TD 02

Configuration d'un automate

Définition : On appelle *configuration d'un automate* en fonctionnement les valeurs de ses différents composants, à savoir la position de la tête L/E, l'état de l'automate et éventuellement le contenu de la mémoire auxiliaire (lorsqu'elle existe).

Il existe deux configurations spéciales :

1. La configuration ***initiale*** est celle qui correspond à l'état initial q0 et où la tête de L/E est positionnée sur le premier symbole du mot à lire.
2. Une configuration ***finale*** est celle qui correspond à un des états finaux qf et où le mot a été entièrement lu.

Mot reconnu par un automate

On dit qu'un mot **est reconnu par un automate** si, à partir d'une configuration initiale, on arrive à une configuration finale à travers une *succession de configurations* intermédiaires.

Un mot w est ***reconnu par l'automate*** A s'il existe une *configuration successive*

Configuration-initiale (w) $\vdash^*$ configuration-final(w)

$(q_0, w) \vdash^* (q_f, \varepsilon)$

La relation $\vdash$ permet de formaliser la notion d'étape élémentaire de calcul d'un automate. Ainsi on écrira, pour a dans A et v dans A* : $(q, \boldsymbol{a}v) \vdash (\delta(q, a); v)$

Langage reconnu par un automate

On dit qu'un ***langage est reconnu par un automate*** X lorsque tous les mots de ce langage sont reconnus par l'automate on note L(X)

L(X)={$w \in A^*$ / Configuration-initiale (w) $\vdash^*$ configuration-final(w)}

L(X)={ $w \in A^*$ / $(q_0, w) \vdash^* (q, \varepsilon)$, avec $q \in QF$ }

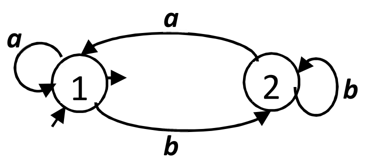
Passage de l'automate vers l'expression régulière

Soit X = (A, Q, q_0, Q_F, δ) un automate à états fini quelconque.
On note par $\mathbf{L_i}$ le langage reconnu par l'automate si son état initial était $\mathbf{q_i}$.
Par conséquent, trouver le langage reconnu par l'automate revient à trouver $\mathbf{L_0}$ étant donné que la reconnaissance commence à partir de l'état initial $\mathbf{q_0}$. L'automate permet d'établir un système d'équations aux langages de la manière suivante :

– si $\delta(q_i, a) = q_j$ alors on écrit : $L_i = aL_j$;
– si $q_i \in Q_F$, alors on écrit : $L_i = \varepsilon$
– si $L_i = \alpha$ et $L_i = \beta$ alors on écrit : $L_i = \alpha|\beta$;

Il suffit ensuite de résoudre le système précédant à des substitutions et en utilisant la règle suivante :
la solution de l'équation $L = \alpha L|\beta$ ($\varepsilon \notin \alpha$) est le langage $L = \alpha^*\beta$ (Le lemme d'Arden)

Exemple :
On cherche à déterminer le langage de l'automate suivant.



On s'intéresse au langage L1 des mots qui passent par l'état 1, et à L2 celui des mots qui passent par 2.
On a les équations suivantes.

- $L_1 = \varepsilon + aL_1 + bL_2$
- $L_2 = aL_1 + bL_2$

Ici ε apparaît puisque l'état 1 est initial.
Par le lemme d'Arden sur la seconde équation, il vient L2 = b*(aL1).
En récrivant la première, on a

- $L_1 = \varepsilon + aL_1 + b(b^*aL_1)$
  $= \varepsilon + aL_1 + b^+aL_1$
  $= b^*aL_1 + \varepsilon$

Par le lemme d'Arden sur cette équation, on obtient finalement que :
$L_1 = (b^*a)^*\varepsilon = (b^*a)^*$

Automate fini déterministe AFD

Définition : Un AEF (A,Q, q0, QF, δ) est dit **déterministe** si les deux conditions sont vérifiées :
- $\forall q_i \in Q$, $\forall a \in X$, il existe au plus un état q_j tel que $\delta(q_i, a) = q_j$;
- L'automate ne comporte pas de ε-transitions.

