# Calculabilité : définitions

Kévin Perrot – L3 Info Aix Marseille Université – printemps 2024

**Définition.** *Une* **machine de Turing** *(***MT***) déterministe est un 7-uplet* $M = (Q, \Gamma, \Sigma, \delta, q_0, B, q_F)$ *où*

- $Q$ *est un ensemble fini : les* **états***,*
- $\Gamma$ *est un ensemble fini : l'***alphabet de ruban***,*
- $\Sigma \subset \Gamma$ *est l'***alphabet d'entrée***,*
- $\delta : (Q \setminus \{q_F\}) \times \Gamma \to Q \times \Gamma \times \{L, R\}$ *est la* **fonction de transition***,*
- $q_0 \in Q$ *est l'***état initial***,*
- $B \in \Gamma \setminus \Sigma$ *est le* **symbole blanc***,*
- $q_F \in Q$ *est l'***état final***.*

**Définition.** *Le langage* **reconnu** *(ou* **accepté***) par la MT* $M$ *est*

$$L(M) = \{w \mid w \in \Sigma^* \text{ et } \iota_w \vdash^* uq_Fv \text{ avec } u, v \in \Gamma^*\}.$$

*C'est l'ensemble des mots d'entrée de* $\Sigma^*$ *pour lesquels la machine* $M$ *atteint l'état* $q_F$*.*

**Définition.** *Un langage est* **semi-décidable** *(on dit également* **récursivement énumérable***) s'il est reconnu par une machine de Turing. Un langage est* **décibable***, s'il est reconnu par une machine de Turing qui s'arrête sur toutes les entrées.*

$\epsilon$ désigne le mot vide. On note $M(w)\downarrow$ lorsque le calcul de $M$ sur $w$ s'arrête, et $M(w)\uparrow$ dans le cas contraire. Si $M(w)\downarrow$ alors on note $M(w)$ le **résultat** du calcul de la MT $M$ sur l'entrée $w$.

**Définition.** *Une fonction* $f : \Sigma^* \to \Gamma^*$ *est* **calculable** *si et seulement si il existe une MT* $M$ *telle que pour tout* $w \in \Sigma^* : f(w) = M(w)$*.*

**Définition.** *Un ensemble est de taille* **finie** *s'il contient un nombre fini d'éléments (sa taille est un entier naturel),* **infinie dénombrable** *s'il est en bijection avec* $\mathbb{N}$*,* **infinie indénombrable** *sinon.*

On note $\langle M \rangle$ le **code** de la machine de Turing $M$.

**Théorème.** *La fonction* **halt** : $(\langle M \rangle, w) \mapsto \begin{cases} 1 & \textit{si } M(w)\downarrow \\ 0 & \textit{sinon} \end{cases}$ *n'est pas calculable.*

*Démonstration.* Par l'absurde, supposons qu'il existe une machine de Turing $M_{halt}$ qui calcule la fonction *halt*. Nous pouvons alors sans difficulté construire la machine $M_{diag}$ suivante :

$$M_{diag}(i) = \begin{cases} 0 & \text{si } M_{halt}(i, i) = 0 \\ \uparrow & \text{si } M_{halt}(i, i) = 1 \end{cases}$$

où $\uparrow$ signifie que $M_{diag}$ entre dans une boucle infinie (et ne termine donc pas). Considérons à présent l'entrée $\langle M_{diag} \rangle$ donnée à la machine $M_{diag}$. Deux cas sont possibles.

- Si $M_{diag}(\langle M_{diag} \rangle) = 0$ alors, par définition de $M_{diag}$, nous avons $M_{halt}(\langle M_{diag} \rangle, \langle M_{diag} \rangle) = 0$ ce qui signifie, par définition de $M_{halt}$, que $M_{diag}(\langle M_{diag} \rangle)\uparrow$, une contradiction.
- Si $M_{diag}(\langle M_{diag} \rangle)\uparrow$ alors, par définition de $M_{diag}$, nous avons $M_{halt}(\langle M_{diag} \rangle, \langle M_{diag} \rangle) = 1$ ce qui signifie, par définition de $M_{halt}$, que $M_{diag}(\langle M_{diag} \rangle)$ s'arrête, une contradiction.

Dans les deux cas nous arrivons à une contradiction, donc notre hypothèse de départ est fausse : il n'existe pas de machine de Turing $M_{halt}$ qui calcule la fonction *halt*. $\square$

**Définition.** *Une* **réduction many-one Turing** *du langage* $A \subseteq \Sigma_A^*$ *vers le langage* $B \subseteq \Sigma_B^*$ *est une fonction calculable* $f : \Sigma_A^* \to \Sigma_B^*$ *telle que pour tout* $w \in \Sigma_A^*$ *on a* $w \in A \iff f(w) \in B$*. Lorsqu'une telle fonction* $f$ *existe on dit que la langage* $A$ *se réduit au langage* $B$*, et l'on note* $A \leq_m^T B$*.*