

Informatique Fondamentale

L1 – 2025–2026

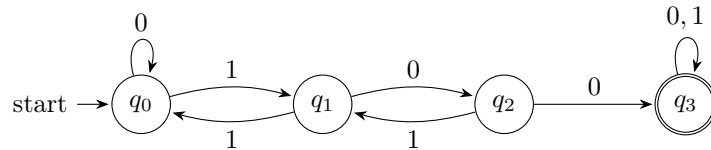
TD : Automates finis

Objectifs du TD :

- Comprendre le fonctionnement d'un AFD
- Savoir concevoir un automate simple
- Maîtriser la détermination (AFN \rightarrow AFD)
- Faire le lien entre expressions régulières et automates

Exercice 1. Simulation d'un AFD

On considère l'automate fini déterministe (AFD) suivant sur l'alphabet $\{0, 1\}$.



1. Compléter le parcours de l'automate pour chaque mot.
2. Indiquer si le mot est accepté ou non.
3. Décrire en français le langage reconnu par cet automate.

Mots à tester :

1. 101
2. 110
3. 1000
4. 1111
5. 0000
6. 1010
7. 1100

Exercice 2. Conception d'un AFD

On souhaite reconnaître les mots sur $\{0, 1\}$ qui se terminent par 01.

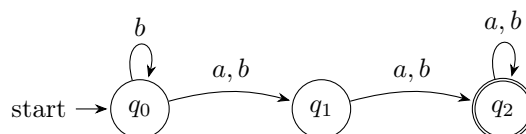
1. Donner quelques exemples de mots acceptés et refusés.
2. Construire un AFD qui reconnaît ce langage.
3. Vérifier votre automate sur les mots suivants :

01, 101, 1001, 10, 111

Exercice 3. Détermination (AFN \rightarrow AFD)

On considère l'AFN suivant :

- États : $\{q_0, q_1, q_2\}$
- Alphabet : $\{a, b\}$
- État initial : q_0
- États accepteurs : $\{q_2\}$



1. Construire les états du nouvel AFD.
2. Donner la table de transition complète.
3. Identifier les états accepteurs.

Exercice 4. Expression régulière \rightarrow AFN

Construire un AFN reconnaissant le langage défini par :

$$(a|b)^*abb$$

1. Décomposer l'expression régulière.
2. Construire l'automate correspondant.
3. Vérifier avec les mots :

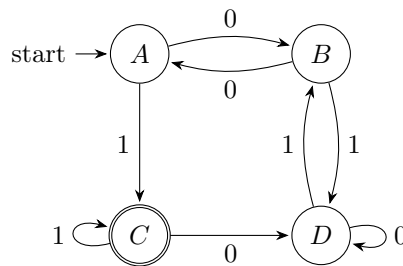
abb, aabb, babb, ab, abba

Indication : il s'agit du langage des mots qui se terminent par *abb*.

Exercice 5. Complément d'un langage

On considère l'AFD suivant :

- États : $\{A, B, C, D\}$
- Alphabet : $\{0, 1\}$
- État initial : A
- États accepteurs : $\{C\}$



1. Construire l'automate qui reconnaît le complément de ce langage.
2. Donner l'ensemble des états accepteurs du nouvel automate.
3. Expliquer pourquoi la méthode fonctionne uniquement si l'automate est complet.

Rappel : ε -fermeture

Dans un automate avec transitions ε , une transition étiquetée ε peut être suivie sans lire de symbole. La ε -fermeture d'un état q , notée ε -fermeture(q), est l'ensemble des états accessibles depuis q en ne suivant que des transitions ε .

