Random generation of finitely generated subgroups of a free group

Frédérique Bassino, Cyril Nicaud, Pascal Weil

Random generation of finitely generated subgroups of a free group*

Frédérique Bassino, bassino@univ-mlv.fr†
Institut Gaspard Monge, Université de Marne la Vallée

Cyril Nicaud, nicaud@univ-mlv.fr‡
Institut Gaspard Monge, Université de Marne la Vallée

Pascal Weil, pascal.weil@labri.fr§
LaBRI, Université de Bordeaux, CNRS

July 21, 2007

Abstract

We give an efficient algorithm to randomly generate finitely generated subgroups of a given size, in a finite rank free group. Here, the size of a subgroup is the number of vertices of its representation by a reduced graph such as can be obtained by the method of Stallings foldings. Our algorithm randomly generates a subgroup of a given size n, according to the uniform distribution over size n subgroups. In the process, we give estimates of the number of size n subgroups, of the average rank of size n subgroups, and of the proportion of such subgroups that have finite index. Our algorithm has average case complexity $\mathcal{O}(n)$ in the RAM model and $\mathcal{O}(n^2 \log^2 n)$ in the bitcost model.

Keywords: subgroups of free groups, random generation

MSC: 05A16, 20E05

*The first and third authors benefitted from the support of the PICASSO project AUTOMATA AND FREE GROUPS. The third author acknowledges partial support from the ESF program AUTOMATHA and the Algebraic Cryptography Center, at the Stevens Institute of Technology.

†Institut Gaspard Monge, Université de Marne la Vallée, 77454 Marne-la-Vallée Cedex 2, France

‡Institut Gaspard Monge, Université de Marne la Vallée, 77454 Marne-la-Vallée Cedex 2, France

§LaBRI, 351 cours de la Libération, 33400 Talence, France.

Algorithmic problems in combinatorial group theory have been the focus of increased interest (see [3, 18, 15, 14, 16, 17, 20] for recent examples). This is especially the case for the theory of free groups and the investigation of their finitely generated subgroups, which is the focus of this paper.

As in other fields, the investigation of algorithmic problems and of their complexity brings to light interesting structural properties of the objects under study. One classical approach is to study the properties of *random* objects, here of *random* finitely generated subgroups of a free group. This naturally depends on the probability distribution we adopt, we come back to this below.

The complexity of algorithms is often estimated according to the worst-case paradigm. It can also be estimated in average, or *generically* [16]. Both these concepts also depend on the choice of a distribution, and can benefit directly from the enumeration and generation results in this paper. Random generation can also be useful to test conjectures or algorithms with large set of *representative* instances (again, depending on the choice of a probability distribution), provided that the random generation algorithm is fast enough.

In this paper, we use the well-known fact (Stallings [26]) that every finitely generated subgroup of a free group F with basis A admits a unique graphical representation of the form $(\Gamma, 1)$, where Γ is a finite directed graph with A-labeled edges and 1 is a designated vertex of Γ — subject to certain combinatorial conditions, see Section 1.1 for details. Then we consider the number of vertices of Γ to be a measure of the *size* of the subgroup represented by $(\Gamma, 1)$. Note that the size of H is strongly dependent on the choice of the basis A of F. Very importantly also, for each $n \geq 1$, there are finitely many subgroups of F of size n. The probability distribution on subgroups discussed in this paper is the uniform distribution on the set of fixed size subgroups: if there are $s(n)$ subgroups of size n, each has probability $\frac{1}{s(n)}$.

A large and growing number of algorithmic problems on free groups admit efficient solutions using these graphical representations (see [14, 15, 16, 17, 18, 21, 24] among others), and this further emphasizes the interest of a random generation scheme based on this representation.

The main result of the paper is an efficient algorithm to randomly generate size n subgroups of F. Its average case complexity is $\mathcal{O}(n)$ in the RAM, or unit-cost model, and $\mathcal{O}(n^2 \log^2 n)$ in the bit-cost model, see Section 3.3 for a more precise discussion. Our algorithm actually generates graphical representations $(\Gamma, 1)$ of subgroups, but we want to emphasize that, as these representations are in bijective correspondence with finitely generated subgroups of F, we truly achieve a uniform distribution of (size n) *subgroups*.

The strategy followed by the algorithm is described in Section 1.2. The algorithm itself is actually simple and easily implementable, besides being fast. Its proof is more complex; it relies on the recursive nature of the combinatorial structures underlying graphical representations of subgroups (see Section 2, and in particular Section 2.1), and we make direct use of the concepts and the tools of the so-called *recursive method* heralded by Nijenhuis and Wilf [22] and systematized by Flajolet, Zimmermann and van Cutsem [11].

In Section 4, we study the distribution of the ranks of size n subgroups and we show that if F has rank $r \geq 2$, then the mean value of the rank of a size n subgroup is $(r-1)n - r\sqrt{n} + 1$, with standard deviation $o(\sqrt{n})$. In Section 5, we show how our strategy can be modified (and simplified) to randomly generate in linear time size n finite index subgroups (that is, subgroups of rank $(r-1)n+1$) — even though these groups are negligible among general size n subgroups. We actually give a precise estimate of the proportion of such subgroups and we prove that it converges to 0 faster than the inverse of any polynomial.

The paper closes on a short discussion of related questions, and in particular of the comparison of our distribution with that which is induced by the random generation of an n-tuple of words and the consideration of the subgroup they generate, see [14, 19, 20] for instance.

Throughout this paper, we denote by $|X|$ the cardinality of a set X, and by $[\![1, n]\!]$ the set $\{1, \ldots, n\}$ (where n is a positive integer).

1 General notions and generation strategy

1.1 Graphical representation of subgroups

Let F be a free group with finite rank $r \geq 2$ and let A be a fixed basis of F. We sometimes write $F = F(A)$ and the elements of F are naturally represented as reduced words over the alphabet $A \sqcup A^{-1}$. It is well-known that the subgroups of F are free as well. Moreover, each finitely generated subgroup of F can be represented uniquely by a finite graph of a particular type, by means of the technique known as *Stallings foldings* [26] (see also [28, 15, 27]). We refer the reader to the literature for a description of this very fruitful technique, and we only record here the results that will be useful for our purpose.

An *A-graph* is defined to be a pair $\Gamma = (V, E)$ with $E \subseteq V \times A \times V$, such that

- if $(u, a, v), (u, a, v') \in E$, then $v = v'$;
- if $(u, a, v), (u', a, v) \in E$, then $u = u'$.

The elements of V are called the *vertices* of Γ, the elements of E are its *edges*, and we sometimes write $V(\Gamma)$ for V and $E(\Gamma)$ for E. We say that Γ is *connected* if the underlying undirected graph is connected. If $v \in V(\Gamma)$, we say that v is a *leaf* if v occurs at most once in (the list of triples defining) $E(\Gamma)$ and we say that Γ is *v-trim* if no vertex $w \neq v$ is a leaf. Finally we say that the pair (Γ, v) is *admissible* if Γ is a v-trim and connected A-graph.

Then it is known that:

- Stallings foldings associate with each finitely generated subgroup H of $F(A)$ a unique admissible pair of the form $(\Gamma, 1)$, which we call the *graphical representation of H* in this paper [26, 28, 15];

- every admissible pair $(\Gamma, 1)$ is the graphical representation of a unique finitely generated subgroup of $F(A)$ [26, 28, 15];
- if H is given by a finite set of generators (in the form of reduced words over $A \sqcup A^{-1}$) of total length n, then the graphical representation of H can be computed in time $\mathcal{O}(n \log^* n)$ [27];
- if $(\Gamma, 1)$ is the graphical representation of H, then $\mathsf{rank}(H) = |E(\Gamma)| - |V(\Gamma)| + 1$ [26, 28, 15];
- if $(\Gamma, 1)$ is the graphical representation of H, then H has finite index if and only if for each $v \in V(\Gamma)$ and for each $a \in A$, there is an edge of the form $(v, a, w) \in E(\Gamma)$ [26, 28, 15], if and only if $\mathsf{rank}(H) = (|A| - 1)|V(\Gamma)| + 1$.

We sometimes identify H and its graphical representation $(\Gamma, 1)$ — for instance when we say that we randomly generate subgroups of F: what we generate is actually the graphical representation of such subgroups. As explained in the introduction, we consider the number of vertices to be a measure of the size of Γ and we write $|H| = |\Gamma| = |V|$. In particular, F has finitely many subgroups of size n.

1.2 Enumeration and random generation

As we shall see, A-graphs fall in the category of decomposable structures, that is, structures that can be built from unit elements and from operations such as the union, direct product, set formation, etc. We will use the so-called *recursive method* to enumerate and to randomly generate such structures [11]. Details are given further in the paper, concerning the enumeration (Section 2) and the random generation algorithm and its complexity (Section 3). At this point, let us simply say that the random generation of size n A-graphs requires a pre-computation phase in $\mathcal{O}(n)$, after which each draw takes time $\mathcal{O}(n)$.

The rest of this section is devoted to an overview of our strategy.

Remark 1.1 There exists another method than the recursive method, to derive a random generation algorithm from a combinatorial specification, this time according to a *Boltzmann distribution*. Recall that, in such a distribution, an object γ receives a probability essentially proportional to an exponential of its size $|\gamma|$. (More precisely this probability depends upon a positive real parameter x, and it is proportional to $x^{|\gamma|}$ when γ is an unlabeled structure and to $x^{|\gamma|}/|\gamma|!$ when γ is labeled; see Section 1.2.1 below about labeled vs. unlabeled structures.) In particular, Boltzmann samplers do not generate objects of a fixed size. They depend on the real parameter $x > 0$ and, for any given integer n, the value of x can be chosen such that the average size of the generated elements is n. Even though the size of the objects generated is not fixed, Boltzmann samplers guarantee that two elements of the same size have the same probability to be generated.

A method to systematically produce Boltzmann samplers was recently introduced by Duchon, Flajolet, Louchard and Schaeffer [7] for labeled structures

(Flajolet, Fusy and Pivoteau for unlabeled structures in [8]). The evaluation of x is the only required precomputation and the complexity of generation itself is linear as long as small variations in size are allowed. This approach can also be used for exact-size generation, but in the case of A-graphs it is less efficient than the recursive method (see Remark 2.3). □

1.2.1 We count labeled A-graphs

Enumeration for us, is the enumeration of structures up to isomorphism. The structures which we want to generate are admissible pairs $(\Gamma, 1)$, that is, A-graphs with one vertex labeled 1, that are connected and 1-trim. We later use the phrase *admissible A-graphs*. Leaving aside for a moment the properties of connectedness and 1-trimness, we are interested in A-graphs with a distinguished vertex. This is an intermediary situation between labeled and unlabeled structures, which are two great categories of structures for which there exist a large toolkit for enumeration and random generation [11, 10, 7, 8].

An A-graph $\Gamma = (V, E)$ of size n is said to be *labeled* if it is equipped with a bijection $\lambda: [\![1, n]\!] \to V$. Of course, there are $n!$ different such bijections, but some of them may yield isomorphic labeled A-graphs. For instance, if $E = \emptyset$ (so that Γ consists of n isolated vertices), all labelings of Γ are isomorphic.

