
TD
d'Algo/Complexité/Calculabilité

Année 2009

Version 2.2

Université de Montpellier
Place Eugène Bataillon
34095 Montpellier Cedex 5

RODOLPHE GIROUDEAU
161, RUE ADA
34392 MONTPELLIER CEDEX 5
TEL : 04-67-41-85-40
MAIL : RGIROU@LIRMM.FR

1 Définitions et applications des flots

Exercice 1 – Applications des définitions

Vérifier chacune des trois propriétés de flot pour le flot f défini par : $f(sb) = f(ba) = 1$, $f(sa) = 3, f(ad) = f(ac) = f(dp) = f(cp) = 2, f(bd) = 4$.

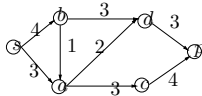


FIG. 1 – Un exemple de flot

Exercice 2 – Equivalence de définitions

Définition : Un flot f sur $G = (X, U)$ est une application $f : U \rightarrow R$ vérifiant

1. pour tout $x \in X : f(X, x) - f(x, X) = 0$ avec $f(X, x) = \sum_{(y,x) \in U} f(y, x)$ et $f(x, X) = \sum_{(x,y) \in U} f(x, y)$ (conservation du flot). La valeur $f(x, y)$ correspond au flux qui traverse l'arc (x, y) .
2. Pour tout $((x, y) \in U : f(x, y) \leq c(x, y)$
3. Pour tout $(x, y) \in U : f(x, y) \geq 0$

Pour définir le problème du flot maximal, on distingue 2 sommets s et p . On s'intéressera au flux maximum traversant l'arc ps éventuellement ajouté, ayant une capacité infinie. La valeur d'un flot est donnée par : $f(ps) = f(X, p) = f(s, X)$ Le but de cet exercice est de montrer que l'on peut à partir de la définition précédente et de la formulation du problème du flot maximal retrouver la formulation du cours. Dans un premier temps construisez un nouveau réseau avec une valuation f' et ensuite montrez qu'il y a équivalence des flots.

Considérons $G = (X, U)$ un graphe orienté, et une source s et un puits p . Trouver les arcs de $G' = (X, U - \{(p, s)\})$ une valuation f' vérifiant :

- $\forall x \in X - \{p, s\} : f'(X, x) - f'(x, X) = 0$
- $\forall (x, y) \in U - \{(p, s)\} : 0 \leq f'(x, y) \leq c(x, y)$
- $Max v(f') = f'(s, X) - f'(X, s) = f'(X, p) - f'(p, X)$ où $v(f')$ est la valeur du flot.

Exercice 3 – Propriété élémentaires des flots.

Montrer que si f est un flot sur $G = (X, U)$ alors $\forall Y \subseteq X, f(Y, \bar{Y}) - f(\bar{Y}, Y) = 0$ (avec la définition de l'exercice 2).

Exercice 4 – Propriété élémentaires des flots

Soient f et f' deux flots sur $G = (X, U)$. Alors $h = f + f'$ est un flot sur G .

Exercice 5 – Propriétés des flots **Définition :** Un flot f sur $G = (S, A)$ est une application $f : S \times S \rightarrow R$ vérifiant

1. pour tout $u \in S - \{s, t\}$ on a nécessairement : $\sum_{v \in S} f(u, v) = 0$ (conservation du flot)
2. Pour tout $u, v \in S : f(u, v) \leq c(u, v)$ (contrainte de capacité)
3. Pour tout $(x, y) \in U : f(u, v) = -f(v, u)$ (symétrie)

La quantité $f(u, v)$ qui peut-être positive et négative est appelée flot net du sommet u au sommet v . La valeur d'un flot f est définie par $|f| = \sum_{v \in S} f(s, v)$ autrement dit, le flot net total partant de la source. Ici la notation $|\cdot|$ indique la valeur de flot, et non une valeur absolue ou un cardinal d'ensemble). Dans le problème du flot maximal, on part d'un réseau de transport G de source s et de puits t , et on souhaite trouver un flot de valeur maximale entre s et t . Soit $G = (X, E)$ un réseau et soit f un flot de G . Montrer que

1. $\forall Y \subseteq X, f(Y, Y) = 0$.
2. $\forall Y, Z \subseteq X, f(Y, Z) = -f(Z, Y)$.
3. $\forall Y, Z, S \subseteq X$ avec $Y \cap Z = \emptyset, f(Y \cup Z, S) = f(Y, S) + f(Z, S)$ et $f(S, Y \cup Z) = f(S, Y) + f(S, Z)$.

Exercice 6 – Propriétés des flots bis Cette propriété reste valable pour n'importe quel définition. Soit f un flot dans un réseau, et soit α un nombre réel. Le produit scalaire-flot αf est une fonction de $X \times X$ vers R définie par

$$(\alpha f)(u, v) = \alpha \cdot f(u, v).$$

Démontrer que le flots d'un réseau forme un ensemble convexe en montrant que si f_1 et f_2 sont deux flots, alors $\alpha f_1 + (1 - \alpha) f_2$ est aussi un flot pour tout α dans l'intervalle $0 \leq \alpha \leq 1$.

Exercice 7 – Application des formules Cet exercice utilise la définition donné par l'exercice 1.

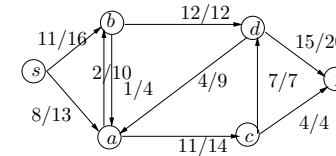


FIG. 2 – Exemple

Pour le réseau de transport $G = (V, E)$ et le flot donné par la figure 2, trouver un couple de sous-ensembles $X, Y \subseteq V$ pour lequel $f(X, Y) = -f(V - X, Y)$. Puis, trouver un couple de sous-ensembles $X, Y \subseteq V$ pour lequel $f(X, Y) \neq -f(V - X, Y)$.

Exercice 8 – Propriété des flots

Démontrer que $|f| = f(s, t)$ où t est le puits sachant que $|f| = f(s, S)$ en utilisant la définition donnée par l'exercice 1.

Exercice 9 – Théorème des valeurs entières

Démontrer le théorème suivant : si dans un réseau toutes les capacités finies ont des valeurs entières et s'il existe une coupe de capacité finie, alors il existe un flot de valeur maximale, prenant des valeurs entières sur chaque arc.

Exercice 10 – Le problème avec des sources et des puits multiples

1. Etendre les définitions de flot au problème avec sources et puits multiples.

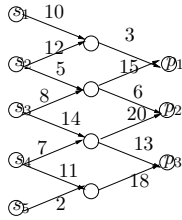


FIG. 3 – Un exemple de réseau à sources et puits multiples

- Démontrer que tout problème de flot sur un réseau à sources et puits multiples peut se traduire en un problème de flot à source et puits unique. Donner la transformation.

Exercice 11 – décomposition en circuits On considère un réseau $R = (G, s, p)$ avec un arc retour (ps). Nous montrons l'existence d'un ensemble $C(f)$ de coupes $(\mu, \lambda(\mu))$ où μ est un circuit élémentaire de G et $\lambda(\mu)$ un entier strictement positif, tel que : $f = \sum_{\mu \in C(f)} \lambda(\mu) g_\mu$. Un tel ensemble $C(f)$ est appelé décomposition du flot f sur les circuits de G . La fonction g_μ est appelée par convention flot canonique du circuit μ . Soit $G(f)$ le graphe partiel de g appelé graphe support de f , composé des arcs de G de flux strictement positif.

- Montrer qu'un flot entier $f \geq 0$ est nul si et seulement si si son graphe support $G(f)$ est sans circuit.
- Démontrer le théorème décomposition : tout flot non nul, à valeurs positives ou nulles, peut s'écrire comme une combinaison linéaire à coefficients positifs de flots dont chacun vaut 1 sur les arcs d'un circuit du graphe, et 0 sur tout autre arc. Un circuit est un chemin dont les extrémités coïncident.
- Appliquer ce théorème sur le figure 4.

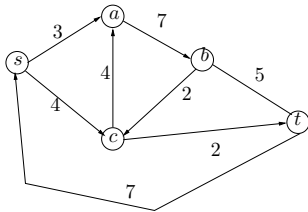


FIG. 4 – Exemple

2 Notion sur les coupes

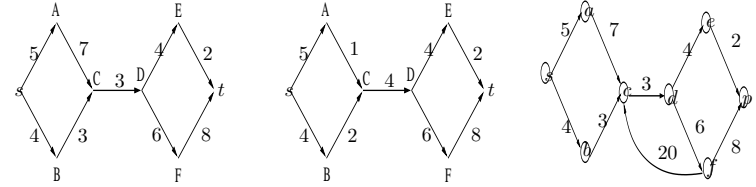
Exercice 12 – Propriétés des coupes

Vrai ou faux ?

- Si tous les arcs de G ont des valeurs différentes, G admet une coupe minimale unique.

- Si G contient un sommet différent de s sans arcs entrants, on peut supprimer ce sommet sans changer la valeur du flot maximum.
- Si on multiplie la capacité de chaque arc par un entier ≥ 0 , la coupe minimale reste inchangée.
- Si on additionne un entier ≥ 0 à la capacité de chaque arc la coupe minimale reste inchangée.

Exercice 13 – Calcul de coupes Pour chacun des réseaux suivants, donner une coupe minimale et sa capacité (la valeur numérique sur l'arc est la capacité de cet arc).



Exercice 14 – Sur le nombre de coupes

Soit un réseau $G(X, A, C, s, t)$, avec $|X| = n$, $|A| = m$, $s = 1$ et $t = n$. Calculer le nombre de (s, t) -coupes. Conclure.

Exercice 15 – Exercice sur les coupes Dans cet exercice on veut montrer que si $C(Y, \overline{Y})$ et $C(Z, \overline{Z})$ sont des coupes minimales alors $C(Y \cup Z, \overline{Y \cup Z})$ et $C(Y \cap Z, \overline{Y \cap Z})$ sont des coupes minimales. Pour cela on considère la figure 5 et soit $Y = V_1 \cup V_3$ et $Z = V_1 \cup V_2$.

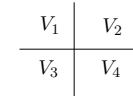


FIG. 5 – Une représentation possible

- Calculer $C(Y, \overline{Y})$ et $C(Z, \overline{Z})$.
- Calculer $C(Y \cup Z, \overline{Y \cup Z})$ et $C(Y \cap Z, \overline{Y \cap Z})$ et calculer la somme de ses deux expressions.
- Que peut-on dire des valeurs $C(V_2, V_3)$ et $C(V_3, V_2)$.
- Conclure.

Exercice 16 – Formulation de problèmes en informatique par les coupes

Soit $G = (V, E)$ un graphe non orienté sans boucle où V représente l'ensemble des sommets et E l'ensemble des arêtes. Soit la formule suivante : $\langle X, Y \rangle = \{xy \in E \mid x \in X \text{ et } y \in Y\}$.

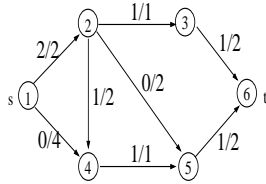
- Que représente la formule précédente ?
- Exprimer les problèmes suivants à l'aide de la formule précédente. Justifier votre réponse et donner un exemple.
 - Le problème *MAX CUT* consiste à déterminer un ensemble U de sommets qui maximise le nombre d'arêtes coupées.

- (b) Le problème *MAX INDEPENDENT SET* consiste à déterminer un plus grand sous-ensemble de sommets $U \subseteq V$ tel que U soit le plus grand stable.
- (c) Le problème *MIN VERTEX COVER* consiste à déterminer le plus petit sous-ensemble de sommets $U \subseteq V$ tel que chaque arête de E soit couverte par un sommet de U .
- (d) Le problème *MIN DOMINATING* consiste à déterminer le plus petit sous-ensemble de sommets $U \subseteq V$ tel que chaque sommet de G est soit dans U soit lié avec un sommet de U .
- (e) Les deux problèmes précédent semblent sensiblement identiques. Trouver une instance tel que les solutions des deux problèmes soient différentes.

