

ESPACE DES CYCLES D'UN GRAPHE (ENS 2008)

Durée : 4 heures

Ce problème s'intéresse à la structure cyclique de certains graphes. La première partie porte sur le calcul d'un arbre couvrant et des cycles fondamentaux d'un graphe non orienté. La seconde partie étudie plusieurs algorithmes de calcul d'une base minimale de l'espace des cycles d'un graphe non orienté et pondéré. La seconde partie pourra être largement abordée même si la première n'a pas été complètement résolue.

Préliminaires

Pseudo-programmes, structures de données et algorithmes. Pour les questions demandant d'écrire un *pseudo-programme*, on utilisera un langage ou pseudo-langage au choix, avec les structures de données (tableaux uni- et multi-dimensionnels, listes, files, piles, etc.) et de contrôle (si, tant que, pour, etc.) usuelles. Les indices d'un tableau **tab** de taille t vont de 1 à t , et ses éléments seront notés **tab**[1], ..., **tab**[t]. Les déclarations de variables ne sont pas imposées. Les questions demandant de donner un algorithme n'imposent pas un pseudo-programme mais une description concise et précise, en français, de l'algorithme. Dans les deux cas, la qualité de la rédaction ainsi que les justifications (correction, coût) seront des éléments d'appréciation.

Estimations de coût. Le coût d'un algorithme ou d'un pseudo-programme est le nombre d'opérations élémentaires qu'il effectue dans le cas le pire : lecture ou écriture dans une variable ou une case de tableau, opération arithmétique (notamment : comparaison, addition, soustraction, multiplication) sur des entiers ou sur des éléments d'un corps, test, ajout ou suppression en tête de liste, empilement ou dépilement dans une pile, etc. On ne cherchera pas à calculer les coûts demandés exactement. Ils seront seulement estimés en ordre de grandeur, avec des expressions du type $O(m+n)$, $O(mn^2 \log n)$, etc., où m, n, \dots sont par exemple des paramètres en entrée de l'algorithme ou du pseudo-programme.

Autres conventions. On note \mathbb{N} l'ensemble des entiers naturels. Si x est un nombre réel alors $\lfloor x \rfloor$ désigne le plus grand entier inférieur ou égal à x . Pour $j \in \mathbb{N}$ et $k \in \mathbb{N} \setminus \{0\}$, l'entier naturel égal à $j - \lfloor j/k \rfloor k$ sera noté $j \bmod k$. Pour un corps \mathbb{K} , on note $\mathbb{K}^{p \times q}$ l'ensemble des matrices à p lignes et q colonnes, dont les éléments sont dans \mathbb{K} . Ces matrices pourront être représentées par des tableaux bi-dimensionnels.

Définitions et notations

Les graphes considérés dans ce problème sont finis et non orientés. Pour $n \in \mathbb{N} \setminus \{0\}$ et $m \in \mathbb{N}$, un graphe (fini, non orienté) à n sommets et m arêtes est un couple (S, A) où S , l'ensemble des sommets du graphe, est identifié à $\{1, 2, \dots, n\}$, et où A , l'ensemble de ses arêtes, est un sous-ensemble de $\{\{s, t\} \mid s \in S, t \in S, s \neq t\}$ de cardinal m . Une arête $\{s, t\} \in A$ sera notée indistinctement (s, t) ou (t, s) ; on dira que s est un *voisin* de t dans le graphe et que t est un voisin de s dans le graphe.

On appelle chaîne de longueur λ reliant s à t dans le graphe (S, A) toute suite $(s_0, s_1, s_2, \dots, s_\lambda)$ de sommets du graphe tels que $s_0 = s$, $s_\lambda = t$ et $(s_{i-1}, s_i) \in A$ pour $1 \leq i \leq \lambda$. Un graphe est *connexe* si pour chaque paire $\{s, t\}$ de sommets il existe au moins une chaîne reliant s à t dans ce graphe. On dira qu'une chaîne est *élémentaire* si de plus $s_0, s_1, \dots, s_\lambda$ sont distincts deux à deux.

