L3-INFO Algorithmique et Modélisation

UFR IM²AG Université Grenoble Alpes

Examen session 1 – 26 mai 2022 – durée 3 h

Sont interdits : les documents, les ordinateurs, les téléphones (incluant smartphone, tablettes,... tout ce qui contient un dispositif électronique).
Seuls les dictionnaires papier pour les personnes de langue étrangère sont autorisés.
En cas de doutes sur l'énoncé, préciser les choix que vous faites sur votre copie et continuer.
Il sera tenu compte de la qualité de la rédaction et de la clarté de la présentation (2 pts).
Le barème indicatif : Exercice I : 4 points ; Problème II.1 : 4 points II.2 : 4 points II.3 : 4 points II.4 : 4 points
L'exercice et les différentes questions du problème peuvent être traités indépendamment.
Les durées sont indicatives, penser à se relire.

Exercice : Bouger la caisse ... *($\sim$ 40 mn)*

Dans un petit jeu traditionnel, un pousseur doit rejoindre une caisse par un chemin le moins coûteux... Le problème se modélise par un graphe connexe, orienté, avec une pondération positive des arcs. On note le graphe $\mathcal{G} = (\mathcal{X}, \mathcal{A}, \mathcal{C})$ avec

$$\mathcal{X} = \{x_1, \cdots, x_n\}.$$

La relation d'adjacence $\mathcal{A}$ est donnée par les voisins successeurs des sommets, on note $\mathcal{V}_i$ l'ensemble des successeurs de x_i, c'est à dire que $x_j \in \mathcal{V}_i$ si et seulement si il existe dans $\mathcal{G}$ un arc de x_i vers x_j. On note m le nombre d'arcs du graphe.

$\mathcal{C} = ((c_{i,j}))$ est une pondération strictement positive des arcs, $c_{i,j} > 0$ est le coût de l'arc $(x_i, x_j) \in \mathcal{A}$ par convention on prendra $c_{i,j} = +\infty$ si $x_j \notin \mathcal{V}_i$.

La position initiale du pousseur est notée s (comme source) et la position de la caisse t (comme target). Un algorithme a été implémenté pour associer à chaque sommet x_i du graphe le coût minimal d_i d'un chemin de x_i à t.

I.1. Proposer un algorithme en $\mathcal{O}(m)$ vérifiant que, pour chaque sommet x_i, la valeur d_i est bien le coût minimal d'un chemin de x_i à t. *Indication : trouver une propriété que doit vérifier l'ensemble des d_i et montrer que cette propriété est suffisante pour assurer un coût minimal.*

I.2. En quoi l'hypothèse de *coût strictement positif* est-elle nécessaire ?

I.3. Écrire l'algorithme de routage du pousseur, c'est à dire que lorsque le pousseur est en position x_i la fonction `routage` (x_i, t) fournit le sommet successeur de x_i sur un chemin de coût minimal pour rejoindre t.

Problème : Stabilité dans un graphe ...

Soit $\mathcal{G} = (\mathcal{X}, \mathcal{E}, \mathcal{P})$ un graphe non-orienté pondéré. On note $\mathcal{X} = \{x_1, x_2, \cdots, x_n\}$ les n sommets du graphe, les arêtes $\mathcal{E}$ sont données par

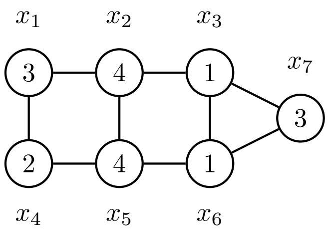
$$e_{i,j} = \begin{cases} Vrai & \text{s'il existe une arête entre } x_i \text{ et } x_j ; \\ Faux & \text{sinon.} \end{cases}$$

Les sommets sont pondérés par des valeurs positives $\mathcal{P} = \{p_1, p_2, \cdots, p_n\}$ (p_i est le poids du sommet x_i). On définit le poids d'un sous-ensemble $\mathcal{S}$ de sommets comme la somme des poids de chacun des sommets de $\mathcal{S}$.

Un sous-ensemble $\mathcal{S}$ de sommets est dit **stable** si et seulement si $\mathcal{S}$ ne contient pas 2 sommets qui sont voisins.

II.1. **Analyse à la main sur l'exemple de la figure 1a** (page suivante) *($\sim$ 20 mn)*

(a) Combien y a-t'il de sous-ensembles différents de $\mathcal{X}$?



