Enumeration of minimal acyclic automata via generalized parking functions

Jean-Baptiste Priez[†]

LRI, Université Paris-Sud, Orsay, France

Abstract. We give an exact enumerative formula for the minimal acyclic deterministic finite automata. This formula is obtained from a bijection between a family of generalized parking functions and the transitions functions of acyclic automata.

Résumé. On donne une formule d'énumération exacte des automates finites déterministes acycliques minimaux. Cette formule s'obtient à partir d'une bijection entre une famille fonctions de parking généralisées et les fonctions de transitions des automates acycliques.

Keywords: minimal acyclic deterministic finite automata, finite language, generalized parking function, species

Introduction

The study of the enumeration of minimal acyclic deterministic finite automata (MADFA) has been undertaken by several authors in the last decade. DOMARATZKI and *al.* [DKS02, Dom03, Dom04] presented different lower and upper bounds, CÂMPEANU and HO [CH04] gave a good upper bound of MADFA with some constraints, ALMEIDA and *al.* [RMA05, AMR07, AMR08] gave a canonical representation of MADFA and obtained a method for exact generation.

In this paper, we refer to the study of LISKOVETS [Lis06]. The latter gave a recurrence relation to enumerate acyclic finite deterministic automata (ADFA). The main idea of [Lis06] has been to define an extended notion of ADFA with more than one *absorbing state*. Unfortunately its approach of the enumeration of ADFA is not fine enough to enumerate MADFA. The goal of this paper is to give a finer enumeration of ADFA. In particular, the formula given in this paper expresses properties on the right language of ADFA. Therefore, it allows to enumerate MADFA. The main tool is a bijection with generalized parking functions.

VIRMAUX and the author studied in [PV] the generalization of parking functions defined by [SP02] and gave a generalized generating series in non-commutative symmetric functions as in [NT08]. Those generating series, called the non-commutative Frobenius characteristic of the natural action of the 0-Hecke algebra, contain substantial information on combinatorial objects.

In the first section, we recall the definition of extended ADFA of [Lis06] and then enrich the definition with some constraints.

[†]jean-baptiste.priez@lri.fr

In the second section, we recall the background on generalized parking functions and we give an isomorphism with a noteworthy family of generalized parking functions. We explicit the bijection and we transport some interesting properties on the right language of ADFA directly on parking functions. The substantial information provided by the Frobenius characteristic shows us how to extract sub-families of parking functions. We use this information to extract the sub-family of parking functions which exactly encodes (by the bijection) the ADFA such that all states have their right language distinct. Finally in the same way as [Lis06], one uses a bijection between ADFA and couple of extended ADFA (with constraints) and MADFA. This defines a recurrence relation for the enumeration of minimal acyclic deterministic finite automata.

1 Acyclic deterministic finite automata

For a basic background on automaton, the reader may consult [Hop79].

A *deterministic finite automaton* (DFA) of n states (labeled by a set N) over an alphabet Σ of k symbols is a tuple (i, A, δ) with

- $i \in N$ the *initial state*,
- $A \subset N$ the *accepting states*, and
- $\delta : N \times \Sigma \to N \cup \{\varnothing\}$ the *transition function*.

The special state $\varnothing$ is called the *absorbing state*. We consider N the set of states to avoid worrying about "well-labeled" states from 1 to n.

Extended transition function We extend the transition function δ recursively to words on Σ:

$$\delta^* \;:\; N \cup \{\varnothing\} \times \Sigma^* \quad \longrightarrow \quad N \cup \{\varnothing\}\,,$$

by setting $\delta^*(q, aw) := \delta^*(\delta(q, a), w)$, for any $w \in \Sigma^*$ and any $a \in \Sigma$; $\delta^*(q, \epsilon) = q$ (with ϵ the empty word), for any state $q \in N$; and, $\delta^*(\varnothing, w) := \varnothing$, for any w.

Transitions of a state We denote δ_q the underlying transition function at q defined by $\delta_q(a) := \delta(q, a)$.

Accepting status The *accepting status* of a state q denotes if q is accepting or not (*true* or *false*).

Right language The *right language* of a state q is the language: $RL(q) := \{w \in \Sigma^* \mid \delta^*(q, w) \in A\}$. Two states q and r are *right language equivalent* if $RL(q) = RL(r)$. If $RL(q) \neq RL(r)$ then one says q and r are *distinguished*. The language recognized by the automaton (i, A, δ) is the right language $RL(i)$.

Acyclicity An DFA is *acyclic* (an ADFA) if there is no non-empty sequence of transitions from a state to it self. Formally, one has $\delta^*(q, w) = q$ *if and only if* $w = \epsilon$, for any state $q \in N$.

Reachability A DFA is *reachable* if any state is *reachable* from the *initial state*. That means there exists a word $w \in \Sigma^*$ such that $\delta^*(i, w) = q$ for any state q.

Coreachability A DFA is *coreachable* if for all state reaches an *accepting state*. That means there exists w such that $\delta^*(q, w) \in A$ for any state q.

Non initial DFA We extend the definition of DFA to structures (A, δ) without *initial state*.

Minimal DFA A DFA is *minimal* if there is no DFA with fewer state which recognizes the same language.

1.1 Minimal ADFA

An important point will be the notion of *simple DFA*:

Definition 1: *A DFA is* simple *if all its states are* distinguished.

