Bases formelles du TAL
Partiel

Pierre-Léo Bégay

2 mars 2018

Dans tout ce partiel, on utilise l'alphabet $\Sigma = \{a, b\}$ et on utilise la définition d'automate fini non-déterministe qui **ne** permet **pas** les ϵ-transitions

Les bonus sont moins rentables que les autres questions, gardez-les pour la fin

Exercice 1 [5 points]

Soit les expressions rationnelles suivantes :

$e_1 = a^*$

$e_2 = a^*(ba^+)^*$

$e_3 = \Sigma^*(aa + bb)a\Sigma^*$

$e_4 = b^*a(b^*ab^*a)^*b^*$

Question 1 [0,25 / 0,75 / 0,5 / 0,75] Soient L_1, L_2, L_3 et L_4 les langages $[\![e_1]\!]$, $[\![e_2]\!]$, $[\![e_3]\!]$ et $[\![e_4]\!]$, respectivement. Décrivez-les en langue naturelle, éventuellement en vous aidant d'exemples de mots appartenant ou non aux différents langages.

correction

- L_1 = Les mots composés uniquement de a : $\epsilon, a, aa, aaa, ...$
- L_2 = Les mots 1) où tout b est suivi d'un a. $\epsilon, ba, aba, aababaaabaaa, aaabaabaaba...$
- L_3 = Les mots qui contiennent aaa ou bba (ou les deux).
- L_4 = Les mots qui contiennent un nombre impair de a.

Question 2 [0,5 / 0,75 / 0,5 / 1] Décrivez les langages suivant :

$L_1 \cap L_3$

$L_1 \cap L_4$

$L_4 \cap [\![b^*]\!]$

$\overline{L_2}$

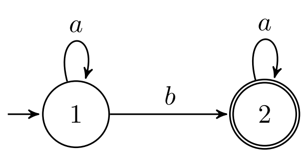
Correction

- Puisque L_1 est l'ensemble qui mots avec uniquement des a, l'option bba pour être dans L_3 est donc *annulée*. C'est le langage des mots contenant uniquement des a et au moins aaa, cad $aaaa^*$ $(aaa, aaaa, aaaaa, aaaaaa, ...)$
- Les mots qui contiennent uniquement des a, et en nombre impair. $a(aa)^*$
- Les mots avec un nombre impair de a (donc au moins un), et uniquement des b. Incompatibilité des deux propriétés, c'est donc $\emptyset$.
- L'ensemble des mots qui contiennent un b qui n'est pas suivi par un a (donc qui contiennent deux b à la suite ou finissent par un b).

Exercice 2 [3 points]

Soit $L = [\![a^*ba^*]\!]$

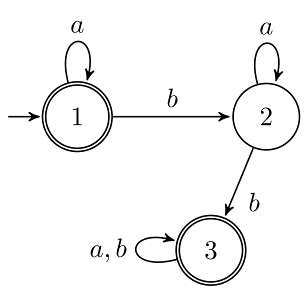
Question 1 [1] Donnez un automate fini qui reconnaît L



Correction

Question 2 [1] Donnez un automate fini qui reconnaît $\overline{L}$

Correction En utilisant la complémentation d'automate fini (bien penser à compléter l'automate avant !)



Question 3 [1] Donnez une expression régulière qui décrit $\overline{L}$

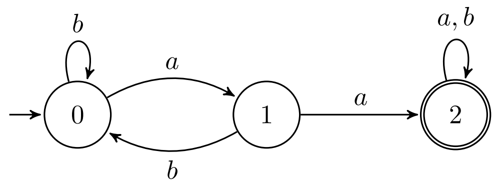
Correction L c'est le langage des mots qui contiennent exactement un b. Pour ne pas être dans le langage, il faut donc soit ne pas en contenir du tout, soit au moins deux[1] $\Rightarrow a^* + \Sigma^*b\Sigma^*b\Sigma^*$

[1] ce qu'on retrouve dans l'automate de la question précédente !

Exercice 3 [10 points]

Soit $L_1 = \{u \in \Sigma^* | \ aa \text{ est un facteur de } u\}$.

