Les Expressions Régulières

S. MAZOUZ & B. LAICHI
FEI-USTHB

Plan

1) Expressions Régulières
   (Définitions, Exemples et Propriétés).

2) Expressions régulières et Automates d'états finis.

3) Applications des expressions régulières.

4) Caractérisation des langages réguliers.

Expressions Régulières

Introduction :

Les langages réguliers sont les langages :

- **générés** par des grammaires de type 3 appelées grammaires régulières.
- **reconnus** par des automates d'états finis. On dit aussi que c'est des langages reconnaissables.

Les mots de tels langages possèdent une forme particulière et peuvent être dénotés par des **expressions régulières (ou expressions rationnelles)**.

Une **expression régulière** est une suite de caractères, appelée motif (ou pattern en anglais), qui décrit ou dénote un ensemble de mots.

Expressions Régulières

Définition : Les expressions régulières sur **un alphabet X** sont définies d'une manière inductive comme suit :

Cas de base :

- ∅ est une expression régulière
  qui **décrit le langage vide**
- **ε** est une expression régulière
  qui **décrit le langage {ε}**
- **a** est une expression régulière ∀**a**∈**X**
  qui **décrit le langage {a}**

Expressions Régulières

Définition (suite)

Cas d'induction :

Si r et s sont deux expressions régulières sur X décrivant respectivement les langages R et S alors :

- **r+s** est une expression régulière
  qui **décrit le langage R∪S**
- **r.s** est une expression régulière
  qui **décrit le langage R.S**
- **r*** est une expression régulière
  qui **décrit le langage R***
- **(r)** est une expression régulière
  qui **décrit le langage R**

Expressions Régulières

Exemples :

- E_1= a+b **dénote le langage {a}∪{b} = {a, b}**
- E_2= a.b **dénote le langage {a}.{b} = {ab}**
- E_3= (a+b).a.b **dénote le langage {a, b}.{ab}= {aab, bab}**
- E_4= a* **dénote le langage {a}* = {a^n /n ≥0}**

Remarque Le symbole de concaténation peut être omis. Par exemple, on peut écrire ab au lieu de a.b

Expressions Régulières

Exemples :

- $E_5 = a^*+b^*$ **dénote le langage $\{a\}^* \cup \{b\}^*$**

- $E_6 = (a+b)^*$ **dénote le langage {a, b}***
  qui correspond à tous les mots sur {a, b}

- $E_7 = (a+b)^*aab$ **dénote tous les mots de {a, b}***
  se terminant par aab

- $E_8 = (a+b)^*aba(a+b)^*$ **dénote tous les mots de {a, b}***
  contenant le facteur aba.

Expressions Régulières

Définition (**Equivalence**)

Deux expressions régulières E_1 et E_2 sont **équivalentes,** notées $E_1 \equiv E_2$, si et seulement si elles **dénotent le même langage**.

Exemple : On veut décrire le langage $\{a, b\}^+$.

E_1= (a+b)(a+b)* /* une lettre a ou b suivie d'une séquence aléatoire de a et b */

E_2= (a+b)*(a+b) /* Une suite aléatoire de a et b suivie par une lettre a ou b */

Dans les deux cas, le langage dénoté est une suite aléatoire de a et b avec au minimum une lettre : a ou b. Donc, $E_1 \equiv E_2$.

Remarque : Pour simplifier les expressions, nous supposons que l'étoile '*' est plus prioritaire que la concaténation '.' qui est plus prioritaire que l'addition '+' : étoile>concaténation>addition

Expressions Régulières

Propriétés sur les expressions régulières :

