4 Théorie des Langages 2006·2007

Voici une correction (très rapide) des exercices de la feuille 4. Mais avant, quelques rappels...

4.1 Langage réguliers

obtenus à partir d'union $\cup$, concaténation $\cdot$ et d'étoile $*$ de langages finis

langage régulier	$\Leftrightarrow$	valeur d'une expression régulière
	$\Leftrightarrow$	reconnu par un automate fini
	$\Leftrightarrow$	engendré par une grammaire régulière (grammaire linéaire gauche ou linéaire droite)
	$\Leftrightarrow$	solution d'un système d'équations linéaires

4.2 Lemme d'itération (ou pumping lemma ou lemme de l'étoile...)

Soit $L \subseteq \Sigma^*$. **Si** L *est régulier* **alors** *il existe* $N > 0$ *tel que pour tout* $m \in L$, $|m| \geq N$
il existe u, v, w *tels que* $m = uvw$, $|v| > 0$
(on peut ajouter aussi $|uv| \leq N$, *donc en particulier* $|v| \leq N$*)*
et pour tout $i \geq 0$, $uv^iw \in L$.

(explication : N nbr d'états de l'automate, si un mot est plus long alors on boucle dans l'automate)

On utilise en général la contraposée pour montrer qu'un langage n'est pas régulier.

En aucun cas ce lemme ne permet de montrer qu'un langage est régulier.

4.3 Solution d'une équation

La plus petite solution de l'équation $X = LX + L'$ est $X = L^*L'$.
Cette solution est unique si $\epsilon \notin L$.

4.4 Automates

4.4.1 Automate fini $(\Sigma, Q, \Delta, D, F)$

- Σ ensemble fini de symboles (alphabet)
- Q ensemble fini d'états
- $\Delta : Q \times \Sigma \times Q$
  Δ est une relation qu'on peut voir comme une fonction $\Delta : Q \times \Sigma \to 2^Q$, $\forall q \in Q, \forall x \in \Sigma, \Delta(q, x) = \{q' \in Q \mid (q, x, q') \in \Delta\}$
  qu'on peut étendre à une fonction $\Delta : 2^Q \times \Sigma \to 2^Q$, $\forall X \subseteq Q, \forall x \in \Sigma, \Delta(X, x) = \bigcup_{q \in X} \Delta(q, x)$
  qu'on peut étendre à $\Delta : 2^Q \times \Sigma^* \to 2^Q$, $\forall X \subseteq Q, \Delta(X, \epsilon) = X$ et $\forall X \subseteq Q, \forall x \in \Sigma, \forall u \in \Sigma^*, \Delta(X, xu) = \Delta(\Delta(X, x), u)$
- $D \subseteq Q$ est l'ensemble des états initiaux
- $F \subseteq Q$ est l'ensemble des états finals

4.4.2 Automate fini déterministe $(\Sigma, Q, \delta, s, F)$

- Σ ensemble fini de symboles (alphabet)
- Q ensemble fini d'états
- $\delta : Q \times \Sigma \to Q$ (qu'on peut étendre à $\delta : Q \times \Sigma^* \to Q$)
- $s \in Q$ est l'état initial
- $F \subseteq Q$ est l'ensemble d'états finals

l'automate est complet quand δ est une fonction totale

4.4.3 Déterminisation

Soit $(\Sigma, Q, \Delta, D, F)$ un automate.
La déterminisation consiste à construire l'automate $(\Sigma, 2^Q, \delta, D, \{X \subseteq Q \mid X \cap F \neq \emptyset\})$ où δ est l'extention de $\Delta : 2^Q \times \Sigma \to 2^Q$ citée au 4.4.1.

4.4.4 Minimisation

Rappelez-vous que la notion d'automate minimal ne concerne que les automates déterministes : il n'y a pas d'automate minimal non déterministe.

Soit $A = (\Sigma, Q, \delta, D, F)$ un automate déterministe.

- $q \in Q$ et $q' \in Q$ inséparables par $w \in \Sigma^*$ si $\delta(q, w) \in F \Leftrightarrow \delta(q', w) \in F$
- $q \equiv_i q'$ si q et q' *inséparables* par tous les mots de longueur $\leq i$
  $\equiv_i$ est une relation d'équivalence
  remarque : si $\equiv_i$ est égale à $\equiv_{i+1}$ alors $\equiv_i$ est égale à $\equiv_j$ pour tout $j \geq i$
- $q \equiv q'$ si pour tout $w \in \Sigma^*$, q et q' sont inséparables par w (i.e. $\delta(q, w) \in F \Leftrightarrow \delta(q', w) \in F$)
  $\equiv$ est une relation d'équivalence
  remarque : si $\equiv_{i+1}$ égale à $\equiv_i$ alors $\equiv_i$ égale à $\equiv$

notations : on note $[q]_\equiv$ la classe d'équivalence de $q \in Q$, on note $\frac{Q}{\equiv}$ l'ensemble quotient de Q par $\equiv$ (l'ensemble des classes d'équivalences)

l'automate quotient de A (par $\equiv$) est

$$\min(A) = (\Sigma, \frac{Q}{\equiv}, \delta_{\min}, [s]_\equiv, \{[q]_\equiv \mid q \in F\})$$