Une chaîne $(s_0, s_1, s_2, \dots, s_\lambda)$ dans le graphe forme un *cycle élémentaire* du graphe si $s_0 = s_\lambda$, $\lambda \geq 3$ et $s_1, s_2, \dots, s_\lambda$ sont distincts deux à deux. Soient $(s_i)_{0 \leq i \leq \lambda}$ et $(s'_i)_{0 \leq i \leq \lambda}$ deux chaînes de même longueur et formant chacune un cycle élémentaire du graphe; s'il existe $j \in \mathbb{N}$ tel que, pour $1 \leq i \leq \lambda$, $s'_i = s_{(i+j) \bmod \lambda}$ alors on dira que ces deux chaînes forment le même cycle élémentaire.

Soit $G = (S, A)$ un graphe connexe. On appelle *arbre couvrant* de G tout graphe (S', A') connexe, sans cycle élémentaire, et tel que $S' = S$ et $A' \subset A$. Étant donné un arbre couvrant $T = (S, A')$ de G et une arête $a = (s, t) \in A \setminus A'$ (si elle existe), le *cycle fondamental* de G par rapport à a , noté φ_a , est le cycle élémentaire de G obtenu en « fermant » avec l'arête a la chaîne élémentaire reliant s à t dans T . L'ensemble des cycles fondamentaux de G par rapport à T est $\varphi(G, T) = \{\varphi_a \mid a \in A \setminus A'\}$.

Dans toute la suite, G désigne un graphe connexe à n sommets et m arêtes. On supposera de plus que G est donné par sa structure d'adjacence, c'est-à-dire par un tableau **Adj** de n listes d'entiers tel que, pour $s \in \{1, \dots, n\}$, **Adj**[s] est la liste (dans un ordre arbitraire) des voisins de s dans G . Les pseudo-programmes et les algorithmes demandés pourront exploiter toutes ces hypothèses sur G .

Partie 1. Arbres couvrants et cycles fondamentaux

Question 1.1

1. (a) Montrer que le nombre de cycles élémentaires de G est fini.
(b) Montrer que G admet au moins un arbre couvrant.
2. Montrer que tout arbre couvrant de G a exactement $n - 1$ arêtes.
3. Écrire un pseudo-programme **SansCycleElem** qui, étant donné G , détermine si G ne contient aucun cycle élémentaire. Pourquoi est-il correct et quel est son coût ?

Question 1.2

1. Donner la structure d'adjacence d'un arbre couvrant de $G = (S, A)$ pour $S = \{1, 2, 3, 4, 5\}$ et $A = \{(1, 2), (1, 3), (1, 4), (2, 3), (3, 4), (3, 5), (4, 5)\}$.
2. Écrire un pseudo-programme **ArbreCouvrant** qui, étant donné G , calcule la structure d'adjacence d'un arbre couvrant de G . Pourquoi est-il correct et quel est son coût ?

Question 1.3

1. Vérifier l'existence et l'unicité de la chaîne élémentaire utilisée à la fin de la partie **définitions et notations** pour définir le cycle fondamental φ_a .
2. Étant donné T un arbre couvrant de G , on note L_T la somme des longueurs des éléments de $\varphi(G, T)$.
 - (a) Majorer L_T en fonction de n .
 - (b) On suppose ici que $m = \frac{n(n-1)}{2}$. En calculant la valeur de L_T pour deux arbres couvrants de G , montrer que l'ordre de grandeur de L_T dépend de T .
3. (a) Écrire un pseudo-programme **Phi** qui, étant donné G , calcule un ensemble de cycles fondamentaux de G (c'est-à-dire $\varphi(G, T)$ pour T un arbre couvrant quelconque de G). Pourquoi est-il correct ?
(b) Exprimer, en le justifiant, le coût de **Phi** en fonction de m et de la somme des longueurs des cycles fondamentaux calculés.

Partie 2. Bases minimales de l'espace des cycles

Dans cette partie, on suppose que G possède au moins un cycle élémentaire et qu'il est de plus pondéré. Un graphe est *pondéré* si à chaque arête a est associé un réel positif ou nul, noté $w(a)$ et appelé *poids* de l'arête a . Dans un graphe pondéré,

le poids $w(\sigma)$ d'une chaîne $\sigma = (s_i)_{0 \leq i \leq \lambda}$ est nul si $\lambda = 0$, et égal à $\sum_{i=1}^{\lambda} w((s_{i-1}, s_i))$ si $\lambda \geq 1$. De plus, pour deux sommets s et t

d'un graphe pondéré, on définit $\delta(s, t)$ de la façon suivante : s'il n'existe pas de chaîne reliant s à t dans le graphe alors $\delta(s, t) = +\infty$; sinon,

$$\delta(s, t) = \min\{w(\sigma) \mid \sigma \text{ est une chaîne reliant } s \text{ à } t \text{ dans le graphe}\}.$$

