Efficient generating algorithm for permutations with a fixed number of excedances

Jean-Luc Baril

LE2I UMR-CNRS 5158, Université de Bourgogne
B.P. 47 870, 21078 DIJON-Cedex France
e-mail: barjl@u-bourgogne.fr

Abstract

In this paper, we develop a constant amortized time (CAT) algorithm for generating permutations with a fixed number of excedances. We obtain a Gray code for permutations having one excedance. We also give a bijection between the set of n-length permutations with exactly one excedance and the set $S_n(321, 2413, 3412, 21534)\backslash\{123\ldots(n-1)n\}$. This induces a Gray code for the set $S_n(321, 2413, 3412, 21534)\backslash\{123\ldots(n-1)n\}$.

keywords : Eulerian numbers, permutation, excedance, descent, pattern avoiding permutation, Gray code, generating algorithm.

1   Introduction

Let S_n be the set of all permutations of length n ($n \geq 1$). We represent permutations in one-line notation, *i.e.*, if $i_1, i_2, \ldots, i_n$ are n distinct values in $[n] = \{1, 2, \ldots, n\}$, we denote the permutation $\sigma \in S_n$ by the sequence $i_1 i_2 \ldots i_n$ if $\sigma(k) = i_k$ for $1 \leq k \leq n$. For instance, the identity permutation of length n, id_n, will be written $12\ldots(n-1)n$. Moreover, if $\gamma = \gamma(1)\gamma(2)\ldots\gamma(n)$ is an n-length permutation then the *composition* (*or product*) $\gamma \cdot \sigma$ is the permutation $\gamma(\sigma(1))\gamma(\sigma(2))\ldots\gamma(\sigma(n))$. In S_n, a *k-cycle* $\sigma = \langle i_1, i_2, \ldots, i_k \rangle$ is an n-length permutation verifying $\sigma(i_1) = i_2$, $\sigma(i_2) = i_3, \ldots$, $\sigma(i_{k-1}) = i_k$, $\sigma(i_k) = i_1$ and $\sigma(j) = j$ for $j \in [n]\backslash\{i_1, \ldots, i_k\}$; in particular, a *transposition* is a 2-cycle. Now, let us consider $\sigma \in S_n$. We say that $\sigma(i)$, $1 \leq i \leq n$, is a *strict excedance* (or *excedance* for short) of σ if $\sigma(i) > i$. For $0 \leq k \leq n - 1$, we denote by $E_{n,k}$ the set of all n-length permutations with exactly k excedances. Obviously, $\{E_{n,k}\}_{0 \leq k \leq n-1}$ forms a partition for S_n and it is well known (see for instance [9]) that the cardinality of $E_{n,k}$ is given by the Eulerian numbers $e(n, k)$ satisfying:

$$e(n, k) = (n - k) \cdot e(n - 1, k - 1) + (k + 1) \cdot e(n - 1, k) \qquad (1)$$

anchored by $e(n, 0) = 1$ and $e(n, n - 1) = 1$ for $n \geq 1$. Notice that $e(n, k)$ also enumerates the n-length permutations with exactly k descents ($\sigma(i)$ is a *descent*

1

of $\sigma \in S_n$ $(1 \leq i \leq n-1$, if $\sigma(i) > \sigma(i+1))$. Moreover, $e(n,1)$ is the number of Dyck paths of semi-length n having exactly one long ascent [23] (*i.e.*, ascent of length at least two).

Eulerian numbers have been widely studied in enumerative combinatorics [9, 11, 18, 19]. Foata and Schützenberger [12] give several fundamental properties of these numbers; some applications to analysis of algorithms are given by Knuth [14] and to combinatorics on words by Lothaire [17]. Many of these references study Eulerian numbers as a distribution statistic (called *Eulerian statistic*) in S_n. However, an efficient algorithm does not exist for generating all n-length permutations with a given number of excedances. Algorithms for generating other permutation classes have been published (generated in lexicographic order and in constant time per permutation, in amortized sense). For example, efficient algorithms are known for derangements [1], involutions [20], up-down permutations [8, 22], permutations with a fixed number of inversions [10], Fibonacci and Lucas permutations [5]. Also, several algorithms for generating permutation classes in Gray code order are given for permutations [13, 24] and their restrictions [15, 20, 21], derangements [7, 16], with a fixed number of cycles [4], involutions and fixed-point free involutions [26] or their generalizations (multiset permutations [25]).

In this paper, we provide an algorithm for generating n-length permutations with a fixed number of excedances, in Constant Amortized Time (CAT). An improvement of this algorithm allows us to list permutations with exactly one excedance in Gray code order. This code is optimal and two consecutive elements in the list differ in at most three positions. We construct a bijection between permutations with exactly one excedance and the set of avoiding permutations $S_n(321, 3413, 3412, 21534) \setminus \{123 \ldots (n-1)n\}$ and we also obtain a Gray code for this last set.

2 Generating permutations with a given number of excedances

In this section, we provide a CAT algorithm for generating the set $E_{n,k}$, *i.e.* the set of n-length permutations with k excedances.

2.1 Algorithm

Here, we show how one can recursively construct $E_{n,k}$ from $E_{n-1,k}$ and $E_{n-1,k-1}$ which will induce a constructive proof of the enumerating relation (1). We also give a lemma, crucial in the construction of our algorithm.

