Automates

Corrigé partiel de la feuille de travaux dirigés n^{o}1

1.
c) On note $L_A(i,j)$ ou simplement $L(i,j)$, le langage reconnu par l'automate A en prenant l'état i comme état initial et l'état j comme état terminal. On a alors à trouver le langage $L(1,1)$ et le langage $L(1,6)$.
Clairement $L(1,1) = \{a,b\}^*$.
Pour $L(1,6)$, on a :
$L(1,6) = \{a,b\}^* c L(2,2) c \{a,b\}^*$.
Il reste à trouver $L(2,2)$, pour cela on montre par induction sur la longueur des mots (en traitant simultanément les 4 ensembles suivants) que **(P)** :
$L(2,2)$ = ensemble des mots sur $\{a,b\}$ dont le nombre de a est pair et le nombre de b est pair
$L(2,3)$ = ensemble des mots sur $\{a,b\}$ dont le nombre de a est impair et le nombre de b est pair
$L(2,5)$ = ensemble des mots sur $\{a,b\}$ dont le nombre de a est pair et le nombre de b est impair
$L(2,4)$ = ensemble des mots sur $\{a,b\}$ dont le nombre de a est impair et le nombre de b est impair.
En effet, il est immédiat que si **(P)** est vraie pour tous les mots de longueur inférieure strictement à n, alors **(P)** devient vraie pour tous les mots de longueur inférieure ou égale à n. Et ceci pour tout $n \geq 1$. Par ailleurs on vérifie que **(P)** est vraie pour les mots de longueur 0 ou 1.
En fait, comme les mots de $L(2,2)$ sont tous de longueur paire, on peut considérer que les lettres "arrivent" par couples. Ainsi on passe d'un automate de quatre états à un automate autrement plus simple de deux états uniquement. De plus, le langage reconnu par celui-ci se déduit facilement :

$$L(2,2) = ((aa+bb) + ((ab+ba)(aa+bb)^\star(ab+ba))^\star$$

2.
b) Un mot quelconque a soit un nombre pair (état 0), soit un nombre impair de c (état 1). Ainsi on peut prendre un automate ayant ces 2 états, l'état 0 est l'état initial (en effet le mot vide contient un nombre pair (0) de c) et l'état 1 sera état terminal, la lecture de la lettre c fait changer d'état, la lecture de toute autre lettre boucle sur l'état où l'on est.
c) A chaque lecture d'une lettre, on regarde où on en est dans la lecture de $baba$:
ε : on a rien lu de $baba$; b : on a lu la lettre b ; ba : on a lu ba ; bab : on a lu bab ; $baba$: on a lu $baba$ entier.
On a ainsi naturellement 5 états, l'état ε est l'état initial et l'état $baba$ est état terminal. Les transitions se déduisent "immédiatement" de l'interprétation de ces états. Par exemple, de l'état ε, une lecture de a ou c maintient dans l'état ε mais une lecture de b fait passer dans l'état b ; de l'état ba, une lecture de a ou c fait passer dans l'état ε, une lecture de b fait passer dans l'état bab ; de l'état $baba$, une lecture de a ou c fait passer dans l'état ε, une lecture de b fait passer dans l'état bab.
d) On peut prendre les mêmes 5 états que dans la question précédente avec les mêmes états distingués (initial, terminal). Les transitions seront les mêmes **SAUF** partant de l'état $baba$ où le fait d'avoir "atteint" le mot $baba$ assure qu'il est facteur du mot lu et donc que maintenant quelque soient les lettres lues on boucle sur cet état "gagnant".
e) et f) On peut reprendre l'automate de la question précédente en prenant comme états terminaux, les états qui ne l'étaient pas dans l'automate **d)**. Attention, cette méthode pour trouver le complémentaire d'un langage ne fonctionne que si l'automate qui reconnaît le langage à complémenter est complet.