Proposition 1: *If a DFA is* simple *then it doesn't have a non-trivial automorphism.*

This proposition expresses the problem of counting M(A)DFA having labeled or unlabeled states are equivalent. So in the following, we consider automata as labeled combinatorial structures/objects. Moreover, from the definition of simple automata, it is easy to use the MYHILL-NERODE theorem about *minimal DFA*:

Theorem 1 (Myhill-Nerode): *A DFA is* minimal *if and only if it is* reachable*,* coreachable*, and* simple*.*

In the following, we enumerate *non-initial ADFA* which are *coreachable* and *simple*. Following [Lis06], we fix an initial state and extract the underlying *reachable* ADFA. Ultimately, the extracted ADFA is *reachable*, *coreachable* and *simple*, and so it is minimal.

1.2 Non-initial ADFA

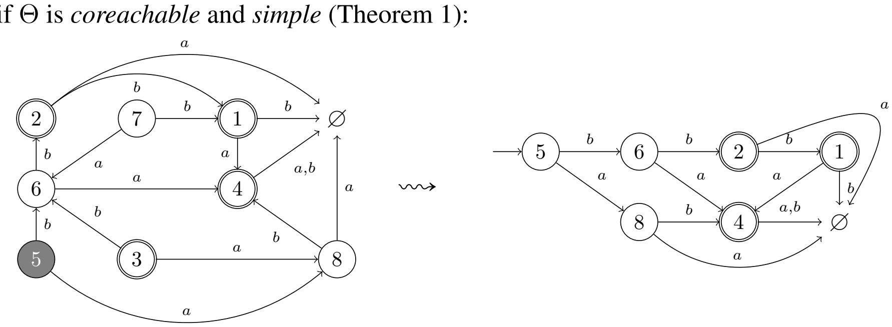
Proposition 2: *Let* Θ *be an ADFA.*
The absorbing state $\varnothing$ *is always* reachable *from any state of* Θ*.*

This proposition is obvious because there is no loop in an ADFA. For the upcoming bijection with parking functions, this point will be important. In particular, the bijection is based on the fact that there exists an order on states wherein $\varnothing$ is minimum. From this order we will put forward the states q such that $\delta(q, a) = \varnothing$ for any symbol $a \in \Sigma$ and which are *accepting* or not.

Lemma 1: *The automaton* Θ *is* coreachable *if all state* q *such that for any* $a \in \Sigma$*, one has* $\delta(q, a) = \varnothing$ *are* accepting.

The interpretation of states in parking functions will give a simple caracterization of generalized parking functions associated to (non-initial) *coreachable ADFA*. Furthermore the flexibility of the generalized parking function definition will easily give a family of parking functions, exactly encoding those coreachable ADFA.

We denote $\mathcal{N}_n^k$ the set of non-initial ADFA with n labeled states (and one absorbing state) over an alphabet of k symbols and call it the graded component of degree n. We also denote $\mathcal{N}^k := \bigsqcup_{n\geqslant 1} \mathcal{N}_n^k$ the graded set of all *non-initial ADFA*. Likewise, we denote $\mathcal{S}^k$ the graded set of *non-initial simple coreachable ADFA*. This first set $\mathcal{N}^k$ will be usefull to recall the extended notion of non-initial ADFA given in [Lis06, Quasi-acyclic automata, §2.15] and the second to define constraints on extended non-initial "simple coreachable" ADFA. Considering a non-initial ADFA Θ and a fixed state i, the sub-automata $\Theta^{(i)}$ extracted from all reachables states from i defines an (initial-connected) ADFA. We will extend ADFA to caracterize the complement of $\Theta^{(i)}$. Furthermore, an important remark is that the automaton $\Theta^{(i)}$ is *minimal* if Θ is *coreachable* and *simple* (Theorem 1):



(1)

A good definition of extended (coreachable and simple) ADFA (with constraint) gives a bijection between non-initial (coreachable and simple) ADFA and couple of extended (coreachable and simple) ADFA (with constraint) and (M)ADFA.

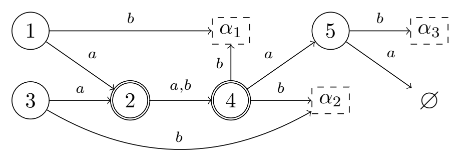
1.3 Extended non-initial ADFA

In [Lis06], the author introduces "quasi-acyclic automata". We call these objects extended non-initial ADFA (with t extra absorbing states), which means one considers tuples (A, T, δ) with A as the *accepting state set*, T the set of extra absorbing states, and δ an extension of the transition function definition:

$$\delta : N \times \Sigma \quad \longrightarrow \quad N \cup \{\varnothing\} \cup T$$

with N the set of n (labeled) states, Σ the alphabet. One denotes by $\mathcal{E}^{k,t} := \bigsqcup_{n \geqslant 0} \mathcal{E}^{k,t}_n$ the graded set of *extended non-initial ADFA* (with t extra absorbing states over an alphabet of k symbols).

Example 1: This example represents an extended non-initial ADFA with 3 extra absorbing states $T = \{\alpha_1, \alpha_2, \alpha_3\}$ over the alphabet $\{a, b\}$. This structure $(\{2, 4\}, \delta)$ is in $\mathcal{E}^{2,3}_5$ with $\delta(1, a) = 2$, $\delta(1, b) = \alpha_1$, $\delta(2, a) = \delta(2, b) = 4$, and so on.



Remark 1: An non-initial ADFA is an extended non-initial ADFA with 0 extra absorbing states, $\mathcal{N}^k = \mathcal{E}^{k,0}$.

