Random generation of possibly incomplete deterministic automata

Frédérique Bassino, Julien David, Cyril Nicaud
Institut Gaspard-Monge, Université Paris Est,
77454 Marne-la-Vallée Cedex 2 – France
Email adresses: {bassino,Julien.David,nicaud}@univ-mlv.fr

Abstract

This paper presents an efficient random generator, based on a Boltzmann sampler, for accessible, deterministic and possibly not complete automata. An intermediate result is that $\Theta(1)$ of accessible and deterministic automata are complete.

1 Introduction

The enumeration of finite automata according to various criteria (non-isomorphic [11], up to permutation of the labels of the edges [11], with a strongly connected underlying graph [15, 13, 19, 14], accessible [15, 13, 19], acyclic [16],...) is a problem that was studied since 1959 [21].

In [1] the first and third authors exhibit a bijection between the set $\mathcal{A}_n$ of deterministic, complete and accessible automata with n states on a k-letters alphabet and some diagrams, which can themselves be represented as partitions of the set $\{1, \cdots, kn\}$ into n non-empty subsets. These combinatorial transformations show that the order of magnitude of the cardinality $|\mathcal{A}_n|$ of the set $\mathcal{A}_n$ is related to the Stirling numbers of the second kind that can be be used to reformulate an asymptotic estimate of $|\mathcal{A}_n|$ due to Korshunov [13]. They also provide a uniform random generator for the automata of $\mathcal{A}_n$, based on Boltzmann samplers [6, 7], that is more efficient than former ones [17, 4] using a recursive algorithm [18, 9].

This paper is an extension of the study [1] of deterministic, complete and accessible automata to possibly incomplete automata. The combinatorial transformations are slighty changed and a careful analysis of the complexity is done to ensure that the generator obtained is still efficient; as in the case of complete automata, its average complexity is $\mathcal{O}(n^{3/2})$, where n is the number of states of automata. An interesting intermediate result is that for any finite alphabet, the proportion of complete automata with n states amongst deterministic and accessible ones is greater than a positive constant.

The paper is organized as follows. Bijections used to transform automata into set partitions are presented in Section 3. Section 4 is devoted to enumeration results used in the analysis of the complexity of the generator. The random generator, together with the analysis of its efficiency, is given in Section 5.

2 Automata

Our goal is to study from a combinatorial point of view the set of accessible and deterministic automata with n states. Therefore we first recall some definitions about finite automata, referring the readers to [12, 20] for basic elements of this theory.

2.1 Deterministic and accessible automata

A *deterministic finite automaton* $\mathcal{A}$ over a finite alphabet A is a quintuple $\mathcal{A} = (A, Q, \cdot, q_0, F)$ where Q is a finite set of *states*, $q_0 \in Q$ is the initial state, $F \subset Q$ is the set of final states and the *transition function* $\cdot$ is an element of $Q \times A \mapsto Q \cup \emptyset$. If $p \cdot a = \emptyset$ for a given state $p \in Q$ and letter $a \in A$, then $p \cdot a$ is an *undefined transition*. A deterministic finite automaton without undefined transition is *complete*. If $\mathcal{A} = (A, Q, \cdot, q_0, F)$ is a deterministic finite automaton, its transition function is extended by morphism to $Q \times A^*$ making use of the convention $\emptyset \cdot a = \emptyset$ for every $a \in A$.

A deterministic finite automaton $\mathcal{A}$ is *accessible* when for each state q of $\mathcal{A}$, there exists a word $u \in A^*$ such that $q_0 \cdot u = q$.

Two deterministic finite automata $\mathcal{A} = (A, Q, \cdot, q_0, F)$ and $\mathcal{A}' = (A, Q', \cdot, q_0', F')$ over the same alphabet are *isomorphic* when there exists a bijection τ from $Q \cup \emptyset$ to $Q' \cup \emptyset$ such that, $\tau(q_0) = q_0'$, $\tau(\emptyset) = \emptyset$, $\tau(F) = F'$ and for each $(q, \alpha) \in Q \times A$, $\tau(q \cdot \alpha) = \tau(q) \cdot \alpha$. Two isomorphic automata only differ by the labels of their states.

