Accessible and Deterministic Automata: Enumeration and Boltzmann Samplers

Frédérique Bassino[1†] and Cyril Nicaud[1‡]

[1] *Institut Gaspard Monge*
Université de Marne-la-Vallée
77454 Marne-la-Vallée Cedex 2 - France

received 10 Apr 2006, *revised* , *accepted* .

We present a bijection between the set $\mathcal{A}_n$ of deterministic and accessible automata with n states on a k-letters alphabet and some diagrams, which can themselves be represented as partitions of the set $[\![\, 1..(kn+1) \,]\!]$ into n non-empty parts. This combinatorial construction shows that the asymptotic order of the cardinality of $\mathcal{A}_n$ is related to the Stirling number $\left\{ {kn \atop n} \right\}$. Our bijective approach also yields an efficient random sampler of automata with n states, of complexity $O(n^{3/2})$, using the framework of Boltzmann samplers.

Keywords: finite automata, bijection, asymptotic enumeration, random generation, Boltzmann samplers

1 Introduction

To any regular language, one can associate in a unique way its minimal automaton, that minimizes the number of states. Therefore the space complexity of a regular language can be seen as the number of states of its minimal automaton. The worst case complexity of algorithms handling finite automata is most of time known [23]. But the average case analysis of algorithms requires the enumeration of the objects that are handled [7] and a good knowledge of their combinatorial properties. From a theoretical and practical point of view, a precise enumeration (see [5]) and algorithms of random generation of minimal automata is useful for the study of regular languages.

In this paper we address the problem of the enumeration of the set $\mathcal{A}_n$ of non-isomorphic accessible (also called initially connected) complete and deterministic automata with n states on a k-letters alphabet. These automata are not all minimal, but they contain minimal automata and experimentally, a constant proportion of them seems to be minimal [18, 3]. Moreover these automata constitute a very often used representation of regular languages even if they have more states than minimal automata. Empirically again, the minimization of such an automaton provides in average a gain of only one or two states.

The enumeration of finite automata according to various criteria (with or without initial state [13], non-isomorphic [11], up to permutation of the labels of the edges [11], with a strongly connected underlying

[†] email: `bassino@univ-mlv.fr`
[‡] email: `nicaud@univ-mlv.fr`

graph [16, 13, 21, 14], acyclic [17], accessible [15, 13, 21],...) is a problem that was studied since 1959 [22]. In particular Korshunov obtained [13] an asymptotic estimate of the cardinality $|\mathcal{A}_n|$ of $\mathcal{A}_n$ by successive estimations of the cardinalities of classes of graphs that approximate the underlying graphs of this class of automata.

In the following, we present a bijection between the set $\mathcal{A}_n$ of deterministic and accessible automata with n states on a k-letters alphabet and some diagrams, which can themselves be represented as partitions of the set $[\![\, 1..(kn+1) \,]\!]$ into n non-empty parts. Making use of these combinatorial transformations, we establish by a simple, but technical, estimation of the exact enumeration formula [18, 3] that $|A_n|$ is $\Theta\big(n 2^n \left\{ {kn \atop n} \right\}\big)$, where $\left\{ {kn \atop n} \right\}$ is a number of Stirling of second kind. We also reformulate the asymptotic estimate due to Korshunov [13] in the same terms as the bounds we obtained.

To generate uniformly at random accessible complete and deterministic automata with n states one can use a recursive algorithm [18, 3]. But this kind of method, introduced by Nijenhuis and Wilf [19] and systematized by Flajolet, Zimmermann and Van Custem [9], requires an important memory space. In this paper we present an algorithm, based on Boltzmann samplers [6], for the uniform random generation of the elements of $\mathcal{A}_n$ that runs in $\mathcal{O}(n^{3/2})$ time complexity with almost no precalculus.

This paper is an extended abstract of [2] in which the proofs of the results mentionned in the following can be found.

2 Bijective construction of automata

For every $n, m \in \mathbb{N}$ with $n \geq m$, we denote by $[\![\, m, n \,]\!]$ the set of integers $\{i \in \mathbb{N} \mid m \leq i \leq n\}$.

