

# Machines de Turing

Si votre navigateur affiche mal les symboles spéciaux de HTML, voici une [version sans symbole spécial](#).

## Introduction

Une *machine de Turing* se compose d'une partie de contrôle et d'une bande infinie sur laquelle se trouvent écrits des symboles. La partie de contrôle est constituée d'un nombre fini d'états possibles et de transitions qui régissent les calculs de la machine. Les symboles de la bande sont lus et écrits par l'intermédiaire d'une *tête de lecture*.

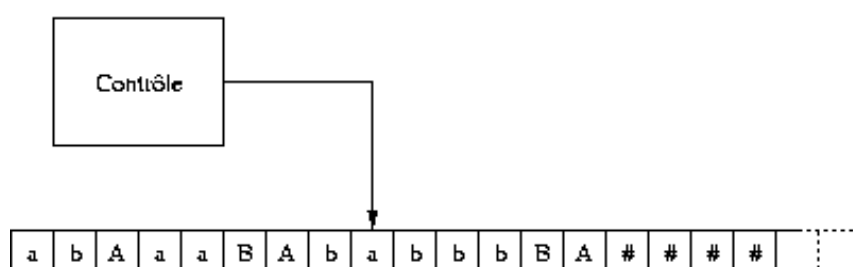


Fig. 1 : machine de Turing

Les machines de Turing sont une abstraction des ordinateurs. La partie de contrôle représente le microprocesseur. Un élément essentiel est que le nombre d'états est *fini*. Ceci prend en compte que les microprocesseurs possèdent un nombre déterminé de registres d'une taille fixe et que le nombre de configurations possibles est fini. La bande représente la mémoire de l'ordinateur. Ceci comprend la mémoire centrale ainsi que les mémoires externes telles les disques durs. La tête de lecture représente la bus qui relie le microprocesseur à la mémoire. Contrairement à un ordinateur, la mémoire d'une machine de Turing est infinie. Ceci prend en compte qu'on peut ajouter des disques durs à un ordinateur de façon (presque) illimitée. Une autre différence entre une machine de Turing et un ordinateur est que l'ordinateur peut accéder à la mémoire de manière directe (appelée aussi aléatoire) alors que la tête de lecture de la machine de Turing se déplace que d'une position à chaque opération.

Le contrôle de la machine est constitué d'un nombre fini d'états  $\{ q_0, \dots, q_n \}$ . À chaque instant, la machine se trouve dans un de ces états. Au départ, la machine se trouve dans l'état  $q_0$  qu'on appelle *état initial*. Le calcul d'une machine de Turing est formé d'une suite d'*étapes de calcul* qui sont effectuées par la machine. Chaque étape consiste à changer l'état de contrôle, écrire un symbole sous la tête de lecture et déplacer la tête de lecture. Les étapes de calcul possibles sont décrites par les transitions de la machine.

## Définition

Nous donnons maintenant la définition précise d'une machine de Turing. De manière formelle, une machine de Turing  $M$  est un septuplet  $(Q, \Sigma, \Gamma, E, q_0, F, \#)$  où

$Q$

est l'ensemble des *états de contrôle*. C'est un ensemble *fini*  $\{q_0, \dots, q_n\}$ .

$\Sigma$

est l'*alphabet d'entrée*. C'est un ensemble *fini* de symboles qui ne contient pas le symbole blanc  $\#$ . Cet alphabet est utilisé pour écrire la donnée initiale sur la bande.

$\Gamma$

est l'*alphabet de bande*. C'est un ensemble *fini* qui comprend tous les symboles qui peuvent être écrits sur la bande. Ceci inclut bien sûr l'alphabet d'entrée  $\Sigma$  et le symbole blanc  $\#$ .

$E$

est un ensemble fini de *transitions* de la forme  $(p, a, q, b, x)$  où  $p$  et  $q$  sont des états,  $a$  et  $b$  sont des symboles de bande et  $x$  est un élément de  $\{L, R\}$ . Une transition  $(p, a, q, b, x)$  est aussi notée  $p, a \rightarrow q, b, x$ .

$q_0$

est l'*état initial*. C'est un état particulier de  $Q$  dans lequel se trouve machine au début d'un calcul.