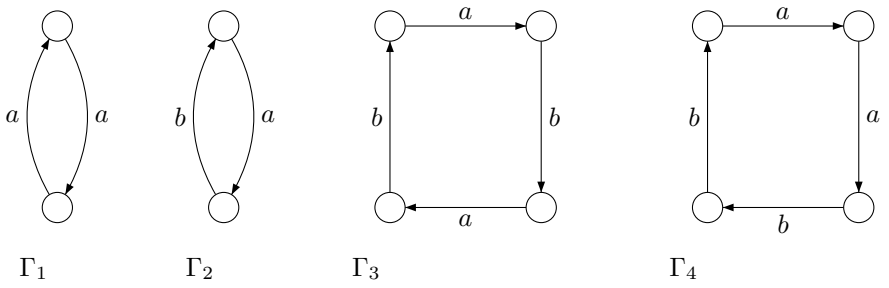


Figure 1: Four A-graphs with different numbers of non-isomorphic labelings

Example 1.2 Consider the A-graphs in Figure 1. Then all labelings of Γ_1 are isomorphic but Γ_2 has 2 non-isomorphic labelings. How many non-isomorphic labelings does Γ_3 have? And Γ_4? □

Counting structures such as A-graphs can lead to complex considerations involving, say, the automorphism groups of each of the connected components of the A-graphs, but standard methods of analytic combinatorics actually solve this enumeration problem for us, see Section 1.2.2. These methods rely on the use of labeled structures, and we will therefore enumerate and generate labeled A-graphs. Why this is justified even for the purpose of randomly generating admissible pairs is discussed in Section 1.2.3.

1.2.2 Generating series

Let $\mathcal{A}$ be a class of combinatorial structures. If $\mathcal{A}$ has a_n elements of size n, the formal power series $\sum_n a_n z^n$ and $\sum_n \frac{a_n}{n!} z^n$ are called respectively the *(ordinary) generating series* and the *exponential generating series* (*EGS*) of the class $\mathcal{A}$.

As it turns out, certain operations on classes of labeled combinatorial structures have a direct translation over their EGS. For instance, suppose that $\mathcal{A}$ is the union of the disjoint classes $\mathcal{B}$ and $\mathcal{C}$ (that is, a size n element of $\mathcal{A}$ is a size n element of either $\mathcal{B}$ or $\mathcal{C}$), and let $A(z)$, $B(z)$ and $C(z)$ be the EGS of the three classes. It is immediate that $A(z) = B(z) + C(z)$.

For more complex operations, one needs to handle the question of relabelings. If S is a size n structure with a labeling function λ, we say that μ is an expansion of λ if the domain of μ is of the form $\{k_1, \dots, k_n\} \subset \mathbb{N}$ with $k_1 < \cdots < k_n$ and $\mu(k_i) = \lambda(i)$ for each i. If $S_1, \dots, S_r$ are structures of size $n_1, \dots, n_r$ with labeling functions $\lambda_1, \dots, \lambda_r$, then the sequence $S = (S_1, \dots, S_r)$ is a structure of size $n = \sum_i n_i$ and we say that a labeling λ of S is *compatible* with the λ_i if it is obtained by the combination of expansions of the λ_i, whose domains form a partition of $[\![1, n]\!]$. In particular, λ is not uniquely determined by the λ_i.

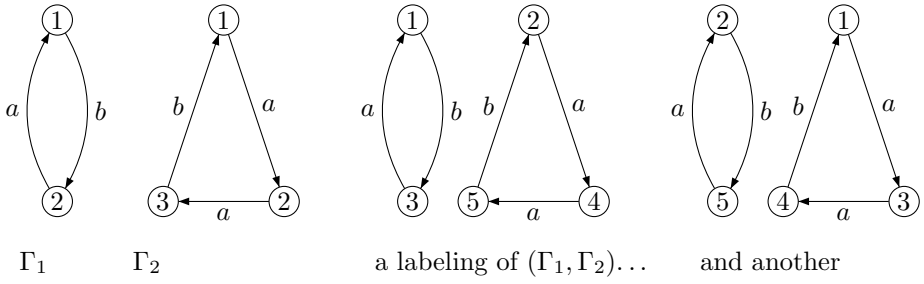


Figure 2: Two labeled A-graphs and two compatible labelings of the sequence they compose

Example 1.3 Figure 2 shows two labeled A-graphs of sizes 2 and 3, and several compatible labelings of the sequence they compose. □

We record here three such operations, that will be important for our purpose, and we refer the readers to [9, 10] for the proof of this important result.

Proposition 1.4 *Let $\mathcal{A}$ be a class of labeled structures with EGS $A(z)$.*

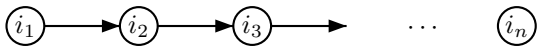
- *Let $\mathcal{B}$ be the class of sequences of structures from class $\mathcal{A}$ – that is, a size n labeled structure in $\mathcal{B}$ is a tuple $(S_1, \dots, S_r)$ of labeled structures in $\mathcal{A}$ (each S_i of size n_i with $n = \sum_i n_i$), equipped with a labeling function compatible with the labelings of the S_i. Then the EGS of $\mathcal{B}$ is*

$$A(z) = \frac{1}{1 - B(z)}.$$

- *Let $\mathcal{C}$ be the class of cycles of structures from class $\mathcal{A}$, where a cycle of structures is an equivalence class of sequences of structures in $\mathcal{A}$, under the relation which identifies a sequence with its cyclic permutations. Then the EGS of $\mathcal{C}$ is*
$$C(z) = \log\left(\frac{1}{1-B(z)}\right).$$

- *Let $\mathcal{D}$ be the class of sets of structures from class $\mathcal{A}$, where a set of structures is an equivalence class of sequences of structures in $\mathcal{A}$, under the relation which identifies a sequence with its permutations. Then the EGS of $\mathcal{D}$ is*
$$D(z) = \exp\big(B(z)\big).$$

Let us first apply this *calculus* to a simple example. Let $\mathcal{A}$ be the class consisting of a single graph, with one vertex and no edges (the vertex being necessarily labeled 1). Its EGS is $A(z) = z$. The class of labeled sequences of structures in $\mathcal{A}$ is in bijection with the class of labeled line graphs of the form with $\{i_1, \ldots, i_n\} = [\![1, n]\!]$, and its EGS is, according to Proposition 1.4, equal



to $\frac{1}{1-z} = \sum_n z^n$. This corresponds to the fact that the number of such size n sequences is $n!$ (so that its quotient by $n!$ is 1).

Similarly, the EGS of the class of cycles of structures in $\mathcal{A}$, that is, the class of labeled cyclic graphs (such as the graphs in Example 1.2 that are labeled with a single letter) is $\log\left(\frac{1}{1-z}\right)$.

We extend these examples in Section 2.1 to compute the EGS of labeled A-graphs.

1.2.3 A rejection algorithm

The enumeration and random generation of labeled A-graphs is however not our objective. We want to generate admissible pairs, that is, A-graphs with a distinguished vertex 1, that are connected and 1-trim. This will be achieved by a *rejection algorithm* (see [5]).

Suppose one wants to draw a number between 1 and 5, using a dice. It is natural to throw the dice repeatedly, until the result is different from 6. This is a semi-algorithm, since in the worst case it may never end — if we draw only 6's —, but we will loosely call it an algorithm.

Formally, suppose we want to generate elements of a set X, according to a probability law p_X. Suppose that X is a subset of Y, and that we have a probability law p_Y on Y, whose restriction to X is p_X. If we have an algorithm to generate elements of Y according to p_Y, we may use this algorithm to generate elements of X as follows: repeatedly draw an element of Y, reject it if it is not in X, stop if it is in X. The average complexity of such an algorithm depends on the complexity of the generating algorithm on Y, on the complexity of the

test whether an element of Y is in X, and on the average number of rejects. One can show that if $p_Y(X)$ is the probability for an element of Y to be in X, the average number of rejects is $1/p_Y(X)$.

Concretely, we will show that the probability p_n for a size n labeled A-graph to be connected and 1-trim (rigorously: $\lambda(1)$-trim) tends to 1 when n tends to infinity (Section 2.2 and 2.3). This justifies the use of a rejection algorithm since the average number of rejects tends to 0 when n tends to infinity.

There remains one problem: such a rejection algorithm will generate labeled connected 1-trim A-graphs, and we are interested only in the information contained in $(\Gamma, \lambda(1))$ – that is, we do not care how the $n-1$ vertices different from $\lambda(1)$ are labeled. In other words, an admissible pair, obtained from a labeled A-graph by forgetting the labeling of the vertices numbered 2 to n, will be counted several times. The following lemma shows that this is not an obstacle.

Lemma 1.5 *Let Γ be an A-graph of size n and let v be a vertex of Γ. If Γ is connected, then there are $(n-1)!$ isomorphism classes of labeled structures on Γ such that vertex v is labeled 1.*

Proof. Let $\bar{A} = \{\bar{a} \mid a \in A\}$ be a disjoint copy of A, and let $\tilde{\Gamma}$ be obtained from Γ by adding, for each edge (x, a, y), a new edge $(y, \bar{a}, x)$. It is immediately verified that $\tilde{\Gamma}$ is an $(A \cup \bar{A})$-graph. For each vertex $w \neq v$, let u_w be a finite word on the alphabet $A \cup \bar{A}$ labeling a path in $\tilde{\Gamma}$ from v to w. Then w is the unique vertex accessible from v following a path labeled u_w, and the words u_w are pairwise distinct. This observation guarantees that distinct labelings of Γ mapping 1 to v, are non isomorphic. □

Thus each size n admissible pair $(\Gamma, 1)$ is counted the same number of times, namely $(n-1)!$ times. Therefore, applying a rejection algorithm that randomly generates size n labeled connected 1-trim A-graphs and forgetting labels 2 to n also guarantees a random generation of admissible pairs for the uniform distribution on all admissible pairs of size n.

Summarizing the algorithmic strategy, we will randomly and equally likely generate a labeled A-graph, reject it if it is not connected and 1-trim, draw another labeled A-graph, etc, until we draw a connected 1-trim labeled A-graph. We then ignore the labeling of the vertices numbered 2 to n. Details on the algorithm and its complexity are discussed in Section 3.

For convenience, we will call a labeled A-graph Γ *admissible* if the pair $(\Gamma, 1)$ is admissible in the sense of Section 1.1.

2 Enumeration of A-graphs

We first observe that in an A-graph $\Gamma = (V, E)$, for each $a \in A$, the edges in E of the form (x, a, y) can be interpreted as the description of a partial injection from V to V (partial means that the domain of this injection is a subset of V). If the A-graph is labeled and has size n, each letter can therefore be interpreted

as a partial injection from $[\![1, n]\!]$ to itself. The labeled A-graph itself can then be seen as an A-tuple of partial injections.

In this section, we discuss the enumeration of partial injections on $[\![1, n]\!]$, the probability for an A-tuple of such partial injections to yield a labeled connected A-graph, the probability for that graph to be 1-trim, and finally the number of size n subgroups.

2.1 Partial injections and A-graphs

Each partial injection is a set of disjoint cycles and non-empty sequences (in analogy to the decomposition of a permutation as a union of cycles). The EGS for cycles is $\log(\frac{1}{1-z})$, and that for sequences is $\frac{1}{1-z}$ (Section 1.2.2). It follows that the EGS for non-empty sequences is $\frac{1}{1-z} - 1 = \frac{z}{1-z}$, and the EGS for the union of the (disjoint) classes of cycles and non-empty sequences is $\frac{z}{1-z} + \log(\frac{1}{1-z})$. Then Proposition 1.4 shows that the EGS for partial injections is

$$I(z) = \exp\left(\frac{z}{1-z} + \log\left(\frac{1}{1-z}\right)\right) = \frac{1}{1-z}\,\exp\left(\frac{z}{1-z}\right).$$

Let I_n be the number of partial injections from $[\![1, n]\!]$ to itself, so that $I(z) = \sum_n \frac{I_n}{n!} z^n$.

