TD 1 – Langages, AFD et AFN

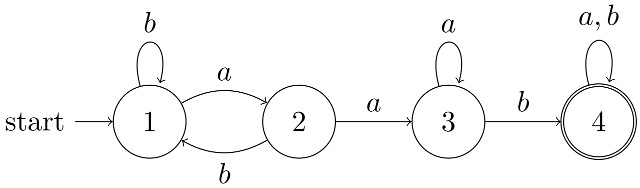
1 Construction d'AFD

Exercice 1. Pour chacun des langages suivants, donner un automate fini déterministe (AFD) le reconnaissant.

1. Les mots sur l'alphabet $\{a, b\}$ contenant le facteur aab ou $aaab$.
2. Les mots sur l'alphabet $\{a, b\}$ contenant un nombre pair de a et impair de b.
3. Les mots sur l'alphabet $\{a\}$ de longueur multiple de 3.
4. Pour chaque $d \in \mathbb{N}$ les mots sur l'alphabet $\{a\}$ de longueur multiple de d.
5. Les représentations binaires d'entiers pairs. Ici entier est entendu au sens positif et les nombres sont donnés dans l'ordre gros-boutiste (c'est-à-dire l'ordre normal de lecture des nombres : 1 puis 0 puis 1 puis 0 puis 1 puis 0 c'est le nombre binaire 101010 soit 42 en décimal).
6. Pour chaque $d \in \mathbb{N}$, les représentations binaires des entiers multiples de d.
7. Pour chaque $(d, c) \in \mathbb{N}^2$, les représentations binaires des entiers de la forme $c + k \cdot d$ pour $k \in \mathbb{N}$.

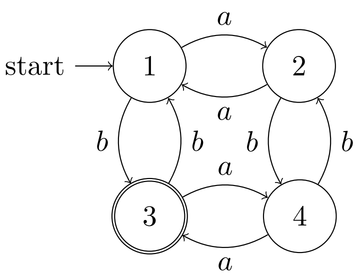
Solution 1.

1. On construit l'automate suivant :



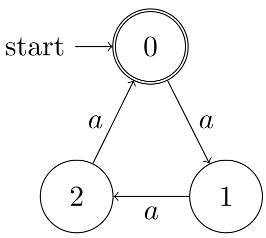
.

2. On construit l'automate suivant :



.

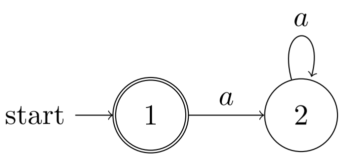
3. On construit l'automate suivant :



.

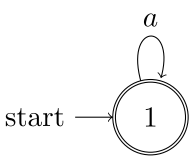
4. On distingue trois cas :

— $d = 0$:



.

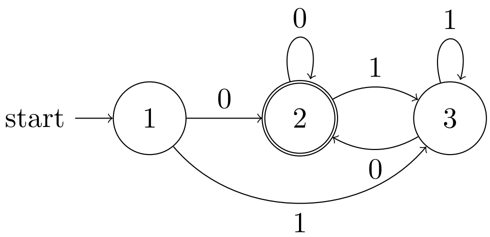
— $d = 1$:



.

— $d \geq 2$:
$(Q, \{a\}, \delta, q_0, \{q_0\})$ avec $Q = \{q_0, \ldots, q_{d-1}\}$ et $\delta(q_i, a) = q_{(i+1) \bmod d}$ pour tous $i \in [\![0, d-1]\!]$ et $a \in \Sigma$.

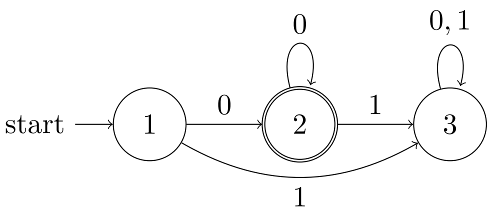
5. On construit l'automate suivant :



.

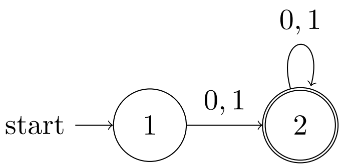
6. On distingue trois cas :

— $d = 0$:



.

— $d = 1$:



.

