Les Langages Réguliers

S. Mazouz & B. Laichi

Plan

1) Automates d’états finis

2) Grammaires régulières et automates d’états finis

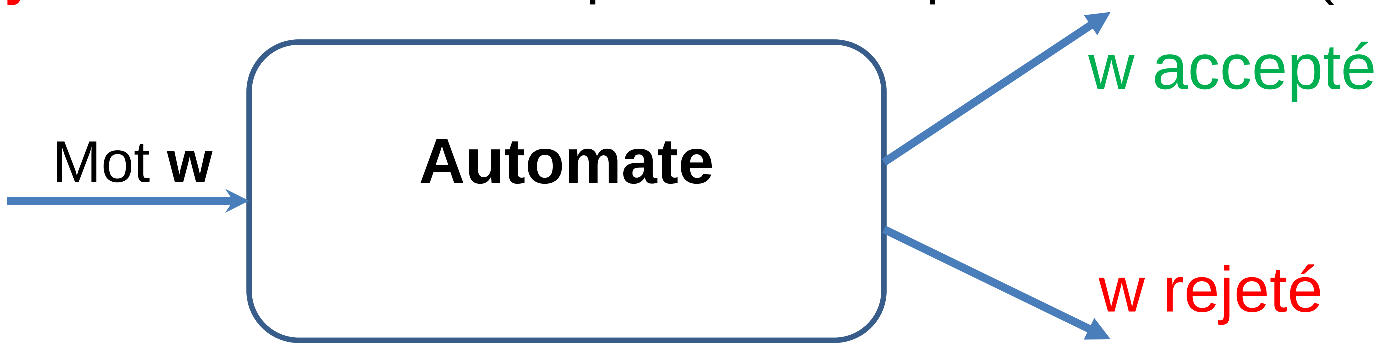
3) Expressions régulières

4) Expressions régulières et automates d’états finis

AUTOMATES D’ETATS FINIS

Définition : Un automate **(ou système de reconnaissance)** est une **machine abstraite** qui permet de **reconnaître** les mots d’un langage. Il prend en entrée un mot **w**, et fournit comme résultat :

- **accepté :** si le mot est reconnu par l’automate ($\in$Langage)
- **rejeté :** si le mot n’est pas reconnu par l’automate ($\notin$Langage)



A chaque type de langage, on associe un type d’automate :

- Aux langages de Type 3 $\longrightarrow$ Les Automates d’Etats Finis
- Aux langages de Type 2 $\longrightarrow$ Les Automates à Piles
- Aux langages de Type 1 $\longrightarrow$ Les Automates à Bornes Linéaires
- Aux langages de Type 0 $\longrightarrow$ Les Machines de Turing

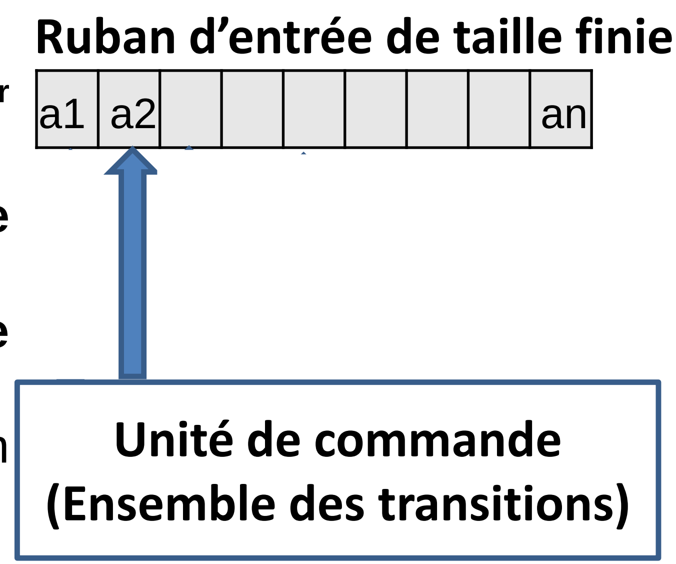
AUTOMATES D’ETATS FINIS (AEF)

Un AEF lit les symboles d’un mot à reconnaitre un par un et va d’état en état selon les transitions. Le mot lu est soit accepté par l'automate soit rejeté

Fonctionnement :

Le ruban d’entrée contient le mot à reconnaître.

- L’automate **démarre à l’état initial.**
- La tête de lecture **est initialement sur le 1er symbole du mot d’entrée**
- Les symboles du mot sont lus **de la gauche vers la droite.**
- L’automate avance d’une **case vers la droite** à chaque exécution d’une transition.
- L’automate évolue d’un état à un autre en fonction du **symbole lu** et de **l’état courant**.
- Le mot est **reconnu** si et seulement si :
  - l’automate **a terminé** la lecture du mot
  - **et** se trouve dans **un état final**.



AUTOMATES D'ETATS FINIS

Définition (Automate d'états finis)

Un automate d'états finis déterministe est un cinquplet

$$\mathbf{A = (X, Q, q_0, \delta, F)} \text{ où :}$$

- X est un **alphabet d'entrée**, fini et non vide
- Q est un **ensemble d'états**, fini et non vide
- $q_0 \in Q$ est un **état initial**
- $F \subseteq Q$ est un **ensemble d'états finaux**
- δ est une **fonction de transitions** définie de Q x X dans Q et qui associe à un état donné **p** et un symbole donné **a** un état d'arrivée **q** i.e. $\delta(p, a)=q$

Remarque : $\delta(p, a)=q$ signifie que l'automate réalise un déplacement (une transition) de l'état **p** vers l'état **q** en lisant la lettre **a**.

AUTOMATES D'ETATS FINIS

Représentation graphique :

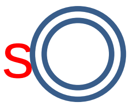
Les automates d'états finis sont souvent représentés par des graphes orientés dont :

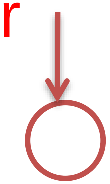
- les sommets correspondent aux états
- les arcs correspondent aux transitions
- l'arc ayant comme extrémité initiale $p \in Q$ et pour extrémité terminale $q \in Q$ et étiqueté par $a \in X$ représente la transition

$$\delta(p, a) = q$$

AUTOMATES D’ETATS FINIS

Représentation graphique :

- un état final est représenté par deux cercles concentriques 
- un état initial est représenté par une flèche incidente sur l’état initial



Remarque :