First recall some definitions about finite automata. Basic elements of theory of finite automata can be found in [12]. A *deterministic finite automaton* $\mathcal{A}$ on the 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 $Q \times A \mapsto Q$. If $\mathcal{A} = (A, Q, \cdot, q_0, F)$ is a deterministic finite automaton, we extend by induction its transition function to $Q \times A^* \mapsto Q$. 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 $q_0 \cdot u = q$. A finite automaton $\mathcal{A}$ is *complete* when for each $(q, \alpha) \in Q \times A$, $q \cdot \alpha$ is defined.

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

Our goal is to count the number $|\mathcal{A}_n|$ of accessible complete and deterministic automata with n states up to isomorphism and to generate these automata at random for the uniform distribution on $\mathcal{A}_n$.

2.1 *The set $\mathcal{D}_\mathbf{n}$ of structure automata*

We introduce a representation of the elements of $\mathcal{A}_n$, that allows us to enumerate them easily. A *simple path* in a deterministic automaton $\mathcal{A}$ is a path labelled by a word u such that for every prefix v and v' of u such that $v \neq v'$, $q_0 \cdot v \neq q_0 \cdot v'$. In other words, on 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 complete and deterministic finite automaton on the alphabet A and w be the map from Q to A^* defined for every state q of 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 a *base automaton* when $Q \subset A^*$ (the states are labelled by words) and for all $u \in Q$, $w(u) = u$. 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 complete and deterministic transition structures. Such structures exactly correspond to 2^n automata, since accessibility and determinism prevent distinct choices of final sets to form the same automaton. Denote by $\mathcal{D}_n$ the set of all the accessible complete and deterministic transition structures of base automata with n states, then $|\mathcal{A}_n| = 2^n|\mathcal{D}_n|$. Note that forbiding 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 at random for the uniform distribution on $\mathcal{D}_n$.

2.2 A bijection

In the following we establish a bijection between the transition structures of $\mathcal{D}_n$ and couples of integer sequences represented by 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 $x_m = n$, represented classically 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 couple 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.

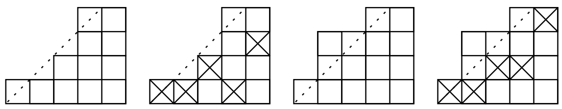


Fig. 1: A diagram of width 5 and height 4, a boxed diagram, a 2-Dyck diagram and a 2-Dyck boxed diagram

Theorem 1 ([18]) *The set $\mathcal{D}_n$ of accessible, complete and deterministic transition structures of size n on a k-letters alphabet is in bijection with the set $\mathcal{B}_n$ of k-Dyck boxed diagrams of size n.*

As a consequence, we get the following exact enumeration formula for $\mathcal{A}_n$ due to Nicaud [18] for two-letter alphabets and generalized to finite alphabets in [3].

Corollary 1 ([18, 3]) *For any integer $n \geq 1$, the number $|\mathcal{A}_n|$ of accessible, complete and deterministic non-isomorphic automata of size n on a k-letters alphabet is equal to $2^n|\mathcal{B}_n|$.*

From transition structures to k-Dyck boxed diagrams: we associate to any transition structure $\mathcal{D}$ of size n on a k-letters alphabet, using a depth-first algorithm, a k-Dyck boxed diagram of size n. Starting from q_0, recursively visit, for each state q that has not yet been visited, every $q \cdot a$, following the lexicographical order. If $q \cdot a$ has already been visited, store the current number of already visited states and the position of $q \cdot a$ in the prefix order as a part of the result, respectively on the first and second coordinate of the boxed diagram. In the execution of the algorithm, two kinds of transitions are distinguished in the structure: those who belong to the covering tree induced by the depth-first algorithm and the other ones that produce the integers of the result.

