Flot Maximum

V. Limouzy

2 novembre 2020

Le problème

Domaine d'application

- Réseau routier
- Réseau de canalisation
- Réseau de machines de productions
- Réseau internet
- ...

Informellement

On cherche à faire transiter le plus de matière sur un réseau. Chaque lien du réseau a une capacité. (p. ex 2500 voitures/heures pour une route). On ne doit pas excéder la capacité.

Intérêt

- Problème très général.
- Permet de modéliser et résoudre beaucoup de problème.
- Temps polynomial pour trouver une solution.
- Problème qui affiche une dualité min./max. : en général algo polynomial.
- Sera considéré sous forme de programme linéaire au S6.

Formellement

Donnée :

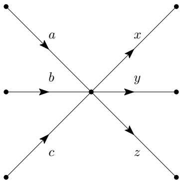
- Donnée :$G = (V, E, c, f)$ graphe orienté
- s et t deux sommets spéciaux : source et puits
- c fonction de capacité $c : E \to \mathbb{R}^+$
- f un flot qui transite sur les arêtes $f(e) \leq c(e) \forall e \in E$.
- Objectif : trouver la plus grande quantité de flot à faire transiter la plus grande quantité entre s et t.

Loi de Kirchoff

Pour tous les sommets v, sauf s et t, on a la conservation du flot :

$$\sum_{u \in N^-(v)} f(u,v) = \sum_{w \in N^+(v)} f(v,w)$$

En d'autres termes, les sommets de stocke pas et ne fabrique pas de la matière. Tout ce qui arrive doit repartir aussi tôt.



$a + b + c = x + y + z$

La source s et le puits t peuvent générer (resp. absorber) une quantité infinie de matière.
On peut mesurer la valeur du flot, en comptabilisant tout ce qui sort de s (ou tout ce qui entre sur t).

Flot inverse

Si on a un arc (u, v) dans G avec un flot $f(u, v)$ qui transite. On supposera que sur (v, u) le flot qui transite est $-f(u, v)$.

Flot inverse

Si on a un arc (u, v) dans G avec un flot $f(u, v)$ qui transite. On supposera que sur (v, u) le flot qui transite est $-f(u, v)$.

Arcs manquants

Pour les non-arcs (a, b) du réseau, *i.e.* qui n'appartiennent pas à E. On considère $c(a, b) = 0$.

Flot inverse

Si on a un arc (u, v) dans G avec un flot $f(u, v)$ qui transite. On supposera que sur (v, u) le flot qui transite est $-f(u, v)$.

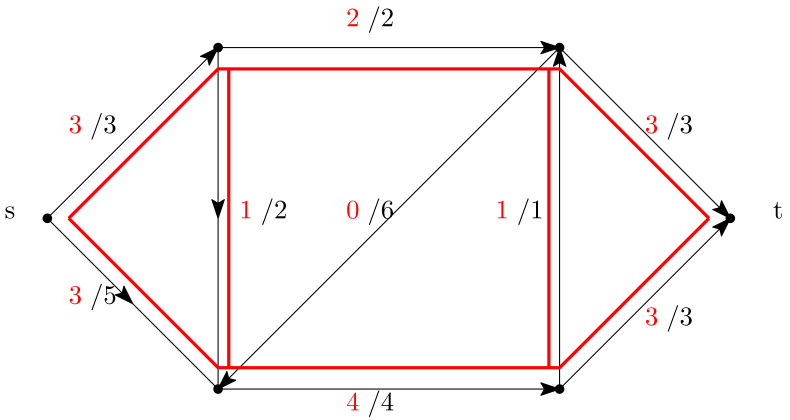
Arcs manquants

Pour les non-arcs (a, b) du réseau, *i.e.* qui n'appartiennent pas à E. On considère $c(a, b) = 0$.

Valeur du flot

On note $|f|$ la valeur du flot qui transite sur G.

Exemple



Multi sources/ Multi puits

On peut considérer un problème plus général où on a :

- Plusieurs sources $S = \{s_1, s_2, \ldots, s_k\}$
- Plusieurs puits $T = \{t_1, t_2, \ldots, t_l\}$

Et on cherche à maximiser le flot total.

Équivalence avec le problème initial

On peut se ramener au problème initial (*i.e.* une seule source/ un seul puits). En modifiant le réseau :

Multi sources/ Multi puits

On peut considérer un problème plus général où on a :

- Plusieurs sources $S = \{s_1, s_2, \ldots, s_k\}$
- Plusieurs puits $T = \{t_1, t_2, \ldots, t_l\}$

Et on cherche à maximiser le flot total.