Question 1 [1] Donnez un automate fini déterministe complet reconnaissant L_1



Correction

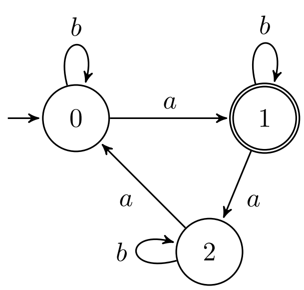
Question 2 [1] Donnez une expression rationnelle décrivant L_1

Correction $\Sigma^* aa \Sigma^*$

Soit $L_2 = \{u \in \Sigma^* | \ |u|_a \equiv 1 \ [3]\}$[2].

Question 3 [1] Donnez un automate fini déterministe complet reconnaissant L_2

Correction En s'inspirant des exemples vus en cours sur la parité du nombre de a ou de b :



Notez que les noms des états correspondent au nombre de a modulo 3 dans le mot qui y amène.

Question 4 [1] Donnez une expression rationnelle décrivant L_2

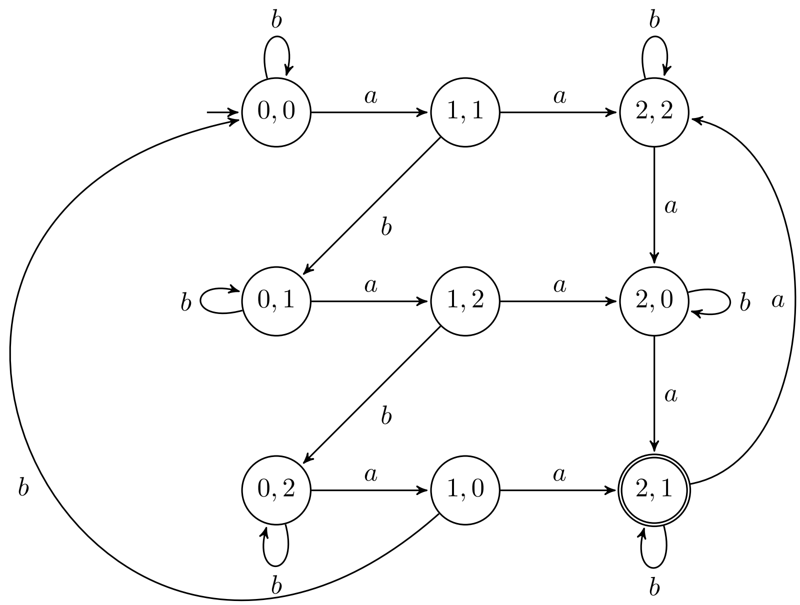
Correction $b^*a(b^*ab^*ab^*a)^*b^*$

Soit $L_3 = L_1 \cap L_2$

Question 5 [1,5] Donnez un automate déterministe complet reconnaissant L_3

[2] On peut également définir L_2 comme $\{u \in \Sigma^* | \ \exists k. \ |u|_a = 3k + 1\}$, ou encore l'ensemble des mots dont le nombre de a est 1, 4, 7, 10, 13, 16, etc...

En utilisant un produit d'automates :



Question 6 [2 + 1] Minimisez l'automate obtenu. **Bonus :** Expliquez 'moralement' ce résultat

Correction En appliquant l'algorithme, on obtient le même automate (modulo le nom des états), ce qui signifie qu'il était en fait déjà minimal. On peut expliquer ça par le fait que les deux langages sont *orthogonaux*, dans le sens où être ou non dans l'un n'influe en rien le fait d'être dans l'autre. Il faut donc garder la trace **précise** de l'évaluation de chaque mot par chacun des deux automates.

