Travaux pratiques 8

Flots

Option informatique MP1, MP2 & MP3

L'objet de ce TP/TD est d'étudier un problème d'optimisation dans un graphe valué. Comment transporter un maximum de quantité dans un graphe quand chaque arête a un débit maximal ?

I Méthode de Ford et Fulkerson

Dans cette sous-partie nous allons définir et caractériser la notion de flot maximal. Cette caractérisation va permettre d'imaginer une méthode de calcul qu'il faudra préciser ensuite.

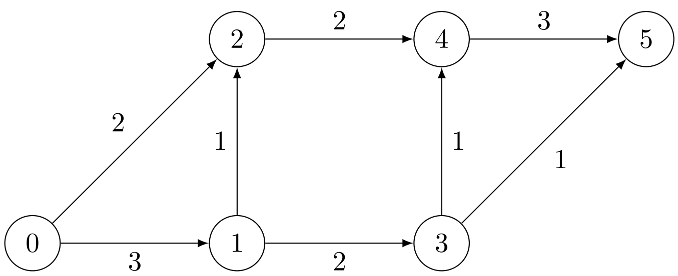
I.1 Flot

Définition 1 - Réseau

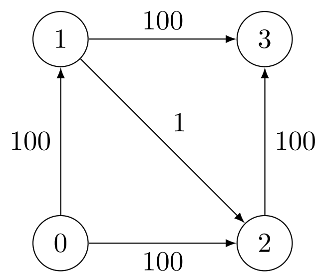
Un réseau est un quintuplé $R = (S, A, w, s, p)$ tel que

- $G = (S, A, w)$ graphe orienté valué,
- w est à valeurs dans $\mathbb{N}$,
- s est un sommet de G, la source,
- p est un sommet de G, le puits, avec $s \neq p$.

Exemples de réseaux



R_1, $s = 0$, $p = 5$



R_2, $s = 0$, $p = 3$

Définition 2 - Capacité

La capacité du réseau est la fonction c de $S \times S$ vers $\mathbb{N}^+$ définie par

- $c(i, j) = w(i, j)$ si $(i, j) \in A$
- $c(i, j) = 0$ sinon.

Définition 3 - Flot

Un flot sur un réseau $R = (S, A, w, s, p)$ est une fonction φ de $S \times S$ vers $\mathbb{N}$ telle que

- $\varphi(i,j) \leqslant c(i,j)$ pour tous sommets i et j,
- $\varphi(j,i) = -\varphi(i,j)$ pour tous sommets i et j
- $\displaystyle\sum_{j\in S} \varphi(i,j) = 0$ pour tout sommet i distinct de s et p.

La valeur du flot est $|\varphi| = \displaystyle\sum_{j\in S} \varphi(s,j)$.

Exercice 1 - Nullité
Prouver que si φ est un flot d'un graphe et si ni (i,j), ni (j,i) n'est une arête du graphe alors $\varphi(i,j) = 0$. En particulier $\varphi(i,i) = 0$.

Exercice 2 - Valeur au puits
Prouver que, pour toute partie T de S, $\displaystyle\sum_{i\in T}\sum_{j\in T} \varphi(i,j) = 0$.

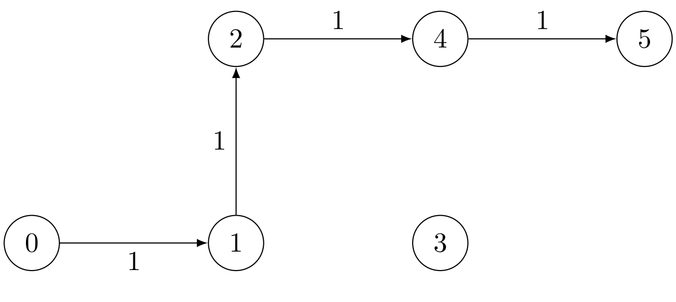
En déduire que $|\varphi| = \displaystyle\sum_{i\in S} \varphi(i,p)$.