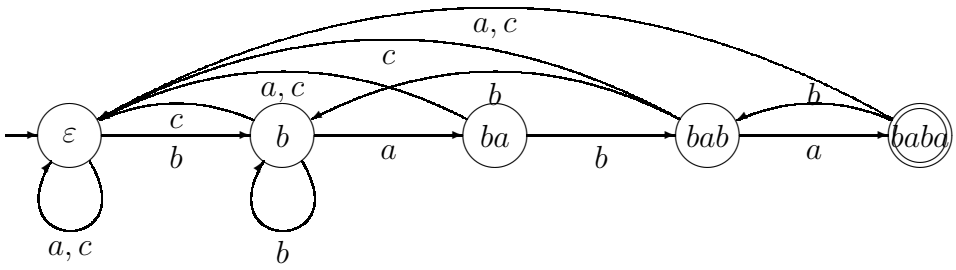

FIGURE 1 – Un automate qui répond à **2.c)**

3.
Comme indiqué dans l'énoncé, on considère que l'automate lit alternativement une lettre de chaque mot (en commençant par le premier !). On peut alors "naturellement" construire un automate répondant à la question et ayant 7 états :
0. état $(\varepsilon, \varepsilon)$: on a lu autant de lettres dans chacun des 2 mots et les préfixes ainsi lus sont égaux
1. état (a, ε) : on a lu une lettre de plus dans le premier mot, cette lettre est un a et hormis cette dernière lettre le préfixe lu dans le premier mot et le préfixe lu dans le deuxième sont égaux
2. état (a, b) : on a lu autant de lettres dans chacun des 2 mots, la dernière lettre du premier mot est a alors que la dernière lettre du deuxième est b, et hormis cette dernière lettre le préfixe lu dans le premier mot et le préfixe lu dans le deuxième sont égaux (on a donc alors : $u < v$)
3. état (b, ε) : on a lu une lettre de plus dans le premier mot, cette lettre est un b et hormis cette dernière lettre le préfixe lu dans le premier mot et le préfixe lu dans le deuxième sont égaux
4. état (b, a) : on a lu autant de lettres dans chacun des 2 mots, la dernière lettre du premier mot est b alors que la dernière lettre du deuxième est a, et hormis cette dernière lettre le préfixe lu dans le premier mot et le préfixe lu dans le deuxième sont égaux (on a donc ainsi : $u > v$)
5. état $(\#, \varepsilon)$: on a lu une lettre de plus dans le premier mot, cette lettre est un $\#$ donc le premier mot est fini et hormis ce marqueur de fin, le préfixe lu dans le premier mot (i.e. le mot entier !) et le préfixe lu dans le deuxième sont égaux (ainsi le premier mot est un préfixe du deuxième, et donc $u < v$)
6. état $(a, \#)$ **ou** $(b, \#)$: on a lu autant de lettres dans chacun des 2 mots, la dernière lettre du premier mot est a ou b alors que la dernière lettre du deuxième est $\#$, donc le deuxième mot est fini alors que le premier ne l'est pas (ainsi le deuxième mot est un préfixe du premier, et donc $u > v$).
Par construction, l'état 0 est état initial et les états 0, 2 et 5 sont états terminaux. Les transitions se déduisent immédiatement à partir de l'interprétation des états. Par exemple, de l'état 0, en lisant un a on passe à l'état 1 et de l'état 1 en lisant un a on revient à l'état 0, ... Si une lecture se termine dans les états 4 ou 6, le deuxième mot est plus petit (dans l'ordre lexicographique) que le premier. Si l'entrée est conforme à l'hypothèse (le symbole $\#$ est marqueur de fin de mot), aucune lecture ne termine dans les états 1 ou 3.

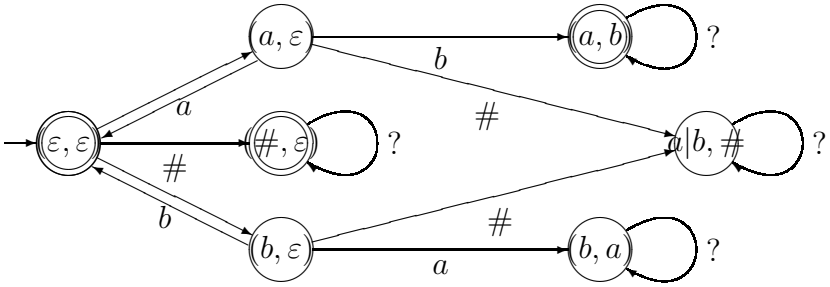

FIGURE 2 – Un automate qui répond à **3**.

4. a) On peut commencer par un automate (non complet) des mots commençant et finissant par a. On a alors "naturellement" 3 états :
0. état ε : rien n'est lu
1a. état (a, a) : la première lettre lue et la dernière lettre lue sont a
2a. état $(a, non\ a)$: la première lettre lue est a et la dernière lettre lue est b ou c
L'état 0 est état initial et l'état 1a est terminal. Les transitions se déduisent immédiatement de la "valeur" des 3 états. Par exemple de l'état ε en lisant un a on passe dans l'état (a, a) et de cet état ε il n'y a pas de transitions avec b ou c ; de l'état $(a, non\ a)$ en lisant un a on passe dans l'état (a, a) et en lisant b ou c on reste dans l'état $(a, non\ a)$.
Pour répondre à l'énoncé **5.a)**, il suffit de faire de même pour les lettres b et c. Ce qui donne un automate **déterministe** ayant 7 états.