Remark 2.1 The series $I(z)$ turns out to be also the (ordinary) generating series of the average number of increasing subsequences in a random permutation. The first values of the sequence $(I_n)_{n\geq 0}$, referenced *EIS* **A002720** in [25] are

$$1,\ 2,\ 7,\ 34,\ 209,\ 1546,\ 13327,\ 130922,\ 1441729,\ 17572114,\ 234662231, \ldots$$

□

The above expression of $I(z)$ yields a simple recurrence relation for the sequence $(I_n)_n$. Indeed, we find that the series $I'(z) = \sum_{n\geq 0} \frac{I_{n+1}}{n!} z^n$ is equal to

$$I'(z) = \frac{2-z}{(1-z)^3}\,\exp\left(\frac{z}{1-z}\right) = \frac{2-z}{(1-z)^2}\,I(z).$$

Thus $(1-z)^2 I'(z) = (2-z)I(z)$ and the following recurrence relation follows:

$$\forall n \geq 2, \quad I_n = 2n\, I_{n-1} - (n-1)^2 I_{n-2}, \tag{1}$$

with $I_0 = 1$ and $I_1 = 2$.

Lemma 2.2 *For each integer $n \geq 1$, we have*

$$(n+1)! \ \leq\ (n+1)I_{n-1} \ \leq\ I_n \ \leq\ n\, e^{1/\sqrt{n}} I_{n-1} \ \leq\ n!\ e^{2\sqrt{n}-1}.$$

Proof. We proceed by induction on n, noting that $I_0 = 1$. The inequality $(n+1)I_{n-1} \leq I_n$ is verified for $n = 1$ and $n = 2$. Suppose that $n \geq 2$ and $(n+1)I_{n-1} \leq I_n$. By the recurrence relation, we have

$$\begin{aligned} I_{n+1} &= (2n+2)I_n - n^2 I_{n-1} &&= (2n+2)I_n - (n+1)^2 I_{n-1} + (2n+1)I_{n-1} \\ &&&\geq (2n+2)I_n - (n+1)I_n + (2n+1)I_{n-1} \\ &&&\geq (n+1)I_n + 2nI_{n-1} \\ &&&\geq (n+1)I_n + I_n = (n+2)I_n, \end{aligned}$$

with the last inequality derived from the recurrence relation on the I_n. Thus, for each $n \geq 1$, we have $I_n \geq (n+1)I_{n-1}$. The inequality $(n+1)! \leq (n+1)I_{n-1} \leq I_n$ follows immediately.

For $n \geq 1$ let $u_n = \frac{I_n}{I_{n-1}}$. We proceed by induction on n to prove that $u_n \leq ne^{1/\sqrt{n}}$, noting that $u_1 = 2 \leq e$. From Equation (1)

$$u_{n+1} = 2(n+1) - \frac{n^2}{u_n} \leq 2(n+1) - ne^{-1/\sqrt{n}}$$

It remains to show that

$$2(n+1) - ne^{-1/\sqrt{n}} \leq (n+1)e^{1/\sqrt{n+1}}.$$

or

$$2(n+1) \leq ne^{-1/\sqrt{n}} + (n+1)e^{1/\sqrt{n+1}}.$$

For any real number x,

$$e^x \geq 1 + x + \frac{x^2}{2} + \frac{x^3}{6}.$$

Therefore

$$ne^{-1/\sqrt{n}} + (n+1)e^{1/\sqrt{n+1}} \geq 2(n+1) + \sqrt{n+1} - \sqrt{n} + \frac{1}{6\sqrt{n+1}} - \frac{1}{6\sqrt{n}}$$

and we want to show that

$$\sqrt{n+1} - \sqrt{n} + \frac{1}{6\sqrt{n+1}} - \frac{1}{6\sqrt{n}} = \sqrt{n+1} - \sqrt{n} + \frac{\sqrt{n} - \sqrt{n+1}}{6\sqrt{n}\sqrt{n+1}} \geq 0,$$

or equivalently,

$$\begin{aligned} 6(n+1)\sqrt{n} - 6n\sqrt{n+1} + \sqrt{n} - \sqrt{n+1} &\geq 0 \\ (6n+7)\sqrt{n} &\geq (6n+1)\sqrt{n+1} \\ (6n+7)^2 n &\geq (6n+1)^2(n+1). \end{aligned}$$

Now the difference $\big((6n+7)^2 n\big) - \big((6n+1)^2(n+1)\big)$ is equal to $36n^2 + 36n - 1$, which is positive for all $n \geq 1$. This completes the proof that $u_n \leq ne^{1/\sqrt{n}}$ for all $n \geq 1$.

Consequently $I_n \leq n\, e^{1/\sqrt{n}} I_{n-1}$ and

$$I_n = \frac{I_n}{I_0} = \prod_{i=1}^{n} u_i \leq n!\, e^{\sum_{i=1}^{n} \frac{1}{\sqrt{i}}}.$$

As the function $x \mapsto \frac{1}{\sqrt{x}}$ is decreasing on the positive domain, we find that $\frac{1}{\sqrt{i+1}} \leq \int_i^{i+1} \frac{dx}{\sqrt{x}}$ for each $i \geq 1$, and

$$\sum_{i=2}^{n} \frac{1}{\sqrt{i}} = \sum_{i=1}^{n-1} \frac{1}{\sqrt{i+1}} \leq \int_1^n \frac{dx}{\sqrt{x}} = 2\sqrt{n} - 2.$$

Thus $e^{\sum_{i=1}^{n} \frac{1}{\sqrt{i}}} \leq e^{2\sqrt{n}-1}$, which concludes the proof. □

Remark 2.3 The computation of $I(z)$ allows us to justify our assertion that Boltzmann samplers are less efficient than the random generation based on the recursive method, see Remark 1.1. More precisely, the behavior of Boltzmann samplers is often such that the size of the generated object sits between $(1-\varepsilon)n$ and $(1+\varepsilon)n$ with high probability. In our case, it is essential that the tuple of partial injections we generate all have the same size. It is often the case that a Boltzmann sampler can be used to produce an exact-size sampler, using a rejection algorithm. In the case of partial injections however, as the distribution of the sizes of partial injections is not sufficiently concentrated around the mean size, each draw of a random partial injection of size exactly n takes times $\mathcal{O}(n^{7/4})$, which is not very satisfactory. Here is why.

The mean size of a partial injection produced under the exponential Boltzmann model is (see [7, Proposition 1]):

$$\mathbb{E}_x(\text{size of a partial injection}) = x\frac{I'(x)}{I(x)} = x\frac{2-x}{(1-x)^2}$$

and its variance is

$$\sigma_x^2(\text{size of a partial injection}) = \frac{d}{dx}\mathbb{E}_x(\text{size of a partial injection}) = \frac{2}{(1-x)^3}.$$

Thus, to generate partial injections of expected size $\mathbb{E}_x = n$, one has to choose $x = 1 - 1/\sqrt{n+1}$. In this case , $\sigma_x^2 = 2(n+1)^{3/2}$. From [7, Theorem 4] dealing with H-admissible generating functions (see Section 2.3.2) the exact-size generation requires $\sqrt{2\pi}\sigma_x = \mathcal{O}(n^{3/4})$ rejections in average and the overall cost of exact-size sampling is $\mathcal{O}(n\sigma_x) = \mathcal{O}(n^{7/4})$ in average. □

As discussed at the beginning of this section, if $r = |A|$, then a labeled A-graph of size n can be assimilated to a r-tuple of partial injections on $[\![1, n]\!]$, so the EGS of labeled A-graphs is

$$\sum_{n\geq 0} \frac{I_n^r}{n!}\, z^n.$$

2.2 Connectedness

Recall that an A-graph is connected if the underlying undirected graph is connected. In this section, we show the following result.

Theorem 2.4 *Let A be an alphabet of cardinality $r \geq 2$ and let p_n be the probability for an n-vertex labeled A-graph to be connected. Then $\lim_{n\mapsto\infty} p_n = 1$ and more precisely, $p_n = 1 - \frac{2^r}{n^{r-1}} + o(\frac{1}{n^{r-1}})$.*

In particular, this shows that labeled A-graphs are asymptotically connected. The proof of Theorem 2.4, given below, relies on the following theorem, due to Bender (see [1, p. 497] for a survey and [2] for a complete proof).

Theorem 2.5 *Let $F(z, y)$ be a two-variable real function which is analytic at $(0, 0)$. Let $J(z) = \sum_{n>0} j_n z^n$, $C(z) = \sum_{n>0} c_n z^n$ and $D(z) = \sum_{n>0} d_n z^n$ be functions such that*

$$C(z) = F(z, J(z)) \quad \text{and} \quad D(z) = \frac{\partial F}{\partial y}(z, J(z)).$$

If the sequence $(j_n)_{n>0}$ satisfies $j_{n-1} = o(j_n)$, and if for some $s \geq 1$, we have $\sum_{k=s}^{n-s} |j_k j_{n-k}| = \mathcal{O}(j_{n-s})$, then

$$c_n = \sum_{k=0}^{s-1} d_k j_{n-k} + \mathcal{O}\left(j_{n-s}\right).$$

Proof of Theorem 2.4. Let $J(z) = \sum_{n\geq 1} j_n z^n$ where $j_n = \frac{I_n^r}{n!}$. Then the EGS of labeled A-graphs is $1 + J(z)$.

Decomposing these graphs into their connected components (connected components of the underlying undirected graph) and using Proposition 1.4, we find that $1 + J(z) = \exp(C(z))$, where $C(z) = \sum_{k\geq 1} c_k z^k$, with $c_k = \frac{C_k}{k!}$ and C_k is the number of connected labeled A-graphs with k vertices.