Un AEF possède un seul état initial mais peut avoir plusieurs états finaux

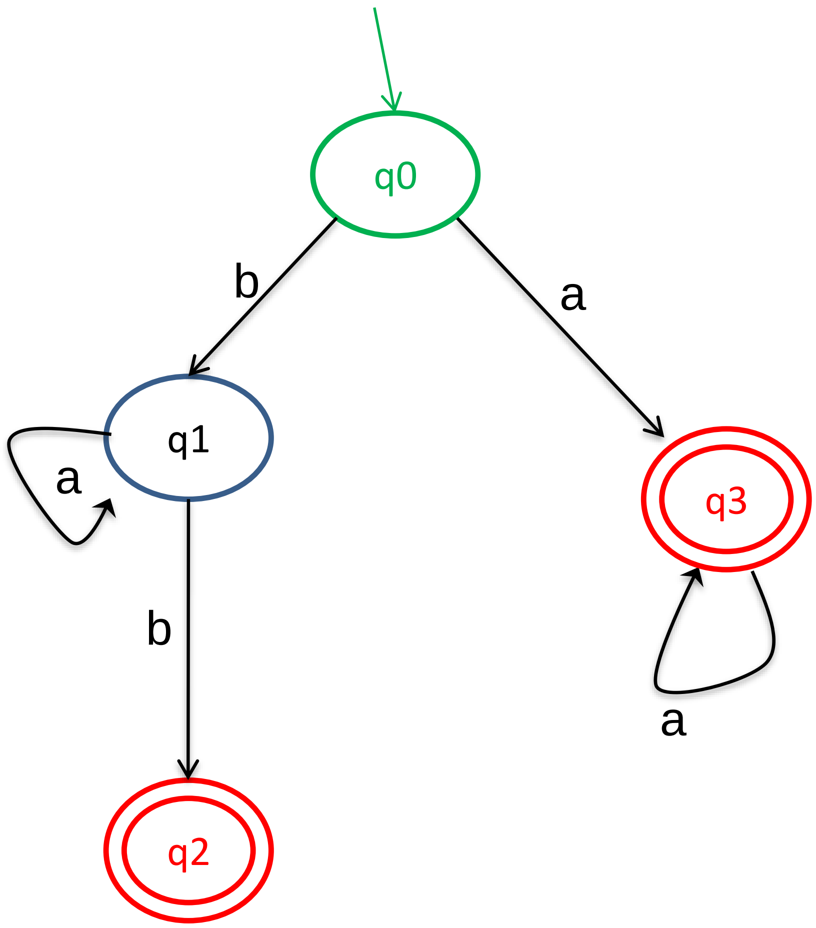
Notation : Deux transitions ou plus entre deux états seront représentées par un seul arc entre ces deux états comme suit :

Automates d'Etats Finis

Exemple : Soit l'AEF suivant

A =(X, Q, q0, δ, F) tels que :

- X = {a, b}
- Q = {q0, q1, q2, q3}
- q0 est l'état initial
- δ(q0, a)=q3
  δ(q0, b)=q1
  δ(q1, a)=q1
  δ(q1, b)=q2
  δ(q3, a)=q3
- F = {q2, q3}



AUTOMATES D'ETATS FINIS

Représentation matricielle :

Les automates d'états finis peuvent aussi être représentés par des matrices dont :

- les indices des lignes correspondent aux états
- les indices des colonnes correspondent aux éléments de X.
- un élément de la matrice de ligne **q** et colonne **a** correspond à la transition **δ(q, a).**

Symbole

état\lettre		a	
q		δ(q, a)	

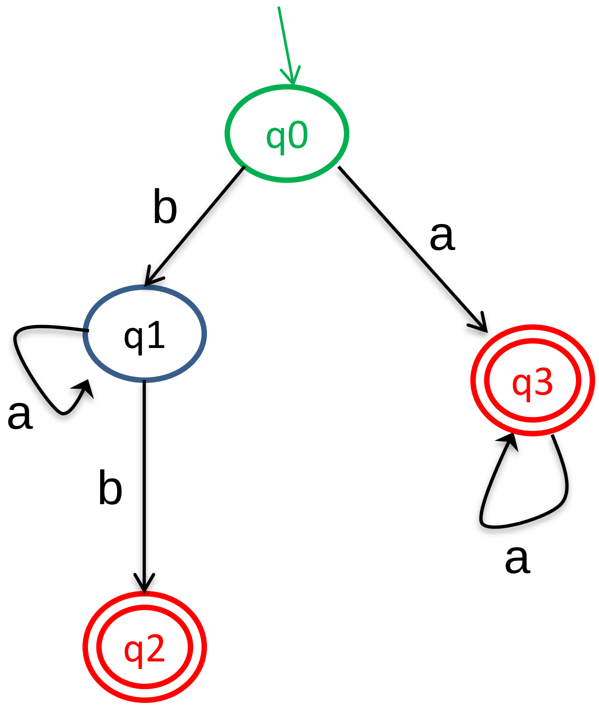
état

Transition

AUTOMATES D'ETATS FINIS

Exemple :

Représentation Graphique



Représentation Matricielle

etat\lettre	a	b
q0	q3	q1
q1	q1	q2
q2	–	–
q3	q3	–

$\delta(q0, a)$

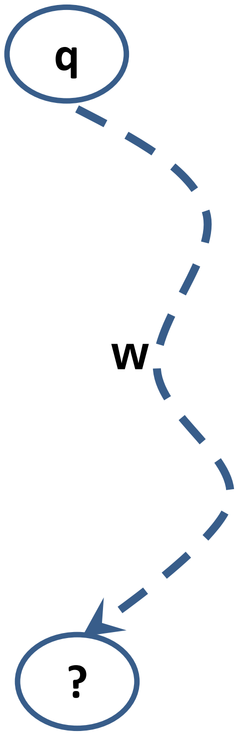
Transition non définie

AUTOMATES D'ETATS FINIS

Langage reconnu par un automate d'états finis :

Question :

- Quelle est la **succession de transitions** permettant de lire un mot **w** à partir d'un état donné **q**, si elle existe ?
- Quel est l'état d'arrivée après cette succession de transitions.



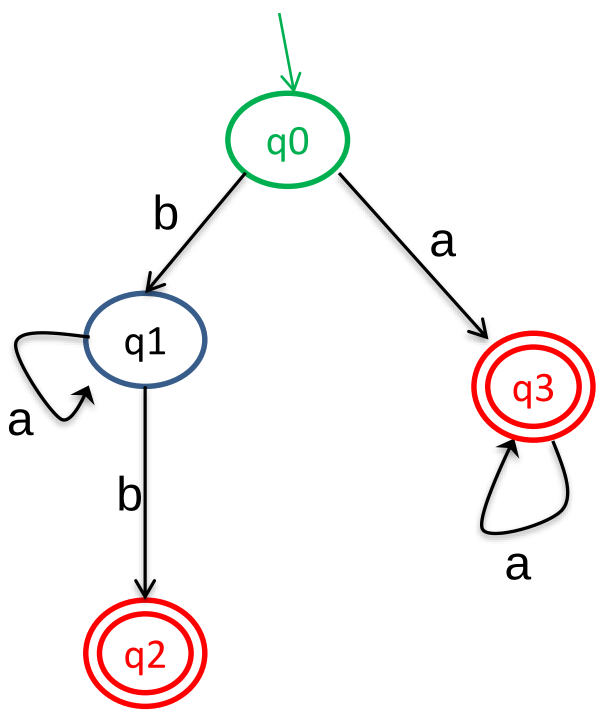
Définir la fonction de

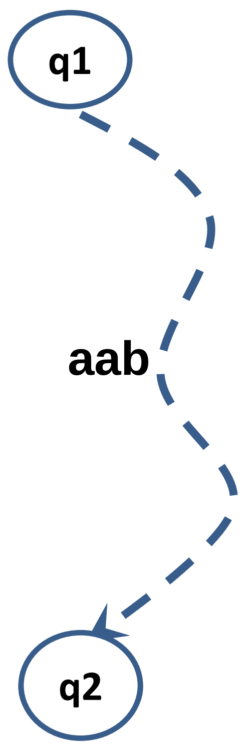
succession de transitions.