Optimisation combinatoire
TD – Séance n° 4

Exercice 1 – Graphe d'écart

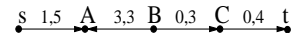
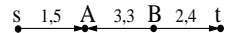
$s = 1$ (la source) et $t = 6$ (le puits) et un flot de débit 2, comporte au moins un arc saturé sur tout chemin de s à t . Dans le couple de valeurs numériques sur chaque arc, le premier nombre désigne le flot et le second, la capacité.



1. Donner le graphe d'écart $G^e(\phi)$ associé à ϕ .
2. Trouver un chemin de s à t dans $G^e(\phi)$. Donner la chaîne améliorante dans G associé au chemin dans le graphe d'écart.
3. Comment peut-on augmenter le flot dans G ? Conclure.
4. Donner deux coupes minimales différentes en précisant les ensembles S et \bar{S} , les arcs mis en jeu et la valeur de la coupe minimale et rerouter les résultats démontrés précédemment. Donner pour les coupes la valeur de $f(S, \bar{S})$ et de $f(\bar{S}, S)$.

Exercice 2 – Chaînes améliorantes

Pour chacune des chaînes améliorantes suivantes, donner la valeur maximale d'augmentation du flot de s à t :



Exercice 3 – Chaînes améliorantes bis

Démontrer le lemme suivant : soit f un flot dans un réseau, f est un flot de valeur maximum de valeur α si et seulement si il n'existe pas de chaîne augmentante de s à t .

Exercice 4 – Bijection

Montrer qu'il existe une bijection entre les chaînes améliorantes de G et les chemins issus de s et terminant à t dans le graphe d'écart.

Exercice 5 – Relation entre un flot résiduel et un flot du réseau de transport initial

Définition : Un flot f sur $G = (S, A)$ est une application $f : S \times S \rightarrow R$ vérifiant

1. pour tout $u \in S - \{s, t\}$ on a nécessairement : $\sum_{v \in S} f(u, v) = 0$ (conservation du flot)
2. Pour tout $u, v \in S : f(u, v) \leq c(u, v)$ (contrainte de capacité)
3. Pour tout $(u, v) \in U : f(u, v) = -f(v, u)$ (symétrie)

1. Démontrer le lemme suivant : soit $G = (S, A)$ est un réseau de transport de source s et de puit t , et soit un flot f de G . Soit G_f le réseau résiduel de G induit par f , et soit f' un flot de G_f . Alors, la somme $f + f'$ est un flot de G de valeur $|f + f'| = |f| + |f'|$
2. Vérifier que cette proposition reste valable pour toutes les définitions.

Exercice 6 – Exemple

Soit $G = (X, E)$, le graphe valué $c : E \rightarrow N$ donné par la figure 1. Les valeurs données sur le graphe indique la capacité des arcs.

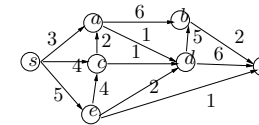


FIG. 1 – Un exemple de flot

1. Quelle est la capacité de la coupe $\{s, a, c, e\}$? Calculer la capacité minimale d'une coupe. Démontrer que la valeur d'un flot est toujours bornée par la capacité d'une coupe quelconque.
2. Soit $f(sa) = f(ca) = f(ad) = f(ab) = f(bp) = f(dp) = f(se) = f(ec) = 1$ et $f(sc) = f(cd) = f(ed) = f(ep) = f(db) = 0$. Montrer que f est un flot. f est-il maximum? Le chemin (s, a, d, p) est-il améliorant pour f ? Montrer que le chemin (s, a, b, p) est améliorant pour f . En déduire un flot f' améliorant de f . f' est-il maximal?
3. Chercher une suite de flots améliorés jusqu'à trouver un flot maximum.
4. Retrouver le résultat de l'exercice 3.

Exercice 7 – Relation entre un flot résiduel et un flot du réseau de transport initial suite

Définition : Un flot f sur $G = (S, A)$ est une application $f : S \times S \rightarrow R$ vérifiant

1. pour tout $u \in S - \{s, t\}$ on a nécessairement : $\sum_{v \in S} f(u, v) = 0$ (conservation du flot)
2. Pour tout $u, v \in S : f(u, v) \leq c(u, v)$ (contrainte de capacité)
3. Pour tout $(x, y) \in U : f(u, v) = -f(v, u)$ (symétrie)

Démontrer le lemme suivant : soit $G = (S, A)$ un réseau de transport, soit f un flot de G , et soit p un chemin améliorant de G_f . On définit une fonction $f_p : S \times S \rightarrow R$ par

$$\begin{aligned} f_p(u, v) &= c_f(p) \text{ si } (u, v) \in p \\ &= -c_f(p) \text{ si } (v, u) \in p \\ &= 0 \text{ sinon} \end{aligned}$$

où $c_f(p) = \min\{c_f(u, v) : (u, v) \in p\}$ et $c_f(u, v) = c(u, v) - f(u, v)$. Alors f_p est un flot de G_f de valeur $|f_p| = c_f(p) > 0$

Exercice 8 – Relation entre un flot résiduel et un flot du réseau de transport initial suite et fin

Démontrer le lemme suivant : soit $G = (S, A)$ un réseau de transport, soit f un flot de G , et soit p un chemin améliorant de G_f . soit f_p une fonction définie comme précédemment. Définir une fonction $f' : S \times S \rightarrow R$ par $f' = f + f_p$. Alors f' est un flot de G de valeur $|f'| = |f| + |f_p| > |f|$.

Exercice 9 – flot net et coupe Définition : Un flot f sur $G = (S, A)$ est une application $f : S \times S \rightarrow R$ vérifiant

1. pour tout $u \in S - \{s, t\}$ on a nécessairement : $\sum_{v \in S} f(u, v) = 0$ (conservation du flot)
2. Pour tout $u, v \in S : f(u, v) \leq c(u, v)$ (contrainte de capacité)
3. Pour tout $(x, y) \in U : f(x, y) = -f(y, x)$ (symétrie)

Démontrer le lemme suivant : soit f un flot dans un réseau de transport G de source s et de puits t , et soit (E, T) une coupe de G . Alors, le flot net à travers (E, T) est $f(E, T) = |f|$.

Exercice 10 – équivalences de conditions : flot maximum et coupe minimale

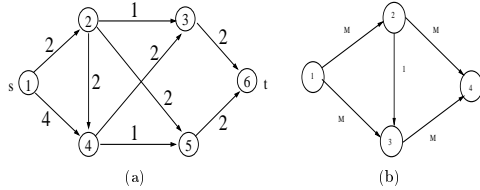
Si f est un flot dans un réseau de transport $G = (S, A)$ de source s et de puits t , montrer que les conditions suivantes sont équivalentes :

1. f est un flot maximum dans G ,
2. Le réseau résiduel G_f ne contient aucun chemin améliorant.
3. $|f| = c(E, T)$ pour une certaine coupe (E, T) de G .

Optimisation combinatoire
TD – Séance n° 5

Exercice 1 – Méthode de Ford-Fulkerson

Trouver un flot maximal dans le réseau de la figure 1(a) en utilisant l'algorithme de Ford-Fulkerson. Donner une coupe de capacité minimale. Si l'algorithme de Ford-Fulkerson augmente



le flot alternativement le long des chaînes $\{1,2,3,4\}$ et $\{1,3,2,4\}$, combien d'itérations sont nécessaires pour trouver le flot maximum? Conclure.

Exercice 2 – La méthode de Ford-Fulkerson ne converge pas vers le flot optimal

Dans cet exercice, nous nous intéressons au cas où les capacités des arcs ne sont pas entières. Nous allons montrer qu'il existe des cas où le flot trouvé par la méthode de Ford-Fulkerson ne converge pas vers le flot maximum. Pour cela nous considérons le réseau donné par la figure 1. Les capacités des arcs sont les suivantes :

- $c(c, d) = c(e_1) = 1 = a_0$
 - $c(a, b) = c(e_2) = r = a_1$ avec $r = (\sqrt{5} - 1)/2 = 0.62$,
 - $c(c, b) = c(e_3) = 1$
 - et les autres arcs admettent pour capacité un entier $M \geq 4$
- On suppose que $a_n = a_{n+1} + a_{n+2}$.

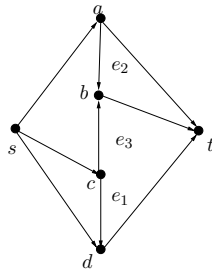


FIG. 1 – Réseau sur lequel la méthode Ford-Fulkerson ne converge pas

1. Démontrer par récurrence que $a_i = r^i, i = 0, 1, \dots$
2. Donner la valeur de la coupe minimale.

3. Dans un premier temps nous faisons passer un flot d'une unité le long du chemin s, c, b, t . Donner la valeur du flot résiduel ou la valeur de l'augmentation possible le long des arcs e_1, e_2 et e_3 après le passage du flot d'une unité sur le chemin précédent.
4. On suppose que les capacités résiduelles des arcs e_1, e_2 et e_3 est $c(e_1, e_2, e_3) = (a_n, a_{n+1}, 0)$. Nous considérons les trois chemins donnés par la figure 2. Donner les capacités résiduelles

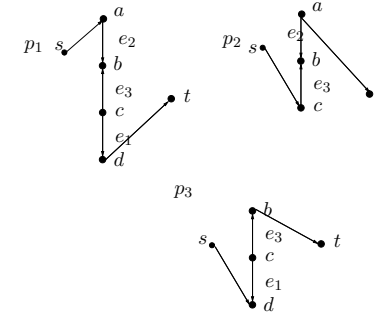


FIG. 2 – Trois chemins p_1, p_2 et p_3 pour le réseau donné par la figure 1

des arcs e_1, e_2 et e_3 après le passage du chemin p_1 puis du chemin p_2 et du chemin p_1 une nouvel fois et pour finir après p_3 .

5. Montrer que le flot converge vers $1 + 2 \sum_{n=2}^{\infty} a_n = 3$.
6. Conclure

Exercice 3 – Amélioration de la méthode de Ford-Fulkerson

Le but de cet exercice est de montrer que si dans l'algorithme de Ford-Fulkerson on choisit à chaque étape une chaîne améliorante ayant un nombre minimal d'arcs, on obtient un flot maximal en $O(nm^2) = O(n^5)$. C'est l'algorithme de Edmonds-Karp.

Préliminaire : soit P un plus court de s à p dans un graphe G , et x et y deux sommets rencontrés dans cet ordre sur P . Alors la portion P_{xy} de P comprise entre x et y est un plus court chemin de x à y . Prouvez-le.

1. On note par $d_i(v)$ la plus courte distance en nombre d'arcs entre s et v à l'itération i . Démontrer que si l'algorithme de Edmonds-Karp est exécuté sur un réseau de transport de source s et de puits $t, \forall i, d_{i+1}(t) \geq d_i(t)$ c'est-à-dire la distance du plus court chemin dans le graphe résiduel augmente de façon monotone avec chaque augmentation du flot. Nous allons montrer dans un premier temps $\forall i, v, d_{i+1}(v) \geq d_i(v)$. Pour cela, considérer un plus court chemin P dans le graphe d'écart G_{i+1}^e .
 - (a) Si P est également un chemin dans G_i^e , conclure.
 - (b) Si P admet un arc arrière (u, u) dans G_i^e . Que devient-il dans G_{i+1}^e ? Calculer les distances $d_i(u)$ en fonction de $d_i(w), d_{i+1}(w)$ en fonction $d_{i+1}(u)$. Conclure.
2. Si l'algorithme de Edmonds-Karp est exécuté sur un réseau de transport $G = (V, E)$ de source s et de puits t , alors le nombre total d'augmentation de flot effectuées par l'algorithme vaut au plus $O(nm)$. Pour cela, considérez un arc critique quelconque : On dit qu'un arc (u, v) dans un réseau résiduel (ou graphe d'écart) $G^e(f)$ est critique sur un chemin améliorant p si la capacité résiduelle de p est égale à la capacité résiduelle de (u, v)

c'est-à-dire si $c_f(p) = c_f(u, v)$. Regarder combien de fois (u, v) peut-il être critique pendant l'exécution de l'algorithme de Edmonds-Karp ? Evaluer la distance en nombre d'arcs entre le moment où l'arc (u, v) est critique et celui où devient à nouveau critique.