$F$

est l'ensemble des *états finaux* appelés aussi *états d'acceptation*

$\#$

est le *symbole blanc* qui, au départ, remplit toutes les positions de la bande autres que celles contenant la donnée initiale.

Une machine de Turing est *déterministe* si pour tout état  $p$  et tout symbole  $a$ , il existe au plus une transition de la forme  $p, a \rightarrow q, b, x$ . Lorsque la machine est déterministe, l'ensemble  $E$  de transitions est aussi appelé *fonction de transition* et est noté  $\delta$ . La fonction  $\delta$  associe à chaque paire  $(p, a)$  l'unique triplet  $(q, b, x)$ , s'il existe, tel que  $p, a \rightarrow q, b, x$  soit une transition. Lorsque la machine n'est pas déterministe, l'ensemble  $E$  peut encore être vu comme une fonction  $\delta$ . Dans ce cas, la fonction  $\delta$  associe à chaque paire  $(p, a)$  l'ensemble des triplets  $(q, b, x)$  tels que  $p, a \rightarrow q, b, x$  soit une transition.

À la place de  $L$  et de  $R$ , on utilise parfois les flèches  $\leftarrow$  et  $\rightarrow$  pour désigner les déplacements de la tête de lecture. C'est en particulier le cas sur certaines figures.

## Description

Une machine de Turing est décrite en donnant explicitement les sept éléments  $Q, \Sigma, \Gamma, E, q_0, F$  et  $\#$ . Afin de rendre plus simple l'énumération des transitions de  $E$ , celles-ci sont souvent données sous la forme d'un tableau ou sous la forme d'un diagramme similaire à celui d'un automate.

Lorsque  $E$  est décrit sous la forme d'un tableau, ce tableau possède une ligne pour chaque état et une colonne pour chaque symbole. La case à l'intersection de la ligne de l'état  $p$  et de la colonne du symbole  $a$  contient tous les triplets  $(q, b, x)$  tel que  $p, a \rightarrow q, b, x$  est une transition. Si la machine est déterministe, chaque case du tableau contient au plus un triplet. Lorsque  $E$  est décrit sous la forme d'un diagramme, ce dernier représente un graphe dont les sommets sont les états de la machine. Le graphe possède une arête de l'état  $p$  à l'état  $q$  étiquetée par  $a, b, x$  si  $p, a \rightarrow q, b, x$  est une transition de la machine.

Pour illustrer ces deux méthodes, nous allons décrire la même machine très simple à l'aide des deux méthodes. Soit  $M$  la machine  $(\{0\}, \{a, b\}, \{a, b, \#\}, E, 0, \{0\}, \#)$ . La machine  $M$  possède donc un unique état  $0$  qui est initial et final. Son alphabet d'entrée  $\Sigma$  contient les deux symboles  $a$  et  $b$  et son alphabet de bande  $\Gamma$  contient en plus le symbole blanc  $\#$ . L'ensemble  $E$  contient les trois transitions

$$0, a \rightarrow 0, \#, R \quad 0, b \rightarrow 0, \#, R \quad 0, \# \rightarrow 0, \#, L.$$

Ces trois transitions sont aussi décrites par le tableau suivant. Ce tableau a une seule ligne qui correspond à l'unique état et trois colonnes qui correspondent à chacun des symboles de l'alphabet de bande.

E	a	b	#
0	0, #, R	0, #, R	0, #, L

Tableau des transitions.

Ces trois transitions sont aussi décrites par le diagramme de la figure ci-dessous. Les petites flèches entrante et sortante de l'état  $0$  marquent qu'il est respectivement initial et final.

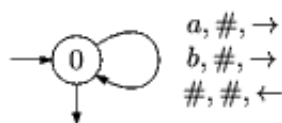


Fig. 2 : Diagramme des transitions

## Configurations et calculs

Le principe global de fonctionnement d'une machine de Turing est le suivant. Une entrée, c'est-à-dire un mot fini sur l'alphabet d'entrée, est écrit sur la bande la machine. Le reste de la bande est rempli avec le symbole blanc. La tête de lecture est placée sur le premier symbole du mot d'entrée. Ensuite, la machine commence son calcul jusqu'à arriver éventuellement dans un état où elle accepte l'entrée.