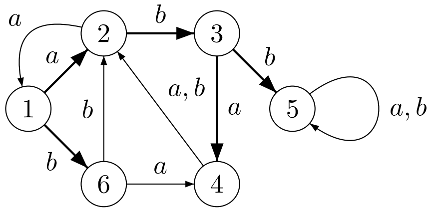


Fig. 2: A transition structure on a 2-letter alphabet having 1 as initial state

In the example given in Fig.2, the states are numbered following the prefix order and the bold edges correspond to the covering tree. Starting from state 1, consider first the transition $1 \cdot a = 2$, and then $2 \cdot a = 1$ that has already been visited. Therefore set $x_1 = 2$, since two states have already been visited and $y_1 = 1$ since $2 \cdot a = 1$. Next, consider the transitions $2 \cdot b = 3$ and $3 \cdot a = 4$. As $4 \cdot a = 2$, set $x_2 = 4$ and $y_2 = 2$, and so on. The result for this transition structure is the 2-Dyck boxed diagram $((2, 4, 4, 5, 5, 6, 6), (1, 2, 2, 5, 5, 4, 2))$ of size 5.

From an accessible complet and deterministic transition structure $\mathcal{D}$ of size n on a k-letters alphabet, the algorithm produces a k-Dyck boxed diagram, as there are kn transitions in $\mathcal{D}$ and $(n-1)$ of them belong to the covering tree of root q_0. The growth condition on the first sequence is due to the fact that the automata is deterministic and complete on a k-letters alphabet.

From k-Dyck boxed diagrams to transition structures: the idea is to reconstruct from any k-Dyck boxed diagram of size n of $\mathcal{B}_n$ its associated transition structure of size n on k-letters alphabet in $\mathcal{D}_n$.

We define a *missing transition* as a transition of the transition structure that has not yet been defined. The algorithm uses a stack S of missing transitions, initialized with all the transitions going from the initial state, put in reverse lexicographical order of their labels, so that the transition (i, a) where a is the smallest element of the alphabet is the first one to be selected.

Two indexes $i \in [\![1, (k-1)n + 1]\!]$ and $j \in [\![1, n]\!]$ indicate the current position in the graphical representation of the k-Dyck boxed diagram of size n

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

As long as $j < x_i$, the first element (q, a) (q is the state and a the letter of the missing transition) of the stack S is in the covering tree. Therefore the algorithm creates a new state q' and a transition $q \cdot a = q'$; moreover j is incremented by one and all the missing transitions (q', a) are added to the stack, in reverse lexicographical order of their labels.

When $j = x_i$, the first element of the stack is a transition that does not belong to the covering tree, then y_i becomes the image of the top of the stack $q \cdot a$ and i is incremented by one.

The algorithm runs while the stack S is not empty.

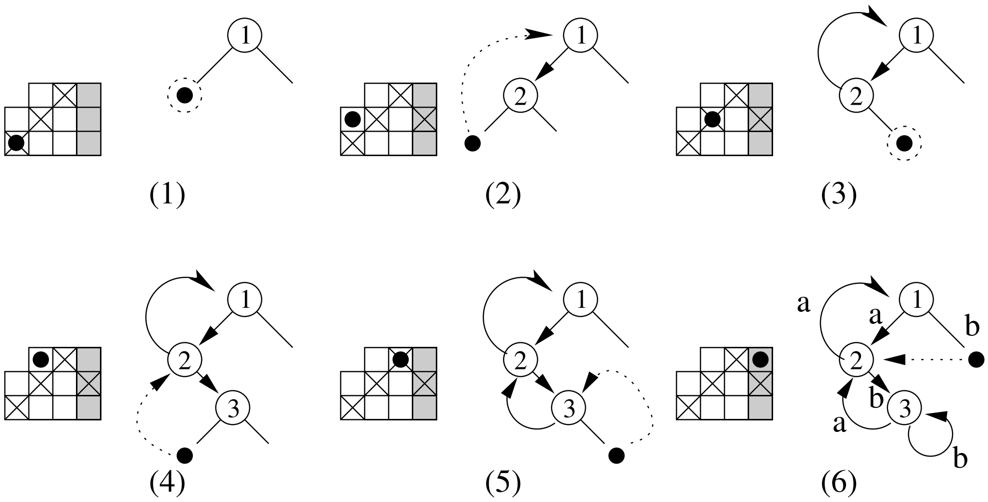


Fig. 3: From a 2-Dyck boxed diagram B' to a transition structure of $\mathcal{D}_n$