1.3.1 Enumeration of underlying transition functions

In [Lis06], the author gives a formula $\mathfrak{d}$ to enumerate the number of extended transition function δ, underlying an extended non-initial ADFA:

$$\mathfrak{d}(k, t;\ n) = \sum_{j=0}^{n-1} \binom{n}{j} (-1)^{n-j-1} (j + t + 1)^{k(n-j)} \mathfrak{d}(k, t;\ j)\,, \qquad \text{[Lis06, Theorem 3.1]}$$

with k the cardinal of the alphabet, t the number of extra absorbing states and n the number of states. In the same way this formula can be adapted to enumerate $\mathcal{E}^{k,t}_n$:

Corollary 1 (of [Lis06, Theorem 3.1]): *The extended non-initial ADFA with n states and t extra absorbing states over an alphabet of k symbols, $\mathcal{E}^{k,t}_n$, are enumerated by the formula:*

$$\mathfrak{e}(k, t;\ n) = \sum_{j=0}^{n-1} \binom{n}{j} (-1)^{n-j-1} (2(j + t + 1)^k)^{n-j} \mathfrak{e}(k, t;\ j)\,,$$

for any $n \geqslant 1$ and $\mathfrak{e}(k, t;\ 0) = 1$.

PROOF: Immediate from the fact $\mathfrak{e}(k, t, n) = 2^n \mathfrak{d}(k, t, n)$. ∎

In the next section we use this formula to show their there is an isomorphism with some generalized parking functions using formula in [KY03] (also recall in [PV]).

1.3.2 Enumeration of ADFA

In this subsection, we recall a well-know method of counting connected graphs, used in [Lis06, Theorem 3.2]. This method points out that, for a fixed state i, for any *non-initial ADFA* Θ, one has a reversible splitting of Θ into an *ADFA* $\Theta^{(i)}$ (initially connected by a fixed state i, see (1)), and its complement $\bar{\Theta}^{(i)}$ which is an extended non-initial ADFA. It results an enumeration formula $\mathfrak{a}(k; t)$ of ADFA over an alphabet of k symbols with t states (and one fixed label: the initial state). This formula is given by the linear recurrence:

$$\mathfrak{e}(k,0;\ n) = \sum_{t=1}^{n} \binom{n-1}{t-1} \mathfrak{e}(k,t;\ n-t)\mathfrak{a}(k;\ t)\,. \qquad \text{[Lis06, Theorem 3.2]}$$

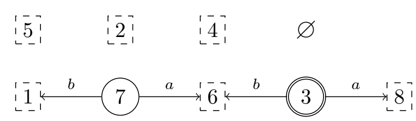
The complement $\bar{\Theta}^{(i)}$ is defined as follow: let $\Theta \in \mathcal{N}_n^k$ be a non-initial ADFA with state set is N and let $\Theta^{(i)} = (i, A_i, \delta_i)$ be an ADFA with its state set is N_i (the reachable states from i) and δ_i is the restriction of δ to states of N_i. We set $\bar{\Theta}^{(i)}$ the complement of $\Theta^{(i)}$ as an *extended non-initial ADFA* of $n - t$ states $\bar{N}_i = N \backslash N_i$ with t extra absorbing states N_i, the accepting states are $\bar{A}_i = A \cap \bar{N}_i$ and the extended transition function $\bar{\delta}_i$ is defined by:

$$\bar{\delta}_i : \begin{array}{ccc} \bar{N}_i \times \Sigma & \longrightarrow & N \cup \{\varnothing\}\,, \\ q \times a & \longmapsto & \delta(q,a)\,. \end{array}$$

We denote $\kappa_i : \mathcal{N}^k \to \mathcal{A}^k \times \mathcal{E}^k$ that splitting bijection (with $\mathcal{E}^k = \bigsqcup_{t \geqslant 1} \mathcal{E}^{k,t}$). The inverse bijection consists simply (in term of graph) to merge extra absorbing states q of $\bar{\Theta}^{(i)}$ with the state q of $\Theta^{(i)}$.

Remark 2: $N \cup \{\varnothing\} = \bar{N}_i \cup \{\varnothing\} \cup N_i$.

Example 2: In Equation (1), we have a non-initial ADFA Θ on the left and $\Theta^{(5)}$ on the right. We represent (here, on the right) $\bar{\Theta}^{(5)}$, the complement of $\Theta^{(5)}$, as an extended ADFA (with extra absorbing states framed with dashed rectangles).



1.4 Extended coreachable simple non-initial ADFA with constraints

In this subsection, we focus on *non-initial ADFA* which are *coreachable* and *simple*. We start by giving a definition of *extended simple non-initial ADFA*:

Definition 2: *An* extended non-initial ADFA (A, δ, T) *is* simple *if one has* $RL(q) \neq RL(r)$ *or there exists* $w \in \Sigma^*$ *such that* $\delta^*(q,w) \neq \delta^*(r,w)$ *with* $\delta^*(q,w) \in T$*, for any distincts states* q, r.

This definition is another way to say that if κ_i is applied on simple ADFA Θ then it gives a couple of simple structures: $(\Theta^{(i)}, \bar{\Theta}^{(i)})$. Unlike the splitting of [Lis06] described in the previous subsection, the restriction of κ_i to $\mathcal{S}^k$ is not a bijection. From any couple of simple structures an extended coreachable simple non-initial ADFA and a MADFA do not necessary produce a simple non-initial ADFA. The reason is (without well-chosen constraints) for some MADFA Θ and some extended simple (and coreachable) ADFA Π could share the "same" transitions on their respective states. Due to this therefore we add constraints to obtain a reversible splitting.