Pour décrire le fonctionnement précis d'une machine de Turing, il est nécessaire d'introduire la

notion de configuration. Une *configuration* d'une machine de Turing est l'état *global* de la machine à un instant donné. Elle comprend

1. l'état de contrôle qui est un élément de  $Q$ ,
2. le contenu de la bande,
3. la position de la tête de lecture sur la bande.

Si la machine se trouve dans un état  $q$ , la configuration est écrite  $uqv$  où  $u$  est le contenu de la bande (strictement) à gauche de la tête de lecture et  $v$  est le contenu de la tête à droite de la tête de lecture (cf. la figure ci-dessous). Le symbole sous la tête de lecture est donc le premier symbole de  $v$ . Quand on décrit une configuration de la machine, les symboles blancs qui se trouvent dans la partie infinie à droite de la bande sont omis. Ceci permet de décrire une configuration de façon finie. Si par exemple, la machine de la figure 1 se trouve dans l'état  $q$ , sa configuration est  $abAaaBAbqabbbbBA$  parce que le contenu de la bande avant la tête de lecture est  $abAaaBAb$  et que le contenu après la tête de lecture est  $abbbbBA$  suivi de symboles blancs. Tous les symboles  $\#$  à droite sont sous-entendus et ne sont pas écrits.

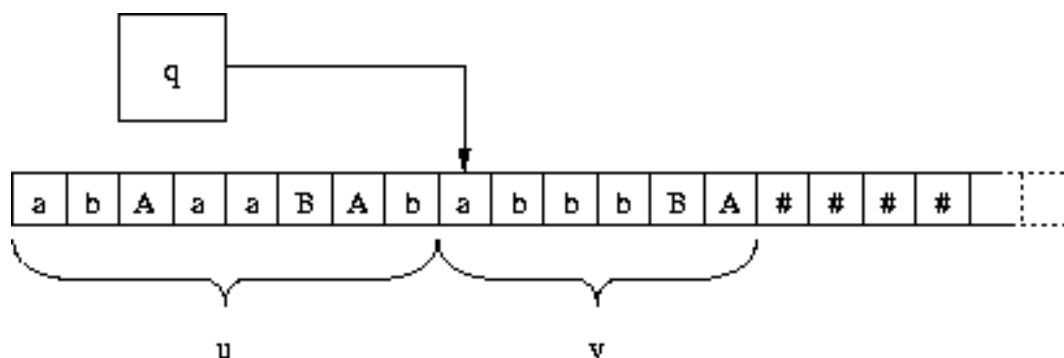


Fig. 3 : Configuration  $abAaaBAbqabbbbBA$

Au départ, la bande contient la donnée initiale et la tête de lecture se trouve sur la première position de la bande. La *configuration initiale* s'écrit donc  $q_0w$  où  $q_0$  est l'état initial et  $w$  est la donnée initiale.

Un calcul d'une machine de Turing se décompose en étapes. Une étape d'un calcul consiste à passer d'une configuration à une autre configuration en appliquant une des transitions de l'ensemble  $E$ . Une *étape de calcul* comprend les trois actions suivantes.

1. changer l'état de contrôle
2. écrire un symbole à la place du symbole sous la tête de lecture
3. déplacer sa tête de lecture d'une position vers la gauche ou la droite

Une transition de la forme  $p, a \rightarrow q, b, x$  peut uniquement être appliquée si la machine se trouve dans l'état  $p$  et si le symbole sous la tête de lecture est  $a$ . Après application de cette transition, la machine se trouve dans l'état  $q$ , le symbole  $a$  est remplacé sur la bande par  $b$  et la tête de lecture se déplace d'une position vers la gauche si  $x = L$  (Left) ou d'une position vers la droite si  $x = R$ .

(Right).