2.2 Transition structures

Now we introduce a representation of accessible and deterministic automata that uses the minimal labels of simple paths and allows us to enumerate and generate them easily. More precisely a *simple path* in a deterministic automaton $\mathcal{A}$ is a path labelled by a word u whose all prefixes v and v' of u such that $v \neq v'$ satisfy $q_0 \cdot v \neq q_0 \cdot v'$. In other words, in the graphical representation of $\mathcal{A}$ the path labelled by u does not go twice through the same state. Let $\mathcal{A}$ be an accessible and deterministic automaton on the alphabet A and let w be the map from Q to A^* defined for every state $q \in Q$ by

$$w(q) = \min_{lex}\{u \in A^* \mid q_0 \cdot u = q \text{ and } u \text{ is a simple path in } \mathcal{A}\},$$

where the mininum is taken according to the lexicographic order. Note that $w(q)$ always exists since $\mathcal{A}$ is accessible. An automaton $\mathcal{A} = (A, Q, \cdot, q_0, F)$ is called a *base automaton* when $Q \subset A^*$ (the states are labelled by words) and for all $u \in Q$, $w(u) = u$. Note that by construction, if $u \in Q$ and v is a prefix of u, then $v \in Q$. As two distinct base automata cannot be isomorphic, we can directly work on isomorphism classes using base automata.

The *transition structure* of an automaton $\mathcal{A} = (A, Q, \cdot, q_0, F)$ is $\mathcal{D} = (A, Q, \cdot, q_0)$: in $\mathcal{D}$ there is no more distinguished final states. We can define similarly accessible and deterministic transition structures.

Denote by $\mathcal{D}_n$ the set of accessible and deterministic transition structures of base automata with n states, and by $\mathcal{C}_n$ the set of complete transition structures belonging to $\mathcal{D}_n$.

Given an element $\mathcal{D}$ of $\mathcal{D}_n$, there are exactly 2^n automata whose transition structure is $\mathcal{D}$, since the accessibility prevents distinct choices of final sets to form the same automaton. Therefore the number of deterministic and accessible automata, up to isomorphism, is $2^n|\mathcal{D}_n|$.

Note that forbidding or not the set of final states to be empty does not basically change the results, since the probability of this event is $1/2^n$.

Our purpose is to enumerate the elements in $\mathcal{D}_n$ and to generate them randomly for the uniform distribution on $\mathcal{D}_n$.

3 Bijections

In this section we show that accessible and deterministic transition structures are in bijection with pairs of integer sequences represented by boxed diagrams.

3.1 Boxed diagrams and k-Dyck boxed diagrams

A *diagram* of width m and height n is a sequence $(x_1, \ldots, x_m)$ of weakly increasing nonnegative integers such that $x_m = n$, classically represented as a diagram of boxes, see Figure 1. A *k-Dyck diagram* of size n is a diagram of width $(k-1)n+1$ and height n such that $x_i \geq \lceil i/(k-1) \rceil$ for each $i \leq (k-1)n$. A *boxed diagram* is a pair of sequences $((x_1, \ldots, x_m), (y_1, \ldots, y_m))$ where $(x_1, \ldots, x_m)$ is a diagram and for each $i \in [\![1..m]\!]$, the y_ith box of the column i of the diagram is marked, see Figure 1. As a consequence, a diagram gives rise to $\prod_{i=1}^{m} x_i$ boxed diagrams. A *k-Dyck boxed diagram* of size n is a boxed diagram such that its first coordinate $(x_1, \ldots, x_{(k-1)n+1})$ is a k-Dyck diagram of size n.

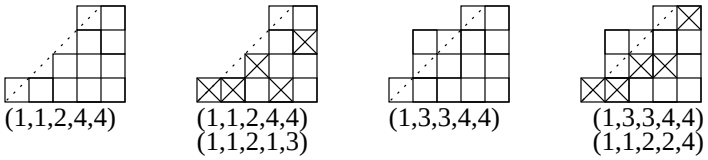

Figure 1: A diagram of width 5 and height 4, a boxed diagram, a 2-Dyck diagram and a 2-Dyck boxed diagram

3.2 From transition structures to k-Dyck boxed diagrams

First recall the bijection established in [17] for a two-letters alphabet and generalized to any finite alphabet in [4]:

Theorem 1 *There exists a bijection between the set $\mathcal{C}_n$ of accessible, complete and deterministic transition structures with n states on a k-letters alphabet A and the set of k-Dyck boxed diagrams of size n. This transformation and its inverse can be computed in linear time.*