It follows that $C(z) = \log(1 + J(z))$. We note that the map $F(z, y) = \log(1 + y)$ is analytic at $(0, 0)$ and we let

$$D(z) = \frac{\partial F}{\partial y}(z, J(z)) = \frac{1}{1 + J(z)}.$$

By Lemma 2.2, $I_{n-1} \leq \frac{I_n}{n+1}$, and therefore

$$j_{n-1} = \frac{I_{n-1}^r}{(n-1)!} \leq \frac{I_n^r}{(n+1)^r (n-1)!} \leq \frac{I_n^r}{n^{r-1}\, n!} = \frac{j_n}{n^{r-1}}.$$

In particular, we have

$$j_{n-1} = \mathcal{O}\left(\frac{j_n}{n^{r-1}}\right) = o\left(j_n\right). \tag{2}$$

We now want to verify whether $\sum_{k=s}^{n-s} j_k j_{n-k} = \mathcal{O}(j_{n-s})$ (this is the last hypothesis of Theorem 2.5, for a fixed $s \geq 1$). Let S be the sum above. By symmetry, we get

$$S = \sum_{k=s}^{n-s} \frac{I_k^r}{k!} \frac{I_{n-k}^r}{(n-k)!} \leq 2 \left(\sum_{k=s}^{\lfloor n/2 \rfloor} \frac{I_k^r}{k!} \frac{I_{n-k}^r}{(n-k)!} \right).$$

We show that for n large enough, the finite sequence $\left(\frac{I_k^r}{k!} \frac{I_{n-k}^r}{(n-k)!} \right)_{s \leq k \leq \lfloor n/6 \rfloor}$ is decreasing. From Equation (1), we have

$$\frac{I_{k+1}}{I_k} \;=\; 2(k+1) - k^2 \frac{I_{k-1}}{I_k} \;\leq\; 2(k+1);$$

and by Lemma 2.2, we have $\frac{I_{n-(k+1)}}{I_{n-k}} \leq \frac{1}{n-k+1}$. Therefore

$$\frac{I_{k+1}^r I_{n-(k+1)}^r}{I_k^r I_{n-k}^r} \leq \frac{2^r (k+1)^r}{(n-k+1)^r}.$$

Moreover $\frac{k!}{(k+1)!} \frac{(n-k)!}{(n-k-1)!} = \frac{n-k}{k+1}$, and it follows that

$$\begin{aligned} \frac{I_{k+1}^r I_{n-(k+1)}^r}{(k+1)!(n-k-1)!} \frac{k!(n-k)!}{I_k^r I_{n-k}^r} &\leq \frac{2^r (k+1)^r}{(n-k+1)^r} \frac{n-k}{k+1} \\ &\leq \frac{2^r (k+1)^{r-1}}{(n-k+1)^{r-1}} \frac{n-k}{n-k+1} \\ &\leq \frac{2^r (k+1)^{r-1}}{(n-k+1)^{r-1}}. \end{aligned}$$

This value is less than or equal to 1 when $\frac{k+1}{n-k+1}$ is less than or equal to $c = 2^{-\frac{r}{r-1}}$. Since $r \geq 2$, we have $\frac{1}{4} \leq c \leq \frac{1}{2}$, and for any $k \leq \frac{n-3}{5}$, $\frac{k+1}{n-k+1} \leq \frac{1}{4} \leq c$. For any $n \geq 18$, $\lfloor \frac{n}{6} \rfloor \leq \frac{n-3}{5}$, thus the sequence is decreasing on the domain $s \leq k \leq \lfloor \frac{n}{6} \rfloor$.

$$\sum_{k=s+1}^{\lfloor n/6 \rfloor} \frac{I_{n-k}^r}{(n-k)!} \frac{I_k^r}{k!} \;\leq\; (\lfloor n/6 \rfloor - s) \frac{I_{s+1}^r}{(s+1)!} \frac{I_{n-s-1}^r}{(n-s-1)!}$$

From Lemma 2.2, $I_{n-s-1}^r \leq \frac{1}{(n-s+1)^r} I_{n-s}^r$ and $I_{s+1}^r \leq (s+1)^r e^{r/\sqrt{s+1}} I_s^r$. Therefore

$$\begin{aligned} \frac{I_{n-s-1}^r}{(n-s-1)!} \frac{I_{s+1}^r}{(s+1)!} &\leq \frac{(s+1)^r e^{r/\sqrt{s+1}} I_s^r I_{n-s}^r}{(s+1)!\,(n-s-1)!\,(n-s+1)^r} \\ &\leq \frac{I_{n-s}^r}{(n-s)!} \frac{I_s^r\,(s+1)^{r-1}\,e^{r/\sqrt{s+1}}}{s!\,(n-s+1)^{r-1}} \end{aligned}$$

and we have

$$\sum_{k=s+1}^{\lfloor n/6\rfloor} \frac{I_{n-k}^r}{(n-k)!}\,\frac{I_k^r}{k!} \;\leq\; (\lfloor n/6\rfloor - s)\frac{I_{n-s}^r}{(n-s)!}\;\frac{I_s^r\,(s+1)^{r-1}\,e^{r/\sqrt{s+1}}}{s!\;(n-s+1)^{r-1}},$$

which is $\mathcal{O}\left(\frac{I_{n-s}^r}{(n-s)!}n^{-(r-2)}\right)$. Note that it is $\mathcal{O}\left(\frac{I_{n-s}^r}{(n-s)!}\right)$ when $r=2$.

We now study the remaining part of the sum S. It follows from Lemma 2.2 that if $k \geq s$, then $I_{n-k} \leq \frac{(n-(k-1))!}{(n-(s-1))!}I_{n-s}$, so we have

$$\begin{aligned}\frac{I_{n-k}^r}{(n-k)!} &\leq \frac{(n-(k-1))!^{r-1}}{(n-(s-1))!^{r-1}}\frac{(n-k+1)}{(n-s+1)}\frac{I_{n-s}^r}{(n-s)!}\\ &= \frac{(n-k)!^{r-1}}{n!^{r-1}}(n-k+1)^{r-1}\prod_{i=0}^{s-2}(n-i)^{r-1}\,\frac{(n-k+1)}{(n-s+1)}\frac{I_{n-s}^r}{(n-s)!}.\end{aligned}$$

By Lemma 2.2 again, we have $I_k \leq k!\;e^{2\sqrt{k}-1}$, and hence

$$\frac{I_k^r}{k!} \leq k!^{r-1}e^{(2\sqrt{k}-1)r}$$

It follows that

$$\frac{I_{n-k}^r}{(n-k)!}\,\frac{I_k^r}{k!} \leq \binom{n}{k}^{-(r-1)}\frac{(n-k+1)^r}{n-s+1}\prod_{i=0}^{s-2}(n-i)^{r-1}e^{(2\sqrt{k}-1)r}\frac{I_{n-s}^r}{(n-s)!}.$$

Note that for $s=1$, $\prod_{i=0}^{s-2}(n-i)^{r-1}=1$.

Now observe that for each $s \leq k \leq n/2$, $\frac{(n-k+1)^r}{n-s+1}\prod_{i=0}^{s-2}(n-i)^{r-1} < n^{s(r-1)}$. We get

$$\sum_{k=\lfloor n/6\rfloor+1}^{\lfloor n/2\rfloor} \frac{I_{n-k}^r}{(n-k)!}\,\frac{I_k^r}{k!} \;\leq\; n^{s(r-1)}\,\frac{I_{n-s}^r}{(n-s)!}\sum_{k=\lfloor n/6\rfloor+1}^{\lfloor n/2\rfloor}\binom{n}{k}^{-(r-1)}e^{(2\sqrt{k}-1)r},$$

For any k such that $\lfloor n/6\rfloor+1 \leq k \leq \lfloor n/2\rfloor$ we have $\binom{n}{k} \geq \binom{n}{\lfloor n/6\rfloor}$. Using Stirling formula ($n! \sim \sqrt{2\pi}e^{-n}n^{n+\frac{1}{2}}$) we get

$$\begin{aligned}\binom{n}{\lfloor n/6\rfloor}^{-1} &\sim \sqrt{2\pi}\frac{\lfloor n/6\rfloor^{\lfloor n/6\rfloor+1/2}}{n^{n+1/2}}(n-\lfloor n/6\rfloor)^{n-\lfloor n/6\rfloor+1/2}\\ &\sim \sqrt{2\pi n}\left(\frac{\lfloor n/6\rfloor}{n}\right)^{\lfloor n/6\rfloor+1/2}\left(1-\frac{\lfloor n/6\rfloor}{n}\right)^{n-\lfloor n/6\rfloor+1/2}\end{aligned}$$

Since $\frac{\lfloor n/6\rfloor}{n} \leq \frac{1}{6}$ and $1-\frac{\lfloor n/6\rfloor}{n} < 1$, there exists $0<C<1$ such that $\binom{n}{\lfloor n/6\rfloor}^{-1} \leq C^n$ for n large enough. Hence,

$$\begin{aligned}\sum_{k=\lfloor n/6\rfloor+1}^{\lfloor n/2\rfloor}\binom{n}{k}^{-(r-1)}e^{(2\sqrt{k}-1)r} &\leq \frac{n}{3}\binom{n}{\lfloor n/6\rfloor}^{-(r-1)}e^{(2\sqrt{\lfloor n/2\rfloor}-1)r}\\ &\leq \frac{n}{3}\,C^{(r-1)n}\,e^{r\sqrt{2n}}\end{aligned}$$

In particular, for any D such that $C < D < 1$ we have

$$\sum_{k=\lfloor n/6\rfloor+1}^{\lfloor n/2\rfloor} \binom{n}{k}^{-(r-1)} e^{(2\sqrt{k}-1)r} = \mathcal{O}\left(D^{(r-1)n}\right).$$

Consequently

$$\sum_{k=\lfloor n/6\rfloor+1}^{\lfloor n/2\rfloor} \frac{I_{n-k}^r}{(n-k)!}\frac{I_k^r}{k!} = \frac{I_{n-s}^r}{(n-s)!}\,\mathcal{O}\left(n^{s(r-1)}D^{(r-1)n}\right) \quad \text{where } 0 < D < 1$$

Finally we find that

$$\begin{aligned} S = \sum_{k=s}^{n-s} \frac{I_k^r}{k!}\frac{I_{n-k}^r}{(n-k)!} &= 2\frac{I_{n-s}^r}{(n-s)!}\left(\frac{I_s^r}{s!} + \mathcal{O}\left(n^{-(r-2)}\right) + \mathcal{O}\left(n^{s(r-1)}D^{(r-1)n}\right)\right) \\ &= 2\frac{I_{n-s}^r}{(n-s)!}\mathcal{O}\left(1\right). \end{aligned}$$

Hence $\sum_{k=s}^{n-s} j_k j_{n-k} = \mathcal{O}(j_{n-s})$. Thus we can apply Theorem 2.5 for any fixed positive integer s, it yields

$$c_n = \sum_{k=0}^{s-1} d_k j_{n-k} + \mathcal{O}\left(j_{n-s}\right).$$

Since $d_0 = 1$ and $d_1 = -j_1 = -I_1^r = -2^r$, we get that

$$c_n = j_n - 2^r j_{n-1} + \mathcal{O}\left(j_{n-2}\right).$$

Now Equation (2) above yields

$$j_{n-2} = \mathcal{O}\left(\frac{j_{n-1}}{n^{r-1}}\right) = \mathcal{O}\left(\frac{j_n}{n^{2(r-1)}}\right),$$

and the independent technical Proposition 2.10 below yields

$$\frac{I_n}{n!} = \frac{e^{-1/2}}{2\sqrt{\pi}}\, n^{-1/4} e^{2\sqrt{n}}(1 + o(1)).$$

Therefore

$$\frac{I_{n-1}}{I_n} = \frac{1}{n}\left(1 - \frac{1}{n}\right)^{-1/4} e^{2\sqrt{n}\left(\sqrt{1-\frac{1}{n}}-1\right)}(1 + o(1)) = \frac{1}{n}(1 + o(1))$$

and

$$\frac{I_{n-1}^r}{I_n^r} = \frac{1}{n^r}(1 + o(1)).$$

Finally

$$j_{n-1} = \frac{I^r_{n-1}}{(n-1)!} = \frac{I^r_n}{n!}\,\frac{1}{n^{r-1}}\,(1+o(1)) = \frac{j_n}{n^{r-1}}\,(1+o(1)).$$

We conclude that $c_n = j_n\left(1-\frac{2^r}{n^{r-1}}+o\left(\frac{1}{n^{r-1}}\right)\right)$ or

$$C_n \;=\; I^r_n\left(1-\frac{2^r}{n^{r-1}}+o\left(\frac{1}{n^{r-1}}\right)\right). \tag{3}$$

Recall that we denote by p_n the probability for an n-vertex graph whose transitions are defined by an r-tuple of partial injections on $[\![1,n]\!]$ to be connected. Equation 3 shows that

$$p_n \;=\; \frac{C_n}{I^r_n} \;=\; 1-\frac{2^r}{n^{r-1}}+o\left(\frac{1}{n^{r-1}}\right),$$

which concludes the proof of Theorem 2.4. □

2.3 1-trimness

Recall that for a labeled A-graph Γ to be admissible, Γ must be 1-trim, that is, no vertex $v \neq 1$ may be a leaf (see Section 1.2.3, and also Section 1.1). Moreover, vertex v is a leaf if and only if it has less than 2 images or preimages in the $|A|$ partial injections that define Γ.