1.4.1 Preservation of simplicity and constraints

To be sure that the assembly of a couple of an *extended simple non-initial ADFA* and a *simple ADFA* remains *simple* (non-initial ADFA), one considers the couple of extended ADFA, and ADFA satisfying a set of constraints C. The idea is to forbid them from sharing states with the same transitions, and the same accepting status. A set of constraints C is a set of couples (ν, b) with $\nu : \Sigma \to T \cup \{\varnothing\}$ and b an accepting status (*true* or *false*). A couple (Θ, Π) of ADFA and extended ADFA satisfies C if

- The state set of Θ is T and for each state q of Θ there exists an unique couple (ν, b) such that $\delta_q = \nu$ and q is accepting if b is *true*,

- The set of extra absorbing states of Π is T and for any state q of Π and any couple (ν, b) of C, one has $\delta_q \neq \nu$ or the accepting status of q is the negation of b.

Proposition 3: *For any couple* (Θ, Π) *of simple structures satisfying* C*, one has* $\kappa_i^{-1}(\Theta, \Pi)$ *is simple.*

Remark 3: There does not always exist a couple of ADFA and extended ADFA that satisfies any set of constraints.

1.4.2 Preservation of coreachability

Both structures Θ and Π must not contain state q such that $\delta(q, a) = \varnothing$ (for any symbol a) and q is not accepting. Furthermore Π must not contain state q such that $\delta(q, a) = \varnothing$ at all. Otherwise, $\kappa_i^{-1}(\Theta, \Pi)$ is not coreachable. So we define the notion of *coreachability* of extended ADFA by:

Definition 3: *An extended ADFA is* coreachable *if there is no state* q *such that* $\delta(q, a) = \varnothing$*, for any* $a \in \Sigma$.

We denote by $\bar{\mathcal{E}}^{k,C}$ and $\mathcal{M}^{k,C}$ respectively the graded set of *extended simple coreachable ADFA* and of *MADFA* satisfying C. Due to this is a well-chosen definition of extended coreachable and simple ADFA, we obtain the bijection:

Lemma 2:

$$\mathcal{S}^k \simeq \bigsqcup_{C} \mathcal{M}^{k,C} \times \bar{\mathcal{E}}^{k,C} .$$

This gives us a description of coreachable and simple non-initial ADFA as the direct sum (over all sets of constraints C) of couple MADFA, and extended simple coreachable ADFA satisfying C. In the next section, the set of all available constraints will become aparent.

2 Generalized parking functions

In this section, we recall the constructive definition of generalized parking functions given by VIRMAUX and the author in [PV]. In a first part we define a bijection between of *non-initial ADFA* and a remarkable family of *generalized parking functions*. This bijection will reveal two interesting points:

- The localization of (some) *non-distinguishable* states. According to this, one extracts a sub-family of parking functions that are isomorphic to the *simple non-initial ADFA*.
- An easy translation of some constraints. In particular, in this first part, we give an analoguous family of parking functions which are isomorphic to the *coreachable non-initial ADFA*.

By combining both of these points we obtain an isomorphism with *(extended) simple coreachable ADFA*. Finally in the next section we go back to Lemma 2 to formulate a recurrence relation which enumerate MADFA.

2.1 Definition

Parking functions were first introduced in [KW66] to model hashing problems in computer science and appear in many different contexts in combinatorics, such as *labeled trees*, *prüfer sequence*, *hyperplane arrangements*, *etc*. A parking function on a finite set N is a function $f : N \to \mathbb{N}_+$ such that $\#f^{-1}([k]) \geqslant k$, for any $k \in [n]$ (with $n = \#N$ and $[k] = \{1, \cdots, k\}$). A generalization of parking functions was formulated in [SP02] and well studied in [KY03, PV]. Let $\chi : \mathbb{N}_+ \to \mathbb{N}$ be an non-decreasing function; a χ-parking function is a function f such that $\#f^{-1}([\chi(k)]) \geqslant k$, for any $k \in [n]$.

Remark 4: Usual parking functions are χ-parking function with $\chi(k) = k$.

2.2 Constructive definition

In this paper, we encode χ-parking functions by the sequence (Q_j) defined by $Q_j = f^{-1}(\{j\})$. We therefore define χ-parking functions on N as a sequence of $\chi(n)$ disjoint subsets (Q_j) of N satisfying:

$$\sum_{i=1}^{\chi(k)} \#Q_i \geqslant k\,, \qquad \text{for any } k \in [n]\,. \tag{2}$$

Remark 5: The parking function condition imposes that $\#f^{-1}(\{k\}) = 0$ for any $k > \chi(n)$. So the definition in terms of a set sequence allows to complete the sequence with an arbitrary sequence of empty sets. However one considers χ-parking functions as finite sequences of sets.

The main advantage of this definition (in terms of sequences of sets) is that it involves a natural recursive definition (see [PV, §2.1]). A convenient language for this is the *species theory* [BLL98] (or equivalently *decomposable combinatorial class* [FS09]). Let $\mathtt{E}$ be the species of sets (such that $\mathtt{E}[U] := \{U\}$ for any finite set U), let $\mathbf{1}$ be the characteristic empty species (such that $\mathbf{1}[U] = \{\varnothing\}$ if $U = \varnothing$ and $\varnothing$ in otherwise). We denote by $+$ (and $\sum$) the sum of species (disjoint union of labeled combinatorial structures: $(\mathtt{P}+\mathtt{Q})[U] = \mathtt{P}[U] \cup \mathtt{Q}[U]$), by $\cdot$ (, $\prod$ and the exponentiation) the product of species (cartesian product of labeled combinatorial structures: $(\mathtt{P}\cdot\mathtt{Q})[U] = \sum_{S \sqcup T = U} \mathtt{P}[S] \times \mathtt{Q}[T]$).