For $n \geq 1$, let $\mathcal{D} = (A, Q, \cdot, \varepsilon) \in \mathcal{C}_n$ be a deterministic, accessible and complete transition structure of a base automaton. Since $\mathcal{D}$ is complete, it contains kn transitions of the form (u, α), with $u \in Q$ and $\alpha \in A$. We partition these transitions depending on whether they belong to the spanning tree induced by the depth-first traversal according to the lexicographical order of the structure or not. Using the properties of the labelling of the states of $\mathcal{D}$, the partition can be described as follows, for any $u \in Q$:

- If $u\alpha \in Q$ then $u \cdot \alpha = u\alpha$ and (u, α) is a *tree transition*. It belongs to the spanning tree.

- If $u\alpha \notin Q$ then $u \cdot \alpha <_{lex} u\alpha$, and (u, α) is called a *missing transition.* It does not belong to the spanning tree.

There are $n-1$ tree transitions and $(k-1)n+1$ missing transitions.

Let ν be the unique increasing bijection from the set Q (lexicographically ordered) to $\{1, \cdots, n\}$, that is, $\nu(q)$ is the number of elements of Q smaller or equal to q for the lexicographical order. To any missing transition $t = (q, \alpha)$ we associate the pair of integers (x_t, y_t) defined by

$$\begin{cases} x_t &= |\{u \in Q \mid u <_{lex} q\alpha\}| \\ y_t &= \nu(q \cdot \alpha) \end{cases}$$

We then order the transitions of $\mathcal{D}$ accordin to the relation $(u, \alpha) < (v, \beta)$ if and only if $u\alpha <_{lex} v\beta$. The bijection Ψ between $\mathcal{C}_n$ and the set of k-Dyck boxed diagrams of size n can be defined as follows: let $(t_1, \cdots, t_{(k-1)n+1})$ be the ordered sequence of missing transitions of $\mathcal{D}$,

$$\Psi(\mathcal{D}) = ((x_{t_1}, \cdots, x_{t_{(k-1)n+1}}), (y_{t_1}, \cdots, y_{t_{(k-1)n+1}})).$$

The map Ψ is a bijection (see [17, 4] for details). The sequence $(x_{t_1}, \cdots, x_{t_{(k-1)n+1}})$ represents the depth-first spanning tree of $\mathcal{D}$ and defines the labelling of the states of $\mathcal{D}$; the sequence $(y_{t_1}, \cdots, y_{t_{(k-1)n+1}})$ carries all the informations about missing transitions:

$$u \cdot \alpha = \nu^{-1}(y_{(u,\alpha)}).$$

3.3 From transition structures to complete transitions structures

In theory of automata an incomplete automaton is classically changed into a complete one recognizing the same language by the addition of a sink state. This transformation is not suitable for our combinatorial construction. Indeed if two incomplete automata have the same depth-first spanning tree, they may not have the same one after the addition of a sink state.

Therefore we introduce another transformation denoted by ϕ and defined as follows: to any $\mathcal{D} \in \mathcal{D}_n$, with $\mathcal{D} = (A, Q, \cdot, \varepsilon)$, we associate the complete transition structure $\phi(\mathcal{D}) = (A, Q', *, \varepsilon)$ in $\mathcal{C}_{n+1}$ with $Q' = \{\varepsilon\} \cup a_k Q$ where $a_k = \max_{lex}\{\alpha \in A\}$ and whose transitions are defined by:

$$\begin{cases} \varepsilon * \alpha = \varepsilon & \text{if } \alpha \neq a_k \\ \varepsilon * a_k = a_k & \\ q' * \alpha = a_k(q \cdot \alpha) & \text{if } \exists q \in A^*,\ q' = a_k q \text{ and } q \cdot \alpha \neq \emptyset \\ q' * \alpha = \varepsilon & \text{if } \exists q \in A^*,\ q' = a_k q \text{ and } q \cdot \alpha = \emptyset \end{cases}$$

This construction consists of

- adding a new state, that becomes the initial state ε of $\phi(\mathcal{D})$ and a transition $\varepsilon * a_k = q_0$, labelled by the greatest letter and where q_0 is the initial state $\mathcal{D}$,
- relabelling the transition structure to obtain the transition structure of a base automaton,
- changing any undefined transition $q \cdot \alpha = \emptyset$ into $q * \alpha = \varepsilon$.

Note that ϕ does not preserve the language recognized.