In this section, we show the following result.

Theorem 2.6 *Let A be an alphabet of cardinality at least 2. The probability for an n-vertex labeled A-graph to have no leaf is $1+o(1)$.*

We immediately record the following corollary, which gives our random generation strategy its final justification.

Corollary 2.7 *Let A be an alphabet of cardinality at least 2. The probability for a given size n labeled A-graph Γ to be admissible, is $1+o(1)$.*

Proof. In view of Theorems 2.4 and 2.6, an n-vertex labeled A-graph is connected with probability $1+\mathcal{O}\left(\frac{1}{n}\right)$, and without leaves (in particular: 1-trim) with probability $1+o(1)$. It follows that the probability for an n-vertex labeled A-graph to be connected and without leaves is

$$\left(1+\mathcal{O}\left(\frac{1}{n}\right)\right)\;(1+o(1)) = 1+o(1),$$

and the result follows. □

The rest of this section is devoted to the proof of Theorem 2.6. We first observe that it suffices to establish this result when $|A| = 2$: indeed, if $A = \{a_1,\ldots,a_r\}$ with $r > 2$ and if the $\{a_1,a_2\}$-part of an A-graph has no leaf, then neither does the whole A-graph. So we assume that $r = 2$ for the rest of this section.

2.3.1 Number of sequences

We note that a leaf in an A-graph forms a length 1 sequence in the functional graph of one of the partial injections, and is an endpoint of a sequence in the functional graph of the other.

We start with a study of the parameter X_n, which counts the number of sequences in the functional graph of a partial injection on $[\![1, n]\!]$. We will compute the expectation and the variance of X_n in Section 2.3.3. Note that the number of endpoints of sequences in the functional graph of a random partial injection is bounded above by the quantity accounted for by the parameter $2X_n$.

We first introduce the bivariate series $J(z, u) = \sum_{n,k} \frac{J_{n,k}}{n!} z^n u^k$, where $J_{n,k}$ denotes the number of partial injections on $[\![1, n]\!]$, whose functional graph has k sequences (that is, such that $X_n = k$). The EGS of non-empty sequences was already computed and it is equal to $\frac{z}{1-z}$ or, in this multivariate setting, $\frac{zu}{1-z}$. The EGS of cycles is $\log \frac{z}{1-z}$ (also in this multivariate setting). The multivariate analogue of Proposition 1.4 (see Flajolet and Sedgewick [10, ex. 7, section III.3]) then shows that

$$J(z, u) = \exp\left(\frac{zu}{1-z} + \log\left(\frac{1}{1-z}\right)\right) = \frac{1}{1-z}\exp\left(\frac{zu}{1-z}\right).$$

In particular, $J(z, 1) = I(z)$.

The expectation of variables X_n and X_n^2 are

$$\mathbb{E}(X_n) = \frac{\sum_k k J_{n,k}}{\sum_k J_{n,k}} \quad \text{and} \quad \mathbb{E}(X_n^2) = \frac{\sum_k k^2 J_{n,k}}{\sum_k J_{n,k}}.$$

Now $\sum_k J_{n,k}$ is the coefficient of z^n in $J(z, 1)$ and $\sum_k k J_{n,k}$ is the coefficient of z^n in $\frac{\partial}{\partial u} J(z, u)\big|_{u=1}$. The coefficient of z^n in $\frac{\partial^2}{\partial u^2} J(z, u)\big|_{u=1}$ is $\sum_k k(k-1) J_{n,k}$.

Let

$$H_p(z) = \frac{1}{(1-z)^p} I(z) = \frac{1}{(1-z)^{p+1}} \exp\left(\frac{z}{1-z}\right).$$

Then we have

$$\begin{aligned}
J(z, 1) &= I(z) = H_0(z), \\
\frac{\partial}{\partial u} J(z, u)\Big|_{u=1} &= \frac{z}{(1-z)^2}\exp\left(\frac{z}{1-z}\right) = \frac{z}{1-z} I(z) = z H_1(z), \\
\frac{\partial^2}{\partial u^2} J(z, u)\Big|_{u=1} &= \frac{z^2}{(1-z)^3}\exp\left(\frac{z}{1-z}\right) = \frac{z^2}{(1-z)^2} I(z) = z^2 H_2(z).
\end{aligned}$$

For convenience, if $S(z)$ is a formal power series, we let $[z^n]S(z)$ denote the coefficient of z^n in $S(z)$. Then the expectation of variables X_n and X_n^2 are given by

$$\mathbb{E}(X_n) = \frac{[z^n] z H_1(z)}{[z^n] I(z)} = \frac{[z^{n-1}] H_1(z)}{[z^n] H_0(z)}, \tag{4}$$

$$\mathbb{E}(X_n^2) = \frac{[z^n] z^2 H_2(z)}{[z^n] I(z)} + \frac{[z^n] z H_1(z)}{[z^n] I(z)} = \frac{[z^{n-2}] H_2(z)}{[z^n] H_0(z)} + \frac{[z^{n-1}] H_1(z)}{[z^n] H_0(z)}. \tag{5}$$

The variance of X_n is

$$\sigma^2(X_n) \quad = \quad \mathbb{E}(X_n^2) - \mathbb{E}(X_n)^2. \tag{6}$$

Thus finding asymptotic estimates of $\mathbb{E}(X_n)$ and $\sigma^2(X_n)$ requires finding estimates of the coefficients of the functions $H_p(z)$ for $p = 0, 1, 2$.

2.3.2 Saddlepoint asymptotics

Saddlepoint analysis is a powerful method to find asymptotic estimates of the coefficients of analytic functions which exhibit exponential-type growth in the neighborhood of their singularities. We refer the reader to the books by Flajolet and Sedgewick [9] and [10, Chap. VIII], and to the survey by Odlyzko [23] for a thorough presentation of saddlepoint analysis.

The fast growth of the coefficients of $H_p(z)$ justifies the application of saddlepoint analysis. The theorem we want to use, Theorem 2.9 below, requires an additional hypothesis, namely the H-admissibility of the functions H_p [10, Section VIII.5]. We now verify that this rather technical condition is satisfied.

Let $f(z)$ be a function that is analytic at the origin, with radius of convergence ρ, positive on $]0, \rho[$. Put $f(z)$ into its exponential form $f(z) = e^{h(z)}$ and let

$$a(r) = rh'(r) \quad \text{and} \quad b(r) = r^2 h''(r) + rh'(r).$$

The function $f(z)$ is said to be *H-admissible* if there exists a function $\delta :]0, \rho[\longrightarrow]0, \pi[$ such that the following three conditions hold:

(H1) $\lim_{r\to\rho} b(r) = +\infty$.

(H2) Uniformly for $|\theta| \le \delta(r)$

$$f(re^{i\theta}) \sim f(r)e^{i\theta a(r) - \frac{1}{2}\theta^2 b(r)} \quad \text{when } r \text{ tends to } \rho.$$

[That is, $f(re^{i\theta}) = f(r)e^{i\theta a(r) - \frac{1}{2}\theta^2 b(r)}(1 + \gamma(r, \theta))$ with $|\gamma(r, \theta)| \le \tilde{\gamma}(r)$ when $|\theta| < \delta(r)$ and $\lim_{r\to\rho} \tilde{\gamma}(r) = 0$.]

(H3) and uniformly for $\delta(r) \le |\theta| \le \pi$

$$f(re^{i\theta})\sqrt{b(r)} = o(f(r)) \qquad \text{when } r \text{ tends to } \rho.$$

Lemma 2.8 *The functions $H_0(z)$, $H_1(z)$ and $H_2(z)$ are H-admissible.*

Proof. First it is elementary to verify that $H_p(z) = \frac{1}{(1-z)^{p+1}} \exp\left(\frac{z}{1-z}\right)$ $(p = 0, 1, 2)$ is analytic at the origin, with radius of convergence $\rho = 1$, and is positive on the real segment $]0, \rho[$. Following the definition of H-admissibility above, we find that $H_p(z) = e^{h_p(z)}$ with $h_p(z) = \frac{z}{1-z} - (p+1)\log(1-z)$, so that

$$a_p(r) = r\frac{(p+2) - (p+1)r}{(1-r)^2} \quad \text{and} \quad b_p(r) = r\frac{(p+2) - pr}{(1-r)^3}. \tag{7}$$

Therefore Condition (H1) is satisfied.

Let $\delta(r) = (1-r)^{17/12}$ (for a discussion on the choice of δ, we refer the readers to [10, Chap. VIII]). For θ small enough, one can expand $h_p(re^{i\theta})$ into

$$h_p(re^{i\theta}) = h_p(r) + \sum_{m=1}^{\infty} \alpha_m(r) \frac{(i\theta)^m}{m!},$$

with $\alpha_m(r) = r\frac{d}{dr}\alpha_{m-1}(r)$ and $\alpha_0(r) = h_p(r)$ (by definition of a Taylor development). In particular, we find

$$\begin{aligned} \alpha_1(r) &= a_p(r), \\ \alpha_2(r) &= b_p(r), \quad \text{and} \\ \alpha_3(r) &= \frac{r}{(1-r)^4}\left((2+p) + 4r - pr^2\right). \end{aligned}$$

Thus, as r tends towards 1, for $|\theta| \leq \delta(r)$, $\alpha_3(r)\theta^3 = \mathcal{O}\left((1-r)^{1/4}\right) = o(1)$. More generally, $\alpha_m(r)\theta^m = \mathcal{O}\left((1-r)^{1/m+1}\right) = o(\alpha_{m-1}(r))$. Therefore, uniformly for $|\theta| \leq \delta(r)$

$$h_p(re^{i\theta}) = h_p(r) + i\theta a(r) - \frac{1}{2}\theta^2 b(r) + o(1)$$

and Condition (H2) follows.