On note  $C \Rightarrow C'$  une étape de calcul qui fait passer la machine d'une configuration  $C$  à une configuration  $C'$  en appliquant une transition de  $E$ . On alors les étapes de calcul suivantes où  $u$  et  $v$  sont des mots quelconques sur l'alphabet de bande.

$$\begin{aligned} ucpav &\Rightarrow uqcbv & \text{si } p, a \rightarrow q, b, L \text{ est une transition} \\ upav &\Rightarrow ubqv & \text{si } p, a \rightarrow q, b, R \text{ est une transition} \end{aligned}$$

Il faut noter qu'une transition de la forme  $p, a \rightarrow q, b, L$  (avec déplacement de la tête de lecture vers la gauche) peut uniquement être appliquée si la tête de lecture ne se trouve pas sur le premier symbole de la bande.

Le fait que la présence de symboles blancs est implicite à la fin de la configuration implique que l'on a aussi les étapes de calcul suivantes où  $u$  est un mot quelconque sur l'alphabet de bande.

$$\begin{aligned} ucp &\Rightarrow uqcb & \text{si } p, \# \rightarrow q, b, L \text{ est une transition} \\ up &\Rightarrow ubq & \text{si } p, \# \rightarrow q, b, R \text{ est une transition} \end{aligned}$$

Ces relations concernent les cas où la tête de lecture se trouve sur un des symboles blancs implicites à la fin de la configuration.

Un *calcul* de la machine de Turing est une suite  $C_0, \dots, C_n$  de configurations telles que

$$C_0 \Rightarrow C_1 \Rightarrow C_2 \Rightarrow \dots \Rightarrow C_{n-1} \Rightarrow C_n.$$

Ceci signifie que la machine peut passer de la configuration  $C_i$  à la configuration  $C_{i+1}$ , c'est-à-dire  $C_i \Rightarrow C_{i+1}$  pour tout  $0 \leq i < n$ . Ceci est noté  $C_0 \Rightarrow^* C_n$  en abrégé (la relation  $\Rightarrow^*$  est en fait la clôture réflexive et transitive de la relation  $\Rightarrow$ ).

### Exemple

Les notions précédentes sont illustrées à l'aide de la machine de Turing  $M = (\{0\}, \{a, b\}, \{a, b, \#\}, E, 0, \{0\}, \#)$  avec les trois transitions suivantes.

$$0, a \rightarrow 0, \#, R \quad 0, b \rightarrow 0, \#, R \quad 0, \# \rightarrow 0, \#, L$$

Cette machine est simplissime et ne calcule rien de très intéressant. Son seul objectif est d'illustrer la notion de calcul.

Soit  $w = aba$  la donnée fournie à la machine. La configuration initiale est donc  $C_0 = 0aba$ . Les configurations suivantes sont données dans le tableau ci-dessous.

Configuration	Commentaire
---------------	-------------

$C_0 = 0aba$	Configuration initiale
$C_1 = \#0ba$	après application de la transition 0, $a \rightarrow 0, \#, R$
$C_2 = \##0a$	après application de la transition 0, $b \rightarrow 0, \#, R$
$C_3 = \###0$	après application de la transition 0, $a \rightarrow 0, \#, R$
$C_4 = \##0$	après application de la transition 0, $\# \rightarrow 0, \#, L$
$C_5 = \#0$	après application de la transition 0, $\# \rightarrow 0, \#, L$
$C_6 = 0$	après application de la transition 0, $\# \rightarrow 0, \#, L$

Calcul avec l'entrée aba.

On obtient alors le calcul suivant.

$$0aba \Rightarrow \#0ba \Rightarrow \##0a \Rightarrow \###0 \Rightarrow \##0 \Rightarrow \#0 \Rightarrow 0$$

qui est constitué de six étapes. La dernière configuration est bloquante. Aucune transition ne peut plus être appliquée. La tête de lecture se trouve sur le premier symbole de la bande et la seule transition qui lit le symbole  $\#$  donnerait un déplacement de la tête de lecture vers la gauche.

## Calculs acceptants

Une configuration  $C$  est dite *bloquante* s'il n'existe pas de configuration  $C'$  telle que  $C \Rightarrow C'$ . Ceci signifie qu'aucune transition ne peut être appliquée sur la configuration  $C$  pour passer à une configuration  $C'$ . Une configuration  $uqv$  est dite *acceptante* si l'état  $q$  est acceptant (final). Un calcul  $C_0 \Rightarrow^* C_n$  est dit *acceptant* si la première configuration  $C_0$  est de la forme  $q_0w$  et si la dernière configuration  $C_n$  est acceptante.