On note \mathbb{K} le corps fini à deux éléments, 0 et 1. On suppose connu $T = (S, A')$ un arbre couvrant de $G = (S, A)$, et on numérote les arêtes de G de sorte que $A = \{a_1, a_2, \dots, a_m\}$ et $A \setminus A' = \{a_1, a_2, \dots, a_N\}$ avec $N = m - n + 1$. On appelle *support* tout sous-ensemble non vide de $A \setminus A'$. Soit E l'ensemble de tous les sous-ensembles de A ; en particulier, E contient A et l'ensemble vide, noté $\{\}$. On appelle *vecteur d'incidence* d'un élément e de E l'unique vecteur $v = (v_i)_{1 \leq i \leq m}$ de \mathbb{K}^m tel que, pour $1 \leq i \leq m$, $v_i = 1$ si $a_i \in e$ et $v_i = 0$ sinon. On identifiera E au \mathbb{K} -espace vectoriel \mathbb{K}^m en identifiant chaque élément de E à son vecteur d'incidence. Pour $e \in E$ et $f \in E$, on notera $\langle e \mid f \rangle$ leur produit scalaire dans la base canonique

$$\{\{a_1\}, \{a_2\}, \dots, \{a_m\}\} : \text{ si } e = \sum_{i=1}^m e_i \{a_i\} \text{ et } f = \sum_{i=1}^m f_i \{a_i\} \text{ avec } e_i \in \mathbb{K} \text{ et } f_i \in \mathbb{K} \text{ pour } 1 \leq i \leq m, \text{ alors } \langle e \mid f \rangle = \sum_{i=1}^m e_i f_i \in \mathbb{K}.$$

Si $C = (s_i)_{0 \leq i \leq \lambda}$ est un cycle élémentaire de G , on le considère dans cette partie représenté par ses arêtes et on écrira $C = \{(s_{i-1}, s_i)\}_{1 \leq i \leq \lambda} \in E$. On appelle *cycle* de G un cycle élémentaire de G ou une union de cycles élémentaires de G disjoints deux à deux. L'*espace des cycles* de G , noté $\mathcal{C}(G)$, est le \mathbb{K} -sous-espace vectoriel de E engendré par l'ensemble des cycles de G . Le *poids d'un cycle* C de G est $w(C) = \sum_{a \in C} w(a)$, la somme des poids des arêtes qui composent ce cycle ; le *poids d'une base*

B de $\mathcal{C}(G)$ est $w(B) = \sum_{C \in B} w(C)$, la somme des poids des cycles qui composent cette base. Une *base minimale de l'espace des cycles* de G est une base de $\mathcal{C}(G)$ de poids minimal parmi toutes les bases de $\mathcal{C}(G)$.

Question 2.1

1. Soient $e \in E$ et $f \in E$.

(a) Soit $\alpha \in \mathbb{K}$. Quel sous-ensemble de A chacune des trois expressions αe , $-e$, $e + f$ représente-t-elle ?

(b) Caractériser en termes d'arêtes communes le fait que $\langle e | f \rangle = 1$.

2. Soit S un support. Montrer qu'il existe $C \in \mathcal{C}(G)$ tel que $\langle C | S \rangle = 1$.

3. Montrer que $\varphi(G, T)$ est une base de $\mathcal{C}(G)$.

On appelle **BaseMinimale1** l'algorithme décrit dans l'encadré ci-dessous :

pour i de 1 à N faire

$S_i \leftarrow$ un support tel que $\langle C_k | S_i \rangle = 0$, $1 \leq k \leq i - 1$

$C_i \leftarrow$ un cycle de G de poids minimal tel que $\langle C_i | S_i \rangle = 1$

Dans toute la suite et à l'exception de la question 2.5, S_i et C_i sont spécifiés comme à l'itération i de **BaseMinimale1**.

