

BOLTZMANN SAMPLERS FOR COLORED COMBINATORIAL OBJECTS

O. BODINI, A. JACQUOT

LIP6, UMR 7606, Département CALSCI, Université Paris 6 - UPMC,
104, avenue du président Kennedy,
F-75252 Paris cedex 05, France
{Olivier.Bodini, Alice.Jacquot}@lip6.fr

ABSTRACT. In this paper, we give a general framework for the Boltzmann generation of colored objects belonging to combinatorial constructible classes. We propose an intuitive notion called *profiled objects* which allows the sampling of size-colored objects (and also of k -colored objects) although the corresponding class cannot be described by an analytic ordinary generating function.

Efficient generation of extremely large objects is needed in many situations. For instance in statistical physics for observing limit behaviors [2], in biology for understanding and analysing genome properties [12], or in computer sciences for testing programs [6] or simulating and modeling Network as Internet [7, 3]. In 2003, Duchon, Flajolet, Louchard and Schaeffer [5] proposed a new model, called Boltzmann model, which leads to systematically construct samplers for random generation of objects in combinatorial constructible classes. These samplers depends on a real parameter x and generate an object a in a combinatorial constructible class \mathcal{A} with a probability essentially proportional to $x^{|a|}$ where $|a|$ is the size of a . Hence they draw uniformly in the class \mathcal{A}_n of all the objects of size n in \mathcal{A} . The size of the output is a random variable, and parameter x can be tuned for a targetted mean value. Moreover using rejection, one can obtain exact size samplers or approximate size samplers. This new approach differs from the "recursive method" introduced by Nijenhuis and Wilf [11] by giving the possibility of relaxing the constraint of an exact size for the output and this implies a significant gain in complexity: no preprocessing phase is needed and expected time complexity is linear in the size of the output.

Boltzmann model has been described both in the cases of unlabelled combinatorial constructible (also called specifiable or decomposable) classes [5] and labelled combinatorial constructible classes [5, 8], for the most classical constructions (+, \times , Seq, Set, Cyc,...). In this paper, we are interested in the generation under Boltzmann model of size-colored combinatorial objects. Let a be a combinatorial object with n atoms, we say that this object is *size-colored or colored* if each of its atoms can be colored with a color in $\{1, \dots, n\}$. Our main motivation for this study stems from the following situation : Consider that the construction of an object of size n is distributed to n heterogenous processors. Each processors signs the atoms that it has build. A size-colored object is exactly a possible way to build such an object and we would like to highlight some properties about it. In order to do that, we are going to deal with well-known k -colored combinatorial object (in this case, each atom can be colored with a color in $\{1, \dots, k\}$, and k does not depend on the size). k -colored classes can be found in numerous problems as the k -colored necklaces [10], expression trees with k types of n -ary functions, k -colored planar trees, k -colored Motzkin paths [13],...

For instance, all the 2-colored non-planar general trees of size 3 are :

$$\begin{array}{cccccccccccccccc}
 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 \\
 | & | & | & | & | & | & | & | & / \backslash \\
 1 & 1 & 2 & 2 & 1 & 1 & 2 & 2 & 1 & 1 & 1 & 2 & 2 & 1 & 1 & 1 & 2 & 2 \\
 | & | & | & | & | & | & | & | & | & | & | & | & | & | & | & | & | & | \\
 1 & 2 & 1 & 2 & 1 & 2 & 1 & 2 & & & & & & & & & & &
 \end{array}$$

Each object has to be drawn with the same probability $\frac{1}{14}$. This example points out that symmetries increase the difficulty to build an uniform sampler. That can be also observed by comparing Boltzmann sampler codes for labelled and unlabelled structures. In a naive approach, we could deal with in the classical Boltzmann model by duplicating k times atoms, but we give here a more general and efficient point of view. In particular, we obtain a relatively compressed storage that we explain in a later section and we can postpone the draw of the colors to avoid the unnecessary coloration during the rejection phase.

A *profiled object* is by definition an object associated with a set-partition of its atoms. This concept is derived from Polya theory [1]. We prove that this structure is lifted to all k -colorations. Roughly speaking, the atoms belonging the same partition-class could be interpreted as having the same colors. More precisely, for every k , a sampler for profiled objects can be specialised (by a good choice of the parameters) to become a sampler for any k -colored objects, so that, in a sense, our model unifies all k -colorations. Espacially, profiled object samplers can be adapted to efficient sampling of colored object in approximative size. Profiles enable us to generate colored objects with Boltzmann methods even if the ordinary generating function is not analytic in 0 (which is an obstruction in classical Boltzmann theory). For instance, let \mathcal{A} be the unlabelled constructible class $\text{Seq}(\mathcal{Z})$, it is well known that the ordinary generating function is $1/(1-z)$. In this case, the ordinary generating function for the class $\tilde{\mathcal{A}}$ of colored objects is $\sum_{k \geq 0} n^k z^n$ which is clearly not analytic in 0. At first sight, it seems impossible to have a Boltzmann-type sampler for this class, since the parameter x has to be in the disk of convergence of the generating function.

This paper is organised in four sections. The first section is devoted to the definitions of various taggings for combinatorial classes. We define a general framework, in terms of language, that allows to describe unlabelled and labelled classes as well as less studied classes such as semilabelled or colored classes. We also recall the notion of constructible classes which is central for Boltzmann sampler theory. The second section addresses the notion of profiled objects. We define the generating function associated to a profiled combinatorial class and express its relation with the generating functions of the associated k -colored classes.

In the third section, we construct Boltzmann samplers for the profiled combinatorial constructible classes and we show how to use it to obtain samplers for the k -colored combinatorial constructible classes.

In the last section, we propose an approximate size sampler for the colored combinatorial constructible classes. This sampler is based on a filter that allows to transform a sampler for n -colored objects into a sampler for colored objects.

1. COMBINATORIAL CLASSES

Combinatorial classes are very well studied and classical objects [9]. We propose in this section a general framework, in terms of language, that allows to describe extended tagged combinatorial classes. Our approach can be seen as an introduction to a very simplified species theory [1]. Let \mathfrak{A} be the alphabet on the 7 following letters $\square\{\}\square()$. The square is called the *atom*. The other letters correspond to three different types of parentheses.

We denote by \mathcal{L} the language on \mathfrak{A} defined as follows :

- $\square, \{\}, (), []$ belong to \mathcal{L}
- $\forall k \in \mathbb{N}^*, A_1, \dots, A_k \in \mathcal{L} \Rightarrow (A_1 A_2 \dots A_k) \in \mathcal{L}$, the *sequences*
- $\forall k \in \mathbb{N}^*, A_1, \dots, A_k \in \mathcal{L} \Rightarrow [A_1 A_2 \dots A_k] \in \mathcal{L}$, the *cyclic sequences*
- $\forall k \in \mathbb{N}^*, A_1, \dots, A_k \in \mathcal{L} \Rightarrow \{A_1 A_2 \dots A_k\} \in \mathcal{L}$, the *multisets*

The elements of \mathcal{L} are called the *combinatorial proto-objects*.