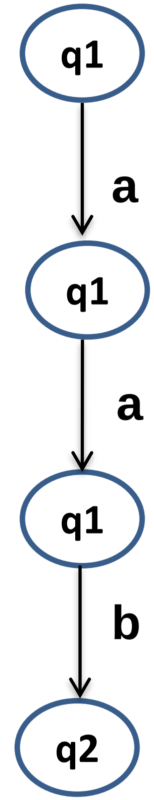
Automates d'Etats Finis

Fonction de succession de transitions :

Illustration par un exemple **:** Quel est la succession de transitions pour lire **aab** à partir de l'état **q1**?





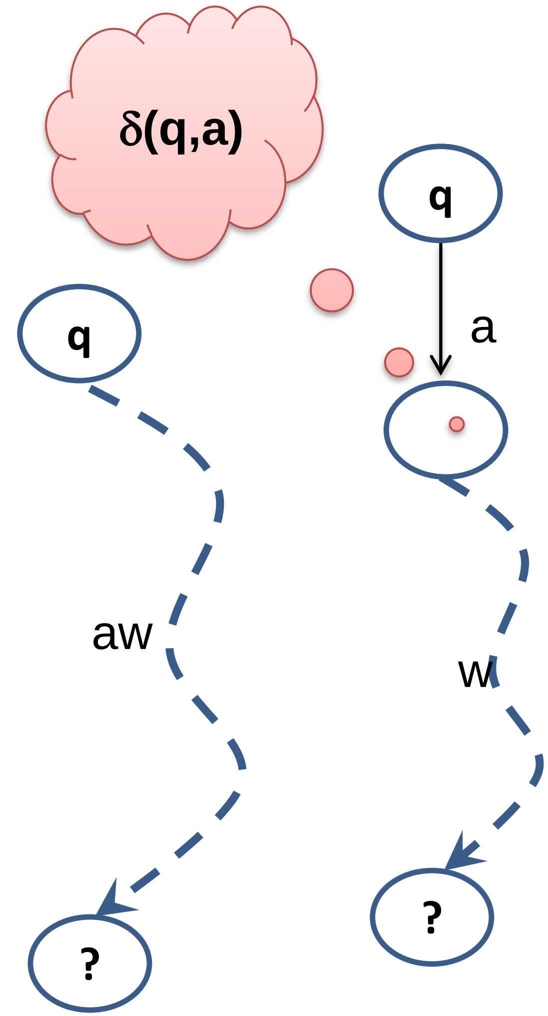


AUTOMATES D'ETATS FINIS

Soit A=(X, Q, q0, δ, F) un automate d'états fini déterministe.

On étend naturellement, la fonction de transition δ à la fonction de succession de transitions δ^* définie de **Q x X*** dans **Q** comme suit :

- $\delta^*(q, \varepsilon) = q$
- $\delta^*(q, a) = \delta(q, a)$ $a \in X$
- $\delta^*(q, aw) = \delta^*(\delta(q, a), w)$ $a \in X$ et $w \in X^*$



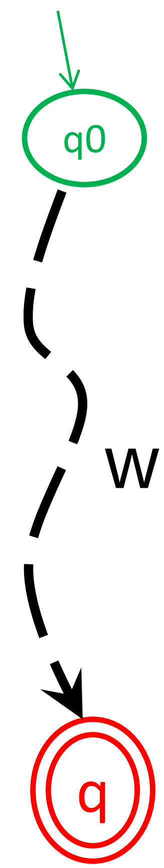
$\delta^*(q, aw) = ?$

AUTOMATES D'ETATS FINIS

Condition de reconnaissance :

Un mot w est **reconnu** **(accepté)** par l'automate A ssi :

- l'automate A lit le mot w à partir de q_0
- et atteint un état final**.**



Autrement dit, **il existe un état final q_F**$\in$F tel que δ ***(q₀, w) = q_F.**

On dit que **A accepte le mot w** (ou que w est accepté par A).

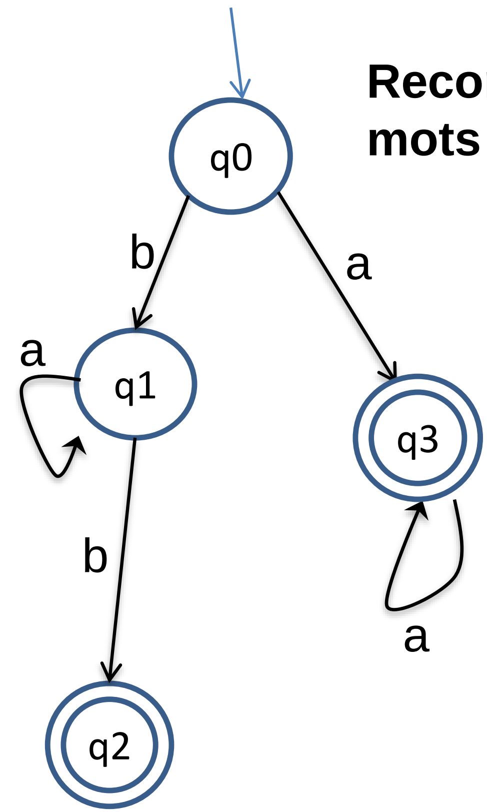
AUTOMATES D'ETATS FINIS

$\delta^*(q, aw)=\delta^*(\delta\ (q, a), w)$, $a \in X$ et $w \in X^*$

$\delta^*(q0, bab) = \delta^*(\delta(q0, b), ab)$
$= \delta^*(\mathbf{q1}, ab)$
$= \delta^*(\delta(q1, a), b)$
$= \delta^*(q1,b)$
$= \delta(q1,b)$
$=$ **q2** et **q2 est un état final**

$\delta^*(q0, \mathbf{a}aa) = \delta^*(\delta(q0, a), aa)$
$= \delta^*(\mathbf{q3}, aa)$
$= \delta\ ^*(\delta\ (q3, a), a)$
$= \delta^*(\mathbf{q3},a)$
$= \delta(q3,a)$
$=$ q3 et **q3 est un état final**