Exercice 3 - Flot associé à un chemin
On suppose que le graphe d'un réseau admet un chemin simple de s vers p :

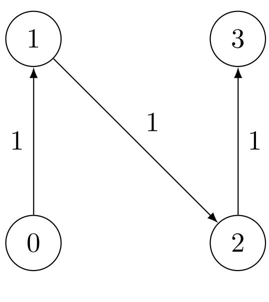
$$s = s_0 \to s_1 \to \cdots \to s_{r-1} \to s_r = p$$

On définit $\omega = \min\{w(s_{i-1}, s_i) \ ; \ 1 \leqslant i \leqslant r\}$
et φ par $\varphi(s_{i-1}, s_i) = \omega$, $\varphi(s_i, s_{i-1}) = -\omega$ et $\varphi(i,j) = 0$ pour les autres couples.
Prouver que φ est un flot (c'est le flot associé au chemin) avec $|\varphi| > 0$.

Exemples de flots associés à un chemin. On ne représente que les flots strictement positifs, les flots strictement négatifs correspondent aux inverses des arêtes.



$|\varphi_1| = 1$

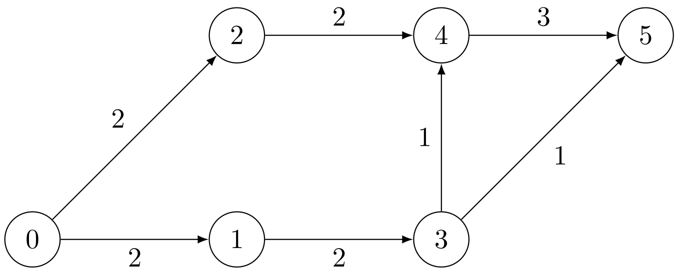


$|\varphi_2| = 1$

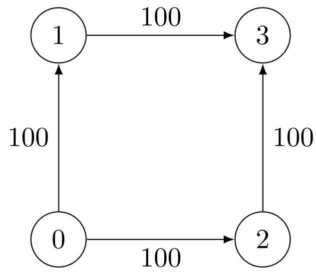
I.1.a Propriétés

- Il existe au moins un flot : le flot nul défini par $\varphi(i,j) = 0$ pour tous sommets.
- Pour tout flot φ, on a $|\varphi| = \sum_{j \in S} \varphi(s,j) \leqslant \sum_{j \in S} c(s,j)$.
- L'ensemble des valeurs des flots est donc non vide et majoré, il admet une borne supérieure.
- On cherche un flot φ_M (s'il en existe) tel que $|\varphi_M|$ est maximum.

Exemples de flots maximaux



$|\varphi_{1,M}| = 43$



$|\varphi_{2,M}| = 200$

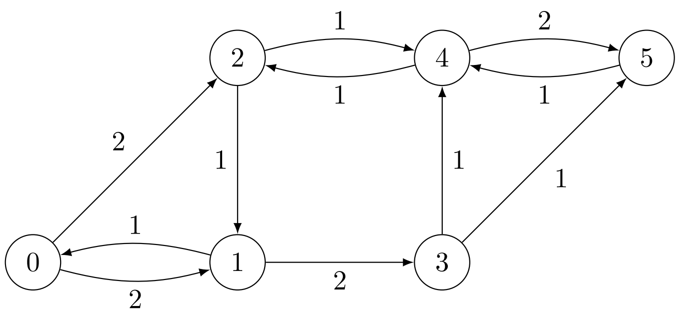
I.2 Chemin améliorant

Nous allons chercher à augmenter la valeur d'un flot.

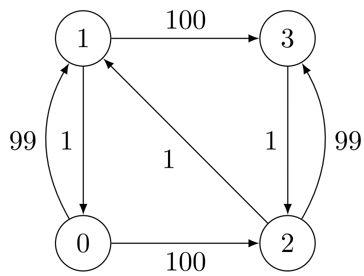
Définition 4 - Réseau résiduel

Si φ est un flot sur le réseau $R = (S, A, w, s, p)$,
le réseau résiduel associé est $R_\varphi = (S, A_\varphi, w_\varphi, s, p)$
où A_φ est l'ensemble des couples (i,j) tels que $\varphi(i,j) < c(i,j)$
et $w_\varphi(i,j) = c(i,j) - \varphi(i,j)$.

