31 - Classes P et NP. Problèmes NP-complets. Exemples

Niveau : MPI Prérequis : - Bases solides d'algorithmique
- aisance en Ocaml.
- Logique propositionnelle.

Motivation : Jusqu'ici la notion de complexité désigne celle d'un algorithme. Le but de ce chapitre est d'introduire la notion de complexité d'un problème, afin d'envisager les limites de ce que peut faire un ordinateur en un temps raisonnable.

I] Introduction

Définition 1 : Dans ce chapitre, on appelle algorithme un programme Ocaml que l'on peut exécuter sur une machine dotée d'une mémoire infinie.

En effet, comme on étudie la complexité d'un problème, on doit considérer des machines avec des mémoires arbitrairement grandes.

Définition 2 : Un problème de décision sur un domaine d'entrées E est une fonction totale f de E vers $\mathbb{B}$. Un algorithme A résout f si, pour tout $e \in E$, l'algorithme A appliqué à e termine et produit $f(e)$. On appelle e une instance du problème f.

Remarque 3 : Les instances des problèmes rencontrés seront toujours des chaînes de caractères. En effet, on étudiera ces problèmes pour des instances arbitrairement grandes.

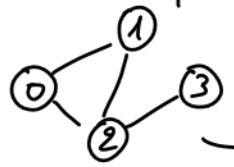
Exemple 4 : On considère le problème Premier, prenant en entrée une chaîne de caractère s représentant un entier n tel que Premier $(s) = V$ si et seulement si n est premier.

Les entiers Ocaml étant bornés par $2^{30}-1$, on ne pourrait pas résoudre ce problème si l'entrée était de type int ! ⚠

Cas des structures complexes

Exemple 5 : Soit Connexe le problème prenant en entrée une chaîne s représentant un graphe $G=(S,A)$ tel que Connexe$(s)=V$ si et seulement si G est connexe.

Si la représentation de G par s n'est pas imposée, on est libre de choisir celle qui nous arrange le plus.



"[1,2];[0,2];[0,1,3];[2]"

"[0,1,1,0];[1,0,1,0];[1,1,0,1];[0,0,1,0]"

Définition 6 : Soit f un problème de décision sur un domaine d'entrée E. On définit la taille de $e \in E$ comme la longueur de la chaîne de caractères la représentant.

Remarque 7 : Les entiers étant représentés en binaire, la taille d'un entier $n \in \mathbb{N}$ est en $O(\log n)$!

II] Classes P et NP

1) Classe P

Définition 8 : La classe P est définie comme l'ensemble des problèmes de décision qui admettent une solution dont la complexité temporelle est majorée asymptotiquement par un polynôme en la taille de l'entrée.

i.e. il existe un algorithme A qui résout f et un polynôme p tel que l'exécution de A sur une entrée de taille n est inférieure à $p(n)$.

Exercice : On pourra vérifier que les problèmes suivants déjà rencontrés appartiennent bien à la classe P :
- Déterminer s'il existe des doublons dans un tableau.
- Déterminer s'il existe un chemin entre 2 sommets d'un graphe.
- Déterminer si un mot est reconnu par un automate fini déterministe.

2) Réduction polynomiale

Définition 9 : Soient deux problèmes de décision $f_1 : E_1 \to \mathbb{B}$ et $f_2 : E_2 \to \mathbb{B}$. On dit que f_1 se réduit polynomialement à f_2, et on note $f_1 \leq_p f_2$, s'il existe une fonction $g : E_1 \to E_2$ de complexité temporelle polynomiale en la taille de son entrée telle que pour tout $e \in E_1$, on ait $f_1(e) = f_2(g(e))$

Théorème 10 : Soient deux problèmes de décision f_2 et f_1. Si $f_2 \in P$ et $f_1 \leq_p f_2$, alors $f_1 \in P$.

Activité 11 : Exercice de DM/DS :
Réduction polynomiale de 2-SAT au problème des composantes fortement connexes DEV1

3) Classe NP

Définition 12 : Un problème de recherche sur un domaine E d'entrées et un domaine de solutions S est défini par une relation binaire $R \subseteq E \times S$. Un algorithme A résout un problème de recherche R si $\forall e \in E$, A appliqué à E produit un $s \in S$ tel que $R(e,s)$, lorsque c'est possible.

Définition 13 : Soit $R \subseteq E \times S$ un problème de recherche. Le problème de vérification d'une solution associé à R est le problème de décision f sur l'ensemble $E \times S$ tel que $f((e,s)) = V$ si et seulement si $R(e,s)$.

Définition 14 : La classe NP est l'ensemble des problèmes de décision $f : E \to \mathbb{B}$ tel que :
- Il existe un ensemble C de certificats et une fonction $g : E \times C \to \mathbb{B}$ telle que $f(e) = V$ si et seulement si il existe $c \in C$ polynomial en e tel que $g(e,c) = V$
- Le problème défini par g, appelé problème de vérification de certificat, est dans la classe P.

Définition 15 : Le problème SAT prend en entrée une formule propositionnelle φ. On a $SAT(\varphi) = V$ si et seulement si φ est satisfiable

Propriété 16 : SAT est dans NP.

Exercice 17 : Montrer que le sudoku généralisé est dans NP.

Propriété 18 : $P \subseteq NP$ (on ne sait pas si l'inclusion est stricte !)

III NP-complétude

1) Définitions

Définition 19 : Un problème NP-difficile est un problème f tel que $\forall g \in NP,\ g \leq_p f$
Un problème NP-complet est un problème NP-difficile, qui est dans la classe NP.

Théorème 20 : Si il existe un algorithme polynomial qui résout un problème NP-complet, alors tous le problèmes NP ont une solution polynomiale.

Théorème 21 : Soient f_1 et f_2 deux problèmes de décision. Si f_1 est NP-complet et $f_1 \leq_p f_2$ alors f_2 est NP-complet

Remarque 22 : Il est très difficile de montrer qu'un problème est NP-complet sans réduction, il faut un problème de référence.

Théorème 23 (de Cook-Levin) : SAT est NP-complet

2) Problèmes NP-complets

a) SAT-CNF

Définition 24 : SAT-CNF est la restriction du problème SAT aux formules propositionnelles sous forme CNF.

Théorème 25 : SAT-CNF est NP-complet.

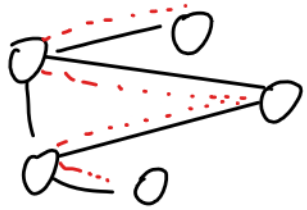
⚠ Certaines formules ont leur formule CNF équivalente exponentiellement plus grande. cette idée ne mène donc pas à une réduction polynomiale !

Propriété 26 : Si Ψ est une formule propositionnelle. Alors la transformée de Tseitin de Ψ est une formule sous forme CNF, équisatisfiable avec Ψ, de taille linéaire en celle de Ψ.

b) Problème du chemin hamiltonien.

Définition 27 : Un chemin hamiltonien d'un graphe G est un chemin qui passe une unique fois par chaque sommets.

Le problème du chemin hamiltonien prend en entrée un graphe orienté ainsi que deux sommets. On se pose alors la question de l'existence d'un chemin hamiltonien entre ces sommets :



Théorème 28 : Le problème du chemin hamiltonien est NP-complet.

c) Problème de la 3-colorabilité d'un graphe

Définition 29 : Un graphe $G = (S, A)$ est 3-colorable si il existe une application $c : S \to \{R, G, B\}$ telle que $\forall (u,v) \in A,\ c(u) \neq c(v)$.

Théorème 30 : Le problème de la 3-colorabilité d'un graphe est NP-complet