

1 STABILITÉ PAR INTERSECTION

La classe des langages reconnaissables par automate fini est stable par intersection. Plus précisément, si deux langages L_1 et L_2 sont reconnus respectivement par deux automates finis

$$\mathcal{A}_1 = (Q_1, \delta_1, q_{01}, F_1) \quad \text{et} \quad \mathcal{A}_2 = (Q_2, \delta_2, q_{02}, F_2),$$

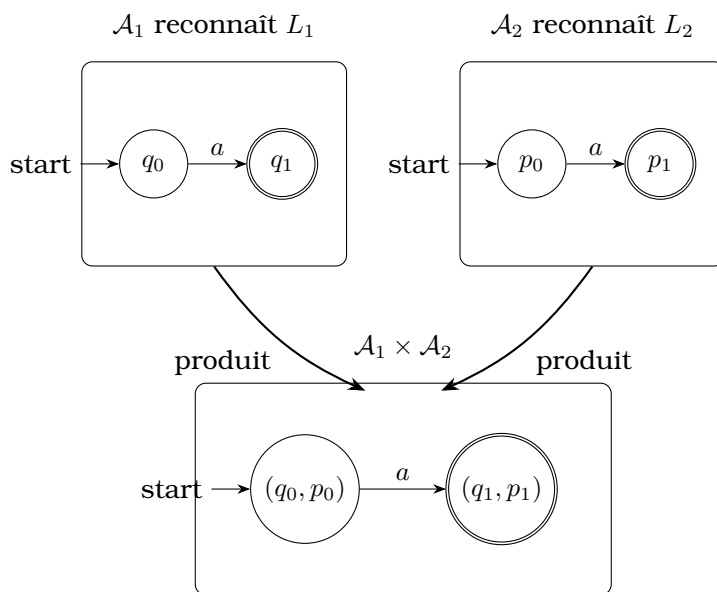
alors le langage $L_1 \cap L_2$ est lui aussi reconnaissable.

Pour le montrer, on construit l'*automate produit*

$$\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2.$$

Ses états sont des couples $(q_1, q_2) \in Q_1 \times Q_2$. Ainsi, l'automate produit simule simultanément les deux automates : à la lecture d'une lettre $a \in \Sigma$, on pose

$$\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a)).$$



Un état est final si ses deux composantes sont finales.

$$L(\mathcal{A}_1 \times \mathcal{A}_2) = L_1 \cap L_2$$

FIGURE 1 – Principe de la construction produit pour l'intersection de deux langages reconnaissables.

En effet, l'état initial de l'automate produit est

$$(q_{01}, q_{02}),$$

et l'ensemble de ses états finaux est

$$F = F_1 \times F_2.$$

Il faut montrer par récurrence sur la longueur du mot le lemme suivant :

$$\forall q_1 \in Q_1, q_2 \in Q_2, w \in \Sigma^*, \delta^*((q_1, q_2), w) = (\delta_1^*(q_1, w), \delta_2^*(q_2, w))$$

Par conséquent, après la lecture d'un mot w , l'automate produit est dans un état final d l'automate produit si et seulement si $\mathcal{A}_1$ et $\mathcal{A}_2$ sont tous les deux dans un état final. On a donc

$$w \in L(\mathcal{A}) \iff w \in L_1 \text{ et } w \in L_2 \iff w \in L_1 \cap L_2.$$

Ainsi,

$$L_1, L_2 \text{ reconnaissables} \implies L_1 \cap L_2 \text{ reconnaissable.}$$

Voici un exemple :

Dans cet automate, E signifie que le nombre de lettres a lues est pair et O qu'il est impair. La lecture d'un a fait donc basculer d'un état à l'autre, tandis que la lecture d'un b ne modifie pas la parité. L'état E est initial et final.

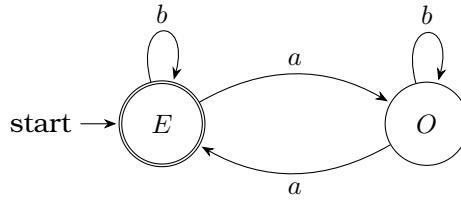


FIGURE 2 – Automate $\mathcal{A}_1$ reconnaissant $L_1 = \{w \in \{a, b\}^* \mid |w|_a \text{ est pair}\}$.