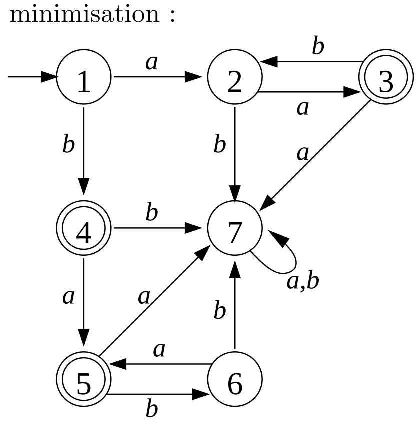
avec $\delta_{\min} : \frac{Q}{\equiv} \times \Sigma \to \frac{Q}{\equiv}, \forall q \in Q, \forall x \in \Sigma, \overline{\delta}([q]_\equiv, x) = [\delta(q, x)]_\equiv$

$\min(A)$ est l'automate (déterministe) minimal qui reconnaît $L(A)$.

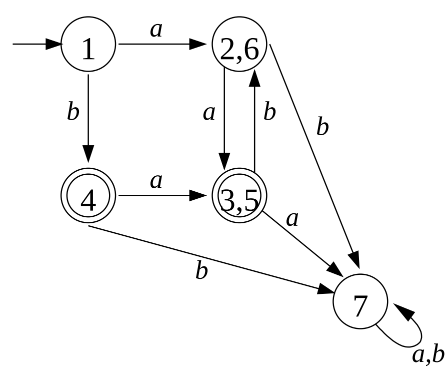
Comment calculer $\equiv$: on calcule $\equiv_0$, $\equiv_1$, $\equiv_2$, ... jusqu'à $\equiv_i$ et $\equiv_{i+1}$ identiques

1. pour $\equiv_0$, il y a deux classes d'équivalences : F et $Q \setminus F$
2. supposons qu'on a calculé les classes d'équivalences de $\equiv_i$, on regarde si les états d'une même classe d'équivalence par rapport à $\equiv_i$ sont séparables par un mot de longueur $i + 1$ (il faut essayer tous les mots de longueur $i + 1$), s'ils sont séparables alors ils sont alors dans 2 classes d'équivalences différentes par rapport à $\equiv_{i+1}$, sinon ils restent dans la même classe pour $\equiv_{i+1}$
3. quand les classes d'équivalences par rapport $\equiv_i$ et à $\equiv_{i+1}$ sont les mêmes on a fini

exemple de minimisation :



automate minimal



on construit le tableau pour calculer les $\equiv_i$ (les états finals sont entourés) :

	$\equiv_0$	$\equiv_1$		$\equiv_2$			
	ϵ	a	b	aa	ab	ba	bb
1	1	2	$\boxed{4}$	$\boxed{3}$	7	$\boxed{5}$	7
2	2	$\boxed{3}$	7	7	2	7	7
3	$\boxed{3}$	7	2	7	7	$\boxed{3}$	7
4	$\boxed{4}$	$\boxed{5}$	7	7	6	7	7
5	$\boxed{5}$	6	7	7	7	$\boxed{5}$	7
6	6	$\boxed{5}$	7	7	6	7	7
7	7	7	7	7	7	7	7
	$\{1,2,6,7\}$ $\{3,4,5\}$	$\{2,6\}$ $\{1\}$ $\{3,5\}$ $\{4\}$ $\{7\}$		$\{2,6\}$ $\{1\}$ $\{3,5\}$ $\{4\}$ $\{7\}$			

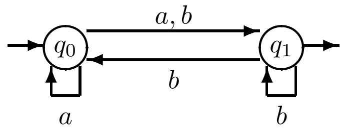
4.5 Langages résiduels

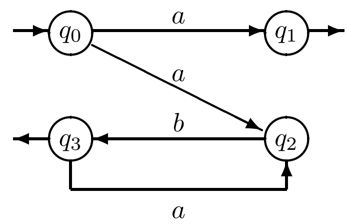
le langage résiduel de L par rapport à u, noté $u^{-1}L$, est $u^{-1}L = \{v \in \Sigma^* \mid uv \in L\}$
on a $v \in u^{-1}L \Leftrightarrow uv \in L$
pour un automate déterministe $A = (\Sigma, Q, \delta, s, F)$, les résiduels sont les $L(q)$, $q \in Q$

$$\boxed{s} \xrightarrow{u} \boxed{q} \xrightarrow{v} \boxed{\text{état final}}$$
$$(\delta(s,u) = q \text{ et } v \in L(q)) \Leftrightarrow uv \in L$$
$$\text{donc } L(q) = u^{-1}L(A)$$

Le nombre de résiduels de L est le nombre d'état de l'automate déterministe minimal qui reconnaît L
remarque : L régulier $\Leftrightarrow$ L a un nombre fini de résiduels

Exercice 4.1 Construire un automate fini déterministe correspondant à chaque automate ci-dessous :

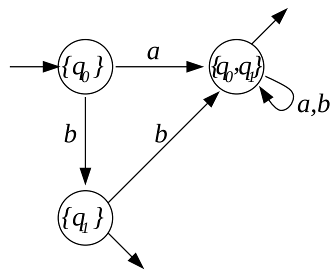




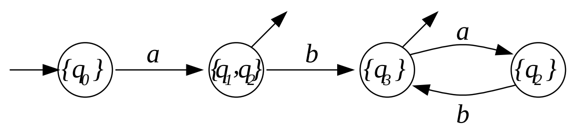
Calculez pour chaque automate une expression régulière pour le langage accepté.