Lorsqu'on considère une suite de configurations

$$C_0 \Rightarrow C_1 \Rightarrow C_2 \Rightarrow C_3 \Rightarrow \dots$$

qui constituent le début d'un calcul, trois situations peuvent survenir.

1. Le calcul atteint une configuration acceptante. Il est alors acceptant.
2. Le calcul atteint une configuration bloquante. Il n'est alors pas acceptant.
3. Le calcul se poursuit de manière infinie sans jamais atteindre ni une configuration acceptante ni une configuration bloquante. Il n'est alors pas acceptant.

## Exemple

Soit la machine  $M = (\{0, 1, 2\}, \{a, b\}, \{a, b, \#\}, E, 0, \{2\}, \#)$  avec les transitions données à la [www.liafa.jussieu.fr/.../turing.html](http://www.liafa.jussieu.fr/.../turing.html)

figure ci-dessous.

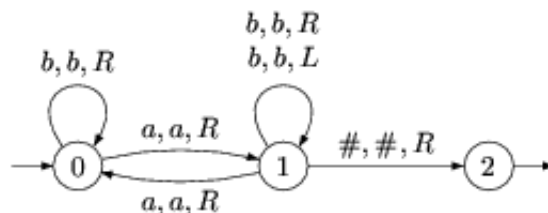


Fig. 4 : une autre machine de Turing

Le mot d'entrée  $w = abba$  conduit aux différents calculs suivants.

$0abba \Rightarrow a1bba \Rightarrow ab1ba \Rightarrow abb1a \Rightarrow abba0$

Ce calcul se bloque dans configuration  $abba0$  et n'est pas acceptant

$0abba \Rightarrow a1bba \Rightarrow 1abba \Rightarrow a0bba \Rightarrow ab0ba \Rightarrow abb0a \Rightarrow abba1 \Rightarrow abba\#2$

Ce calcul atteint l'état 2 et est donc acceptant

$0abba \Rightarrow a1bba \Rightarrow ab1ba \Rightarrow a1bba \Rightarrow ab1ba \dots$

Ce calcul ne termine pas et n'est donc pas acceptant.

Le mot  $abba$  est donc accepté par la machine puisqu'au moins un de ces calculs est acceptant.

Par contre le mot  $aa$  n'est pas accepté car le seul calcul possible est  $0aa \Rightarrow a1a \Rightarrow aa0$  qui n'est pas acceptant.

## Utilisation

Il y a deux modes d'utilisation des machines de Turing. Le premier mode est d'utiliser une machine comme *accepteur* et le second est d'utiliser une machine comme un *calculateur*.

Dans le mode *accepteur*, on fournit un mot d'entrée à la machine et celle-ci répond par *oui* ou par *non*. Quand la machine répond *oui*, on dit que la machine *accepte* le mot. La machine définit alors l'ensemble des mots qui sont acceptés. Par convention, on dit que la machine accepte un mot  $w$  s'il existe **au moins** un calcul acceptant avec  $w$  comme entrée, c'est-à-dire qui commence avec la configuration  $q_0w$ . L'élément important de cette définition est qu'un seul calcul acceptant suffit pour que la machine accepte même si d'autres calculs bloquent ou n'atteignent jamais une configuration acceptante.

Dans le mode *calculateur*, on fournit un mot d'entrée à la machine et celle-ci retourne un ou plusieurs mots de sortie. Quand la machine ne donne toujours qu'un seul mot de sortie, elle calcule une fonction qui associe le mot de sortie au mot d'entrée. Les mots de sortie sont par convention les contenus de la bande des dernières configurations des calculs acceptants. On met donc le mot d'entrée sur la bande, la machine effectue un calcul jusqu'à ce qu'elle atteigne un état final et le contenu de la bande constitue alors un des mots de sortie. Comme il peut y avoir

22/07/2009

Machines de turing

plusieurs calculs acceptants, il peut y avoir plusieurs mots de sortie. Dans le cas d'une machine déterministe, il y a au plus un seul calcul acceptant et donc au plus un seul mot de sortie.