Lemma 1 *Denote by $\mathcal{E}_n$ the subset of transition structures of $\mathcal{C}_n$ such that $\varepsilon \cdot \alpha = \varepsilon$ for $\alpha \in A \setminus \{\max_{lex}\{\alpha \in A\}\}$. The function ϕ is a bijection from $\mathcal{D}_n$ to $\mathcal{E}_{n+1}$.*

By definition of ϕ, $\mathcal{E}_{n+1} = \phi(\mathcal{D}_n)$. Moreover the inverse of ϕ is obtained by removing the initial state, making the state a_k initial, and relabelling the states.

3.4 The k-Dyck boxed diagrams associated with the elements of $\mathcal{E}_n$

For $n \geq 2$, the image $\Psi(\mathcal{E}_n)$ is easy to characterize.

Lemma 2 *Denote by $\mathcal{F}_n$ the set of k-Dyck boxed diagrams*

$$((x_1, \cdots, x_{(k-1)n+1}), (y_1, \cdots, y_{(k-1)n+1}))$$

such that for all $i \in \{1, \cdots, k-1\}$, $x_i = 1$ and $y_i = 1$. For any $n \geq 2$, Ψ is a bijection between $\mathcal{E}_n$ and $\mathcal{F}_n$.

Proof: Let $A = \{a_1 < \ldots < a_k\}$. If $\mathcal{D} = (A, Q, \cdot, \varepsilon) \in \mathcal{E}_n$ then for $i \in \{1, \cdots, k-1\}$, $\varepsilon \cdot a_i = \varepsilon$. Moreover, ε is the only word of Q that does not start with a_k. Thus, the first $k-1$ missing transitions of $\mathcal{D}$ are (ε, a_1), $\cdots$, (ε, a_{k-1}). Therefore for any $i \in \{1, \cdots, k-1\}$, $\{u \in Q \mid u <_{lex} a_i\} = \{\varepsilon\}$ and $x_{(\varepsilon,a_i)} = 1$. Moreover since $1 \leq y_{(\varepsilon,a_i)} \leq x_{(\varepsilon,a_i)}$, $y_{(\varepsilon,a_i)} = 1$.

If $\mathcal{D} = (A, Q, \cdot, \varepsilon) \notin \mathcal{E}_n$, let i be the smallest integer such that $\varepsilon \cdot a_i \neq \varepsilon$. Then the word a_i is the smallest word in $Q \setminus \{\varepsilon\}$. If a missing transition (u, α) is such that $u\alpha <_{lex} a_i$, then $u = \varepsilon$ and $\alpha < a_i$: there are exactly $i-1$ such missing transitions. Hence, the i-th missing transition t_i, in the ordered sequence, is such that $x_{t_i} \geq 2$. □

4 Enumeration

This section is devoted to enumeration problems. The number of accessible automata is related to the Stirling numbers of the second kind whose definition and asymptotic estimate are recalled.

4.1 The Stirling numbers of the second kind

The Stirling number of the second kind $\{{n \atop m}\}$, where n and m are two non-negative integers, is the number of partitions of a set with n elements into m non-empty subsets.

Lemma 3 ([1]) *The number of boxed diagrams of width m and height n is equal to $\{{m+n \atop n}\}$.*

Recall that the LambertW-function [3] is the inverse of the function $x \to xe^x$. Its principal branch W_0 is real-valuted for x in $[-e^{-1}, +\infty[$ and is the unique branch which is analytic at zero. Its series expansion is

$$W_0(z) = \sum_{n=1}^{\infty} \frac{(-n)^{n-1}}{n!} z^n = z - z^2 + \mathcal{O}(z^3) \tag{1}$$

The Stirling numbers of the second kind are asymptotically estimated with the saddle point method.

Theorem 2 (Good [10]) *For n and m both tending towards infinity, and such that $n = \Theta(m)$, the following result holds:*

$$\{{n \atop m}\} \sim \frac{n!(e^\rho - 1)^m}{m!\rho^n \sqrt{2\pi n(1 - \frac{n}{m}e^{-\rho})}}$$

where $\rho = W_0(-\frac{n}{m}e^{-\frac{n}{m}}) + \frac{n}{m}$ is the unique positive root of the equation $m\rho = n(1 - e^{-\rho})$.

4.2 Enumeration of accessible deterministic automata

Recall that the number of automata in a specific class is equal to the number of transition structures of the same class multiplied by 2^n.

