INFO-F408: Computability & complexity

Rémy Detobel

2 Octobre, 2017

1 Turing machine suite

1.1 Non déterministe

$$\delta : Q \times \Gamma \rightarrow P(Q \times \times \Gamma \times \{L, R\})$$

Voir livre : théorème 3.16 : Chaque NTM as un équivalent DTM.
On va donc faire un parcoure de l'arbre en largeur (et non en profondeur).

1.2 Reconnaitre un langage de turing

Voir théorème 3.21 :
Un langage est "turing-recognizable" si et seulement si un "enumerator" l'énumère.

1.2.1 Démonstration

($\Leftarrow$) Supposons qu'il existe un énumérateur "E" :
M = "lorsque l'entrée est w"

1. Exécuter E : chaque fois que E écrit(/output) un string, on le compare avec w
2. si c'est égal, on accepte.

($\Rightarrow$) Supposon qu'il existe une machine de turing qui reconnaisse le langage L.
E = "ignorer l'entrée"

1. Répéter pour $i = 1, 2, 3, ...$
   Exécuter M pour l'étape i, sur l'entrée $S_1, S_2, ...S_i$
   Si un exécution est acceptable, on affiche le S_j correspondant.
   Au pire on fera "i" étapes pour afficher un mot, mais il pourra être affiché avant l'étape "i".

step/input	S_1	S_2	S_3	S_4	S_5
1	x				
2	x	x			
3	x	x	x		
4	x	x	x	x	
5	x	x	x	x	x

1.3 Langage régulier (regular languages)

Langage régulier = reconnaissable par un automate fini (FA : finite automaton)
Langage décidable (= decidable = recursively) = décidable par une machine de turing.
Langage reconnaissable (recognizable languages = recursively enumerable = RE) =

— recognized par une machine de turing
— a un enumerateur ("enumerator")

Régulier < décidable < reconnaissable/recognizable

2 The Church-Turing thesis

C'est une thèse, pas une preuve.
$\Rightarrow$ La notion intuitive d'un algorithme est égal à un algorithme d'une machine de turing

2.1 Hilberts Problem

Est-ce qu'il existe un algorithme qui décide si un polynôme à une racine composée uniquement de nombre entier.
Exemple :

$$P(x) = x_1^2 + x_2x_3^4 - 6x_1x_2^3x_3x_4^2 + 7x_1$$

Et on cherche donc des nombres entier x_1, x_2, x_3, x_4

Il s'agit ici d'un problème "recognazable" (reconnaissable). Car si il y a une solution, on pourra la voir. Par contre, il n'est pas "décidable" parce que s'il n'y a pas de solution, il tournera à l'infini.

L'indécidabilité de ce problème à été prouvé en 1970 par Matijasevic.

3 Halting problème (problème de l'arrêt)

Point 4.2.
Diagonalization (cantor) $f : A \rightarrow B$ est :
"un à un" si tous les élément de A sont projeté de manière distincte sur des éléments de B.
"dans" lorsque tous les éléments de A sont dans B, par exemple :

$$\forall b \in B, \exists a \in A : f(a) = b$$

"one-to-one" (un à un) ET "onto" (dans) = "one-to-one correspondance"
C'est équivalent à une bijection.

Une ensemble A est "countable" (dénombrable) si il est finie OU t'il existe une correspondance un à un ("one-to-one correspondence") entre A et $\mathbb{N}$ (ce qui est équivalent à dire qu'il a la même "taille" que $\mathbb{N}$).
Exemple : est-ce que :

- les nombres paires sont dénombrable ?
  $\rightarrow$ Oui ($\mathbb{N}/2$)
- les nombres rationnels ($\mathbb{Q}$) sont dénombrable ?
  $\rightarrow$ Oui (pour cela il faut juste mettre un ordre. Pour se faire, on peut parcourir un tableau à double entrées représentant les numérateurs et dénominateurs. Il suffirait donc de simplement définir l'ordre de lecture qui logiquement se ferait plutôt en diagonal).
- $\mathbb{Z}$ est dénombrable ?
  $\rightarrow$ Oui (nombre négatif étant des paires, nombre positif étant des impaires. De cette manière on compte tous les nombres).

3.1 Cantor's Diagonal

Théorème : $\mathbb{R}$ est indénombrable ("not countable").
Prouvons cela par contradiction :
Supposons donc que $\mathbb{R}$ est dénombrable. On a donc une liste qui fait correspondre tous les nombres naturels ($\mathbb{N}$) à un nombre présent dans $\mathbb{R}$). On va donc prouver qu'il existe un $x \in [0, 1]$ qui n'est pas dans cette liste. Pour construire le x, on va prendre

1	0,**3**1415926535
2	1,0**0**000000000
3	22,12**3**12312312
4	323,010**1**0101010
5	4,1502**6**535010
6	...

le nom à la position i et l'incrémenter. Ici x vaut donc : $x = 0, 41427...$ Donc, par construction, il ne peut pas être dans la liste.

Prenons $\mathcal{L}$ comme étant l'ensemble des langages sur l'alphabet Σ
Prouver que $\mathcal{L}$ est indénombrable ("uncountable").