutations. *PU.M.A.*, 2007(1-2):1–18, 2006.

Classes	Succession rules	Avoided patterns
$C(n)$	(2) $(2) \rightsquigarrow (2)(2)$	$\{321, 312\}$ [4, 26]
	(2) $(k) \rightsquigarrow (k+1)(1)^{k-1}$	$\{321, 231\}$ [4, 26]
$C_p(n)$	(p) $(k) \rightsquigarrow (1)(2)\cdots(k)$	$\{132, 312, (p+1)p\cdots 21,$ $12\cdots(n-p)(n-p+1)\}$
$C_{\#p}(n)$	(2_0) $(k_0) \quad \rightsquigarrow ((k+1)_0)(k-1)\cdots(2)(1) \quad \text{for } 2 \leq k < p$ $(p_0) \quad \rightsquigarrow (p)(p-1)\cdots(2)(1)$ $(k) \quad \rightsquigarrow (1)(2)\cdots(k-1)(k) \quad \text{for } 1 \leq k \leq p$	$\{132, 312, (p+1)p\cdots 21\}$
$C_{\leq p}(n)$	(2_0) $(2_0) \rightsquigarrow (2_0)(2_1)$ $(2_i) \rightsquigarrow (2_0)(2_{i+1}),\ \text{for } 1 \leq i < p-2$ $(2_{p-2}) \rightsquigarrow (2_0)(1)$ $(1) \rightsquigarrow (2_0)$	$\{321, 312, 234\cdots(p+1)1\}$ [4, 11]
	(2) $(k) \rightsquigarrow (k+1)(1)^{k-1}$ $(p) \rightsquigarrow (p)(1)^{k-1}$	$\{312, 231, (p+1)p\cdots 321\}$ [4, 11]
$C_{1,p}(n)$	(2) $(2) \rightsquigarrow (1_0)(2)$ $(1_i) \rightsquigarrow (1_{i+1}),\ \text{for } 0 \leq i < p-2$ $(1_{p-2}) \rightsquigarrow (2)$	$\{231, 312, 321,$ $21\overline{34\cdots(p+1)}(p+3)(p+2)\}$
$C_{\widehat{p}}(n)$	(2_0) $(2_i) \rightsquigarrow (2_0)(2_{i+1}),\ \text{for } 0 \leq i \leq p-3$ $(2_{p-2}) \overset{1}{\rightsquigarrow} (2_0)$ $\overset{2}{\rightsquigarrow} (2_{p-1})$ $(2_{p-1}) \rightsquigarrow (2_0)(2_{p-1})$	$\{312, 321, 23\cdots(p+1)\dot{1}\}$

Table 1: Classes of compositions, corresponding succession rules and corresponding classes of pattern avoiding permutations.

[5] E. Barcucci, A. Del Lungo, and E. Pergola. Random generation of trees and other combinatorial objects. *Theoretical Computer Science*, 218(2):218–232, 1999.

[6] E. Barcucci, A. Del Lungo, E. Pergola, and R. Pinzani. A methodology for plane tree enumeration. *Discrete Mathematics*, 180:45–64, 1998.

[7] E. Barcucci, A. Del Lungo, E. Pergola, and R. Pinzani. Directed animals, forests of trees and permutations. *Discrete Mathematics*, 204:41–71, 1999.

[8] 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*, 5:435–490, 1999.

[9] J.-L. Baril. More restrictive Gray codes for some classes of pattern avoiding permutations. *Information Processing Letters*, 109:799–804, 2009.

[10] J.-L. Baril. Classical sequences revisited with permutations avoiding dotted pattern. *Electronic Journal of Combinatorics*, 18:p#178, 2011.

[11] J.L. Baril and P.T. Do. ECO-generation for p-generalized Fibonacci and Lucas permutations. *Pu.M.A.*, 17(1-2):19–37, 2006.

[12] J.L. Baril and C. Moreira Dos Santos. Gray code for compositions of n with parts 1 and p. *Advances and Applications in Discrete Mathematics*, 3(1):67–84, 2009.