Solution

1. premier automate :



   deuxième automate



2. premier automate :

$$\begin{cases} L(q_0) = aL(q_0) + aL(q_1) + bL(q_1) & (1) \\ L(q_1) = bL(q_1) + bL(q_0) + \epsilon & (2) \end{cases}$$

   de (2) on déduit $L(q_1) = b^*(bL(q_0) + \epsilon)$ on reporte dans (1) : $L(q_0) = aL(q_0) + (a+b)b^*(bL(q_0) + \epsilon)$
   i.e. $L(q_0) = (a + (a+b)b^*b)L(q_0) + (a+b)b^*$
   on en déduit $L(q_0) = (a + (a+b)b^*b)^*(a+b)b^*$

   deuxième automate

$$\begin{cases} L(q_0) = aL(q_1) + aL(q_2) & (1) \\ L(q_1) = \epsilon & (2) \\ L(q_2) = bL(q_3) & (3) \\ L(q_3) = aL(q_2) + \epsilon & (4) \end{cases}$$

   à partir de (3) et (4) on a $L(q_2) = b(aL(q_2) + \epsilon)$, i.e. $L(q_2) = (ba)^*b$
   on reporte dans (1) : $L(q_0) = a + a(ba)^*b$

Exercice 4.2 Soit $A = (\Sigma, Q, \Delta, q_0, F)$ un automate fini. On définit les ensembles suivants :

- $X_{s,t} = \{x \in \Sigma \mid (s,x,t) \in \Delta\}$, $s \in Q, t \in Q$;
- $W_{s,P,t} = \{u \in \Sigma^* \mid$ il existe un chemin reconnaissant u, menant de s à t et n'utilisant que des états de P comme états intermédiaires$\}$, $s \in Q$, $t \in Q$, $P \subseteq Q \setminus \{s,t\}$;
- $Z_{s,P,t} = \{u \in \Sigma^* \mid$ il existe un chemin reconnaissant u, menant de s à t et n'utilisant que des états de P comme états intermédiaires$\}$, $s \in Q$, $t \in Q$, $P \subseteq Q$.

1. Montrer que $X_{s,t}$ est régulier.
2. Soient $s \in Q$, $t \in Q$, $P \subseteq Q \setminus \{s,t\}$ et $q \in P$. On pose $P' = P \setminus \{q\}$. Montrez que $W_{s,P,t}$ est régulier. Exprimez $W_{s,P,t}$ en fonction de $W_{s,P',t}$, $W_{s,P',q}$, $W_{q,P',q}$, $W_{q,P',t}$.
3. Exprimez $Z_{s,P,t}$ en fonction de W_{s_i,P_i,t_i}, les s_i, P_i et t_i étant judicieusement choisis. En déduire que $Z_{s,P,t}$ est régulier.
4. En déduire que $\mathcal{L}(A)$ est un langage régulier.
5. Utilisez cette méthode pour trouver une expression régulière du langage reconnu par :

$$A = (\{x,y\}, \{1,2,3\}, \{(1,x,1), (1,y,2), (2,x,3), (2,y,1), (3,x,2), (3,y,3)\}, 1, \{3\})$$

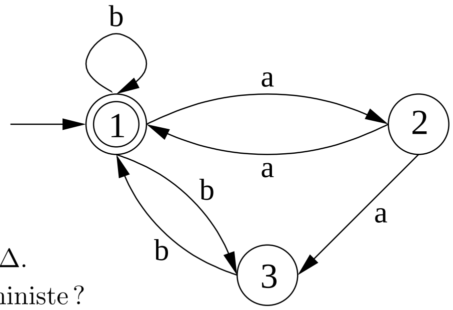
6. Comparez cette expression avec celle obtenue à l'aide du système régulier correspondant à A.