Comparez cette situation au produit d'automates reconnaissent des langages qui se croisent, comme par exemple les mots avec un nombre pair de a d'un côté, et le langage des mots avec un nombre de a multiple de 4 de l'autre. En faisant le produit, on va obtenir un automate à 8 états. Or, le premier langage est inclus dans le deuxième (c'est-à-dire que si un mot a un nombre de a multiple de 4, alors forcément il a un nombre pair de a), on pourrait donc reconnaître le langage avec uniquement le second automate, qui a 4 états. Le fait de 'suivre' le parcours d'un mot dans le premier automate est superflu, d'où le fait que le produit ne soit pas minimal[3]

Question 7 (bonus) [1] Donnez une expression rationnelle décrivant L_3

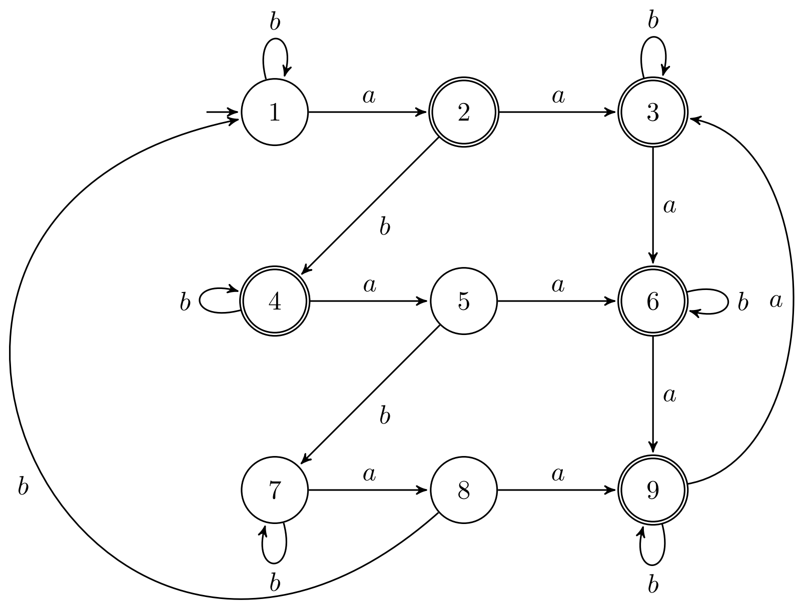
[3] C'est un exemple un peu extrême, mais on retrouve le même phénomène avec des langages tels qu'on n'a pas l'un inclus dans l'autre. A vue de nez, je dirais que ça marche aussi avec l'intersection de $\Sigma^* aba \Sigma^*$ et $\Sigma^* bab \Sigma^*$ par exemple

Correction Soit $e_1 = (ab^*ab^*ab^*)^*$. Alors, L_3 est décrit par $(e_1ab^*ab^*aab^*e_1) + (e_1ab^*aab^*ab^*e_1) + (e_1aab^*ab^*ab^*e_1)$. Dans les trois cas on a bien 1 a modulo 3, et la disjonction (les +) permet de placer le aa après 0, 1 ou 2 a modulo 3.

Soit $L_4 = L_1 \cup L_2$

Question 8 [0,5] Donnez un automate déterministe complet reconnaissant L_4.

Correction On reprend l'automate précédent en changeant les états terminaux (et on en profite pour changer les noms d'états)



Question 9 [2] Minimisez l'automate obtenu.

Correction En appliquant l'algorithme :

C_f 2 vs 3 vs 4 vs 6 vs 9 : $9 \not\equiv 4$ (par a), $6 \not\equiv 4$ (par a), $2 \not\equiv 4$ (par a), $9 \not\equiv 3$ (par a). On sépare donc la classe en $C_{f_1} = \{4\}$ d'un côté, et $C_{f_2} = \{2, 3, 6, 9\}$ de l'autre.

