Master 2 Mathematics and Computer Science
Symbolic Dynamics. Lecture 0. Automata

Marie-Pierre Béal

University Gustave Eiffel
Laboratoire d'informatique Gaspard-Monge UMR 8049

Overview

- Basics of automata theory
  - Automata and recognizable languages
  - Deterministic automata
  - Minimal automata
  - Kleene's theorem
- Automata in symbolic dynamics
  - Automata and sequences
  - Deterministic and minimal automata of irreducible sofic shifts
  - Deterministic local automata

Basics of automata theory

Words

Let A be a finite alphabet. The elements of A are called letters or symbols.

A word on A is a finite sequence of elements of A, denoted by $a_0 a_1 \cdots a_{n-1}$.

The set of words on A is denoted by A^*, the *empty word* by ε, and he set of nonempty words by A^+.

A word u is a *block* (or is a *factor*) of a word w if $w = pus$ for some words p, s.

A *language* is a set of words.

Automata and recognizable languages

An *automaton* over the alphabet A is given by a set Q of *states*, a set $E \subseteq Q \times A \times Q$ of *edges*, and two sets $I, T \subseteq Q$ of *initial* and *terminal* states respectively. The automaton is denoted by (Q, E, I, T). The automaton is *finite* if A, Q, and E are finite sets. An edge (p, a, q) is also denoted by $p \xrightarrow{a} q$.

When I, T are not specified, we assume that $I = T = Q$. The automaton is also called a *directed labeled graph*.

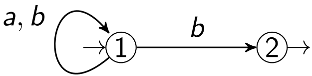
A (finite) *path* in the automaton $\mathcal{A}$ is a sequence $(p_i, a_i, p_{i+1})_{0 \leq i \leq n-1}$ of consecutive edges. Its *label* is the word $a_0 a_1 \cdots a_{n-1}$.

The language $L(\mathcal{A})$ *recognized* (or *accepted*) by the automaton $\mathcal{A} = (Q, E, I, T)$ is the set of labels of finite paths from I to T.

Two automata are *equivalent* if they recognize the same language.

A language L is *recognizable* (or *regular*) if there is a finite automaton $\mathcal{A}$ such that $L = L(\mathcal{A})$.

Example



The unique initial state 1 is indicated by an incoming arrow, and the unique terminal state 2 is indicated by an outgoing arrow.

The automaton represented recognizes the language A^*b of all words on A ending with b.

Matrix of an automaton

Given a finite automaton $\mathcal{A} = (Q, E, I, T)$ with set of edges E, the *matrix* of $\mathcal{A}$ is the $Q \times Q$-matrix $M = M(\mathcal{A})$ with nonnegative integer coefficients defined by

$$M_{p,q} = \text{Card}\{e \in E \mid e = (p, a, q) \text{ for some } a \in A\}.$$

An automaton is *irreducible* (or *strongly connected*) if, for every $p, q \in Q$, there is a path from p to q and there is a path from q to p.

An automaton $\mathcal{A} = (Q, E, I, T)$ on the alphabet A is *unambiguous* if, for every $p, q \in Q$ and $w \in A^*$, there is at most one path from p to q labeled by w.

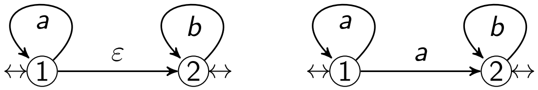
Generalized automata

A *generalized automaton* over the alphabet A is the same as an automaton $\mathcal{A} = (Q, E, I, T)$, but where the edges form a finite subset of $Q \times A^* \times Q$.
In particular, there can be edges labeled by the empty word (called ε-*transitions*). The notions of path and recognized language are defined straightforwardly.

Theorem

Every generalized finite automaton is equivalent to a finite automaton.

The proof involves eliminating ε-transitions.



Both automata recognize the language a^*b^*.

