

Master d'Informatique. D22. Complexité Algorithmique.

LE MODÈLE DE LA MACHINE DE TURING

Alan Turing a proposé un modèle mathématique formalisant la notion d'algorithme dans un article célèbre de 1936 intitulé "Sur les nombres calculables avec une application au problème des zéros de Hilbert". Ce modèle est largement utilisé en théorie car son extrême simplicité permet d'élaborer des raisonnements parfois difficiles à concevoir avec d'autres modèles. Nous commencerons par en donner une description *physique* avant d'en donner une formalisation mathématique. Alan Turing a basé son modèle en observant le fonctionnement d'une machine à écrire en omettant le "retour-chariot" qui n'a pour seule utilité que de rendre la lecture d'un texte plus aisée.

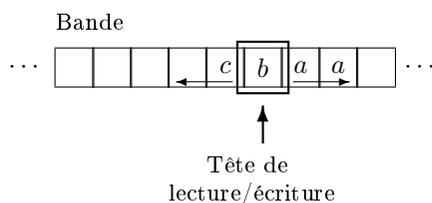


FIGURE 1. Une machine de Turing

Une machine de Turing (figure 1) est constituée de deux éléments :

- (1) Une **bande infinie** à gauche et à droite divisée en cellules. Une cellule peut être vide ou contenir un symbole d'un alphabet fini Σ arbitraire.
- (2) Une **tête de lecture/écriture** placée sur une cellule (dite *active*) et qui peut :
 - (a) lire le contenu de cette cellule.
 - (b) écrire un symbole de Σ dans cette cellule.
 - (c) effacer le symbole contenu dans cette cellule.
 - (d) se déplacer d'une cellule vers la gauche ou la droite.

Par soucis de simplification, une opération d'effacement sera confondue avec une opération d'écriture avec un symbole particulier \square appelé *blanc*. Il faut noter qu'une opération d'écriture "écrase" le symbole présent dans la cellule active.

C'est sur cette bande que les données de l'algorithme seront lues par le programme, ces données étant tout simplement représentées à l'aide d'un mot de Σ^* virtuellement "saisi" par l'utilisateur au préalable (le mot *cbaa* dans la figure 1). Par convention la tête de lecture est initialement placée sur le symbole le plus à gauche du mot qui constitue l'entrée de l'algorithme. C'est sur cette même bande que la machine écrit le résultat de son calcul, elle joue donc à la fois le rôle du périphérique d'entrée et de sortie.

A chaque instant, la machine est dans un *état* particulier. L'ensemble Q des états possibles est fini et on appelle *état initial* celui dans lequel la machine se trouve avant

l'exécution. Le fonctionnement de la machine est spécifié à l'aide d'un programme constitué d'une séquence d'instructions qui sont *toutes* et *exclusivement* du type conditionnel :

SI condition ALORS action.

Une condition est tout simplement un couple (q, α) constitué d'un état $q \in Q$ et d'un symbole $\alpha \in \Sigma$ ou le symbole "blanc". Symétriquement, une action est un couple (α', q') dont le premier terme α' est un symbole de Σ ou le symbole "blanc" ou l'un des deux symboles \leftarrow ou \rightarrow ; le deuxième terme q' est un état de Q . L'instruction conditionnelle se précise en SI (q, α) ALORS (α', q') et s'interprète

Si la machine est dans l'état q et que la cellule active contient le symbole α alors :

- si $\alpha' \in \Sigma$, on écrit α' dans la cellule active;
 - si $\alpha' = \square$, on "efface" la cellule active;
 - si $\alpha' = \leftarrow$, on déplace la tête d'une cellule vers la gauche;
 - si $\alpha' = \rightarrow$, on déplace la tête d'une cellule vers la droite;
- et le nouvel état de la machine est q' .

Pour simplifier les écritures, on résumera chaque instruction par un simple quadruplet (q, α, α', q') . On peut à présent donner une définition formelle :

Définition. Une machine de Turing est un quintuplet $(Q, \Sigma, \delta, q_0, \square)$ où

- (1) Q : un ensemble fini d'états;
- (2) Σ : un alphabet fini;
- (3) \square : le symbole blanc ($\notin \Sigma$);
- (4) q_0 : l'état initial;
- (5) δ : la fonction de transition $Q \times (\Sigma \cup \{\square\}) \rightarrow (\Sigma \cup \{\square, \leftarrow, \rightarrow\}) \times Q$.

Les instructions sont modélisées par la fonction de transition, le couple de l'ensemble de départ représente la condition à satisfaire et le couple image est l'action à réaliser.

Définition. On appelle instruction d'une machine de Turing $(Q, \Sigma, \delta, q_0, \square)$ tout quadruplet (q, α, α', q') tel que $(\alpha', q') = \delta(q, \alpha)$. On appelle programme ou algorithme, l'ensemble de ces quadruplets, c'est-à-dire le graphe de la fonction de transition.

Un algorithme est un ensemble d'instructions, le séquençage éventuel dans lequel on présentera les éléments de l'ensemble reflètera évidemment la structure de l'algorithme. On appellera *bande vierge* une bande dont toutes les cellules sont vides. On dit qu'un programme *s'arrête* quand aucune condition n'est satisfaite, i.e. quand le couple (q, α) n'admet pas d'image par la fonction de transition.

Exemple : Avec l'alphabet binaire $\Sigma = \{0, 1\}$ et $Q = \{q_0, q_1\}$, la machine de Turing définie par le programme $\{(q_0, 0, \rightarrow, q_0), (q_0, 1, \rightarrow, q_0), (q_0, \square, \leftarrow, q_1), (q_1, 0, 1, q_1)\}$ réalise l'opération $n \mapsto n + 1$ si n est pair et $n \mapsto n$ dans le cas impair quand n est représenté sur la bande par son écriture binaire.