Ecole Supérieure d'Economie Numérique

Cours complexité algorithmique (3LFIG)

cours 7: NP-Complétude

Dr. Dhouha Maatar Razgallah
2019/2020

Outline

- *Introduction*
- *Classification des problèmes*
- *NP-Complétude*

2

Introduction

➢**Un problème d'optimisation combinatoire**: un problème qui consiste à chercher une meilleure solution parmi un ensemble de solutions réalisables.
Exemple : le problème PLUS-COURT-CHEMIN qui consiste à trouver le plus court chemin entre deux sommets u et v d'un graphe G.
➢**Problème de décision**: un problème qui consiste à apporter une réponse "oui" ou "non" à une question.
Exemple : le problème CHEMIN qui répond à la question « étant donné un graphe G, deux sommets u et v et un entier positif k, existe-t-il dans G un chemin de u à v de longueur au plus k ? ».

Restriction aux problèmes de décision:

➢Dans le cadre de la théorie de la NP-complétude, on se restreint aux **problèmes de décision.**
➢Pour appliquer la théorie de la NP-complétude sur les problèmes d'optimisation, le plus souvent on les reformulera sous la forme d'un problème de décision en imposant une borne sur la valeur à optimiser, comme on l'a fait en passant du problème PLUS-COURT-CHEMIN au problème CHEMIN.

3

Introduction

Algorithme déterministe:

➢un algorithme dont la solution qu' il produit peut être déduite des spécifications de l'algorithme lui-même.
➢Un algorithme qui n'a qu'une possibilité à un moment donné.

Algorithme non-déterministe:

➢un algorithme dont la solution est devinée puis vérifiée.
➢un algorithme qui a plusieurs choix à un moment donné, et qui nécessite donc d'explorer ces différentes possibilités.

Algorithme efficace (facile):

➢Un algorithme est dit **efficace (facile)** si sa complexité en temps est polynomiale, c'est-à-dire en $O(n^k)$ pour un entier k.
➢Un problème est de complexité polynomiale s'il existe un algorithme efficace le résolvant.

4

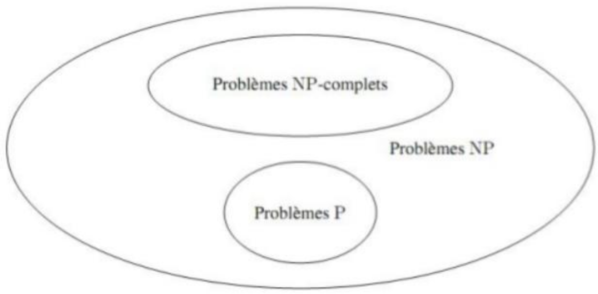
Classification des problèmes | Classes de NP-Complétude

On classe les problèmes et leurs complexités dans la théorie de la NP-complétude en grandes catégories de complexité à savoir:

➢**La classe P** : classe des problèmes faciles à résoudre.

➢**La classe NP** (Non deterministic Polynomial) : classe des problèmes faciles à vérifier (pas nécessairement à résoudre).

➢**NP-complet**: classe des problèmes aussi dur que tout problème raisonnable.



NP-Complétude | P

➢**La classe P** :

✓C'est l'ensemble des problèmes de **décision** qu'on peut résoudre avec des algorithmes **déterministes** de complexité **polynomiale** (facile).

✓La classe P est la classe des problèmes de décision pour lesquels il existe un algorithme de résolution polynomial.

✓La classe P, qui regroupe les problèmes les plus simples de la classe NP, contient les problèmes pour lesquels on connaît au moins un algorithme polynomial pour les résoudre.

✓Comment montrer qu'un problème est dans P ?

• En trouvant un algorithme polynomial pour le résoudre.

Exemples de problèmes P:

-Problème de "être premier" qui à tout entier naturel associe oui si il est premier, non sinon.

-Problème de "Etre pair" est P.

-Problème de "Etre trié" pour une liste d'entiers est P.

NP-Complétude | NP

➢La classe NP :

✓un problème de classe NP est un problème de **décision** qu'on peut résoudre avec un algorithme **Non-déterministe** de complexité **polynomiale** (facile).

✓Pour montrer qu'un problème est dans la classe NP il suffit de trouver un algorithme qui vérifie si une solution donnée est valide en temps polynomial.

✓La classe NP est la classe des problèmes de décision pour lesquels il existe un algorithme de vérification polynomial.

✓Si l'on nous donne une solution, nous pouvons vérifier que cette solution est correcte en temps polynomial.

✓Les problèmes de la classe NP ne sont pas nécessairement résolus en temps polynomial

✓Comment montrer qu'un problème est dans NP ?

•On utilise la méthode des certificats (en utilisant un algorithme de validation).

Exemples de problèmes NP:

•Problème du chemin de longueur <=k

•Problème du cycle Hamiltonien

•Problème de K-coloriabilité

7

NP-Complétude | NP

Algorithme de validation:

Un **algorithme de validation** pour un problème Q est un algorithme de décision A à deux arguments, où un argument est une instance x du problème, et où l'autre argument est **un certificat** y.

L'algorithme A valide l'entrée x si et seulement si il existe un certificat y tel que A(x;y) = vrai.

Exemple 1:Problème du chemin de longueur <=k

Considérons le problème CHEMIN et une de ses instances (G,u,v,k). La question qui nous intéresse est donc : **existe-t-il dans le graphe G un chemin reliant les sommets u et v dont la longueur est inférieure ou égale à k** ?