1. **Commutativité** : $p+q \equiv q+p$
2. **Associativité** : $p+(q+r) \equiv (p+q)+ r$ $\quad p(qr) \equiv (pq)\, r$
3. **Distribution** : $(p+q)r \equiv pr + qr$ $\quad p(q+r) \equiv pq + pr$
4. **Elément neutre** : $p.\varepsilon \equiv \varepsilon.p \equiv p$ $\quad p+\varnothing \equiv \varnothing+p \equiv p$
5. **Elément absorbant** : $p.\varnothing \equiv \varnothing.p \equiv \varnothing$
6. $\varnothing^* \equiv \varepsilon$
7. $(p^*)^* \equiv p^*$
8. $(p^*+q^*)^* \equiv (p^*.q^*)^* \equiv (p+q)^*$
9. $p.\, p^* \equiv p^*.p$
10. $p^* \equiv (p+\varepsilon)^*$

Remarque : $pq \neq qp$

Expressions Régulières

Exemple :

Soient les expressions régulières suivantes :

$E_1=(a^*b^*)^*$ $E_2=(a^*+b^*)^*$ et $E_3=(a+b)^*$

Or on a la propriété $(p^*+q^*)^* \equiv (p^*.q^*)^* \equiv (p+q)^*$

Ces trois expressions sont équivalentes. En effet, elles dénotent le même langage {a, b}*.

Définition : Un langage L sur un alphabet X est un **langage rationnel** si et seulement s'il existe une expression régulière E sur l'alphabet X qui le dénote.

On note **Rat(X*)** la famille des langages rationnels sur X.

Expressions Régulières et Automates d'Etats Finis

Théorème de Kleene :

L'ensemble des langages rationnels (décrits par des expressions régulières) sur un alphabet X est exactement l'ensemble des langages sur X reconnaissables par automate d'états finis.

Nous avons **Rat(X*) = Rec(X*) où**

Rat(X*) est la famille des langages rationnels sur X (tout langage décrit par une expression régulière est un langage rationnel).

Rec(X*) est la famille des langages reconnaissables sur X (tout langage reconnu par un automate d'états finis est un langage reconnaissable).

Expressions Régulières et Automates d'Etats Finis

Proposition :

A toute expression régulière E, il existe un automate d'états fini A(E), reconnaissant le langage dénoté par E.

Les méthodes les plus répandues pour la construction d'un automate à partir d'une expression régulière sont :

- La méthode de Thompson
- La méthode de Glushkov
- La méthode de Brzozowski (méthode des dérivées).

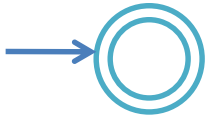
Expressions Régulières et Automates d'Etats Finis

Méthode de Thompson :

L'algorithme consiste à construire l'automate petit à petit, en utilisant des constructions standard pour l'union, la concaténation et l'étoile en se basant sur la structure de l'expression régulière.

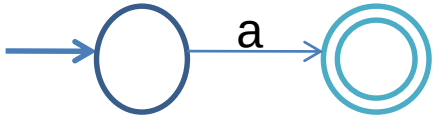
- **L'expression ε**

  on lui associe l'automate :



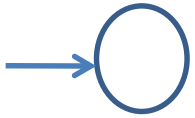
- **L'expression a**

  on lui associe l'automate :



- **L'expression ∅**

  on lui associe l'automate :

