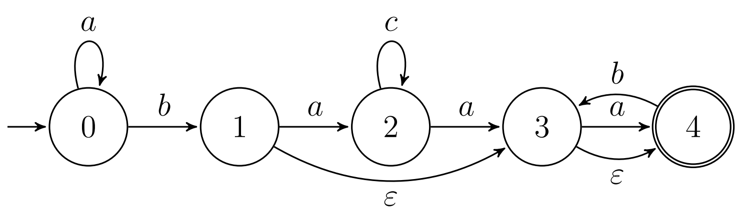
Langages Formels - TD 3

guillaume.scerri@lmf.cnrs.fr

11 février 2025

Exercise 1 : Déterminisation

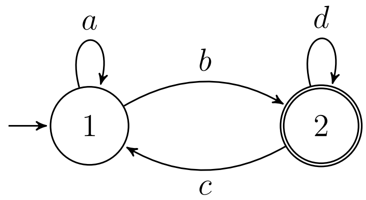
Quel est le langage reconnu par $\mathcal{A}$? Déterminisez $\mathcal{A}$.



(a) Automate $\mathcal{A}$

Exercise 2 :

Apply the algorithm of McNaughton-Yamada on the following automaton. Detail each step.



Exercise 3 : Brzozowski-McCluskey algorithm

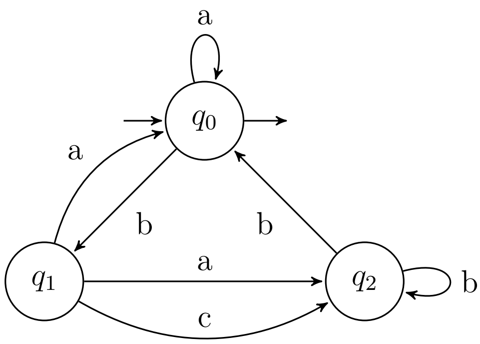
The goal of this exercise is to translate a finite automaton into a rational expression, giving an alternate proof of the associated implication of Kleene's theorem. We will proceed by successive transformations of the automaton.

1. We call *strongly normalized* every automaton which has a unique initial state to which no transition leads and a unique final state with no exiting transition, i.e. an automaton $\mathcal{A} = \langle Q, \Sigma, \{i\}, \{f\}, \delta\rangle$ such that for every state q and letter a, $(q, a, i) \notin \delta$ and $(f, a, q) \notin \delta$. Show that for all finite automaton, there is a strongly normalized automaton which recognizes the same language.

We will use a generalization of the definition of finite automata : the transition function will be a subset of $Q \times 2^{\Sigma^*} \times Q$. An execution of such an automaton recognizes the concatenation of languages of the transitions' labels. The automaton recognizes the union of the languages of all its accepting executions.

2. Show that every generalized automaton is equivalent to a generalized automaton in which there exists exactly one transition between each pair of states : $q' \in \delta(q, L)$ et $q' \in \delta(q, L')$ implies $L = L'$.

3. Let $\mathcal{A}$ be a strongly normalized generalized automaton with initial state i and final state f. Let $q \in Q_{\mathcal{A}}$, $q \notin \{i, f\}$. Show that there exists an automaton equivalent to $\mathcal{A}$ with set of states $Q_{\mathcal{A}} \setminus \{q\}$.
4. Conclude that if L is recognized by a strongly normalized generalized automaton $\mathcal{A}$, then L belongs to the rational closure of the labels of the transitions of $\mathcal{A}$.
5. Show that every finite automaton has an equivalent generalized automaton.
6. Give a procedure which, given a finite automaton, outputs a rational expression of same language.
7. Apply the construction to compute a rational expression corresponding to the following automaton :



Exercise 4 : ANTIMIROV's automaton

The goal of this exercise is to build an automaton from a regular expression. We will define a *partial derivative* operation $\partial_a(E)$ which corresponds to $a^{-1}\mathcal{L}(E)$ (via interpretation of expressions). Formally, for every expression E and letter a, we define the *set of expressions* $\partial_a(E)$ as follows :

$$\begin{aligned}
\partial_a(\underline{\emptyset}) &= \emptyset \\
\partial_a(\underline{b}) &= \begin{cases} \emptyset & \text{if } a \neq b \\ \{\underline{\emptyset}^*\} & \text{else} \end{cases} \\
\partial_a(E + E') &= \partial_a(E) \cup \partial_a(E') \\
\partial_a(E^*) &= \partial_a(E) \cdot \{E^*\} \\
\partial_a(E \cdot E') &= \begin{cases} \partial_a(E) \cdot \{E'\} & \text{if } \varepsilon \notin E \\ (\partial_a(E) \cdot \{E'\}) \cup \partial_a(E') & \text{else} \end{cases}
\end{aligned}$$

where concatenation is naturally extended over sets of expressions.
We define $\partial_w(E)$ for a word w inductively with $\partial_\varepsilon(E) = \{E\}$ and $\partial_{wa}(E) = \partial_a(\partial_w(E))$, where $\partial_w(S) = \bigcup_{E \in S} \partial_w(E)$ when S is a set of expressions.
Given a set of regular expressions S, we denote by $\mathcal{L}(S)$ the set $\bigcup_{E \in S} \mathcal{L}(E)$.

1. Give the partial derivatives of $(ab + b)^*ba$ by a and b.
2. Prove that for every $L, L' \subseteq \Sigma^*$ and $a \in \Sigma$,

$$a^{-1}(L \cup L') = (a^{-1}L) \cup (a^{-1}L')$$
$$a^{-1}L^* = (a^{-1}L) \cdot L^*$$
$$a^{-1}(L \cdot L') = \begin{cases} a^{-1}L \cdot L' & \text{si } \varepsilon \notin L \\ (a^{-1}L \cdot L') \cup (a^{-1}L') & \text{sinon} \end{cases}$$

3. Show that $\mathcal{L}(\partial_w(E)) = w^{-1}\mathcal{L}(E)$.

4. We define the set of non empty suffixes of a word :

$$\mathrm{Suf}(w) = \{\ v \in \Sigma^+\ :\ \exists u, w = uv\ \}$$

Show that for every $w \in \Sigma^+$:

$$\begin{aligned} \partial_w(E + E') &= \partial_w(E) \cup \partial_w(E') \\ \partial_w(E \cdot E') &\subseteq (\partial_w(E) \cdot E') \cup \bigcup_{v \in \mathrm{Suf}(w)} \partial_v(E') \\ \partial_w(E^*) &\subseteq \bigcup_{v \in \mathrm{Suf}(w)} \partial_v(E) \cdot E^* \end{aligned}$$

5. Let $\|E\|$ be the number of occurences of letters of Σ in E. Show that the set of partial derivatives different to E has at most $\|E\| + 1$ elements.
6. Conclude, and apply the construction to the expression $(ab + b)^*ba$.

Exercise 5 : A rational half ?

Let L be a rational language over a finite alphabet Σ. Show that $\mathsf{Half}(L) = \{\ f \in \Sigma^*\ :\ ff \in L\ \}$ is rational.
Is $\mathsf{FH}(L) = \{\ f \in \Sigma^*\ :\ \exists h \in \Sigma^*.\ |h| = |f|,\ fh \in L\ \}$ rational ?
Is $\mathsf{Double\text{-}pad}(L) = \{\ fh \in \Sigma^*\ :\ \ f \in L,\ h \in \Sigma^*,\ |h| = |f|\ \}$ rational ?