We directly give the recursive solution (of [PV, Eq. 2]) which defines χ-parking functions, $\mathcal{F}(\chi)$, grade by grade (see [PV, Proposition 2.5]) as a sum over all compositions π of the integer n, noted $\pi \models n$:

$$\mathcal{F}(\chi) = \mathbf{1} + \sum_{n\geqslant 1} \mathcal{F}_n(\chi) \qquad \text{with} \qquad \mathcal{F}_n(\chi) = \sum_{\pi\models n} \prod_{i=1}^{\ell(\pi)} \left(\mathtt{E}^{\Upsilon(\chi;\pi,i)}\right)_{\pi_i}\,, \tag{3}$$

where Υ is

$$\Upsilon(\chi;\pi,i) := \begin{cases} \chi(1) & \text{if } i = 1, \\ \chi(1+\pi(i-1)) - \chi(1+\pi(i-2)) & \text{otherwise,} \end{cases} \tag{4}$$

with $\pi(i) = \pi_1 + \cdots + \pi_i$ the partial sum of the first i parts of π.

Example 3: Here is the constructive definition of $\mathcal{F}_3(m^2)$ expanding (3):

$$\mathcal{F}_3(m^2) = \left(\mathtt{E}^1\right)_3 + \left(\mathtt{E}^1\right)_2 \cdot \left(\mathtt{E}^8\right)_1 + \left(\mathtt{E}^1\right)_1 \cdot \left(\mathtt{E}^3\right)_2 + \left(\mathtt{E}^1\right)_1 \cdot \left(\mathtt{E}^3\right)_1 \cdot \left(\mathtt{E}^5\right)_1$$

By abuse of notation we denote χ directly by its image over m. For example, we denote $\mathcal{F}(m^2)$ the generalized parking functions $\mathcal{F}(\chi)$ with $\chi(m) := m^2$. Furthermore by abuse again, we identify $\mathcal{F}_n(\chi)$ to the $\mathcal{F}(\chi)$-structures on the set $[n]$.

Example 4: We denote $(\{a,b,c,\cdots\},\{d,e,f,\cdots\},\ldots)$ by $(abc\cdots \mid def\cdots \mid \ldots)$. The first $\mathcal{F}_n(m^2)$-structures for $n = 0, 1$ and 2 are:

$$\mathcal{F}_0(n^2) = \{\ ()\ \}\,, \qquad \mathcal{F}_2(n^2) = \{\ (12 \mid \cdot \mid \cdot \mid \cdot), (1 \mid 2 \mid \cdot \mid \cdot), (1 \mid \cdot \mid 2 \mid \cdot), (1 \mid \cdot \mid \cdot \mid 2),$$
$$\mathcal{F}_1(n^2) = \{\ (1)\ \}\,, \qquad (2 \mid 1 \mid \cdot \mid \cdot), (2 \mid \cdot \mid 1 \mid \cdot), (2 \mid \cdot \mid \cdot \mid 1)\ \}\,.$$

And from Example 3, some of the 27 structures of $\mathcal{F}_3(m^2)$ resulting from $(\mathtt{E}^1)_2 \cdot (\mathtt{E}^8)_1$ are:

$$(12 \mid 3 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot),\ (12 \mid \cdot \mid 3 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot),\ \cdots,\ (12 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid 3),$$
$$(13 \mid 2 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot),\ (13 \mid \cdot \mid 2 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot),\ \cdots,\ (23 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid 1).$$

2.3 Enumeration and interpretation

In [KY03] the authors gave a recurrence relation to enumerate $\mathcal{F}_n(\chi)$-structures:

$$\mathfrak{f}(\chi;n) = \sum_{j=1}^{n}(-1)^{j-1}\binom{n}{j}\chi\,(n-j+1)^j\,\mathfrak{f}(\chi;n-j)\,. \qquad \text{[KY03, Theorem 4.2]}$$

From this formula and the Corollary 1, we immediately obtain:

Theorem 2: *There is a bijection between* $2(m+t)^k$*-parking functions and* extended non-initial ADFA $\mathcal{E}^{k,t}$.

Thanks to [KY03, Theorem 4.2] and Corollary 1. In the following we explicit the bijection. To do that we need to use a more precise formula to extract simple ADFA/parking functions. In [PV], we use the constructive definition (3) to obtain (automatically) this more expressive formula, *i.e.* the non-commutative Frobenius characteristic of the natural action of the 0-Hecke algebra on generalized parking functions [PV, Theorem 3.4] (expressed in complete $(\mathbf{S}^\pi)$-basis of non-commutative symmetric functions):

$$\mathbf{ch}(\mathcal{F}_n(\chi)) = \sum_{\pi\models n}\left(\sum_{\tau\models\ell(\pi)}\prod_{i=1}^{\ell(\tau)}\binom{\Psi_\tau(\chi;\pi,i)}{\tau_i}\right)\mathbf{S}^\pi \qquad (5)$$

with Ψ_τ a generalization of Υ (4):

$$\Psi_\tau(\chi;\pi,i) = \begin{cases}\chi\,(1) & \text{if } i=1,\\ \chi\,(1+\pi(\tau(i))) - \chi\,(1+\pi(\tau(i-1))) & \text{in otherwise,}\end{cases}$$

Remark 6: The complete non-commutative symmetric functions $(\mathbf{S}^\pi)$ are a convenient algebraic way to encode the action of relabeling of set sequence. The coefficient of $\mathbf{S}^\pi$ (with π a composition of n) is the number of χ-parking functions of n (upto isomorphism) such that the first non-empty set contains π_1 elements, the second one contains π_2 elements, and so forth.