Let $\gamma \in E_{n-1,k}$ be an $(n-1)$-length permutation with k excedances, $n \geq 2$, $0 \leq k \leq n-2$; let i be an integer, $1 \leq i \leq n-1$. If we denote by σ the permutation in S_n obtained from γ by replacing $\gamma(i)$ by n and by appending $\gamma(i)$ on the right of γ, then we consider two cases:

(a) if $\gamma(i)$ is an excedance in γ, then $\sigma \in E_{n,k}$;

(b) otherwise, $\sigma \in E_{n,k+1}$.

Moreover, (c) if σ is obtained from γ by appending n on the right, then $\sigma \in E_{n,k}$. Conversely, each permutation in $E_{n,k}$, $n \geq 2$, can be uniquely obtained by one of these three constructions (a), (b) and (c), which gives a combinatorial proof of the relation (1).

Then we define below the functions ϕ_n and ψ_n as follow:

Definition 1 *For $0 \leq k \leq n-2$, an integer $i \in [n-1]$ and a permutation $\gamma \in E_{n-1,k}$, we define an n-length permutation $\sigma = \phi_{n-1}(i,\gamma)$ by*

$$\sigma(j) = \begin{cases} n & \text{if } j = i \\ \gamma(i) & \text{if } j = n \\ \gamma(j) & \text{otherwise.} \end{cases}$$

We also define the function ψ_{n-1} from $E_{n-1,k}$ to S_n that transforms $\gamma \in E_{n-1,k}$ in the permutation $\sigma \in E_{n,k}$ obtained from γ by appending n on its right.

For example, if $\gamma = 3142 \in E_{4,2}$, we have $\phi_4(3,\gamma) = 31524 \in E_{5,2}$; $\phi_4(2,\gamma) = 35421 \in E_{5,3}$ and $\psi_4(\gamma) = 31425 \in E_{5,2}$. More generally and with the same hypothesis of Definition 1, we can easily remark that:

$$\phi_{n-1}(i,\gamma) = \psi_{n-1}(\gamma) \cdot \langle i, n \rangle = \langle \gamma(i), n \rangle \cdot \psi_{n-1}(\gamma).$$

In the sequel of this paper, we will omit the index n in ϕ_n and ψ_n since it will be known by the context.

The above comments induce the following straightforward lemma that is the key of our algorithm.

Lemma 1 *Let n, m, ℓ, k be four natural numbers such that $1 \leq m < n$, $0 \leq \ell \leq m-1$, and $0 \leq k \leq n-1$. Let also $\gamma \in E_{m,\ell}$.*

- *If there exists i, $1 \leq i \leq m-1$, such that $\gamma(i)$ is an excedance of γ, then there is a permutation $\sigma \in E_{n,k}$ such that σ is obtained from $\phi(i,\gamma)$ (respectively $\psi(\gamma)$) by applying several times the functions ϕ and ψ if and only if: (i) $\ell \leq k$ and $m+1-\ell \leq n-k$.*

- *Now let us consider i, $1 \leq i \leq m$, such that $\gamma(i)$ is not an excedance of γ, then there is a permutation $\sigma \in E_{n,k}$ such that σ is obtained from $\phi(i,\gamma)$ by applying several times the functions ϕ and ψ if and only if: (ii) $\ell+1 \leq k$ and $m-\ell \leq n-k$.*

- *If $\sigma \in E_{n,k}$ is obtained from $\gamma \in E_{m,\ell}$ by the first previous construction and $\tau \in E_{n,k}$ from the second, then σ and τ are different.*

Now we explain the main difficulties in implementing our procedure $\mathbf{gen}(m,\ell)$ given in Figure 1. The procedure $\mathbf{gen}(1,0)$ generates recursively all permutations $\sigma \in E_{n,k}$. Indeed, let us consider that when we run the recursive call $\mathbf{gen}(m,\ell)$, the current permutation is $\sigma \in E_{m,\ell}$ where its excedances are in

positions $i_1, i_2, \ldots, i_\ell$, (*i.e.* the excedances of σ are $\sigma(i_1), \ldots, \sigma(i_\ell)$). Then, an array t_1 contains these ℓ positions $i_1, i_2, \ldots, i_\ell$. On the other hand, the indices in $[m] \backslash \{i_1, i_2, \ldots, i_\ell\}$ are stored in an array t_2 of length $r = m - \ell$. By considering the necessary conditions of Lemma 1, the procedure $\mathbf{gen}(m, \ell)$ generates permutations in $E_{m+1,\ell}$ or $E_{m+1,\ell+1}$ by applying to σ the functions ϕ and/or ψ. In order to obtain:

- $\psi(\sigma) \in E_{m+1,\ell}$; we add the index $(m+1)$ on the right of array t_2 ($t_2[r+1] = m + 1$) and we call $\mathbf{gen}(m + 1, \ell)$.

- $\phi(i_j, \sigma)$ with $j \in [\ell]$; we add $(m+1)$ on the right of t_2 ($t_2[r+1] = m+1$). We update $\sigma = \sigma \cdot \langle t_1[j], m + 1 \rangle$ and we call $\mathbf{gen}(m + 1, \ell)$.