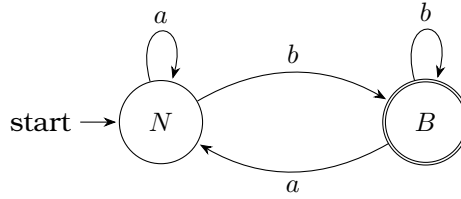


FIGURE 3 – Automate $\mathcal{A}_2$ reconnaissant $L_2 = \{w \in \{a, b\}^* \mid w \text{ se termine par } b\}$.

Ici, l'état N signifie que le mot lu ne se termine pas par b , tandis que l'état B signifie que la dernière lettre lue est b . L'état B est donc le seul état final.

Les deux automates ayant respectivement pour ensembles d'états

$$Q_1 = \{E, O\} \quad \text{et} \quad Q_2 = \{N, B\},$$

l'automate produit possède les quatre états

$$Q_1 \times Q_2 = \{(E, N), (E, B), (O, N), (O, B)\}.$$

Un état du produit est final si et seulement si ses deux composantes sont finales. Comme E est le seul état final de $\mathcal{A}_1$ et B le seul état final de $\mathcal{A}_2$, le seul état final du produit est donc

$$(E, B).$$

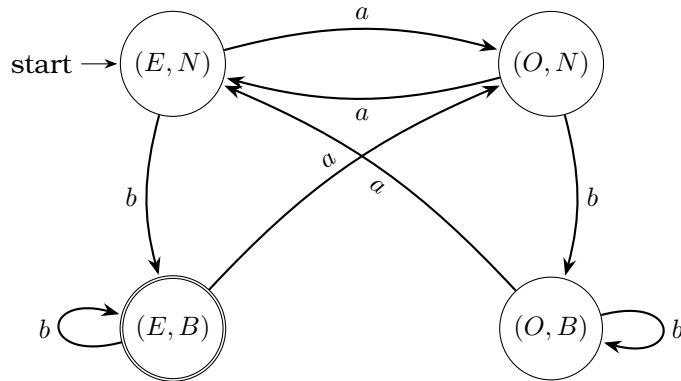


FIGURE 4 – Automate produit $\mathcal{A}_1 \times \mathcal{A}_2$ reconnaissant $L_1 \cap L_2$.

L'état (E, B) est le seul état final de l'automate produit. En effet, atteindre cet état signifie simultanément que :

- le nombre de lettres a lues est pair ;
- la dernière lettre du mot est b .

Ainsi, l'automate produit reconnaît exactement le langage

$$L_1 \cap L_2 = \{w \in \{a, b\}^* \mid |w|_a \text{ est pair et } w \text{ se termine par } b\}.$$

2 STABILITÉ PAR UNION

On a vu que l'union pouvait se montrer très facilement à l'aide des automates finis non déterministes mais on remarque de plus que l'automate produit permet aussi de reconnaître l'union de deux langages reconnus par des automates finis déterministes tout en gardant le caractère déterministe : pour cela la seule chose à

modifier est l'ensemble des états terminaux en prenant l'ensemble des couples où l'une des deux coordonnées est dans F_1 ou F_2 au lieu d'imposer que les deux le soit simultanément.

3 STABILITÉ PAR COMPLÉMENTAIRE

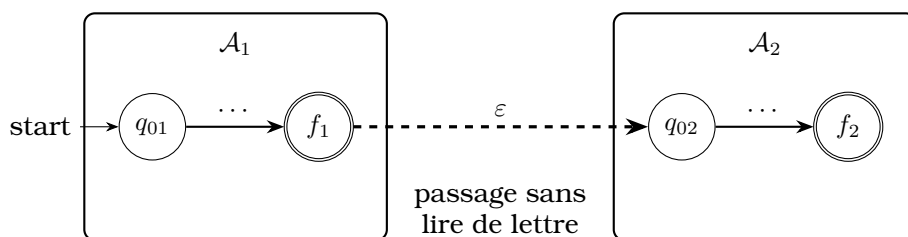
On peut aussi montrer que la famille des langages reconnaissables par automates est stable par complémentaire. En effet, soit $A = (Q, q_0, F, \delta)$ un automate fini déterministe **complet** qui reconnaît un langage L alors $A' = (Q, q_0, Q \setminus F, \delta)$ reconnaît $\Sigma^* \setminus L$. On remarque que l'hypothèse de complétude est ici indispensable car sinon certains mots étaient non acceptés car bloqués dans A et seraient bloqués aussi dans A' et donc non acceptés par ce dernier.