Example 5:

$$\begin{aligned}\mathcal{F}_3(m^2) &= \underbrace{(\mathtt{E}^1)_3}_{} + \underbrace{(\mathtt{E}^1)_2\cdot(\mathtt{E}^8)_1}_{} + \underbrace{(\mathtt{E}^1)_1\cdot(\mathtt{E}^3)_2}_{} + \underbrace{(\mathtt{E}^1)_1\cdot(\mathtt{E}^3)_1\cdot(\mathtt{E}^5)_1}_{}\\ \mathbf{ch}(\mathcal{F}_3(m^2)) &= \quad \mathbf{S}^3 \quad + \quad 8\mathbf{S}^{21} \quad + \; 3\mathbf{S}^{12} + 3\mathbf{S}^{111} + \quad 15\mathbf{S}^{111}\end{aligned}$$

From the non-commutative characteristic (5) the specialization of $\mathbf{S}^\pi$ to the multinomial $\binom{n}{\pi_1,\cdots,\pi_k}$ gives automatically another formula: $\mathfrak{f}(\chi;n) = \sum_{\pi\models n}\left(\sum_{\tau\models\ell(\pi)}\prod_{i=1}^{\ell(\tau)}\binom{\Psi_\tau(\chi;\pi,i)}{\tau_i}\right)\binom{n}{\pi_1,\cdots,\pi_{\ell(\pi)}}$. Unfortunately this new expression of $\mathfrak{f}$ is a double sum over compositions of integers, so this is not (computationally) efficient. Finally we reuse that non-commutative characteristic $\mathbf{ch}(\mathcal{F}_n(\chi))$ to extract an enumeration of a sub-family of $\mathcal{F}(\chi)$ according to *simple ADFA* (§2.4.4). Meanwhile, one explicits the bijection between $\mathcal{F}(2(m+t)^k)$ and $\mathcal{E}^{k,t}$.

2.4 Explicit bijection $\mathcal{F}(2(m+t)^k) \simeq \mathcal{E}^{k,t}$

The following bijection is based on the parking functions condition (2) involving a natural division of (Q_j) into n factors splitting between each $\chi\,(i)$. This natural division will caracterize the sets of all transition functions ν from an alphabet Σ of k symbols into a set of p states fixing one state q. This caracterisation will make sure that the built automaton is acyclic.

2.4.1 Division factors

Let N be a finite set of cardinal n. The *division factors* of a χ-parking function (Q_j) on N is defined as the sequence of factor $(D_p)_{p\in[n]}$ with $D_p = (Q_j)$ with $j \in [\chi(j-1)+1, \chi(j)]$.

Example 6: The $\mathcal{F}(m^2)$-structures $(\ 3 \mid \cdot \mid 1 \mid \cdot \mid 2 \mid \cdot \mid \cdot \mid \cdot \mid \cdot\ )$ has as *division factors* (D_1, D_2, D_3) with $D_1 = (\ 3\ )$, $D_2 = (\ \cdot \mid 1 \mid \cdot\ )$ and $D_3 = (\ 2 \mid \cdot \mid \cdot \mid \cdot \mid \cdot\ )$.

2.4.2 Linear order on parking functions

Division factors will be used to fix an order between transitions according to a "fixing state". From now on we set a total order on N associated to any $\mathcal{F}(\chi)$-structures. We suppose that we are given a total order $<_N$ on N (in examples we use the natural order on $[n]$) and let (Q_j) be a χ-parking function on N. We define a second total order $<_q$ by:

$$q <_q q' \iff \begin{cases} q \in Q_k \text{ and } q' \in Q_{k'} & \text{with } k < k' \text{ or} \\ q, q' \in Q_k & \text{and } q <_N q'. \end{cases}$$

Remark 7: The order $<_q$ is the linear order defined by the inverse of the standardization of the parking functions seen as words. (In Example 6, the parking function (Q_j) could be represented by the word $w = 351$ (with $w_i = j$ *iff* $i \in Q_j$) and the inverse of it standardization is 312)

Example 7: The m^2-parking functions $(\ 4 \mid 12 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid 3 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot\ )$ on $[4]$ and $(\ 3 \mid \cdot \mid \cdot \mid 25 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid 1 \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid \cdot \mid 4 \mid \cdot \mid \cdot \mid \cdot \mid \cdot\ )$ on $[5]$ define respectively the orders $4 <_q 1 <_q 2 <_q 3$ and $3 <_q 2 <_q 5 <_q 1 <_q 4$.

2.4.3 Non-initial ADFA from $2m^k$-parking functions

In this subsection, we associate to any $\mathcal{F}(2m^k)$-structure a non-initial ADFA (A, δ) of $\mathcal{N}^k$. By definition this bijection will transport interesting properties on the right language of states. This construction will be easily generalized to (extended) non-initial ADFA with constraints, and finally extended to coreachable simple non-initial ADFA. Let (Q_j) be a $\mathcal{F}(2m^k)$-structures on N and $(D_p)_{p\in[n]}$ its *division factors*. We complete the total order $<_q$ with an other element/state, the absorbing state: $\varnothing <_q j$ for any $j \in N$. So we have $\varnothing =: q_0 <_q q_1 <_q q_2 <_q \cdots <_q q_n$.

Proposition 4: *There is exactly* $p^k - (p-1)^k$ *maps* $\nu : \Sigma \to \{q_1, \cdots, q_{p-1}\} \cup \{\varnothing\}$ *such that we have* $\nu(a) = q_{p-1}$ *for at least one* $a \in \Sigma$.