Question 2.2 Dans cette question, on suppose que S_1, \dots, S_N et C_1, \dots, C_N ont pu être trouvés par **BaseMinimale1** et on étudie certaines propriétés des C_i .

1. Pour $2 \leq i \leq N$, montrer que C_i est linéairement indépendant de C_1, \dots, C_{i-1} .

2. Montrer que $\{C_1, \dots, C_N\}$ est une base minimale de $\mathcal{C}(G)$.

Question 2.3 On s'intéresse dans cette question au coût du calcul de S_1, \dots, S_N indépendamment de C_1, \dots, C_N . (On ignorera donc le coût du calcul de C_1, \dots, C_N dans toute cette question.)

1. (a) Soit $i \in \{2, \dots, N\}$ fixé. Ecrire un pseudo-programme **Support** qui, étant donnés les vecteurs d'incidence des cycles C_1, \dots, C_{i-1} de G , calcule le vecteur d'incidence d'un support S_i tel que $\langle C_k | S_i \rangle = 0$ pour $1 \leq k \leq i - 1$.

(b) Donner en le justifiant le coût de **Support**. Quel coût total (en fonction de m) obtient-on alors pour $\{S_i\}_{1 \leq i \leq N}$ avec cette méthode ?

2. (a) Montrer comment calculer $\{S_i\}_{1 \leq i \leq N}$ en introduisant pour $1 \leq i \leq N$ des ensembles de supports $\{R_j^{(i)}\}_{i \leq j \leq N}$ qui vérifient les deux conditions suivantes :

- pour $1 \leq i \leq N$, $R_i^{(i)}, \dots, R_N^{(i)}$ sont linéairement indépendants ;

- $\langle C_k | R_j^{(i)} \rangle = 0$ pour $1 \leq k < i \leq j \leq N$ et $2 \leq i \leq N$.

(b) Réécrire l'algorithme **BaseMinimale1** de façon à ce que l'itération i calcule S_i , C_i et, si $i < N$, $R_{i+1}^{(i+1)}, \dots, R_N^{(i+1)}$. Quel nouveau coût total (en fonction de m) obtient-on pour $\{S_i\}_{1 \leq i \leq N}$?

Question 2.4 Soit $i \in \{1, \dots, N\}$ fixé. Dans cette question, on associe à $G = (S, A)$ et à S_i le *graphe signé* G_i défini de la façon suivante. Le graphe G_i a $2n$ sommets distincts, obtenus en dupliquant les n sommets de G : pour $s \in S$, on notera s^+ et s^- les deux sommets de G_i correspondants ; de plus, pour toute arête (s, t) de G :

- si $(s, t) \notin S_i$ alors G_i possède les deux arêtes (s^+, t^+) et (s^-, t^-) , chacune de poids $w((s, t))$;

- si $(s, t) \in S_i$ alors G_i possède les deux arêtes (s^+, t^-) et (s^-, t^+) , chacune de poids $w((s, t))$.

1. Soit $G = (S, A)$ où $S = \{1, 2, 3, 4, 5\}$ et $A = \{(2, 3), (4, 5), (1, 2), (1, 3), (1, 4), (1, 5)\}$. On suppose de plus que $w(a) = 0$ pour tout $a \in A$. Dessiner les sommets et les arêtes de G et G_i pour $S_i = \{(2, 3), (4, 5)\}$.

2. Pour $s \in S$ donné, soit $\sigma = (s_j)_{0 \leq j \leq \lambda}$ une chaîne reliant $s_0 = s^+$ à $s_\lambda = s^-$ dans G_i .

(a) Montrer qu'à σ correspond un cycle de G , qu'on notera C , tel que $w(C) \leq w(\sigma)$ et $\langle C | S_i \rangle = 1$.

(b) On suppose que σ est représenté de façon partielle par un tableau **chaîne** d'entiers tel que, pour $1 \leq j \leq \lambda$, $\text{chaîne}[j] = k$ si $s_{j-1} = s^\alpha$, $s_j = t^\beta$, $(s, t) = a_k$ et $\alpha, \beta \in \{+, -\}$. Écrire un pseudo-programme qui calcule le vecteur d'incidence de C à partir de **chaîne**. Quel est son coût ?

(c) Montrer que si $w(\sigma) = \min\{\delta(t^+, t^-) \mid t \in S\}$ alors $w(C) = \min\{w(D) \mid D \in \mathcal{C}(G) \text{ et } \langle D | S_i \rangle = 1\}$.