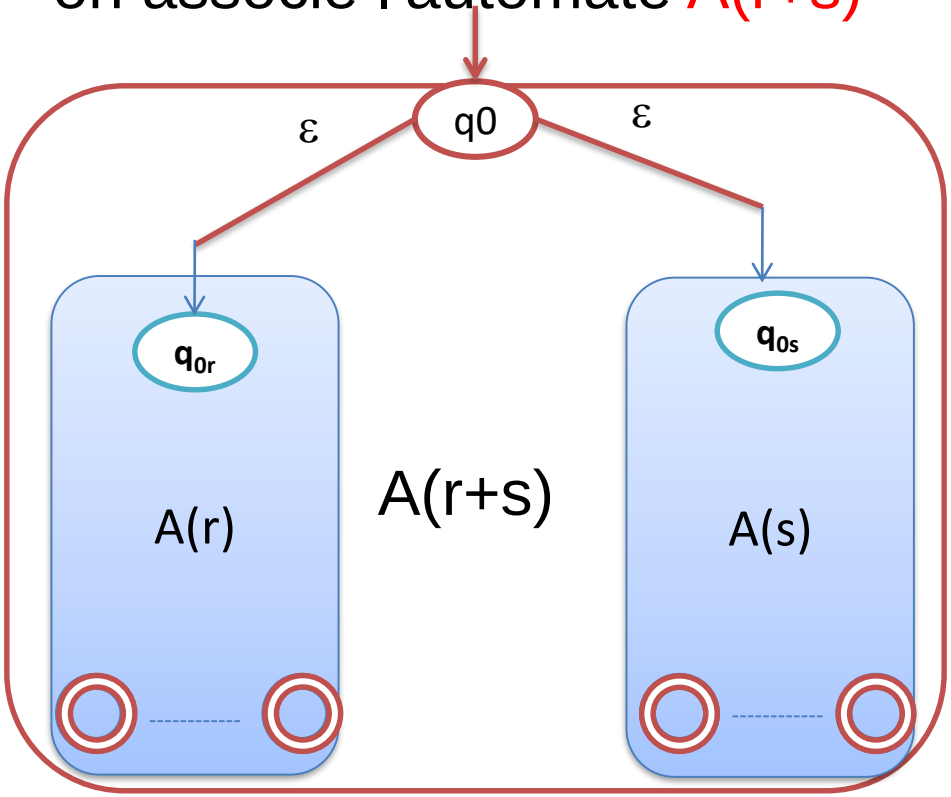


Remarque : Un automate sans état final ne reconnaît aucun mot.

Expressions Régulières et Automates d'Etats Finis

Pour l'expression r+s,

on associe l'automate A(r+s)



Données :

A(r) = (Xr, Qr, q_{0r}, δr, Fr)

A(s) = (Xs, Qs, q_{0s}, δs, Fs)

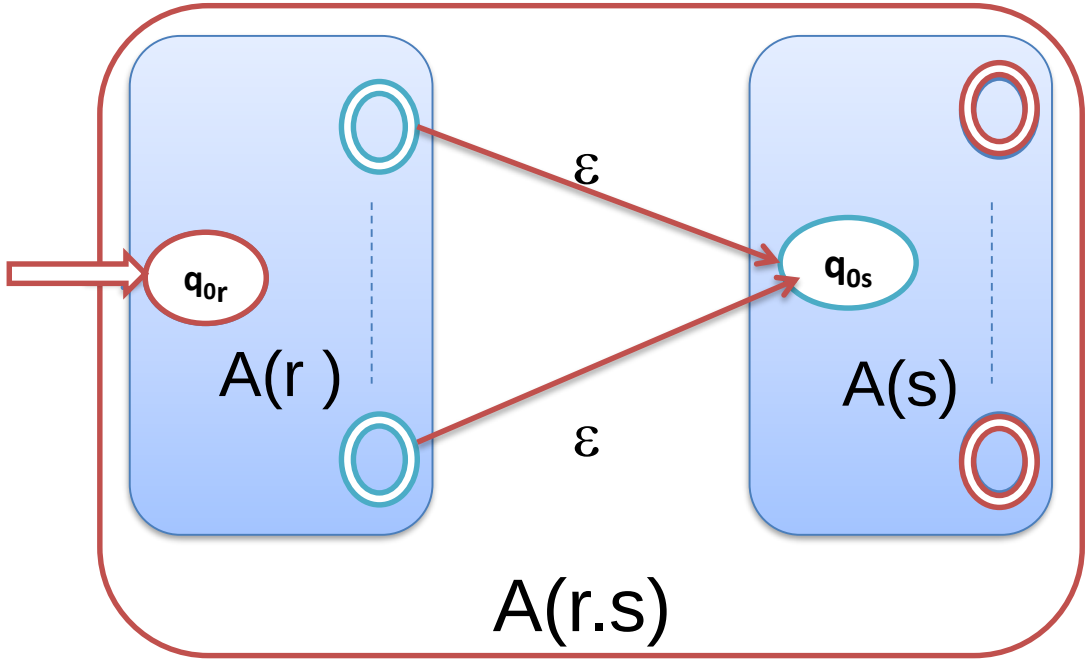
Formellement, on a :