3. Conclure.

Exercice 4 – Méthode du chemin le plus rapide

La performance de l'algorithme de Ford-Fulkerson peut être améliorée en choisissant le chemin le plus rapide c'est-à-dire ayant la capacité résiduelle maximum. Pour cela, nous avons un résultat préliminaire à prouver.

1. Démontrer le théorème décomposition : tout flot non nul, à valeurs positives ou nulles, peut s'écrire comme une combinaison linéaire à coefficients positifs de flots dont chacun vaut 1 sur les arcs d'un circuit du graphe, et 0 sur tout autre arc. Un circuit est un chemin dont les extrémités coïncident.
2. Appliquer ce théorème sur la figure 3.

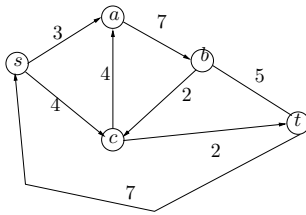


FIG. 3 – Exemple

3. On suppose que le flot a été décomposé en m chemins (circuits) en utilisant le théorème. Démontrer que $\Delta \geq |f_{max}|/m$ où Δ est le flot du chemin le plus rapide et f_{max} est un flot maximum sur G_f^e .
4. Calculer la complexité à l'itération i .
5. Donner le critère d'arrêt.
6. Sachant que le flot maximum et donc qu'un flot est borné par $m * C$ où C est la capacité maximale, montrer que i est en $O(m \log(mC))$
7. Donner la complexité finale.

Exercice 5 – Méthode de Dinic

Comme l'algorithme de Ford-Fulkerson, cet algorithme consiste, à partir d'un flot initial, à trouver une chaîne améliorante dans le réseau pour construire un flot de valeur supérieure. L'existence d'une chaîne améliorante étant liée à l'existence d'un chemin de la source s aux puits p dans le graphe d'écart associé au flot considéré, on procède de la manière suivante :

L'algorithme de Dinic procède par étapes. Pour définir une étape donnée, appelons d la distance (en nombre d'arcs) de s à p dans $G^e(f)$ au début de cette étape. L'algorithme va chercher des chaînes augmentantes de s à p de longueur d pour augmenter le flot, jusqu'à ce qu'il n'existe plus de telles chaînes. Pour cela, on construit à partir de $G^e(f)$ ce qu'on appelle le *graphe de couches*. On pose :

- $S_0 = \{s\}$
- pour i qui varie de 1 à $d-1$, S_i = l'ensemble des sommets de $G^e(f)$ qui sont à distance (en nombre d'arcs) i de s ;

- $S_d = \{p\}$

Les arcs du graphe de couches sont les arcs de $G^e(f)$ ayant leur origine dans S_i pour i compris entre 0 et $d-1$ et leur extrémité dans S_{i+1} .

Dans un graphe de couches donné, on répète alors les opérations suivantes :

- Chercher un chemin de s à p dans le graphe de couches. On procède pour cela par un parcours à partir de s . Pendant cette recherche lorsque utilisant un arc (x, y) on atteint un sommet y de demi-degré extérieur nul, on supprime l'arc (x, y) du graphe de couches.
- Si on trouve un chemin de s à p , on augmente le flot dans G en utilisant la chaîne augmentante correspondante. On actualise le graphe de couches en supprimant les arcs (x, y) du graphe de couches provenant d'un arc (x, y) de G , qui devient saturé et les arcs (y, x) du graphe provenant d'un arc (x, y) de G sur lequel le flux est devenu nul.
- Si on ne trouve pas de tel chemin, l'étape est terminée.

On passe alors à l'étape suivante après avoir mis à jour le graphe d'écart et déterminé d et le graphe de couches correspondant.

Nous allons détailler cet algorithme. Pour cela nous allons considérer le graphe donné par la figure 4. La première valeur indique le flot qui circule et la seconde la capacité.

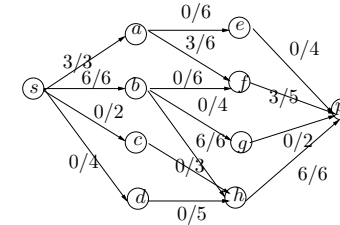


FIG. 4 – Graphe G

1. Donner le graphe d'écart G^e associé au graphe donné par la figure 4.
2. A partir des définitions, donner le graphe de couche associé au graphe d'écart G^e .
3. Nous allons déterminer un flot bloquant sur le graphe de couche. La définition d'un flot bloquant est la suivante : un flot f est un flot bloquant si il n'existe aucun chemin c de s à p avec $c(u, v) > f(u, v)$ pour tout $(u, v) \in E$.
 - (a) Démontrer qu'un flot maximal est forcément bloquant mais pas l'inverse.
 - (b) Donner un flot bloquant sur le graphe de couche déterminer précédemment.
4. Donner le flot maintenant sur le graphe G .
5. Donner maintenant le nouveau graphe d'écart G'^e . Comment obtient-t'on G'^e à partir de G^e . Existe-t'il des chemins de longueur cinq de s à p dans le nouveau graphe d'écart ? Donner le nouveau graphe de couche et conclure.
6. Justifier l'algorithme 6 sur la recherche d'un flot bloquant. Donner sa complexité.
7. Montrer que les graphes de couches successifs sont d'ordre strictement croissant.
8. En déduire que le nombre d'itérations est au plus de $n-1$.
9. Donner la complexité finale.

Algorithm 1 Algorithme pour la recherche d'un flot bloquant

Require: $G_c = (S_c, A_c)$ le graphe de couche

Ensure: g un flot qui représente l'augmentation du flot sur chaque arc.

```
for  $(u, v) \in A_c$  do
   $g(u, v) = 0$ 
end for
repeat
   $v = t, a = +\infty, chemin = \{\emptyset\}$ 
  for  $i = d$  à 1 do
    choisir un arc  $e_i = (u, v)$ 
     $a = \min\{C_c(e_i), a\}$ 
     $v = u$ 
     $i = i - 1$ 
     $chemin = chemin \cup e_i$ 
  end for
  for  $e \in chemin$  do
     $g(e) = g(e) + a$ 
     $C_c(e) = C_c(e) - a$ 
    if  $C_c(e) == 0$  then
      enlever  $e$  de  $A_c$ 
    end if
  end for
  for  $i = 1$  à  $d$  do
    for  $v \in S_i$  do
      if  $d^-(v) == 0$  then
        enlever  $v$  et les arcs sortant de  $v$ 
      end if
    end for
  end for
until  $t \notin S_d$ 
```

Exercice 6 – L'approche de Malhotra, Kumar et Mahaswari (1978)

Nous allons proposer un nouvel algorithme pour la recherche d'un flot bloquant. Pour chaque sommet $v \in S$, nous définissons le flot potentiel $p(v)$ de la manière suivante :

$$p(v) = \min\left\{\sum_{e^- = v} c(e), \sum_{e^+ = v} c(e)\right\} \text{ où } e^+ \text{ (resp. } e^-) \text{ désigne les arcs sortants de } v \text{ (resp. entrants).}$$

1. Que représente $p(v)$? Un sommet u est appelé un *point minimal* si $p(u) \leq p(v), \forall v$. Intuitivement il est possible de construire un flot g de valeur $w(g) = p(u)$ en "poussant" le flot de u à t et en "tirant" de u à s . C'est l'idée de l'algorithme.
2. Donner l'algorithme *RETIRER*.
3. Démontrer qu'un arc du graphe de couche n'est éliminé que s'il n'y a pas de chemin de s à t utilisant e ayant que des arcs avant.
4. Démontrer qu'à chaque fin d'étape, le flot trouvé est un flot bloquant.
5. Donner la complexité de l'algorithme 2.
6. Donner la complexité totale.
7. Appliquer la méthode sur le graphe donné par la figure 5.

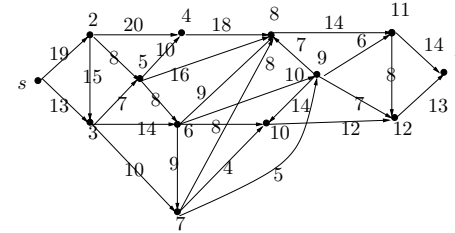


FIG. 5 – Réseau de transport

Exercice 7 – Application de la méthode de Dinic

Nous nous plaçons maintenant dans le cas où la capacité des arcs est unitaire.

1. Soit $S_i = V_0 \cup V_1 \cup \dots \cup V_i$ pour $1 \leq i \leq l - 1$ où V_i désigne l'ensemble des sommets qui se trouve à une distance i de s . Que représente (S_i, \bar{S}_i) ?
2. Donner la valeur maximale de (S_i, \bar{S}_i) .
3. En déduire que $|\hat{f}| \leq |V_i| \times |V_{i+1}|$ où \hat{f} désigne la valeur du flot maximum. Nous avons ainsi majoré la valeur du flot maximum par des données du problème.
4. Démontrer que la distance l entre s et t ne peut pas être supérieure à $\sqrt{|\hat{f}|} \leq 2|V|/l$ où \hat{f} est le flot maximum.
5. Démontrer que l'algorithme de Dinic nécessite $O(n^{2/3})$ étapes. Pour cela vous considérez deux cas $|\hat{f}| \leq 2n^{2/3}$ et $|\hat{f}| > 2n^{2/3}$. Pour le deuxième cas vous vous regarderez la valeur du flot maximum dans le graphe d'écart après que le flot maximum soit supérieur à $|\hat{f}| > 2n^{2/3}$.
6. Conclure sur la complexité totale.

Exercice 8 – Application de la méthode de Dinic bis

Un réseau simple est un réseau dans lequel tous les sommets admettent soit un arc sortant soit un arc entrants.

Algorithm 2 Algorithme pour la recherche d'un flot bloquant pour MKM

Require: $G = (S, A)$ le réseau**Ensure:** Flot de valeur maximum

```
for  $e \in E$  do
   $g(e) \leftarrow 0$ 
end for
for  $v \in V$  do
  if  $v = t$  then
     $p^-(v) \leftarrow \infty$ 
  else
     $p^-(v) \leftarrow \sum_{e^- = v} c(e)$ 
  end if
  if  $v = s$  then
     $p^+(v) \leftarrow \infty$ 
  else
     $p^+(v) \leftarrow \sum_{e^+ = v} c(e)$ 
  end if
end for
repeat
  for  $v \in V$  do
     $p(v) \leftarrow \min\{p^+(v), p^-(v)\}$ 
  end for
   $w = \min\{p(v), v \in S\}$ 
   $POUSSER(w, p(w))$ 
   $TIRER(w, p(w))$ 
  while  $\exists v \setminus p^+(v) = 0$  or  $p^-(v) = 0$  do
    for  $e \in \{e \in E, e^- = v\}$  do
       $u \leftarrow e^+$ 
       $p^+(u) \leftarrow p^+(u) - c(e)$ 
      Retirer  $e$  de  $A$ 
    end for
    for  $e \in \{e \in E, e^+ = v\}$  do
       $u \leftarrow e^-$ 
       $p^-(u) \leftarrow p^-(u) - c(e)$ 
      Retirer  $e$  de  $A$ 
    end for
    Retirer  $v$  de  $S$ 
  end while
until  $s \notin V$  ou  $t \notin V$ 
```

Algorithm 3 Procédure POUSSER (y, k)

Soit Q une file avec un seul élément y

```
for  $u \in S$  do
   $b(u) \leftarrow 0$ 
end for
 $b(y) \leftarrow k$ 
repeat
  Retirer le premier élément de  $Q$ 
  while  $v \neq t$  et  $b(v) \neq 0$  do
    choisir un arc  $e = vu$ 
     $m \leftarrow \min\{c(e), b(v)\}$ 
     $c(e) \leftarrow c(e) - m$ ,  $g(e) \leftarrow g(e) + m$ 
     $p^+(u) \leftarrow p^+(u) - m$ ,  $b(u) \leftarrow b(u) + m$ 
     $p^-(u) \leftarrow p^-(u) - m$ ,  $b(v) \leftarrow b(v) - m$ 
    Mettre  $u$  dans la file
    if  $c(e) = 0$  then
      Retirer  $e$  de  $A$ 
    end if
  end while
end repeat
until  $Q = \emptyset$ 
```

1. Démontrer que le cas d'un réseau simple avec la distance entre s et t supérieure ou égale à l on a $|\hat{f}| \leq |V|/(l-1)$
2. Démontrer que le nombre d'étapes est en $O(\sqrt{n})$.
3. conclure sur la complexité totale.