Deterministic automata

An automaton $\mathcal{A} = (Q, E, I, T)$ is *deterministic* if $I = \{i\}$, and for every $(p, a) \in Q \times A$, there is at most one edge (p, a, q) with source p and label a.

When it exists, the unique state q such that (p, a, q) is an edge is denoted by $p \cdot a$. The notation extends by associativity to words with $p \cdot ua = (p \cdot u) \cdot a$.

A deterministic automaton is usually denoted (Q, i, T), without explicitly indicating the set of edges.
The maps $p \mapsto p \cdot a$, for $p \in Q$ and $a \in A$ are called its *transitions*.

Finite automata that are not necessarily deterministic are often referred to as *non-deterministic* automata.

Determinization

A deterministic automaton $\mathcal{A} = (Q, i, T)$ is *complete* if for every $p \in Q$ and $a \in A$, there is an edge starting from p labeled a. Every deterministic automaton has a *completion* which is complete and recognizes the same language, achieved by adding at most one state.

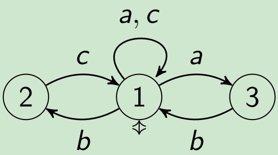
Proposition

For every finite automaton $\mathcal{A}$, there is a finite complete deterministic automaton equivalent to $\mathcal{A}$.

Indeed, let $\mathcal{A} = (Q, E, I, T)$ be a finite automaton.
Let $R = 2^Q$ the set of parts of Q and let F be the set of triples (U, a, V) where U, V are subsets of Q such that $V = \{q \in Q \mid p \cdot a = q \text{ for some } p \in U\}$.
It is easy to verify that $\mathcal{A}$ is equivalent to the deterministic automaton $(R, F, I, \mathcal{T})$ with $\mathcal{T} = \{U \subseteq Q \mid U \cap T \neq \emptyset\}$.
The computation of a deterministic automaton as described above is referred to as the *determinization algorithm*.

Example

Consider $L = \{a, ab, bc, c\}^*$. A non-deterministic automaton recognizing L is depicted in the figure on the left. Its determinization is shown on the right.



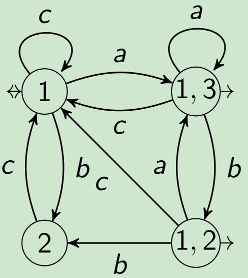


Figure: A non-deterministic automaton and its determinization.

Complement

The complement of the set recognized by a complete deterministic automaton $\mathcal{A} = (Q, i, T)$ is recognized by the automaton $(Q, i, Q \setminus T)$.

The following result follows from the fact that any recognizable language can be recognized by a complete deterministic automaton.

Proposition

The family of recognizable languages is closed under complement.

Proposition

The family of recognizable languages is closed under union and intersection

Let $\mathcal{A} = (Q, E, I, T)$ and $\mathcal{A}' = (Q', E', I', T')$ be two finite automata on the same alphabet A.

The *product* of $\mathcal{A}$ and $\mathcal{A}'$ is the automaton $\mathcal{A} \times \mathcal{A}' = (Q \times Q', F, J, S)$, where F is the set of $(p, p') \xrightarrow{a} (q, q')$ for all $p \xrightarrow{a} q$ in E and $p' \xrightarrow{a} q'$ in E', with $J = I \times I'$ and $S = T \times T'$.

It is easy to verify that the automaton $\mathcal{A} \times A'$ recognizes $L(\mathcal{A}) \cap L(\mathcal{A}')$.

Minimal automaton

For $L \subseteq A^*$ and $u \in A^*$, we define

$$u^{-1}L = \{v \in A^* \mid uv \in L\}. \tag{1}$$

This set is called a *left residual* of the language L.

Let Q denote the set of *nonempty residuals* of L, and let $\mathcal{A}(L) = (Q, i, T)$ be the deterministic automaton defined by

$$(u^{-1}L) \cdot a = (ua)^{-1}L,$$

with $i = L$ and $T = \{u^{-1}L \mid u \in L\}$.