Fig. 3 shows an exemple of the execution of the algorithm on a two-letter alphabet, for

$$B' = ((2,3,3,3),(1,2,3,2)).$$

The grey column corresponds to the last transition. First create the initial state, set $i = j = 1$. At steps (1) and (3): as $j < x_i$ (the dot can go up), create a new state and its missing transitions, j is incremented (the dot goes up). At steps (2) and (4-6): $j = x_i$ (the dot can not go up): the missing transition is directed to the state y_i, and i is incremented (the dot goes right). At the end of step (6), the stack is empty. The algorithm ends.

Consequently, the set $\mathcal{D}_n$ of accessible, complete and deterministic transition structure of size n on a k-letters alphabet is in bijection with the set $\mathcal{B}_n$ of k-Dyck boxed diagram of size n. Moreover for any integer $n \geq 1$, the number $|\mathcal{D}_n|$ of accessible complete and deterministic transition structures of size n on a k-letters alphabet is equal to the number $|\mathcal{B}_n|$ of k-Dyck diagram of size n and $|\mathcal{A}_n| = 2^n|\mathcal{B}_n|$ as stated in Corollary 1.

3 Representation of set partitions

We describe in this part a bijection between boxed diagrams of width m and height n and set partitions of $n+m$ elements in n parts, based on a construction due to Bernardi [1].

Proposition 1 *The set of boxed diagrams of width m and height n and the of set partitions of $n+m$ elements into n parts are in bijection.*

Given a boxed diagram of width m and height n, we add n boxed columns $c_1, c_2, \ldots, c_n$. Each c_i is of height i and its highest box is marked. Each column is inserted at the left most position that statisfies the weakly increasing condition. Figure 4 gives an example of such a transformation.

The associated set partition is obtained from the sequence of the second coordinates $(y_1, \ldots, y_{m+n})$ corresponding to the marked boxes: two elements i and j are in the same part if and only if $y_i = y_j$.

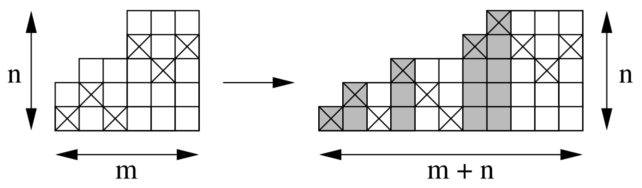


Fig. 4: From a boxed diagram to the set partition $\{\{1,3,6\},\{2,5\},\{4,10\},\{7,9,11\},\{8\}\}$

Now we present an algorithm that transforms a set partition $\mathcal{P}$ of a set with $m+n$ elements into n parts into its corresponding boxed diagram of width m and height n.

The input of the algorithm is a partition $\mathcal{P}$ given by an array `part`, with indices from 1 to $m+n$ and values in $[\![1,n]\!]$, such that $\texttt{part}[i] = \texttt{part}[j]$ if and only if i and j are in the same part of $\mathcal{P}$ and for every $j \in [\![2, m+n]\!]$ such that $\texttt{part}[j] \geq 2$, there exists $i < j$ such that $\texttt{part}[i] = \texttt{part}[j] - 1$. In other words, the parts of $\mathcal{P}$ are sorted in the order of their smallest element.

For instance, for $m = 3$ and $n = 4$, the partition $\{\{1,3,6\},\{5\},\{2,7\},\{4\}\}$ is represented by the array `part`:

`part`	1	2	1	3	4	1	2

Then, to each i in $[\![1, m+n]\!]$, associate the maximum m_i of $\texttt{part}[j]$ for $j \leq i$ and denote by `max` the new array containing the m_i's. Following with the previous example, we get:

`max`	1	2	2	3	4	4	4
`part`	1	2	1	3	4	1	2

Finally remove the columns with the first occurence of each value in `max`. In the example, we obtain:

2	4	4
1	1	2

The boxed diagram associated to the set partition $\{\{1,3,6\},\{2,7\},\{4\},\{5\}\}$ is then $((2,4,4),(1,1,2))$. The complexity in time and space of this algorithm is $\mathcal{O}(n+m)$.