Reconnaître les mots : bab, aaa



Les mots bab et aaa sont acceptés par cet automate

AUTOMATES D'ETATS FINIS

$\delta^*(q0, ab) = \delta^*(\boldsymbol{\delta(q0, a)}, b)$

$= \delta^*(q3, b)$

$= \delta(q3, b)$

mais **à l'état q3, l'automate ne peut pas lire b**

$\delta^*(q0, ba) = \delta^*(\boldsymbol{\delta(q0, b)}, a)$

$= \delta^*(\mathbf{q1}, a)$

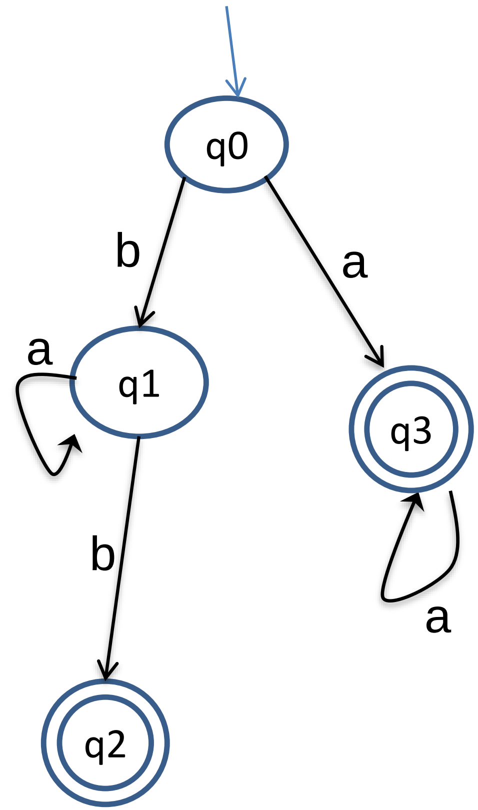
$= \delta(\mathbf{q1}, a)$

$= q1$

mais **q1 n'est pas final**

Les mots ab et ba ne sont pas acceptés

Reconnaître les mots : ab, ba



AUTOMATES D'ETATS FINIS

Définition (Langage Reconnu par AEF)

Soit $A=(X, Q, q_0, \delta, F)$ un automate déterministe.

Le langage reconnu par l'automate A est l'ensemble

$$\mathbf{L(A) = \{w \in X^* / \delta^*(q_0, w) \in F\}.}$$

Un langage L sur X est **régulier** (ou **reconnaissable**) s'il existe au moins un automate d'états finis A ayant X comme alphabet d'entrée tel que **L=L(A).**

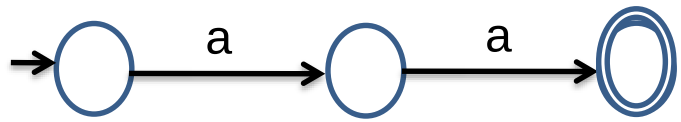
Notation : On note **Rec(X*)** la famille des langages reconnaissables sur l'alphabet X.

AUTOMATES D'ETATS FINIS

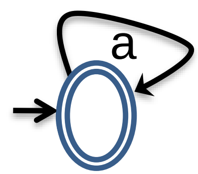
Exemples : Donner un AEF pour chacun des langages :

L1= $\{a\}^2$,   L2= $\{a^n / n \geq 0\}$ et L3= $\{a^n / n \geq 1\}$,

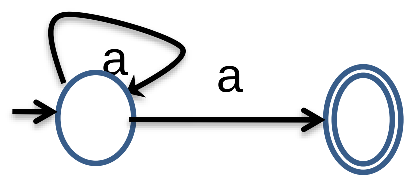
AEF de L1= $\{a\}^2$:

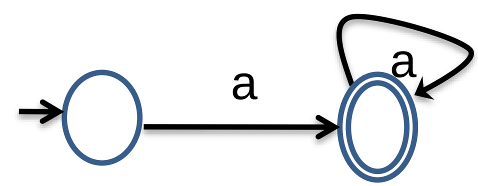


AEF de L2= $\{a^n / n \geq 0\}$:



AEF de L3= $\{a^n / n \geq 1\}$:





Cet AEF n'est pas déterministe

Rappel : Pour un AEF déterministe

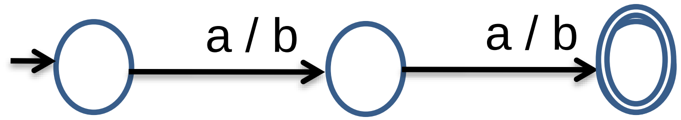
La **fonction de transitions** δ définie de Q x X dans Q

AUTOMATES D'ETATS FINIS

Exemples : Donner un AEF pour chacun des langages

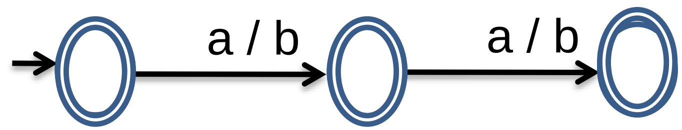
L1= $\{a, b\}^2$, L2= $\{w \,/\, w \in \{a, b\}^*, |w| \leq 2\}$ et L3= $\{a, b\}^*$

AEF de L1= $\{a,b\}^2$:

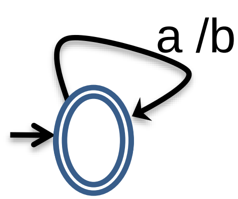


AEF de L2 : on a $\{a,b\}^2 \subseteq$ L2 :

L2= $\{a,b\}^2 \cup \{a, b, \varepsilon\}$:

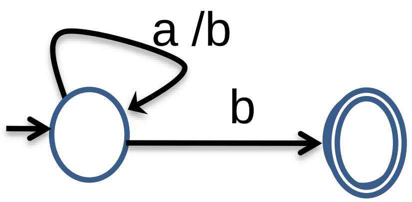


AEF de L3= $\{a,b\}^*$:

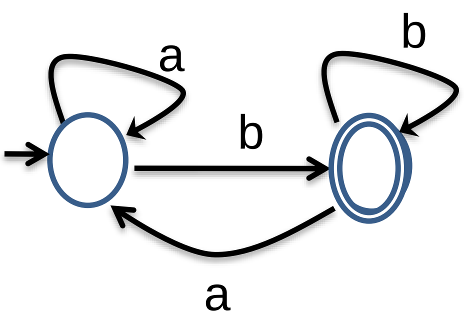


AUTOMATES D'ETATS FINIS

Exemple Donner un AEF déterministe reconnaissant le langage { wb / w$\in${a, b}*}



Cet AEF n'est pas déterministe



Cet AEF est déterministe

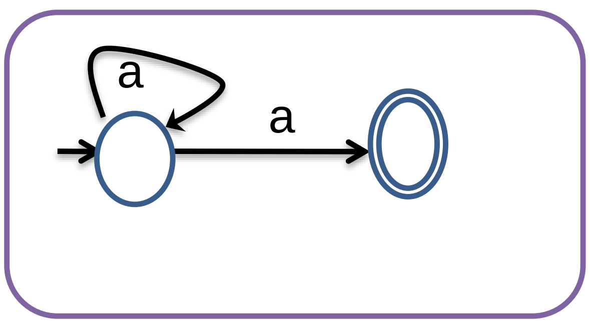
AUTOMATES D'ETATS FINIS

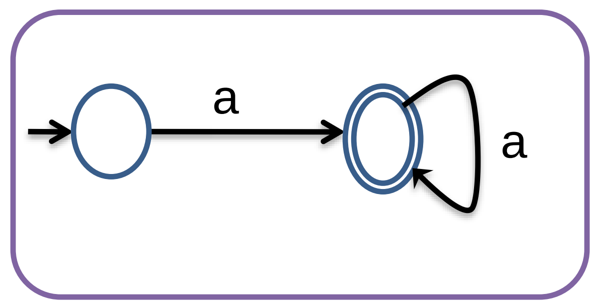
Définition (Equivalence de deux AEFs)

Deux automates d'états finis A1 et A2 sont équivalents, noté A1≡A2, si et seulement s'ils acceptent le même langage.

$$A1 \equiv A2 \;\Leftrightarrow\; L(A1)=L(A2)$$

Exemple : Ces deux automates sont équivalents.