Solution

1. $X_{s,t} = \{x \in \Sigma \mid (s,x,t) \in \Delta\}$ est fini donc régulier (rappelez-vous que Σ est fini et on a $X_{s,t} \subseteq \Sigma$)
2. récurrence sur $|P|$
   - si $|P| = 0$, i.e. $P = \emptyset$, alors $W_{s,P,t} = X_{s,t}$ est régulier
   - supposons $W_{s,P,t}$ régulier pour tout $|P| = n$
     soit $|P| = n+1$, soit $q \in P$, soit $P' = P \setminus \{q\}$
     $W_{s,P,t} = W_{s,P',t} + W_{s,P',q}(W_{q,P',q})^* W_{q,P',t}$
     en effet, pour aller de s à t en utilisant des états de P :
     - soit le chemin ne passe pas par q : $W_{s,P',t}$
     - soit le chemin passe par q, alors on va de s à la première fois qu'on passe par q : $W_{s,P',q}$
       ensuite on repasse éventuellement plusieurs fois pas q : $(W_{q,P',q})^*$
       et on va du dernier passage par q à t : $W_{q,P',t}$

     $W_{s,P',t}$, $W_{s,P',q}$, $W_{q,P',q}$ et $W_{q,P',t}$ sont réguliers d'après l'hypothèse donc $W_{s,P,t}$ est régulier
   - pour tout s, P, t : $W_{s,P,t}$ est régulier
3. $Z_{s,P,t} = (W_{s,P\setminus\{s\},s})^* W_{s,P\setminus\{s,t\},t}(W_{t,P\setminus\{s,t\},t})^*$ donc régulier
   en effet, pour aller de s à t en utilisant des états de P :
   de s au dernier passage par s : $(W_{s,P\setminus\{s\},s})^*$
   du dernier passage par s au premier passage par t : $W_{s,P\setminus\{s,t\},t}$
   du premier passage par t à la fin : $(W_{t,P\setminus\{s,t\},t})^*$
4. $L(A) = \bigcup_{q \in F} Z_{q_0,Q,q}$ donc régulier
5. $L(A) = Z_{1,\{1,2,3\},3}$
   $Z_{1,\{1,2,3\},3} = (W_{1,\{2,3\},1})^* W_{1,\{2\},3}(W_{3,\{2\},3})^*$
   $W_{1,\{2,3\},1} = W_{1,\{3\},1} + W_{1,\{3\},2}(W_{2,\{3\},2})^* W_{2,\{3\},1}$
   $$\begin{aligned} W_{1,\{3\},1} &= W_{1,\emptyset,1} + W_{1,\emptyset,3}(W_{3,\emptyset,3})^* W_{3,\emptyset,1} \\ &= x + \emptyset y^* \emptyset \\ &= x \end{aligned}$$
   etc...
   et on obtient (si je ne me suis pas trompé) :
   $L(A) = (x + y(xy^*x)^*y)^* yx(xx+y)^*$
6. je vous laisse résoudre le système d'équations :
   $$\begin{cases} L(1) = xL(1) + yL(2) \\ L(2) = xL(3) + yL(1) \\ L(3) = xL(2) + yL(3) + \epsilon \end{cases}$$
   vous trouverez sans doute une expression différente, mais équivalente !

Exercice 4.3 On considère l'automate $A = (\{a, b\}, \{1, 2, 3\}, \Delta, \{1\}, \{1\})$ suivant :



1. Donnez la table décrivant Δ.
2. Cet automate est-il déterministe ?
3. Le mot *baabab* est-il accepté par l'automate A ?
4. Donnez l'automate fini déterministe minimal qui reconnaît le même langage que A.

Solution

1. valeur de Δ :

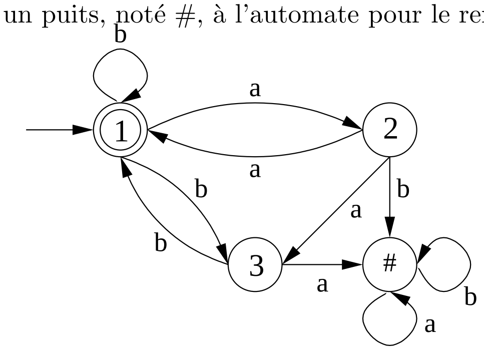
Δ	a	b
1	$\{2\}$	$\{1, 3\}$
2	$\{1, 3\}$	$\emptyset$
3	$\emptyset$	$\{1\}$

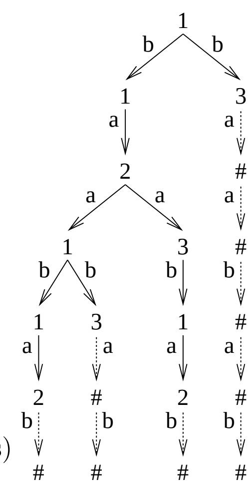
i.e. $\Delta = \{(1, a, 2), (1, b, 1), (1, b, 3), (2, a, 1), (2, a, 3), (3, b, 1)\}$

2. l'automate n'est pas déterministe car $\Delta(1, b) = \{1, 3\}$ par exemple, i.e. il y a 2 arcs étiquetés par b qui partent de 1.
3. *baabab* n'est pas accepté par l'automate :
   On peut voir Δ comme une fonction de $2^{\{1,2,3\}} \times \Sigma^* \to 2^{\{1,2,3\}}$, ce qui donne :

   $\Delta(\{1\}, baabab) = \Delta(\{1, 3\}, aabab) = \Delta(\{2\}, abab) = \Delta(\{1, 3\}, bab) = \Delta(\{1, 3\}, ab) = \Delta(\{2\}, b) = \emptyset$
   Comme $\Delta(\{1\}, baabab)$ ne contient aucun état final on peut conclure que *baabab* n'est pas accepté par l'automate.