The *size* of a proto-object A is the number of atoms contained in A . That is to say, the number of occurrence of \square in A . For instance, $(\{\square\square\}\square)$ is a combinatorial proto-object of size 5.

Let A be a proto-object, we denote $atom(A)$ the set of the occurrences of \square in A . In particular, $|atom(A)|$ is the size of A and each element of $atom(A)$ corresponds exactly to one occurrence of \square in A .

At this stage, we just have a support but we need to explain the meaning of the cycles and the multisets by some equivalences between proto-objects. In order to do that, let B be a set, a *B-colored proto-object* is a pair $\langle A, f \rangle$ where A is a word of \mathcal{L} and f is a function from $atom(A)$ to B . Let a be an atom of A , $f(a)$ is called the *color* of a . We denote by \mathcal{L}_B the set of all B -colored proto-objects.

We define the equivalence \equiv_B between B -colored proto-objects of \mathcal{L}_B as follows :

- Two B -colored squares are equivalent if and only if they are the same color :

$$\langle \square, f \rangle \equiv_B \langle \square, f' \rangle \Leftrightarrow f(\square) = f'(\square)$$

- Two sequences are equivalent if their corresponding substructures are equivalent :

$$\forall A_1, \dots, A_k, B_1, \dots, B_k \in \mathcal{L}, \langle (A_1 A_2 \dots A_k), f \rangle \equiv_B \langle (B_1 B_2 \dots B_k), f' \rangle$$

$$\Leftrightarrow$$

$$\forall i, f(Atom(A_i)) = f'(Atom(B_i)) \text{ and } \langle A_i, f|_{Atom(A_i)} \rangle \equiv_B \langle B_i, f'|_{Atom(B_i)} \rangle$$

- Two cycles are equivalent if their substructures are equivalent up to a circular shift :

$$\forall A_0, \dots, A_{k-1}, B_0, \dots, B_{k-1} \in \mathcal{L}, \langle [A_0 A_1 \dots A_{k-1}], f \rangle \equiv_B \langle [B_0 \dots B_{k-1}], f' \rangle$$

$$\Leftrightarrow$$

$$\exists n \forall i, f(Atom(A_i)) = f'(Atom(B_{i+n \bmod k}))$$

and

$$\langle A_i, f|_{Atom(A_i)} \rangle \equiv_B \langle B_{i+n \bmod k}, f'|_{Atom(B_{i+n \bmod k})} \rangle$$

- Two sets are equivalent if their substructures are equivalent up to permutations (we denote by S_k the symmetric group of order k) :

$$\forall A_0, \dots, A_{k-1}, B_0, \dots, B_{k-1} \in \mathcal{L}, \langle \{A_0 \dots A_{k-1}\}, f \rangle \equiv_B \langle \{B_0 \dots B_{k-1}\}, f' \rangle$$

$$\Leftrightarrow$$

$$\exists \sigma \in S_k, \forall i, f(Atom(A_i)) = f'(Atom(B_{\sigma(i)}))$$

and

$$\langle A_i, f|_{Atom(A_i)} \rangle \equiv_B \langle B_{\sigma(i)}, f'|_{Atom(B_{\sigma(i)})} \rangle$$

The set \mathcal{O}_B of B -colored combinatorial objects is \mathcal{L}_B / \equiv_B . Now, we can define the fundamental notion of combinatorial classes as follows :

Definition 1.1. A B -colored combinatorial class is a sub-multiset of \mathcal{L}_B / \equiv_B with a finite number of object of each size.

We opt in this definition for a multiset version which avoid the traditional problem of the disjoint union. The union $A \cup B$ of two multisets is the multiset obtained with all the existing elements and where the multiplicity of a in $A \cup B$ is the sum of the multiplicity a in A and the multiplicity of a in B .

The *size* of a B -colored combinatorial object A is the size of any proto-object which represents it.

Every object of size 0 is called *neutral object*. For instance, $\{()\}()$ is a neutral object. Clearly, \mathcal{O}_B contains an infinite number of neutral objects. we carefully avoid the confusion between the empty combinatorial class \emptyset and a combinatorial class reduced to a neutral element ϵ .

For example, the following $\{1, 2\}$ -colored combinatorial class \mathcal{C} contains 3 objects. The both objects of size 3 are identical.

$$\mathcal{C} = \left\{ \left(\boxed{1} \right), \left\{ \left[\boxed{2} \boxed{1} \right] \boxed{1} \right\}, \left\{ \boxed{1} \left[\boxed{1} \boxed{2} \right] \right\} \right\}$$

Now, we can add some restrictions on f to obtain different types of *tagged* combinatorial objects. Some of them are classical (unlabelled, labelled), the other ones are more unusual but reasonable.

Function	Codomain	Labelling
one-to-one map	$\{1, \dots, n\}$ for an object of size n	labelled
constant map	for instance $\{1\}$	unlabelled
quasi-constant map	$\{0, 1\}$ ($\exists! y \forall x \neq y f(x) = 1$ and $f(y) = 0$)	pointed
surjective map	$\{1, \dots, k\}$ with $k \leq n$	semi-labelled
arbitrary map	$\{1, \dots, n\}$ for an object of size n	colored
arbitrary map	$\{1, \dots, k\}$ with k fixed	k -colored

FIGURE 1. Different labellings.

For each of these tagged objects, we can easily define the notion of combinatorial classes.

We essentially study in this paper the notion of colored and k -colored classes. The semilabelled classes are the topic of another work.

Let \mathcal{A} be a combinatorial (unlabelled) class, we denote by \mathcal{A}_n the set of objects of size n .

1.1. Constructible combinatorial classes. A *constructible combinatorial class* $\mathcal{A}(\mathcal{Z})$ is a class which can be build from neutral classes and an atomic class \mathcal{Z} by using some of these builders ($+, \times, Seq, MSet, PSet, Cyc, \dots$). (the builders $Seq, MSet, PSet, Cyc$ cannot be applied to classes having neutral objects.) For instance, $MSet(\mathcal{Z} \times Seq(\mathcal{Z}))$. We also accept the recursive construction : $\mathcal{T} = f(\mathcal{T})$.

These constructions are exactly the ones that we recurrently find in modern theories of combinatorial analysis. We do not develop here the precise grammar of building. This can be found for instance in [9].

A *colored constructible combinatorial class* (resp. *k -colored constructible combinatorial class*) is build from a constructible combinatorial class $\mathcal{A}(\mathcal{Z})$ as follows :

$\tilde{\mathcal{A}} = \bigcup_{n=0}^{\infty} \mathcal{A}_n(\mathcal{Z}_1 + \dots + \mathcal{Z}_n)$ (resp. $\mathcal{A}^{[k]} = \mathcal{A}(\mathcal{Z}_1 + \dots + \mathcal{Z}_k)$). The unique atom of the atomic class \mathcal{Z}_k corresponds to an atom colored with the color k . Strictly speaking, $\tilde{\mathcal{A}}$ and $\mathcal{A}^{[k]}$ are not combinatorial tagged classes as defined in the previous section. But they can be obviously identify by expressing the colors in terms of function on the atoms.