A(r+s)=(X, Q, q_0, δ, F) où :

- $X = Xr \cup Xs$
- $Q = Qr \cup Qs \cup \{q_0\}$
- $q_0 / q_0 \notin (Qr \cup Qs)$
- $\delta = \delta r + \delta s + \delta(q_0, \varepsilon) = \{q_{0r}, q_{0s}\}$
- $F = Fr \cup Fs$

Expression Régulière et Automate d'Etats Finis

- Pour l'expression r.s,
  on associe l'automate A(r.s)



Données :

A(r)=(Xr, Qr, q_{0r}, δr, Fr)

A(s)=(Xs, Qs, q_{0s}, δs, Fs)

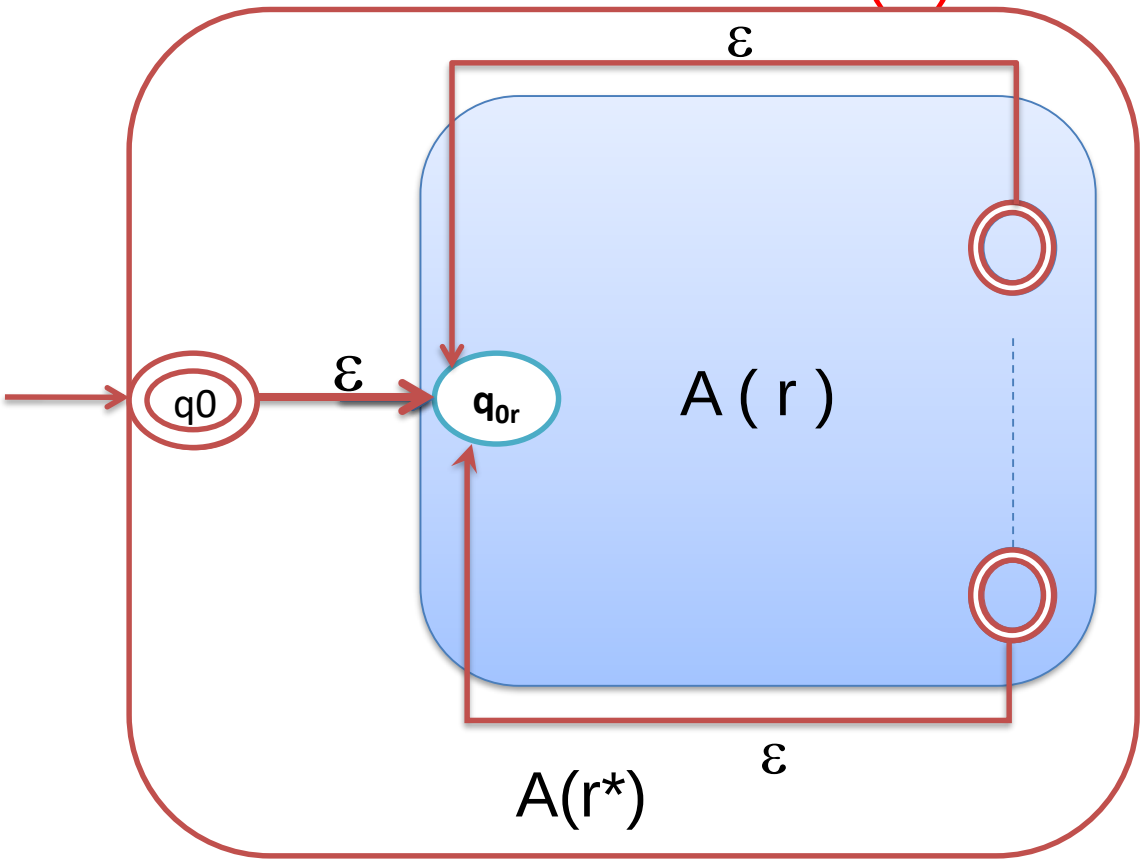
Formellement, on a :