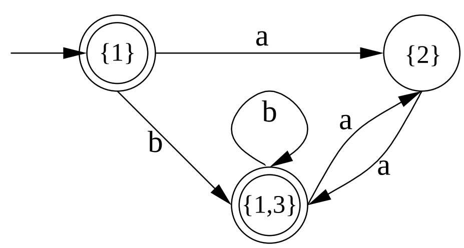
   Une autre solution consiste à construire l'arbre de dérivation.
   On peut ajouter un puits, noté #, à l'automate pour le rendre complet :





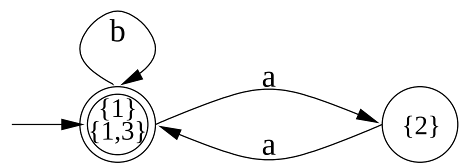
   On obtient alors l'arbre de dérivation ci-contre pour lequel aucune feuille ne correspond à un état final (toutes les dérivations terminent dans le puits)

4. automate déterministe :

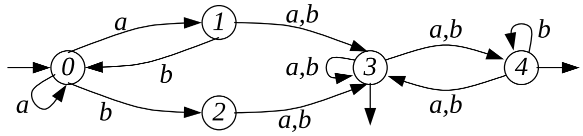


	ϵ	a	b
$\{1\}$	$\underline{\{1\}}$	$\{2\}$	$\underline{\{1, 3\}}$
$\{2\}$	$\underline{\{2\}}$	$--$	$\underline{--}$
$\{1, 3\}$	$\underline{\{1, 3\}}$	$\{2\}$	$\underline{\{1, 3\}}$

les états $\{1\}$ et $\{1, 3\}$ sont inséparables
on trouve donc l'automate minimal



Exercice 4.4 Déterminisez l'automate ci-dessous, puis calculez une expression régulière décrivant le langage reconnu.

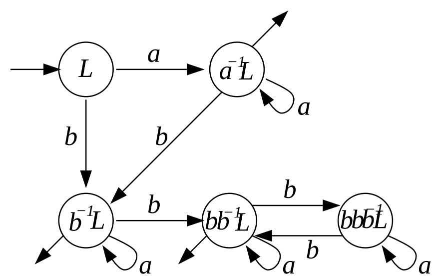


Solution Trop facile, je ne le corrige pas.

Exercice 4.5 On considère le langage $\mathcal{L}$ sur l'alphabet $\{a, b\}$ composé de tous les mots d'au moins un symbole qui contiennent exactement une occurrence du symbole b ou un nombre pair d'occurrences du symbole b. Calculez tous les langages résiduels de $\mathcal{L}$, puis en déduire l'automate minimal qui reconnaît $\mathcal{L}$.
Le langage résiduel (ou langage dérivé) de $\mathcal{L}$ par rapport à un mot u est $u^{-1}L = \{v \in \{a,b\}^* \mid uv \in \mathcal{L}\}$.

Solution les résiduels sont :
$\epsilon^{-1}\mathcal{L} = \mathcal{L} = a^*ba^* + a^+ + (a^*ba^*b)^+a^*$
$a^{-1}\mathcal{L} = \mathcal{L} \cup \{\epsilon\} = a^*ba^* + (a^*ba^*b)^*a^* \qquad (= (a^i)^{-1}\mathcal{L}, i > 0)$
$b^{-1}\mathcal{L} = a^* + (a^*ba^*b)^*a^*ba^* \qquad (= (a^iba^j)^{-1}\mathcal{L}, i, j \geq 0)$
$(bb)^{-1}\mathcal{L} = (a^*ba^*b)^*a^* \qquad (= ((a^{i_k}ba^{j_k}b)^k a^l)^{-1}\mathcal{L}, k > 0, l \geq 0, \forall k : i_k \geq 0, j_k \geq 0)$
$(bbb)^{-1}\mathcal{L} = (a^*ba^*b)^*a^*ba^* \qquad (= ((a^{i_k}ba^{j_k}ba^{h_k}b)^k a^l)^{-1}\mathcal{L}, k > 0, l \geq 0, \forall k : i_k \geq 0, j_k \geq 0, h_k \geq 0)$
on a donc 5 résiduels différents :
$\epsilon^{-1}\mathcal{L}$ le résiduel du mot vide
$a^{-1}\mathcal{L}$ le résiduel de tous les mots qui ont au moins un a et zéro b
$b^{-1}\mathcal{L}$ le résiduel de tous les mots qui ont exactement un b
$(bb)^{-1}\mathcal{L}$ le résisuel de tous les mots qui ont un nombre pair $\neq 0$ de b
$(bbb)^{-1}\mathcal{L}$ le résiduel de tous les mots qui ont un nombre impair $\neq 1$ de b

l'automate a donc 5 états : $\mathcal{L}$, $a^{-1}\mathcal{L}$, $b^{-1}\mathcal{L}$, $(bb)^{-1}\mathcal{L}$ et $(bbb)^{-1}\mathcal{L}$
l'état initial est $\mathcal{L}$
l'ensemble d'état final est $a^{-1}\mathcal{L}$, $b^{-1}\mathcal{L}$, $(bb)^{-1}\mathcal{L}$ car ils contiennent ϵ



Exercice 4.6 Considérons un langage $\mathcal{L}$ sur un alphabet Σ.