Algorithme : Déterminiser un AEF sans les ε-transitions

Principe : considérer des ensembles d'états plutôt que des états (dans l'algorithme suivant, chaque ensemble d'états représente un état du futur automate).

1- Partir de l'état initial $E^{(0)} = \{q_0\}$ (c'est l'état initial du nouvel automate) ;
2- Construire $E^{(1)}$ l'ensemble des états obtenus à partir de $E^{(0)}$ par la transition a :

$$E^{(1)} = \bigcup_{q' \in E(0)} \delta(q', a)$$

3- Recommencer l'étape 2 pour toutes les transitions possibles et pour chaque nouvel ensemble $E^{(i)}$;

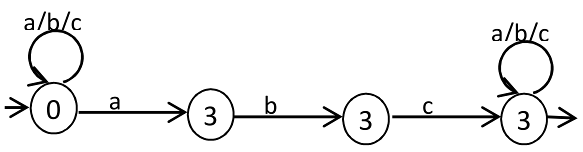
$$E^{(i)} = \bigcup_{q' \in E(i-1)} \delta(q', a)$$

4- Tous les ensembles contenant au moins un état final du premier automate deviennent finaux ;
5- Renuméroter les états en tant qu'états simples.

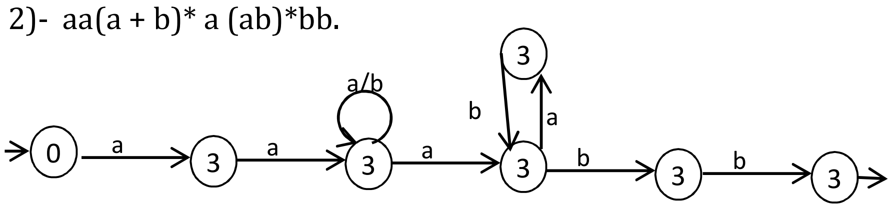
Exercice 1: ***Expressions vs automates***

I. Trouver, intuitivement, des automates qui acceptent les langages dénotés par les expressions régulières :

1)- (a + b + c)*abc(a + b + c)* ;



2)- aa(a + b)* a (ab)*bb.