— $d \geq 2$:
$(Q, \{0, 1\}, \delta, s, \{q_0\})$ avec $Q = \{s, q_0, \ldots, q_{d-1}\}$ et $\delta \colon Q \times \Sigma \to Q$ telle que :
— $\delta(s, b) = q_b$ pour tous $b \in \{0, 1\}$;
— $\delta(q_i, b) = q_{(2i+b) \bmod d}$ pour tous $i \in [\![0, d-1]\!]$ et $b \in \{0, 1\}$.

7. On distingue deux cas :

— $d = 0$:
$(Q, \{0, 1\}, \delta, s, \{q_c\})$ avec $Q = \{s, q_0, \ldots, q_c, q_{c+1}\}$ et $\delta \colon Q \times \Sigma \to Q$ telle que :
— $\delta(s, b) = q_b$ pour tout $b \in \{0, 1\}$;
— $\delta(q_i, b) = \begin{cases} q_{2i+b} & \text{si } 2i + b \leq c \\ q_{c+1} & \text{sinon} \end{cases}$ pour tous $i \in [\![0, d-1]\!]$ et $b \in \{0, 1\}$.

— $d \geq 1$:
$(Q, \{0,1\}, \delta, s, \{q_{c,c \mod d}\})$ avec $Q = \{s\} \cup \{q_{t,r} \mid (t,r) \in \{0,\dots,c\} \times \{0,\dots,d-1\}\}$ et $\delta\colon Q \times \Sigma \to Q$ telle que :
— $\delta(s,b) = \begin{cases} q_{b,b \mod d} & \text{si } c \geq 1 \\ q_{c,b \mod d} & \text{sinon} \end{cases}$ pour tout $b \in \{0,1\}$;
— $\delta(q_{t,r}, b) = \begin{cases} q_{2t+b,(2r+b) \mod d} & \text{si } 2t+b \leq c \\ q_{c,(2r+b) \mod d} & \text{sinon} \end{cases}$ pour tous $(t,r) \in \{0,\dots,c\} \times \{0,\dots,d-1\}$ et $b \in \{0,1\}$.

2 Puzzles

Exercice 2. Pour tout alphabet Σ, donner l'ensemble des mots $(x,y) \in (\Sigma^*)^2$ tels que $xy = yx$.

Solution 2. Il s'agit de l'ensemble de toutes les paires de mots $(x,y) \in (\Sigma^*)^2$ telles qu'il existe $u \in \Sigma^*$ et $i, j \in \mathbb{N}$ tels que $x = u^i$ et $y = u^j$.

La démonstration se fait par récurrence sur $|xy|$, en observant que soit $x = \varepsilon$ ou $y = \varepsilon$, soit, sans perte de généralité, $|x| \geq |y|$ et on peut montrer qu'alors $x = yz$ avec $z \in \Sigma^*$, ce qui implique que $zy = yz$ avec $|zy| < |xy|$.

Exercice 3. Expliciter la forme des langages rationnels unaires (c'est-à-dire, sur un alphabet à une lettre).

Solution 3. Si $\Sigma = \{a\}$ est l'alphabet, ce sont les langages qui sont des unions finies, pour $k \in \mathbb{N}$ et $p \in \mathbb{N}_{>0}$ fixés, de langages de la forme $\{a^i\}$ pour $i \in \mathbb{N}, i < k$ et $\{a^j \mid j \in \mathbb{N}, j \geq k \wedge j \equiv r \mod p\}$ pour $r \in [\![0, p-1]\!]$.

3 Rationalité

Exercice 4. Étant donné un langage rationnel L sur un alphabet Σ, prouver que les langages suivants sont rationnels.

1. $Init(L) = \{u \in \Sigma^* \mid \exists v \in \Sigma^*, uv \in L\}$.
2. $Min(L) = \{w \in L \mid \nexists u \in L, u$ préfixe propre de $w\}$.
3. $Max(L) = \{w \in L \mid \forall u \in \Sigma^*, wu \in L \Rightarrow u = \varepsilon\}$.
4. $Cycle(L) = \{uv \in \Sigma^* \mid vu \in L\}$.
5. $\frac{1}{2}L = \{u \in \Sigma^* \mid \exists v \in \Sigma^*, uv \in L \wedge |v| = |u|\}$.

Solution 4.