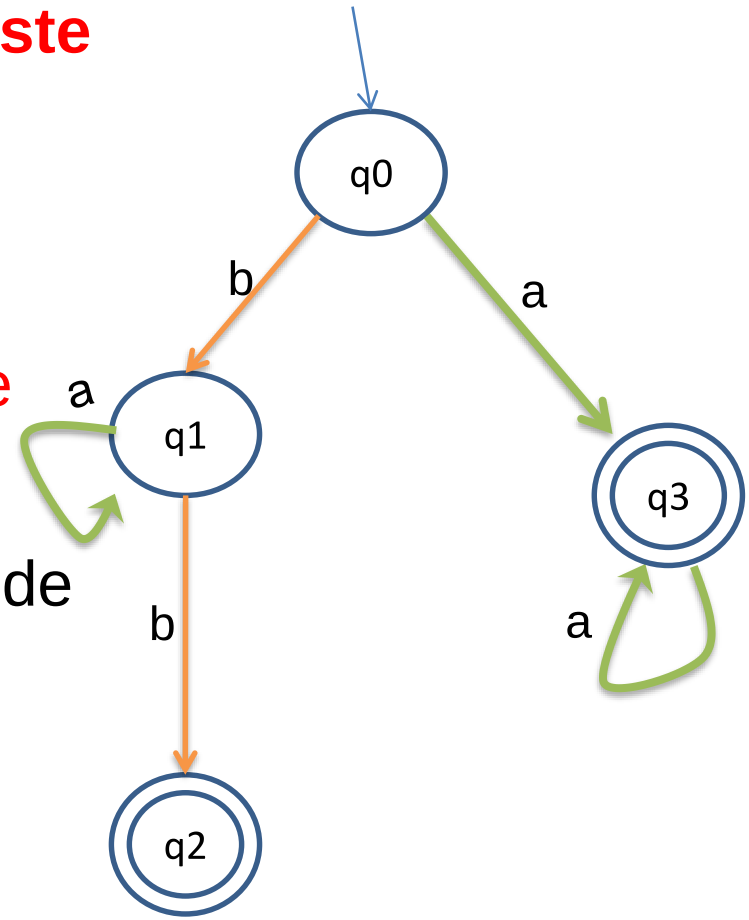
Remarque : Un langage peut être reconnu par plusieurs automates. Par contre un automate ne peut reconnaitre qu'un seul langage.

AUTOMATES DÉTERMINISTES

Un automate d'états fini est **déterministe** si et seulement si :

- à un état
- à un symbole d'entrée

la fonction δ associe au plus une seule transition.



Autrement dit, la fonction δ est définie de Q x X dans Q.

Remarque :

Dans les automates déterministes, il n y a pas de choix à faire pour l'état suivant après la lecture d'un certain symbole.

AUTOMATES DÉTERMINISTES

Un automate déterministe est dit complet ssi :

à toute paire $(q,a) \in Q x X$, la fonction δ associe exactement un état.

Ainsi, la fonction de transition **δ** est une **application fonctionnelle** : Q x X dans Q.

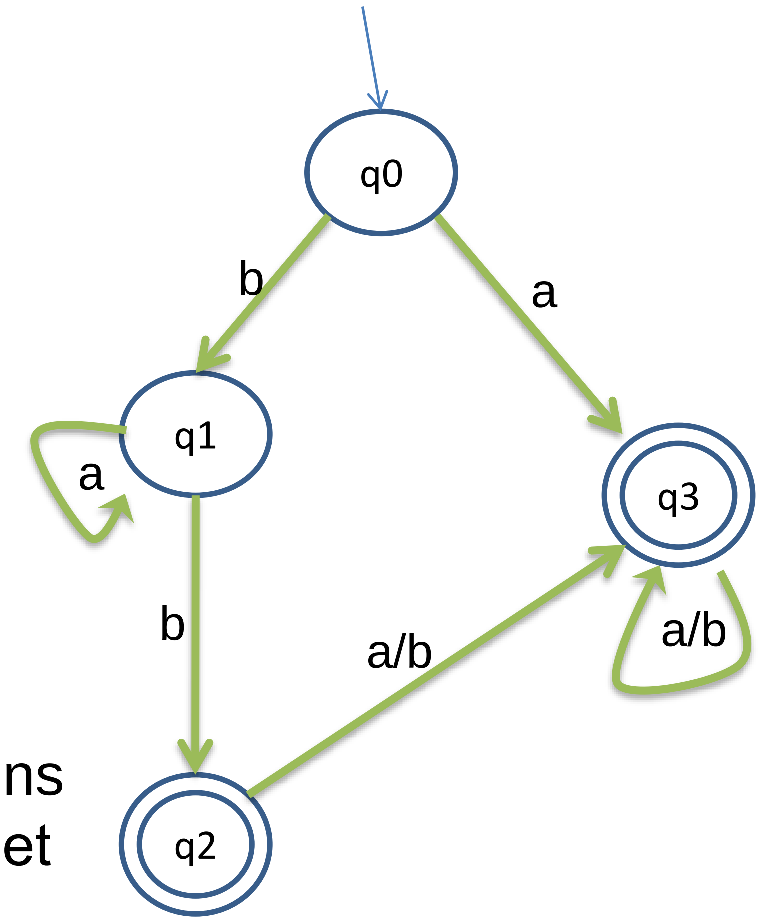
Remarque :

Dans un automate complet, il y a possibilité de lire n'importe quel symbole à partir de n'importe quel état.

AUTOMATES DÉTERMINISTES

Exemple :

état\lettre	a	b
q0	q3	q1
q1	q1	q2
q2	q3	q3
q3	q3	q3



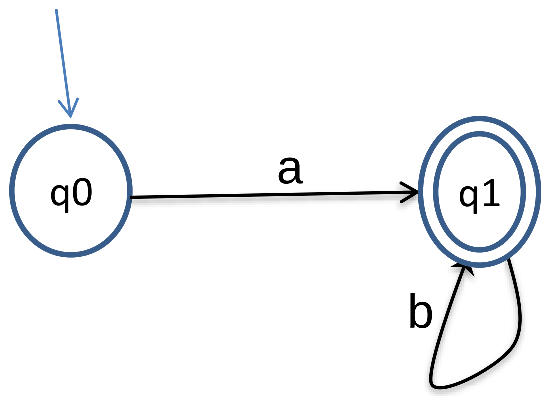
Automate déterministe et complet

Remarque : La matrice des transitions d'un automate déterministe et complet est pleine.