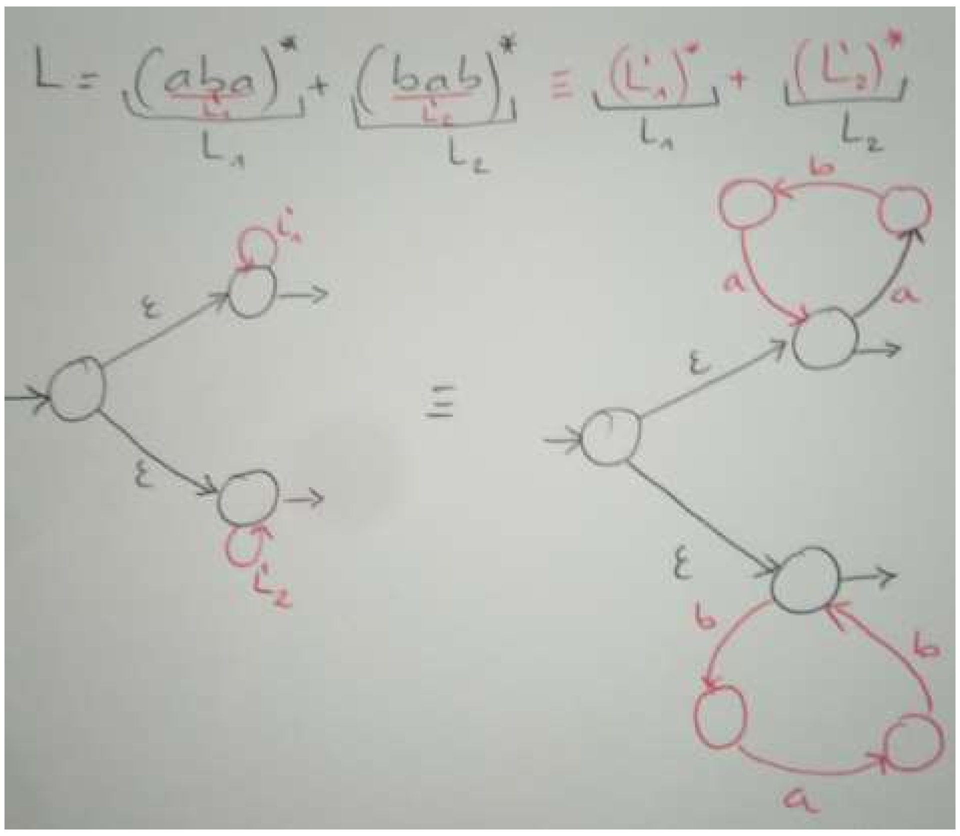


3)- (aba)*+ (bab)*

3-1 avec ε-transition

(aba)* + (bab)* (L1=L((aba)*) ∪ L2=L((bab)*)

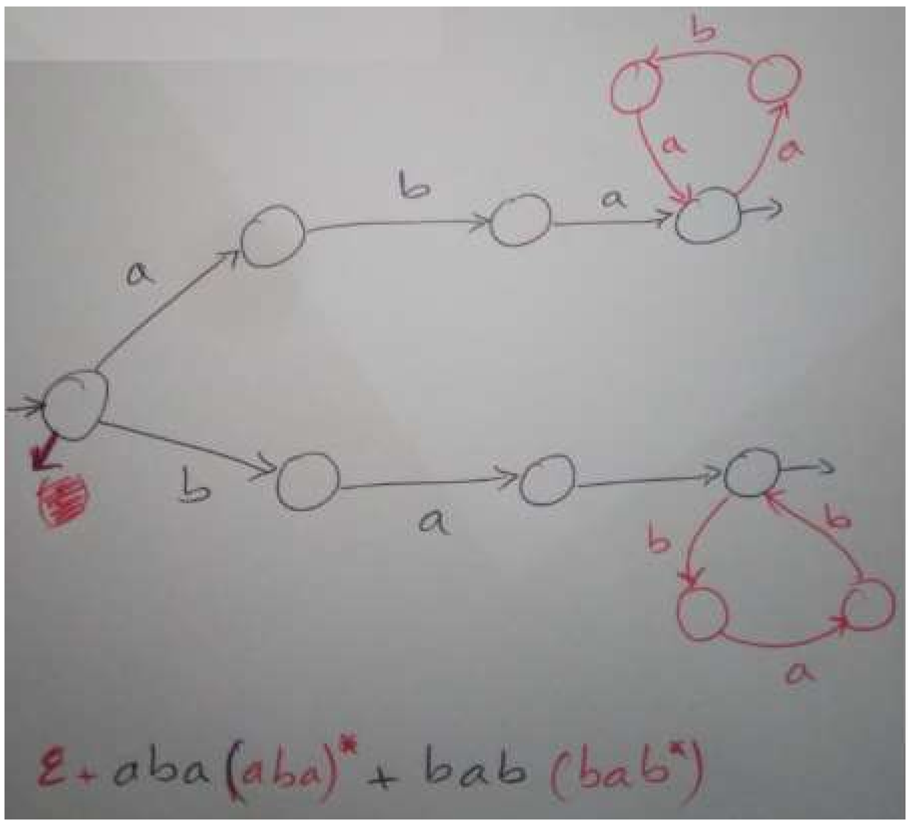
Donc l'automate de L1∪L2



3-2 sans ε-transition

$(aba)^* + (bab)^* = [\varepsilon + (aba)^+] + [\varepsilon + (bab)^+] = [\varepsilon + aba(aba)^*] + [\varepsilon + bab(bab)^*]$