(a) Graphe pondéré

$\mathcal{S} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$

	Partie	poids	
S_1	$\{x_1, x_6\}$	4	stable
S_2	$\{x_2, x_4, x_7\}$	9	stable
S_3	$\{x_2, x_5, x_7\}$	11	pas stable

Attention : la partie S_3 n'est pas stable car x_2 et x_5 sont voisins.

(b) Sous-ensembles stables

FIGURE 1 – Exemple de quelques sous-ensembles stables

Solution: Un sommet donné peut être sélectionné ou non, il y a donc $2^{|\mathcal{X}|} = 2^7 = 128$ sous-ensembles possibles.

(b) Quels sont les sous-ensembles stables de $\mathcal{S}$?

(c) Déterminer le (ou les) sous-ensemble(s) stable(s) de poids maximum.

Solution: Tout sous-ensemble d'un ensemble stable est également stable.
Ici on a :

Cardinal	Sous-ensembles stables
1	$\{x_1\}, \{x_2\}, \cdots \{x_7\}$
2	$\{x_1, x_3\}, \{x_1, x_6\}, \{x_1, x_7\}, \{x_2, x_4\}, \{x_2, x_6\}$, $\{x_2, x_7\}, \{x_3, x_4\}, \{x_3, x_5\}, \{x_4, x_6\}, \{x_5, x_7\}$
3	$\{x_1, x_3, x_5\}, \{x_1, x_5, x_7\}, \{x_2, x_4, x_6\}, \{x_2, x_4, x_7\}$

Les sous-ensembles de poids maximal seront nécessairement des sous-ensembles pour lesquels on ne peut rajouter de sommet en restant stable (les poids étant positifs). Les sous-ensembles de cardinal 3 sont dans ce cas, et les sous-ensembles de cardinal 2 non inclus dans un sous-ensemble de cardinal 3. Les sous-ensembles de cardinal 1 ne peuvent être de poids maximal dans notre exemple.

Partie	poids
$\{x_1, x_6\}$	4
$\{x_3, x_4\}$	3
$\{x_1, x_3, x_5\}$	8
$\{x_1, x_5, x_7\}$	10
$\{x_2, x_4, x_6\}$	7
$\{x_2, x_4, x_7\}$	9

Il y a donc un seul sous-ensemble de poids maximal $\{x_1, x_5, x_7\}$ de poids 10.

Dans le cas général d'un graphe, on propose de résoudre le problème en plusieurs étapes. Un petit futé qui a lu l'ouvrage de Cormen et al. a découvert que le problème de l'existence d'un sous-ensemble stable de cardinal k dans un graphe est un problème de la classe $\mathcal{NP}$. Il y a donc peu de chance de pouvoir résoudre ce problème sans utiliser la force brute de l'énumération. Il propose donc de transformer l'algorithme d'énumération des parties

procédure `énumérer_visiter_parties` (X,S)
Données : X et S ensembles disjoints d'éléments
Résultat : Une et une seule visite de chaque sous-ensemble de $X \cup S$ de la forme $p \cup S$ avec $p \subset X$

if $X = \emptyset$
  `Visiter` (S)
else
  x =`Choisir` (X)
  `énumérer_visiter_parties` ($X \backslash \{x\}$,S)
  `énumérer_visiter_parties` ($X \backslash \{x\}$,$S \cup \{x\}$)

II.2. **Amélioration de l'algorithme d'énumération** *($\sim$ 40 mn)*

(a) Quel est l'appel initial ?

(b) Sans aucune modification, donner le coût de cet algorithme en nombre d'appels à `énumérer_visiter_parties`

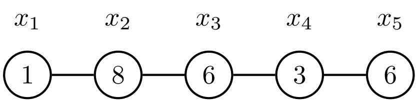
(c) Proposer une condition de branchement ligne 2 pour garantir qu'un sous-ensemble n'est visité que s'il est stable. Déterminer l'invariant associé à cette propriété et qu'il est vérifié sur toutes les branches de l'arbre des appels récursifs.

(d) Proposer une condition de branchement ligne 1 qui garantisse que lorsque l'on visite un sous-ensemble, il est de poids plus grand que les poids de chacun des sous-ensembles précédemment visités.

(e) Donner une version finale de l'algorithme, on évitera, autant que possible, la recopie de paramètres lors des appels récursifs (passage par référence).