In the sequel, we denote identically the colored constructible combinatorial class and its associated unlabelled combinatorial class. For instance, we could speak about the colored constructible combinatorial class $MSet(\mathcal{Z} \times Seq(\mathcal{Z}))$ to say that we consider the colored constructible combinatorial class build from the unlabelled constructible class $MSet(\mathcal{Z} \times Seq(\mathcal{Z}))$.

2. PROFILED COMBINATORIAL CLASSES

Let A be a set, a subset S of $\mathcal{P}(A)$ (the set of all parts of A) is called a *set-partition* of A if and only if the elements of S are disjoint, S does not contain \emptyset unless that $A = \emptyset$ and the union of elements of S are equal to A .

A *profiled combinatorial object* $\langle a, \sigma_a \rangle$ is a combinatorial object a equipped with a set-partition σ_a of $atom(a)$ which represents its *profile* or *symmetry*.

Remark : In the classical Polya theory [1], we deal with an automorphism s of $atom(a)$ but this is overdimensioned according to what we need. We only want to mark the undistinguished atoms, a partition is clearly sufficient.

Definition 2.1. *A profiled combinatorial class is a multiset of profiled combinatorial objects with a finite number of object of each size.*

Now, we can define the product and the diagonal of profiled combinatorial objects as follows :

Product (concatenation). We define the symmetry σ_m of $m = b \times c$ as $\sigma_b \cup \sigma_c$. In other words, $\langle b \times c, \sigma_b \cup \sigma_c \rangle$.

Diagonal. We define $\delta_k a = a \dots a$ where a is repeated k times.

The symmetry of $\delta_k a$ is defined as follows : Let $\{P_1, \dots, P_m\}$ be the symmetry of a then the symmetry of $\delta_k a$ is $\{\bigcup_i P_1^i, \dots, \bigcup_i P_m^i\}$ where P_i^i corresponds to P_i but on the atoms of the i -th occurrence of a . For instance, let $\langle a, \sigma_a \rangle = \langle (\{\{\square\square\}(\square\square)\}\square), \{\{1, 3\}, \{2, 5\}, \{4\}\} \rangle$ (the first \square is linked to the third $\square \dots$), then $\delta_2 \langle a, \sigma_a \rangle$ is $\langle aa, \{\{1, 3\} \cup \{6, 8\}, \{2, 5\} \cup \{7, 10\}, \{4\} \cup \{9\}\} \rangle$

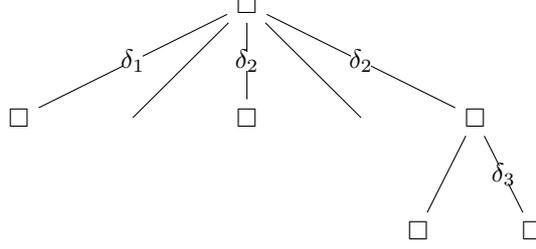
A coloration of a profiled object of size n and symmetry σ is *consistent* if on each subset of σ every atom has the same color. We also say that a profile is *c-admissible* for a coloration c if each part of σ is monochrom.

2.1. Constructible profiled combinatorial classes. the constructible profiled combinatorial classes are exactly the classes that we can build with the following list of operators :

- **Atomic class** : $\mathcal{Z} = \{\square\}$, the set-partition is reduce to the singleton.
- **Union** : $\mathcal{A} \cup \mathcal{B}$ is the multiset-union of the classes.
- **Product** : $\mathcal{A} \times \mathcal{B} = \{\langle a \times b, \sigma_a \cup \sigma_b \rangle; a \in \mathcal{A}, b \in \mathcal{B}\}$
- **Sequence** : $Seq(\mathcal{A}) = \bigcup_{i=0}^{\infty} \mathcal{A}^i$
- **Diagonal** : $\Delta_k(\mathcal{A}) = \{\delta_k a; a \in \mathcal{A}\}$
- **MultiSet** :
 $MSet(\mathcal{A}) = \{\{B\}; B \text{ a concatenation of diagonals of elements in } \mathcal{A}\}$
- **Cycle** :
 $Cyc(\mathcal{A}) = \{[B]; B \text{ a diagonal of concatenated elements of } \mathcal{A}\}$

A classical tool associated to combinatorial classes is the notion of generating functions. We explain below the building rules for the generating multivariate functions on the infinite many variables s_1, s_2, \dots associated to constructible profiled combinatorial classes. The variable s_i are going to express a set of i linked atoms.

For example, this profiled general non-planar tree of size 14 :



has $s_1^2.s_2^3.s_6$ as associated monomial.

In fact, the multivariate generating function associated to constructible profiled combinatorial classes can be defined inductively as follows :

- **Neutral class** : $F_{\mathcal{E}} = 1$
- **Atomic class** : $F_{\mathcal{Z}} = s_1$
- **Sum** : $F_{\mathcal{A}+\mathcal{B}} = F_{\mathcal{A}} + F_{\mathcal{B}}$
- **Product** : $F_{\mathcal{A}\times\mathcal{B}} = F_{\mathcal{A}} \times F_{\mathcal{B}}$
- **Sequence** : $F_{Seq(\mathcal{A})} = \frac{1}{1 - F_{\mathcal{A}}}$
- **Diagonal** : $F_{\Delta_k(\mathcal{A})}(s_1, s_2, \dots) = F_{\mathcal{A}}(s_k, s_{2k}, \dots)$
- **MultiSet** : $F_{MSet(\mathcal{A})} = \exp \sum_{k \geq 1} \frac{1}{k} F_{\Delta_k(\mathcal{A})}$
- **Cycle** : $F_{Cyc(\mathcal{A})} = \sum_{k \geq 1} -\frac{\varphi(k)}{k} \ln(1 - F_{\Delta_k(\mathcal{A})})$

Let us notice that $F_{\mathcal{A}}$ is not an enumerative generating function for the profiled objects in \mathcal{A} . But it has the following very interesting property. Let us put $f_{\mathcal{A}}(x, t) = F_{\mathcal{A}}(t.x, t.x^2, \dots, t.x^k, \dots)$. It is a classical result of Polya theory that $[x^n]f_{\mathcal{A}}(x, t) = |\mathcal{A}_n(\mathcal{Z}_1 + \dots + \mathcal{Z}_t)|$. The substitution of s_i by $t.x^i$ only says that the set of i linked atoms represented by s_i can exactly be colored in t different ways and it must be considered as an object of size x^i . So, the function $f_{\mathcal{A}}(x, t)$ is the ordinary generating function for the t -colored combinatorial class $\mathcal{A}^{[t]}$. In a sense, the generating function of $F_{\mathcal{A}}$ contains the informations of all the generating functions for the t -colored combinatorial classes. In fact, this multivariate generating function is generally called cycle index sum and it is also fundamental in [4]. For instance, consider $\mathcal{S} = Mset(\mathcal{Z})$, the first terms of the profiled generating function are $F_{MSet(\mathcal{Z})} = 1 + s_1 + s_1^2/2 + s_2/2 + s_1^3/6 + s_1s_2/2 + s_3/3 + \dots$. To obtain the enumerative generating function for the 3-colored $Mset(\mathcal{Z})$, it suffices to replace s_i by $3x^i$. So, we find $F_{MSet(\mathcal{Z})} = 1 + 3x + 6x^2 + 10x^3 + \dots$

Remark : In constructive profiled classes, every symetries are only obtained by the diagonal operator. So, in the implementation, we do not need to stock the partition, we simply keep the diagonals unexpanded. That's why our profiled samplers return only an "object" a , instead of a couple $\langle a, \sigma \rangle$ as in our proofs. Moreover, this choice induces a storage gain, as big as the object has symmetries.

