

Les automates représentent des objets mathématiques et non pas des objets physiques, ils peuvent donc être définis et non pas construits. Les automates effectuent leurs calculs grâce à deux classes d'objets qu'ils manipulent : la classe des entrées que l'automate accepte et la classe des sorties que l'automate produit. Le rapport entre les éléments de l'entrée et les éléments de la sortie est déterminé par la structure de l'automate.

6.1. Automates finis déterministes

Les automates finis déterministes sont les plus simples (au sens moins sophistiqués) des automates.

6.1.1. Définition

Un automate fini² déterministe (AFD) est un quintuplet $A = (E, V, \tau, e_0, F)$ où :

- E est l'ensemble fini d'états ;
- V est un vocabulaire (d'entrée) fini ;
- $\tau : E \times V \longrightarrow E$ est une fonction dite de transition ;
- $e_0 \in E$ est l'état initial ;
- $F \subset E$ est l'ensemble des états terminaux (ou états d'acceptation).

Concrètement, on peut représenter un automate fini déterministe comme une unité mécanique possédant une tête de lecture posée sur un ruban se déplaçant dans un seul sens. En cours de fonctionnement, l'automate est dans l'état $e \in E$ et sa tête de lecture est située devant le symbole $a \in V$. Il passe alors dans l'état $\tau(e, a)$ et la tête de lecture avance d'une case vers la droite (ou bien le ruban se déplace d'une case vers la gauche). Ainsi, à l'état initial un automate devant un mot $a_1 a_2 \dots a_n$ peut être représenté comme sur la figure 6.1. La tête de lecture se trouve alors sur le symbole du ruban situé le plus à gauche.

Si, après avoir parcouru complètement le mot de départ $w = a_1 a_2 \dots a_n$, l'automate se trouve dans un des états terminaux, on dit alors que w a été reconnu ou accepté par l'automate.

NOTE.– *Un automate fini déterministe ne peut s'arrêter avant d'avoir parcouru entièrement le mot en entrée. En effet, τ est toujours complètement définie.*

Il existe deux principales façons de représenter une fonction de transition :

2. On parle aussi d'automate déterministe à un nombre fini d'états.

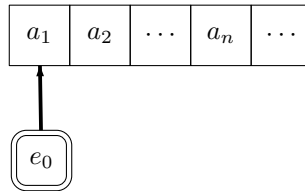


Figure 6.1. Représentation d'un automate fini déterministe à l'état initial

- ① une table de transition qui est un tableau représentant complètement τ . Les lignes de ce tableau correspondent aux différents états de l'automate alors que les colonnes correspondent aux données d'entrée possibles. Le contenu du tableau est alors le résultat de l'application τ ;
- ② un diagramme de transition qui est un graphe défini de la façon suivante :
 - il existe un nœud associé à chacun des états de l'automate ;
 - pour tout état $q \in E$ et pour tout symbole $a \in V$, soit $p = \tau(q, a)$. Le diagramme possède alors un arc de q à p étiqueté par a . Si plusieurs symboles conduisent à p au départ de q , on peut alors n'avoir qu'un arc (étiqueté par la liste des symboles concernés) ;
 - e_0 est marqué par un arc entrant sans origine ;
 - les nœuds correspondant à des états terminaux sont doublement cerclés.

☞ EXEMPLE.— Soit $A = (\{e_0, e_1, e_2, e_3\}, \{0, 1\}, \tau, e_0, \{e_0\})$. τ est représentée par la table de transition suivante :

	0	1
e ₀	e ₁	e ₂
e ₁	e ₀	e ₃
e ₂	e ₃	e ₀
e ₃	e ₂	e ₁

Cette table de transition est représentée de manière totalement équivalente par le diagramme de la figure 6.2. La figure 6.3 représente les trois premiers états successifs de l'automate placé devant le mot 111010. D'une manière plus compacte, on notera les états successifs de la manière suivante : $e_0111010$, $1e_211010$, $11e_01010$, $111e_2010$, $1110e_310$, $11101e_10$ et $111010e_0$. L'automate étant dans un état terminal en fin de mot, le mot a été reconnu par l'automate.

☛ EXERCICE 6.1. – Donner les états successifs de l'automate de l'exemple précédent devant le mot 1101 ? Le mot est-il reconnu par l'automate ?

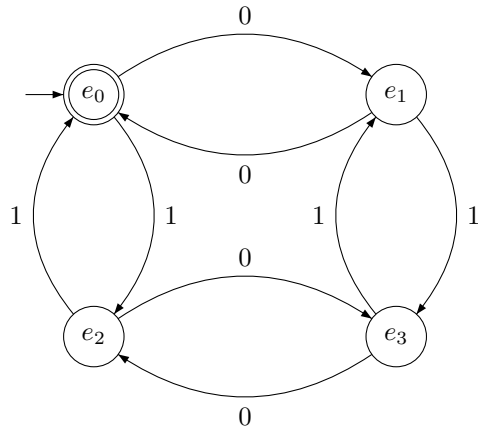


Figure 6.2. Diagramme de transition associé à un automate

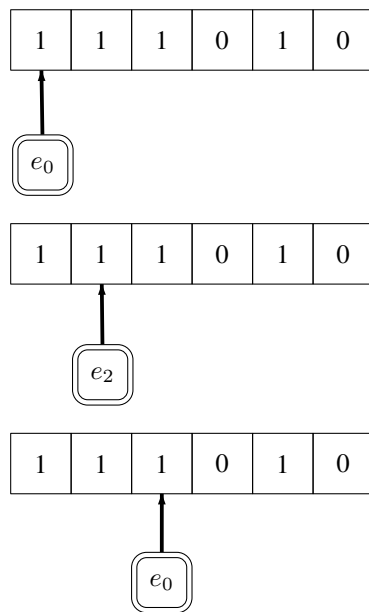


Figure 6.3. Comportement d'un automate devant un mot donné

6.1.2. Langage reconnu par un automate

À partir de τ , on définit une application $\hat{\tau}$ de $E \times V^*$ dans E telle que :