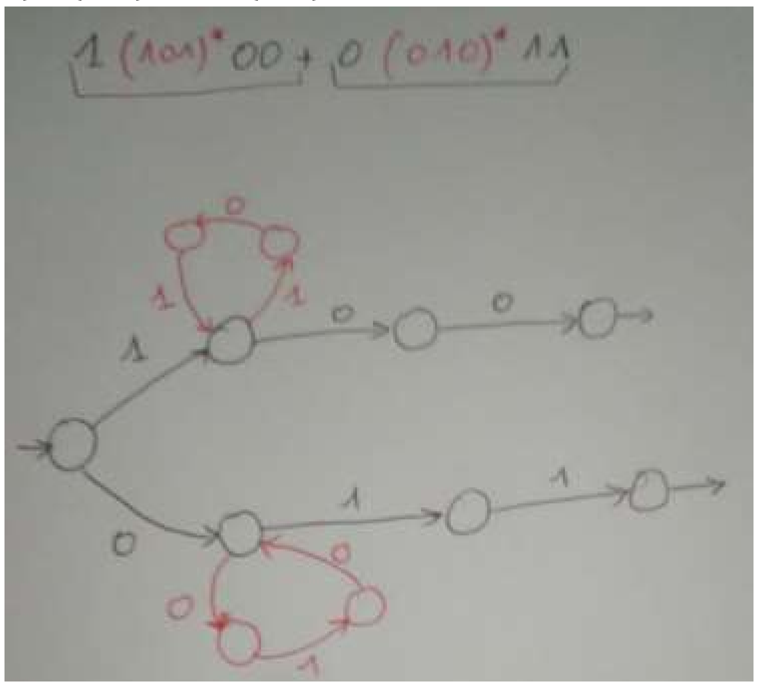
$= \varepsilon + aba(aba)^* + bab(bab)^*$



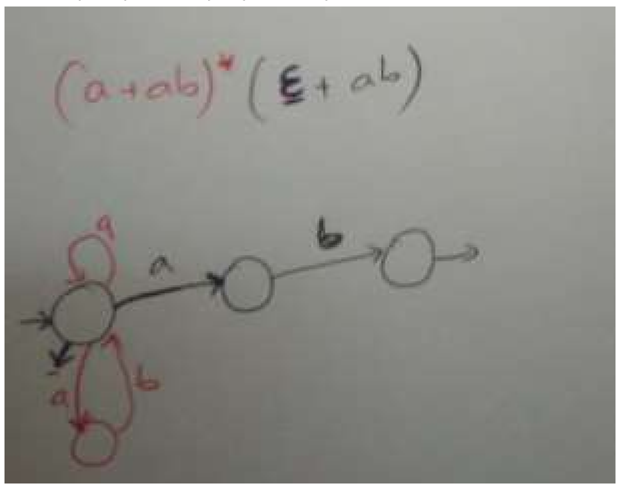
4)- a*b*a*b*

---

5)- 1(101)*00 + 0(010)*11



6) - (a + ab)*(ε + ab)



II. En utilisant l'algorithme de Glushkov, construire des automates correspondants aux expressions régulières :

1)- (a b + b)* b a
  1 2 3 4 5

	a	b
0	1	3,4
1	-	2
2	1	3,4
3	1	3,4
4	5	-
5	-	-

Etat d'enté : 0

Etats de sorties : 5

2)- a* b* a* b*
  1 2 3 4

	a	b
0	1,3	2,4
1	1,3	2,4
2	3	2,4
3	3	4
4	-	4

Etat d'enté : 0

Etats de sorties : {0,1,2,3,4}

3)- (a b + a)* b a
  1 2 3 4 5

	a	b
0	1,3	4
1	-	2
2	1,3	4
3	1,3	4
4	5	-
5	-	-

Etat d'enté : 0

Etats de sorties : {5}

4)- (1 1* 0 0* 1)* 0 1*

q_1 q_2 q_3 q_4 q_5 q_6 q_7

	0	1
q_0	q_6	q_1
q_1	q_3	q_2
q_2	q_3	q_1
q_3	q_4	q_5
q_4	q_4	q_5
q_5	q_6	q_1
q_6	-	q_7
q_7	-	q_7

Etat d'enté : q0

Etats de sorties : {q_6,q_7}

5)- (a + b a)* b b b* a

1 2 3 4 5 6 7

	a	b
0	1	2,4
1	1	2,4
2	3	-
3	1	2,4
4	-	5
5	7	6
6	7	6
7	-	-

Etat d'enté : 0

Etats de sorties : {7}

6)- (a + b)*(abb + ε).