This automaton is called the *minimal automaton*.

It recognizes L, is finite if and only if L is recognizable, and it has the minimal possible number of states among all deterministic (non necessarily complete) automata recognizing L.

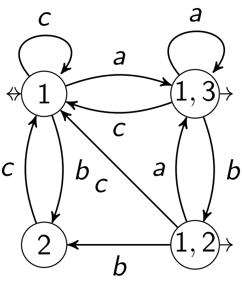
A deterministic automaton $\mathcal{A}$ recognizing a language L is *minimal* if it can be identified with $\mathcal{A}(L)$.
Note that these automata are not necessarily complete.

The minimal automaton $\mathcal{A}(L) = (Q, i, T)$ of a language L is not always complete, but it is trim (for each state p, there is a path from the initial state to p and there is a path from p to some terminal state).

The automaton $\mathcal{A}(L)$ can be obtained from any *trim deterministic automaton* $\mathcal{A} = (Q, i, T)$ recognizing L by considering the *Nerode equivalence* on Q defined as $p \sim q$ if $p \cdot u \in T \Leftrightarrow q \cdot u \in T$ for all $u \in A^*$. The automaton defined using the set of classes is the minimal automaton.

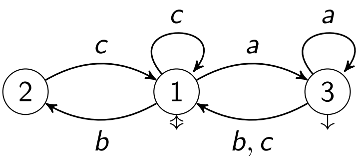
Minimal automaton: example

Consider the deterministic automaton below. The automaton is not minimal because the states $\{1\}$ and $\{1,2\}$ are Nerode equivalent. The equivalent minimal automaton is shown on the right.



The automaton is not minimal because the states $\{1\}$ and $\{1,2\}$ are Nerode equivalent.

The equivalent minimal automaton is:



Rational languages

Let A be an alphabet. For $L, M \subseteq A^*$, the *product* LM is defined by

$$LM = \{uv \mid u \in L, v \in M\},$$

and the *star* L^* by

$$L^* = \{u_1 u_2 \cdots u_n \mid n \geq 0, u_i \in L\}.$$

Note that $\varepsilon \in L^*$ for all L, and by convention $L^* = \{\varepsilon\}$ for $L = \emptyset$.

The family of *rational languages* is the smallest family of subsets of A^* that includes the finite subsets of A and which is closed under finite union, product, and star operations.

Thus, every rational language is described by a *rational expression* using words, unions, products, and stars.

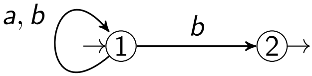
Theorem (Kleene)

On a finite alphabet, the family of rational languages coincides with the family of recognizable languages.

The proof in one direction consists in building an automaton for the product and for the star. In the other direction, it needs to build a rational expression for the language recognized by a finite automaton.

Rational languages: example

The language recognized by the automaton:



is the rational language $L = A^* b$.

It follows from Kleene's theorem that the class of rational languages is closed under complement, as this is true for recognizable languages.

Automata in symbolic dynamics

Sequences

Let A be a finite alphabet.

A two-sided sequence is a sequence $x = (x_n)_{n \in \mathbb{Z}} \in A^{\mathbb{Z}}$.

For $i \leq j$, we write

$$x_{[i,j]} = x_i x_{i+1} \cdots x_j \text{ and } x_{[i,j)} = x_i x_{i+1} \cdots x_{j-1}.$$

A word u *occurs in* a sequence x if $u = x_{[i,j)}$ for some i, j. One also says that u is a *block* of x, and that the integer i is an *occurrence* of u in x.

Automata accepting sequences

An *automaton* (or a *directed labeled graph*) (Q, E) over the alphabet A is given by a set Q of *states*, a set $E \subseteq Q \times A \times Q$ of *edges*. The automaton is *finite* if A, Q, and E are finite sets.
An edge (p, a, q) is also denoted by $p \xrightarrow{a} q$.
All states are both initial and final.