This proposition is obvious. By definition of (D_p) each factor D_p is a sequence of $2(p^k - (p-1)^k)$ sets. Each D_p is associated to the set of all maps $\nu : \Sigma \to \{q_1, \cdots, q_{p-1}\} \cup \{\varnothing\}$ such that we have $\nu(a) = q_{p-1}$ for at least one symbol $a \in \Sigma$. (In others terms, we associate the unique map $\nu : \Sigma \to \{\varnothing\}$ to D_1, the maps $\nu : \Sigma \to \{q_1\} \cup \{\varnothing\}$ such that $\nu(a) = q_1$ for at least one $a \in \Sigma$, are associated to D_2 and so on.)

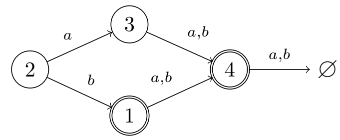
For a fixed total order $<_\Sigma$ on Σ: $a_1 <_\Sigma a_2 <_\Sigma \cdots <_\Sigma a_k$, the lexicographical order on the sequence of the image of ν seen as words: $\nu(a_1)\nu(a_2)\cdots\nu(a_k)$ defines a total order on maps ν: $\nu_1 < \nu_2 < \cdots < \nu_{p^k-(p-1)^k}$ in each *division factors*. We denote $\nu_j^{(p)}$ the maps associated to the factor D_p. We whizz up all that to finally define the bijection. Let ζ be the map which associates to any (Q_j) the automaton structure (A, δ) defined by:

- the set of states is N and the absorbing state is $\varnothing$,
- the set of accepting states A is the union of Q_j with j even,

- the transition function δ is setting by $\delta_q := \nu_j^{(p)}$ *iff* $q \in Q_{2(p-1)^k+2j-1}$ or $q \in Q_{2(p-1)^k+2j}$.

Example 8: Let (Q_i) be the $2m^2$-parking functions described by the division factors: $D_1 = (\;\cdot\; | \; 4 \;)$, $D_2 = (\;\cdot\; | \;\cdot\; | \;\cdot\; | \;\cdot\; | \; 3 \; | \; 1 \;)$, D_3 a sequence of 10 empty sets and D_4 a sequence of 16 sets with only the 7th is non-empty set is: $Q_{25} = \{2\}$ $(25 = (4-1)^2 + 7)$.

The non-initial ADFA $\zeta(Q_i)$.



Let (Q_i) be a $2m^k$-parking function.

Proposition 5: *The automaton structure $\zeta(Q_i)$ is a non-initial ADFA of $\mathcal{N}_n^k$.*

PROOF: From the construction, each state has transitions defined, so it is a well defined non-initial DFA. Then the parking function condition (2) and the definition of δ assert that the structure is acyclic. ∎

Lemma 3: *The map ζ is a bijection.*

PROOF: The different total orders ($<_N$, $<_q$, $<_\Sigma$ and the lexicographical order on maps ν) assert that each non-initial ADFA is produced once and Theorem 2 asserts that each is produced. ∎

The map ζ associates the same transitions to each state in the same set Q_j. Furthermore, the order on Q_j defined by the parking function involves a "height property" as in [Rev92] to define an efficient minimization.

Lemma 4: *The non-initial ADFA $\zeta(Q_j)$ is simple and coreachable if and only if $\#Q_j \leqslant 1$, for any $j \in [2n^k]$ and $Q_1 = \varnothing$.*

PROOF: The idea is that if there exists distincts states q and s such that $RL(q) = RL(s)$ then there is a Q_j such that $\#Q_j > 1$ in (Q_j) as in [Rev92]. So if $\#Q_i \leqslant 1$ for any i then $\zeta(Q_i)$ is simple and the reciprocal is trivial. As seen in the previous section, a non-initial ADFA is coreachable if the states whose transitions going only to the absorbing state, are accepting. ∎

2.4.4 Simple parking functions

We define *simple parking functions* as the χ-parking function (Q_j) satisfying $\#Q_j \leqslant 1$ for any j.

Remark 8: The simple m-parking functions are permutations.

Example 9: In $\mathcal{F}_2(m^2)$ all parking functions are simple except $(12 \mid \cdot \mid \cdot \mid \cdot)$ (see Example 4).

The interpretation of Frobenius characteristic $\mathbf{ch}(\mathcal{F}(\chi))$ (5) reveals an easy way to extract an enumeration formula. The term $\mathbf{S}^{1^n}$ in $\mathbf{ch}(\mathcal{F}_n(\chi))$ encodes χ-parking functions such that $\#Q_i \leqslant 1$ (*cf.* Remark 6 and see [PV, Eq. (7) and (9)]).

Lemma 5: *The* simple χ-parking functions *are enumerated by:*

$$\mathfrak{s}(\chi; n) = n! \sum_{\tau \models n} \prod_{i=1}^{\ell(\tau)} \binom{\Xi(\chi;\tau,i)}{\tau_i} \quad \textit{with} \quad \Xi(\chi;\tau,i) = \begin{cases} \chi(1) & \textit{if } i = 1 \\ \chi(1+\tau(i)) - \chi(1+\tau(i-1)) & \textit{in otherwise.} \end{cases}$$

Remark 9: Simple χ-parking functions do not have non-trivial automorphisms.

2.4.5 Coreachability and parking functions

From ζ definition, the sets Q_1 and Q_2 (of a parking function) encode the states which have transitions to the absorbing states. States of Q_1 are non-accepting. The idea is to fix a constraint such that it is forbidden to have states in Q_1. More generally from the subsection 1.4, we want to consider $t+1$ constraints. The construction ζ can be easily generalized to *extended ADFA*, then we remark that all transitions to all absorbing states (extra ones and $\varnothing$) are associated to the *first division factors*.

