04-Calculabilité

PEPS

Les problèmes de la théorie de la calculabilité

- la décidabilité
    - la notion de machine universelle
    - le problème de la HALT
    - le théorème de Rice
- la complexité
    - la notion de temps et d'espace
    - le non déterminisme
    - la classification: P
- sujets connexes
    - déduction et calcul
    - la complexité algorithmique de l'information
    - la complexité de Kolmogorov

La nature dynamique de la théorie du calcul : les systèmes étudiés dans les théorie sont dynamiques dans le sens qu'ils évoluent au cours du temps et "font quelque choses".

Le codage

> Let problem A be presented via a finite sequence of positive integers, $m_1, m_2, \ldots, m_n$, together with an (n + 1)th positive integer N . The "problem" is to determine whether there is a subset of the integers m_i that sum to N. (A is often called the **subset-sum problem**.)

> Let problem B be presented by: (a) a set of locations; (b) a matrix of nonnegative numbers whose (i, j)th entry is the "cost" of traveling from location i to location j—it makes no difference how this "cost" is measured: it is just a number; (c) a target cost C. The "problem" is to determine whether there is a tour that visits each location precisely once and that incurs aggregate cost ≤ C. (B is often called the **traveling salesman problem**.)

On peut trouver un moyen (un processus calculatoire, un programme, un algorithme) qui permet d'encoder les instances du problème A comme des instances du problème B de telle sorte que les réponses "yes" des instances de A correspondent aux réponses "yes" du problème B et idem pour le "no".
$\exists \phi_B . \forall A . \forall i . A(i) = \phi_B(A, i)$
On peut aussi encoder les instances de B dans les instances de A.

Machine universelle

Sur la bande d'une machine de Turing (MT), on met une suite de 0 et de 1. On peut coder un programme comme une suite de 0 et de 1 (et donc comme un entier).

On peut écrire un programme qui exécute n'importe qu'elle machine codée sur la bande (cela marche aussi avec les autres modèles, dès que l'on a compris qu'**un programme n'est qu'un entier et que l'on a les "moyens" de coder et décoder cet entier**.

Le problème de l'arrêt

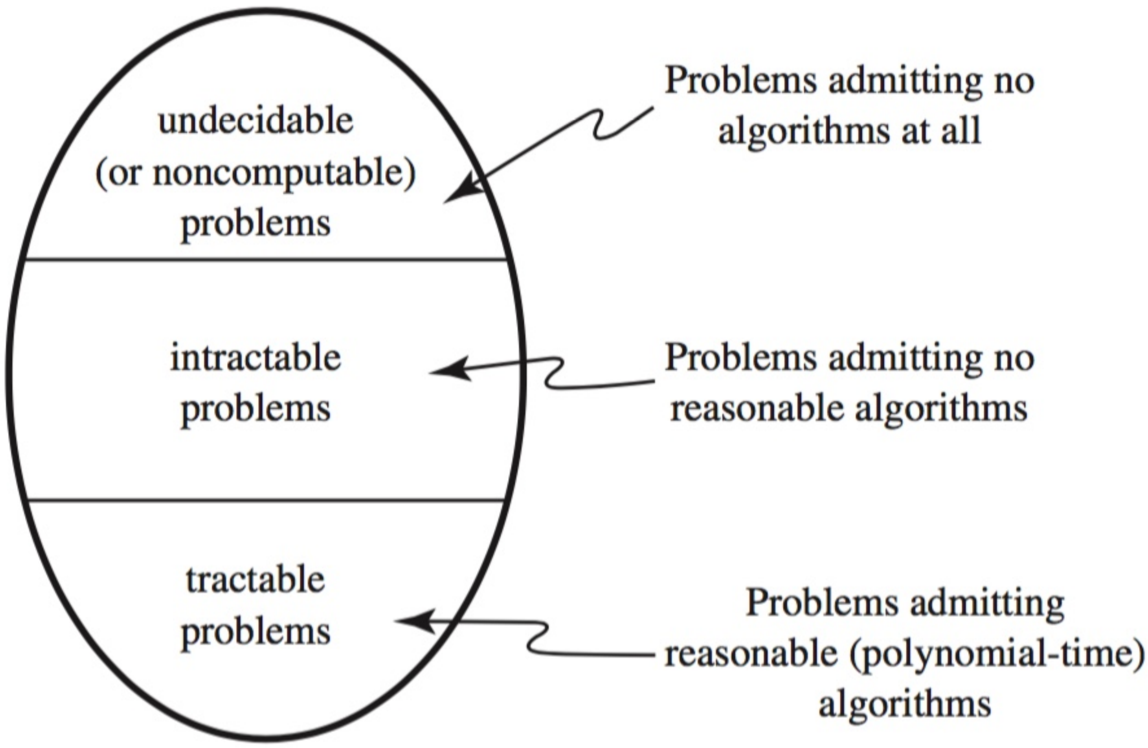
Le problème est de décider étant donné un programme P et une entrée x pour P si le programme P s'arrête sur x :
On veut savoir s'il existe un procédé qui permet de dire si "oui le programme P s'arrête sur x" ou "non le programme P ne s'arrête pas sur x". On souhaite que ce procédé pour tout P et pour tout x.

La réponse : Non il n'existe pas de tel procédé; on dit que le problème de l'arrêt est *indécidable*.

Théorème de Rice

> Aucun problème sur les propriétés des calculs n'est décidable (Th. de Rice).

Complexité


undecidable
(or noncomputable)
problems
Problems admitting no
algorithms at all
intractable
problems
Problems admitting no
reasonable algorithms
tractable
problems
Problems admitting
reasonable (polynomial-time)
algorithms