Équivalence avec le problème initial

On peut se ramener au problème initial (*i.e.* une seule source/ un seul puits). En modifiant le réseau :

- En ajoutant une super source s qui est reliée à toutes les autres : on ajoute les arcs (s, s_i) pour tout i entre 1 et k.

Multi sources/ Multi puits

On peut considérer un problème plus général où on a :

- Plusieurs sources $S = \{s_1, s_2, \dots, s_k\}$
- Plusieurs puits $T = \{t_1, t_2, \dots, t_l\}$

Et on cherche à maximiser le flot total.

Équivalence avec le problème initial

On peut se ramener au problème initial (*i.e.* une seule source/ un seul puits). En modifiant le réseau :

- En ajoutant une super source s qui est reliée à toutes les autres : on ajoute les arcs (s, s_i) pour tout i entre 1 et k.
- En ajoutant un super puits t qui est relié à tous les puits : on ajoute (t_j, t) pour tout j entre 1 et l.

Multi sources/ Multi puits

On peut considérer un problème plus général où on a :

- Plusieurs sources $S = \{s_1, s_2, \ldots, s_k\}$
- Plusieurs puits $T = \{t_1, t_2, \ldots, t_l\}$

Et on cherche à maximiser le flot total.

Équivalence avec le problème initial

On peut se ramener au problème initial (*i.e.* une seule source/ un seul puits). En modifiant le réseau :

- En ajoutant une super source s qui est reliée à toutes les autres : on ajoute les arcs (s, s_i) pour tout i entre 1 et k.
- En ajoutant un super puits t qui est relié à tous les puits : on ajoute (t_j, t) pour tout j entre 1 et l.
- La capacité des arcs ajoutés est infinie (ce n'est pas le facteur limitant).

Multi sources/ Multi puits

On peut considérer un problème plus général où on a :

- Plusieurs sources $S = \{s_1, s_2, \ldots, s_k\}$
- Plusieurs puits $T = \{t_1, t_2, \ldots, t_l\}$

Et on cherche à maximiser le flot total.

Équivalence avec le problème initial

On peut se ramener au problème initial (*i.e.* une seule source/ un seul puits). En modifiant le réseau :

- En ajoutant une super source s qui est reliée à toutes les autres : on ajoute les arcs (s, s_i) pour tout i entre 1 et k.
- En ajoutant un super puits t qui est relié à tous les puits : on ajoute (t_j, t) pour tout j entre 1 et l.
- La capacité des arcs ajoutés est infinie (ce n'est pas le facteur limitant).
- On résout le problème entre s et t.

Chemin augmentant/améliorant

Pour augmenter la valeur d'un flot, on doit augmenter la quantité de matière qui transite sur certains arcs.
Comme on doit assurer la conservation du flot tout du long. On ne peut pas modifier des arcs individuellement.
On cherche donc un chemin P de G que l'on peut améliorer.

Definition (Chemin augmentant)

Un chemin P de G est améliorant si pour tout arc (u, v) de P on a $f(u, v) < c(u, v)$.

Chemin augmentant/améliorant

Pour augmenter la valeur d'un flot, on doit augmenter la quantité de matière qui transite sur certains arcs.
Comme on doit assurer la conservation du flot tout du long. On ne peut pas modifier des arcs individuellement.
On cherche donc un chemin P de G que l'on peut améliorer.

Definition (Chemin augmentant)

Un chemin P de G est améliorant si pour tout arc (u, v) de P on a $f(u, v) < c(u, v)$.

Valeur de l'augmentation

Soit P un chemin augmentant. La quantité de matière supplémentaire à faire transiter par P est obtenue en trouvant la valeur $\min\{c(u, v) - f(u, v) | (u, v) \in E(P)\}$.

Réseau résiduel

Commenter trouver un chemin augmentant ?

Graphe auxiliaire

On va construire un graphe auxiliaire G_f dans lequel un chemin de s à t dans G_f correspond à un chemin augmentant dans G.

Réseau résiduel

Commenter trouver un chemin augmentant ?

Graphe auxiliaire

On va construire un graphe auxiliaire G_f dans lequel un chemin de s à t dans G_f correspond à un chemin augmentant dans G.