Exemples de réseaux résiduels
On part des flots associés à un chemin, vus ci-dessus



R_{φ_1}



R_{φ_2}

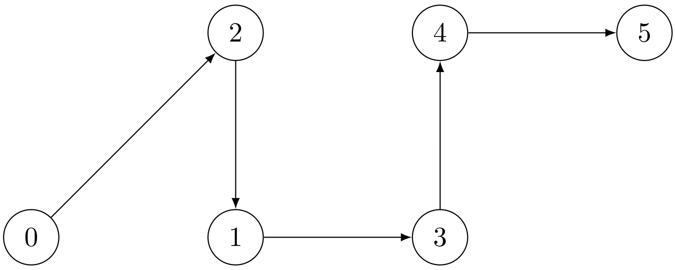
On remarque qu'on a enlevé des arêtes de R et introduit des arêtes supplémentaires en sens inverse par rapport à celles de R. Ces arêtes inversées correspondent à la possibilité de diminuer le flot qui suit l'arête originale.

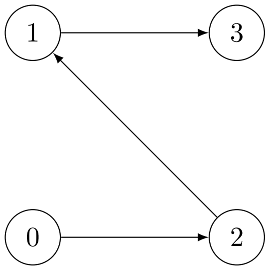
Exercice 4 - Combinaison de flots
Prouver que si φ est un flot du réseau $R = (S, A, w, s, p)$ et si ψ est un flot du réseau R_φ alors $\varphi + \psi$ est un flot de R avec $|\varphi + \psi| = |\varphi| + |\psi|$.

Définition 5 - Chemin améliorant

Si φ est un flot sur le réseau $R = (S, A, w, s, p)$, un chemin améliorant est un chemin de s vers p dans le réseau résiduel associé R_φ.

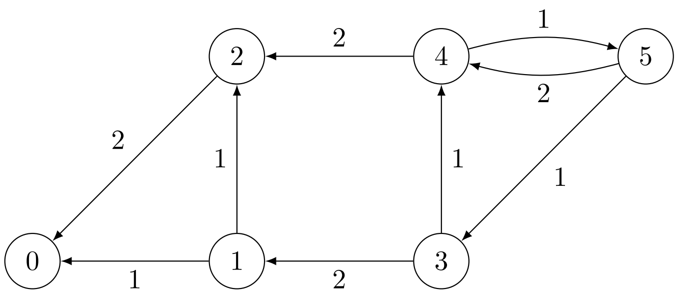
Exemples de chemins améliorants : on reprend les même exemples de flots



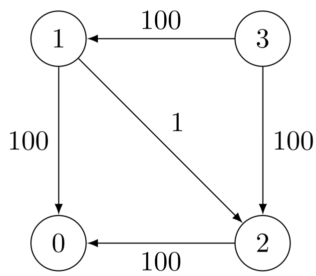


Exercice 5 - Condition nécessaire
Prouver que si φ est un flot maximal de R alors R_φ n'admet pas de chemin améliorant.

Exemples de réseaux résiduels pour des flots maximaux
Ils n'admettent pas de chemin de s à p.



$R_{\varphi_{1,M}}$



$R_{\varphi_{2,M}}$

La méthode de Ford-Fulkerson s'en déduit

- R est un réseau
- φ est le flot nul
- $R' \leftarrow R$, (réseau résiduel associé au flot nul)
- **tant que** R' admet un chemin entre s et p
- on choisit un chemin de s à p
- on détermine le flot de R' associé à ce chemin
- $\varphi \leftarrow \varphi + \psi$
- $R' \leftarrow R_\varphi$
- φ est un flot maximal ?

Il reste à prouver que l'absence de chemin améliorant caractérisait la maximalité du flot.
Il faut prouver la réciproque du résultat de l'exercice 5.