▪Si l'on se donne également un chemin p de u vers v, on cherche à vérifier que la longueur de p est au plus égale à k et, si on peut facilement le vérifier, on peut voir p comme un **certificat** que le problème de décision CHEMIN renvoie vrai sur l'instance (G,u,v,k)**.** Donc validation effectuée à partir du certificat p sur cette instance.

8

NP-Complétude | NP

Exemple 2: Problème du cycle Hamiltonien:
Etant donné le graphe G= (X, E) non orienté, déterminer s'il existe un cycle hamiltonien c'est-à-dire décider s'il existe une chaine de G passant une fois et une seule par chacun des sommets et revenant à son point de départ.

▪Dans le problème du cycle hamiltonien, **le certificat** est la liste des sommets du cycle hamiltonien**.**
Si un graphe est hamiltonien, le cycle lui-même offre toute l'information nécessaire pour le prouver. Réciproquement, si un graphe n'est pas hamiltonien, il n'existe aucune liste de sommets capable de faire croire à l'algorithme de validation que le graphe est hamiltonien : l'algorithme de validation se rend bien compte que le cycle décrit par la liste des sommets n'est pas un cycle du graphe étudié.

NP-Complétude | NP

Exemple 3:problème de coloriage de graphe
Etant donné le graphe G= (X, E) non orienté, déterminer le nombre chromatique de G (nombre minimal de couleurs pour colorier les sommets X de G de façon que deux sommets adjacents soient de couleur différentes).
Le problème de décision correspondant est :

- ▪soit un graphe G =(X,E) et un entier k
- ▪Déterminer si le graphe G admet un coloriage avec au moins k couleurs
- ❖Ce problème de décision est connu sous le nom de problème de K-coloriabilité.
- ❖Ce problème est aussi de classe NP

.

NP-Complétude | NP-complet

La classe NP-complet :
✓englobe les problèmes dont les solutions peuvent être vérifiées en temps polynomial, mais ces solutions prennent un temps exponentiel à être trouvées.
✓La classe NP-Complet regroupe les problèmes les plus difficiles de la classe NP.
✓Elle contient les problèmes de la classe NP tels que n'importe quel problème de la classe NP leur est polynomialement **réductible**.

✓Les problèmes NP-complets sont les problèmes les plus « difficiles » de NP.

11

NP-Complétude | NP-complet

Réductibilité
➢Soit A et B deux problèmes. Si A se réduit à B (noté **A ≤ B**)alors:
•Le problème A est plus facile que le problème B
•Le problème B est plus difficile que le problème A.

➢Intuitivement, un problème Q1 peut être ramené à (se réduit à) un problème Q2 si une instance quelconque x de Q1 peut être « facilement reformulée » comme une certaine instance y de Q2.
•Dans ce cas, la résolution du problème Q2(y) nous fournira la solution du problème Q1(x) et le problème Q1 n'est, dans un certain sens, « pas plus difficile à résoudre » que le problème Q2.
Exemple trivial : le problème de la résolution d'équations linéaires à une inconnue $(a \times x + b = 0)$ peut être ramenée à la résolution d'équations quadratiques $(a \times x^2 + b \times x + c = 0)$.
•Intuitivement, si un problème est plus facile qu'un problème polynomial alors il est polynomial.

12

NP-Complétude | NP-complet

Preuve de la NP-Complétude:

Pour prouver la NP-complétude d'un problème B il suffit de prouver que:

1. B est dans NP.
2. $A \leq B$ pour un problème A que l'on sait déjà NP-complet.

❖ La difficulté est d'arriver à en produire un premier problème NP-complet

Remarque

Un problème qui vérifie la propriété 2 mais pas nécessairement la propriété 1 est dit **NP-difficile.**

Exemple de problème NP-Complet:

– CLIQUE : un graphe donné contient-il une clique (un sous-graphe complet) de taille *k ?*

– CYCLE-HAMILTONIEN.

– VOYAGEUR-DE-COMMERCE : le voyageur de commerce veut faire la tournée d'un ensemble de villes (cycle hamiltonien) la plus courte possible.

– 3-COLORIAGE D'UN GRAPHE : peut-on colorier à l'aide de trois couleurs les sommets d'un graphe de sorte que deux sommets adjacents aient des couleurs différentes ?

NP-Complétude | Comparaison entre NP et P

Clairement, on sait que $P \subset NP$ mais la question qui se pose est ce que P = NP ?

✓On n'en sait rien.

✓C'est l'une des questions non résolues les plus célèbres qui défie les chercheurs depuis plus de 40 ans.

✓Intérêt: si P = NP alors tous les problèmes vérifiables polynomialement seraient décidables en temps polynomial.

✓La majorité des chercheurs pense que $P \neq NP$ car il y a un grand nombre de problèmes pour lesquels on n'arrive pas à produire d'algorithmes polynomiaux depuis plus de 40 ans

•Une des raisons qui laissent à penser que $P \neq NP$ est l'existence de la classe des problèmes *NP-complets :*

•Si un seul problème NP-complet peut être résolu en temps polynomial, alors tous les problèmes de NP peuvent être résolus en temps polynomial et P = NP.

•Mais aucun algorithme polynomial n'a jamais été découvert pour aucun problème NP-complet.