3. On suppose disposer d'un algorithme **PlusCourteChaîne** de coût $O(m + n \log n)$ qui, à partir de G , de S_i et d'un sommet s de G , calcule le couple (**distance**, **chaîne**) où **distance** = $\delta(s^+, s^-)$ et où **chaîne** est la représentation partielle (définie à la question 2.4.2.(b)) d'une chaîne σ reliant s^+ à s^- dans G_i et telle que $w(\sigma) = \delta(s^+, s^-)$.

(a) Écrire un pseudo-programme calculant le vecteur d'incidence de C_i à partir de G et S_i . Quel est son coût ?

(b) Conclure en exprimant le coût de l'algorithme **BaseMinimale1** en fonction de m et n .

Question 2.5 Dans cette question, ω désigne un nombre réel tel que $2 \leq \omega < 2.39$, et on suppose disposer d'un algorithme **MulMat** de coût $O(p^\omega)$ pour multiplier deux matrices de $\mathbb{K}^{p \times p}$.

1. (a) Soient $P \in \mathbb{K}^{p \times q}$ et $Q \in \mathbb{K}^{q \times p}$ avec $1 \leq p \leq q$. Donner un algorithme qui calcule le produit PQ et exprimer son coût en fonction de p , q et ω .
 (b) Soit $P \in \mathbb{K}^{p \times p}$ triangulaire et inversible. Soit $Q \in \mathbb{K}^{p \times p}$. Donner un algorithme qui calcule la matrice $X \in \mathbb{K}^{p \times p}$ telle que $PX = Q$, et exprimer son coût en fonction de p et ω .
2. Soient p et q dans \mathbb{N} tels que $p + 2q \leq N$. Soient C_1, \dots, C_{p+q} des cycles de G , soient $U_1, \dots, U_q, V_1, \dots, V_q$ des supports, et soit $M \in \mathbb{K}^{m \times 2q}$ la matrice dont la colonne j est égale au vecteur d'incidence de U_j si $1 \leq j \leq q$ et à celui de V_{j-q} si $q < j \leq 2q$. Pour $1 \leq \ell \leq q$, on suppose que les trois propriétés suivantes sont vérifiées :
 - $\langle C_k | U_\ell \rangle = 0$ pour $1 \leq k \leq p + \ell - 1$;
 - $\langle C_{p+\ell} | U_\ell \rangle = 1$;
 - $\langle C_k | V_\ell \rangle = 0$ pour $1 \leq k \leq p$.

On suppose de plus que U_1, \dots, U_q et V_1, \dots, V_q sont tels que la matrice M a la structure suivante : pour $1 \leq i \leq m$ et $1 \leq j \leq 2q$, l'élément situé sur la ligne i et la colonne j de M est nul si $i > p + j$ et égal à 1 si $i = p + j$.

- (a) Montrer, en introduisant un système linéaire de la forme $PX = Q$ avec P et Q deux matrices de $\mathbb{K}^{q \times q}$ dont on explicitera les éléments, que l'on peut transformer $\{V_\ell\}_{1 \leq \ell \leq q}$ en un autre ensemble de supports, noté $\{W_\ell\}_{1 \leq \ell \leq q}$ et tel que $\langle C_k | W_\ell \rangle = 0$ pour $1 \leq k \leq p + q$ et $1 \leq \ell \leq q$.
- (b) Proposer un algorithme **MiseAJour** qui calcule les vecteurs d'incidence de W_1, \dots, W_q à partir de ceux de $C_1, \dots, C_{p+q}, U_1, \dots, U_q, V_1, \dots, V_q$. Exprimer son coût en fonction de m, q et ω .
3. On suppose que $N = 2^v$ avec $v \in \mathbb{N}$ et on fait la même hypothèse qu'à la question 2.4.3.
 - (a) Proposer un algorithme **BaseMinimale2** qui calcule les vecteurs d'incidence d'une base minimale $\{C_1, \dots, C_N\}$ de $\mathcal{C}(G)$ à l'aide de **MiseAJour**.
 - (b) Montrer que le coût de **BaseMinimale2** est $O(m^2 n + mn^2 \log n)$. Pourquoi cet algorithme est-il correct ?