- $\phi(i, \sigma)$ with $i \notin \{i_1, i_2, \ldots, i_\ell\}$; *i.e.*, for each $j \leq r$, we put $temp = t_2[j]$. We add $temp$ on the right of t_1 ($t_1[\ell + 1] = temp$) and we replace $t_2[j]$ by $m+1$; then we update $\sigma = \sigma \cdot \langle temp, m+1 \rangle$ and we call $\mathbf{gen}(m+1, \ell+1)$.

After each recursive call, we update t_1 and t_2 and σ in order to obtain the previous configuration before the call. These statements require a complexity in $\mathcal{O}(1)$ (*i.e.* independent of the permutation length). See Figure 1 for our generating algorithm and for some examples.
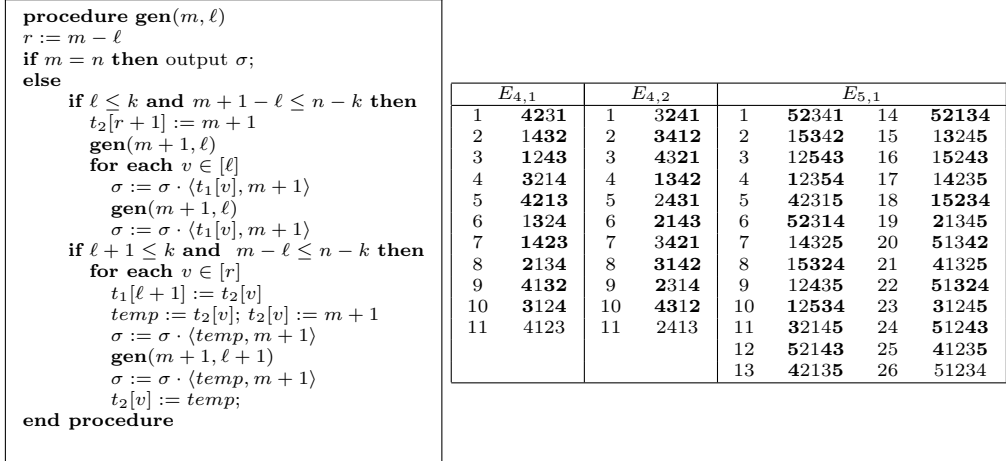
```
procedure gen(m, ℓ)
r := m − ℓ
if m = n then output σ;
else
    if ℓ ≤ k and m + 1 − ℓ ≤ n − k then
        t₂[r + 1] := m + 1
        gen(m + 1, ℓ)
        for each v ∈ [ℓ]
            σ := σ · ⟨t₁[v], m + 1⟩
            gen(m + 1, ℓ)
            σ := σ · ⟨t₁[v], m + 1⟩
    if ℓ + 1 ≤ k and m − ℓ ≤ n − k then
        for each v ∈ [r]
            t₁[ℓ + 1] := t₂[v]
            temp := t₂[v]; t₂[v] := m + 1
            σ := σ · ⟨temp, m + 1⟩
            gen(m + 1, ℓ + 1)
            σ := σ · ⟨temp, m + 1⟩
            t₂[v] := temp;
end procedure
```

$E_{4,1}$		$E_{4,2}$		$E_{5,1}$			
1	4231	1	3241	1	52341	14	52134
2	1432	2	3412	2	15342	15	13245
3	1243	3	4321	3	12543	16	15243
4	3214	4	1342	4	12354	17	14235
5	4213	5	2431	5	42315	18	15234
6	1324	6	2143	6	52314	19	21345
7	1423	7	3421	7	14325	20	51342
8	2134	8	3142	8	15324	21	41325
9	4132	9	2314	9	12435	22	51324
10	3124	10	4312	10	12534	23	31245
11	4123	11	2413	11	32145	24	51243
				12	52143	25	41235
				13	42135	26	51234

Figure 1: Generating algorithm for permutations with a fixed number of excedances and the lists $E_{4,1}$, $E_{4,2}$ and $E_{5,1}$.

2.2 Complexity

Here we do not consider the trivial case $k = 0$ ($k = n - 1$ resp.) where $E_{n,0} = \{12 \ldots n\}$ ($E_{n,n-1} = \{2 \ldots n1\}$ resp.). Let us assume $k \neq 0$ and $k \neq n-1$. Then our algorithm produces all permutations of $E_{n,k}$ in constant amortized

time (CAT); *i.e.* the amount of computation is proportional to the number of generated objects. Indeed, in our procedure the computation is proportional to the number of recursive calls performed by it, and each call generates a combinatorial object, or produces at least two calls. A Java implementation of this algorithm can be viewed at http://www.u-bourgogne.fr/jl.baril/applet.html.

3   Gray code for permutations with exactly one excedance

In this part, we will adapt the previous algorithm for generating $E_{n,1}$, $n \geq 1$, in a Gray code order. A *Gray code* is a family $\{\mathcal{L}_n\}_{n \geq 0}$ of lists of n-length sequences such that in each list the Hamming distance d between any two consecutive sequences (*i.e.* the number of positions in which they differ) is bounded by a constant independently of their length. If this constant is minimal then the code is called *optimal*.