3. PROBABILITY AND BOLTZMANN SAMPLERS

The principle of Boltzmann samplers can be describe as follows : Let \mathcal{A} be a combinatorial unlabelled class and $A(x)$ its ordinary generating function. Consider a non negative real number x in the convergence disk of $A(x)$, a Boltzmann sampler $\Gamma_{\mathcal{A}}$ is a random generator that draws each object $a \in \mathcal{A}$ with probability $\mathbb{P}(a) = \frac{x^{|a|}}{A(x)}$. There are simple rules, described in [5] to build automatically Boltzmann samplers from the combinatorial specification of \mathcal{A} .

For our purpose, we want to draw t -colored objects coming from a specified class $\mathcal{A}(\mathcal{Z})$ with the following Boltzmann probability $\mathbb{P}_{t,x}(a) = \frac{x^{|a|}}{f_{\mathcal{A}}(x,t)}$.

We produce samplers for (non colored) profiled objects with the following Boltzmann distribution : the probability to obtain a profiled object $\langle a, \sigma \rangle$ such that σ has n_i parts of size i is $\frac{[s_1^{n_1} s_2^{n_2} \dots] F_{\mathcal{A}}(s_{(n) \in \mathbb{N}^*}) \cdot s_1^{n_1} s_2^{n_2} \dots}{F_{\mathcal{A}}(s_{(n) \in \mathbb{N}^*})}$. We then prove that after a constant coloration and a good choice of the parameters, we also obtain a Boltzmann sampler for the t -colored objects. The notion of powersets is not very relevant on profiled combinatorial classes. Indeed, in a profiled object, some atoms are undistinguished, but there is no information on distinguished atoms, they can be the similar or not. Below, a tabular with the classical distributions of probability and the design rules for basic sampler constructions where $\Gamma \mathcal{A}$ designs a Boltzmann sampler for the class \mathcal{A} .

Distribution	Notation	Definition
Bernoulli	$\text{Bern}(p)$	$\mathbb{P}(0) = 1 - p$ and $\mathbb{P}(1) = p$ (with $0 \leq p \leq 1$)
Geometric	$\text{Geom}(\lambda)$	$\mathbb{P}(k) = \lambda^k (1 - \lambda)$ (with $k \in \mathbb{N}$ and $0 \leq \lambda < 1$)
Poisson	$\text{Pois}(\lambda)$	$\mathbb{P}(k) = e^{-\lambda} \frac{\lambda^k}{k!}$ (with $k \in \mathbb{N}$ and $\lambda \in \mathbb{R}^{+*}$)
Positive Poisson	$\text{Pois}_{\geq 0}(\lambda)$	$\mathbb{P}(k) = \frac{1}{e^\lambda - 1} \frac{\lambda^k}{k!}$ (with $k \in \mathbb{N}^*$ and $\lambda \in \mathbb{R}^{+*}$)

FIGURE 2. Distributions of use in Boltzmann sampling.

In the sequel, the symbol \leftarrow designs the affection.

- **Neutral class** : $\Gamma \mathcal{E}_{(s_1, s_2, \dots)} := \text{return } ()$
- **Atomic class** : $\Gamma \mathcal{Z}_{(s_1, s_2, \dots)} := \text{return } \square$
- **Sum** : $\Gamma (\mathcal{A} + \mathcal{B})_{(s_1, s_2, \dots)} :=$

Draw X following a Bernoulli law of parameter $\text{Bern}\left(\frac{F_{\mathcal{A}}(s_1, s_2, \dots)}{F_{\mathcal{A} + \mathcal{B}}(s_1, s_2, \dots)}\right)$

If $X = 1$ return $\Gamma \mathcal{A}_{(s_1, s_2, \dots)}$ else return $\Gamma \mathcal{B}_{(s_1, s_2, \dots)}$

- **Product** : $\Gamma (\mathcal{A} \times \mathcal{B})_{(s_1, s_2, \dots)} :=$
Let $A \leftarrow \Gamma \mathcal{A}_{(s_1, s_2, \dots)}$ and $B \leftarrow \Gamma \mathcal{B}_{(s_1, s_2, \dots)}$
return (AB)
- **Diagonal** : $\Gamma \Delta_k(\mathcal{A})_{(s_1, s_2, \dots)} :=$
let $A \leftarrow \Gamma \mathcal{A}_{(s_k, s_{2k}, \dots)}$
return $\delta_k A$.
- **Sequence** : See Algorithm 1.
- **MultiSet** : See Algorithm 2.
- **Cycle** : See Algorithm 3.

Proposition 3.1. *The previous samplers are valid Boltzmann samplers for profiled objects.*

Algorithm 1: $\Gamma \text{Seq}(\mathcal{A})_{(s_1, s_2, \dots)}$

Input: the parameters s_1, s_2, \dots

Output: a sequence.

- 1 Draw k following the geometric law $\text{Geom}(F_{\mathcal{A}}(s_1, s_2, \dots))$.
 - 2 **for** i **from** 1 **to** k **do**
 - 3 $A_i \leftarrow \Gamma \mathcal{A}_{(s_1, s_2, \dots)}$
 - 4 **return** $(A_1 \dots A_k)$.
-

Algorithm 2: $\Gamma MSet(\mathcal{A})_{(s_1, s_2, \dots)}$

Input: the parameters s_1, s_2, \dots
Output: a multiset.