1. Indiquez différentes manières de montrer que le langage $\mathcal{L}$ est régulier.
2. Donnez une manière de montrer que $\mathcal{L}$ n'est pas régulier.
3. Que pensez-vous des différentes méthodes qui consisteraient à montrer que $\mathcal{L}$ n'est pas régulier
   (a) en montrant qu'il est l'union de deux langages qui ne sont pas réguliers ?
   (b) en montrant qu'il est l'union d'un langage régulier et d'un langage qui n'est pas régulier ?
   (c) en montrant qu'il est l'intersection de deux langages qui ne sont pas réguliers ?
   (d) en montrant qu'il est le complémentaire d'un langage qui n'est pas régulier ?

Solution

1. Indiquez différentes manières de montrer que le langage $\mathcal{L}$ est régulier.
   (voir les rappels au début)
   – $\mathcal{L}$ valeur d'une expression régulière
   – $\mathcal{L}$ reconnu par un automate fini
   – $\mathcal{L}$ engendré par une grammaire régulière
   – $\mathcal{L}$ solution d'un système d'équations régulières
   – $\mathcal{L}$ obtenu par union, intersection, concaténation, étoile, complémentaire de langages réguliers
   – $\mathcal{L}$ a un nombre fini de résiduels
   – ...
2. Donnez une manière de montrer que $\mathcal{L}$ n'est pas régulier.
   (voir les rappels 4.2 au début)
   On utilise le lemme d'itération et on montre qu'il n'existe pas de N tel que..., ce qui implique que le langage n'est pas régulier (contraposée du lemme).
   En général, pour montrer qu'il n'existe pas de N, on suppose qu'il existe et on obtient une contradiction.
3. Que pensez-vous des différentes méthodes qui consisteraient à montrer que $\mathcal{L}$ n'est pas régulier
   (a) en montrant qu'il est l'union de deux langages qui ne sont pas réguliers ?
   l'union de deux langages qui ne sont pas réguliers peut être un langage régulier !
   par exemple, si L n'est pas régulier alors $\overline{L}$ n'est pas régulier, pourtant $L \cup \overline{L} = \Sigma^*$ est régulier
   (b) en montrant qu'il est l'union d'un langage régulier et d'un langage qui n'est pas régulier ?
   encore mauvais, prenons un L qui n'est pas régulier, Σ^* est régulier et $L \cup \Sigma^* = \Sigma^*$ qui est régulier
   (c) en montrant qu'il est l'intersection de deux langages qui ne sont pas réguliers ?
   il est facile de donner deux langages non réguliers disjoints, dans ce cas l'intersection des deux est $\emptyset$ et $\emptyset$ est régulier
   (d) en montrant qu'il est le complémentaire d'un langage qui n'est pas régulier ?
   Cette fois ça marche, en effet le complémentaire d'un langage régulier est régulier, donc le complémentaire d'un langage non régulier ne peut pas être régulier (car le complémentaire du complémentaire est le langage lui-même)

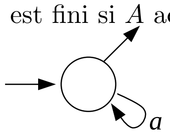
Exercice 4.7 Soit A un automate déterministe de k états.

Solution
Dans cet exercice il faut penser au lemme d'itération (voir les rappels 4.2) en pensant que pour le N du lemme on peut prendre le nombre d'états d'un automate (évenutellement minimal).

– $L(A) \neq \emptyset$ si et seulement si A accepte un mot de longueur inférieure à k.
  vrai
  $\Leftarrow$ évident : si A accepte un mot alors A n'est pas vide !
  $\Rightarrow$ si A n'est pas vide alors A accepte au moins un mot w, si $|w| < k$ c'est gagné
  sinon, on a $|w| \geq k$, considérons le chemin qui reconnaît w, il va de l'état initial à un état final
  ce chemin a obligatoirement des circuits car il est de longueur $\geq k$, si on supprime tous les circuits, on obtient un chemin de longueur $< k$ qui va de l'état initial à un état final, donc A accepte un mot de longueur $< k$

– $L(A)$ est infini si et seulement si A accepte un mot de longueur n, où $k \leq n < 2k$.
  vrai
  on prend $N = k$ dans le lemme d'itération...
  $\Leftarrow$ soit $m \in L(A)$, $|m| \geq k$ donc il existe u, v, w, $m = uvw$ avec $|v| > 0$ et pour tout $i \in \mathbb{N}$, $uv^iw \in L(A)$
  donc $L(A)$ est infini
  $\Rightarrow$ soit m le plus petit mot de $L(A)$ de longueur $\geq k$
  on suppose $|m| \geq 2k$, d'après le lemme, il existe u, v, w, $m = uvw$, $|v| \leq k$, $uv^0w \in L(A)$
  $|uw| < |uvw|$, comme m est le plus petit de longueur $\geq k$, $|uw| < k$
  et comme $|v| \leq k$, $|uvw| < k + k$, ce qui contredit l'hypothèse $|uvw| \geq 2k$
  donc A accepte un mot de longueur n, où $k \leq n < 2k$