Complete automata

Korshunov [14] gave an asymptotic equivalent of the cardinality $|\mathcal{C}_n|$ of complete, deterministic and accessible transition structures, that can be reformulated [1] in terms of the Stirling numbers of the second kind:

Theorem 3 (Korshunov [13, 14]) *The number $|\mathcal{C}_n|$ of accessible complete and deterministic transition structures with n states on a k-letters alphabet satisfies*

$$|\mathcal{C}_n| \sim E_k\, n \left\{ {kn \atop n} \right\} \quad \textit{where} \quad E_k = \frac{1 + \sum_{r=1}^{\infty} \frac{1}{r} \binom{kr}{r-1} \left(e^{k-1}\beta_k\right)^{-r}}{1 + \sum_{r=1}^{\infty} \binom{kr}{r} \left(e^{k-1}\beta_k\right)^{-r}}, \quad \beta_k = \frac{(k\zeta_k)^k}{e^{k-1}(e^{\zeta_k}-1)}$$

and ζ_k is the positive root of $\rho = k(1-e^{-\rho})$.

Possibly incomplete automata

In the proof of the main theorem of this section, we use the following lemma:

Lemma 4 *For any fixed $k \geq 2$, as n tends toward infinity, one has*

$$\left\{ {kn+1 \atop n+1} \right\} \sim e^{\zeta_k} \left\{ {kn \atop n} \right\} \quad \textit{with} \quad \zeta_k = W_0(-ke^{-k}) + k.$$

Proof: The following proof is based on the comparison of the estimations of $\left\{ {kn \atop n} \right\}$ and $\left\{ {kn+1 \atop n+1} \right\}$ obtained with Theorem 2.

In the case of $\left\{ {kn \atop n} \right\}$, $\zeta_k = W_0(-ke^{-k}) + k$ is the positive root of $\rho = k(1-e^{-\rho})$. Theorem 2 and Stirling's formula give (see [1] for details):

$$\left\{ {kn \atop n} \right\} \sim \frac{(kn)!}{n!} \frac{(e^{\zeta_k}-1)^n}{\zeta_k^{kn}\sqrt{2\pi kn(1-ke^{-\zeta_k})}} \sim \alpha_k \beta_k^n n^{(k-1)n-1/2}$$

with $\alpha_k = (2\pi(\zeta_k - (k-1)))^{-\frac{1}{2}}$ and $\beta_k = \frac{(k\zeta_k)^k}{e^{k-1}(e^{\zeta_k}-1)}$.

Denote by f the function $f(x) = W_0(-xe^{-x}) + x$. To use Theorem 2 for $\left\{ {kn+1 \atop n+1} \right\}$ we have to compute $\rho_{n,k} = f\left(\frac{kn+1}{n+1}\right) = f(k - \frac{k-1}{n+1})$. Because of the analycity of f, we can use Taylor expansion:

$$\rho_{n,k} = f\left(k - \frac{k-1}{n+1}\right) = f(k) - \frac{k-1}{n+1} f'(k) + \mathcal{O}\left(\frac{1}{n^2}\right) = \zeta_k - (k-1)f'(k)\frac{1}{n} + \mathcal{O}\left(\frac{1}{n^2}\right).$$

From Theorem 2 we get:

$$\left\{ {kn+1 \atop n+1} \right\} \sim \frac{(kn+1)!(e^{\rho_{n,k}}-1)^{n+1}}{(n+1)!\rho_{n,k}^{kn+1}\sqrt{2\pi(kn+1)(1-\frac{kn+1}{n+1}e^{-\rho_{n,k}})}}.$$

Usual estimations and Stirling's formula lead to:

$$\frac{(kn+1)!}{(n+1)!} \sim e^{-(k-1)n} k^{3/2} k^{kn} n^{(k-1)n}$$

$$\sqrt{2\pi(kn+1)(1-\frac{kn+1}{n+1}e^{-\rho_{n,k}})} \sim \sqrt{2\pi kn(1-ke^{-\zeta_k})}$$

$$(e^{\rho_{n,k}}-1)^{n+1} \sim (e^{\zeta_k}-1)^{n+1} e^{-\frac{(k-1)f'(k)e^{\zeta_k}}{e^{\zeta_k}-1}}$$