Now we explain how one can modify the procedure of Figure 1 in order to produce a Gray code listing for $E_{n,1}$. Let us remark that the generating tree associated with the previous algorithm has the following properties:

- for each $k \leq n-1$, a permutation $\sigma \in E_{k,1}$ has two sons: $\phi(i, \sigma)$ if $\sigma(i)$ is the excedance in σ; and $\psi(\sigma)$;

- for each $k \leq n-2$, $\sigma = id_k = 12\ldots k$ has $(k+1)$ sons: $\psi(id_k) = id_{k+1}$, $\phi(1, id_k), \ldots, \phi(k-1, id_k), \phi(k, id_k)$;

- for $k = n-1$, $\sigma = id_{n-1} = 12\ldots(n-1)$ has $(n-1)$ sons: $\phi(1, id_{n-1}), \ldots, \phi(n-1, id_{n-1})$.

This means that each level $k \leq n-1$ of the generating tree contains all permutations of $E_{k,1} \cup \{id_k\}$ and the last level n contains those of $E_{n,1}$. Then for each $k \leq n-1$ and for a permutation $\sigma \in E_{k,1} \cup \{id_k\}$, we associate:

- a *direction*, *up* or *down*. A permutation σ with the direction *up* (*down* respectively) will be denoted by σ^1 (σ^0 respectively).

- a list of its sons in the generating tree considered with their directions. The list of successors of σ^0 is obtained by reversing the list of successors of σ^1 and by reversing the direction of each element of the list.

Now, let $\sigma^1 \in E_{k,1} \cup \{id_k\}$. We define the sequence L of integer where $L(i)$ denotes the ith integer of the sequence L:

$$L = \begin{cases} k-1, k-3, k-5, \ldots, 1, 2, 4, \ldots, k & \text{if } k \text{ is even,} \\ k-1, k-3, k-5, \ldots, 2, 1, 3, \ldots, k & \text{if } k \text{ is odd,} \end{cases}$$

and we distinguish two cases:

- if σ^1 has two sons where $\sigma(i)$ is an excedance, the list of its successors is
$$\phi(i,\sigma)^0, \psi(\sigma)^1.$$

- if $\sigma^1 = id_k^1$, $k \le n-1$, the list of its successors is
$$\phi(L(1),\sigma)^0, \phi(L(2),\sigma)^1, \ldots, \phi(L(k),\sigma)^{(k-1) \mod 2}, \psi(\sigma)^1,$$

Recall that for σ^0, we take the reverse list of σ^1.

So we recursively define our Gray code $\mathcal{E}_{n,1} \cup \{id_n\}$ for $E_{n,1} \cup \{id_n\}$ as follow:

$$\mathcal{E}_{n,1} = \begin{cases} 1^1 & \text{if } n=1 \\ \bigcirc_{i=1}^{e_{n-1,1}+1} \Pi(i) & \text{otherwise,} \end{cases}$$

where $\Pi(i)$ is the list of the ith permutation of the $\mathcal{E}_{n-1,1} \cup \{id_{n-1}\}$ considered with its direction, and $\bigcirc$ is the concatenation operator for lists. See Figure 2 for instance.

Theorem 1 *The list $\mathcal{E}_{n,1} \cup \{id_n\}$ defined above is an optimal Gray code for $E_{n,1} \cup \{id_n\}$. Two consecutive elements in the list differ in at most three positions. Consequently, the restricted list $\mathcal{E}_{n,1}$ is also an optimal Gray code for the set $E_{n,1}$.*

Proof. We proceed by induction on n. The property is true for $n \le 3$. We assume that it holds for $k \le n-1$ and we will prove that $\mathcal{E}_{n,1} \cup \{id_n\}$ is a Gray code.

First, we consider transitions between the successors of a permutation $\sigma \in \mathcal{E}_{n-1,1} \cup \{id_{n-1}\}$, *i.e.* the transitions of the form (i) $\phi(i,\sigma)$, $\phi(j,\sigma)$ for $1 \le |i-j| \le 2$, or (ii) $\phi(i,\sigma)$, $\psi(\sigma)$, with $1 \le i,j \le n-1$. For the transition (i), we easily have $\phi(i,\sigma) = \phi(j,\sigma) \cdot \langle j,i,n \rangle$. For (ii), we have $\phi(i,\sigma) = \psi(\sigma) \cdot \langle i,n \rangle$. Thus two successive permutations in $\mathcal{E}_{n,1} \cup \{id_n\}$, that have the same predecessors in $\mathcal{E}_{n-1,1} \cup \{id_{n-1}\}$, differ in at most three positions.

Now we assume that σ and τ are two successive permutations in $\mathcal{E}_{n,1} \cup \{id_n\}$ that do not have the same predecessor in $\mathcal{E}_{n-1,1} \cup \{id_{n-1}\}$. Let $\sigma_1 \in \mathcal{E}_{n-1,1} \cup \{id_{n-1}\}$ (respectively $\tau_1 \in \mathcal{E}_{n-1,1} \cup \{id_{n-1}\}$) be the predecessor of σ (respectively τ). Via the recurrence hypothesis, σ_1 and τ_1 differ by at most three positions. We distinsguish three cases:

- (a) If σ_1 has the direction 1 (*up*), and τ_1 the direction 0 (*down*), then $\sigma = \psi(\sigma_1)$ and $\tau = \psi(\tau_1)$ which implies that σ and τ differ by at most three positions.