A bi-infinite *path* in the automaton $\mathcal{A}$ is a sequence $(p_i, x_i, p_{i+1})_{i \in \mathbb{Z}}$ of consecutive edges. Its *label* is the sequence $x = (x_i)_{i \in \mathbb{Z}} \in A^{\mathbb{Z}}$.

The set of sequences $X(\mathcal{A})$ *recognized* (or *accepted*, or *presented*) by the automaton $\mathcal{A} = (Q, E)$ is the set of labels of bi-infinite paths of $\mathcal{A}$.

Two automata $\mathcal{A}, \mathcal{B}$ are equivalent if $X(\mathcal{A}) = X(\mathcal{B})$.

A set of sequences X is *sofic* if there is a finite automaton $\mathcal{A}$ such that $X = X(\mathcal{A})$.

This automaton is called a *presentation* of X.

Sofic sets of sequences are called *sofic shifts*.

Let S denote the *shift transformation*, defined for $x \in A^{\mathbb{Z}}$ by $S(x) = y$ if $y_n = x_{n+1}$ for $n \in \mathbb{Z}$.

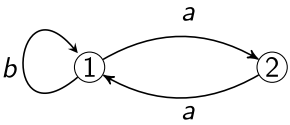
A set of sequences is *shift-invariant* if if $S(X) = X$ (or, equivalently $S^{-1}(X) = X$).
In informal terms, a set of sequences is shift-invariant if the position of the index 0 in a sequence x is irrelevant for determining whether it belongs to X.

By definition, sofic shifts are shift-invariant.

An automaton is *trim* if every state has at least one incoming edge and at least one outgoing edge.

A sofic shift is accepted by a trim automaton.

Example of a sofic shift: the even shift



The set $X(\mathcal{A})$ of labels of two-sided infinite paths in $\mathcal{A}$ is the set of bi-infinite sequences of letters over the alphabet $\{a, b\}$ having an even number of a between two b.

Deterministic automata accepting sequences

An automaton $\mathcal{A} = (Q, E)$ over A is *deterministic* if for every $(p, a) \in Q \times A$, there is at most one edge (p, a, q) with source p and label a.

Proposition

Every sofic shift is accepted by a trim deterministic automaton.

Indeed, let $\mathcal{A} = (Q, E)$ be a finite trim automaton.
Let $R = 2^Q$ the set of parts of Q and let F be the set of triples (U, a, V) where U, V are subsets of Q such that
$V = \{q \in Q \mid p \cdot a = q \text{ for some } p \in U\}$.

Let $\mathcal{D}$ be the trim part of (R, F), obtained from (R, F) by removing the states with no outgoing edge or no incoming edge.
Note that $\mathcal{D}$ has no state that is the empty set.

It is easy to verify that $\mathcal{A}$ is equivalent to the deterministic automaton $\mathcal{D}$.

Indeed, let x be a sequence labeling a path $(p_i, x_i, p_{i+1})_{i\in\mathbb{Z}}$ of $\mathcal{A}$. Let P_j be the set of states $p \in Q$ such that there is a left-infinite path labeled by $(x_i)_{i\leq j-1}$, with $x_{j-1} = p$.
Then, there is an edge (P_j, x_j, P_{j+1}) in $\mathcal{D}$ and x is a sequence accepted by $\mathcal{D}$.

Conversely, each label of a bi-infinite path of $\mathcal{D}$ is the label of some bi-infinite path of $\mathcal{A}$.
Indeed, let $(P_j, x_j, P_{j+1})_{j\in\mathbb{Z}}$ be a path of $\mathcal{D}$. For each $p_j \in P_j$, there is at least one $p_{j-1} \in P_{j-1}$ such that (p_{j-1}, x_{j-1}, p_j) is an edge of $\mathcal{A}$. And there is at least one state $p'_j \in P_j$ such that there is a right-infinite path in $\mathcal{A}$ starting at p'_j and labeled by $(x_i)_{i\geq j}$.