$$\rho_{n,k}^{kn+1} \sim \zeta_k^{kn+1} e^{-\frac{k(k-1)f'(k)}{\zeta_k}}$$

Moreover as ζ_k satisfies $\zeta_k = k(1-e^{-\zeta_k})$, $e^{\zeta_k}/(e^{\zeta_k}-1) = k/\zeta_k$. Finally we obtain

$$\left\{ {kn+1 \atop n+1} \right\} \sim \alpha'_k \beta_k^n n^{(k-1)n-1/2}$$

with $\alpha'_k = \frac{e^{\zeta_k}}{\sqrt{2\pi(\zeta_k-(k-1))}}$. Thus $\left\{ {kn+1 \atop n+1} \right\} \sim e^{\zeta_k} \left\{ {kn \atop n} \right\}$, concluding the proof. □

Theorem 4 *The number $|\mathcal{D}_n|$ of accessible and deterministic transition structures of base automata with n states is $\Theta(n \left\{ {kn \atop n} \right\})$.*

Proof: First, as $\mathcal{C}_n \subset \mathcal{D}_n$, $|\mathcal{C}_n| \leq |\mathcal{D}_n|$. And Theorem 3 leads to the lower bound.

In Section 3 we exhibited a bijection in two steps between the set $\mathcal{D}_n$ and the set $\mathcal{F}_{n+1}$ of k-Dyck boxed diagrams $((x_1, \cdots, x_{(k-1)(n+1)+1}), (y_1, \cdots, y_{(k-1)(n+1)+1}))$ such that for all $i \in \{1, \cdots, k-1\}$, $x_i = 1$ and $y_i = 1$.

Now the number of elements in $\mathcal{F}_{n+1}$ is smaller than the number of boxed diagrams of width $(k-1)(n+1)+1 = (k-1)n+k$ and height $n+1$, whose $k-1$ first columns have height 1, and the last column has height $n+1$. Note that it is an overestimation of $|\mathcal{F}_{n+1}|$ since diagrams that do not satisfy the diagonal condition are taken into account. Therefore the elements of $\mathcal{F}_{n+1}$ are approximated by boxed diagrams made of $k-1$ columns of height 1, a boxed diagram of width $(k-1)n$ and height $n+1$ and a column of height $n+1$. There are $n+1$ possibilities for the last column. Thus, by Lemma 3, we obtain that $|\mathcal{D}_n| \leq (n+1) \left\{ {kn+1 \atop n+1} \right\}$. We conclude using Lemma 4. □

Corollary 1 *As n tends towards infinity, $|\mathcal{C}_n| = \Theta(|\mathcal{D}_n|)$.*

5 Random generation

In this section, in order to uniformly generate deterministic and accessible automata, we adapt an algorithm described in [1] and used to generate complete automata.

The first step of the algorithm is based on a Boltzmann sampler that generates specific set partitions. The second one consists of the transformation of these set partitions into accessible and deterministic automata.

5.1 A Boltzmann sampler to generate random partitions

The Boltzmann sampler used here is a direct application of the work of Duchon, Flajolet, Louchard and Schaeffer [6]. Boltmann samplers do not generate fixed size objects. They depend on a real parameter $x > 0$ and, for any given integer n, the value of x can be chosen so that the average size of the generated elements is n. The size is not fixed, but Bolztmann samplers guarantee that two elements of the same size have the same probability to be generated.

In order to uniformly generate set partitions of a set with $kn+1$ elements into $n+1$ non-empty subsets, we first consider the set of partitions of a set into $n+1$ non-empty sets. Its exponential generating function is $P_{n+1}(z) = \frac{(e^z-1)^{n+1}}{(n+1)!}$. Using Boltzmann sampler construction, each of the $n+1$ sets are generated assuming that its size follows a Poisson law $\text{Pois}_{\geq 1}$ of parameter x (a truncated Poisson variable K, where K is conditionned to be ≥ 1). The average size of the partition is then:

$$\mathbb{E}_x(\text{size of the partition}) = x\frac{P'_{n+1}(x)}{P_{n+1}(x)} = (n+1)x\frac{e^x}{e^x-1}.$$

