Master Informatique - M1 - UE Complexité
Chapitre 4 : Cadre formel

Philippe Jégou

Laboratoire d'Informatique et Systèmes - LIS - UMR CNRS 7020

Équipe COALA - COntraintes, ALgorithmes et Applications

(Algorithmique et Complexité de l'Intelligence Artificielle)

Campus de Saint-Jérôme

Département Informatique et Interactions

Faculté des Sciences

Université d'Aix-Marseille

philippe.jegou@univ-amu.fr

21 septembre 2022

En préambule

Ce qui a été vu

- **Problèmes de décision :**

  si un problème de décision est "difficile", alors les problèmes qui en découlent (recherche, optimisation, etc.) le sont

  $\Rightarrow$ **on commence par l'étude des problèmes de décision**

En préambule

Ce qui a été vu

- **Problèmes de décision :**

  si un problème de décision est "difficile", alors les problèmes qui en découlent (recherche, optimisation, etc.) le sont

  $\Rightarrow$ **on commence par l'étude des problèmes de décision**

- **Difficulté (i.e. complexité) d'un problème :**

  complexité du meilleur algorithme de résolution

En préambule

Ce qui a été vu

- **Problèmes de décision :**

  si un problème de décision est "difficile", alors les problèmes qui en découlent (recherche, optimisation, etc.) le sont

  $\Rightarrow$ **on commence par l'étude des problèmes de décision**

- **Difficulté (i.e. complexité) d'un problème :**

  complexité du meilleur algorithme de résolution

- **Complexité d'un algortihme :**

  fonction de la taille de la donnée en entrée

En préambule

Ce qui a été vu

- **Problèmes de décision :**

  si un problème de décision est "difficile", alors les problèmes qui en découlent (recherche, optimisation, etc.) le sont

  $\Rightarrow$ **on commence par l'étude des problèmes de décision**

- **Difficulté (i.e. complexité) d'un problème :**

  complexité du meilleur algorithme de résolution

- **Complexité d'un algortihme :**

  fonction de la taille de la donnée en entrée

- **Taille d'une donnée :**

  liée à son codage

En préambule

Ce qui va être vu :

- **Proposition d'un "système" de codage :**
  - raisonnable

En préambule

Ce qui va être vu :

- **Proposition d'un "système" de codage :**
  - raisonnable
  - permettant de coder tout type de données : des nombres, des noms, des graphes, des bases de données, etc.

En préambule

Ce qui va être vu :

- **Proposition d'un "système" de codage :**
  - raisonnable
  - permettant de coder tout type de données : des nombres, des noms, des graphes, des bases de données, etc.
  - dont la taille des objets codés sera précisément connue :
    - toute donnée sera représentée par un mot au sens des langages formels
    - taille d'une donnée en entree : longueur du mot la codant

En préambule

Ce qui va être vu :

- **Proposition d'un "système" de codage :**
  - raisonnable
  - permettant de coder tout type de données : des nombres, des noms, des graphes, des bases de données, etc.
  - dont la taille des objets codés sera précisément connue :
    - toute donnée sera représentée par un mot au sens des langages formels
    - taille d'une donnée en entree : longueur du mot la codant
- **Résolution d'un problème de décision :**
  - équivaut à un problème de reconnaissance de langage (formel)
  - question à traiter : est-ce qu'un mot appartient à un langage ?

En préambule

Ce qui va être vu :

- **Proposition d'un "système" de codage :**
  - raisonnable
  - permettant de coder tout type de données : des nombres, des noms, des graphes, des bases de données, etc.
  - dont la taille des objets codés sera précisément connue :
    - toute donnée sera représentée par un mot au sens des langages formels
    - taille d'une donnée en entree : longueur du mot la codant
- **Résolution d'un problème de décision :**
  - équivaut à un problème de reconnaissance de langage (formel)
  - question à traiter : est-ce qu'un mot appartient à un langage ?
- **Temps de calcul connu sans ambiguïté :**
  nombre de transitions nécessaires par la "machine" utilisée : automate fini, machine de Turing, etc.