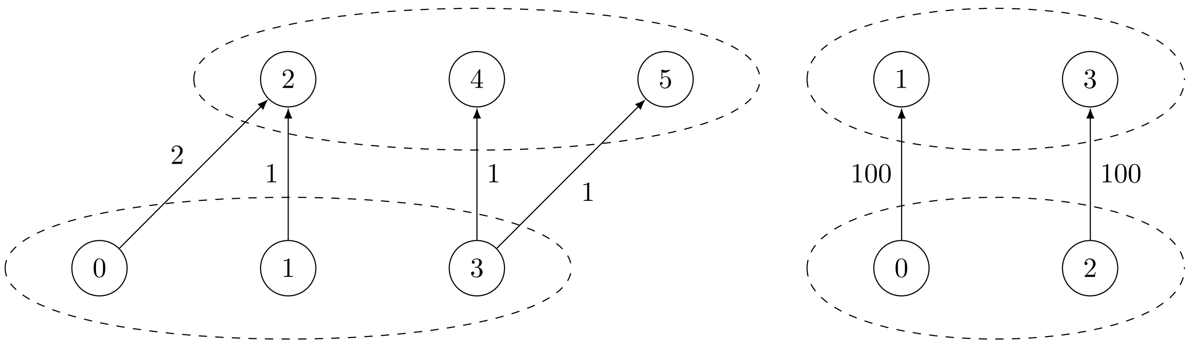
II Coupure

On introduit une nouvelle notion qui va servir d'étape intermédiaire pour prouver l'algorithme de Fulkerson

Définition 6 - Coupure

Une coupure d'un réseau $R = (S, A, w, s, p)$ est une partition (S_s, S_p) de l'ensemble S des sommets telle que $s \in S_s$ et $p \in S_p$.
La capacité d'une coupure est $c(S_s, S_p) = \sum_{i \in S_s} \sum_{j \in S_p} c(i, j)$.

Exemples de coupures



$c = 5$ $c = 200$

Exercice 6 - Majoration de la valeur d'un flot
Prouver que si φ est un flot du réseau $R = (S, A, w, s, p)$
et si (S_s, S_p) est coupure du réseau alors $|\varphi| \leqslant c(S_s, S_p)$.
On pourra prouver qu'on a $|\varphi| = \sum_{i \in S_s} \sum_{j \in S_p} \varphi(i, j)$.

On retrouve un autre majorant pour les valeurs d'un flot : c'est la capacité d'une coupure.

Exercice 7 - Coupure minimale pour un flot
Prouver que s'il existe une coupure (S_s, S_p) de R telle que $c(S_s, S_p) = |\varphi|$ pour un flot φ alors ce flot est maximal.
Une telle coupure sera dite **minimale** pour un flot φ.

Exercice 8 - Existence d'une coupure minimale pour un flot maximal
Prouver que s'il n'existe pas de chemin améliorant dans R_φ alors il existe une coupure minimale pour φ dans R.

En rassemblant les exercice 7, 8 et 5 on a le

Théorème 1 - Théorème de Ford et Fulkerson

Si φ est un flot d'un réseau $R = (S, A, w, s, p)$ alors les propriétés suivantes sont équivalentes.

1. φ est un flot maximal.
2. Il existe une coupure minimale pour φ dans R.
3. Il n'existe pas de chemin améliorant dans R_φ.

Remarque Si les poids étaient réels, le théorème ne permettrait pas d'obtenir un algorithme de calcul du flot maximal. En effet une suite strictement croissante des valeurs des flots réelles n'atteint pas toujours sa limite après un nombre fini d'itérations. La méthode n'est pas assurée de terminer.

Exercice 9 - Complexité
Prouver que cette méthode termine après au plus C itérations avec $C = \sum_{i \in S} c(s, i)$.
En considérant le graphe R_2 avec un choix systématique d'un chemin améliorant qui passe par la diagonale $(1, 2)$ montrer que ce majorant peut être atteint.