Optimisation combinatoire
TD – Séance n° 6

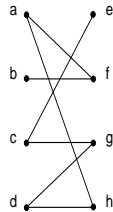
Exercice 1 – Applications des flots

Soit $G = (P, Q, E)$ un graphe biparti.

1. Démontrer qu'un couplage M de G correspond à un flot à valeurs entières dans \tilde{G} .
2. Montrer que à tout couplage M de G correspond un flot dans \tilde{G} de débit $|M|$, et à tout flot dans \tilde{G} de débit F correspond un couplage M de G tel que $|M| = F$.
3. Le cardinal d'un couplage maximum dans un graphe biparti G est la valeur d'un flot maximum dans son réseau \tilde{G} correspondant.

Exercice 2 – Application des flots suite

Soit G le graphe suivant :



1. Donner le réseau de transport \tilde{G} tel que le problème consistant à trouver un couplage de taille maximum dans G soit équivalent au problème consistant à trouver un flot de débit maximum dans \tilde{G} .
2. Trouver un flot de débit maximum dans \tilde{G} à l'aide de l'algorithme de Ford-Fulkerson. La suite de chaînes améliorantes sera choisie de telle manière à ce que l'on ait au moins une chaîne améliorante comportant un arc arrière.
3. En déduire un couplage de taille maximum pour le graphe G .

Exercice 3 – Modélisation par les flots

On doit organiser des transports maritimes transocéaniques entre les USA et l'Europe. On connaît l'ensemble S des ports d'expédition, avec pour chaque port i une disponibilité a_i en tonnes de fret, ainsi que l'ensemble T des ports de destination, avec pour tout port j de ce type une demande d_j . Pour toute liaison entre un port i de S et un port j de T , on donne le coût de transport w_{ij} d'une tonne de fret. Le but est d'acheminer toutes les marchandises de S jusqu'aux ports de T , à coût total minimal. Convertir ce problème en un problème de flot à coût minimal.

Application :

$$S = \{s_1, s_2, s_3\}, T = \{t_1, t_2, t_3, t_4\},$$

$$a_1 = 5, a_2 = 4, a_3 = 6, d_1 = 5, d_2 = 3, d_3 = 5, d_4 = 2$$

$$w_{ij} = \begin{pmatrix} 5 & 4 & 7 & 6 \\ 2 & 5 & 3 & 2 \\ 6 & 3 & 4 & 4 \end{pmatrix}$$

Exercice 4 – Modélisation par les flots suite

Considérons r points d'eau (barrages, sources) connectés à v villes par un réseau de canalisations. On connaît pour chaque point d'eau la quantité maximale qu'il peut fournir par jour, et chaque ville la consommation totale d'eau par jour prévue dans cinq ans. On connaît également les débits permis par jour pour chaque canalisation du réseau actuel. On veut savoir si le réseau actuel pourra satisfaire les consommations prévues dans cinq ans. Modéliser ce problème en terme de flot.

Exercice 5 – Modélisation par les flots suite La technologie WDM s'intéresse à la propagation des faisceaux laser multiples à travers une fibre unique à condition que chaque faisceau laser possède une fréquence optique différente. Intuitivement, cela correspond à des rayons de lumière de couleurs différentes. Le réseau est modélisé par un graphe G . Une connexion entre deux noeuds x et y correspond donc à un chemin de x vers y et la donnée d'une couleur. Deux chemins utilisant le même arc doivent donc avoir des couleurs différentes. Pour cet exercice un noeud x doit diffuser en permanence vers un ensemble P de noeuds, de cardinalité p . Pour réaliser cette diffusion il faut donc trouver p chemins partant de x et se terminant sur chaque sommet de P . La charge d'un arc du graphe est le nombre de chemins passant par cet arc. On peut montrer (mais ce n'est pas le but de cet exercice) que si la charge de l'arc le plus chargé est k alors k longueurs d'onde suffisent pour faire la diffusion.

1. On veut savoir si k longueurs suffisent pour faire une diffusion dans G à partir d'un sommet x vers un ensemble P . Montrer que ce problème peut être modélisé par un problème de flot.
2. En déduire un algorithme polynomial pour connaître le nombre minimal de longueur d'onde nécessaire pour diffuser de x vers P .

Exercice 6 – Applications des flots suite

On veut élire un conseil de la ville représentant l'ensemble des résidents et de partis politiques. La ville possède :

- r résidents $\{R_1, \dots, R_r\}$
- q clubs $\{C_1, \dots, C_q\}$
- p parti politiques $\{P_1, \dots, P_p\}$

Chaque résident est un membre d'un seul club et d'un seul parti politique. Chaque club nomme un des membres pour le représenter dans le conseil de la ville. Le nombre de conseillers appartenant à un parti politique P_k est au plus u_k .

Question : Est-il possible de trouver un conseil de la ville où chaque club est représenté? Montrer également que le flot maximal est égal q ssi chaque club est représenté au conseil de la ville.

Exercice 7 – Applications des flots suite

Calcul distribué sur une machine à 2 processeurs

On dispose d'un programme composé de plusieurs modules, et on veut affecter les différents modules aux deux processeurs d'une machine tel que le coût de communication entre les deux processeurs soit minimisé. On note par

- α_i le coût du calcul du module i sur le processeur 1.
- β_i le coût du calcul du module i sur le processeur 2.
- C_{ij} le coût de la communication entre le module i et j si i est assigné au processeur 1 et j au processeurs 2 (ou vice versa). Si i et j sont affectés au même processeur alors $C_{ij} = 0$.

i	1	2	3	4
α_i	6	5	10	4
β_i	4	10	3	8

et

	1	2	3	4
1	0	5	0	0
2	5	0	6	2
3	0	6	0	1
4	0	2	1	0

Formuler ce problème comme un problème de coupe minimale sur un graphe non orienté G . Montrer que les coupes séparantes de G sont en bijection avec les affectations de modules aux deux processeurs et que la capacité d'une coupe séparante est égale au coût de l'affectation. Résoudre le problème avec les données ci-dessous.

Exercice 8 – Forte arête-connectivité d'un graphe non orienté

Soit G un graphe non orienté et G^* le *symétrisé* de G , c'est-à-dire que G^* a le même ensemble de sommets que G et toute arête (u, v) de G donne naissance dans G^* , aux deux arcs opposés (u, v) et (v, u) . Par ailleurs, on attribue à chaque arc de G^* une capacité de 1.

Etant donnés deux sommets a et b de G (et donc de G^*), on considère les quatre paramètres suivants :

- $N(a, b)$ est le nombre minimum d'arêtes qu'il faut supprimer dans G pour qu'il n'existe plus de chaîne de a vers b ;
- $N^*(a, b)$ est le nombre minimum d'arcs qu'il faut supprimer dans G^* pour qu'il n'existe plus de chemin de a vers b dans G^* ;
- $P(a, b)$ représente le nombre maximum de chaînes deux à deux arêtes-disjointes de G entre a et b ;
- $P^*(a, b)$ représente le nombre maximum de chemins deux à deux arcs-disjointes de G^* entre a et b .

1. Soit $val(f_{ab})$ la valeur maximum d'un flot de a à b dans G^* . On a alors,

$$N(a, b) = N^*(a, b) = P(a, b) = P^*(a, b) = val(f_{ab})$$

Exercice 9 – Application suite Deux usines à gaz, g_1 et g_2 , alimentent trois villes, v_1 et v_2 et v_3 , par l'intermédiaire du réseau de distribution ci-dessous ; les nombres associés aux arcs et aux sommets de ce réseau représentent les capacités journalières :

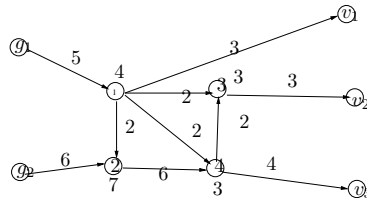


FIG. 1 – réseau

Quelle est la production journalière maximale que peuvent écouler ces deux usines ? Si cette production est atteinte, à quelle consommation journalière chacune des villes peut-elle prétendre ?

Exercice 10 – Application Soit $\mathcal{E} = \{E_1, E_2, \dots, E_m\}$ un ensemble de parties de Y . $\{e_1, e_2, \dots, e_m\} \subseteq$

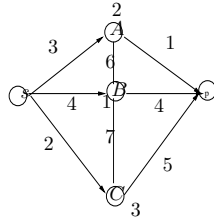
Y est un système de représentants distincts (SRD) de \mathcal{E} si et seulement si $e_i \in E_i, \forall i = 1, 2, \dots, m$ et $e_i \neq e_j, \forall i \neq j$.

1. Soit la famille constitué d'ensemble $\{1, 2, 3\}, \{3, 4\}, \{1, 2, 3\}$ et $\{5, 6, 7\}$. Donner un système de représentant distincts.
2. Montrer que ce problème que ce problème revient à chercher un couplage dans un graphe particulier. Vérifier votre modélisation en utilisant la famille donnée précédemment.

Optimisation combinatoire
TD – Séance n° 5

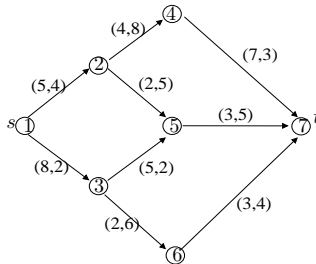
Exercice 1 – Transformation

Transformer le réseau ci-dessous en un réseau de transport. Les capacités sont marquées à côté des arcs, des arêtes ou à côté des noeuds.



Exercice 2 – Flots à coûts minimum

- Déterminer un flot maximum de coût minimum dans le réseau ci-dessous, où les couples de nombres associés aux arcs représentent respectivement les capacités et les coûts unitaires. $s = 1$ est la source et $t = 7$ est le puits.
- Examiner comment varie le coût minimal total $f_m(\phi)$ en fonction de la valeur du flot ϕ choisie.



Exercice 3 – Flots réalisable

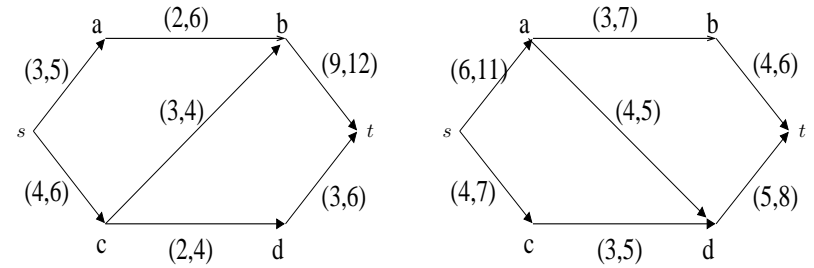
Pour les deux réseaux ci-dessous (figure , dire s'ils admettent un flot réalisable (Utiliser le théorème de Hoffman (vu en cours) et si oui, en trouver un, puis trouver un flot maximum. Les couples associés aux arcs représentent respectivement les capacités minimales et les capacités maximales.

Exercice 4 – Le problème du flot compatible

On pose $B^+(i) = \sum_{j \in \Gamma^+(i)} b_{ij}$ et $B^-(i) = \sum_{j \in \Gamma^-(i)} b_{ji}$.