Plan

1. Problèmes de décision : partition de l'ensemble des instances
2. Codage des instances : c'est facile avec des mots
3. Problèmes de décision : résolution par reconnaissance de langages

Plan

Problèmes de décision

Le problème de l'isomorphisme de sous-graphe

ISO-SOUS-GRAPHE
Donnée : Deux graphes non-orientés $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$
Question : G_1 contient-il un sous-graphe isomorphe à G_2 ?

Problèmes de décision

Le problème de l'isomorphisme de sous-graphe

ISO-SOUS-GRAPHE
Donnée : Deux graphes non-orientés $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$
Question : G_1 contient-il un sous-graphe isomorphe à G_2 ?

G_1 contient un sous-graphe $G' = (S', A')$ isomorphe à G_2 si
$G' = (S', A')$ sous-graphe de G_1 avec $S' \subseteq S_1$ et $A' \subseteq A_1$, tels que :

Problèmes de décision

Le problème de l'isomorphisme de sous-graphe

ISO-SOUS-GRAPHE
Donnée : Deux graphes non-orientés $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$
Question : G_1 contient-il un sous-graphe isomorphe à G_2 ?

G_1 contient un sous-graphe $G' = (S', A')$ isomorphe à G_2 si
$G' = (S', A')$ sous-graphe de G_1 avec $S' \subseteq S_1$ et $A' \subseteq A_1$, tels que :

- $|S'| = |S_2|$ et $|A'| = |A_2|$, et

Problèmes de décision

Le problème de l'isomorphisme de sous-graphe

ISO-SOUS-GRAPHE
Donnée : Deux graphes non-orientés $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$
Question : G_1 contient-il un sous-graphe isomorphe à G_2 ?

G_1 contient un sous-graphe $G' = (S', A')$ isomorphe à G_2 si
$G' = (S', A')$ sous-graphe de G_1 avec $S' \subseteq S_1$ et $A' \subseteq A_1$, tels que :

- $|S'| = |S_2|$ et $|A'| = |A_2|$, et
- $\exists$ une bijection $\phi : S_2 \rightarrow S'$ vérifiant $\{x, y\} \in A_2 \Leftrightarrow \{\phi(x), \phi(y)\} \in A'$

Problèmes de décision

Le problème de l'isomorphisme de sous-graphe

ISO-SOUS-GRAPHE
Donnée : Deux graphes non-orientés $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$
Question : G_1 contient-il un sous-graphe isomorphe à G_2 ?

G_1 contient un sous-graphe $G' = (S', A')$ isomorphe à G_2 si
$G' = (S', A')$ sous-graphe de G_1 avec $S' \subseteq S_1$ et $A' \subseteq A_1$, tels que :

- $|S'| = |S_2|$ et $|A'| = |A_2|$, et
- $\exists$ une bijection $\phi : S_2 \rightarrow S'$ vérifiant $\{x, y\} \in A_2 \Leftrightarrow \{\phi(x), \phi(y)\} \in A'$

Intérêt pratique : Est-ce que la "forme" représentée par G_2 se trouve dans G_1 ? (applications pour la recherche de motifs structurels).