Finally, we have

$$\begin{aligned} |H_p(re^{i\theta})| &= \frac{1}{|1-re^{i\theta}|^{p+1}} \exp\left(\Re\left(\frac{re^{i\theta}}{1-re^{i\theta}}\right)\right) \\ &= \frac{1}{(1+r^2-2r\cos\theta)^{(p+1)/2}} \exp\left(\frac{r(\cos\theta - r)}{1+r^2-2r\cos\theta}\right). \end{aligned}$$

We observe that for $r > 0$, $r(\cos\theta - r)$ and $(1+r^2-2r\cos\theta)^{-1}$ are decreasing functions of θ on $]0,\pi[$. Thus, for each $r > 0$ and $\delta(r) \leq |\theta| < \pi$, $|H_p(re^{i\theta})|$ is bounded above by $|H_p(re^{i\delta(r)})|$, namely

$$|H_p(re^{i(1-r)^{\frac{17}{12}}})| = \frac{\exp\left(\frac{r(\cos(1-r)^{\frac{17}{12}} - r)}{1+r^2-2r\cos(1-r)^{\frac{17}{12}}}\right)}{(1+r^2-2r\cos(1-r)^{\frac{17}{12}})^{(p+1)/2}}$$

In the neighborhood of 0, $\cos\theta = 1 - \frac{1}{2}\theta^2 + \mathcal{O}(\theta^4)$. So when r tends towards 1, we have

$$\begin{aligned} \cos(1-r)^{17/12} &= 1 - \frac{1}{2}(1-r)^{17/6} + \mathcal{O}\left((1-r)^{17/3}\right), \\ 1 + r^2 - 2r\cos(1-r)^{17/12} &= 1 + r^2 - 2r + r(1-r)^{17/6} + \mathcal{O}\left((1-r)^{17/3}\right) \\ &= (1-r)^2\left(1 + r(1-r)^{5/6} + \mathcal{O}\left((1-r)^{11/3}\right)\right), \\ \frac{1}{1+r^2-2r\cos(1-r)^{17/12}} &= \frac{(1-r)^{-2}}{1 + r(1-r)^{5/6} + \mathcal{O}\left((1-r)^{11/3}\right)} \\ &= (1-r)^{-2}\left(1 - r(1-r)^{5/6} + \mathcal{O}\left((1-r)^{5/3}\right)\right). \end{aligned}$$

Moreover

$$\begin{aligned}\frac{r(\cos(1-r)^{\frac{17}{12}}-r)}{1+r^2-2r\cos(1-r)^{\frac{17}{12}}} &= \frac{r}{(1-r)^2}\left(1-r-\frac{1}{2}(1-r)^{17/6}+\mathcal{O}\left((1-r)^{17/3}\right)\right)\\ &\qquad \left(1-r(1-r)^{5/6}+\mathcal{O}\left((1-r)^{5/3}\right)\right)\\ &= \frac{r}{1-r}\left(1-\frac{1}{2}(1-r)^{11/6}+\mathcal{O}\left((1-r)^{14/3}\right)\right)\\ &\qquad \left(1-r(1-r)^{5/6}+\mathcal{O}\left((1-r)^{5/3}\right)\right)\\ &= \frac{r}{1-r}\left(1-(1-r)^{5/6}+\mathcal{O}\left((1-r)^{5/3}\right)\right).\end{aligned}$$

It follows that

$$\begin{aligned}|H_p(re^{i(1-r)^{17/12}})| &= (1-r)^{-(p+1)}\left(1-r(1-r)^{5/6}+\mathcal{O}\left((1-r)^{5/3}\right)\right)^{(p+1)/2}\\ &\qquad \exp\left(\frac{r}{1-r}\left(1-(1-r)^{5/6}+\mathcal{O}\left((1-r)^{5/3}\right)\right)\right).\end{aligned}$$

As a result,

$$\begin{aligned}|H_p(re^{i\theta})| &\le (1-r)^{-(p+1)}(1+\mathcal{O}(1-r)^{5/6})\\ &\qquad \exp\left(\frac{r}{1-r}\left(1-(1-r)^{5/6}+\mathcal{O}\left((1-r)^{5/3}\right)\right)\right),\\ |H_p(re^{i\theta})| &\le \left(H_p(r)+\exp\left(\frac{r}{1-r}\right)\mathcal{O}((1-r)^{\frac{5}{6}-(p+1)})\right)\\ &\qquad \exp\left(-\frac{r}{(1-r)^{\frac{1}{6}}}+\mathcal{O}((1-r)^{\frac{2}{3}})\right),\end{aligned}$$

and hence

$$|H_p(re^{i\theta})|\sqrt{b_p(r)}=o(H_p(r)),$$

that is, Condition (H3) is satisfied. □

We now want to use the following theorem [10, Theorem VIII.5].

Theorem 2.9 (coefficients of H-admissible functions) *Let $f(z)$ be a H-admissible function and $\zeta=\zeta(n)$ be the unique solution in the interval $]0,\rho[$ of the saddlepoint equation*

$$\zeta\frac{f'(\zeta)}{f(\zeta)}=n.$$

Then

$$[z^n]f(z)=\frac{f(\zeta)}{\zeta^n\sqrt{2\pi b(\zeta)}}\left(1+o(1)\right).$$

where $b(z)=z^2h''(z)+zh'(z)$ and $h(z)=\log f(z)$.

Let us record immediately an application of this result.

Proposition 2.10 *With the notation above, for $p = 0, 1, 2$,*

$$[z^n]H_p(z) = \frac{e^{-1/2}}{2\sqrt{\pi}}\, n^{p/2-1/4} e^{2\sqrt{n}}(1 + o(1)).$$

Proof. For any positive integer n the saddle point $\zeta_p(n)$ is the least positive solution of $z\frac{H_p'(z)}{H_p(z)} = n$, that is, the least positive solution of

$$(n + p + 1)z^2 - (2n + p + 2)z + n = 0$$

and it follows that

$$\zeta_p(n) = 1 - \frac{p + \sqrt{4n + (p + 2)^2}}{2(n + p + 1)} = 1 - \frac{1}{\sqrt{n}} - \frac{p}{2n} + \mathcal{O}\left(\frac{1}{n^{3/2}}\right).$$

Moreover from Equation (7), the function $b_p(r)$ is $b_p(r) = r\frac{(p+2)-pr}{(1-r)^3}$. Therefore

$$[z^n]H_p(z) = \frac{e^{\frac{\zeta_p}{1-\zeta_p}}}{\sqrt{2\pi\left((p + 2) - p\zeta_p\right)}}\, \frac{1}{\zeta_p^n(1 - \zeta_p)^p} \sqrt{\frac{1 - \zeta_p}{\zeta_p}}\,(1 + o(1)).$$

Now $\sqrt{\frac{1-\zeta_p}{\zeta_p}} = n^{-\frac{1}{4}}(1+o(1))$, $(1-\zeta_p)^p = n^{-\frac{p}{2}}(1+o(1))$ and $\sqrt{2\pi\left((p + 2) - p\zeta_p\right)} = 2\sqrt{\pi}(1+o(1))$. Finally $\frac{\zeta_p}{1-\zeta_p} = (\sqrt{n}-1-\frac{p}{2})(1+o(1))$ and $\zeta_p^{-n} = \exp(-n\log\zeta_p) = \exp(\sqrt{n} + \frac{p+1}{2})(1 + o(1))$. So

$$\begin{aligned}
[z^n]H_p(z) &= \frac{e^{\sqrt{n}-1-p/2}}{2\sqrt{\pi}}\, e^{\sqrt{n}+(p+1)/2} n^{p/2-1/4}(1 + o(1)) \\
&= \frac{e^{-1/2}}{2\sqrt{\pi}}\, n^{p/2-1/4} e^{2\sqrt{n}}(1 + o(1)).
\end{aligned}$$

□

2.3.3 Expected value and standard deviation of X_n

We now conclude the study of the expected value and the standard deviation of the number of sequences in a random partial injection.

Lemma 2.11 *The expected number $\mathbb{E}(X_n)$ of sequences in a random partial injection of size n is asymptotically equal to $\sqrt{n}$ with standard deviation $o(\sqrt{n})$.*

Proof. Recall that the expected values of the random variable X_n and X_n^2 and the variance of X_n are given in Equations (4), (5) and (6) in Section 2.3.1. We can use Proposition 2.10 to estimate these quantities.

From Equation (4) we find that

$$\mathbb{E}(X_n) = \frac{(n-1)^{1/4}e^{2\sqrt{n-1}}}{n^{-1/4}e^{2\sqrt{n}}}(1+o(1)) = \sqrt{n}(1+o(1)),$$

that is, the expected number $\mathbb{E}(X_n)$ of sequences in a random partial injection of size n is asymptotically equal to $\sqrt{n}$.

Similarly, from Equation (5) we get

$$\mathbb{E}(X_n^2) = \frac{(n-2)^{3/4}e^{2\sqrt{n-2}}}{n^{-1/4}e^{2\sqrt{n}}}(1+o(1)) + \sqrt{n}(1+o(1)) = n(1+o(1)),$$

and from Equation (6) $\sigma^2(X_n) = o(n)$. Thus the standard deviation $\sigma(X_n)$ of the number of sequences in a random partial injection of size n is $o(\sqrt{n})$. □

2.3.4 Proof of Theorem 2.6

Before we prove Theorem 2.6, let us recall the statement of Chebyshev's inequality.

Proposition 2.12 (Chebyshev's inequality) *If X is a random variable of expectation $\mathbb{E}(X)$ with a finite variance $\sigma^2(X)$ then for any positive real α*

$$\mathbb{P}\{|X - \mathbb{E}(X)| \geq \alpha\} < \frac{\sigma^2(X)}{\alpha^2}. \tag{8}$$

As we already noted, given a pair of partial injections, a leaf in the resulting A-graph is an endpoint of a sequence in one of the partial injections, and a singleton in the other. In view of Lemma 2.11, Chebyshev's inequality (Equation (8)) applied to X_n and $\alpha = \sqrt{n}$, shows that

$$\mathbb{P}\{|X_n - \sqrt{n}| \geq \sqrt{n}\} = o(1). \tag{9}$$

In other words, the probability that a partial injection of size n contains more than $2\sqrt{n}$ sequences tends towards 0 when n tends towards ∞. Let us call such a partial injection *sequence-rich*. Then n-vertex labeled A-graphs defined by a pair of partial injections, one of which at least is sequence-rich, occur with probability $o(1)$.

Let us now focus on the labeled A-graphs defined by partial injections, each of which contains less than $2\sqrt{n}$ sequences. Again, a vertex is a leaf if it is an endpoint of sequence for one of the injections and a singleton in the other one. As the two injections play symmetric rôles, we estimate the probability for a vertex to be an endpoint in the first injection and a singleton for the second one: the total estimate will be bounded above by twice that probability.

Since the first injection has at most $2\sqrt{n}$ sequences, it has at most $4\sqrt{n}$ endpoints. To estimate the number of second partial injections in which at least one of these vertices is a singleton, we count the number of partial injections that do not involve such a vertex. For any given vertex, there are I_{n-1} such

partial injections, and as we have to consider up to $4\sqrt{n}$ potential endpoints, the number of partial injections in which at least one of these vertices is a singleton is bounded above by $4\sqrt{n}I_{n-1}$.

Therefore there are at most $8\sqrt{n}I_{n-1}I_n$ pairs of partial injections that are not sequence-rich and that exhibit at least a leaf. As $I_{n-1} \leq \frac{I_n}{n}$, this number is less than or equal to $\frac{8}{\sqrt{n}}I_n^2$, and the associated probability is less than or equal to $\frac{8}{\sqrt{n}}$.

Consequently the probability for an n-vertex labeled A-graph to have at least a leaf is less than or equal to

$$o(1) + \frac{8}{\sqrt{n}} = o(1),$$

which concludes the proof.

2.4 The number of size n subgroups

Let $S_{n,r}$ be the number of size n subgroups of $F = F(A)$, where $\mathsf{rank}(F) = r$. By Corollary 2.7, the number of admissible labeled A-graphs of size n is $I_n^r(1+o(1))$. By Lemma 1.5, each size n subgroup is represented by $(n-1)!$ distinct admissible labeled A-graphs, so

$$S_{n,r} \sim \frac{I_n^r}{(n-1)!}.$$

Proposition 2.10 gives us an equivalent of $I_n/n!$, and it follows that

$$S_{n,r} \sim n\ n!^{r-1}\frac{I_n^r}{n!^r} \sim n\ n!^{r-1}\frac{e^{-r/2}}{2^r\pi^{r/2}}\,n^{-r/4}e^{2r\sqrt{n}}.$$

By Stirling's formula, $n!$ is equivalent to $\sqrt{2\pi}e^{-n}n^{n+\frac{1}{2}}$ and it follows that

$$S_{n,r} \sim \frac{(2e)^{-r/2}}{\sqrt{2\pi}}e^{-(r-1)n+2r\sqrt{n}}n^{(r-1)n+\frac{r+2}{4}}.$$

3 Random generation algorithm

As discussed in Section 1.2, and in particular in Section 1.2.3 (see also Section 3.2 below), the core of our random generation algorithm for admissible A-graphs, is a procedure to randomly generate size n partial injections.