- $q \in \varepsilon$ -fermeture(q)
- si $q \xrightarrow{\varepsilon} q'$, alors $q' \in \varepsilon$ -fermeture(q)

Exemple :

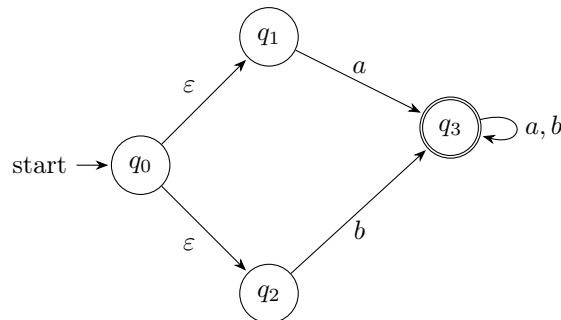
$$q_0 \xrightarrow{\varepsilon} q_1 \xrightarrow{\varepsilon} q_2 \Rightarrow \varepsilon\text{-fermeture}(q_0) = \{q_0, q_1, q_2\}$$

Pour un ensemble S :

$$\varepsilon\text{-fermeture}(S) = \bigcup_{q \in S} \varepsilon\text{-fermeture}(q)$$

Exercice 6. Automate avec transition ε

On considère l'automate non déterministe suivant sur l'alphabet $\{a, b\}$.



1. Calculer la ε -fermeture de q_0 .
2. Déterminer si les mots suivants sont acceptés :

a, b, ab, ba, aa, bb, ε

- Décrire le langage reconnu.
- Transformer cet automate en un automate sans transition ε .

Élimination des transitions ε

Pour supprimer les transitions ε :

- Calculer $E(q) = \varepsilon$ -fermeture(q) pour chaque état.
- Définir :

$$\delta'(q, a) = \varepsilon\text{-fermeture}\left(\bigcup_{p \in E(q)} \delta(p, a)\right)$$

- Un état q est accepteur si $E(q)$ contient un état accepteur.

Exercice 7. Expressions régulières et automates

On considère l'expression régulière suivante sur l'alphabet $\{a, b\}$:

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

- Décrire en français le langage défini par cette expression.
- Donner quelques exemples de mots acceptés et refusés.
- Construire un AFN reconnaissant ce langage.
- Transformer cet AFN en un AFD.

Vérification : tester votre automate sur

$ab, aab, bab, abb, a, b, ba$

Équivalence des expressions régulières

Deux expressions régulières E_1 et E_2 sont équivalentes si :

$$L(E_1) = L(E_2)$$

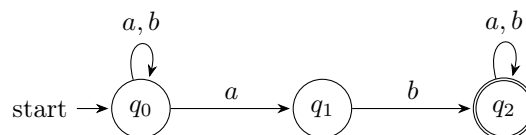
Exemples :

- $a|b = b|a$ (commutativité)
- $(a^*)^* = a^*$
- $(a|b)^*ab = (a^*b^*)^*ab$ (tous les mots sur $\{a, b\}$ qui se terminent par ab)

Conclusion : plusieurs expressions différentes peuvent décrire exactement le même langage.

Exercice 8. Automate vers expression régulière

On considère l'automate suivant :



- Décrire le langage reconnu par cet automate.
- Donner une expression régulière équivalente.

Indication : réfléchir à la position du motif ab dans le mot.

(Optionnel) Donner une autre expression régulière équivalente.

Exercice 9. Équivalence d'expressions régulières

On travaille sur l'alphabet $\{a, b\}$.

1. Dire si les expressions suivantes sont équivalentes. Justifier en décrivant le langage ou en donnant des exemples / contre-exemples.

- $a|b$ et $b|a$
- $(a^*)^*$ et a^*
- $a(a|b)^* | b(a|b)^*$ et $(a|b)(a|b)^*$
- $(ab)^*$ et a^*b^*

2. Simplifier les expressions suivantes :

a) a^*a^*

b) $(a|a)^*$

c) εa^*

3. Donner une expression régulière équivalente à :

a) tous les mots sur $\{a, b\}$ contenant au moins un a

b) tous les mots qui commencent par b

c) tous les mots de longueur au moins 1

4. Trouver deux expressions régulières différentes qui décrivent :

— les mots qui se terminent par a