- (b) If σ_1 has the direction 0, and τ_1 the direction 1, then they necessarily have the same predecessor $\gamma \in \mathcal{E}_{n-2,1} \cup \{id_{n-2}\}$.
If $\gamma = id_{n-2}$ then $\sigma_1 = \phi(i, id_{n-2})$, $\tau_1 = \phi(j, id_{n-2})$ with $1 \le |i-j| \le 2$ and $1 \le i,j \le n-2$; or $\sigma_1 = \phi(n-2, id_{n-2})$ and $\tau_1 = \psi(id_{n-2}) = id_{n-1}$. Therefore, for the first case, $\sigma = \phi(i, \sigma_1)$, $\tau = \phi(j, \tau_1)$ and $\tau = \sigma \cdot \langle j,i,n-1 \rangle$; and for the second case, we have $\sigma = \phi(n-2, \sigma_1)$ and $\tau = \phi(n-2, id_{n-1})$ which means $\tau = \sigma \cdot \langle n-1, n \rangle$. Thus σ and τ differ in at most three positions.

6

If $\gamma \neq id_{n-2}$ then we obtain $\sigma_1 = \phi(i, \gamma)$ and $\tau_1 = \psi(\gamma)$ (or conversely) where $\gamma(i)$ is the only one excedance of γ. Thus we have $\sigma = \phi(i, \sigma_1)$ and $\tau = \phi(i, \tau_1)$, which implies that $\tau = \sigma \cdot \langle n-1, n \rangle$. Thus σ and τ differ in at most three positions.

- (c) If σ_1 has the direction 1, and τ_1 the direction 1, then they necessarily have the same predecessor $id_{n-2} \in \mathcal{E}_{n-2,1} \cup \{id_{n-2}\}$. Thus, $\sigma_1 = \phi(n-2, id_{n-2})$, $\tau_1 = id_{n-1}$ and we have $\sigma = \phi(n-2, \sigma_1)$ and $\tau = \phi(n-2, id_{n-1})$ which also prove that σ and τ differ in at most three positions.

For the optimality, it suffices (for instance) to remark for instance that $\mathcal{E}_{4,1} \cup \{id_4\}$ contains seven odd permutations and five even permutations. This allows us to prove that we can not obtain Gray code for $E_{4,1} \cup \{id_4\}$ such that two consecutive permutations differ by two positions. $\qquad \square$

Notice that, since id_n is the last permutation of the Gray code $\mathcal{E}_{n,1} \cup \{id_n\}$, the restricted list $\mathcal{E}_{n,1}$ is also an optimal Gray code. Figure 2 illustrates the generating tree for $\mathcal{E}_{4,1}$.
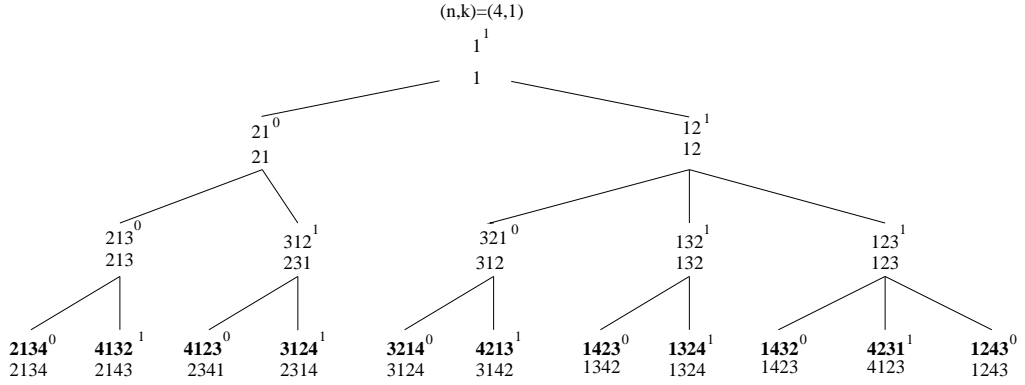

Figure 2: The generating tree for the Gray codes $\mathcal{E}_{4,1}$. Each permutation of $\mathcal{E}_{4,1}$ with its direction, lies on the first line of the leaves (in boldface). The second line generates permutations in $S_4(321, 2413, 3412, 21534) \backslash \{1234\}$ studied in Section 4.

Remark 1 *Notice that we can obtain the generating tree of $E_{n,1} \cup \{id_n\}$ using the method ECO [2, 3]. Indeed, the following succession rules*

$$\begin{cases} (2) & root \\ (k) & \to (2_1)^{k-1}(k+1) \\ (2_1) & \to (2_1)^2 \end{cases}$$

also provide $E_{n,1} \cup \{id_n\}$. *Here, the rules for obtaining a successor of a permutation* $\sigma \in E_{k,1}$ *is not the classical insertion between two entries of* σ; *but we*

obtain the successor of σ by appending $(k+1)$ at the end of σ and by a product (on the right) of a transposition $\langle i, k+1 \rangle$ for some $i \in [k+1]$.