Problèmes de décision

Le problème de l'isomorphisme de sous-graphe

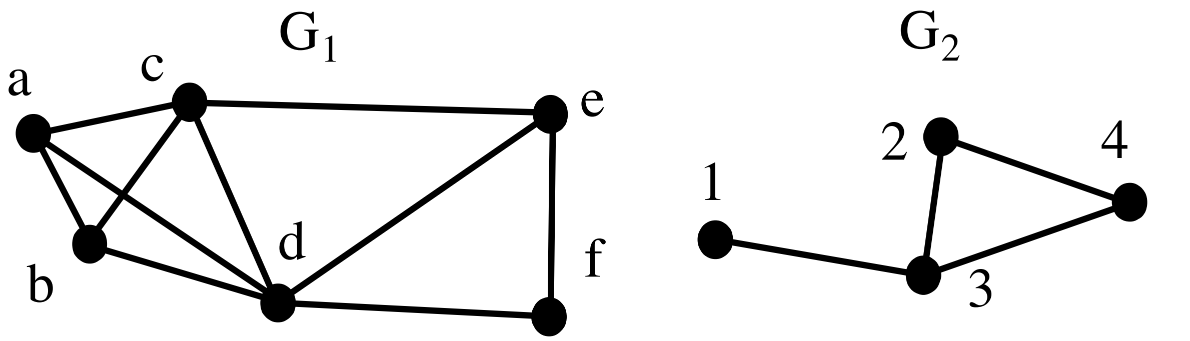
ISO-SOUS-GRAPHE
Donnée : Deux graphes non-orientés $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$
Question : G_1 contient-il un sous-graphe isomorphe à G_2 ?

G_1 contient un sous-graphe $G' = (S', A')$ isomorphe à G_2 si
$G' = (S', A')$ sous-graphe de G_1 avec $S' \subseteq S_1$ et $A' \subseteq A_1$, tels que :

- $|S'| = |S_2|$ et $|A'| = |A_2|$, et
- $\exists$ une bijection $\phi : S_2 \to S'$ vérifiant $\{x, y\} \in A_2 \Leftrightarrow \{\phi(x), \phi(y)\} \in A'$

Intérêt pratique : Est-ce que la "forme" représentée par G_2 se trouve dans G_1 ? (applications pour la recherche de motifs structurels).

Exemple d'instance du problème ISO-SOUS-GRAPHE : un couple (G_1, G_2)



La réponse à la question du problème de décision ici est...

Problèmes de décision

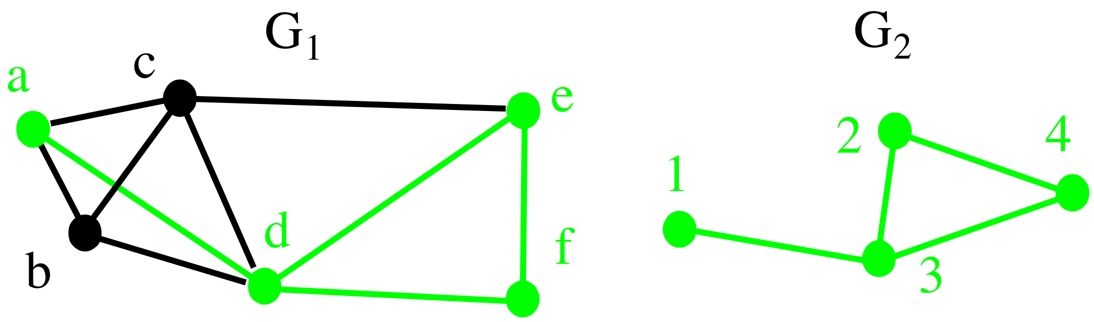
Le problème de l'isomorphismes de sous-graphe

ISO-SOUS-GRAPHE
Donnée : Deux graphes non-orientés $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$
Question : G_1 contient-il un sous-graphe isomorphe à G_2 ?

G_1 contient un sous-graphe isomorphe à G_2 si $\exists S' \subseteq S_1$ et $\exists A' \subseteq A_1$ tels que :

- $|S'| = |S_2|$ et $|A'| = |A_2|$, et
- $\exists$ une bijection $\phi : S_2 \to S'$ vérifiant $\{x, y\} \in A_2 \Leftrightarrow \{\phi(x), \phi(y)\} \in A'$

Exemple d'instance du problème ISO-SOUS-GRAPHE : un couple (G_1, G_2)



La réponse à la question du problème de décision ici est **OUI**

$S' = \{a, d, e, f\}$ et $A' = \{\{a, d\}, \{d, e\}, \{d, f\}, \{e, f\}\}$
et la bijection ϕ est $\phi(1) = a, \phi(2) = e, \phi(3) = d, \phi(4) = f$