- 1 let K be a random variable in \mathbb{N}^* verifying
 $\mathbb{P}(K \leq k) = \prod_{j>k} \exp(\frac{1}{j} F_{\mathcal{A}}(s_j, s_{2j}, \dots))$
- 2 Draw k following the law of K .
- 3 $S \leftarrow \epsilon$
- 4 **if** $k = 1$ **then**
 - 5 Draw q following a Poisson law of parameter $F_{\mathcal{A}}(s_1, s_2, \dots)$.
 - 6 **for** i **from** 1 **to** q **do**
 - 7 $A_i \leftarrow \Gamma \mathcal{A}_{(s_1, s_2, \dots)}$
 - 8 $S \leftarrow Concat(S, A_i)$.
 - 9 **return** $\{S\}$
- 10 **else**
 - 11 **for** j **from** 1 **to** $k - 1$ **do**
 - 12 Draw q following a Poisson law of parameter $\frac{1}{j} F_{\mathcal{A}}(s_j, s_{2j}, \dots)$.
 - 13 **for** i **from** 1 **to** q **do**
 - 14 $A_i \leftarrow \Gamma \Delta_j(\mathcal{A})_{(s_1, s_2, \dots)}$
 - 15 $S \leftarrow Concat(S, A_i)$.
 - 16 Draw q following a Poisson $_{\geq 1}$ law of parameter $\frac{1}{k} F_{\mathcal{A}}(s_k, s_{2k}, \dots)$.
 - 17 **for** i **from** 1 **to** q **do**
 - 18 $A_i \leftarrow \Gamma \Delta_k(\mathcal{A})_{(s_1, s_2, \dots)}$
 - 19 $S \leftarrow Concat(S, A_i)$.
 - 20 **return** $\{S\}$

Algorithm 3: $\Gamma Cyc(\mathcal{A})_{(s_1, s_2, \dots)}$

Input: the parameters s_1, s_2, \dots
Output: a cycle.

- 1 let K be a random variable in \mathbb{N}^* verifying
 $\mathbb{P}(K = k) = -\frac{1}{F_{Cyc(\mathcal{A})}} \frac{\varphi(k)}{k} \ln(1 - F_{\Delta_k(\mathcal{A})})$
- 2 Draw k following the law of K .
- 3 let L be a random variable in \mathbb{N}^* verifying $\mathbb{P}(L = l) = -\frac{(F_{\Delta_k(\mathcal{A})})^l}{l} \frac{1}{\ln(1 - F_{\Delta_k(\mathcal{A})})}$
- 4 Draw l following the law of L .
- 5 $M \leftarrow \epsilon$
- 6 **for** i **from** 1 **to** l **do**
 - 7 $A_i \leftarrow \Gamma \mathcal{A}_{(s_k, s_{2k}, \dots)}$
 - 8 $M \leftarrow Concat(M, A_i)$.
- 9 **return** $[\delta_k M]$.

Proof of the proposition 3.1. We only prove that for the sampler for MSet and Cyc. The other ones are easy exercices. Let us begin with the MSet sampler. First, suppose that during the execution of the algorithm, we have drawn $k = 1$. So, $\mathbb{P}(K = 1) = \exp(F_{\mathcal{A}}(s_1, s_2, \dots))$. After that, we have chosen q with probability :

$$\mathbb{P}(q) = \exp(-F_{\mathcal{A}}(s_1, s_2, \dots)) \cdot \frac{F_{\mathcal{A}}(s_1, s_2, \dots)^q}{q!}$$

and we have drawn q objects in \mathcal{A} , each of them has profile σ such that the number of parts of σ of cardinal i is n_i with probability :

$$\mathbb{P}_{\mathbf{n}} = \frac{[\mathbf{s}^{\mathbf{n}}]F_{\mathcal{A}}(s_{(n) \in \mathbb{N}^*}) \cdot \mathbf{s}^{\mathbf{n}}}{F_{\mathcal{A}}(s_{(n) \in \mathbb{N}^*})}.$$

where we denote $s_1^{n_1} s_2^{n_2} \dots$ by $\mathbf{s}^{\mathbf{n}}$. So, the probability to obtain an object of profile σ such that the number of parts of σ of cardinal i is n_i (when we draw $k = 1$) is the product of all these terms. This can be simplify in :

$$\frac{\mathbf{s}^{\mathbf{n}}}{\prod_{j=1}^{\infty} \frac{1}{j} F_{\mathcal{A}}(s_j, s_{2j}, \dots)} \sum_{q=0}^{\infty} \sum_{(\mathbf{m}_1, \dots, \mathbf{m}_q); \sum_d \mathbf{m}_d = \mathbf{n}} \frac{\prod_{d=1}^q [\mathbf{s}^{\mathbf{m}_d}] F_{\mathcal{A}}(s_{(n) \in \mathbb{N}^*})}{q!}.$$

Now, for $k > 1$, in the same way, we can show that the probability is

$$\sum_{k > 1} \sum_{q_1 > 0, \dots, q_k > 0} \sum_{(\mathbf{m}_{1,1}, \dots, \mathbf{m}_{k,q_k}); \sum_{i=1}^k \sum_{j=1}^{q_i} \mathbf{m}_{i,j} = \mathbf{n}} \frac{1}{1^{q_1} q_1! \dots k^{q_k} q_k!} \cdot \frac{\prod_{p=1}^k \prod_{d=1}^{q_p} [\mathbf{s}^{\mathbf{m}_{p,d}}] F_{\mathcal{A}}(s_{(n) \in \mathbb{N}^*}) \cdot \mathbf{s}^{\mathbf{m}_{p,d}}}{\prod_{j=1}^{\infty} \frac{1}{j} F_{\mathcal{A}}(s_j, s_{2j}, \dots)}.$$

So, using this sampler, the probability to obtain an object of profile σ such that the number of parts of σ of cardinal i is n_i is

$$\sum_{k \geq 1} \sum_{q_1 > 0, \dots, q_k > 0} \sum_{(\mathbf{m}_{1,1}, \dots, \mathbf{m}_{k,q_k}); \sum_{i=1}^k \sum_{j=1}^{q_i} \mathbf{m}_{i,j} = \mathbf{n}} \frac{1}{1^{q_1} q_1! \dots k^{q_k} q_k!} \cdot \frac{\mathbf{s}^{\mathbf{n}} \prod_{p=1}^k \prod_{d=1}^{q_p} [\mathbf{s}^{\mathbf{m}_{p,d}}] F_{\mathcal{A}}(s_{(n) \in \mathbb{N}^*})}{\prod_{j=1}^{\infty} \frac{1}{j} F_{\mathcal{A}}(s_j, s_{2j}, \dots)}.$$

Indeed, this is all the ways to obtain an object of such a profile.