4 Asymptotic order

In this section we give upper and lower bounds of the same order of magnitude for the numbers $|\mathcal{B}_n|$ of k-Dyck boxed diagram of size n and therefore for $|\mathcal{A}_n|$. The bounds are related to Stirling numbers of second kind. Next we reformulate a stronger result due to Korshunov [13] in the same terms as the bounds we obtained for $|\mathcal{A}_n|$.

4.1 The Stirling numbers of second kind

Recall that the *Stirling number of second kind*, denoted by $\left\{ {n \atop m} \right\}$, is the number of ways of partitioning a set of n elements into m nonempty subsets. By convention $\left\{ {0 \atop 0} \right\} = 1$, and for $n \geq 1$ we have $\left\{ {n \atop 0} \right\} = 0$. They can be computed using the following recurrence relation

$$\forall n, m > 0, \quad \left\{ {n \atop m} \right\} = m \left\{ {n-1 \atop m} \right\} + \left\{ {n-1 \atop m-1} \right\},$$

and asymptotically estimated with the saddle point method [8]. The following lemma is a special case of the asymptotic expansion obtained by Good [10] for Stirling numbers of the second kind $\left\{ {n \atop m} \right\}$ when n and m tend towards infinity with $n/m = \Theta(1)$.

Lemma 1 ([10]) *Let ζ_k be the postive root of $(\zeta_k - k)e^{\zeta_k} = -k$, then*

$$\left\{ {kn \atop n} \right\} = \alpha_k \beta_k^n n^{(k-1)n-1/2} \big(1 + \mathcal{O}(\frac{1}{n})\big)$$

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

4.2 Bounds

Theorem 2 *The number $|\mathcal{A}_n|$ of accessible, complete and deterministic automata with n states on a k-letters alphabet is $\Theta\Big(n\, 2^n \left\{ {kn \atop n} \right\}\Big)$.*

Recall that from Corollary 1 $|\mathcal{A}_n| = 2^n |\mathcal{B}_n|$. In the following, we briefly explain how to estimate the numbers $|\mathcal{B}_n|$.

The upper bound: Any k-Dyck boxed diagram of size n is also a boxed diagram of width $(k-1)n+1$ and height n whose last column is always of height n. We obtain an upper bound by removing the Dyck condition. As from Proposition 1 the set of boxed diagrams of width m and height n is in bijection with the set of set partitions of $n+m$ elements in n parts are in bijection, we obtain $|\mathcal{B}_n| \leq n \left\{ {kn \atop n} \right\}$.

The lower bound: The computation of a lower bound, which is more technical, is based on an overestimation of the number of boxed diagrams of width $(n-1)k+1$ and height n that are not k-Dyck boxed diagrams.

Recall that if $|z| < 1$, the *polylogarithm* function is defined as $\operatorname{polylog}(s, z) = \sum_{i=1}^{\infty} z^i / i^s$.

Proposition 2 *For all n large enough, one has the inequality*

$$|\mathcal{B}_n| \geq C_k\, n \left\{ {kn \atop n} \right\}$$

with $C_k = 1 - \sqrt{\frac{k-1}{2\pi k}} \operatorname{polylog}\left(\frac{1}{2}, \mu_k\right) + \mathcal{O}\left(\frac{1}{\sqrt[3]{n}}\right)$ *and* $\mu_k = \frac{k^k}{e^{k-1}(k-1)^{k-1}\beta_k}$.