1 2 345

	a	b
0	1,3	2
1	1,3	2
2	1,3	2
3	-	4
4	-	5
5	-	-

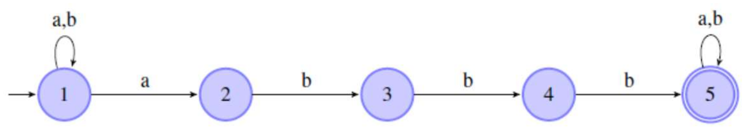
Etat d'enté : 0

Etats de sorties : {0,1,2,5}

Exercice 2: ***Automates vs expressions***

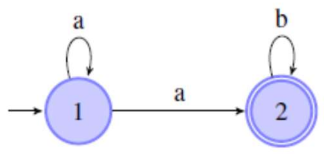
Soit A = {a,b}. Pour chacun des automates suivants, dire s'il est déterministe et s'il est complet. Décrire ensuite le langage reconnu par cet automate.

Automate M1



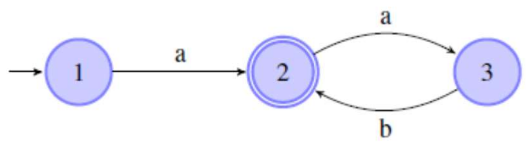
Exp(M1)=**(a+b)*abbb(a+b)***

Automate M2



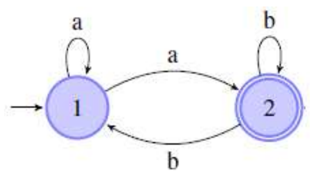
$Exp(M2)=\mathbf{a^*ab^*}$

Automate M3



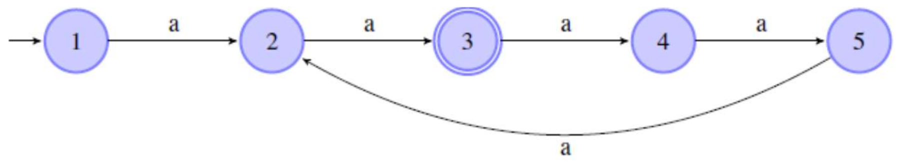
$Exp(M3)=\mathbf{a+a(ab)^*=(a(\varepsilon+(ab)^*)=a(ab)^*}$

Automate M4



$Exp(M4)=\mathbf{(}$

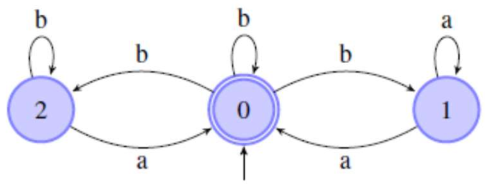
Automate M5



$Exp(M5)=\mathbf{aa+aa(aaaa)^*=aa(aaaa)^*}$

Exercice 3 :

Soit l'automate M suivant



1. Déterminiser l'automate M

	a	b
$\{q_0\}$ 0	-	0,1,2
$\{q_1\}$ 0,1,2	0,1	0,1,2
$\{q_2\}$ 0,1	0,1	0,1,2

Etat d'entré : $\{q_0\}$

Etats de sortie : $\{q_0, q_1, q_2\}$

2. Donner le système d'équations de l'automate M

- $L_0=bL_1+\varepsilon$
- $L_1=bL_1+aL_2+\varepsilon$
- $L_2=aL_2+\varepsilon$

3. Donner le langage reconnu par cet automate

On résoudre le système d'équations de l'automate M

- $L_0=bL_1+\varepsilon \rightarrow L_0=b(b^*(aa^*+\varepsilon))+\varepsilon=bb^*aa^*+bb^*+\varepsilon$ ------(3)
- $L_1=bL_1+aL_2+\varepsilon \rightarrow L_1=bL_1+aa^*+\varepsilon \rightarrow L_1=b^*(aa^*+\varepsilon)$ ------(2)
- $L_2=aL_2+\varepsilon \rightarrow L_2=a^*\varepsilon= a^*$ -----(1)

Le langage reconnu par l'automate M est $L_0= bb^*aa^*+bb^*+\varepsilon$