Algorithmique 2

L3 RI

Table des matières

1 NP-Complétude

1.1 Définitions

Réduction Soient L_1, L_2 des langages. Une réduction polynomiale de L_1 à L_2 est une fonction f calculable en temps polynomial telle que :

$$f(x) \in L_2 \iff x \in L_1$$

On note $L_1 \leq L_2$ (L_1 plus facile que L_2 = Si on sait résoudre L_2, on sait résoudre L_1)

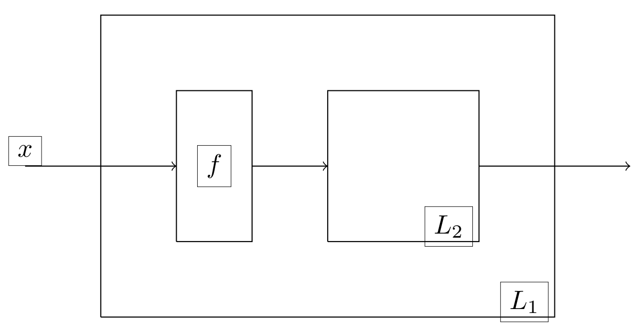

Figure 1 – Schéma de réduction de L_1 à L_2

Classe NP Soit L un langage. L est dit dans la classe NP s'il existe une machine de Turing non-déterministe en temps polynomial par rapport à la taille de l'entrée qui décide L.

NP-Complétude Soit L un langage. L est dit NP-dur si pour tout $L' \in$ NP, $L' \leq L$. L est dit NP-complet si $L \in$ NP et L est NP-dur.

Remarques Si un problème NP-complet est dans P, alors P = NP. Pour montrer que L' est NP-dur, il suffit de montrer qu'il existe L NP-dur tel que $L \leq L'$.

1.2 Satisfiabilité d'une formule

Problème de décision : SAT

Entrée φ formule de la logique propositionnelle
Sortie Oui si φ est satisfiable, c'est-à-dire s'il existe une valuation v des variables qui rend φ vraie

Théorème de Cook SAT est NP-complet.

Problème de décision : i-SAT

Restriction de SAT à des formules en forme normale conjonctive (CNF) tel que chaque clause contient au plus i littéraux

Théorème 3-SAT est NP-complet.

Théorème 2-SAT est dans P.

1.3 Ensembles indépendants dans un graphe

Problème de décision : ENS_INDEP

Entrée $G = (V, E)$ un graphe non-orienté et $k \in \mathbb{N}$
Sortie Oui s'il existe $V' \subseteq V$ tel que $|V'| = k$ et $(V' \times V') \cap E = \varnothing$

Théorème ENS_INDEP est NP-complet

2 Algorithmes d'approximation

3 Algorithmes probabilistes

4 Géométrie algorithmique

5 Algorithmes distribués