Chapitre 4. Automates.

1 Automates finis déterministes

1.1 Introduction et exemples

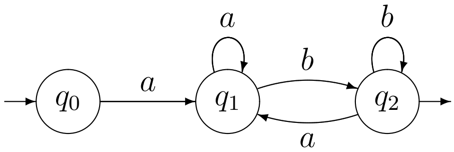
Nous avons déjà considéré dans le chapitre précédents des automates sans en donner une définition précise.

De manière informelle, un automate est une machine abstraite qui peut prendre un nombre fini d'*états*, qui reçoit en entrée un mot écrit sur un alphabet Σ et qui change d'état à la lecture des lettres de ce mot.

On sélectionne au préalable un état particulier appelé état *initial* et un certain nombre d'états qui sont qualifiés d'*acceptants* ou de *finaux*.
La lecture des mots se fait en partant de l'état initial. À la fin de la lecture du mot, si l'état dans lequel se trouve l'automate est un état acceptants, on dira que le mot est *accepté* ou *reconnu* et sinon, on dira qu'il est *rejeté*.

On représentera les automates sous la forme d'un graphe valué dont les sommets sont les différents états possibles de l'automate, et les arêtes représentent les transitions entre les états et sont étiquetées par des lettres de Σ.

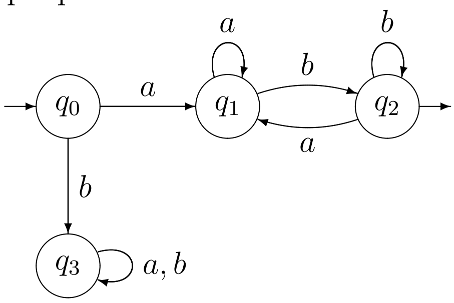
Considérons par exemple l'automate représenté ci-dessous, sur l'alphabet $\Sigma = \{a, b\}$.



L'état initial (ici q_0) est désigné par une flèche entrante, les états acceptants (ici q_2) sont représentés par une flèche sortante. [1].

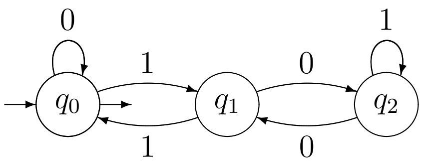
On se convaincra aisément que pour qu'un mot puisse être lu par cet automate, il faut qu'il commence par un a, et qu'alors la lecture du mot se termine à l'état q_1 si la dernière lettre du mot est un a et à l'état q_2 si la dernière lettre du mot est un b : on en déduit que les mots acceptés par cet automate sont les mots commençant par a et dont la dernière lettre est un b. On dit que le *langage reconnu* par cet automate est $a\Sigma^*b$.

Faute de transition partant de q_0 et étiquetée par b, un mot débutant par un b ne peut être lu par l'automate précédent. On dit que (q_0, b) est un *blocage* de l'automate. un automate sans blocage est dit *complet*. Il est toujours possible de rendre complet un automate présentant des blocages en rajoutant un état vers lequel aboutissent tous les blocages, état qu'on ne peut ensuite plus quitter. Cet état est appelé un *puits*. Le nouvel automate est alors complet, il permet de lire tous les mots de Σ^* et les mots reconnus sont les mêmes que pour l'automate initial. Sur notre exemple, l'automate obtenu est alors :



1. certains auteurs représentent les états acceptants en les entourant d'un double cercle

Exercice : On considère l'automate sur l'alphabet $\Sigma = \{0, 1\}$ représenté par :



Les mots sur l'alphabet $\Sigma = \{0, 1\}$ seront identifiés aux entiers naturels via leur écriture en base 2.
a) Pour les entiers de $[\![0, 8]\!]$, indiquer l'état auquel on aboutit à la fin de la lecture du mot associé.
b) Déterminer, en justifiant, le langage reconnu par l'automate précédent.

1.2 Définitions

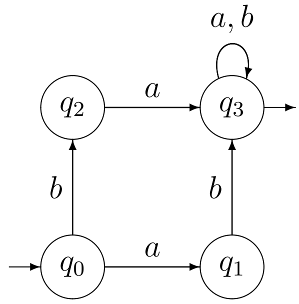
Définition 1. *Un automate fini déterministe (AFD ou DFA pour deterministic finite automaton) est un quintuplet $A = (\Sigma, Q, q_0, F, \delta)$ où :*
- *Σ est un alphabet (fini) ;*
- *Q est un ensemble fini dont les éléments sont appelés les* états *de A ;*
- *q_0 est un élément de Q appelé état* initial *;*
- *F est une partie de Q dont les éléments sont appelés les états* acceptants *ou* finaux *;*
- *δ est une application d'une partie de $Q \times \Sigma$ dans Q, appelée* fonction de transition

Lorsque δ est définie sur $Q \times \Sigma$ tout entier, l'automate A est dit complet*.*
Si δ n'est pas définie en $(q, a) \in Q \times \Sigma$, on dit que (q, a) est un blocage *de A.*

Exemple : Considérons l'automate A_0 défini sur $\Sigma = \{a, b\}$ à 3 états nommés q_0, q_1, q_2 avec comme état initial q_0 et comme états acceptants $F = \{q_3\}$ et la fonction de transition définie par le tableau

δ	a	b
q_0	q_1	q_2
q_1	$-$	q_3
q_2	q_3	$-$
q_3	q_3	q_3

. A_0 peut-être représenté par le graphe :



Notations : Si $\delta(q_i, a) = q_j$, on note encore $q_i.a = q_j$ et cette transition est représentée par $q_i \xrightarrow{a} q_j$.

Définition 2. *Si $A = (\Sigma, Q, q_0, F, \delta)$ est un AFD, on appelle* fonction de transition étendue aux mots *la fonction δ^* définie récursivement sur $Q \times \Sigma^*$ par :*
- $\forall\, q \in Q,\ \delta^*(q, \varepsilon) = q$
- $\forall (q, m, a) \in Q \times \Sigma^* \times \Sigma,\ \delta^*(q, ma) = \delta(\delta^*(q, m), a)$

Si $\delta^*(q, m) = q'$ on note encore $q.m = q'$.

Remarque : Si l'automate A présente des blocages, la fonction δ^* n'est définie que sur une partie de $Q \times \Sigma^*$. Plus précisément, si $m = a_1 a_2 \ldots a_n$ avec les a_i dans Σ et $q \in Q$, $\delta^*(q, m)$ est défini s'il existe une succession de transitions $q \xrightarrow{a_1} q_{i_1} \xrightarrow{a_2} \ldots \xrightarrow{a_n} q_{i_n}$ débutant en q et on a alors $\delta^*(q, m) = q_{i_n}$. Autrement dit, $\delta^*(q, m)$ est l'unique état (s'il n'y a pas blocage) auquel on aboutit quand on lit le mot m à partir de l'état q.

Définition 3. $\diamond$ *Soit un automate $A = (\Sigma, Q, q_0, F, \delta)$ fini déterministe et δ^* la fonction de transition étendue*

- *Un mot $m \in \Sigma^*$ est dit* reconnu *par A si $\delta^*(q_0, m) \in F$*
- *Le* langage reconnu *par A, noté $\mathscr{L}(A)$ est l'ensemble des mots reconnus par A.*

$\diamond$ *Un langage L sur l'alphabet Σ est dit reconnaissable s'il existe un AFD A sur Σ tel que $L = \mathscr{L}(A)$. On note $Rec(\Sigma)$ l'ensemble des langages reconnaissables sur Σ.*