Since we want a partition of a set having $kn+1$ elements, the value of the parameter x_n is chosen so that

$$(n+1)x_n\frac{e^{x_n}}{e^{x_n}-1} = kn+1,$$

that is, $x_n = \rho_{n,k}$ (see the proof of Lemma 4). When the Boltzmann parameter x_n is equal to $\rho_{n,k}$, the probability for a random partition to be of size $kn+1$ is [6]:

$$\mathbb{P}_{\rho_{n,k}}(N = nk+1) = \frac{\rho_{n,k}^{kn+1}[z^{kn+1}]P_{n+1}(z)}{P_{n+1}(\rho_{n,k})} = \frac{\left\{{kn+1 \atop n+1}\right\}\rho_{n,k}^{kn+1}}{(kn+1)!}\frac{(n+1)!}{(e^{\rho_{n,k}}-1)^{n+1}}$$

This quantity can be asymptotically estimated using the same method as in the proof of Lemma 4:

$$\mathbb{P}_{\rho_{n,k}}(N = nk+1) \sim \frac{\alpha_k}{\sqrt{kn}} \tag{2}$$

5.2 Random generator of deterministic and accessible automata

The two following algorithms, that are described into details in [1], basically change a set partition into a transition structure:

- `PartitionToBoxed(`$\mathcal{P}$`)` transforms a partition $\mathcal{P}$ of $\{1,\cdots,n\}$ in m non-empty subsets into a boxed diagram of width $n-m$ and height m. To do this, first order the subsets of $\mathcal{P}$ according to their smallest element: the smallest element is in E_1, the next ones are in E_2, $\cdots$, E_m. For any $i \in \{1,\cdots,n\}$ let $E(i)$ be the subset containing i. From left to right, for i from 1 to n, build a column of height $\max_{1\leq j\leq i}\{E(j)\}$ with a mark at height $E(i)$. Remove for every $i \in \{1,\cdots,n\}$ the left most occurence of a column of height i marked at height i.
- `kDyckBoxedToCn(`$\mathcal{B}$`)` builds from a k-Dyck boxed diagram $\mathcal{B}$ the associated accessible and deterministic transition structure as briefly described in Section 3. Let

  $$\mathcal{B} = ((x_1,\cdots,x_{(k-1)n+1})(y_1,\cdots,y_{(k-1)n+1}))$$

  be the k-Dyck boxed diagram. Then $(x_1,\cdots,x_{(k-1)n+1})$ determines uniquely the depth-first spanning tree of the transition structure, and $(y_1,\cdots,y_{(k-1)n+1})$ represents the states where the missing transitions end.

The following algorithm generates uniformly an accessible and deterministic automaton with n states over a k-letters alphabet

0. Compute $\rho_{n,k}$ using Equation (1) p.5. For fixed n and k this step is performed once.
1. Following Section 5.1, generate $n+1$ integers, $\lambda_1, \cdots, \lambda_{n+1}$, with a Poisson law $\mathrm{Pois}_{\geq 1}$ of parameter $\rho_{n,k}$.
2. If $\lambda_1 + \cdots + \lambda_{n+1} \neq kn+1$, return to step 1.
3. Generate uniformly a permutation σ of $\{1, \cdots, kn+1\}$. Label the partition $\mathcal{P} = E_1 \cup E_2 \cup \cdots E_{n+1}$ with σ. In other words, since $|E_i| = \lambda_i$, $E_i = \{\sigma(l) \mid l = \lambda_1 + \cdots + \lambda_{i-1} + j,\ j \in \{1, \cdots, \lambda_i\}\}$.
4. Use `PartitionToBoxed(`$\mathcal{P}$`)` to produce a boxed diagram $\mathcal{B}$ of width $(k-1)n$ and height $n+1$.
5. Add $k-1$ boxed columns of height 1 at the left of $\mathcal{B}$ and a column of height $n+1$, marked at random, at its right. The result is is a new boxed diagram $\mathcal{B}'$.
6. If $\mathcal{B}'$ is not a k-Dyck boxed diagram of size $n+1$, return to step 1.
7. Use `kDyckBoxedToCn(`$\mathcal{B}'$`)` to change $\mathcal{B}'$ into an element $\mathcal{A}' \in \mathcal{E}_{n+1}$.
8. Compute $\mathcal{A} = \phi^{-1}(\mathcal{A}')$, where ϕ^{-1} is described in Section 3.3.
9. For every state of $\mathcal{A}$, choose uniformly whether it is a final state or not.