Let $\mathcal{S}_{m,n}^{(>=i)}$ be the set of boxed diagrams of width m and height n whose first column is of height greater or equal to i. To prove Proposition 2, we consider for each boxed diagram which is not a k-Dyck boxed diagram the first integer i such that $x_{(k-1)i+1} = i$. The first index of a column that does not satisfy the Dyck condition is necessarily of this kind. The boxed diagram can then be decomposed into two parts:

a k-Dyck boxed diagram of size i on the left and an element of $\mathcal{S}^{(>=i)}_{(k-1)(n-i),n}$ on the right, as shown on Figure 5.

The number of boxed diagrams that are not k-Dyck boxed diagrams is then:

$$\sum_{i=1}^{n-1} |\mathcal{B}_i|\, |\mathcal{S}^{(>=i)}_{(k-1)(n-i),n}|$$

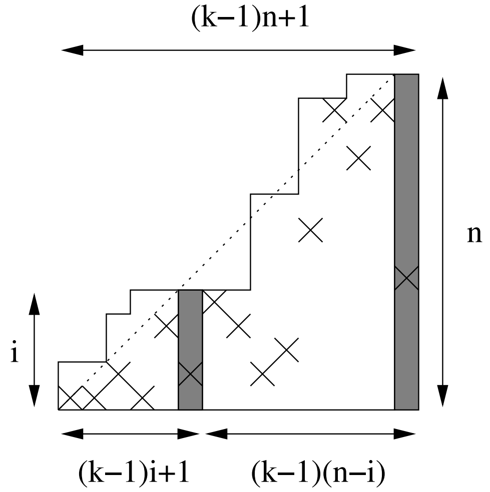


Fig. 5: Representation of the decomposition: the left part is a k-Dyck boxed diagram of size i

Next we compute an upper bound for this quantity, partitioning this summation in three parts, for $i \in [\![1, n/e]\!], i \in [\![n/e, n - \sqrt[3]{n}]\!]$ and $i \in [\![n - \sqrt[3]{n}, n-1]\!]$. The contribution of the two first parts is negligible and only the third part of the sum has the same order of magnitude as $n \left\{ {kn \atop n} \right\}$.

4.3 The estimate of Korshunov

We derived from simple bijective constructions the asymptotic order of magnitude of the number of accessible automata, giving a combinatorial interpretation that the asymptotic order is related to the number of set partitions $\left\{ {kn \atop n} \right\}$. Korshunov obtained a more precise result. He gave an asymptotic estimate [13, Theorem 4.8 p.51] of this number. His long proof is based on the estimations, when the number of states tends towards infinity, of cardinalities of classes of graphs that better and better approximate the underlying graphs of this class of automata. A key result [13, Theorem 3.4 p.33] is the estimation of the number of strongly connected graphs.

The link we made between the number of accessible automata and the number of set partitions allows us to reformulate the original estimate of Korshunov in the scale of the Stirling numbers, using their well known asymptotic estimate (see Lemma 1).

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

$$|\mathcal{A}_n| \sim E_k\, n\, 2^n \, \left\{ {kn \atop n} \right\} \quad \textit{where} \quad E_k = \frac{1 + \sum_{r=1}^{\infty} \frac{1}{r} \binom{kr}{r-1} \left(e^{k-1}\beta_k\right)^{-r}}{1 + \sum_{r=1}^{\infty} \binom{kr}{r} \left(e^{k-1}\beta_k\right)^{-r}}.$$

4.4 Numerical results

In the following array, we compare for alphabets of size $k = 2, 3$ and 4 the values of the ratio $\frac{|\mathcal{A}_n|}{2^n n \left\{ {kn \atop n} \right\}}$ for $n = 100$, 200, 300 and 400 with $E_k = \lim_{n\to+\infty} \frac{|\mathcal{A}_n|}{2^n n \left\{ {kn \atop n} \right\}}$. From Theorem 1, one has $|A_n| = n2^n f_n$ and the numbers f_n can be computed making use of a recurrence formula [18, 3]. The values of E_k are obtained from the formula given in Theorem 3. Note that E_k quickly converges towards 1, as k tends towards $+\infty$. For instance, $E_{26} \asymp 0.99999999987$.