Now, $[\mathbf{s}^{\mathbf{n}}]F_{MSet(\mathcal{A})}(s_1, s_2, \dots) = \sum_{\sum_j n_j = n} \prod_j [\mathbf{s}^{n_j}] \exp(\frac{1}{j} F_{\mathcal{A}}(s_j, s_{2j}, \dots))$. Moreover,

$$[\mathbf{s}^{n_j}] \exp(\frac{1}{j} F_{\mathcal{A}}(s_j, s_{2j}, \dots)) = \sum_{\alpha=0}^{\infty} \frac{1}{\alpha!} \sum_{\sum_{p=1}^{\alpha} n_{j,p} = n_j} \prod_{p=1}^{\alpha} [\mathbf{s}^{n_{j,p}}] \frac{F_{\mathcal{A}}(s_j, s_{2j}, \dots)}{j}.$$

So,

$$[\mathbf{s}^{\mathbf{n}}]F_{MSet(\mathcal{A})}(s_1, s_2, \dots) = \sum_{\sum_{j=1}^{\infty} n_j = n} \prod_{j=1}^{\infty} \sum_{\alpha=0}^{\infty} \frac{1}{\alpha!} \sum_{\sum_{p=1}^{\alpha} n_{j,p} = n_j} \prod_{p=1}^{\alpha} [\mathbf{s}^{n_{j,p}}] \frac{F_{\mathcal{A}}(s_j, s_{2j}, \dots)}{j}.$$

We have to change the order of the sums and products. We use two times that $\prod_{j=1}^{\infty} \sum_{\alpha \in A} a_{\alpha,j} = \sum_{\alpha \in A^{\infty}} \prod_{j=1}^{\infty} a_{\alpha,j}$, and we obtain that the probability to draw an object of profile σ is $\frac{\mathbf{s}^{\mathbf{n}} \cdot [\mathbf{s}^{\mathbf{n}}]F_{MSet(\mathcal{A})}(s_1, s_2, \dots)}{F_{MSet(\mathcal{A})}(s_1, s_2, \dots)}$.

Now, we analyse the sampler for Cyc. Suppose that the algorithm returns an object of size n and profile σ such that the number of parts of σ of cardinal i is n_i .

The probability $p_{\mathbf{n}}$ to draw such an object is

$$\sum_{k.l=n} \sum_{(\mathbf{m}_1, \dots, \mathbf{m}_l); \sum_{d=1}^l \mathbf{m}_d = \mathbf{n}} \mathbb{P}(K = k) \cdot \mathbb{P}(L = l) \cdot \prod_{i=1}^l p_{\mathbf{m}_i}$$

$$\text{with } \mathbb{P}(K = k) = -\frac{1}{F_{Cyc(\mathcal{A})}(s_{(n) \in \mathbb{N}^*})} \frac{\varphi(k)}{k} \ln(1 - F_{\Delta_k(\mathcal{A})}),$$

$$\mathbb{P}(L = l) = -\frac{(F_{\Delta_k(\mathcal{A})}(s_{(n) \in \mathbb{N}^*}))^l}{l} \frac{1}{\ln(1 - F_{\Delta_k(\mathcal{A})}(s_{(n) \in \mathbb{N}^*}))}$$

$$\text{and } \forall i \in \{0 \dots l\}, p_{\mathbf{m}_i} = \frac{[\mathbf{s}^{\mathbf{m}_i}] F_{\Delta_k(\mathcal{A})}(s_{(n) \in \mathbb{N}^*}) \cdot \mathbf{s}^{\mathbf{m}_i}}{F_{\Delta_k(\mathcal{A})}(s_{(n) \in \mathbb{N}^*})}$$

So,

$$\begin{aligned} p_{\mathbf{n}} &= \sum_{k.l=n} \sum_{(\mathbf{m}_1, \dots, \mathbf{m}_l); \sum_{d=1}^l \mathbf{m}_d = \mathbf{n}} \frac{1}{F_{Cyc(\mathcal{A})}(s_{(n) \in \mathbb{N}^*})} \frac{\varphi(k)}{k} \frac{1}{l} \prod_{i=1}^l [\mathbf{s}^{\mathbf{m}_i}] F_{\Delta_k(\mathcal{A})}(s_{(n) \in \mathbb{N}^*}) \cdot \mathbf{s}^{\mathbf{m}_i} \\ &= \frac{[\mathbf{s}^{\mathbf{n}}] F_{Cyc(\mathcal{A})}(s_{(n) \in \mathbb{N}^*}) \cdot \mathbf{s}^{\mathbf{n}}}{F_{Cyc(\mathcal{A})}(s_{(n) \in \mathbb{N}^*})} \end{aligned}$$

□

4. BOLTZMANN SAMPLERS FOR t -COLORED CLASSES AND COLORED CLASSES

4.1. The t -colored Boltzmann Samplers. In this section, we mention how to obtain from the sampler for profiled objects the samplers for the t -colored objects.

Proposition 4.1. *Let us substitute s_i by $t.x^i$ and generate a profiled object $\langle a, \sigma \rangle$ as previously. For each part p of σ , draw uniformly a color c_p in $\{1, \dots, t\}$ and let us assign the atoms in p with the color c_p .*

This sampler is a valid Boltzmann sampler for the t -colored objects in the class \mathcal{A} .

Proof. Let us consider a t -colored object a drawn by this sampler. Suppose that a is of size n and have n_i atoms colored with the color i . The probability to obtain a is the sum on the a -admissible profiled objects times the probability to put the good colors. So, we have : $\mathbb{P}(a) = \sum_{\sigma \text{ admissible profile}} \frac{\prod (t.x^i)^{n_i}}{F_{\mathcal{A}}(t.x, t.x^2, \dots)} \cdot \frac{\kappa_{\sigma} \cdot [\mathbf{s}^{\mathbf{n}}] F_{\mathcal{A}}(t.x, t.x^2, \dots)}{t^{\sum n_i}} = \frac{x^{|a|}}{f_{\mathcal{A}}(x, t)} \cdot \sum_{\sigma \text{ admissible profile}} \kappa_{\sigma} \cdot [\mathbf{s}^{\mathbf{n}}] F_{\mathcal{A}}(t.x, t.x^2, \dots)$ where κ_{σ} is the number of \equiv_B -equivalent coloration that corresponds to the color of a .

It can be proved that $\sum_{\sigma \text{ admissible profile}} \kappa_{\sigma} \cdot [\mathbf{s}^{\mathbf{n}}] F_{\mathcal{A}}(t.x, t.x^2, \dots) = 1$.

Another proof of this proposition is just to observe that after the substitution of s_i by $t.x^i$, our sampler is syntactically identical to a classical Boltzmann sampler for k -colored objects, except that we have postponed the Bernoulli choice of the color by the informations stored in the profile. □

So, we have valid Boltzmann samplers for the t -colored objects. We can use it for exact size generation as well as approximate size sampling.

4.2. Approximate-size Boltzmann Samplers for size-colored objects. Now, we want to use profiled object samplers to generate colored objects. Suppose that we want to generate a colored object of size n in a combinatorial class \mathcal{A} . First, we have to solve the following equation to find the parameter x_0 which targets correctly the samplers : $x \cdot \frac{\partial f_{\mathcal{A}}(x, n)}{f_{\mathcal{A}}(x, n)} = n$. We draw a profiled object with the parameters $s_i = n \cdot x_0^i$. Now, if we color this object with n colors, the sampler is uniform only on the class $\mathcal{A}_n^{[n]}$ that is to say when we have drawn a profiled object of size n . We unbias the approximate size generator by adding some rejections. The following lemma explains how we filter the sampler in order to correct it.