The present parking function formalism allows us to easily consider structures with those constraints. The first division factor depends only of $\chi(1)$ and adding a (negative) constant to χ influence only the size of that first factors.

Theorem 3: *There is a bijection between* extended coreachable simple non-initial ADFA with $t+1$ constraints *and simple* $(2(m+t)^k - t - 1)$*-parking functions.*

3 Main result

In section 1 we showed that *simple non-initial ADFA* Θ can be described by couples $(\Theta^{(i)}, \bar{\Theta}^{(i)})$ of *MADFA*, and *extended coreachable simple non-initial ADFA with constraints* (Lemma 2). Thanks to Lemma 5, these sets of extended coreachable simple non-initial ADFA with $t+1$ constraints have same cardinality. Finally thanks to Theorem 3 we can enumerate the *(extended) coreachable simple ADFA*.

Theorem 4: *The MADFA over an alphabet of k symbols with n states (with i the initial state fixed) are enumerated by* $\mathfrak{m}(k;n)$ *satisfying following relation:*

$$\mathfrak{s}(2m^k - 1; n) = \sum_{t=1}^{n} \binom{n-1}{t-1} \mathfrak{s}(2(m+t)^k - t - 1; n-t)\mathfrak{m}(k;t)$$

with $\mathfrak{m}(k;1) = 1$.

Acknowledgements

I would like to thank anonymous reviewers for their precious comments and also François BERGERON for the truly rewarding summer internship in Montreal (supported by LIA LIRCO).

References

[AMR07] Almeida, Moreira, and Reis. Enumeration and generation with a string automata representation. *THEOR COMPUT SCI*, 387(2):93–102, 2007.

[AMR08] Almeida, Moreira, and Reis. Exact generation of minimal acyclic deterministic finite automata. *INT J FOUND COMPUT S*, 19(04):751–765, 2008.

[BLL98] Bergeron, Labelle, and Leroux. Combinatorial species and tree-like structures. 1998.

[CH04] Câmpeanu and Ho. The maximum state complexity for finite languages. *J. of Automata, Languages and Combinatorics*, 9(2-3):189–202, 2004.

[DKS02] Domaratzki, Kisman, and Shallit. On the number of distinct languages accepted by finite automata with n states. *J. of Automata, Languages and Combinatorics*, 7(4):469–486, 2002.

[Dom03] Domaratzki. Improved bounds on the number of automata accepting finite languages. In *LECT NOTES COMPUT SC*, pages 209–219. Springer, 2003.

$n\backslash k$	1	2	3	4
0	1	1	1	1
1	1	1	1	1
2	2	6	14	30
3	5	75	623	4335
4	14	1490	59766	1829410
5	42	41415	10182221	1739056185

(a) The number of (non-labelled) coreachable simple non-initial ADFA over an alphabet of k symbols: $\mathfrak{s}(2m^k - 1; n)/n!$.

$n\backslash k$	1	2	3	4
0	1	1	1	1
1	2	2	2	2
2	5	13	29	61
3	14	166	1298	8830
4	42	3324	124706	3727540
5	132	92718	21256346	3543721650

(b) The number of (non-labelled) (quasi-)simple non-initial ADFA over an alphabet of k symbols: $\mathfrak{s}(2m^k; n)/n!$

$n\backslash k$	1	2	3	4
1	1	1	1	1
2	2	6	14	30
3	4	60	532	3900
4	8	900	42644	1460700
5	16	18480	6011320	1220162880
6	32	487560	1330452032	1943245777800
7	64	15824880	428484011200	5307146859111120
8	128	612504240	190167920278448	23025057433925970000
9	256	27619664640	111649548558856000	149780070423407303443200
10	512	1425084870240	84001095774695390816	1396395902225576206029949920
11	1024	82937356685760	78954926089415009686528	17993790111404399137868446737600

(c) The number of (non-labelled) minimal ADFA over an alphabet of k symbols: $\mathfrak{m}(k; n)/(n-1)!$

Tab. 1: Some enumeration of (extended) simple ADFA. The first values of (1c) were presented in [AMR07].

[Dom04] Domaratzki. Combinatorial interpretations of a generalization of the genocchi numbers. *J. of Integer Sequences*, 7(04.3):6, 2004.

[FS09] Flajolet and Sedgewick. *Analytic combinatorics*. cambridge University press, 2009.

[Hop79] Hopcroft. *Introduction to automata theory, languages, and computation*. Pearson Education India, 1979.

[KW66] Konheim and Weiss. An occupancy discipline and applications. *SIAM J APPL MATH*, 14(6):1266–1274, 1966.

[KY03] Kung and Yan. Goncarov polynomials and parking functions. *J COMB THEORY A*, 102(1):16–37, 2003.

[Lis06] Liskovets. Exact enumeration of acyclic deterministic automata. *DISCRETE APPL MATH*, 154(3):537–551, 2006.

[NT08] Novelli and Thibon. Noncommutative symmetric functions and lagrange inversion. *ADV APPL MATH*, 40(1):8–35, 2008.

[PV] Priez and Virmaux. Non-commutative frobenius characteristic of generalized parking functions. *arXiv*.

[Rev92] Revuz. Minimisation of acyclic deterministic automata in linear time. *THEOR COMPUT SCI*, 92(1):181–189, 1992.

[RMA05] Reis, Moreira, and Almeida. On the representation of finite automata. In *Proc. of DCFS'05*. Citeseer, 2005.

[SP02] Stanley and Pitman. A polytope related to empirical distributions, plane trees, parking functions, and the associahedron. *DISCRETE COMPUT GEOM*, 27(4):603–602, 2002.