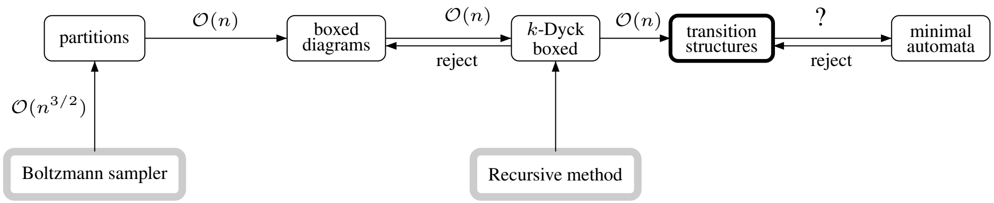
k	100	200	300	400	E_k
2	0.74490782	0.74497737	0.74498956	0.74499374	0.74499902
3	0.87341820	0.87342408	0.87342509	0.87342543	0.87342586
4	0.93931196	0.93931392	0.93931428	0.93931440	0.93931456

5 Random generation

In this section, we describe a new method, different from the recursive one proposed in [18, 3], to equally likely generate transition structures of size n, and thus automata by randomly adding the final states. It is based on Boltzmann samplers introduced in [6] and uses two algorithms that transform the objects: the first one to change a partition into a boxed diagram [1, 2], the other one to build an automaton from a element k-Dyck boxed diagram.

One step is achieved with a reject algorithm.

Moreover a reject algorithm can be used, as mentionned in the diagram, to equally likely generate a minimal automaton with n states. Empirically, in average, less than two draws from the set $\mathcal{A}_n$ are enough to obtain a minimal automaton. Nevertheless the efficiency of this algorithm is not yet proved.



The time complexity of the precalculus of the recursive method [18, 3] is $\mathcal{O}(n^3 \log n)$. The generation of each random element is then done in time $\mathcal{O}(n^2 \log n)$. Using floating-point arithmetic [4] instead of multi-precision one, with a slight loss of uniformity, the algorithm uses $\mathcal{O}(n^2)$ space, the precalculus requires $\mathcal{O}(n^2)$ time and the random generation of each automaton runs in $\mathcal{O}(n)$. Our algorithm runs in time $\mathcal{O}(n^{3/2})$ with almost no precalculus.

Boltzmann samplers Duchon, Flajolet, Louchard and Schaeffer [6], introduced a method to build a random generator for classes of objects that can be described with a combinatorial decomposition. Using automatic rules, the class is translated to a Boltzmann sampler which guarantee that two elements of the same size have the same probability to be generated.

Boltmann samplers do not generate objects of a fixed size. They depend on a real parameter $x > 0$ and, for any given an integer n, the value of x can be chosen such that the average size of the generated elements is n. The value of x can be computed by solving an equation that involves the generating function of the objects and its derivatives.

The behavior of the Boltzmann samplers is often such that the size of the generated object is between $(1-\varepsilon)n$ and $(1+\varepsilon)n$ with high probability. Therefore an exact size sampler can be obtained using a reject algorithm.

We use this technique to uniformly generate random set partitions of a set with kn elements into n nonempty subsets. We then use the constructions of Sections 3 and 2.2 to transform the set partition obtained into a transition structure.

In order to uniformly generate set partitions of a set with kn elements into n parts, we fist consider the set $\mathcal{P}_n$ of partitions of a set into n non-empty sets. The generating function of non-empty sets according to their sizes is $N(z) = e^z - 1$. Thus the generating fonction of $\mathcal{P}_n$ is $P_n(z) = \frac{(e^z-1)^n}{n!}$. Using Boltzmann sampler construction, we generate each of the n sets 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). This ensures that all resulting objects of the same size have the same probability to be generated. The average size of the partition is then:

$$\mathbb{E}_x(\text{size of the partition}) = x\frac{P_n'(x)}{P_n(x)} = nx\frac{e^x}{e^x-1}.$$

