Flots et Couplages

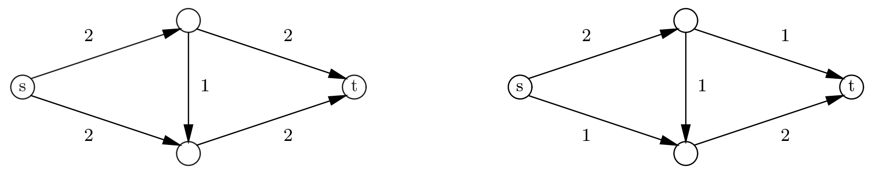
Définitions

Réseau de transport : Graphe orienté $G = (V, E)$, avec deux sommets particuliers s et t, tels que s n'a pas d'arc entrant et t n'a pas d'arcs sortant. Chaque arc a a une *capacité* $c(a)$.

Un *flot* est une application ϕ de A dans $\mathcal{N}$ telle que :

1. Le flot traversant un arc n'excède pas sa capacité : $\phi(a) \leq c(a)$
2. Pour tout sommet $u \neq s, t$, le flot entrant dans u est égal au flot sortant de u.

La *valeur* du flot est égale à la somme des flots sortants de s.
Problème : trouver un flot de valeur maximale.



Un réseau de transport   Un flot sur le graphe

Figure 1: Flot saturé non maximal

Un arc est *saturé* si $\phi(a) = c(a)$. Un flot est saturé si tout chemin de s à t possède un arc saturé.

Coupes

Une *coupe* est une partition de V en deux sous-ensembles disjoints $V = Y \cup Z$ tels que $s \in Y$ et $t \in Z$. Sa *capacité* est la somme des capacités des arcs qui ont leur origine dans Y et leur extrémité dans Z.

Pour un flot ϕ et une coupe Y, Z on pose

$$\phi(Y, Z) = \sum_{a \in (Y,Z)} \phi(a) \text{ et } \Delta(Y, Z) = \phi(Y, Z) - \phi(Z, Y).$$

Flots et coupes

Pour toute coupe (Y, Z) et tout flot, $\Delta(Y, Z)$ est égal à la valeur du flot.

Preuve : par récurrence sur $|Y|$

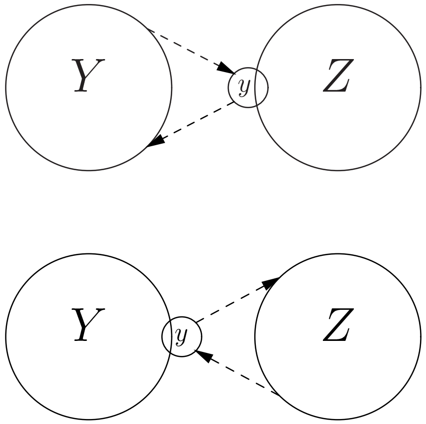


Figure 2: Conservation du flot

Lemme

Pour toute coupe (Y, Z) et tout flot, la valeur du flot est inférieure ou égale à $C(Y, Z)$.

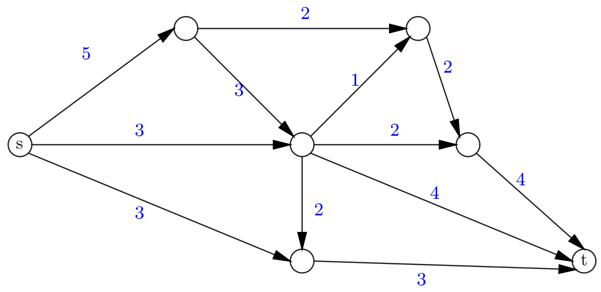
Théorème de Ford-Fulkerson

Le flot maximal a pour valeur la capacité de la coupe minimale.