Language of a sofic shift

If X is a sofic shift, we denote by $\mathcal{B}(X)$ the set of blocks of the sequences of X.
This language $L = \mathcal{B}(X)$ is

- recognizable (or regular)
- extendable: for every $u, v \in L$, there are letters a, b such that $aub \in L$
- factorial: if $uwv \in L$, then $w \in L$

Irreducible automata and irreducible sofic shifts

An automaton $\mathcal{A} = (Q, E)$ is irreducible if it is strongly connected, *i.e.* if, for every $p, q \in Q$, there is a path from p to q and there is a path from q to p. Note that an irreducible automaton is trim.

An *irreducible sofic shift* is a set of sequences that is accepted by an irreducible automaton.

The language of an irreducible sofic shift is *transitive*: for every $u, v \in L$, there is a word w such that that $uwv \in L$.

Minimal automaton of an irreducible sofic shift

Proposition

Every irreducible sofic shift has a unique minimal deterministic automaton (irreducible deterministic, and with the fewest number of states among these automata).

Proof.

Start with a trim deterministic presentation $\mathcal{A} = (Q, E)$ of X.
For $q \in Q$, let
$\mathsf{Fut}(q) = \{w \in A^* \mid \text{there is a path labeled by } w \text{ starting at } q\}$.
We say that $\mathcal{A}$ is *reduced* if $p \neq q$ implies $\mathsf{Fut}(p) \neq \mathsf{Fut}(q)$.
If $\mathcal{A}$ is not reduced, there are two states $p \neq q$ with $\mathsf{Fut}(p) = \mathsf{Fut}(q)$.
Merge p, q: let $\mathcal{A}' = (Q', E')$ with $Q' = Q \setminus \{p, q\} \cup \{(p, q)\}$, where (p, q) is a new state and E' is the set of edges $(\pi(r), a, \pi(s))$ with (r, a, s) in E, $\pi(t) = t$ if $t \neq p, q$, $\pi(p) = \pi(q) = (p, q)$.
The automaton $\mathcal{A}'$ is still a trim deterministic presentation of X. □

Minimal automaton of an irreducible sofic shift

proof continued.

Let $\mathcal{A}$ be a reduced trim deterministic presentation of X.
The automaton $\mathcal{A}$ has a *synchronizing word*: a word w such that all paths labeled by w end in the same state q_w.
Indeed, let
$\mathrm{rank}(w) = \mathrm{Card}\{q \mid \exists$ a path labeled by w ending in $q\}$.
Let w be a word of minimal non-null rank.
Let us show that $\mathrm{rank}(w) = 1$.
If $\mathrm{rank}(w) > 1$, let (p, w, q), (p', w, q') be two paths with $q \neq q'$.
Since $\mathrm{Fut}(q) \neq \mathrm{Fut}(q')$, let u be a word such that $u \in \mathrm{Fut}(q) \setminus \mathrm{Fut}(q')$ (or the converse).
Then, $\mathrm{rank}(wu) < \mathrm{rank}(w)$ and $\mathrm{rank}(wu) \neq 0$.

□

proof continued.

Let $\mathcal{M}$ be the irreducible part of $\mathcal{A}$ containing q_w. Then $\mathcal{M}$ is an irreducible deterministic presentation of X that is reduced.
Indeed, let u be a block in $\mathcal{B}(X)$. There are blocks v, v' such that $wvuv'w \in \mathcal{B}(X)$. Hence, there is a path in $\mathcal{A}$ labeled $vuv'w$ from q_w to q_w, implying that u is the label of a path in $\mathcal{M}$.