The recursive decomposition of partial injections investigated so far allows us to use the *recursive method* introduced by Flajolet, Zimmermann and van Cutsem [11] (following work by Nijenhuis and Wilf [22]) to *efficiently* and *randomly* generate partial injections of size n.

Recall that a partial injection is a set of disjoint *components*, that are either cycles or non-empty sequences. The recursive method consists, in our case, in:

- choosing the size k of a component ($k \in \{1, \ldots, n\}$) according to the distribution of the sizes of components in a random size n partial injection;

- choosing whether that size k component is a cycle or a sequence – according to the distribution of these two types among size k components;
- and choosing a size $n-k$ partial injection.

We give more details below, on how these steps are performed. The result of the procedure is a sequence of symbols of the form $\xi_1(k_1)\dots\xi_r(k_r)$, where $k_1+\cdots+k_r=n$, the ξ_i are in $\{\sigma,\kappa\}$, $\sigma(k)$ stands for *sequence of size* k, and $\kappa(k)$ stands for *cycle of size* k. Such a sequence represents, in a natural way, an unlabeled size n partial injection and the last step of the algorithm consists in randomly labeling that partial injection.

Let us now be more precise.

3.1 Partial injections

The tool to grasp the distribution of the sizes of components in partial injections is the pointing operator Θ: pointing a labeled object consists in marking one of its atoms, or equivalently one of its labels from $\{1,\cdots,n\}$. Naturally, there are n ways of pointing an object of size n. So if the EGS of a labeled combinatorial class $\mathcal{C}$ is $C(z)=\sum c_n z^n/n!$, then the pointed class of $\mathcal{C}$, denoted by $\Theta\mathcal{C}$, has EGS

$$\Theta C(z)=\sum_{n\geq 0}\frac{nc_n}{n!}z^n=z\frac{d}{dz}C(z).$$

If $\mathcal{C}$ is, as in our situation, defined as a set of components of a class $\mathcal{D}$ with EGS $D(z)$, then $C(z)=\exp(D(z))$ and

$$\begin{aligned}\Theta C(z)=z\frac{d}{dz}C(z) &= z\frac{d}{dz}\left(\exp(D(z))\right)\\ &= z\left(\frac{d}{dz}D(z)\right)\exp(D(z))=\Theta D(z)\ C(z).\end{aligned}$$

The combinatorial interpretation of this equality is the following: marking an atom of an element of $\mathcal{C}$ amounts to marking an atom of one of its components (of size, say, k), and the remaining part of the element of $\Theta\mathcal{C}$ is a non-pointed element of $\mathcal{C}$ of size $n-k$.

For partial injections, we have $I(z)=\exp(D(z))$, with $D(z)=\frac{z}{1-z}+\log(\frac{1}{1-z})$. Therefore

$$\Theta I(z)=\Theta D(z)\times I(z)=\left(\frac{z}{(1-z)^2}+\frac{z}{1-z}\right)I(z), \tag{10}$$

where $\frac{z}{(1-z)^2}$ is the EGS of pointed labeled sequences and $\frac{z}{1-z}$ is the EGS of pointed labeled cycles. Now

$$[z^k]\frac{z}{(1-z)^2}=\begin{cases}k & \text{if } k\geq 1,\\ 0 & \text{if } k=0,\end{cases}\qquad [z^k]\frac{z}{1-z}=\begin{cases}1 & \text{if } k\geq 1,\\ 0 & \text{if } k=0,\end{cases}$$

so we have

$$n\frac{I_n}{n!} = \sum_{k=1}^{n}(k+1)\frac{I_{n-k}}{(n-k)!}\ . \tag{11}$$

Therefore the probability $p_k = \mathbb{P}(\text{size } k)$ for the pointed component to be of size k is

$$p_k \ = \ \frac{(k+1)\frac{I_{n-k}}{(n-k)!}}{\frac{I_n}{(n-1)!}} \ = \ \frac{1}{I_n}\left((k+1)\frac{(n-1)!}{(n-k)!}I_{n-k}\right)$$

and the probability for a size k component to be a sequence (resp. a cycle) is

$$\mathbb{P}(\text{sequence}) = \frac{k}{k+1} \qquad\qquad \mathbb{P}(\text{cycle}) = \frac{1}{k+1}.$$

We are now ready to describe the random generation algorithm for partial injections of size n. The discussion of its complexity is postponed to Section 3.3.

Let Uniform($[0,1[$) be the function that returns a real number chosen uniformly at random in the interval $[0,1[$. Recall that if X is a random variable with values in $[\![1,n]\!]$ with probability $\mathbb{P}(X=i)=p_i$ then the value of X can be generated randomly with respect to this probability distribution as follows (see [5] for example).

```
RandomX
  dice = Uniform([0,1[)
  k = 1, S = p_1
  while dice ≥ S
      k = k + 1
      S = S + p_k
  return k
```

Our algorithm to randomly generate a partial injection of size n uses directly this idea. Because the probabilities discussed above (for a component of a random partial injection to have size k, for a size k component to be a cycle) are rational numbers, we choose to express the algorithm entirely in integers, in order to facilitate exact computation, and thereby to guarantee the absence of bias in the distribution of the partial injections. Concretely, we multiply dice and the p_k by I_n.

The algorithm requires a preliminary phase, during which a table containing the values of I_k $(0 \le k \le n)$ is computed using the recurrence relation in Equation (1).

We denote by Uniform(N) a function that returns an integer chosen uniformly at random in the interval $[0,N[$.

```
RandomPartialInjection(n)
  Result = []
  while n > 0
      // Compute the size k of a component
      dice = Uniform(I_n)
      k = 1, T = 1, S = 2I_{n-1}    // That is, S = I_n p_1
```

```
    while dice ≥ S
        k = k + 1
        T = T ∗ (n − k + 1)
        S = S + (k + 1)TI_{n−k}    // That is, S = S + I_n p_k
    // Decide whether the component is a sequence or a cycle
    dice′ = UNIFORM(k + 1)
    if dice′ < k
        then Append σ(k) to Result
        else Append κ(k) to Result
    n = n − k
// Randomly label the final result
Label Result with RANDOMPERMUTATION(n)
return Result
```

The outer **while** loop of the algorithm produces a sequence of symbols of the form $(\xi_1(k_1), \ldots, \xi_r(k_r))$, with each $\xi_i \in \{\sigma, \kappa\}$ and such that $\sum_{i=1}^{r} k_i = n$, that describes the size and the nature of the components of the size n partial injection.

The last step of the algorithm, which randomly generates a permutation of the n elements on which the partial injection is defined, can be performed in linear time and space using the algorithm given in Section 5.

3.2 Admissible A-graphs

Recall (see Section 1.2.3) that our algorithm to generate admissible A-graphs consists in randomly generating, for each letter of the alphabet A, a partial injection of size n, and then using a rejection algorithm to keep only admissible A-graphs.

Assuming that the table for the values of I_n is already computed, the algorithm reads as follows.

```
RANDOMADMISSIBLEAGRAPH(n)
  repeat
      for each a ∈ A
          compute the partial injection I_a for a using RANDOMPARTIALINJECTION(n)
  until the resulting A-graph is admissible.
```

3.3 Complexity

The algorithm requires manipulating large integers: I_n is of the order of $\mathcal{O}(n^n)$. Computing in multiprecision, that is, without any approximation, guarantees the absence of bias in the distribution of the generated objects. However, we also briefly discuss floating point implementations at the end of this section.

We first evaluate the complexity of the algorithm in the RAM model, *i.e.*, under the unit cost assumption (or uniform cost convention) according to which each data element (here: each integer) is stored in one unit of space, and the elementary operations (reading, writing, comparing, performing arithmetic operations, etc) require one unit of time, – even for large numbers.

The pre-computation phase that stores the values of I_n uses the recurrence relation on the I_n given in Equation (1), and it requires $\mathcal{O}(n)$ operations.

The (worst case) time complexity $T(n)$ of RandomPartialInjection(n) (assuming that the values of the I_n are precomputed) satisfies the following inequality

$$T(n) \leq \max_{1 \leq k \leq n} (ck + T(n-k))$$

(where c is a constant), so that $T(n) \leq cn$, that is, $T(n) = \mathcal{O}(n)$.

Next, checking whether the A-graph generated is connected can be done using common algorithms on graphs (depth-first search) in time $\mathcal{O}(n)$. Checking 1-trimness is also done in $\mathcal{O}(n)$, by scanning the list of edges which has at most $|A|\, n$ elements. This part of the algorithm does not require manipulating large numbers.

Corollary 2.7 shows that size n A-graphs are admissible with probability $1 + o(1)$. Therefore the number of rejects (see Section 1.2.3) for lack of admissibility is equal in average to $\frac{1}{1+o(1)} = 1 + o(1)$. Thus the average number of rejects tends to 0 when n tends to infinity.

In conclusion, for the RAM model, the random generation algorithm requires a precomputation that can be done in linear time, and it uses, in average, $\mathcal{O}(n)$ operations to generate each admissible A-graph.

In the case of the bit complexity (or logarithmic cost convention), an integer N is handled via its binary representation, of length $\mathcal{O}(\log N)$. In particular, the representation of I_n has length $\mathcal{O}(n \log n)$. The basic operations on numbers of that size (reading, writing, comparison, addition, multiplication by a number whose binary representation is of length $\mathcal{O}(n)$) are performed in time $\mathcal{O}(n \log n)$, whereas the multiplication of two such numbers takes time $\mathcal{O}(n \log^2 n)$. Under this bit-cost assumption, the time and space required for the pre-computation are $\mathcal{O}(n^2 \log n)$ (instead of $\mathcal{O}(n)$ in the RAM model). And each random draw takes time $\mathcal{O}(n^2 \log^2 n)$.

Remark 3.1 Under the bit-cost assumption, we should also take into consideration the complexity of the function Uniform(N). Since this function returns an integer, it can be performed by a rejection algorithm, randomly choosing each bit of (the binary expansion of) N, that is, in $1 + \lfloor \log n \rfloor$ unit cost operations. In such a process, the probability that the integer generated is greater than N is at most $1/2$ ($\frac{b-1}{b}$ in base b), so the average number of reject is at most 2, and the complexity of Uniform(N) is $\mathcal{O}(\log N)$. □

In practice, it is often convenient to use floating point arithmetic instead of multiprecision arithmetic. In theory, the approximations made in floating point arithmetic induce a loss of precision, and may therefore introduce a bias in the probability distribution of the generated objects.

Denise and Zimmermann [4] showed that the complexity of the floating point implementation is the same as for the RAM model. They also show that, if certain precautions are taken (essentially in the choice of the rounding operator for each operation), a floating point implementation introduces only a negligible

bias in the probability distribution of the generated objects. In the case of partial injections, using the standard rounding operator does not seem experimentally to produce a significant bias, but it is not theoretically proved.

4 On the rank of a size n subgroup

We conclude this paper with a few applications of the above results to the study of the rank distribution of size n subgroups. The first one concerns the expected value of this rank, and the others establish the intuitive results that finite index (resp. fixed rank k) subgroups are asymptotically negligible.