Etant donné le réseau G on construit un réseau G' avec $b'_{ij} = 0$ de la manière suivante :

- Si G n'a pas d'arc (t, s) on le crée avec $c_{ts} = \infty$. Si (t, s) existe alors on change sa capacité en posant $c_{ts} = \infty$.



- Pour tout arc (i, j) on remplace c_{ij} par $c'_{ij} = c_{ij} - b_{ij}$.
- On ajoute une source s^* et un puits t^* .
- $\forall i$, si $\Gamma^+(i) \neq \emptyset$, on ajoute un arc (i, t^*) de capacité $B^+(i)$.
- $\forall i$, si $\Gamma^-(i) \neq \emptyset$, on ajoute un arc (s^*, i) de capacité $B^-(i)$.

Par exemple, à partir du réseau G de la figure 1 a) on obtient le réseau G' représenté à la figure 1 b).

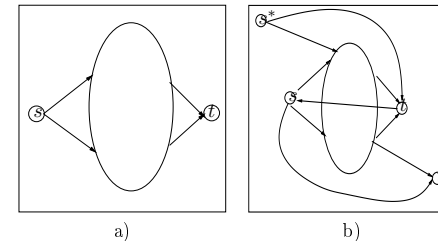


FIG. 1 – La transformation $G \rightarrow G'$

- Démontrer le théorème suivant :Le réseau G admet un flot compatible si et seulement si le réseau G' admet un flot maximum ϕ' de débit $B^+(s^*) = B^-(t^*)$, c.a.d. saturant les arcs d'origine s^* et d'extrémité t^* . De plus pour tout arc (i, j) de G on a $\phi_{ij} = \phi'_{ij} + b_{ij}$.
- Donner le nombre de sommets et d'arcs et la complexité.

Exercice 5 – Problème d'arrondi d'une matrice On se donne une matrice A de $p \times q$ nombres réels, avec α_i la somme de la ligne i et β_j la somme de la colonne j . Le problème consiste à arrondir chaque élément de la matrice ainsi que les sommes des lignes et des colonnes à l'entier immédiatement inférieur ou à l'entier immédiatement supérieur. Cet arrondi est dit **consistant** si

la somme des éléments arrondis d'une ligne soit égale à l'arrondis de cette ligne, et que la somme des éléments arrondis d'une colonne soit égale à l'arrondis de cette colonne.

- Formuler ce problème du calcul d'un arrondi consistant comme un problème du flot compatible avec des bornes inférieures et de capacités des entiers positifs et montrer comment

				α_i
	7,5	6,3	15,4	29,2
le résoudre.	3,9	9,1	3,6	16,6
	15	5,5	21,5	42,0
	β_j 26,4	20,9	40,5	

2. Maintenant on s'intéresse à l'arrondi multiple de 2. Cet arrondi consiste à arrondir chaque élément de la matrice ainsi que les sommes des lignes et des colonnes à l'entier immédiatement inférieur multiple de 2 ou à l'entier immédiatement supérieur multiple de 2. Formuler ce problème du calcul d'un arrondi consistant comme un problème du flot compatible avec des bornes inférieures et de capacités des entiers positifs et montrer comment le résoudre.

				α_i
	7,5	6,3	15,4	29,2
	3,9	9,1	3,6	16,6
	15,0	5,5	21,5	42,0
	β_j 26,4	20,9	40,5	

Exercice 6 – Réseau de transport

Un transporteur veut expédier une certaine marchandise depuis les villes a et b vers les villes d et e . Les moyens de communications que l'on supposera à sens unique, sont les suivants : de a on peut aller vers e ou vers une ville intermédiaire c ; de b on peut aller vers c , d ou e ; de c on peut aller vers d . Les quantités disponibles en a et b sont respectivement 7 et 12 unités; les quantités demandées en d et e sont respectivement 9 et 10 unités; en c il y a ni demande ni excédent de marchandises. Les camions disponibles pour organiser le transport permettent d'envoyer au plus 5 unités de marchandise de a vers c , 14 de de a vers e , 7 de b vers c , 4 de b vers d , 2 de b vers e et 6 de c vers d . Enfin les coûts de transport unitaires valent : 1 de a vers c , 4 de a vers e , 2 de b vers c , 2 de b vers d , 7 de b vers e et 0 de c vers d .

1. Quelle quantité maximum le transporteur peut-il expédier de a et b vers d et e , et à quel prix ?
2. Donner la coupe minimale.
3. Soit $Y = \{b, c, d\}$. Conclure en utilisant le Théorème de Gale.

Exercice 7 – Les arbres de Gomory-Hu

Dans cette exercice, nous nous intéressons aux arbres de Gomory-Hu.

1. Démontrer que $f_{1n} \geq \min(f_{12}, f_{23}, \dots, f_{n-1n})$.
2. Construire un arbre $T = (V, E)$ avec $V = \{1, \dots, 6\}$ dans lequel la valeur du flot maximum entre i et j est l'élément se trouvant en ij dans la matrice ci-dessous.

$$F = \begin{pmatrix} M & 25 & 25 & 25 & 25 & 20 \\ 25 & M & 32 & 32 & 29 & 20 \\ 25 & 32 & M & 37 & 29 & 20 \\ 25 & 32 & 37 & M & 29 & 20 \\ 25 & 29 & 29 & 29 & M & 20 \\ 20 & 20 & 20 & 20 & 20 & M \end{pmatrix}$$

3. Dans cet partie de l'exercice, nous considérons le réseau donné par la figure 2. Les capacités des arêtes sont données à coté des arêtes.
 - (a) Donner la matrice des flots.
 - (b) Donner un arbre ayant la même valeur que la matrice de flots.
 - (c) Donner l'arbre de Gomory-Hu.

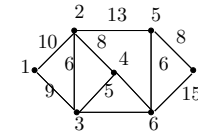


FIG. 2 – Réseau non orienté

Optimisation combinatoire
TD – Séance n° 8

Exercice 1 – Propriété des couplages

On acceptera le théorème suivant dû à Tutte : Le graphe $G = (V, E)$ admet un couplage parfait

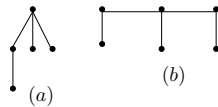


FIG. 1 – Arbres

si et seulement si le nombre de composantes connexes de taille impaire de $(G - S)$ ne dépasse pas $|S|, \forall S \subset V$.

1. Rappeler la définitions d'un couplage parfait.
2. Quelle est la condition nécessaire pour l'existence d'un couplage parfait concernant le nombre de sommets d'un graphe.
3. Rappeler les propriétés des arbres.
4. Montrer qu'un arbre admet au plus qu'un seul couplage parfait.
5. Donner un couplage parfait si il existe pour les graphes donnés par la figure 1.
6. Démontrer qu'un arbre admet un couplage parfait si et seulement si il existe une seule composante connexe de taille impaire quand on enlève un sommet quelconque.

Exercice 2 – Sur le graphe biparti

1. Quel est le nombre d'arêtes d'un graphe biparti complet $G = (V, E)$ où $V = X \cup Y$.
2. Soit un graphe biparti $G = (V, E)$ d'ordre n , montrer que $|E| \leq n^2/4$.

Exercice 3 – Couplage

Définition : Un stable est un ensemble indépendant de sommets $S \subseteq X$ tel que le sous-graphe induit par S n'a pas d'arête. Le cardinal maximum d'un stable de G est noté par $\alpha(G)$.

Notation : un transversal de cardinal minimum est noté $\tau(G)$.

Démontrer le lemme suivant : soit $G = (X, E)$ un graphe et $S \subseteq X$. Alors S est un stable si et seulement si \bar{S} est un transversal, et $\tau(G) + \alpha(G) = |X|$ où .

Exercice 4 – Couplage

Définition : Un recouvrement est un ensemble indépendant d'arêtes $R \subseteq E$ tel que $\forall x \in X$ il existe une arête $e \in R$ telle que $x \in e$. Le cardinal minimum d'un recouvrement de G est noté par $\varphi(G)$.

Notation : un couplage de cardinal maximum est noté $\nu(G)$.

Démontrer le lemme suivant : Si G n'a pas de sommet isolé alors $\varphi(G) + \nu(G) = |X|$.

Exercice 5 – Couplage

Démontrer le théorème de Konig : soit G un graphe biparti sans sommet isolé. Alors $\alpha(G) = \varphi(G)$

Exercice 6 – Couplage

Proposez un algorithme qui utilisent les chaînes améliorantes augmentantes et évaluer sa complexité pour la recherche d'un couplage dans un graphe biparti.

Indication : Faire apparaître un graphe orienté pour la recherche des chaînes améliorantes augmentantes alternées.

Appliquer l'algorithme du couplage maximum vu en cours pour déterminer un couplage maximum dans les graphes donner par la figure 2.

Exercice 7 – Couplage

Un couplage dans un graphe biparti $G = (V, A)$ où $V = X \cup Y$, est un ensemble d'arêtes C de G tel que pour tout $(x, y) \in C, (u, v) \in C$, avec $(x, y) \neq (u, v)$, les sommets u, v, x, y sont différents.

1. Le problème du couplage maximum peut se résoudre par les algorithmes de flot. Proposez la modélisation sous forme d'un réseau de transport G' .
2. Soit C un couplage de G . Une suite de sommets $x_1, y_1, x_2, \dots, y_{k-1}, x_k, y_k$ avec $x_i \in X$ et $y_i \in Y$, est appelée chaîne alternée ssi $(x_i, y_i) \notin C$ et $(y_i, x_{i+1}) \in C$. C'est une chaîne alternée améliorante si x_1 et y_k ne sont pas saturés. Que dire s'il existe une chaîne améliorante pour C ?
3. Calculez les couplages maximum sur les graphes de la figure 2.

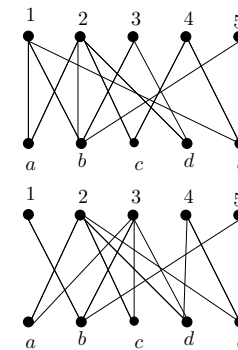


FIG. 2 –

Exercice 8 – Couplage suite Un couplage parfait est un couplage couvrant tous les sommets. Soit $G = (V, A)$ un graphe biparti non orienté tel que $V = X \cup Y$, avec $|X| = |Y|$. Pour tout ensemble de sommet $S \subseteq V$, nous définissons son voisinage par : $N(S) = \{y \in V : \exists x \in S \text{ tel que } (x, y) \in A\}$.

1. Existe-t-il un couplage parfait sur les bipartis de la figure 2 de l'exercice précédent. Que dire d'un couplage dont le cardinal est $|X|$?
2. On se propose de démontrer par récurrence le théorème de Hall. Il existe un couplage parfait dans G ssi $|N(S)| \geq |S|$ pour tout ensemble $S \subseteq X$.
 - (a) Montrer que s'il existe S tel que $|S| > |N(S)|$ alors il n'existe pas de couplage parfait (dans un premier, donnez un exemple).

- (b) Énoncer le cas de base et poser l'hypothèse de récurrence.
 - (c) Supposons que $\forall S \subset X$, nous avons $|N(S)| \geq |S| + 1$.
 - (d) Supposons qu'il existe S tel que $|N(S)| = |S|$.
3. Une autre méthode :
- (a) Supposons que G contienne un couplage M qui sature chaque sommet de l'ensemble X , et soit S un sous-ensemble de X . Conclure.
 - (b) Soit M^* un couplage maximum dans G . Pourquoi ?
 - (c) Est-ce que M^* sature tous les sommets. Soit u le sommet non saturé.
 - (d) Soit $S = Z \cap X$ et $T = Z \cap Y$. Calculer $N(S)$ et conclure.

Exercice 9 – Couplage suite Un graphe biparti $G = (V, A)$ où $V = X \cup Y$, est k -régulier si tout sommet $v \in V$ a un degré exactement égal à k .