[13] S. Brlek, E. Duchi, E. Pergola, and S. Rinaldi. On the equivalence problem for succession rules. *Discrete Math.*, 298:142–154, 2005.

[14] L. Carlitz. Restricted compositions. *The Fibonacci Quart.*, 14:254–264, 1976.

[15] T. Chinburg, C.D. Savage, and H.S. Wilf. Combinatorial families that exponentially far from being listable in Gray code sequence. *Transactions of the AMS*, 351:379–402, 1999.

[16] P. Chinn and S. Heubach. $(1, k)$-Compositions. *Congressus Numerantium*, 164:183–194, 2003.

[17] P. Chinn and S. Heubach. Compositions of n with no occurrence of k. *Congressus Numerantium*, 164:33–51, 2003.

[18] A. Del Lungo, A. Frosini, and S. Rinaldi. ECO method and the exhaustive generation of convex polyominoes. *DMTCS*, pages 103–116, 2003.

[19] E. Deutsch, L. Ferrari, and S. Rinaldi. Production matrices. *Advances in Applied Mathematics*, 34:101–122, 2005.

[20] P.T. Do and V. Vajnovszki. Exhaustive generation of some classes of pattern avoiding permutations using succession functions. *Conference in honor of Donald E. Knuth*, 2007.

[21] E. Duchi, A. Frosini, R. Pinzani, and S. Rinaldi. A note on rational succession rules. *Journal of Integer Sequences, Article 03.1.7*, 6, 2003.

[22] L. Ferrari, E. Pergola, R. Pinzani, and S. Rinaldi. An algebraic characterization of the set of succession rules. *Theoretical Computer Science*, 281:351–367, 2002.

[23] L. Ferrari, E. Pergola, R. Pinzani, and S. Rinaldi. Jumping succession rules and their generating functions. *Discrete Math*, 271:29–50, 2003.

[24] R.P. Grimaldi. Compositions with odd summands. *Congressus Numerantium*, 142:113–127, 2000.

[25] R.P. Grimaldi and S. Heubach. Binary strings without odd runs of zeros. *Ars Combinatoria*, 75:241–255, 2005.

[26] O. Guibert. Combinatoire des permutations motifs exclus en liaison avec mots, cartes planaires et tableaux de Young. *Ph.D Thesis, Université Bordeaux 1*, 1995.

[27] S. Heubach and T. Mansour. Combinatorics of compositions and words. *Discrete Mathematics and its Applications (Boca Raton). CRC Press, Boca Raton, FL.*

[28] S. Heubach and T. Mansour. Compositions of n with parts in a set. *Congressus Numerantium*, 168:127–143, 2004.

[29] A. Juarna and V. Vajnovszki. Combinatorial isomorphism between Fibonacci classes. *Journal of Discrete Mathematical Science and Cryptography*, 11(2):147–158, 2008.

[30] S. Kitaev, T. McAllister, and K. Petersen. Enumerating segmented patterns in compositions and encoding with restricted permutations. *Integers: Electronic Journal of Combinatorial Number Theory*, 6:16pp, 2006.

[31] P. Klingsberg. A Gray code for compositions. *J. Algorithms*, 3:41–44, 1982.

[32] A. Knopfmacher and H. Prodinger. On Carlitz compositions. *European Journal of Combinatorics*, 19(5):579–589, 1998.

[33] T. Mansour, W.M. Dukes, M.F. Flanagan, and V. Vajnovszki. Combinatorial Gray codes for classes of pattern avoiding permutations. *Theoretical Computer Science*, 396:35–49, 2008.

[34] L. Pudwell. Enumeration schemes for permutations avoiding barred patterns. *The electronic Journal of Combinatorics*, 17,R29, 2010.

[35] R. Simion and F.W. Schmidt. Restricted permutations. *European J. Combin.*, 6:383–406, 1985.

[36] V. Vajnovszki. A loopless generation of bitstrings without p consecutive ones. *Discrete Mathematics and Theoretical Computer Science*, 1:227–240, Springer 2001.

[37] T.R. Walsh. Loop-free sequencing of bounded integer compositions. *J. Combin. Math. Combin. Comput.*, 33:323–345, 2000.