II.3. **Réduction du problème à un schéma classique en simplifiant le graphe.** *($\sim$ 40 mn)*

La complexité de l'algorithme ci-dessus pouvant être élevée, on propose de restreindre le problème à un graphe en ligne. Par exemple



(a) Exemple de ligne pondérée

$\mathcal{S} = \{x_1, x_2, x_3, x_4, x_5\}$

	Partie	poids	
S_1	$\{x_1, x_4\}$	4	stable
S_2	$\{x_1, x_3, x_5\}$	13	stable
S_3	$\{x_2, x_4\}$	11	stable

(b) Exemples de sous-ensembles stables de la ligne

FIGURE 2 – Graphe ligne de 5 sommets

On note M_k le poids maximal d'une partie stable du sous graphe construit à partir des sommets $\{x_1, x_2, \cdots, x_k\}$ (ligne de x_1 à x_k) avec $1 \leqslant k \leqslant n$. Par convention on notera $M_0 = 0$.

(a) Écrire une équation de récurrence vérifiée par les variables M_k.

(b) Écrire un algorithme récursif qui calcule M_n (approche top-down). Détailler soigneusement le cas de base.

(c) (Bonus) Quel est le nombre d'appels récursifs effectués par cet algorithme ?

(d) Réécrire cet algorithme grâce à une approche bottom-up pour calculer itérativement les M_k.

(e) Illustrer le fonctionnement de cet algorithme sur l'exemple de la figure 2.

(f) Modifier l'algorithme de manière à obtenir le sous-ensemble stable de poids maximal.

(g) L'approche utilisée pour un graphe ligne peut-elle être généralisée à un graphe quelconque. Justifier votre réponse.

II.4. $\alpha(\mathcal{G})$ **taille du plus grand sous-ensemble stable de sommets (en nombre de sommets)** *($\sim$ 40 mn)*

La recherche d'un sous-ensemble stable de cardinal maximal d'un graphe $\mathcal{G} = (\mathcal{X}, \mathcal{E})$ est un problème de la classe $\mathcal{NP}$ donc "difficile". L'objectif de cet exercice est de donner un minorant de $\alpha(G)$ à partir des caractéristiques du graphe. Cela correspond à fixer les poids de tous les sommets à 1. On note n le nombre de sommets du graphe, m le nombre d'arêtes et d_i le degré du sommet x_i, c'est à dire le nombre de ses voisins.

On définit naturellement d le degré moyen du graphe par

$$d = \frac{1}{n}(d_1 + d_2 + \cdots + d_n)$$

(a) Quel est le lien entre n, d et m ?

On va utiliser une technique probabiliste pour montrer que

$$\alpha(\mathcal{G}) \geqslant \frac{n}{2d}.$$

On va raisonner en plusieurs étapes, d'abord sélectionner aléatoirement un sous-ensemble $\mathcal{S}_0$ de sommets, compter son nombre moyen de sommets, son nombre moyen d'arêtes et puis extraire d'un $\mathcal{S}_0$ particulier un sous-ensemble stable d'au moins $\frac{n}{2d}$ sommets.
Pour construire $\mathcal{S}_0$, pour chaque de sommet x_i de $\mathcal{G}$ un tirage aléatoire à pile ou face selon une même probabilité p décidera si le sommet est placé dans $\mathcal{S}_0$ ou pas.
Pour modéliser ces choix, on note $X_1, X_2, \cdots, X_n$ les valeurs des tirages, les X_i sont des variables aléatoires indépendantes,

$$X_i = \begin{cases} 1 & \text{si } x_i \text{ est sélectionné et placé dans } \mathcal{S}_0 \text{ avec la probabilité } p; \\ 0 & \text{si } x_i \text{ n'est pas sélectionné avec la probabilité } 1-p. \end{cases}$$

On note N le cardinal de $\mathcal{S}_0$.

(b) Quelle est la loi de probabilité de N ? Calculer $\mathbb{E}N$.

Une arête est dite sélectionnée si ses 2 extrémités sont dans $\mathcal{S}_0$

(c) Calculer la probabilité qu'une arête soit sélectionnée. Soit E_0 le nombre d'arêtes sélectionnées, calculer $\mathbb{E}E_0$

(d) Montrer que $\mathbb{E}(N - E_0)$ est maximal pour une probabilité de sélection $p = \frac{1}{d}$ et que, pour cette valeur,

$$\mathbb{E}(N - E_0) = \frac{n}{2d}.$$

(e) En déduire qu'il existe un ensemble $\mathcal{S}_0^*$ particulier tel que le nombre de sommets de $\mathcal{S}_0^*$ excède le nombre d'arêtes de $\mathcal{S}_0^*$ d'au moins $\frac{n}{2d}$.

(f) Montrer qu'il existe un sous-ensemble stable de $\mathcal{G}$ contenu dans $\mathcal{S}_0^*$ ayant au moins $\frac{n}{2d}$ sommets.

(g) Calculer le minorant obtenu pour le graphe de la figure 1a et commenter ce résultat.