AUTOMATES DÉTERMINISTES

Exemple : Cet AEF est-il déterministe et complet ?

Cet automate est **déterministe mais non complet.**



En effet, dans l'état **q0**, on ne peut pas lire la lettre **b**.

Le langage reconnu par cet automate est : $\{ab^n / n \geq 0\}$.

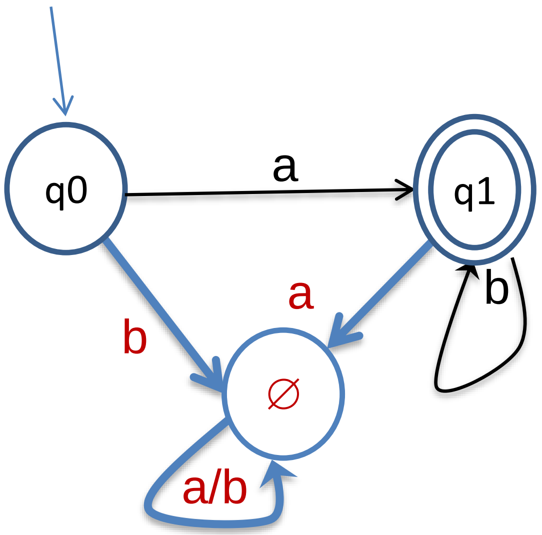
Remarques :

- ✓ Un automate déterministe non complet ne permet pas de lire certains mots de X*.
- ✓ Un automate peut lire des mots mais ne pas les reconnaître.

AUTOMATES DÉTERMINISTES

Pour rendre complet un «automate déterministe non complet» il suffit de :

rajouter un état, appelé «états puits» généralement noté ∅, et de rajouter toutes les transitions manquantes vers cet état.

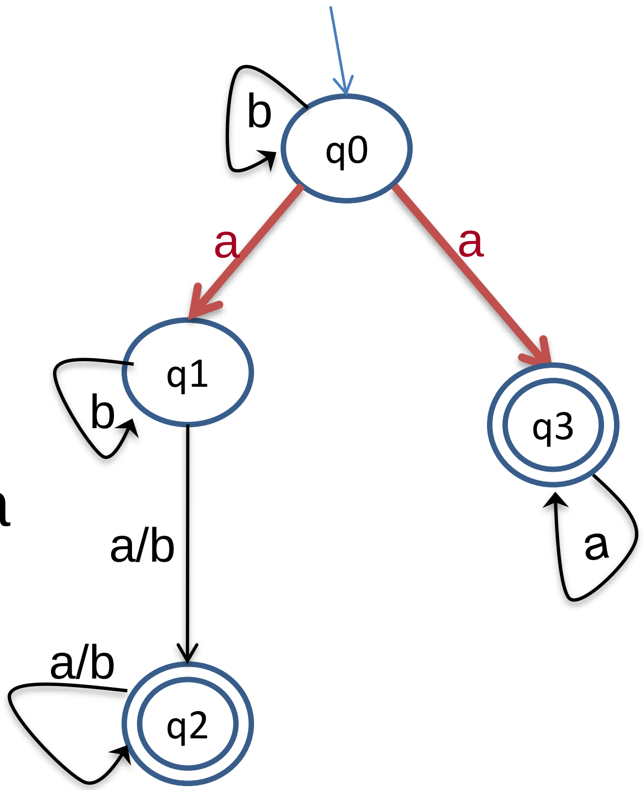


Remarques :

- ✓ Un automate déterministe complet permet de lire tous les mots de l'alphabet (pas nécessairement les reconnaitre)
- ✓ L'automate complet obtenu reconnaît le même langage que l'automate initial (l'état puits n'est pas un état final).

AUTOMATES NON DÉTERMINISTES

Les automates d'états finis non déterministes sont des automates où : l'on permet plusieurs transitions correspondant à la même lettre **à** partir des états de l'automate.



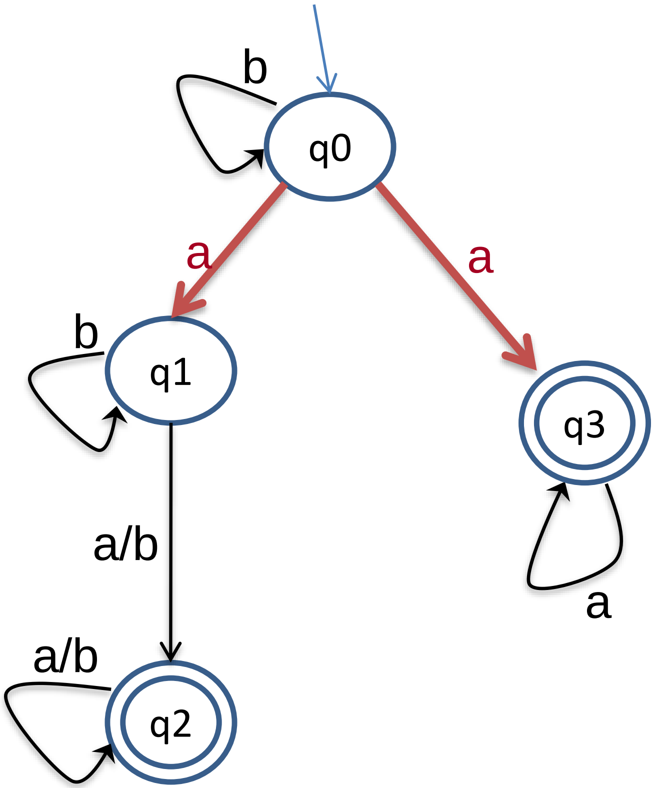
Dans l'exemple, à partir de q0, on a le choix entre deux transitions par a : l'une vers q1 et l'autre vers q3.
Donc, δ (q0, a)=**{q1, q3}**

Dans ce cas, δ est une fonction de transition définie de : QxX dans **l'ensemble des parties** de Q. **δ : QxX→$\mathscr{P}$ (Q)**

AUTOMATES NON DÉTERMINISTES

Exemple :

etat\lettre	a	b
q0	**{q1, q3}**	{q0}
q1	{q2}	**{q1, q2}**
q2	{q2}	{q2}
q3	{q3}	_



Remarque :

- ✓ Pour alléger les notations, on peut omettre les accolades.
- ✓ Dans les automates non déterministes, un choix est permis pour passer à l'état suivant.