Theorem 5 *The average complexity of this random generator is $\mathcal{O}(n^{3/2})$.*

Proof: All steps can be performed in linear time, if we do not take into account the rejects. In a rejection algorithm, if a test is positive with probability p, the average number of rejects is $1/p$. Therefore, as a consequence of Theorem 4 and Lemma 4, the average number of rejects at step 6 is bounded by a constant. Moreover Equation (2) p. 8 shows that step 2 produces in average $\mathcal{O}(\sqrt{n})$ rejects. □

5.3 Experimental results

The random generator has been implemented in `REGAL` [2], a `C++` library to generate random automata. We made some tests using this library mostly to compare deterministic and accessible automata with complete ones. For each test $10,000$ automata with $5,000$ states have been generated:

- For $k = 2$, 80.1% of automata are not complete. For $k = 3$, this proportion raises to 94.1%. Note that Lemma 4 show that proportions are similar before the rejection step.
- For $k = 2$, 85.4% of automata are minimal, it is roughtly the same proportion as for complete automata.
- For $k = 2$, 79.0% are strongly connected, it is about the same as for complete automata.
- For $k = 2$, in average an automaton has about 1.6 undefined transitions.

Acknowledgments The authors were supported by the ANR (project BLAN07-2_195422).

References

[1] F. Bassino, C. Nicaud, Enumeration and random generation of accessible automata, *Theoret. Comput. Sci.* 381 (2007), 86–104.

[2] F. Bassino, J. David, C. Nicaud. REGAL: a library to randomly and exhaustiv ely generate automata. In *12th International Conference on Implementation and Application of Automata (CIAA'07)*. vol. 4783. LNCS (2007), 303–305, Springer-Verlag.

[3] R. Corless, G. Gonnet, D. Hare, D. Jeffrey, D. Knuth, On the Lambert W-function, *Adv. in Comput. Math.*, 5 (1996), 329–359.

[4] J.-M. Champarnaud, T. Paranthoën, Random generation of DFAs, *Theoret. Comput. Sci.*, 330 (2005), 221–235.

[5] M. Domaratzki, D. Kisman, J. Shallit, On the number of distinct languages accepted by finite automata with n states, *J. Autom. Lang. Comb.*, no. 4 (2002), 469–486.

[6] 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.

[7] P. Flajolet, E. Fusy, C. Pivoteau, *Boltzmann Sampling of Unlabelled Structures*, Proceedings of Analytic Combinatorics and Algorithms Conference 2007 (ANALCO'07), SIAM Press.

[8] P. Flajolet, R. Sedgewick, *Analytic combinatorics*, Book in preparation, (Version of January, 1 2008, is available at http://www.algo.inria.fr/flajolet/publist.html).

[9] P. Flajolet, P. Zimmermann, B. Van Cutsem, A calculus of random generation of labelled combinatorial structures, *Theoret. Comput. Sci.*, 132 (1994), no. 1-2, 1–35.

[10] I. Good, An asymptotic formula for the differences of the powers at zero, *Ann. Math. Statist.*, 32 (1961), 249–256.

[11] M. A. Harrison, A census of finite automata, *Canadian Journal of Mathematics*, 17 (1965), 100–113.

[12] J. E. Hopcroft, J. Ullman, *Introduction to automata theory, languages, and computation*, Addison-Wesley, N. Reading, MA, 1980.

[13] D. Korshunov, Enumeration of finite automata, *Problemy Kibernetiki*, 34 (1978), 5–82, In Russian.

[14] A. D. Korshunov, On the number of non-isomorphic strongly connected finite automata, *Journal of Information Processing and Cybernetics*, 9 (1986), 459–462.

[15] V. Liskovets, The number of connected initial automata, *Kibernetika*, 5 (1969), 16–19, In Russian.

[16] V.A. Liskovets, Exact enumeration of acyclic automata, *Discrete Applied Mathematics* 154 (2006), 537–551.

[17] C. Nicaud, *Étude du comportement en moyenne des automates finis et des langages rationnels*, Ph.D. thesis, Université Paris 7, 2000.

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

[19] R. Robinson, Counting strongly connected finite automata, In *Graph theory with Applications to Algorithms and Computer Science*, Y. Alavi et al., Eds., p. 671–685, Wiley, 1985.

[20] J. Sakarovitch, *Eléments de théorie des automates*, Vuibert, 2003. English translation: *Elements of Automata Theory*, Cambridge University Press, to appear.

[21] V. Vyssotsky, A counting problem for finite automata, *Tech. report, Bell Telephone Laboratories*, May 1959.