Graphe auxiliaire

On note G_f le graphe auxiliaire. Il dépend du flot qui transite sur G.
$G_f = (V, E_f)$; V est le même ensemble de sommets et
$E_f = \{(u, v) \in V \times V | c_f(u, v) > 0\}$

Réseau résiduel

Commenter trouver un chemin augmentant ?

Graphe auxiliaire

On va construire un graphe auxiliaire G_f dans lequel un chemin de s à t dans G_f correspond à un chemin augmentant dans G.

Graphe auxiliaire

On note G_f le graphe auxiliaire. Il dépend du flot qui transite sur G.
$G_f = (V, E_f)$; V est le même ensemble de sommets et
$E_f = \{(u, v) \in V \times V | c_f(u, v) > 0\}$

Capacité résiduelle

Pour chaque arc de G on évalue la quantité de flot qu'on peut ajouter.

$$c_f(u, v) = c(u, v) - f(u, v)$$

Réseau résiduel (suite / 1)

Remarque

- Pour les arcs de G, la notion est claire.

Réseau résiduel (suite / 1)

Remarque

- Pour les arcs de G, la notion est claire.
- Par contre si on a un flot qui transite sur (u, v) (*i.e.* $f(u, v) \neq 0$)
  on a $f(v, u) = -f(u, v)$

Réseau résiduel (suite / 1)

Remarque

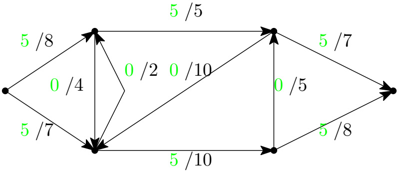
- Pour les arcs de G, la notion est claire.
- Par contre si on a un flot qui transite sur (u, v) (*i.e.* $f(u, v) \neq 0$)
  on a $f(v, u) = -f(u, v)$
- On a dit que pour un non-arc (a, b) sa capacité est 0.

Réseau résiduel (suite / 1)

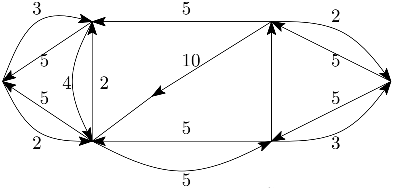
Remarque

- Pour les arcs de G, la notion est claire.
- Par contre si on a un flot qui transite sur (u, v) (*i.e.* $f(u, v) \neq 0$)
  on a $f(v, u) = -f(u, v)$
- On a dit que pour un non-arc (a, b) sa capacité est 0.
- Si un flot de valeur k transite sur (b, a), $c_f(a, b) = 0 - (-k)$ donc $c_f(a, b) = k$.

Réseau résiduel (suite / 2)

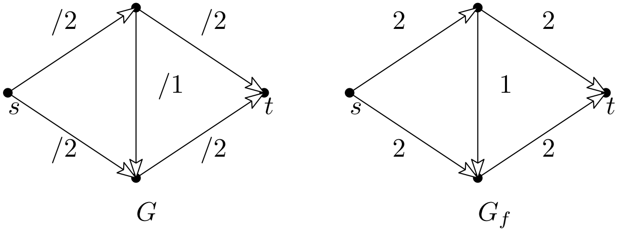


Réseau G

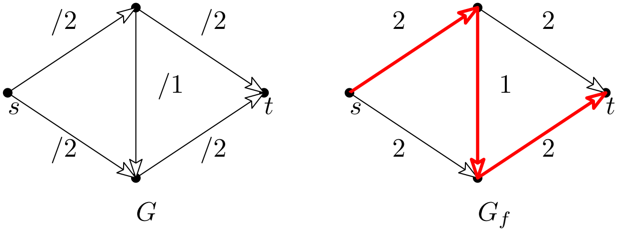


Réseau Résiduel G_f

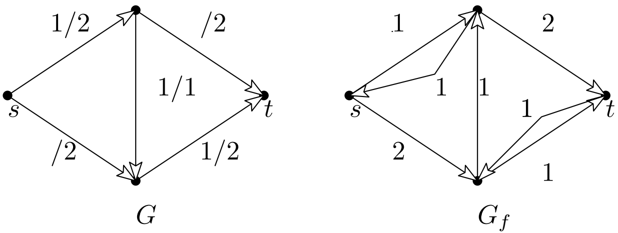
Réseau résiduel (suite / 3)