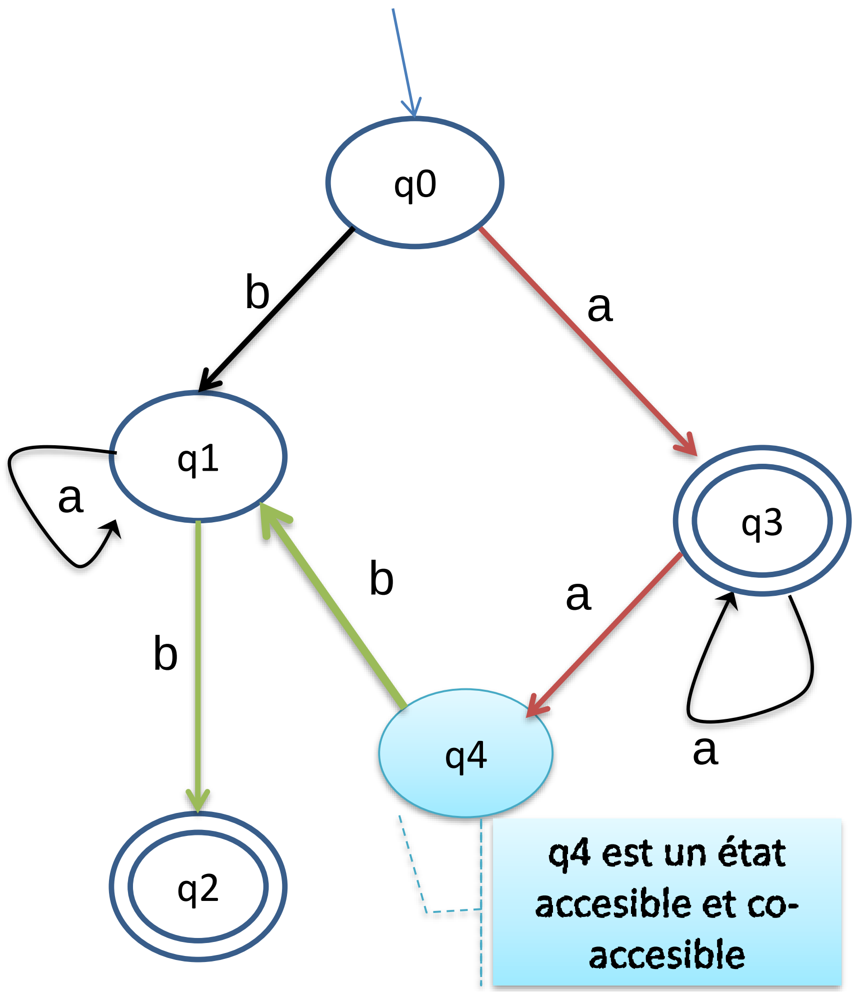
AUTOMATES D'ETATS FINIS

Définitions :

Un état **q** est **accessible** s'il existe un chemin de l'état initial de l'automate vers **q**.

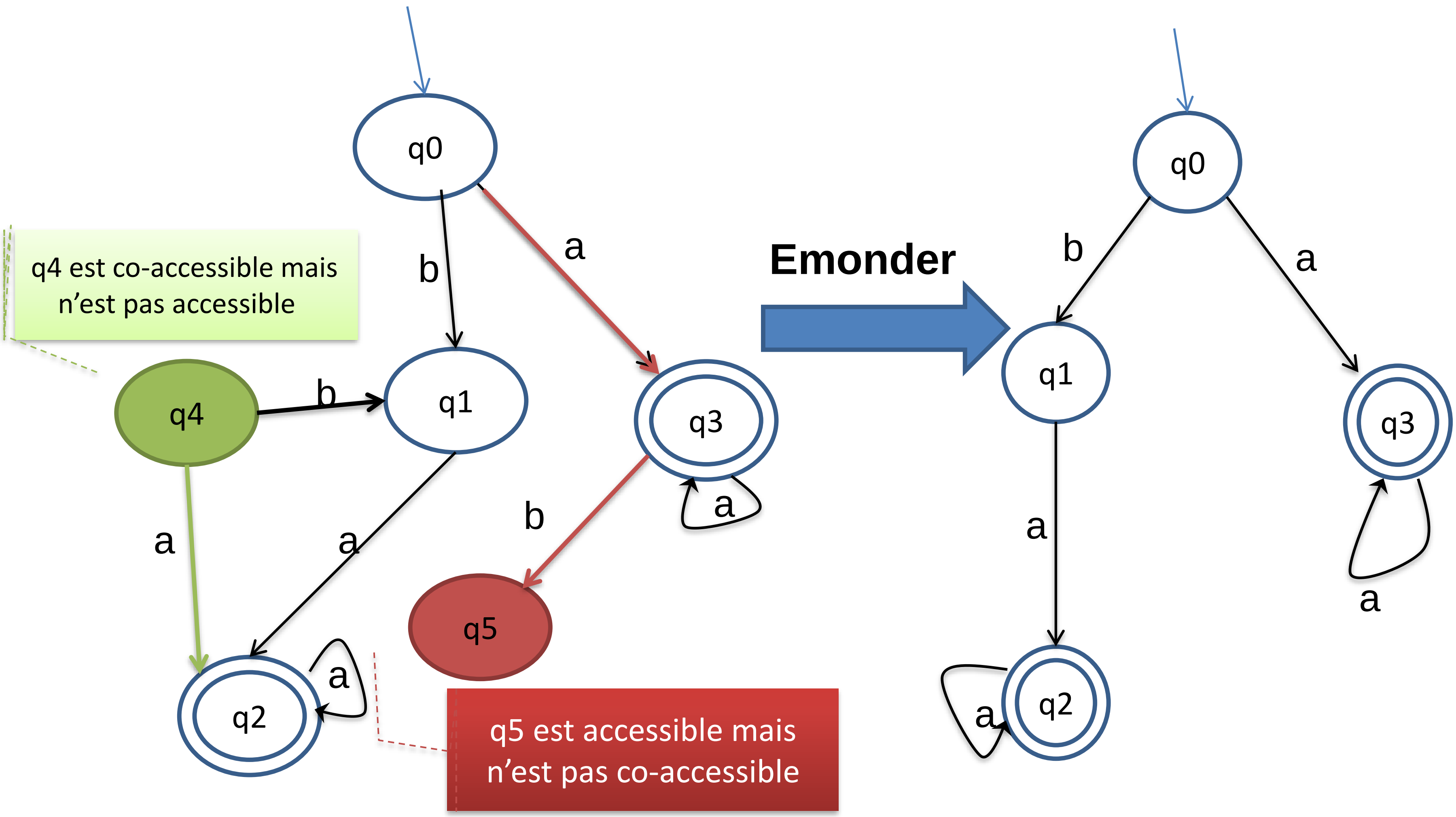
Un état **q** est **co-accessible** s'il existe un chemin de l'état **q** vers un état final.

Un automate est **émondé** si tous ses états sont accessibles et co-accessibles.



AUTOMATES D'ETATS FINIS

Pour rendre un automate émondé, il suffit de supprimer tous les états non accessibles et non co-accessibles.