Recall that the rank of a subgroup H with a size n graphical representation Γ is equal to $|E(\Gamma)| - n + 1$, where $|E(\Gamma)|$ is the number of edges of the graph Γ (see Section 1.1).

Corollary 4.1 *The average rank of a size n subgroup of $F(A)$ is $(|A| - 1)n - |A|\sqrt{n} + 1$, with standard deviation $o(\sqrt{n})$.*

Proof. Let Γ an A-graph. For each letter a of the alphabet A, the number of a-labeled edges is the difference between n and the number of sequences in the functional graph of the partial injection determined by the a-labeled edges. In view of Lemma 2.11, the number of a-labeled edges is therefore asymptotically equal to $n - \sqrt{n}$, with standard deviation $o(\sqrt{n})$, and the announced result follows. □

Corollary 4.2 *Let H be a size n subgroup of $F(A)$ and let $k \geq 1$ be an integer. Then the probability that $\mathsf{rank}(H) \leq k$ is asymptotically $o\left(\frac{1}{n}\right)$.*

Proof. Corollary 4.1 shows that the mean value of the rank of H is $\mathbb{E}(\mathsf{rank}) = (|A| - 1)n - |A|\sqrt{n} + 1$, with variance $\sigma^2(\mathsf{rank}) = o(n)$.

If $\mathsf{rank}(H) \leq k$, then in particular $|\mathsf{rank}(H) - \mathbb{E}(\mathsf{rank})| \geq \mathbb{E}(\mathsf{rank}) - k$. It follows, by Chebyshev's inequality (see Equation (8) above), that

$$\begin{aligned} \mathbb{P}\{\mathsf{rank} \leq k\} &\leq \mathbb{P}\{|\mathsf{rank} - \mathbb{E}(\mathsf{rank})| \geq \mathbb{E}(\mathsf{rank}) - k\} \\ &\leq \frac{o(n)}{\mathcal{O}(n^2)} = o(\frac{1}{n}). \end{aligned}$$

□

5 Finite index subgroups

We saw in Section 1.1 that a finitely generated subgroup has finite index if and only if, in its graphical representation, every letter labels a permutation, that is, a partial injection whose domain is the full set of vertices. Let us say, in that case, that the corresponding A-graph is a *permutation A-graph*. Based on

this observation, we can adapt our approach to get a linear time (in average) random generation algorithm.

As in the general case (see Section 3.2), we use a rejection algorithm: we repeatedly randomly generate a permutation of size n for each letter $a \in A$, until the resulting graph is connected and 1-trim.

Random generation of permutations is a classical object of study, and it can be performed in time $\mathcal{O}(n)$ (in the RAM model, $\mathcal{O}(n \log n)$ in the bit-cost model) using the following algorithm (see [5] for example):

```
RandomPermutation(n)
  for i ∈ {1, ···, n}
    P[i] = i
  for i from 2 to n
    j = 1 + Uniform(i)        // j is a random integer in ⟦1, i⟧
    Swap P[i] and P[j]
  return P
```

Note that this algorithm does not require manipulating large integers.

The efficiency of the rejection algorithm depends on the average number of rejects, and hence on the probability, for a permutation A-graph to be connected and 1-trim. Trimness is a moot point since a permutation A-graph never has any leaf.

Connectedness is not guaranteed, but we note that Dixon [6] uses Bender's theorem (Theorem 2.5 above) to compute the asymptotic expansion of the probability for a pair (or a r-tuple) of size n permutations to generate a transitive subgroup of $\mathbb{S}_n$, that is, to define a connected permutation A-graph. He shows in particular that this probability is of the form $1 - 1/n^{r-1} + \mathcal{O}(1/n^{2(r-1)})$.

Thus the average number of rejects tends to 0 when n tends to infinity, and the average case complexity of the random generation of an admissible permutation A-graph is $\mathcal{O}(n)$ (in the RAM model, $\mathcal{O}(n \log n)$ in the bitcost model).

We can also show that finite index subgroups are asymptotically negligible among subgroups of a given size.

Proposition 5.1 *The probability for a randomly chosen size n subgroup of $F(A)$ to have finite index is $\mathcal{O}(n^{r/4}e^{-2r\sqrt{n}})$. In particular, it is $o\left(n^{-k}\right)$ for any $k \geq 1$.*

Proof. Let $r = \mathsf{rank}(F) = |A|$. The number of size n finite index subgroups is at most the number of r-tuples of permutations, namely $n!^r$. The number of size n subgroups is, according to the discussion in this paper, equivalent to the number of r-tuples of partial injections, that is, it is equal to $I_n^r(1 + o(1))$.

Thus the probability that a size n subgroup has finite index is at most equal to $\left(\frac{n!}{I_n}\right)^r (1 + o(1))$. By Proposition 2.10 (applied with $p = 0$), we know that $I_n/n! = \mathcal{O}(n^{-1/4}e^{2\sqrt{n}})$. Therefore

$$\left(\frac{n!}{I_n}\right)^r = \mathcal{O}\left(n^{r/4}e^{-2r\sqrt{n}}\right),$$

which converges to 0 faster than the inverse of any polynomial. □

6 A few questions

A first question, prompted by Proposition 5.1, is the following. Even though finite index subgroups are negligible among finitely generated subgroups, we saw in Section 5 how to randomly generate them. When k is fixed, rank k subgroups are also asymptotically negligible among finitely generated subgroups (Corollary 4.2). Can we find an efficient random generation algorithm for these subgroups?

Our second question is related with another method used in the literature to generate subgroups (not only for free groups). This method is based on the idea of randomly generating a k-tuple of elements that generate the subgroup — with k fixed and, say, the maximal length of the generators being allowed to tend to infinity. It is used for instance by Jutsikawa [14] to study the distribution of malnormal subgroups in free groups. We refer also to Martino, Turner and Ventura [19] on the distribution of monomorphisms between free groups, and to Miasnikov and Ushakov [20] for a survey of this technique in relation with group-based cryptography.

The question that arises in this context is to compare the distribution of subgroups that occurs with this generation scheme and the distribution we considered in this paper. They must be different since we fix the size of the subgroups generated, whereas they fix the number and the maximal length of a set of generators, which may lead to graphical representations of varying size. One must also take into consideration the fact that each subgroup is generated by a potentially large number of k-tuples of generators. However, it remains possible that generic properties (those that have asymptotically probability 1) coincide for both distributions.

References

[1] E.A. Bender. Asymptotic methods in enumeration, *SIAM Review* **16** (1974) 485-515.

[2] E.A. Bender. An asymptotic expansion for the coefficients of some formal power series, *J. London Math. Soc.* **9** (1974/75) 451-458.

[3] M.R. Bridson, D.T. Wise. Malnormality is undecidable in hyperbolic groups, *Israel J. Mathematics* **124** (2001) 313-316.

[4] A. Denise, P. Zimmermann. Uniform random generation of decomposable structures using floating-point arithmetics, *Theoret. Comput. Sci.* **218** (1999) 233-248.

[5] L. Devroye. *Non-uniform random variate generation*, Springer-Verlag, 1986.

[6] J.D. Dixon. Asymptotics of Generating the Symmetric and Alternating Groups, *The Electronic Journal of Combinatorics 12*(1) (2005) # R56.

[7] P. Duchon, P. Flajolet, G. Louchard, G. Schaeffer. Boltzmann Samplers for the Random Generation of Combinatorial Structures, *Combinatorics, Probability, and Computing* **13** (2004) 577-625.

[8] P. Flajolet, E. Fusy, C. Pivoteau. Boltzmann Sampling of Unlabelled Structures, in (D. Appelgate et al., eds) *Proceedings of the Ninth Workshop on Algorithm Engineering and Experiments and the Fourth Workshop on Analytic Algorithmics and Combinatorics*, pp. 201-211, SIAM Press, 2007.

[9] P. Flajolet, R. Sedgewick. *An introduction to the analysis of algorithms*, Addison-Wesley, 1996.

[10] P. Flajolet, R. Sedgewick. *Analytic combinatorics*, Book in preparation, (Version of October 23, 2006 available at `http://www.algo.inria.fr/flajolet/publist.html`).

[11] P. Flajolet, P. Zimmermann, B. Van Cutsem. A calculus of random generation of labelled combinatorial structures, *Theoret. Comput. Sci.* **132** (1994) 1-35.

[12] D.H. Greene, D.E. Knuth. *Mathematics for the analysis of algorithms*, Birkhaüser, 1981.

[13] W. K. Hayman. A generalization of Stirling's formula, *Journal für die reine und angewandte Mathematik* **196** (1956) 67-95.

[14] T. Jitsukawa. Malnormal subgroups of free groups, in *Computational and statistical group theory* (Las Vegas, NV/Hoboken, NJ, 2001), 83-95, *Contemp. Math.* **298**, Amer. Math. Soc., Providence, RI, 2002.

[15] I. Kapovich, A.G. Miasnikov. Stallings foldings and subgroups of free groups, *J. Algebra* **248** (2002) 608-668.

[16] I. Kapovich, A.G. Myasnikov, P. Schupp, V. Shpilrain. Generic-case complexity, decision problems in group theory and random walks, *J. Algebra* **264** (2003) 665-694.

[17] I. Kapovich, A.G. Myasnikov, P. Schupp, V. Shpilrain. Average-case complexity and decision problems in group theory, *Advances in Math.* **190** (2005) 343-359.

[18] S. Margolis, M. Sapir, P. Weil. Closed subgroups in the pro-V topologies, and the extension problem for inverse automata, *Intern. J. Algebra and Computation* **11** (2001) 405-445.

[19] A. Martino, T. Turner, E. Ventura. Counting monomorphisms of a free group, to appear.

[20] A. Miasnikov, A. Ushakov. Random subgroups and cryptanalysis of the length-based attacks, to appear.

[21] A. Miasnikov, E. Ventura, P. Weil. Algebraic extensions in free groups, in *Algebra and Geometry in Geneva and Barcelona* (G.N. Arzhantseva, L. Bartholdi, J. Burillo, E. Ventura eds.), Trends in Mathematics, Birkhaser (2007), pp. 225-253.

[22] A. Nijenhuis, H. S. Wilf. *Combinatorial Algorithms*, 2nd ed., Academic Press, 1978.

[23] A. M. Odlyzko. Asymptotic enumeration methods, in *Handbook of Combinatorics* (R. Grahams, M. Grötschel, A. Lovász, eds), vol. II, Elsevier, 1995, p. 1063-1229.

[24] A. Roig, E. Ventura, P. Weil. On the complexity of the Whitehead minimization problem, *Intern. J. Algebra and Computation*, to appear. Preprint **721**, Centre de Recerca Matemtica, 2006.

[25] N. J. A. Sloane. *The On-Line Encyclopedia of integer sequences*, 2000, published electronically at `http://www.research.att.com/~njas/sequences`.

[26] J. R. Stallings. Topology of finite graphs, *Inventiones Math.* **71** (1983) 551-565.

[27] N. Touikan. A fast algorithm for Stalling's folding process, *Internat. J. Algebra and Computation* **16**(6) (2006) 1031-1046.

[28] P. Weil. Computing closures of finitely generated subgroups of the free group, in *Algorithmic Problems in Groups and Semigroups* (J.-C. Birget, S. Margolis, J. Meakin, M. Sapir eds.), Birkhaüser, 2000, pp. 289 - 307.