Problèmes de décision

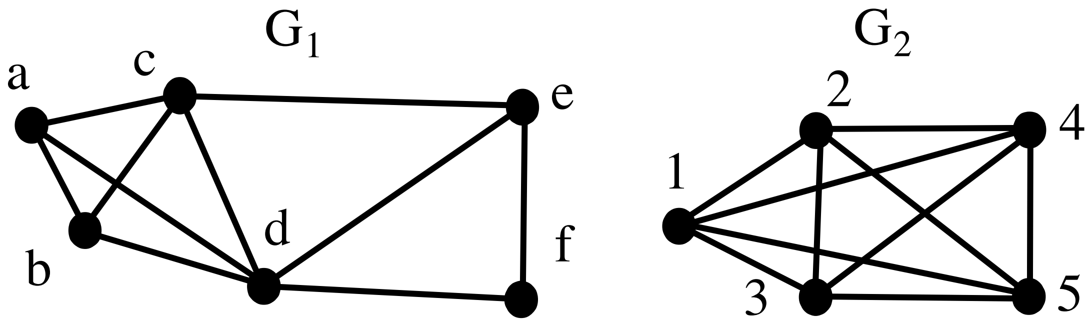
Le problème de l'isomorphismes de sous-graphe

ISO-SOUS-GRAPHE
Donnée : Deux graphes non-orientés $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$
Question : G_1 contient-il un sous-graphe isomorphe à G_2 ?

G_1 contient un sous-graphe isomorphe à G_2 si $\exists S' \subseteq S_1$ et $\exists A' \subseteq A_1$ tels que :

- $|S'| = |S_2|$ et $|A'| = |A_2|$, et
- $\exists$ une bijection $\phi : S_2 \to S'$ vérifiant $\{x, y\} \in A_2 \Leftrightarrow \{\phi(x), \phi(y)\} \in A'$

Exemple d'instance du problème ISO-SOUS-GRAPHE : un couple (G_1, G_2)



Si la question est posée avec un autre graphe G_2

qui est un graphe complet à 5 sommets :

La réponse à la question du problème de décision ici est **NON**

Problèmes de décision

L'ensemble des instances du problème ISO-SOUS-GRAPHE peut se partitionner en

- 1 sous-ensemble des **instances positives** : la réponse à la question du problème de décision pour ces instances est **OUI**
- 1 sous-ensemble des **instances négatives** : la réponse à la question du problème de décision pour ces instances est **NON**

Problèmes de décision

L'ensemble des instances du problème ISO-SOUS-GRAPHE peut se partitionner en

- 1 sous-ensemble des **instances positives** : la réponse à la question du problème de décision pour ces instances est **OUI**
- 1 sous-ensemble des **instances négatives** : la réponse à la question du problème de décision pour ces instances est **NON**

Plus généralement, pour tout problème de décision π on a :

- D_π : l'ensemble des instances du problème π

qui contient deux sous-ensembles disjoints :

- V_π : l'ensemble des instances positives du problème π ($V_\pi \subseteq D_\pi$)
- $D_\pi \setminus V_\pi$: l'ensemble des instances négatives du problème π

Problèmes de décision

L'ensemble des instances du problème ISO-SOUS-GRAPHE peut se partitionner en

- 1 sous-ensemble des **instances positives** : la réponse à la question du problème de décision pour ces instances est **OUI**
- 1 sous-ensemble des **instances négatives** : la réponse à la question du problème de décision pour ces instances est **NON**

Plus généralement, pour tout problème de décision π on a :

- D_π : l'ensemble des instances du problème π

qui contient deux sous-ensembles disjoints :

- V_π : l'ensemble des instances positives du problème π ($V_\pi \subseteq D_\pi$)
- $D_\pi \setminus V_\pi$: l'ensemble des instances négatives du problème π

Conséquence : résoudre un problème de décision π, étant donnée une instance $I \in D_\pi$ consiste à savoir :

- si $I \in V_\pi$ (instance positive : la réponse à la question est oui)
- si $I \in D_\pi \setminus V_\pi$ (instance négative : la réponse à la question est non)