PASSAGE AEF NON DÉTERMINISTE VERS AEF DÉTERMINISTE

Proposition : Pour tout automate fini non déterministe

A= (X, Q, q_0, **δ,** F) il existe un automate fini déterministe équivalent A_d=(X_d, Q_d, q_{0d}, δ_d, F_d) avec :

- $X_d=\mathbf{X}$
- $Q_d=\mathbf{2^{(Q)}}$ (ou $\mathcal{P}(\mathbf{Q})$)
- $q_{0d}=\mathbf{\{q_0\}}$
- Pour tout état $q_d \in Q_d$ avec $q_d=\{q_1, q_2, \ldots, q_n\}$, on a

  $\delta_d(q_d, a) = \boldsymbol{\delta(q_1, a)} \cup \boldsymbol{\delta(q_2, a)} \cup \boldsymbol{\ldots\ldots} \cup \boldsymbol{\delta(q_n, a),}\ \forall a \in X$
- $F_d=\{\mathbf{q_d} \in \mathbf{Q_d} \,\mathbf{/}\, \mathbf{q_d} \cap \mathbf{F} \neq \emptyset\}$

Les états de Q_d contenant au moins un état final de A sont **finaux.**

Passage AEF non déterministe vers AEF déterministe

Pour déterminiser un automate, il est plus pratique d'établir la table des transitions.

La construction de l'automate déterministe se fait sur des ensembles d'états obtenus à **partir de l'état initial** comme suit :

- A partir de l'ensemble contenant **seulement l'état initial {q0}**, on regroupe les transitions étiquetées par la même lettre issue de q0 :

etat\lettre	a1	……	an
→ q0	δ(q0, a1)		δ(q0, an)

AEF Non déterministe

Il suffit de reprendre (recopier) la ligne de l'état initial.

etat\lettre	a1	……	an
→ {q0}	δ(q0, a1)		δ(q0, an)

AEF déterministe

PASSAGE AEF NON DÉTERMINISTE VERS AEF DÉTERMINISTE

Pour chaque ensemble d'états q_d={q1, …, qn} nouvellement obtenu et donc accessibles à partir de q_{0d}, on détermine les transitions issues de q_d={q1, …, qn} i.e. $\delta_d(q_d, a)$, $\forall a \in X$.

On regroupe les transitions étiquetées par la même lettre :

$$\delta_d(q_d, a) = \delta(q1, a) \cup \ldots\ldots \cup \delta(qn, a)$$

Pour chaque lettre, on fait l'union des cases des différentes lignes associées aux états q1, …, qn.

état\lettre	a	……..
q1	δ(q1, a)	
qn	δ(qn, a)	

AEF non déterministe

état\lettre	a
qd	δ(q1, a) ∪ …∪ δ(qn, a)

AEF déterministe

PASSAGE AEF NON DÉTERMINISTE VERS AEF DÉTERMINISTE

On arrêtera la procédure dés que tous les ensembles d'états obtenus seront traités.

Un ensemble d'états (nouvel état obtenu par regroupement) q_d est final s'il contient un état final (de l'automate initial).

$$F_d = \{q_d \in Q_d \ / \ q_d \cap F \neq \emptyset\}$$

PASSAGE AEF NON DÉTERMINISTE VERS AEF DÉTERMINISTE

AEF non déterministe

etat\lettre	a	b
→ q0	{q1, q3}	{q1}
q1	{q2}	{q1}
q2	{q2}	_
q3	{q3}	_

Etat/Lettre	a	b
→ {q0}	{q1, q3}	{q1}
{q1, q3}	{q2, q3}	{q1}
{q1}	{q2}	{q1}
{q2, q3}	{q2, q3}	-
{q2}	{q2}	-

AEF déterministe

- Recopier la ligne de l'état initial.

- {q1,q3} et {q1} : nouveaux états accessibles

- Pour l'état {q1,q3}, on regroupe les transitions partant de q1 et celles partant de q3 par la même lettre et on obtient {q2,q3}.

- Pour l'état {q1}, il suffit de prendre les transitions de q1.

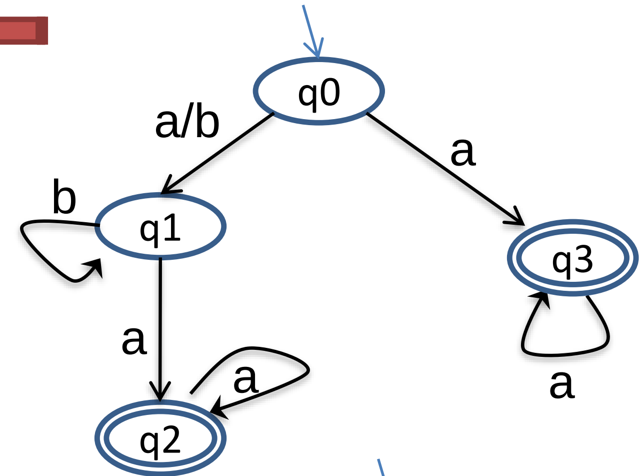
- On traite les nouveaux états accessibles de la même manière

- Les états {q1, q3}, {q2,q3} et {q2} sont finaux.

Passage AEF non déterministe vers AEF déterministe

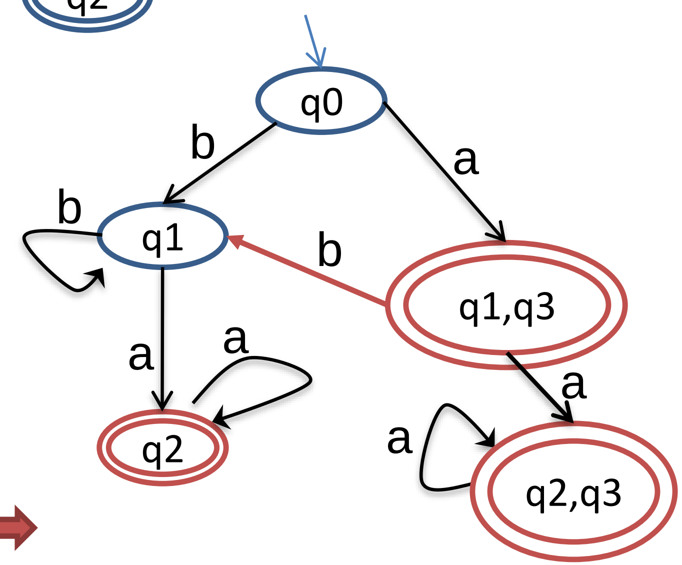
Représentation matricielle

etat\lettre	a	b
q0	{q1, q3}	{q1}
q1	{q2}	{q1}
q2	{q2}	_
q3	{q3}	_



Déterminisation

etat\lettre	a	b
{q0}	{q1, q3}	{q1}
{q1, q3}	{q2, q3}	{q1}
{q1}	{q2}	{q1}
{q2, q3}	{q2,q3}	_
{q2}	{q2}	_



Représentation Graphique