4 STABILITÉ PAR CONCATÉNATION

Soient L_1 et L_2 deux langages reconnaissables, reconnus respectivement par les automates $\mathcal{A}_1$ et $\mathcal{A}_2$. On souhaite construire un automate reconnaissant

$$L_1 L_2 = \{uv \mid u \in L_1, v \in L_2\}.$$

L'idée consiste à faire fonctionner d'abord $\mathcal{A}_1$, puis, dès qu'un état final de $\mathcal{A}_1$ est atteint, à autoriser le passage vers un état initial de $\mathcal{A}_2$ sans lire de lettre. Ce passage est réalisé à l'aide d'une transition ε .



Pour deux automates (pas forcément déterministes)

$$\mathcal{A}_1 = (Q_1, \delta_1, I_1, F_1, \varphi_1) \quad \text{et} \quad \mathcal{A}_2 = (Q_2, \delta_2, I_2, F_2, \varphi_2),$$

on ajoute, pour chaque état final $f \in F_1$ et chaque état initial $i \in I_2$, une transition

$$f \xrightarrow{\varepsilon} i.$$

Les états initiaux du nouvel automate sont ceux de $\mathcal{A}_1$, et ses états finaux sont ceux de $\mathcal{A}_2$.

Un mot est alors accepté ssi sa lecture peut être décomposée en deux parties

$$w = uv,$$

où u conduit $\mathcal{A}_1$ dans un état final et v conduit $\mathcal{A}_2$ dans un état final depuis l'un de ses états initiaux. Autrement dit,

$$u \in L_1 \quad \text{et} \quad v \in L_2.$$

On obtient donc

$$L_1, L_2 \text{ reconnaissables} \implies L_1 L_2 \text{ reconnaissable.}$$

5 STABILITÉ PAR ÉTOILE

Soit maintenant L un langage reconnaissable par un automate fini déterministe (pour plus de simplicité) $\mathcal{A} = (Q, q_0, F, \delta)$.

Pour construire un automate reconnaissant L^* , on ajoute un nouvel état s , qui est à la fois initial et final. Une transition ε permet d'entrer dans l'automate $\mathcal{A}$. Depuis chaque état final de $\mathcal{A}$, une autre transition ε permet de revenir à son état initial afin de commencer la lecture d'un nouveau mot de L .

Le rôle des deux types de transitions ajoutées est important :

- le nouvel état initial s est également final, ce qui permet d'accepter le mot vide ε ;
- la transition

$$s \xrightarrow{\varepsilon} q_0$$

permet de commencer la lecture d'un premier mot de L ;

- la transition d'un état final vers q_0 permet, après avoir reconnu un mot de L , d'en commencer immédiatement un autre.

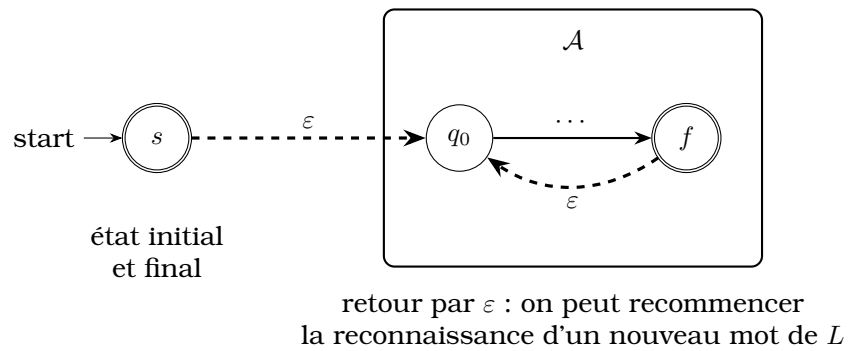


FIGURE 5 – Construction d'un automate reconnaissant $L(\mathcal{A})^*$.

Ainsi, l'automate peut reconnaître successivement

$$u_1, \quad u_1 u_2, \quad u_1 u_2 u_3, \quad \dots \quad (u_i \in L),$$

mais également ε . Il reconnaît donc exactement L^* .

Par conséquent,

$$\boxed{L \text{ reconnaissable} \implies L^* \text{ reconnaissable.}}$$