– $L(A)$ est fini si A accepte un mot de longueur strictement inférieure à k.
  faux

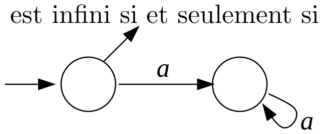


accepte ϵ, $|\epsilon| < 1$ et le langage reconnu est infini

– $L(A)$ est fini $\Leftrightarrow$ il existe $n \in \mathbb{N}$ tel que A n'accepte que des mots de longueur inférieure à n.
  vrai
  $\Leftarrow$ l'ensemble des mots de longueur $< n$ est fini
  $\Rightarrow$ si $L(A)$ est fini, on prend pour n un entier $>$ à la longueur du plus grand mot de $L(A)$

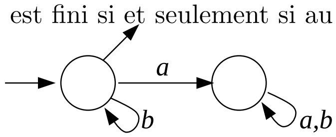
– Il est possible que A n'accepte aucun mot de longueur inférieure à k.
  vrai (dans ce cas $L(A) = \emptyset$)

– $L(A)$ est infini si et seulement si A contient un circuit.
  faux



$L(A) = \{\epsilon\}$

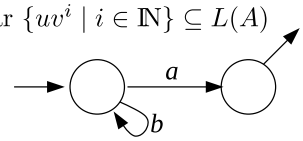
– $L(A)$ est fini si et seulement si au moins un circuit de A passent par la "poubelle".
  faux



$L(A) = b^*$

– Si A contient un circuit alors $L(A)$ peut ne pas être un langage régulier.
  faux
  vous avez vu que : L régulier $\Leftrightarrow$ L reconnu par un automate fini

– $L(A)$ est infini si et seulement si il existe au moins un circuit qui passe par un état d'acceptation et tout état de A est accessible.
  faux
  $\Leftarrow$ est vraie : supposons un circuit qui passe par un état d'acceptation q
  (état d'acceptation = état final)
  supposons que ce circuit correspond au mot u
  q est accessible, donc il existe un chemin de l'état initial à q
  supposons que ce chemin correspond au mot v
  $L(A)$ est infini car $\{uv^i \mid i \in \mathbb{N}\} \subseteq L(A)$
  $\Rightarrow$ est fausse :

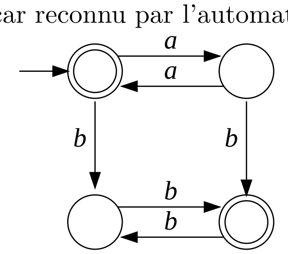


$L(A) = b^*a$

Exercice 4.8 Déterminez parmi les langages suivants ceux qui sont réguliers : $L_1 = \{a^{(i^2)} \mid i \geq 0\}$, $L_2 = \{a^n b^q \mid n + q \text{ pair}\}$, $L_3 = \{a^p \mid p \text{ premier}\}$, $L_4 = \{a^n b^n \mid n \in \mathbb{N}\}$, $L_5 = \{(ab)^n \mid n \in \mathbb{N}\}$, $L_6 = \{a^n b^{2n} \mid n \in \mathbb{N}\}$, $L_7 = \{a^n b^p c^{n+p} \mid n, p \in \mathbb{N}\}$, $L_8 = \{a^{n!} \mid n \in \mathbb{N}\}$, $L_9 = \{ww \mid w \in \{a,b\}^*\}$, $L_{10} = \{w \mid w = w^R, w \in \{a,b\}^*\}$ (voir exercice 1.10), $L_{11} = \{wuw^R \mid w, u \in \{a,b\}^*\}$.

Solution Je ne les fais pas tous...
soit on utilise le lemme d'itération pour montrer que le langage n'est pas régulier, soit on utilise une des méthodes proposées au 1. de l'exercice 4.6 pour montrer qu'il est régulier

- $L_1 = \{a^{(i^2)} \mid i \geq 0\}$
  on prend le N du lemme
  soit $m = a^{N^2}$, d'après le lemme, il existe u, v, w, $m = uvw$...
  on prend $i = 2$ dans le lemme, normalement $uv^2w \in L_1$
  on utilise $0 < |v| \leq N$
  $$\begin{array}{ccccccc} N^2 = |uvw| & < & |uv^2w| & \leq & N^2 + N = N \times (N+1) & < & (N+1)^2 \\ & \uparrow & & \uparrow & & & \\ & \text{car } |v| > 0 & & \text{car } |v| \leq N & & & \end{array}$$
  donc $N^2 < |uv^2w| < (N+1)^2$, $|uv^2w|$ n'est pas un carré parfait, $uv^2w \notin L_1$
  L_1 n'est pas régulier
- $L_2 = \{a^n b^q \mid n + q \text{ pair}\}$ est régulier, car reconnu par l'automate