A(r.s)=(X, Q, q_0, δ, F) où

- X = Xr ∪ Xs
- Q = Qr ∪ Qs
- q_0=q_{0r}
- δ = δr + δs +
  $\delta(q, \varepsilon)=q_{0s}\ \forall q \in Fr$
- F = Fs

Expression Régulière et Automate d'Etats Finis

- Pour l'expression r*,
  on associe l'automate A(r*)



Donnée :

A(r)=(Xr, Qr, q_{0r}, δr, Fr)

Formellement, on a :

A(r*)=(X, Q, q_0, δ, F) où

- X = Xr
- Q = Qr ∪ {q_0}
- q_0 / $q_0 \notin$ Qr
- δ = δr + δ(q_0, ε)=q_{0r} +
  $q_{0r} \in \delta(q, \varepsilon)$ $\forall q \in Fr$
- F =Fr ∪ {q_0}

Expressions Régulières et Automates d'Etats Finis

Proposition :

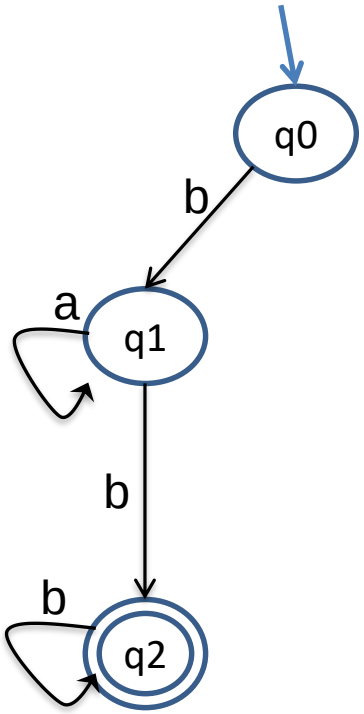
A tout automate d'états fini **A,** il lui correspond une expression régulière qui le dénote.

Le langage reconnu par cet automate est dénoté par l'expression régulière : **ba*bb***

Pour l'obtention d'une expression régulière à partir d'un automate d'états finis, on peut citer les méthodes suivantes :

- La méthode d'élimination ou méthode de Brzozowski et McCluskey
- La méthode par résolution d'équations

Exemple :



Applications des Expressions Régulières

Les expressions régulières ont de nombreuses utilités en informatique, elles servent principalement pour réaliser :

1) des contrôles : vérifier qu'une donnée entrée par un utilisateur a bien le format souhaité.

2) des substitutions : remplacer un motif par une chaine de caractères précise ; par exemple, remplacer les majuscules par des minuscules.

3) des filtres : ne conserver que certaines lignes d'un fichier texte.

4) des découpages : récupérer une partie d'une chaine de caractères par exemple une date placée dans une chaine de caractères.

CARACTÉRISATION DES LANGAGES RÉGULIERS

Les langages réguliers peuvent être **caractérisés** de 3 façons, en utilisant :

1) Les grammaires régulières.
2) Les automates d'états finis.
   (déterministes, non-déterministes ou généralisés).
3) Les expressions régulières.

Remarque :

Pour démontrer qu'un langage est régulier il faut lui trouver : une **grammaire régulière** qui le génère, un **automate d'état finis** qui le reconnait ou une **expression régulière** qui le dénote.

CARACTÉRISATION DES LANGAGES RÉGULIERS

Exemple : Montrer que le langage

L = {w∈{a, b}* / w commence et se termine par la même lettre} est régulier en utilisant les 3 méthodes.

1) L est dénoté par l'expression régulière : a(a+b)*a+b(a+b)*b+a+b

2) L est généré par la grammaire régulière G=(T, N, S, P) avec
T={a, b}, N={S, A, B}, et P est défini par :
S→aA/bB/a/b
A→aA/bA/a
B→aB/bB/b

3) L est reconnu par l'automate d'états fini :