III Algorithme de Edmonds-Karp

L'algorithme de Edmonds-Karp consiste à implémenter la méthode de Ford-Fulkerson en choisissant comme chemin améliorant celui que fournit un parcours en largeur depuis s.

- On part d'un réseau $R = (S, A, w, s, p)$.
- φ est un flot sur R, $R_\varphi = (S, A_\varphi, w_\varphi, s, p)$ est le réseau résiduel associé.
- S_φ est l'ensemble des sommets accessibles depuis s dans R_φ. Pour $t \in S_\varphi$, on note $d_\varphi(t)$ la longueur du plus court chemin (en nombre d'arêtes) de s à t dans R_φ.
- Si on a $p \in S_\varphi$, le parcours en largeur depuis s dans R_φ donne un chemin augmentant c.
- ψ est le flot de R_φ associé ; $\psi(a) = \omega$ pour toute arête de c où ω est poids minimal des arêtes de c (dans $R_{\varphi'}$).
- On note $\varphi' = \varphi + \psi$ le flot augmenté et $R_{\varphi'} = (S, A_{\varphi'}, w_{\varphi'}, s, p)$ est le réseau associé.
- Une arête de R_φ est dite **critique** si elle n'appartient pas à $R_{\varphi'}$.
- Un arête de $R_{\varphi'}$ est dite **émergente** si elle n'appartient pas à R_φ.

III.1 Complexité

Exercice 10 - Caractérisations des arêtes critiques
Prouver que les arêtes critiques sont les arêtes de c de poids ω dans R_φ.

Exercice 11 - Caractérisations des arêtes émergentes
Prouver que les arêtes émergentes sont les arêtes (i, j) n'appartenant pas à $A_{\varphi'}$ telles que (j, i) est une arête de c.
En déduire que i ne peut être égal à s, ni j égal à p.

Exercice 12 - Décroissance de S_φ
Prouver qu'on a $S_{\varphi'} \subset S_\varphi$.

Exercice 13 - Augmentation des distances
On suppose que (i, j) est une arête émergente appartenant à un chemin minimal depuis s.
Montrer qu'on a $d_{\varphi'}(j) - d_{\varphi}(j) = d_{\varphi'}(i) - d_{\varphi}(i) + 2$.
En déduire que $d_{\varphi}(t) \leqslant d_{\varphi'}(t)$ pour tout $t \in S_{\varphi'}$.

Exercice 14 - Dénombrements
Montrer qu'une même arête ne peut être critique que $|S|/2$ fois au maximum.
En déduire l'algorithme de Edmonds-Karp ne peut faire plus de $|A| \times |S|$ itérations de recherche de chemin minimal.
Que peut-on en déduire pour la complexité de l'algorithme de Edmonds-Karp ?

III.2 Implémentation

On choisit de coder les graphe par leur matrice des poids, l'absence d'arête est représentée par un poids nul.
La fonction de flot est aussi représentée par la matrice de ses valeurs.

Exercice 15 - Parcours en largeur
Écrire une fonction `chemin g a b` qui reçoit une matrice représentant un graphe et qui renvoie un chemin sous forme de liste entre a et b. Ce chemin devra être celui que fournit le parcours en largeur.

Dans l'algorithme on modifie à chaque étape la fonction φ et donc la matrice du graphe associé à R_{φ}. On le fait en ajoutant ω à $\varphi(i, j)$ et en le retranchant à $\varphi(j, i)$ pour toute arête du chemin améliorant choisi ; ω est le minimum des poids des arêtes de ce chemin. On devra alors retrancher ω du poids le l'arête (i, j) et l'ajouter au poids le l'arête (j, i) dans la matrice du graphe.

Exercice 16 - Modification des matrices
Écrire une fonction `modif chemin g phi` qui reçoit une liste représentant un chemin dans un graphe `g` et une matrice représentant un flot et qui modifie les matrices `g` et `phi` comme indiqué ci-dessus.

Exercice 17 - Recherche du flot
Écrire une fonction `flotMax g s p` qui reçoit une matrice représentant un graphe et le sommet et le puits et qui renvoie une fonction de flot maximal représentée par sa matrice.