This rule does not satisfy the *stability property* introduced in [6]. However, the rule is almost *stable* since the label of the root is the only one that does not produce two labels (2). Consequently, the method in [6] allows to construct a Gray code in a particular representation of $E_{n,1} \cup \{id_n\}$ (which is not the classical representation of a permutation).

4   A bijection between $E_{n,1} \cup \{12 \dots n\}$ and $S_n(T)$ with $T = \{321, 2413, 3412, 21534\}$

In this part we give a correspondence between $E_{n,1} \cup \{12 \dots n\}$ and a set of avoiding permutations $S_n(321, 2413, 3412, 21534)$. A permutation $\sigma = \sigma_1 \sigma_2 \dots \sigma_n \in S_n$ contains the pattern $\gamma \in S_k$ ($k \geq 2$) if and only if a sequence of indices $1 \leq i_1 < i_2 < \dots < i_k \leq n$ exists such that $\sigma_{i_1} \sigma_{i_2} \dots \sigma_{i_k}$ is ordered as γ. We denote by $S_n(\gamma)$ the set of permutations of S_n avoiding the pattern γ. For example, **234**1 $\notin S_4(123)$ but $4312 \in S_4(123)$.

Now, we define the function ϕ' below:

Definition 2 *For an integer $i \in [n-1]$ and a permutation $\gamma \in S_{n-1}$, we define an n-length permutation $\sigma = \phi'_{n-1}(i, \gamma)$ by*

$$
\sigma(j) = \left\{
\begin{array}{ll}
n & \text{if } j = i \\
\gamma(j) & \text{if } j \leq i-1 \\
\gamma(j-1) & \text{otherwise.}
\end{array}
\right.
$$

Thus the permutation $\phi'_{n-1}(i, \gamma)$ is obtained from γ by inserting n just before $\gamma(i)$.

For example, $\phi'_5(3, 52431) = 52\mathbf{6}431$. As for the function ϕ, we will omit the index in ϕ'_n.

Theorem 2 *If we replace ϕ by ϕ' in the definition of our Gray code $\mathcal{E}_{n,1} \cup \{id_n\}$, we also obtain an optimal Gray code for the set $S_n(321, 2413, 3412, 21534)$ such that two successive permutations differ in at most three positions.*

Proof. Let us prove that our generating tree considered with ϕ', creates all permutations of the set $S_n(321, 2413, 3412, 21534)$. We proceed by induction on n. This holds for $n \leq 5$. Let us assume that the result is true for $k \leq n$. Let σ be a permutation on the level $n+1$. Therefore, σ is obtained from a permutation τ in $S_n(321, 2413, 3412, 21534)$. If $\tau = id_n$ then $\sigma = \phi'(i, id_n)$ with $1 \leq i \leq n$, or $\sigma = \psi(id_n) = id_{n+1}$, which implies that σ belongs to the set $S_{n+1}(321, 2413, 3412, 21534)$. If $\tau \neq id_n$ then $\sigma = \phi'(n, \tau)$ or $\sigma = \psi(\tau)$, and σ is still in the set $S_{n+1}(321, 2413, 3412, 21534)$.

Conversely, let us prove that each permutation $\sigma \in S_{n+1}(321, 2413, 3412, 21534)$ belongs to the level $n+1$ of the generating tree. If we have $\sigma(n) = n+1$ or

$\sigma(n+1) = n+1$ then there exists $\tau \in S_n(321, 2413, 3412, 21534)$ such that, either $\sigma = \phi'(n, \tau)$ or $\sigma = \psi(\tau)$. This proves that σ belongs to the level $n+1$ of the generating tree. For the other cases, there exists $i \leq n-1$, such that $\sigma(i) = n+1$. Since σ avoids 321, we necessarily have $\sigma(i+1) < \sigma(i+2) < \ldots < \sigma(n+1)$. Since σ avoids 2413 and 3412, we necessarily have $\sigma(j) < \sigma(i+1)$ for $j \leq i-1$. Since σ avoids 21534 then $\sigma(1) < \sigma(2) < \ldots < \sigma(i-1)$. For all these reasons, we obtain $\sigma = 123\ldots(i-1)(n+1)i(i+1)\ldots n = \phi(i, id_n)$ which induces that σ belongs to the generating tree. This means that the generating tree with ϕ' allows us to generates all permutations in the set $S_n(321, 2413, 3412, 21534)$. It also provides a Gray code for the set $S_n(321, 2413, 3412, 21534)$ such that two successive permutations differ in at most three positions. This can be proved in the same way as Theorem 1. $\qquad\square$

This theorem induces a constructive bijection between $E_{n,1} \cup \{id_n\}$ and $S_n(321, 2413, 3412, 21534)$.

Corollary 1 *Let $\sigma \in E_{k-1,1} \cup \{id_{k-1}\}$, we recursively define the maps f_k from $E_{k,1} \cup \{id_k\}$ to $S_k(321, 2413, 3412, 21534)$, $1 \leq k \leq n$ by:*

- $f_1(id_1) = id_1$,

- $f_k(\psi(\sigma)) = \psi(f_{k-1}(\sigma))$,

- $f_k(\phi(i, id_{k-1})) = \phi'(i, id_{k-1})$, *for $k \leq n$,*

- $f_k(\phi(i, \gamma)) = \phi'(k-1, f_{k-1}(\gamma))$, *if $\gamma \neq id_{k-1}$.*

Then f_k is a bijection such that $f_k(id_k) = id_k$. Moreover, $f_k(\sigma)$ can be obtained from σ in linear time.