Lemma 4.2. *Let $A = \{x_1, \dots, x_k\}$ be a set of objects and Γ a sampler for A that draws the object x_i with probability $p_i \neq 0$. Let (p'_1, \dots, p'_k) another distribution of probability on A , and suppose that the $\frac{p'_i}{p_i}$ forms a decreasing sequence. Consider the following sampler Γ' :*

Algorithm 4: Γ'

Input: the distribution of p'_i

Output: an object.

- 1 Draw an object x with Γ
 - 2 Let i such that $x = x_i$
 - 3 Draw X following a Bernoulli law of parameter $\frac{p'_i \cdot p_1}{p_i \cdot p'_1}$
 - 4 **if** $X = 1$ **then**
 - 5 | **return** x
 - 6 **else**
 - 7 | **restart** Γ'
-

Then Γ' draws the object x_i with probability p'_i . Moreover, Γ' calls on average $\frac{p'_i}{p_i}$ times Γ to obtain an object.

Proof. First, observe that the parameter $\frac{p'_i \cdot p_1}{p_i \cdot p'_1}$ is in $[0, 1]$. Put $x_i = \frac{p'_i \cdot p_1}{p_i \cdot p'_1}$, the probability to obtain x_i is $\sum_{n=0}^{\infty} (\sum_j p_j \bar{x}_j)^n p_i x_i = \frac{p_i x_i}{1 - \sum_j p_j \bar{x}_j} = \frac{p_i x_i}{\sum_j p_j x_j}$. In this

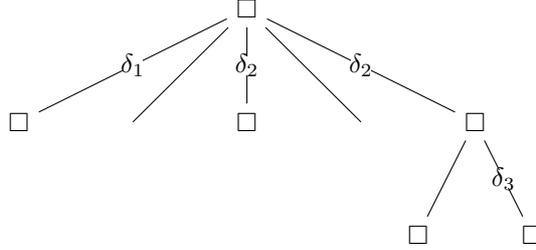
expression, substitute x_i by $\frac{p'_i \cdot p_1}{p_i \cdot p'_1}$, we get $\frac{p_i \frac{p'_i \cdot p_1}{p_i \cdot p'_1}}{\sum_j p_j \frac{p'_j \cdot p_1}{p_j \cdot p'_1}} = \frac{p'_i}{\sum_j p'_j} = p'_i$. The expected

time is $\sum_{n=0}^{\infty} (n+1) (\sum_j p_j \bar{x}_j)^n p_i x_i$. A similar calculation shows that this is equal to $\frac{p'_i}{p_i}$. □

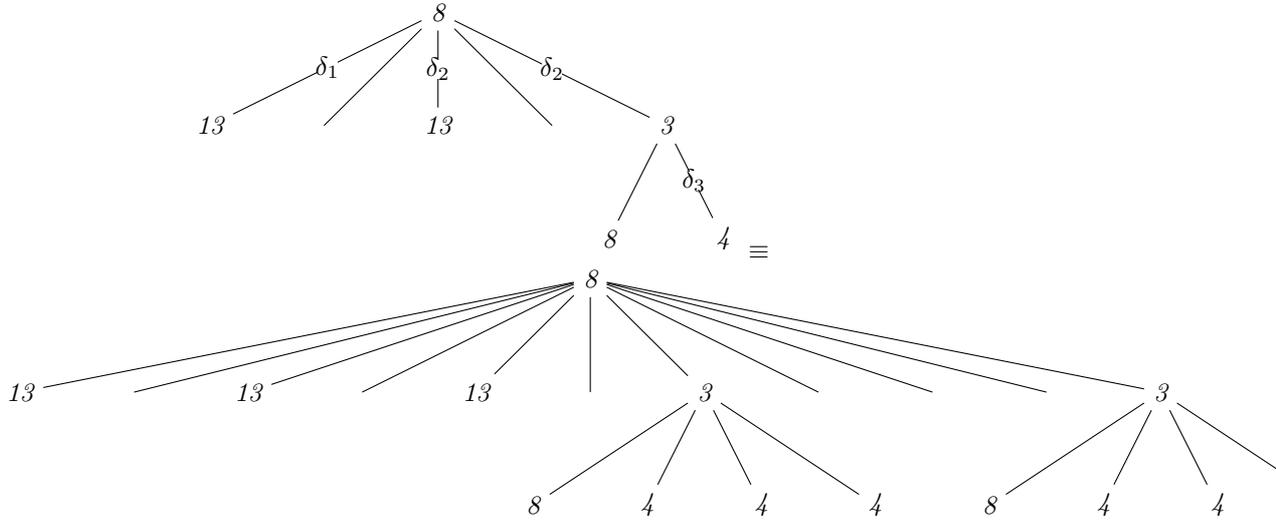
Now, consider the problem for sampling for colored objects of size around n , a sampler is described in Algorithm 5. In a first time, we propose to observe its running on an example :

Example 4.3. *A generation for a colored general tree of specification $T = \mathcal{Z} + \mathcal{ZMSet}(T)$ by Algorithm 5, with parameters 14, x_0 .*

First, a profiled tree A is drawn by Algorithm 2 with $s_i = 14 \cdot x_0^i$:
 We draw $k=2$.
 For $j=1$, we draw $q=1$, consequently we draw an only profiled object using $\Gamma\Delta_1 T_{(s_1, s_2, \dots)}$.
 Then, we draw $q=2$, this involves the sampling of 2 profiled object by $\Gamma\Delta_2 T_{(s_1, s_2, \dots)}$.
 After that, we get the following profiled tree A :



As the size of A is $14 = n$ (So lucky, we are), we just have to return a 14-coloration of A respecting its symetries. We find :



Theorem 4.4. *The following algorithm is a approximate-size Boltzmann sampler for the size-colored objects in \mathcal{A} . Its overall cost is*

$$O\left(n \cdot ((1 - \epsilon) \cdot \frac{f_{\mathcal{A}}(x, n)}{f_{\mathcal{A}}(x, (1 - \epsilon)n)} + (1 + \epsilon)^{(1 + \epsilon)n} \cdot \frac{f_{\mathcal{A}}(x, n)}{f_{\mathcal{A}}(x, (1 + \epsilon)n)})\right)$$

Algorithm 5: Γ_{colored}

Input: the parameters n, x_0
Output: a colored object.

