Classes P et NP. Problèmes NP-Complets.

Exemples.

I. La classe P

A. Problèmes de décision P

Définition Problème de décision

Étant donné un ensemble L, le problème de décision associé à L est le fait de savoir si un x donné appartient à L.

Définition Complexité temporelle d'un problème

Soit $f : \mathbb{N} \to \mathbb{R}$. Un problème de décision Pb appartient à $TIME(f)$ s'il existe un algorithme A qui résoud Pb tel que $A(x)$ s'éxécute en $O(f(|x|))$ pour tout x.

Exemple Le problème de l'appartenance à un tableau trié appartient à $TIME(\log)$.

Définition Un problème Pb appartient à la classe P s'il existe un polynôme f tel que Pb appartient à $TIME(f)$

Exemples :

- 2-SAT :

  Entrée : une formule en 2-CNF φ

  Question : φ est-elle satisfiable ?

  2-SAT est P. $\boxed{\text{DEV}}$

B. Généralisation aux problèmes d'optimisation

Définition : Problème d'optimisation

Un problème d'optimisation est un problème de la forme

Entrée : ensemble X, fonction $f : X \to \mathbb{R}$

Sortie : $\underset{x \in X}{\operatorname{argmin}} f(x)$

La notion de classe de complexité s'étend naturellement.

Exemple

· Couplage parfait de poids minimal

Entrée : un graphe $G = (V, E)$ et $f : E \to \mathbb{R}$ (G non-orienté)

Sortie : $\underset{C \subseteq E}{\operatorname{argmin}} \sum_{e \in C} f(e)$

$\forall u \in V, \exists! v \in V, (u, v) \in C$

Ce problème est P.

II. La classe NP

A. Problèmes de décision NP

Définition Un algorithme de vérification est un algorithme à deux arguments : une chaîne x et une autre chaîne y appelé certificat. On dit que A accepte x s'il existe y tel que $A(x,y) = 1$.

Définition Un problème est NP si son langage est de la forme $\{ x \mid \exists y, |y| = O(f(|x|)), A(x,y)=1 \}$ avec f un polynôme et A un algorithme polynomial de vérification.

Proposition $P \subseteq NP$

Conjecture $P \neq NP$

Exemple : SAT est NP.

Définition Un problème A est réductible en temps polynomial à un problème B si il existe une fonction f calculable en temps polynomial telle que, pour toute entrée x x est solution de A ssi $f(x)$ est solution de B.

Exemple SAT est réductible à 3-SAT (transformation de Tseitin)

Définition Un problème est NP-difficile si tout problème NP peut s'y réduire. Si de plus le problème est NP, on dit qu'il est NP-complet.

Proposition Un problème est NP-difficile ssi il existe un problème NP-difficile qui s'y réduit

Théorème (Cook-Levin)

SAT est NP-complet

Preuve : choisir un modèle de calcul. Pour un problème NP donné, considérer un algorithme qui le vérifie en temps polynomial. Exprimer l'assertion "A accepte x" en logique propositionnelle.

B. Problèmes d'optimisation NP

Définition Deux problèmes (quelconques) A et B sont P-équivalent si étant donné un oracle pour B, il existe un algorithme polynomial qui résout A, et réciproquement.

Définition / Abus de langage Un problème (quelconque) est dit NP-complet s'il est P-équivalent à un problème (de décision) NP-complet

III Problèmes NP-complets et approximations

Tous les graphes de cette partie sont simples et non-orientés.

A. Cliques

Problème CLIQUE

Entrée : $G = (V, E)$, $k \in \mathbb{N}$

Question : G admet-il un clique de taille (au moins) k ?

Proposition CLIQUE est NP-complet

↳ Preuve : réduction depuis 3-SAT

Problème CLIQUE-COVER

Entrée : $G = (V, E)$, $k \in \mathbb{N}$

Question : Peut-on partitionner G en k cliques ?

Proposition CLIQUE-COVER est NP-complet

↳ Preuve : DEV

B. Couverture de sommets

Problème VERTEX-COVER

Entrée : $G = (V, E)$ $k \in \mathbb{N}$

Question : Existe-t-il $U \subseteq V$ tel que $|U| = k$ et toute arête de E a au moins une extrémité dans U.

Proposition VERTEX-COVER est NP-complet

↳ Preuve Clique$((V, E), k) \iff$ Vertex Cover$((V, V^2 \setminus E), |V| - k)$

Problème : MIN-VERTEX-COVER

Entrée : $G = (V, E)$

Sortie : $\min \{ k \mid \text{VERTEX-COVER}(G, k) \}$

Proposition MIN-VERTEX-COVER est NP-complet

Proposition Il existe une 2-approximation en $O(|V| + |E|)$ de MIN-VERTEX-COVER

↳ Algo : Tant qu'il reste des arêtes non-couvertes, en choisir une, ajouter ses sommets à la couverture et retirer toutes les arêtes incidentes.

C. Cycle hamiltonien

Proposition Savoir si un graphe admet un cycle hamiltonien est NP-complet.

Problème du voyageur de commerce (TSP)

Entrée : $G = (V, E)$, $\delta : E \to \mathbb{N}$

Sortie : $\operatorname{argmin}_{C,\ \text{cycle ham. de } G} \sum_{e \in C} \delta(e)$

Proposition : TSP est NP-complet

Proposition : TSP admet une $\frac{3}{2}$-approximation en temps polynomial

↳ Preuve : DEV