Proof. The fact that f_k is a bijection from $E_{k,1} \cup \{id_k\}$ to $S_k(321, 2413, 3412, 21534)$ is directly deduced from Theorem 2. Now let us prove that $f_k(\sigma)$ can be obtained from σ in linear time. Indeed, for the trivial case where $\sigma(k) = id_k$ then $f_k(\sigma) = \sigma$ and there is nothing to do. Now let us assume that $\sigma \neq id_k$ and $\gamma \in E_{k-1,1} \cup \{id_{k-1}\}$ be its predecessor in the generating tree. Therefore, if $\sigma(k) = k$ then $f_k(\sigma) = \psi(f_{k-1}(\gamma))$. Now let us consider $\sigma(k) \neq k$. Thus, there exists i, $1 \leq i \leq k-1$ such that $\sigma(i) = k$ is the only one excedance of σ and we have $\gamma(i) = \sigma(k)$. We distinguish two cases. (a) If $\sigma(k) = i$ then we necessarily have $\gamma = id_{k-1}$ and $f_k(\sigma) = \phi'(i, id_{k-1})$. (b) If $\sigma(k) \neq i$ then $\gamma \neq id_{k-1}$ and $f_k(\sigma(k)) = \phi'(k-1, f_{k-1}(\gamma))$. The recursivity is stopped when γ reaches an identity permutation, *i.e.*, when we reach the trivial case explained above. This means that the case (a) is performed only one time in order to construct $f_k(\sigma)$; for the other levels of the recursivity, we update the current permutation by appending a value on its right and eventually by a product on the right of a transposition. This proves that $f_k(\sigma)$ can be obtained in linear time. $\qquad\square$

For example, $f_4(3214) = f_4(\psi(321)) = \psi(f_3(321)) = \psi(f_3(\phi(1, 12))) = \psi(\phi'(1, 12)) = \psi(312) = 3124$.

Remark 2 *The list $f_n(\mathcal{E}_{n,1})$ is also an optimal Gray code for the set of avoiding permutations $S_n(321, 2413, 3412, 21534) \backslash \{id_n\}$. Two consecutive elements*

9

in the list differ in at most three positions. Notice that despite this last result, this bijection is not an isomorphism in the sense where the images of two close elements are close. Indeed, if d is the Hamming distance, we have for instance $d(12345, 52143) = 3$ and $d(f_5(12345), f_5(52143)) = d(12345, 31254) = 5$. See Table 1 for an illustration of our bijection onto the Gray code $\mathcal{E}_{n,1}$.

Now, let us consider the map ϕ'' defined as follow:

Definition 3 *For an integer $i \in [n-1]$ and a permutation $\gamma \in S_{n-1}$, we define an n-length permutation $\sigma = \phi''_{n-1}(i, \gamma)$ by*

$$\sigma(j) = \begin{cases} n & if\ j = \gamma^{-1}(i) \\ \gamma(j) & if\ j \le \gamma^{-1}(i) - 1 \\ \gamma(j-1) & otherwise. \end{cases}$$

Thus the permutation $\phi''_{n-1}(i, \gamma)$ is obtained from γ by inserting n just before i.

For example, $\phi''_5(3, 52431) = 524\mathbf{6}31$.

Remark 3 *If we replace ϕ by ϕ'' in the definition of our Gray code $\mathcal{E}_{n,1} \cup \{id_n\}$, we also obtain a generating tree for the set $S_n(231, 4132, 4213, 21534)$ which can not be directly implemented in a CAT algorithm. Notice that the generating tree obtained does not list the set $S_n(231, 4132, 4213, 21534)$ in Gray code order. We do not provide the proof of this remark since it can be obtained mutatis mutandis from this one of Theorem 2.*

Table 1: The Gray codes $\mathcal{E}_{4,1}$, $\mathcal{E}_{5,1}$, and their images by the bijection of Corollary 1: $S_4(T)\setminus\{1234\}$ and $S_5(T)\setminus\{12345\}$ where $T = \{321, 2413, 3412, 21534\}$.

$\mathcal{E}_{4,1}$		$\mathcal{E}_{5,1}$				$S_4(T)\setminus\{Id_4\}$		$S_5(T)\setminus\{Id_5\}$			
1	**2**134	1	**21345**	14	15**234**	1	**2134**	1	**21345**	14	13452
2	4**132**	2	51**342**	15	15**243**	2	**2143**	2	21354	15	13254
3	4123	3	51**324**	16	1**3245**	3	**2341**	3	21**453**	16	1**3245**
4	3**124**	4	41**325**	17	14**325**	4	**2314**	4	21**435**	17	14**235**
5	3214	5	41**235**	18	15**324**	5	3124	5	23**415**	18	14**253**
6	4213	6	51**234**	19	5**2314**	6	**3142**	6	234**51**	19	4**1253**
7	1**423**	7	51**243**	20	4**2315**	7	1**342**	7	23**154**	20	**41235**
8	1**324**	8	31245	21	1**2435**	8	1324	8	**23**145	21	1**2435**
9	1**432**	9	**32145**	22	1**2534**	9	1**423**	9	31245	22	1**2453**
10	4231	10	52143	23	1**2543**	10	**4123**	10	31254	23	1**2534**
11	1243	11	52134	24	**52341**	11	1243	11	31**452**	24	**51234**
		12	**42**135	25	15**342**			12	**31**425	25	15**234**
		13	14235	26	12354			13	134**25**	26	12354