- $\forall e \in E \quad \hat{\tau}(e, \epsilon) = e$;
- $\forall e \in E, \forall y \in V, \forall x \in V^* \quad \hat{\tau}(e, xy) = \tau(\hat{\tau}(e, x), y)$.

$\hat{\tau}$ est donc un prolongement de τ . Par définition, le langage reconnu ou accepté par l'automate A est : $T(A) = \{x \in V^* \mid \hat{\tau}(e_0, x) \in F\}$.

☞ EXEMPLE.— Soit $A = (\{e_0, e_1, e_2\}, \{0, 1\}, \tau, e_0, \{e_0, e_1\})$ représenté par le diagramme de la figure 6.4. On peut déterminer $T(A)$ en étudiant les deux classes $cl(e_0)$ et $cl(e_1)$, ensembles des mots commençant en e_0 et terminant en e_0 et e_1 respectivement. $cl(e_0) = 0^*$ et $cl(e_1) = 0^*1^+$. Ainsi, $T(A) = cl(e_0) \cup cl(e_1) = 0^*1^* = \{0^m 1^n \mid (m, n) \in \mathbb{N} \times \mathbb{N}\}$.

Si on désigne par τ_{e_0} l'application de V^* dans E définie par : $\tau_{e_0}(x) = \hat{\tau}(e_0, x)$, on peut poser : $T(A) = \tau_{e_0}^{-1}(F)$.

Un langage reconnu par un automate fini déterministe est dit régulier.

► EXERCICE 6.2. – Montrer que, pour l'automate de l'exercice 6.1.³, $T(A) = \{x \in \{0, 1\}^* \mid \mu_0(x) \in \{00\}^* \wedge \mu_1(x) \in \{11\}^*\}$. Comment peut-on décrire plus naturellement les éléments du langage accepté par l'automate A ?

► EXERCICE 6.3. – Soit l'automate $A = (\{e_0, e_1, e_2, e_3, e_4, e_5, e_6\}, \{0, 1\}, \tau, e_0, \{e_2, e_3, e_4\})$. τ est définie dans le tableau suivant :

τ	0	1
e_0	e_2	e_1
e_1	e_3	e_0
e_2	e_5	e_4
e_3	e_5	e_4
e_4	e_5	e_4
e_5	e_5	e_5

- ❶ Représenter A
- ❷ Pour $i \in [5]$, déterminer $c_i = \{x \in \{0, 1\}^* \mid \tau(e_0, x) = e_i\}$.
- ❸ Déterminer $T(A)$

3. μ_x est désigné le morphisme d'effacement qui ne conserve que les occurrences de x , c'est-à-dire $\forall y \neq x, \mu_x(y) = \epsilon \wedge \mu_x(x) = x$.

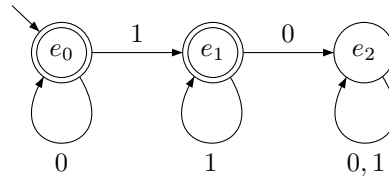


Figure 6.4. Diagramme d'un automate

- EXERCICE 6.4. – Soit $A = (\{e_0, e_1, e_2, e_3, e_4\}, \{a, b\}, \tau, e_0, \{e_2\})$. τ est représentée par la table de transition suivante :

	a	b
e_0	e_1	e_4
e_1	e_4	e_2
e_2	e_3	e_4
e_3	e_4	e_2
e_4	e_4	e_4

Donner le diagramme de l'automate. Quel est le rôle de e_4 . Donner un diagramme simplifié.

- EXERCICE 6.5. – Quel langage permet de reconnaître l'automate suivant, décrit complètement par sa table de transition (l'état de départ est marqué par \rightarrow et l'état d'acceptation, par \star) :

	0	1
$\rightarrow e_0$	e_0	e_1
$\star e_1$	e_1	e_0

- EXERCICE 6.6. – Quel langage permet de reconnaître l'automate suivant, décrit complètement par sa table de transition (l'état de départ est marqué par \rightarrow et les états d'acceptation, par \star) :

	0	1
$\rightarrow \star e_0$	e_1	e_0
$\star e_1$	e_2	e_0
e_2	e_2	e_2

- EXERCICE 6.7. – Donner, pour chacun des langages suivants définis sur $\{0, 1\}$, un automate fini déterministe le reconnaissant :

- ❶ l'ensemble des mots finissant en 00
- ❷ l'ensemble des mots contenant la sous-chaîne 011
- ❸ l'ensemble des mots qui finissent ou commencent par 01