Let $x \in X(\mathcal{M})$. Thus $x_{[-j,j]}$ is the label of a path $(p_i^{(j)}, x_i, p_{i+1}^{(j)})_{-j \le i \le j}$ of $\mathcal{M}$. There is an infinite number of j such that the states $p_0^{(j)}$ are all equal to p_0. Among them, there is an infinite number such that the $p_1^{(j)}$ are all equal to p_1, ... It follows that x is the label of the bi-infinite path $(p_i, x_i, p_{i+1})_{i \in \mathbb{Z}}$ and $x \in \mathcal{M}$.

Conversely, labels of bi-infinite paths of $\mathcal{M}$ belong to X. $\square$

Minimal automaton of an irreducible sofic shift

proof continued.

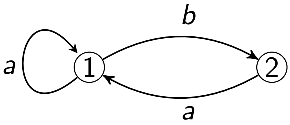
Finally, let $\mathcal{M}' = (Q', E')$ be another reduced irreducible deterministic presentation of X.
Let w be a synchronizing word for $\mathcal{M}$ and w' be a synchronizing word for $\mathcal{M}'$. There is a word u such that $wuw' \in \mathcal{B}(X)$. Thus $z = wuw'$ is a synchronizing word for both $\mathcal{M}$ and $\mathcal{M}'$.

We define a bijection f from Q to Q' as follows: $f(q_z) = q'_z$. If $q \in Q$ and u is the label of a path from q_z to q, we define $f(q)$ as the end of the unique path labeled by u going out of q'_z in $\mathcal{M}$.

The bijection f extends to a bijection between E and E' with $f(p, a, q) = (f(p), a, f(q))$.
Hence, $\mathcal{M}$ and $\mathcal{M}'$ are equal up to a renaming of the states. □

Deterministic local automata

A deterministic automaton $\mathcal{A} = (Q, E)$ is *local* if there is an integer n such that, for each word w of length n, all paths labeled by w end in the same state q_w.



A local automaton with $n = 1$.

Proposition

An deterministic automaton is local if and only if it has at most one cycle with a given label.

A *cycle* is a path $p = p_0 \xrightarrow{a_0} p_1 \xrightarrow{a_1} p_2 \ldots \xrightarrow{a_{m-1}} p_m = p$, where $m > 0$ is the length of the cycle.

Proof.

Exercise. $\square$

Solution

Proof.

Let $\mathcal{A}$ be a deterministic automaton. If $\mathcal{A}$ has two cycles sharing the same label w.

$$p = p_0 \xrightarrow{a_0} p_1 \xrightarrow{a_1} p_2 \ldots \xrightarrow{a_{m-1}} p_m = p',$$
$$p' = p'_0 \xrightarrow{a_0} p'_1 \xrightarrow{a_1} p'_2 \ldots \xrightarrow{a_{m-1}} p'_m = p'.$$

We have $p_i \neq p'_i$ for some $0 \leq i < m$.
Since $\mathcal{A}$ is deterministic, $p_i \neq p'_i$ for all $0 \leq i < m$.
Then, for any k, w^k is the label of a path ending in p and of a path ending in p'.
Thus, $\mathcal{A}$ cannot be local.

□

Solution

Proof.

Conversely, if $\mathcal{A}$ is not local, then, for any integer n there are two paths labeled by a word $w^{(n)}$ of length n ending in distinct states. We choose $n = (\text{Card } Q)^2$. These two paths are

$$p = p_0 \xrightarrow{a_0} p_1 \xrightarrow{a_1} p_2 \ldots \xrightarrow{a_{n-1}} p_n = p',$$
$$p' = p'_0 \xrightarrow{a_0} p'_1 \xrightarrow{a_1} p'_2 \ldots \xrightarrow{a_{n-1}} p'_n = p'.$$

If $p_i = p'_i$ for some $0 \leq i < n$, then $p = p'$, a contradiction. Hence $p_i \neq p'_i$ for all $0 \leq i < n$. By the pigeonhole principle, there are $0 \leq i < j < n$ such that $(p_i, p'_i) = (p_j, p'_j)$, implying the existence of two cycles sharing the same label.

□