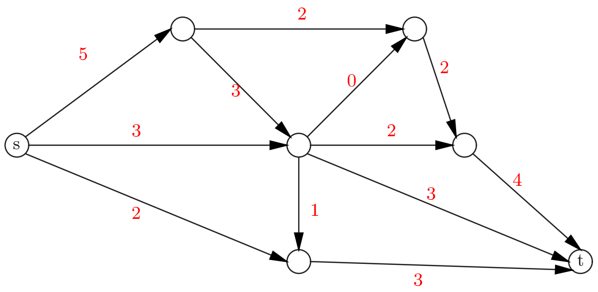
Graphe des augmentations

Etant donné un flot ϕ, on définit un graphe G_ϕ de sommets V et tel que :

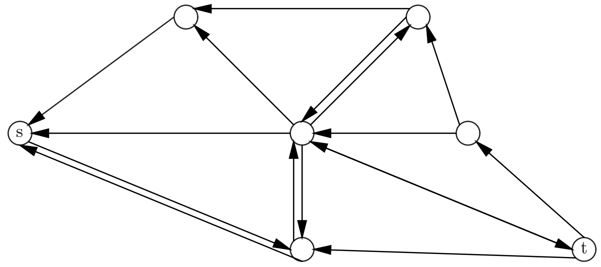
- si $\phi(a) < c(a)$ alors on met l'arc a dans G_ϕ
- si $\phi(a) > 0$ alors on met l'arc inverse de l'arc a dans G_ϕ



Un graphe et ses capacités



Un flot sur ce graphe



Le graphe des augmentations

Lemmes

S'il existe un chemin joignant s à t dans G_ϕ, alors le flot peut être amélioré.

S'il n'existe pas de chemin joignant s à t dans G_ϕ, alors il existe une coupe (Y, Z) de capacité égale à la valeur du flot.

Algorithme de Ford et Fulkerson

Initialisation : On calcule un flot quelconque (par exemple nul)

Boucle : Tant qu'il existe un chemin de s à t dans le graphe des augmentations, on augmente le flot

Sortie : On obtient ainsi le flot maximal

La coupe minimale peut être obtenue en considérant l'ensemble des sommets atteignables à partir de s dans le graphe des augmentations

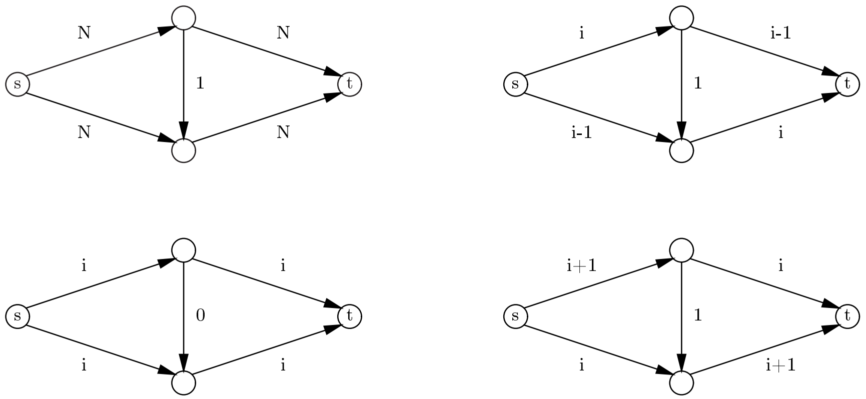


Figure 3: Un exemple de calcul très lent

Algorithme de Dinic, Edmonds et Karp

Initialisation : On calcule un flot saturé

Tant qu'il existe un chemin dans le graphe des augmentations, on augmente le flot en utilisant le *plus court chemin* de s à t (parcours en largeur)

Analyse

Proposition : Lors de l'exécution, la longueur du chemin amélioré ne décroît jamais.

Théorème : Le coût de l'algorithme d'Edmonds-Karp est $O(nm^2)$ opérations.

Application : choix d'objets à emporter

Entrée :

- On peut emmener dans une navette spatiale un certain nombre d'instruments $A_1, A_2, \ldots, A_p$; coût de transport c_i
- Commandes d'expériences dans l'espace $B_1, B_2, \ldots, B_q$; le gain réalisé si l'on effectue l'expérience B_j est noté b_j
- Pour pouvoir réaliser B_i, il faut d'avoir emporté un sous-ensemble de l'ensemble des instruments

Sortie : Déterminer quels instruments emporter et quelles expériences réaliser pour maximiser le gain

Réduction à un problème de coupe

sommets $\{s, A_1, \ldots A_p, B_1 \ldots B_q, t\}$

arc (s, A_i) de capacité c_i

arc (B_j, t) de capacité b_j

arc (A_i, B_j) si l'expérience B_j nécessite l'instrument A_i ; capacité non bornée

Couplage maximum dans un graphe biparti

Un graphe non orienté est *biparti* s'il existe une partition des sommets en deux sous ensembles tels que toute arête ait une extrémité dans chacun des deux sous-ensembles.