- 1 Draw a profiled object a with the parameters $s_i = n \cdot x_0^i$
- 2 Let $\tilde{n} \leftarrow$ size of a
- 3 Let \tilde{n}_i the number of parts of the profile of cardinal i .
- 4 **if** $\tilde{n} \notin [(1 - \epsilon)n, (1 + \epsilon)n]$ **then**
- 5 | restart Γ_{colored}
- 6 **else**
- 7 | **switch** the value \tilde{n} **do**
- 8 | | **case** $\tilde{n} = n$
- 9 | | | **return** a n -coloration of a .
- 10 | | **case** $\tilde{n} < n$
- 11 | | | Let α be the minimum in
| | | $\{\sum n_i; \sum i n_i = \tilde{n} \text{ and } [s_1^{n_1} s_2^{n_2} \dots] F_{\mathcal{A}}(s_1, s_2, \dots) \neq 0\}$
- 12 | | | Draw X following a Bernoulli of parameter $(\frac{\tilde{n}}{n})^{-\alpha + \sum \tilde{n}_i}$
- 13 | | | **if** $X = 1$ **then**
- 14 | | | | **return** a \tilde{n} -coloration of a
- 15 | | | | **else**
- 16 | | | | | restart Γ_{colored}
- 17 | | **case** $\tilde{n} > n$
- 18 | | | Draw X following a Bernoulli of parameter $(\frac{\tilde{n}}{n})^{-\tilde{n} + \sum \tilde{n}_i}$
- 19 | | | **if** $X = 1$ **then**
- 20 | | | | **return** a \tilde{n} -coloration of a
- 21 | | | | **else**
- 22 | | | | | restart Γ_{colored}

Proof of the correctness of Γ_{colored} . Using our sampler of profiled objects with the parameters $s_i = n \cdot x_0^i$, the probability to draw a profiled object of size \tilde{n} having a profile with \tilde{n}_i parts of cardinal i is $\mathbb{P}_n(\tilde{\mathbf{n}}) = \frac{n^{\sum \tilde{n}_i} \cdot x_0^{\tilde{n}} \cdot [\mathbf{s}^{\tilde{\mathbf{n}}}] F_{\mathcal{A}}}{f_{\mathcal{A}}(x, n)}$. We have to add after

that a filter in such a way that the probability becomes $\mathbb{P}_{\tilde{n}}(\tilde{\mathbf{n}}) = \frac{\tilde{n}^{\sum \tilde{n}_i} \cdot x_0^{\tilde{n}} \cdot [\mathbf{s}^{\tilde{\mathbf{n}}}] F_{\mathcal{A}}}{f_{\mathcal{A}}(x, \tilde{n})}$.

So, we only have to know for which profile the ratio $\frac{\mathbb{P}_{\tilde{n}}(\tilde{\mathbf{n}})}{\mathbb{P}_n(\tilde{\mathbf{n}})}$ is maximum. If $\tilde{n} < n$ (resp. $\tilde{n} > n$), this happens when $\sum n_i$ is minimum belongs the (n_1, n_2, \dots) such that $\sum i n_i = \tilde{n}$ and $[s_1^{n_1} s_2^{n_2} \dots] F_{\mathcal{A}}(s_1, s_2, \dots) \neq 0$ (resp. when $\sum n_i$ is maximum belongs the (n_1, n_2, \dots) such that $\sum i n_i = \tilde{n}$ and $[s_1^{n_1} s_2^{n_2} \dots] F_{\mathcal{A}}(s_1, s_2, \dots) \neq 0$). Notice that the maximum is always \tilde{n} . The choice of the Bernoulli parameters follows directly from the lemma 4.2.

Under classical conditions explain in [5], the sampling in approximate-size of a k -colored constructible structure can be done in linear time. Now, it suffices to evaluate the cost of the added rejection phase. The average number of loops is $(1 - \epsilon) \cdot \frac{f_{\mathcal{A}}(x, n)}{f_{\mathcal{A}}(x, (1 - \epsilon)n)}$ when $\tilde{n} < n$ and $(1 + \epsilon)^{(1 + \epsilon)n} \cdot \frac{f_{\mathcal{A}}(x, n)}{f_{\mathcal{A}}(x, (1 + \epsilon)n)}$ otherwise. \square

Remark 4.5. *At this stage, the explicit evaluation of the ratio $\frac{f_{\mathcal{A}}(x, n)}{f_{\mathcal{A}}(x, (1 + \epsilon)n)}$ is out of reach, but it experimentally seems that it is often better to choose $\tilde{n} > n$. In this case, it is more advisable to modify the line 4 in the previous algorithm by (4') if $\tilde{n} \notin [n, (1 + \epsilon)n]$ then restart Γ_{colored} .*

5. CONCLUSION

We have proved that the profiled objects are powerful tools to generate under Boltzmann model k -colored objects and to obtain a generator in approximate size for colored objects. Surprisingly, we only have to know the minimum in $\{\sum n_i; \sum in_i = \tilde{n} \text{ and } [s_1^{n_1} s_2^{n_2} \dots]F_{\mathcal{A}}(s_1, s_2, \dots) \neq 0\}$ to unbiased the generation of colored objects. We also expect to develop our ideas to even more general coloration types as for instance to impose constraints on the number of atoms with the same colors. Finally, we would like to thank E. Fusy, C. Pivoteau and M. Soria for their precious comments and corrections.

REFERENCES

- [1] F. Bergeron, G. Labelle, P. Leroux, "Combinatorial species and tree-like structures.", Encyclopedia of Mathematics and its Applications, 67. Cambridge University Press, Cambridge, 1998. xx+457 pp.
- [2] O. Bodini E. Fusy and C. Pivoteau, "Random sampling of plane partitions." Proceedings of Gascom'06.
- [3] O. Bodini A. Darrasse, M. Soria, "Distances in random Apollonian network structures", Proceedings of FPSAC'08, 2008.
- [4] M. Bodirsky, E. Fusy, M. Kang and S. Vigerske, "An unbiased pointing operator for unlabeled structures, with applications to counting and sampling", Proceedings of SODA'07.
- [5] P. Duchon, P. Flajolet, G. Louchard and G. Schaeffer. "Boltzmann Samplers for the Random Generation of Combinatorial Structures", In Combinatorics, Probability, and Computing, Special issue on Analysis of Algorithms, 2004, Vol. 13, No 4-5, pp. 577-625.
- [6] A. Denise, M-C. Gaudel, S-D. Gouraud, R. Lassaigne, and S. Peyronnet. "Uniform random sampling of traces in very large models." Proceedings of the First International Workshop on Random Testing (RT 2006), July 2006, pages 10-19.
- [7] A. Darrasse, M. Soria, "Degree distribution of random Apollonian network structures and Boltzmann sampling", Proceedings of International Conference on Analysis of Algorithms DMTCS, 2007.
- [8] P. Flajolet, E. Fusy and C. Pivoteau, "Boltzmann Sampling of Unlabelled Structures", Proceedings of ANALCO'07 (Analytic Combinatorics and Algorithms) Conference, New Orleans, January 2007. SIAM Press, in print.
- [9] P. Flajolet and R. Sedgewick, "Analytic Combinatorics" published soon by Cambridge University Press.
- [10] H. Fredricksen and J. Maiorana, "Necklaces of beads in k colors and k -ary de Bruijn sequences", Discrete Mathematics 23 (1978), 207-210.
- [11] A. Nijenhuis and H. Wilf, 'Combinatorial Algorithms', Academic Press, ISBN 0125192509 (1975).
- [12] Y. Ponty, M. Termier and A. Denise. "GenRGenS: Software for generating random genomic sequences and structures." Bioinformatics 22 (2006) 1534-1535.
- [13] A. Sapounakis and P. Tsikouras, "On k -colored Motzkin words", J. Integer Seq. 7 (2004), Article 04.2.5.