1. Montrer que $|X| = |Y|$,
2. Calculer à l'aide des flots un couplage maximum sur le graphe donné par la figure 3.

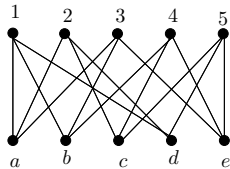


FIG. 3 –

3. Montrer que pour tout ensemble on a $|N(S)| \geq |S|$. En déduire que pour un biparti k -régulier tout couplage maximum est parfait.

Exercice 10 – L'échiquier

Considérons une plaque carrée subdivisée en n^2 cases identiques dans laquelle on supprime la case en haut à gauche et la case en bas à droite.

Comment découper cette plaque en dominos (de deux cases chacun) sans faire de déchets ?

Exercice 11 – Couplages

Soit $G = (V, E)$. Un couplage C dans G est un ensemble d'arêtes non adjacentes de E . Un couplage C est dit *maximal pour l'inclusion* si en rajoutant une arête quelconque dans C on ne conserve pas un couplage. Un couplage C est dit *parfait* si chaque sommet est l'extrémité d'une des arêtes de C .

Lorsque c'est possible, décrivez un couplage parfait, sinon donnez un couplage maximum pour l'inclusion, dans les graphes K_n , C_n , grilles, tores, H_d .

Exercice 12 – Couplages

- Un couplage maximal de G est :
- un couplage de G tel que :
 - pour toute arête e de $E - M$, $M \cup \{e\}$ n'est pas un couplage.

Un couplage maximal minimum M_0 de G est :

- Un couplage maximal de G tel que :
- M_0 est le plus petit (en nombre d'arêtes) couplage maximal possible de G ($|M_0|$ est minimum).

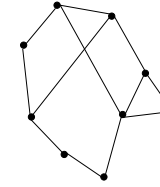


FIG. 4 – Exemple

1. Donner un couplage maximal et un couplage maximal minimum sur la figure 4.
2. Démontrer : soit un couplage M maximal de G . Notons C l'ensemble des sommets des arêtes de M . Alors, pour toute arête e de E , au moins une des deux extrémités de e est dans C .
3. Démontrer : soit G un graphe, M un couplage maximal de G , M_0 un couplage maximal minimum de G . On a alors $|M_0| \leq |M| \leq 2|M_0|$.
4. Donner un exemple tel que $|M| = 2|M_0|$.

Exercice 13 – Couplages Appellons "libre" une arête qui appartient à un couplage maximum mais non à tous. Montrer qu'une arête e est libre si et seulement si pour un couplage maximum E_0 arbitraire, elle appartient soit à une chaîne alternée paire commençant par un sommet insaturé, soit à un cycle alterné pair.

Exercice 14 – Couplages montrer que le nombre maximum d'arêtes d'un graphe simple d'ordre n avec un couplage maximum de q arêtes ($n \geq 2q > 0$) est :

- C_2^{2q} si $2q = n$
- C_2^{2q+1} si $2q \leq n \leq \frac{5q-1}{2}$
- $C_2^q + q(n-q)$ si $n > \frac{5q-1}{2}$

Exercice 15 – Couplages de maximum de poids minimum dans un graphe biparti

Soit la matrice des coûts pour le graphe biparti $(K_{5,5}, w)$ où la fonction de poids w est donné par la matrice suivante :

$$A = \begin{pmatrix} 6 & 8 & 4 & 12 & 9 \\ 3 & 4 & 11 & 5 & 1 \\ 7 & 2 & 7 & 9 & 2 \\ 3 & 5 & 6 & 8 & 8 \\ 12 & 6 & 3 & 6 & 12 \end{pmatrix}$$

$$B = \begin{pmatrix} 3 & 8 & 9 & 1 & 6 \\ 1 & 4 & 1 & 5 & 5 \\ 7 & 2 & 7 & 9 & 2 \\ 3 & 1 & 6 & 8 & 8 \\ 2 & 6 & 3 & 6 & 2 \end{pmatrix}$$

Donner la valeur du couplage maximum de poids minimum en

utilisant :

1. Donner la valeur du couplage maximum de poids minimum sur la matrice A .
 - (a) L'algorithme Hongrois
 - (b) En utilisant l'algorithme qui utilise le flot maximum de coût minimum.

Exercice 16 – Couplages de maximum de poids maximum dans un graphe biparti

Soit la matrice des coûts pour le graphe biparti $(K_{5,5}, w)$ où la fonction de poids w est donné par la matrice suivante :

$$A = \begin{pmatrix} 5 & 4 & 2 & 4 \\ 6 & 3 & 3 & 5 \\ 6 & 2 & 5 & 2 \\ 6 & 3 & 2 & 7 \end{pmatrix}$$

$$B = \begin{pmatrix} 2 & 2 & 2 & 5 & 2 \\ 1 & 3 & 3 & 1 & 4 \\ 6 & 4 & 7 & 4 & 2 \\ 5 & 6 & 5 & 5 & 2 \\ 9 & 8 & 5 & 5 & 2 \end{pmatrix}$$

1. Donner la valeur du couplage maximum de poids minimum pour la matrice A .
2. De même pour la matrice B

Exercice 17 – Couplages de maximum de poids maximum dans un graphe biparti

Soit la matrice des coûts pour le graphe biparti $(K_{5,5}, w)$ où la fonction de poids w est donné par la matrice suivante :

$$\begin{pmatrix} 3 & 8 & 9 & 1 & 6 \\ 1 & 4 & 1 & 5 & 5 \\ 7 & 2 & 7 & 9 & 2 \\ 3 & 1 & 6 & 8 & 8 \\ 2 & 6 & 3 & 6 & 2 \end{pmatrix}$$

1. Soit l' le nouveau label sur les sommets déterminer à partir de l en utilisant la valeur α_l défini dans le cours. Montrer que l' admet les propriétés suivantes :
 - (a) l' est un ensemble d'étiquettes réalisables pour G .
 - (b) Le couplage M' et l'arbre M' -alterné H sont contenus dans $G_{l'}$.
 - (c) $T \subset N_{G_{l'}}(S)$ et alors l'arbre M' -alterné H peut-être étendu dans $G_{l'}$.
2. Donner la valeur du couplage maximum de poids maximum en utilisant :
 - (a) L'algorithme Hongrois. Vous préciserez toutes les étapes, et vous donnerez l'arbre(s) alterné(s).
 - (b) En utilisant l'algorithme qui utilise le flot maximum de coût minimum.
 - (c) Rappeler la définition du problème de la couverture sommet de poids minimum.
 - (d) Donner une couverture sommet pondéré de poids minimum pour le graphe biparti $(K_{5,5}, w)$ ci-dessus.

1 Coloration de sommets

Exercice 1 – coloration des sommets

Si H est un sous-graphe de G alors $\chi(H) \leq \chi(G)$.

Exercice 2 – coloration des sommets

Démontrer que $\chi(G) = \max\{\chi(C), C \text{ composante connexe de } G\}$

Exercice 3 – coloration des sommets

Montrer que le problème de chercher un k -stable est équivalent à montrer l'existence d'un graphe k -coloriable.

Exercice 4 – coloration des sommets

Montrer qu'un graphe est biparti si et seulement si il est 2-coloriable.

Exercice 5 – coloration des sommets

Soit $G = (X, E)$ un graphe. On note $\alpha(G)$ le nombre de stabilité, c'est à dire la taille d'un stable maximum. Montrer $\lceil \frac{n}{\alpha(G)} \rceil \leq \chi(G) \leq n - \alpha(G) + 1$

Exercice 6 – coloration des sommets

Démontrer que le graphe complet K_n d'ordre n est n -coloriable et en déduire que $\omega(G) \leq \chi(G)$ où $\omega(G)$ est la taille de la plus grande clique d'un graphe quelconque.

Exercice 7 – coloration des sommets

Soit $G = (X, E)$ un graphe et son complémentaire $\bar{G} = (X, \bar{E})$. Montrer que

1. $\chi(G) \times \chi(\bar{G}) \geq n$
2. $\chi(G) + \chi(\bar{G}) \leq n + 1$ (par récurrence)
3. $\chi(G) + \chi(\bar{G}) \geq 2\sqrt{n}$
4. $\chi(G) \times \chi(\bar{G}) \leq (\frac{n+1}{2})^2$

Exercice 8 – coloration des sommets

Donner la valeur $\chi(G)$ pour les figures 1 et 2.

Exercice 9 – coloration des sommets

où $V(x_i)$ désigne le voisinage de x_i

1. Appliquer l'algorithme 1 sur la figure 3.
2. Trouver un exemple pour lequel deux couleurs suffisent pour colorier le graphe et en utilisant l'algorithme on obtient $n/2$ couleurs.

Exercice 10 – coloration des sommets

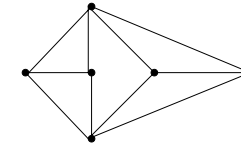


FIG. 1 – Exemple 1

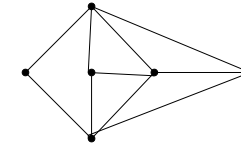


FIG. 2 – Exemple 2

On considère l'algorithme suivant :

1. Appliquer l'algorithme 2 sur la figure 3.
2. Trouver un exemple pour lequel deux couleurs suffisent pour colorier le graphe et en utilisant l'algorithme on obtient n couleurs.

Exercice 11 – coloration des sommets

Considérons le problème de codage suivant. On dispose d'un dictionnaire contenant quatre mots, chaque mot étant une séquence de trois symboles binaires $\{0, 1\}$: mot a : 000, mot b : 100, mot c : 110 et mot d : 101. Lorsqu'un mot est communiqué, on est certain que deux au moins des trois symboles binaires sont transmis sans erreur : une erreur portant sur un seul des trois symboles est possible. On désire retenir dans ce dictionnaire un sous-ensemble de cardinal maximum, dans les limites duquel un codage parfait soit possible, indépendamment des erreurs de transmission éventuelles (2 mots ne doivent pas être confondus en cas de transmission imparfaite).

Exercice 12 – coloration des sommets

Un groupe d'étudiants Y doit présenter un certain nombre d'épreuves écrites; l'ensemble de toutes les épreuves est noté X ; Comme il s'agit d'examens écrits, on désire que tous les étudiants qui doivent passer une même épreuve la présentent simultanément. Si un étudiant passe au plus un examen par jour, quelle est la durée minimale de la session ?

Algorithm 1 Donner la valeur de $\chi(G)$

Soit x_1, \dots, x_n une numérotation des sommets de G ,
 $C = \{1, 2, \dots, k\}$ un ensemble de couleurs;
for $i = 1$ à n **do**
 $\chi(x_i) = \min\{k \in C \text{ tel que pour tout } y \in V(x_i), \chi(y) \neq k\}$
end for

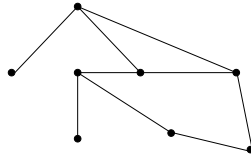


FIG. 3 –

Algorithm 2 Donner la valeur de $\chi(G)$

$C = \emptyset, k = \emptyset$

$d_1 \geq d_2 \geq \dots \geq d_n$, les degrés des sommets

repeat

$k := k + 1$

 colorier le sommet x ayant le plus haut degré avec la $k^{ième}$ couleur.

 marquer les voisins de x

$C := C \cup \{x\}$

while il existe un sommet y non marqué et non coloré **do**

 Colorier le sommet ayant le plus haut degré avec la $k^{ième}$ couleur.

 Marquer les voisins de y .

end while

 Effacer toutes les marques

until Jusqu'à ce que tous les sommets soient colorés

Exercice 13 – coloration de sommets

Démontrer la propriété suivante : Si dans un graphe G nous avons la séquence des degrés $d_1 \geq d_2 \geq \dots \geq d_n$ alors $\chi(G) \leq 1 + \max\{\min\{d_i, i - 1\} | 1 \leq i \leq n\}$.

Exercice 14 – coloration de sommets

1. Effectuer une coloration des sommets du graphe donné par la figure 4.
2. Soit $G' = (V', E')$ le sous-graphe construit à partir de G en ne conservant que les cinq premiers sommets x_1, \dots, x_5 . Notons $\bar{G}' = (V', \bar{E}')$ le graphe complémentaire de G' . Le graphe \bar{G}' est-il biparti?