- $L_3 = \{a^p \mid p \text{ premier}\}$
  supposons que L_3 soit régulier, donc d'après le lemme
  $\exists N$ t.q. $\forall m \in L_3$, $|m| \geq N$, $\exists uvw = m$, $|v| > 0$, $|uv| \leq N$, $\forall i \geq 0\ uv^iw \in L_3$
  prenons un q premier, $q \geq N$
  $a^q \in L_3$ et $|a^q| \geq N$, donc $\exists u, v, w$, $uvw = a^q$
  donc $\exists j, k, l$, $u = a^l$, $v = a^j$, $w = a^k$ avec $l + j + k = q$ premier
  $\forall i \geq 0\ uv^iw \in L_3$, donc $uw = a^{l+k} \in L_3$, d'où $l + k$ premier
  et $uv^{l+k}w = a^{(l+k)(j+1)} \in L_3$, d'où $(l+k)(j+1)$ premier
  or $(l+k)(j+1)$ ne peut pas être premier puisque divisble par $l + k$ qui est premier et par $j + 1$ qui est > 1 (car $j > 0$)
  donc L_3 n'est pas régulier
- $L_4 = \{a^n b^n \mid n \in \mathbb{N}\}$
  comme à chaque fois, on suppose que le N du lemme existe
  on prend le mot $a^N b^N$, $\exists u, v, w$, $uvw = a^N b^N$, donc $\exists j, k$, $k > 0$, $u = a^j$, $v = a^k$, $w = a^{N-(j+k)} b^N$
  $uv^0w = a^{N-k} b^N \notin L_4$, donc L_4 n'est pas régulier
- $L_5 = \{(ab)^n \mid n \in \mathbb{N}\}$
  régulier : $L_5 = (ab)^*$
- $L_6 = \{a^n b^{2n} \mid n \in \mathbb{N}\}$
  on montre comme pour L_4 que L_6 n'est pas régulier
- $L_7 = \{a^n b^p c^{n+p} \mid n, p \in \mathbb{N}\}$
  par régulier, même genre que L_4 aussi
- $L_8 = \{a^{n!} \mid n \in \mathbb{N}\}$
  pas régulier, prendre $m = a^{N!}$...
- $L_9 = \{ww \mid w \in \{a,b\}^*\}$
  pas régulier, on prend $uvw = a^N b a^N b$ puis uv^0w...
- $L_{10} = \{w \mid w = w^R, w \in \{a,b\}^*\}$
  pas régulier, on prend $uvw = a^N b a^N$ puis uv^0w...
- $L_{11} = \{wuw^R \mid w, u \in \{a,b\}^*\}$
  pas régulier, on prend $uvw = a^N b a^N$ puis uv^0w...

Exercice 4.9 On a vu que $L = \{a^n b^n \mid n \in \mathbb{N}\}$ (voir exercice 4.8) n'est pas un langage régulier.

1. Est-ce que $L_m = \{a^n b^n \mid n \leq m\}$ est régulier ($m \in \mathbb{N}$ est fixé) ?
2. Que pensez-vous de l'égalité $L = \bigcup_{m \in \mathbb{N}} L_m$?
3. On a $L = \{a^n b^p \mid n = p\}$. Est-ce que $L' = \{a^n b^p \mid n \neq p\}$ est régulier ?

Solution

1. Est-ce que $L_m = \{a^n b^n \mid n \leq m\}$ est régulier ($m \in \mathbb{N}$ est fixé) ?
   L_m est régulier car L_m est fini
2. Que pensez-vous de l'égalité $L = \bigcup_{m \in \mathbb{N}} L_m$?
   elle est vraie
   – $\subseteq$ soit $u \in L$, donc $\exists n \in \mathbb{N}$ t.q. $u = a^n b^n$ donc $u \in L_n \subseteq \bigcup_{m \in \mathbb{N}} L_m$
   – $\supseteq$ soit $u \in \bigcup_{m \in \mathbb{N}} L_m$, donc $\exists n \in \mathbb{N}$ t.q. $u \in L_n$, i.e. $\exists k \in \mathbb{N}$, $k \leq n$ t.q. $u = a^k b^k$, donc $u \in L$
   On sait que l'union de 2 langages réguliers forme un langage régulier. Ce résultat s'étend à une réunion finie de langages réguliers mais pas à une réunion infinie (si c'était le cas, tout langage serait régulier car tout langage est la réunion des singletons formés par chaque mot du langage).
3. On a $L = \{a^n b^p \mid n = p\}$. Est-ce que $L' = \{a^n b^p \mid n \neq p\}$ est régulier ?
   L' n'est pas régulier
   en effet, $L'' = a^* b^*$ est régulier
   si L' est régulier alors $\overline{L'}$ est régulier et $\overline{L'} \cap L''$ est régulier
   or $\overline{L'} \cap L'' = L$ qui n'est pas régulier, <u>contradiction</u>