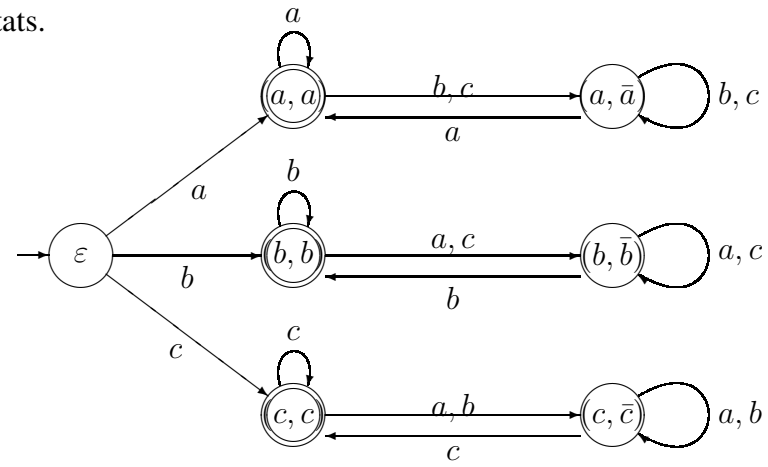

FIGURE 3 – Un automate qui répond à **4.a)**.

b) On s'intéresse aux 2 dernières lettres des mots, on va donc considérer toutes les possibilités (chacune correspondant à un état) pour ces 2 dernières lettres :
0. état ε : pour le mot vide
1a. état a : pour le mot a (idem avec b et c)
2aa. état aa : pour les mots se terminant par aa
2ab. état ab : pour les mots se terminant par ab
. . .
Soit en tout $1 + 3 + 9$ états, l'état 0 est initial ; les 6 états **2xy** (avec **x** et **y** différents) sont états "gagnants". Les transitions se déduisent immédiatement, par exemple de l'état **2ab** en lisant un c on passe dans l'état **2bc**.
Remarque : l'automate ainsi construit n'est pas minimal.
L'automate minimal est :

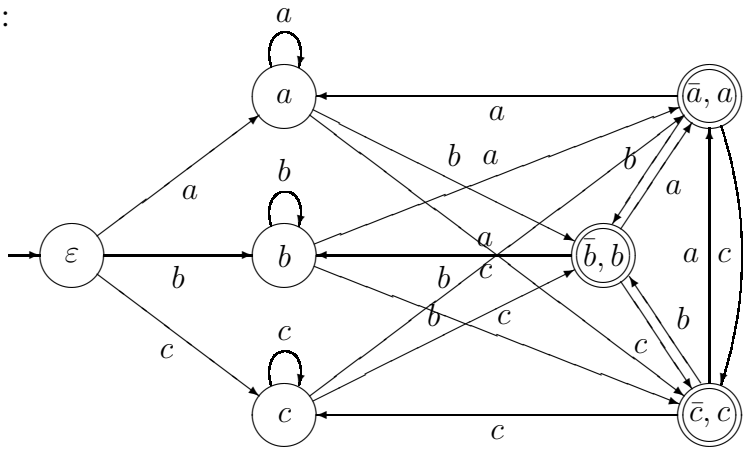

FIGURE 4 – Un automate qui répond à **4.b)**.

5. On obtient l'automate déterministe suivant :

δ	a	b
$\rightarrow 0,1,7$	$E(2,8) = 1,2,3,4,6,7,8$	$E(\emptyset) = \emptyset$
$\leftarrow 1,2,3,4,6,7,8$	$E(2,8) = 1,2,3,4,6,7,8$	$E(5) = 1,4,5,6,7$
$1,4,5,6,7$	$E(2,8) = 1,2,3,4,6,7,8$	$E(5) = 1,4,5,6,7$

Après renumérotation on a :

δ	a	b
$\rightarrow 0$	1	$-$
$\leftarrow 1$	1	2
2	1	2

On le complète pour obtenir :

δ	a	b
$\rightarrow 0$	1	P
$\leftarrow 1$	1	2
2	1	2
P	P	P