5 Final remarks

In this paper, we give an efficient (CAT) algorithm for generating permutations with a fixed number of excedances. An improvement of this algorithm allows us to obtain optimal Gray code for permutations with exactly

one excedance. We provide a constructive bijection between $E_{n,1}$ and the set $S_n(321, 2413, 3412, 21534)\backslash\{12\ldots(n-1)n\}$ which induces an optimal Gray code for $S_n(321, 2413, 3412, 21534)\backslash\{12\ldots(n-1)n\}$. Moreover we remark that there is a one-to-one map from $S_n(321, 2413, 3412, 21534)$ to $S_n(231, 4132, 4213, 21534)$ that does not induces a Gray code for the second set. Can one find a Gray code for the set $S_n(231, 4132, 4213, 21534)$? More generally, can one find a Gray code for the set $E_{n,k}$ for $2 \leq k \leq n-1$? Is it possible to obtain a set of avoiding permutations which is in bijection with $E_{n,k}$ for $2 \leq k \leq n-1$? Eulerian numbers also enumerate permutations of length n with a fixed number of descents; can one develop an efficient (CAT) algorithm for generating these objects?

References

[1] S.G. Akl. A new algorithm for generating derangements. *BIT*, 20: 2–7, 1980.

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

[3] 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.

[4] J.L. Baril. Gray code for permutations with a fixed number of cycles. *Discrete Math.*, 307, 13(6): 1559–1571, 2007.

[5] J.L. Baril and P.T. Do. ECO-generation for p-generalized Fibonacci and Lucas permutations. *Journal of Pure Mathematics and Applications*, 17(1-2): 19–37, 2006.

[6] A. Bernini, E. Grazzini, E. Pergola and R. Pinzani. A general exhaustive generation algorithm for Gray structures. *Acta Inform.*, 44(5): 361–376, 2007.

[7] J.L. Baril and V. Vajnovszki. Gray code for derangements. *Discrete Appl. Math.*, 140(1-3): 207–221, 2004.

[8] B. Bauslaugh and F. Ruskey. Generating alternating permutations lexicographically. *BIT*, 30: 17–26, 1990.

[9] M. Bóna. Combinatorics of permutations. *Chapman & Hall/CRC*, 2004.

[10] S. Effler and F. Ruskey. A CAT algorithm for generating permutations with a fixed number of inversions. *Information Process. Letters*, 86: 107–112, 2003.

[11] R. Ehrenborg and E. Steingrímsson. The excedance set of a permutation *Advances in Applied Mathematics* 24:284–299, 2000.

[12] D. Foata and M.P. Schützenberger. Théorie géométrique des polynômes Eulériens. *Lectures Notes in Mathematics*, 138, Springer-Verlag, Berlin, 1970.

[13] S.M. Johnson. Generation of permutations by adjacent transpositions. *Mathematics of computation*, 17: 282–285, September 1963.

[14] D.E. Knuth. *The art of computer programming, combinatorial algorithms*, volume 4. Pre-fascicle 2b, (Generating all permutations), 2002.

[15] J.F. Korsh. Loopless generation of up-down permutations. *Discrete Math.*, 240(1-3): 97–122, 2001.

[16] J.F. Korsh. Constant time generation of derangements. *Information Process. Letters*, 90(4): 181–186, 2004.

[17] M. Lothaire. Combinatorics on words. *Encyclopedia of Mathematics and its applications*, 17, Addison-Wesley, 1983.

[18] R. Mantaci and F. Rakotondrajao. A permutation representation that knows what ”Eulerian” means. *Discrete Math. and Theo. Comput. Science*, 4: 101–108, 2001.

[19] J. Riordan. An introduction to combinatorial analysis. *New York, J. Wiley*, 1958.

[20] D. Roelants van Baronaigien. Constant time generation of involutions. *Congressus Numerantium*, 90: 87–96, 1992.

[21] F. Ruskey and U. Taylor. Fast generation of restricted classes of permutations. *Manuscript*, 1995.

[22] F. Ruskey and D. Roelants van Baronaigien. Generating permutations with given ups and downs. *Discrete Appl. Math.*, 36(1): 57–67, 1992.

[23] N.J.A Sloane. *The Encyclopedia of Integer Sequences*, http://www.research.att.com/~njas/sequences.

[24] H.F. Trotter. *PERM (Algorithm 115)*. Communications of ACM, 5(8), 434–435, 1962.

[25] V. Vajnovszki. A loopless algorithm for generating the permutations of a multiset. *Theoretical Computer Science*, 307: 415–431, 2003.

[26] T. Walsh. Gray code for involutions. *J. Combin. Math. Combin. Comput.*, 36: 95–118, 2001.