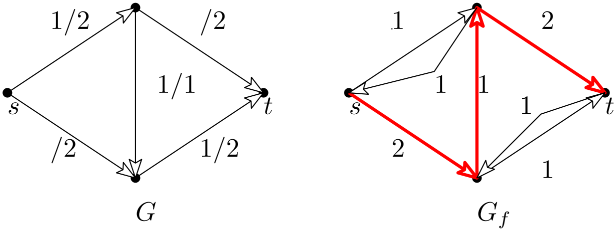
Réseau résiduel (suite / 3)



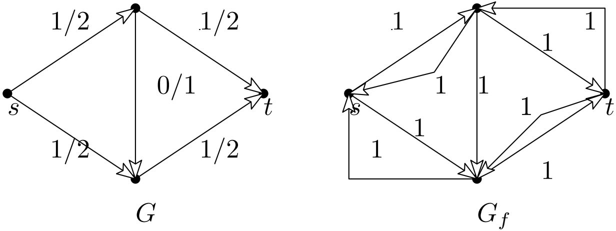
Réseau résiduel (suite / 3)



Réseau résiduel (suite / 3)



Réseau résiduel (suite / 3)



Coupe Minimum

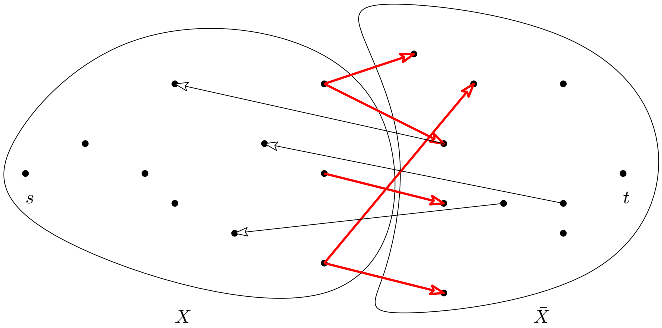
Definition (Coupe)

Soit X sous-ensemble de sommet $s \in X$ et $t \in \bar{X}$.
La coupe associée à X est l'ensemble des arcs qui commence dans X et qui arrivent dans $\bar{X}$

Coupe Minimum

Definition (Coupe)

Soit X sous-ensemble de sommet $s \in X$ et $t \in \bar{X}$.
La coupe associée à X est l'ensemble des arcs qui commence dans X et qui arrivent dans $\bar{X}$



Coupe Minimum (suite)

Valeur de coupe

La valeur (ou sa capacité) d'une coupe $X, \bar{X}$ est définie comme la somme des capacité des arcs de la coupe.
La valeur est notée :

$$c(X, \bar{X})$$

Coupe Minimum (suite)

Valeur de coupe

La valeur (ou sa capacité) d'une coupe $X, \bar{X}$ est définie comme la somme des capacité des arcs de la coupe.
La valeur est notée :

$$c(X, \bar{X})$$

Coupe Minimum

La coupe minimum est la coupe de l'ensemble $X, \bar{X}$ qui minimise la valeur de la coupe.

Coupe Minimum/Flot Maximum

Remarque

Le flot qui transite sur une coupe est toujours inférieur à la valeur de la coupe.
Pour toute coupe $X, \bar{X}$ on a

$$|f| \leq c(X, \bar{X})$$

Coupe Minimum/Flot Maximum

Remarque

Le flot qui transite sur une coupe est toujours inférieur à la valeur de la coupe.
Pour toute coupe $X, \bar{X}$ on a

$$|f| \leq c(X, \bar{X})$$

Theorem

Soit f un flot qui transite sur le réseau G de sources s et de puits t. Les conditions suivantes sont équivalentes :

1. *f est un flot maximum de G.*
2. *Le graphe résiduel G_f ne contient aucun chemin de s à t.*
3. *Il existe une coupe $X, \bar{X}$ telle que la valeur du flot $|f|$ est égale à $c(X, \bar{X})$.*

Coupe Minimum/Flot Maximum (suite)

Preuve

1. $\Rightarrow$ 2. Par l'absurde : On suppose f maximum et G_f admet un chemin de s à t.
Soit P ce chemin. Dans G on peut augmenter la valeur du flot f en augmentant les valeurs de flots sur P.
On obtient f' et $|f'| > |f|$ donc f n'est pas maximum : contradiction

Coupe Minimum/Flot Maximum (suite)

Preuve

1. $\Rightarrow$ 2. Par l'absurde : On suppose f maximum et G_f admet un chemin de s à t.
Soit P ce chemin. Dans G on peut augmenter la valeur du flot f en augmentant les valeurs de flots sur P.
On obtient f' et $|f'| > |f|$ donc f n'est pas maximum : contradiction

2. $\Rightarrow$ 3. Considérons que G_f ne contienne pas de chemin de s à t (donc pas de chemin améliorant dans G). Soit X l'ensemble des sommets de G_f (et de G) atteignable à partir s dans G_f.
Par construction, tout arc (u, v) avec $u \in X$ et $v \in \bar{X}$ on a $f(u, v) = c(u, v)$ (Si tel n'était pas le cas, on aurait un arc (u, v) dans G_f et donc v serait dans X et non dans $\bar{X}$).