1. Soit $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ un AFD qui reconnaît L, soit F' l'ensemble des états co-accessibles, c'est à dire les états $q \in Q$ tels qu'il existe $w \in \Sigma^*$ vérifiant $\hat{\delta}(q,w) \in F$.
   On a que l'AFD $(Q, \Sigma, \delta, q_0, F')$ reconnaît $Init(L)$.
2. Soit $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ un AFD qui reconnaît L. On supprime juste toutes les transitions sortant des états de F pour obtenir un AFN reconnaissant $Min(L)$.
3. Soit $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ un AFD qui reconnaît L. On limite les états finaux de $\mathcal{A}$ aux états finaux qui ne sont pas co-accessibles autrement que par le mot vide pour obtenir un AFD reconnaissant $Max(L)$.
4. Soit $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ un AFD qui reconnaît L.
   On suppose que $\varepsilon \notin \Sigma$ et on définit $Q' = \{q_0\} \cup Q \times Q \times \{0,1\}$.
   On définit $\delta'\colon Q' \times (\Sigma \cup \{\varepsilon\}) \to \mathfrak{P}(Q')$ de la manière qui suit :
   — pour tout $a \in \Sigma$, $\delta'(q_0, a) = \emptyset$ et $\delta'(q_0, \varepsilon) = \cup_{q \in Q}\{(q,q,0)\}$;

— pour tous $a \in \Sigma$, $q, p \in Q$, $\delta'((q,p,0),a) = \{(\delta(q,a),p,0)\}$ et

$$\delta'((q,p,0),\varepsilon) = \begin{cases} \emptyset & \text{si } q \notin F \\ (q_0,p,1) & \text{sinon} \end{cases} ;$$

— pour tous $a \in \Sigma$, $q, p \in Q$, $\delta'((q,p,1),a) = \{(\delta(q,a),p,1)\}$ et $\delta'((q,p,1),\varepsilon) = \emptyset$.

On a que l'AFN $(Q', \Sigma, \delta', q_0, \cup_{q \in Q}\{(q,q,1)\})$ reconnaît $Cycle(L)$.

5. Soit $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ un AFD qui reconnaît L.
   On suppose que $\varepsilon \notin \Sigma$ et on définit $Q' = \{q_0\} \cup Q^3$.
   On définit $\delta' \colon Q' \times (\Sigma \cup \{\varepsilon\}) \to \mathfrak{P}(Q')$ de la manière qui suit :
   — pour tout $a \in \Sigma$, $\delta'(q_0,a) = \emptyset$ et $\delta'(q_0,\varepsilon) = \cup_{q \in Q}\{(q_0,q,q)\}$;
   — pour tous $a \in \Sigma$, $(q,p,r) \in Q^3$, $\delta'((q,p,r),a) = \bigcup_{b \in \Sigma}\{(\delta(q,a),p,\delta(r,b))\}$ et $\delta'((q,p,r),\varepsilon) = \emptyset$.
   On a que l'AFN $(Q', \Sigma, \delta', q_0, \cup_{(q,r) \in Q \times F}\{(q,q,r)\})$ reconnaît $\frac{1}{2}L$.

4 Caractérisations

Soit Σ un alphabet.

Étant donné un AFD $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$, un état $q \in Q$ est dit *co-accessible* si et seulement s'il existe $w \in \Sigma^*$ vérifiant $\hat{\delta}(q,w) \in F$; il est appelé *puits* lorsque $\delta(q,a) = q$ pour tout $a \in \Sigma$. On dira de plus que q se *trouve dans un circuit* lorsqu'il existe $w \in \Sigma^+$ tel que $\hat{\delta}(q,w) = q$.

Exercice 5. Donner une condition nécessaire et suffisante pour qu'un langage sur Σ soit reconnu par un AFD dont les états co-accessibles non puits ne se trouvent dans aucun circuit.

Solution 5. Il faut que le langage soit une union finie de langages de la forme $\{uv \mid v \in \Sigma^*\}$ ou $\{u\}$ pour $u \in \Sigma^*$.

Exercice 6. Donner une condition nécessaire et suffisante pour qu'un AFD sur Σ reconnaisse un langage fini ou cofini.

Solution 6. Il faut que l'AFD vérifie que ses états co-accessibles non puits ne se trouvent dans aucun circuit et que ses états puits accessibles soient tous soit finaux, soit non finaux.