Exercice 15 – coloration de sommets

Sachant que $\omega(G) \leq \chi(G)$ où $\omega(G)$ est la taille de la plus grande clique, on va montrer que cette borne peut-être très faible par rapport à la valeur $\chi(G)$. On veut trouver des graphes sans triangles est k -coloriables avec k aussi grand que l'on veut.

1. On va construire par récurrence ce graphe : le graphe M_1 est égal à K_1 , M_2 est égal à K_2 et M_{k+1} reprend M_k et ses p sommets x_1, x_2, \dots, x_p . On rajoute p autres sommets y_1, y_2, \dots, y_p , le sommet y_i étant relié à chaque voisin de x_i dans M_k , puis un sommet z relié à chaque y_i . Construisez M_1, M_2, M_3 et M_4 .
2. Montrer par récurrence que M_k admet $3 \times 2^{k-2} - 1$ sommets.
3. Donner le nombre d'arêtes.

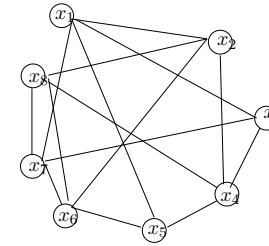


FIG. 4 –

4. Montrer qu'il est k -coloriable.

Exercice 16 – Des clients et des serveurs

Monsieur A a un immeuble. Dans cet immeuble des câbles ont été posés. Dans chaque pièce de l'immeuble arrive l'extrémité d'au moins un câble. Chaque câble permet de connecter directement deux ordinateurs. Monsieur A veut installer deux types de machines, des postes légers, les *clients* et des gros *serveurs*. Sa seule contrainte est de disposer exactement une machine par pièce, de la connecter à tous les câbles de la pièce dans laquelle elle se trouve, de telle sorte que deux machines du même type ne soient pas à l'extrémité du même câble.

- Modélisez ce problème et donnez une condition nécessaire et suffisante pour que cela soit réalisable.
- Si la condition est vérifiée, comment monsieur A peut-il disposer ses machines? (donnez un algorithme pour le faire automatiquement).

Exercice 17 – coloration de sommets

Soit $T = \{t_1, t_2, \dots, t_7\}$ un ensemble de 7 travaux, $M = \{m_1, m_2, \dots, m_7\}$ un ensemble de sept machines. Chaque travail t_i utilise un certain nombre de machines : t_1 utilise l'ensemble de machines $M_1 = \{m_1, m_3, m_5\}$, t_2 utilise l'ensemble de machines $M_2 = \{m_1, m_2, m_4\}$, t_3 utilise l'ensemble de machines $M_3 = \{m_2, m_3, m_5\}$, t_4 utilise l'ensemble de machines $M_4 = \{m_2, m_4, m_7\}$, t_5 utilise l'ensemble de machines $M_5 = \{m_5, m_6, m_7\}$, t_6 utilise l'ensemble de machines $M_6 = \{m_4, m_6, m_7\}$, t_7 utilise l'ensemble de machines $M_7 = \{m_5, m_6, m_7\}$. Le temps nécessaire à l'exécution de chaque travail t_i est le même, mais deux travaux t_i et t_j ($i \neq j$) ne peuvent être exécutés simultanément que s'ils utilisent des machines différentes ($M_i \cap M_j = \emptyset$). Représenter les contraintes de ce problème au moyen d'un graphe. Comment effectuer l'ensemble des travaux en un temps minimum?

Exercice 18 – coloration de sommets Soit un graphe G quelconque, alors $\chi(G) \leq 1 + \max\{\delta(H) : H \text{ est un sous-graphe induit de } G\}$ où $\delta(H)$ représente le degré minimale de H .

Exercice 19 – Polynômes chromatiques

Soit G un graphe quelconque. Nous appelons $P_G(k)$ le nombre de possibilités pour colorier les sommets du graphe G avec k couleurs. Si nous prenons le graphe donné par la figure 5 on a $P_G(k) = k(k-1)^2$ pour la chaîne et $P_G(k) = k(k-1)(k-2)$ pour le triangle.

1. Donner la fonction chromatique associée au graphe complet K_n .
2. Dans le cas où $k < \chi(G)$ que peut-on dire de $P_G(k)$ et dans le cas où $k \geq \chi(G)$.

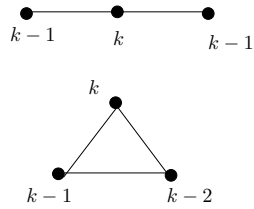


FIG. 5 – Exemple de fonction chromatique associé à un graphe

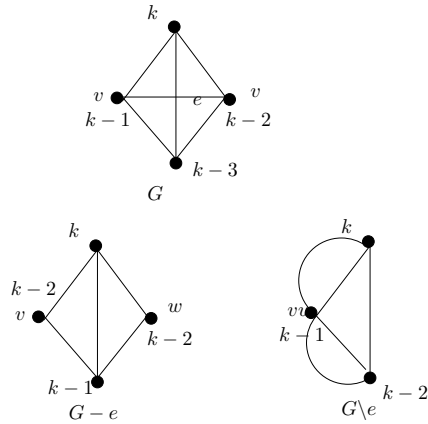


FIG. 6 – Illustration

3. Montrer que si G est un graphe simple, et soit $G - e$ et $G \setminus e$ les graphes obtenus à partir de G par suppression et contraction de l'arête e alors $P_G(k) = P_{G-e}(k) - P_{G \setminus e}(k)$. La figure 6 vous aidera à la démonstration. Dans un premier temps déterminer le polynôme chromatique du graphe donné par la figure 6 en utilisant le résultat.
4. Montrer que la fonction chromatique est un polynôme.
5. Donner le polynôme chromatique associé au graphe donné par la figure 7.
6. Démontrer (par récurrence) que pour un graphe simple G ayant n sommets et m arêtes alors le coefficient de k^n est 1, que le coefficient de k^{n-1} est $-m$ et que les coefficients de $P_G(k)$ sont alternativement positifs et négatifs.
7. Donner le polynôme chromatique pour le graphe complet K_5 .
8. Donner le polynôme chromatique pour le graphe biparti complet $K_{1,5}$.
9. Que vaut le polynôme chromatique d'un graphe G si admet plusieurs composantes connexes ?

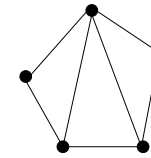


FIG. 7 – polynôme chromatique associé ?

10. Démontrer que si $P_G(k) = k(k-1)^{n-1}$ alors G est un arbre.
11. Déterminer les trois graphes ayant pour polynôme chromatique $k^5 - 4k^4 + 6k^3 - 4k^2 + k$.

2 Coloration d'arêtes

Exercice 20 – coloration d'arêtes

Soit $G = (X, E)$ un graphe. On note $\nu(G)$ la taille maximale d'un couplage. Montrer que

1. $\Delta(G) \leq \chi'(G)$
2. $\lceil \frac{|E|}{\nu(G)} \rceil \leq \chi'(G)$

Remarque on a pour tout graphe $G = (X, E)$, $\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1$

Exercice 21 – coloration d'arêtes

Quatre professeurs P_1, P_2, P_3, P_4 doivent donner un certain nombre d'heures de cours à trois classes C_1, C_2, C_3

Le professeur	doit donner une heure de cours aux classes
P_1	C_1 et C_2
P_2	C_1 et C_2
P_3	C_2 et C_3
P_4	C_2

Quel est le nombre minimum d'heures nécessaires pour disposer tous les cours ? Déterminer un horaire s'étalant sur ce nombre d'heures.

Exercice 22 – coloration d'arêtes

Montrer que

1. $\chi'(K_{2n}) = 2n - 1$
2. $\chi'(K_{2n+1}) = 2n + 1$

Algorithmique/Calculabilité/Complexité
TD – Séance n° 8

Exercice 1 – Calculabilité

1. Soit une suite quelconque d'ensembles E_i . Construire un ensemble qui n'appartient pas à cette suite (en vous inspirant de la diagonalisation).

Exercice 2 – Calculabilité

Soit $f : \mathbb{N} \rightarrow \{0, 1\}$ une fonction totale non calculable.

1. Rappeler la définition d'une fonction totale et une fonction non calculable.
2. Construire une fonction $g : \mathbb{N} \rightarrow \mathbb{N}$ totale, croissante et non calculable à partir de f .

Exercice 3 – Calculabilité

1. Une fonction $\mathbb{N} \rightarrow \mathbb{N}$ totale, non calculable et croissante, peut-elle être bornée?

Exercice 4 – Calculabilité

1. Montrer que l'inverse d'une fonction calculable et bijective est calculable.

Exercice 5 – Calculabilité

Nous considérons le tableau universel $t_i(j)$ = la valeur de la procédure i sur la donnée j . On considère une colonne j quelconque de ce tableau.

1. Montrer qu'il existe une ligne égale à cette colonne. Que pensez-vous de la réciproque?

Exercice 6 – Calculabilité

Nous considérons le tableau universel $t_i(j)$ = la valeur de la procédure i sur la donnée j . On considère une colonne j quelconque de ce tableau.

1. Montrer qu'il existe une ligne égale à cette colonne. Que pensez-vous de la réciproque?

Exercice 7 – Calculabilité

Montrer qu'une fonction totale $\mathbb{N} \rightarrow \mathbb{N}$ est calculable si et seulement si son graphe

$$G = \{(x, f(x)) | x \in \mathbb{N}\}$$

est décidable.

Exercice 8 – Calculabilité

Soient E un ensemble et ϕ une fonction telle que $\phi(n)$ est égale au nombre d'éléments de E strictement inférieur à n .

1. Montrer que ϕ est calculable si et seulement si E est décidable.

Exercice 9 – Calculabilité

En vous inspirant du théorème de Rice, donnez le prédicat (indécidable) et la fonction contradictoire qui prouve par l'absurde le résultat d'indécidabilité pour chacun des exemples suivants : on ne peut décider si une procédure calcule

1. une fonction totale
2. une fonction injective
3. une fonction croissante
4. une fonction à valeurs bornées

Exercice 10 – Calculabilité

1. Rappeler la définition d'un ensemble décidable.
2. Peut-on décider si un ensemble décidable est borné?

Exercice 11 – Calculabilité

On a vu en cours que tout ensemble fini est décidable.

1. Que peut-on en déduire sur le nombre de procédures qui calcule une fonction calculable bornée? Ce résultat est-il surprenant?

Exercice 12 – Calculabilité

On peut assimiler une procédure à son numéro. Donc $p < q$ où p et q sont des procédures, a un sens. On s'intéresse ici aux procédures C sans argument.

Par exemple

```
int p (void) {return 33;}
```

```
int q (void) {int i;e=1;
for (i=0;i<10;i++)
e:=2*e;
return e;}
```

soit des procédures telle que $p() = 33$ et $q() = 1024$. On peut définir $f(x) = \min\{p|p() = x\}$.

1. Montrer que f est une fonction totale qui n'est pas calculable. On pourra construire par exemple une procédure γ sans argument telle que γ calcule t et $\gamma < f(t)$.
2. Il semble que ce résultat soit surprenant (il semble facile d'avoir un algorithme qui calcule f). Où est l'erreur?

Exercice 13 – Calculabilité

Soit T une fonction calculable totale; comme les procédures peuvent être numérotées, T peut-être vue comme une transformation sur les procédures.

1. Montrer qu'il existe une procédure p telle que p et $T(p)$ calcule la même fonction (théorème du point fixe).
2. Montrer que T possède une infinité de points fixes.

Exercice 14 – Calculabilité

Soit f une fonction calculable, un ensemble B et son image réciproque par f , A :

$$A = f^{-1}(B) = \{x | f(x) \in B\}$$

1. Rappeler la définition d'un ensemble décidable et d'un ensemble récursivement énumérable.
2. A-t-on B décidable implique A décidable?
3. A-t-on B récursivement énumérable implique A récursivement énumérable?