Coupe Minimum/Flot Maximum (suite)

3. $\Rightarrow$ 1. Si la valeur du flot correspond à la capacité d'un coupe $X, \bar{X}$ ($|f| = c(X, \bar{X})$)
alors on ne peut pas augmenter la valeur du flot sans violer la condition $f(u, v) \leq c(u, v)$ qui doit être vérifiée pour tout arc (u, v). Donc f est maximum. $\square$

Algorithme de Ford-Fulkerson

Algorithme 1 : Algorithme de flots maximum (Ford&Fulkerson)

Input : $G = (V, A, c)$ un réseau
Output : f un flot maximum de G

$f := \varnothing$
while *Il existe* P *un chemin dans* G_f **do**
  Augmenter la valeur de f à $|f| + c_f(P)$ le long de P
  Calculer G_f
return f

Algorithme de Ford-Fulkerson (suite)

Validité de l'algorithme

L'algorithme de Ford-Fulkerson calcule bien un flot maximum. Car s'arrête une fois qu'il n'y a plus de chemin améliorant dans G. D'après le théorème précédent, le flot obtenu est maximum.

Algorithme de Ford-Fulkerson (suite)

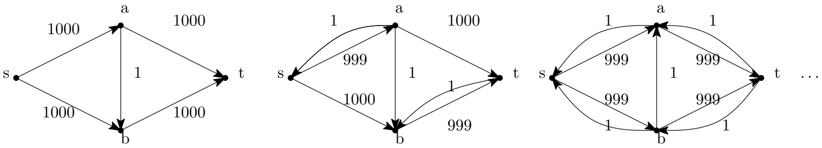
Validité de l'algorithme

L'algorithme de Ford-Fulkerson calcule bien un flot maximum. Car s'arrête une fois qu'il n'y a plus de chemin améliorant dans G. D'après le théorème précédent, le flot obtenu est maximum.

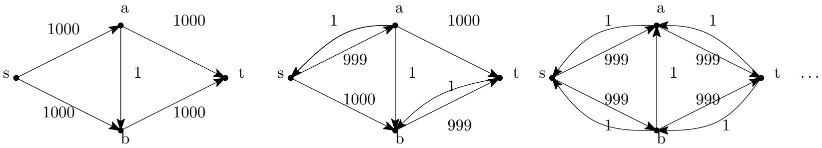
Complexité

L'algorithme n'est pas polynomial en l'état.

Exemple critique de FF

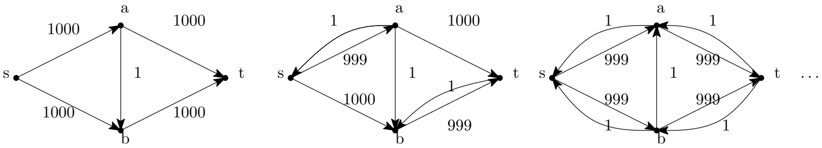


Exemple critique de FF



Sur l'exemple précédent on peut faire 2000 itérations, en utilisant à chaque fois l'arête du milieu.

Exemple critique de FF



Sur l'exemple précédent on peut faire 2000 itérations, en utilisant à chaque fois l'arête du milieu.

Si les capacités sont flottantes, l'algorithme de FF peut donner des résultats incorrects

Algorithme d'Edmonds-Karp

Amélioration polynomiale

Jack Edmonds et Richard Karp ont montré que si on prend à chaque fois un plus court chemin dans le graphe résiduel l'algorithme devient polynomial.

On peut trouver un plus court chemin en faisant un parcours en largeur.

L'algorithme fonctionne correctement même avec des capacités flottantes.

Remarque

Capacité entières

Si les capacité de chacun des arcs est un nombre entier. Le flot maximum est aussi un entier. Et il existe un flot maximum tel que sur chaque arc transitent des valeurs entières.