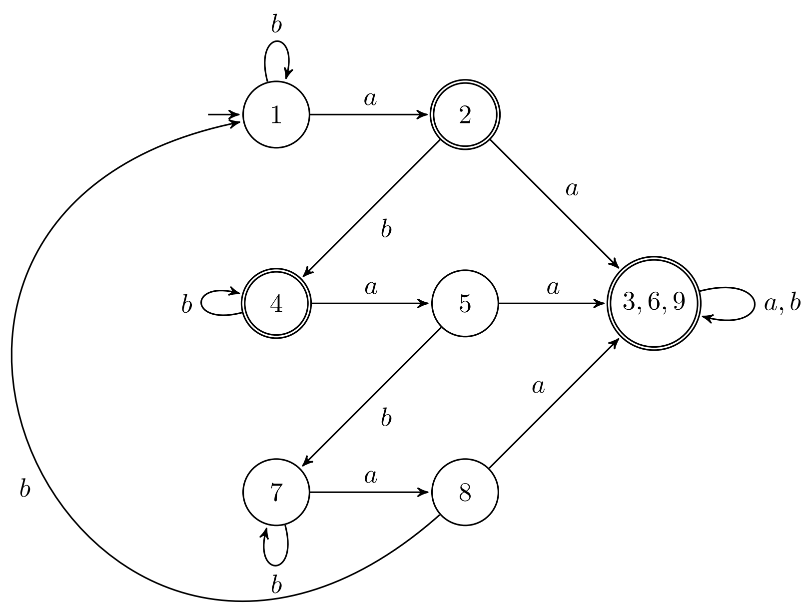
C_a 1 vs 5 vs 7 vs 8 : $8 \not\equiv 7$ (par a), $1 \not\equiv 7$ (par a), $5 \not\equiv 7$ (par a). On sépare donc la classe en $C_{a_1} = \{7\}$ d'un côté, et $C_{a_2} = \{1, 5, 8\}$ de l'autre.

C_{f_2} 2 vs 3 vs 6 vs 9 : $2 \not\equiv 9$ (par b (puisque 4 a été séparé des autres terminaux)), $2 \not\equiv 6$ (par b), $2 \not\equiv 3$ (par b). On sépare donc la classe en $C_{f_{2_1}} = \{2\}$ d'un côté, et $C_{f_{2_2}} = \{3, 6, 9\}$ de l'autre.

C_{a_2} 1 vs 5 vs 8 : $1 \not\equiv 8$ (par a), $5 \not\equiv 8$ (par b), $1 \not\equiv 5$ (par a). On sépare donc la classe en $\{1\}, \{5\}, \{8\}$.

La seule classe non-atomique est $C_{f_{2_2}}$, qu'on n'arrivera pas à séparer.

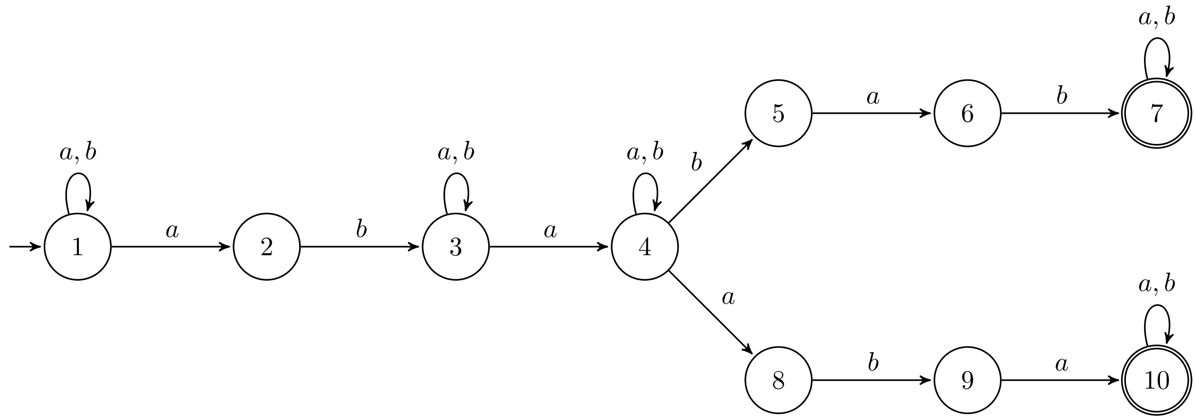
Les seuls états équivalents sont donc 3, 6 et 9[4]. On obtient alors l'automate minimal suivant :



Exercice 4 [2 points]

Déterminisez l'automate suivant :

[4]Notez que c'est un 'puit d'acceptation' qui correspond au fait d'avoir lu un *aa*, ce qui ne peut pas être *annulé*



Correction La vie est trop courte pour faire cet automate en LaTeX. Au tableau dans un cours.