Couplage du graphe : ensemble d'arêtes sans extrémités communes.

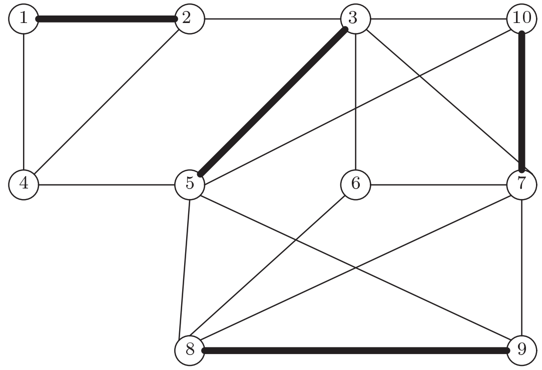


Figure 4: Un couplage

Réduction à un problème de flot

On ajoute deux sommets s et t, on relie s à tous les sommets de Y et tous les sommets de Z à t, on oriente les arêtes de Y vers Z, la capacité est de 1 pour tous les arcs.

Chemins alternants

- Les chemins dans le graphe des augmentations deviennent ici des chemins alternants
- Chemins formés alternativement d'une arête extérieure au couplage et d'une arête appartenant au couplage
- Arête extérieure : on peut augmenter le flot. Arête du couplage : on peut diminuer le flot
- On peut montrer que le couplage est maximum si et seulement si il n'existe pas de chemin alternant joignant deux sommets non saturés

Généralisation : mariages stables

Entrée : Deux ensembles $X = \{x_1, x_2, \ldots, x_p\}$ et $Y = \{y_1, y_2, \ldots, y_p\}$.

A chaque x_i est associée une liste ordonnée F_i d'éléments de Y.

A chaque y_j est associée une liste ordonnée G_j d'éléments de X.

On travaille dans le graphe biparti où les arêtes sont les couples $\{x, y\}$ tels que y figure dans la liste F_x et x dans la liste G_y

Un couplage C est *stable* si et seulement si pour tout couple x, y qui n'est pas dans C on n'a pas : x préfère y à sa partenaire, et y préfère x à son partenaire.

Mariages stables

Considérons le cas $p = q = 3$ et les listes

$$\begin{array}{llll} F_1 & : & y_2 & y_1 & y_3 \\ F_2 & : & y_1 & y_3 & y_2 \\ F_3 & : & y_1 & y_2 & y_3 \end{array} \qquad \begin{array}{lllll} G_1 & : & x_1 & x_3 & x_2 \\ G_2 & : & x_3 & x_1 & x_2 \\ G_3 & : & x_1 & x_3 & x_2 \end{array}$$

Le couplage $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ n'est pas stable.

Le couplage $(x_1, y_1), (x_2, y_3), (x_3, y_2)$ est stable.

Algorithme

On initialise $\alpha(x)$ au premier élément de la liste F_x et $\beta(y)$ indéfini. Ensuite on effectue des modifications itérativement de β et de α comme suit :

1. Pour chaque y choisir l'élément x le plus intéressant dans la liste G_y parmi les x tels que $\alpha(x) = y$. et poser $\beta(y) = x$
2. Pour chaque x,
   - S'il existe y tel que $\beta(y) = x$, alors $\alpha(x) = y$.
   - S'il n'existe pas de y tel que $\beta(y) = x$ et si $\alpha(x)$ n'est pas le dernier élément de la liste F_x alors remplacer $\alpha(x)$ par l'élément qui le suit dans dans la liste F_x,
   - Si aucune des deux conditions précédentes n'est remplie alors $\alpha(x)$ est indéfini.