Since we want a partition of kn elements, we choose $x = x_n$ such that $nx_n\frac{e^{x_n}}{e^{x_n}-1} = kn$. With notations of Lemma 1, $x_n = \zeta_k$. Hence x_n is a constant function of n, only depending upon the size k of the alphabet. The Boltzmann sampler algorithm to uniformly generate a partition of a set of size kn into n parts is then:

```
BoltzmannSampler(n, k)
  computes the value of ζ_k
  do
    for each of the n nonempty sets E of P
      size(E) = NonZeroPoissonLaw(ζ_k)
    end for
  until (the sum of the sizes is equal to kn)
  return P
```

```
NonZeroPoissonLaw(x)
  k = 1 and p = (e^x − 1)^−1
  dice = uniform([0, 1[)
  while (dice >= p)
    dice = dice − p
    k = k + 1 and p = x ∗ p/k
  end while
  return k
```

To complete the task, label the structure obtained with a random permutation of $[\![1, kn]\!]$.

Using floating point approximation, the average cost of the generation of a partition is $\mathcal{O}(n)$. Testing if the sum of the sizes of the parts of such a partition is equal to kn is also linear. Therefore to compute the average complexity of this algorithm, we must estimate the probability that a partition has the correct size.

Since the generating function of these partitions is $P_n(z)$ and the Boltzmann parameter is equal to ζ_k, the probability for a random partition to be of size kn is [6]:

$$\mathbb{P}_{\zeta_k}(N = nk) = \frac{[z^{kn}]P_n(z)}{P_n(\zeta_k)} = \frac{\left\{ {kn \atop n} \right\}}{(kn)!} \frac{n!}{(e^{\zeta_k} - 1)^n}$$

Using Lemma 1 and Stirling formula, we obtain the following estimate: $\mathbb{P}_{\zeta_k}(N = nk) \sim \frac{\alpha_k}{\sqrt{kn}}$. Thus, the average number of rejects is $\mathcal{O}(\sqrt{n})$ and the average complexity of the random generation of an element F of $\mathcal{F}_n$ based on the Boltzmann sampler, using floating point approximation, is $\mathcal{O}(n^{3/2})$.

Open problem To conclude, the estimation of the proportion of minimal automata in $\mathcal{A}_n$ remains an important open problem. We conjecture that a constant proportion of accessible complete and deterministic automata of $\mathcal{A}_n$ is minimal. If it is true, the efficiency of a reject algorithm to generate minimal automata from accessible complete and deterministic ones would be proved and the asymptotic estimation $\Theta\big(n 2^n \left\{ {kn \atop n} \right\}\big)$ would also hold for minimal automata.

Acknowledgement We would like to thank an anonymous referee for its comments that greatly helped us improving the presentation of our results.

References

[1] O. Bernardi, A note on Stirling numbers, *preprint*.

[2] F. Bassino, C. Nicaud, Enumeration and random generation of accessible automata, *preprint*, available at http://igm.univ-mlv.fr/ bassino/publi.html.

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

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

[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*, Special issue on Analysis of Algorithms, 13 (2004), 577–625.

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

[8] P. Flajolet, R. Sedgewick, *Analytic combinatorics*, Book in preparation, (Version of August 16, 2005 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] A. 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. A. Liskovets, The number of connected initial automata, *Kibernetika*, 5 (1969), 16–19, In Russian.

[16] V. A. Liskovets, Enumeration of non-isomorphic strongly connected automata, *Vesci Akad. Navuk BSSR, Ser. Fiz.-Mat. Navuk* 3 (1971), 26-30, in Russian.

[17] V. A. Liskovets, Exact enumeration of acyclic automata, in *FPSAC'03*, available at http://www.i3s.unice.fr/fpsac/FPSAC03/ARTICLES/5.pdf.

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

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

[20] C. E. Radke, Enumeration of strongly connected sequential machines, *Inform. Control*, 8 (1965), 377-389.

[21] 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.

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

[23] S. Yu, Q. Zhuang and K. Salomaa, The state complexities of some basic operations on regular languages, *Theoret. Comput. Sci.*, 125 (1994), 315–328.