Exercice 15 – Calculabilité

1. Montrer qu'un ensemble énuméré par une fonction calculable croissante f est décidable.
2. En déduire que tout ensemble récursivement énumérable contient un sous-ensemble infini et décidable.

Exercice 16 – Calculabilité

Soit E l'ensemble $val(f)$ où f est calculable partielle.

1. Montrer que E est récursivement énumérable (inspirez-vous de la preuve que l'arrêt en t unités de temps est décidable)

Exercice 17 – Calculabilité

1. Montrer que tout ensemble récursivement énumérable peut-être énuméré par une fonction sans répétition.

Exercice 18 – Calculabilité

Soient A et B deux ensembles décidables :

1. Est-on sûr que le complémentaire de A est décidable?
2. Est-on sûr que l'union de A et B est décidable?
3. Est-on sûr que l'intersection de A et B est décidable?
4. Même question en remplaçant décidables par récursivement énumérables.

Exercice 19 – Calculabilité

1. soit A un ensemble décidable de couples d'entiers. Montrer que la projection de A à savoir $E = \{x \mid \exists y \text{ tel que } (x, y) \in A\}$ est récursivement énumérable.
2. Montrer que réciproquement tout ensemble récursivement énumérable est la projection d'un ensemble décidable.

Complexité
TD – Séance n° 6

PROBLEMES DE DECISION ET PROBLEMES D'OPTIMISATION.

Exercice 1 – Passage du problème de décision au problème d'optimisation

1. Pour chacun des problèmes d'optimisation suivants, trouver le problème de décision associé et effectuer la réduction polynomiale de Turing.
 - (a) **Titre** : Chemin hamiltonien.
Données : un graphe orienté $G = (X, E)$.
Résultat : S'il existe, un chemin hamiltonien dans ce graphe, sinon le signaler.
 - (b) **Titre** : Satisfiabilité d'expressions booléennes
Données : un ensemble U de variables booléennes et un ensemble F de formules booléennes sur U .
Résultat : Fournir, s'il en existe, une interprétation qui satisfasse F . S'il n'en existe pas, le signaler.
 - (c) **Titre** : Isomorphisme.
Données : Deux graphes non orientés, $G = (X, U)$ et $H = (Y, V)$.
Résultat : Un sous graphe de H isomorphe à G s'il en existe, sinon le signaler.
 - (d) **Titre** : 3Mar
Données : Un ensemble $M \subseteq X \times Y \times Z$ où X, Y , et Z sont trois ensemble disjoints de même taille t .
Résultat : Un mariage, c'est à dire un sous ensemble $M' \subseteq M$ de taille t tel que deux points de M' n'aient aucune coordonnée en commun ¹
 - (e) **Titre** : Stable max.
Données : un graphe non orienté $G = (X, E)$.
Résultat : un stable (c'est à dire un sous-ensemble de sommets tel que deux sommets de ce sous ensemble ne soient jamais reliés par une arête) de taille maximum.

¹voici de ce problème une très morale présentation tirée de *Complexité algorithmique et problèmes de communications* de Barthélemy, Cohen et Lobstein :

On considère :

- un ensemble \mathcal{F} de jeunes filles,
- un ensemble \mathcal{B} de garçons jeunes, beaux, vigoureux mais pauvres,
- un ensemble \mathcal{R} de célibataires vieux, laids, moins vigoureux mais riches.

Chaque fille $f \in \mathcal{F}$ veut épouser un élément de \mathcal{R} et prendre comme amant un élément de \mathcal{B} en tenant compte de trois contraintes sociales, légales et hygiénique :

- f choisira mari et amant dans son cercle de connaissances. On dispose donc d'un sous ensemble $M \subseteq \mathcal{F} \times \mathcal{B} \times \mathcal{R}$ et $(f, b, r) \in M$ signifie que f connaît b et r .
- la loi interdit la bigamie : f n'aura qu'un seul mari et chaque mari n'aura qu'une épouse.
- chaque fille ne veut qu'un seul amant, et être la seule maîtresse de celui-ci.

- (f) **Titre** : couverture par les sommets.
Données : un graphe non orienté $G = (X, E)$.
Résultat : une couverture par les sommets (c'est à dire un sous-ensemble de sommets tel que toute arête du graphe aie au moins une extrémité dans ce sous ensemble) de taille minimum.
- (g) **Titre** : coloration
Données : un graphe non orienté $G = (X, E)$.
Résultat : une coloration minimale du graphe.
- (h) **Titre** : biparti maximum
Données : un graphe non orienté $G = (X, E)$.
Résultat : un sous ensemble $E' \subseteq E$ de taille maximum tel que (X, E') soit biparti.
- (i) **Titre** : Arbre recouvrant degré min.
Données : un graphe non orienté $G = (X, E)$.
Résultat : un arbre recouvrant de G de degré minimal.
- (j) **Titre** : plus long chemin.
Données :
 - i. un graphe orienté $G = (X, E)$;
 - ii. une valuation des arcs $v : E \rightarrow \mathbb{N}$
 - iii. deux sommets s et $t \in X$.**Résultat** : le plus long chemin de s à t .
- (k) Trouver un problème de décision associé à Partition et effectuer la réduction polynomiale de Turing.

CERTIFICATS POLYNOMIAUX
REDUCTIONS

Exercice 1 – Fonction à sens uniques

Une fonction à sens unique est une fonction f tel que le calcul de $y = f(x), \forall x$ est facile à calculer mais à parit de $y = f(x)$ il est difficile à calculer x . L'existence de fonctions à sens unique est une conjecture aussi difficile que $\mathcal{NP} \neq \mathcal{P}$.

On considère la famille $(f_n)_{n \in \mathbb{N}}$ tel que f_n est une bijection de $A_n = \{1, \dots, n\}$ dans $B_n = \{1, \dots, n\}$.

Soit le problème \mathcal{F} suivant : soit $n \in \mathbb{N}$. Soient x et t deux entiers entre 1 et n . Soit f_n la fonction décrite précédemment. A-t'on $f_n(x) \leq t$? Si $\forall n, f_n(x)$ est polynomial, le problème \mathcal{F} est trivialement dans \mathcal{P} . Nous supposons que $\mathcal{F} \in \mathcal{P}$.

Le calcul de l'inverse des f_n est associé au problème de décision suivant noté \mathcal{F}^{-1} : soit $n \in \mathbb{N}$. Soient x et t deux entiers entre 1 et n . A-t'on $f_n^{-1}(x) \leq t$?

Montrer que $\mathcal{F}^{-1} \in \mathcal{NP}$.

Exercice 2 – Certificats polynomiaux et réductions polynomiales

1. Quels sont les certificats des problèmes de décision suivants et sont-ils des certificats polynomiaux :
 - (a) Partition
 - (b) Circuit Hamiltonien
 - (c) SAT
 - (d) Clique Max

Mêmes questions pour les co-problèmes (la question est formulée de façon négative).

2. La raison pour laquelle la réduction de SAT à 3-SAT est polynomiale.
Soit un ensemble de clauses composé d'une clause de longueur 4 et d'une clause de longueur 5. Pour construire comme vu dans le cours un ensemble équivalents composé seulement de clauses de longueur 3, combien dois-je introduire de variables booléennes supplémentaires?
3. On veut prouver que si $(k+1)$ SAT est NP-complet, alors k -SAT est NP-complet.
Pourquoi ce résultat est-il manifestement erroné?
Où est l'erreur du raisonnement suivant :
Soit \mathcal{C} un ensemble de clauses de longueur k , j'applique l'outil \mathcal{O}_1 (vu en cours) à chacune des clauses pour obtenir un ensemble équivalent de (deux fois plus de) clauses de longueur $k+1$. Cette transformation est polynomiale et comme $(k+1)$ SAT est NP-complet, k -SAT l'est aussi.
4. On admet que 3-coloration est NP-Complet. Montrer que 4-coloration est aussi NP-complet (c'est à dire réduire 3-coloration à 4-coloration).

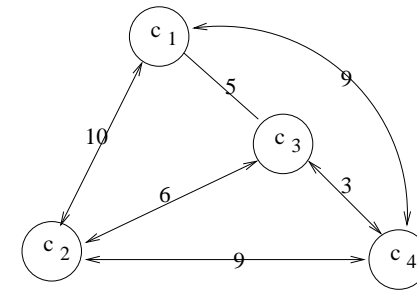


FIG. 1 – Un exemple de problème du Voyageur de commerce.

5. Réduire chaîne hamiltonienne à cycle hamiltonien.
6. En admettant que circuit hamiltonien est NP-complet, montrer que le problème du voyageur de commerce est NP complet :
Titre : voyageur de commerce.
Données :
 - un graphe orienté
 - une valuation sur les arcs de ce graphe
 - un entier k**Résultat :** existe-t-il dans le graphe un circuit hamiltonien dont la valeur soit inférieure ou égale à k ?
Quelle est la valeur k maximum pour laquelle la réponse est OUI dans le graphe de la figure 1?
7. montrer que si l'un des 3 problèmes Clique, Stable, Couverture par les sommets est NP-complet, les 3 le sont.

Exercice 2 – Sur le problème du stable maximum

1. Écrire l'algorithme TSM avec une complexité dans $\mathcal{O}(1, 39^n)$.

PROGRAMMATION DYNAMIQUE. RECHERCHE DE STABLE MAXIMUM

Exercice 1 – Programmation dynamique

1. Montrer que le problème *Somme de sous ensemble* est NP complet.
2. Écrire l'algorithme pseudo polynomial qui résolve le problème de décision *Somme de sous ensemble*.
3. Écrire l'algorithme pseudo polynomial qui résolve le problème d'optimisation *Somme de sous ensemble*.
4. On va s'intéresser au problème du *sac à dos*
 - rappeler le problème, puis trouver un algorithme pseudo polynomial
 - qui résolve le problème de décision.
 - qui résolve le problème d'optimisation.

5. Produit de matrices rectangulaires.

– **Le problème.**

On peut multiplier une matrice M_1 de l_1 lignes et c_1 colonnes par une matrice M_2 de l_2 lignes et c_2 colonnes si et seulement si $c_1 = l_2$ et cette multiplication nécessite alors $l_1 \times l_2 \times c_2$ multiplications.

Mais que se passe-t-il quand on veut multiplier une suite de matrices $M_1, M_2 \dots M_n$? Pour dénoter les dimensions de ces matrices, on utilisera une suite de nombres $n_1, n_2 \dots n_n, n_{n+1}$ et on dénotera n_i le nombre de lignes de la matrice M_i et n_{i+1} son nombre de colonnes.

Il s'agit de minimiser le nombre de multiplications.

– **Exemple : multiplication de trois matrices M_1, M_2 et M_3**

avec $n_1 = 10, n_2 = 100, n_3 = 5, n_4 = 50$.

Le produit $M_1 \times M_2$ demande 5000 multiplications, faire le produit de ce résultat par M_3 nécessite encore 2500 multiplications, soit au total 7500.

Le produit $M_2 \times M_3$ demande 25000 multiplications, faire le produit de M_1 par ce résultat nécessite encore 50000 multiplications, soit au total 75000.

La première solution est donc la meilleure.

Mais comment faire le produit de 6 matrices, avec une suite de dimensions (30, 35, 15, 5, 10, 20, 25)?

– **La solution.**

On note $c_{i,j}$ le nombre minimal de multiplications pour faire le produit $M_i, M_{i+1} \dots M_j$.

On a $c_{i,i} = 0$ et $c_{i,i+1} = n_i \times n_{i+1} \times n_{i+2}$.

On découpe alors le produit $M_i \dots M_j$ en le produit $M_i \dots M_k$ et le produit $M_{k+1} \dots M_j$ en choisissant le meilleur k

$$c_{i,j} = \min_{k \in \{i \dots j-1\}} c_{i,k} + c_{k+1,j} + n_i n_{k+1} n_{j+1}$$