

Définition informelle

procédure de calcul bien définie qui prend en entrée une valeur, ou un ensemble de valeurs, et qui donne en sortie une valeur, ou un ensemble de valeurs.

Exemples de problèmes

- ▶ Tris (toutes les applications ou presque)
- ▶ Graphes (gestion d'itinéraires, routage internet)
- ▶ Calcul numérique (cryptographie, commerce électronique)
- ▶ Recherche d'optimum (gestion de ressources)
- ▶ ...

Définition

moyen de stocker et d'organiser des données pour faciliter l'accès à ces données et leur modification

Exemple de structures

- ▶ Elémentaires
 - ▶ Piles, Files, Listes, Arbres
- ▶ Tables de hachage
- ▶ Arbres binaires de recherche, B-Arbres, Arbres rouge-noir
- ▶ Tas binomiaux, Tas de Fibonacci
- ▶ Graphes
- ▶ Ensembles

Algorithme

Suite d'instructions permettant de passer des données initiales d'un problème au résultat final.

Complexité

Mesure de l'efficacité (performance) d'un algorithme

1. Nombre d'opérations élémentaires :
 - 1.1 opérations élémentaires sur les entiers et réels
 - 1.2 opérations sur vecteurs dans \mathbb{R}^n (resp. sur matrices) définies à partir de n opérations élémentaires (resp. n^2)
2. Taille des données : nombre de bits nécessaires pour coder ces données dans la machine (ex : n sommets $\rightarrow n^2$ bits).

Exemple

- ▶ Algorithmes polynomiaux en $\mathcal{O}(n^k)$
- ▶ Algorithmes exponentiels en $\mathcal{O}(\exp(n))$

Remarque

1. Entre deux algorithmes linéaires ($\mathcal{O}(n)$), en doublant la taille des données, on double le temps de calcul.
2. Entre deux algorithmes cubiques ($\mathcal{O}(n^3)$), en doublant la taille des données, on multiplie par 8 le temps de calcul.

Problèmes difficiles

- ▶ Efficacité d'un algorithme : vitesse ?
- ▶ Certains problèmes n'ont pas de solution efficace : NP-complets

Voyageur de commerce

Visiter plusieurs villes et revenir au point de départ en parcourant le minimum de distance. Pas d'algorithmes efficaces exactes mais certains algorithmes efficaces satisfaisants.

Complexités exponentielles et problèmes NP

ALGORITHMIQUE - EISTI - ING 1

Yannick Le Nir
yannick.lenir@eisti.fr

Ecole Internationale des Sciences du Traitement de l'Information

Cadre général

La classe P

La classe NP

Problèmes
 NP -complets

Résolution de
problèmes NP

Sommaire

Cadre général

La classe P

La classe NP

Problèmes NP —complets

Résolution de problèmes NP

Complexités
exponentielles et
problèmes NP
ALGORITHMIQUE
- EISTI - ING 1

Yannick Le Nir
yannick.lenir@eisti.fr

Cadre général

La classe P

La classe NP

Problèmes
 NP —complets

Résolution de
problèmes NP

Complexité intrinsèque d'un problème

- ▶ Complexité minimale d'un algorithme résolvant ce problème.
- ▶ Existe-t-il un algorithme polynomial pour résoudre un problème donné.
- ▶ Comment-dire qu'un algorithme est optimal (en complexité).
- ▶ Comment montrer qu'un algorithme polynomial n'existe pas.
- ▶ Qu'est-ce qu'un problème dur.
- ▶ Comment prouver qu'un problème est aussi dur qu'un autre.

Cadre général

La classe P

La classe NP

Problèmes
 NP -complets

Résolution de
problèmes NP

Problèmes praticables

Définition

La classe P (ou $PTIME$) est la classe des problèmes pour lesquels il existe un algorithme de résolution polynomial en temps.

Praticable

- ▶ Définition indépendante du modèle d'algorithme choisi (langage C, machine de Turing, etc ..), excepté les ordinateurs quantiques.
- ▶ Par convention, praticable = polynomial (abus de langage pour degrés élevés du polynome).

Généralités

NP contient P et est contenue dans $EXPTIME$ et $PSPACE$.

Elle est souvent associée à des problèmes courants :

- ▶ emploi du temps
- ▶ placement de tâches
- ▶ problèmes de tournées...

Cadre général

La classe P

La classe NP

Problèmes
 NP -complets

Résolution de
problèmes NP

Non-déterministe Polynomial

- ▶ Conjecture $P \neq NP$ ouverte en 1971
- ▶ Institut Clay : 1 million de dollars pour sa résolution
- ▶ Chercheur partagés mais conjecture dominante

Equivalence des problèmes NP

- ▶ Conjecture $P \neq NP$ liée à l'existence de problèmes NP -complets
- ▶ Si un problème NP -complet peut être résolu en temps polynomial, alors tous les problèmes de NP peuvent être résolus en temps polynomial (i.e $P = NP$)
- ▶ Malgré des années de recherches, aucun algorithme polynomial n'a jamais été découvert pour quelque problème NP -complet que ce soit.
- ▶ Ce sont les problèmes les plus difficiles de NP

Cadre général

La classe P

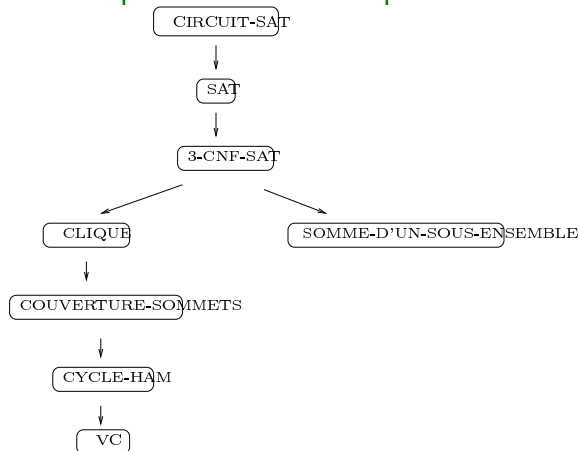
La classe NP

Problèmes
 NP -complets

Résolution de
problèmes NP

Quelques problèmes NP -complets

Liste de problèmes NP -complets



Comment résoudre les problèmes NP ?

Impossibilité d'énumérer les solutions

- ▶ algorithmes dits naïfs
- ▶ explosion combinatoire : algorithmes exponentiels
- ▶ complexité en temps ou en espace trop importante

Solutions possibles

- ▶ méthodes exactes d'amélioration :
 - ▶ diviser pour régner
 - ▶ algorithmes gloutons sous certaines conditions
- ▶ méthodes heuristiques : obtention rapide (polynomiale) de solutions approchées, pas forcément optimales.

Algorithmes gloutons

Contexte

Algorithmes souvent utilisables si la solution résulte d'un ensemble de choix

- ▶ jeux
- ▶ parcours d'un arbre ou d'un graphe
- ▶ ordonnancement de tâches
- ▶ ...

Principe

Toujours choisir le meilleur choix localement sans jamais revenir sur ce choix : ne garantit pas toujours de trouver la solution optimale

Exemples

Algorithmes souvent issus d'applications d'intelligence artificielle (IA), qui seront étudiés en ING2 à l'EISTI :

- ▶ algorithmes de recherche tabou
- ▶ algorithmes du recuit simulé
- ▶ algorithmes génétiques
- ▶ réseaux de neurones
- ▶ ...