R5.A.12/R5.B.10 Modélisations mathématiques

Thibault Godin, Lucie Naert

Séquence 2 : Flots et Affectations

IUT de Vannes 11 juillet 2024

Plan

Flots

Couplage

Réseau et capacité

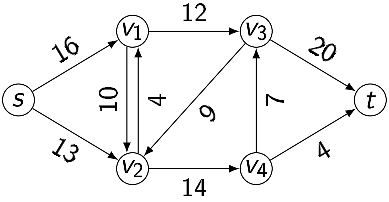
Un graphe de capacité (ou "réseau de transport") $G = (S, A)$ est un graphe **orienté** tel que :

- $\forall (u, v) \in A$, capacité $c(u, v) > 0$.
- Si $(u, v) \notin A$, on pose $c(u, v) = 0$
- présence de deux sommets particuliers :
  - s : "source" (pas d'arc entrants)
  - t : "puits" (pas d'arc sortants)

Réseau et capacité

Un graphe de capacité (ou "réseau de transport") $G = (S, A)$ est un graphe **orienté** tel que :

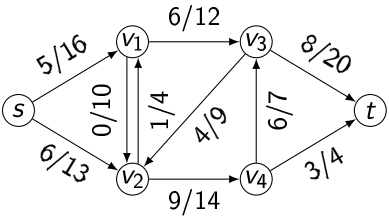
- $\forall (u, v) \in A$, capacité $c(u, v) > 0$.
- Si $(u, v) \notin A$, on pose $c(u, v) = 0$
- présence de deux sommets particuliers :
  - s : "source" (pas d'arc entrants)
  - t : "puits" (pas d'arc sortants)



Flot

Un *flot* est une fonction $f : S^2 \to \mathbb{R}$ telle que :

- **Contraintes de capacité** : $f(u,v) \leq c(u,v)$
- **Anti-symétrie** $f(u,v) = -f(v,u)$
- **Conservation du flot** $\sum_{w \in S} f(u,w) = 0$, sauf si $u = s$ ou $u = t$



La *valeur* d'un flot est $\sum_{(s,u)\in A} f(s,u) = \sum_{(v,t)\in A} f(v,t)$

Flot maximum

Un problème classique est la recherche d'un flot maximum à partir d'un graphe de capacité.

Par exemple, si l'on considère que notre graphe de capacité représente un réseau de communication dont la capacité représente le débit. On voudrait connaitre le débit maximum pour envoyer des fichiers entre un émetteur (la source) et un récepteur (le puits).

Plan

Algorithme d'Edmonds-Karp

Il s'agit d'une variante d'un autre algorithme, l'algorithme de Ford-Fulkerson.

Entrées : Un graphe $G = (S, A)$ avec une capacité c, une source s, et un puits t

Sortie : Flot f de s à t de valeur maximum

Principe

Initialisation : le flot est mis à 0 pour chaque arc.

Tant qu'il existe un **plus court chemin augmentant** p allant de s à t dans le **graphe résiduel** :

1. Chercher la capacité $c_f(p)$ de p : minimum des capacités des arêtes formant p
2. Pour chaque arête $(u, v) \in p$:
   a. $f(u, v) \leftarrow f(u, v) + c_f(p)$
   b. $f(v, u) \leftarrow f(v, u) - c_f(p)$
3. Calcul du graphe résiduel et du plus court chemin augmentant.

Plus court chemin augmentant

Le plus court chemin augmentant est le plus court chemin (en nombre d'arcs) pour aller d'un point a à un point b (ici, pour aller de la source au puits).

Ce chemin est obtenu grâce à un parcours en largeur (BFS, ou *Breadth-First Search*).

Graphe résiduel

Un graphe résiduel doit être calculé à chaque itération. Il s'agit du graphe de capacité auquel on a soustrait le graphe de flot.

Ainsi, si, sur un arc, le flot est égal à la capacité (impossible de rajouter du flot), l'arc correspondant sur le graphe résiduel aura une capacité de 0 et ne pourra donc plus être emprunté lors du parcours en largeur.

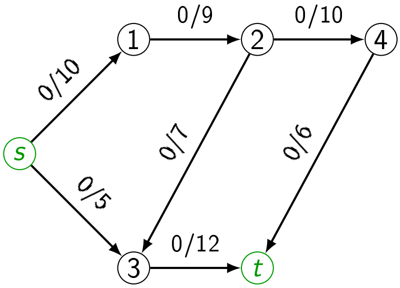
Plan

Flots

Application sur un exemple

Soit le graphe de capacité suivant. On cherche à trouver le flot maximum sur ce graphe.

Au départ, le flot sur chaque arc est à 0.

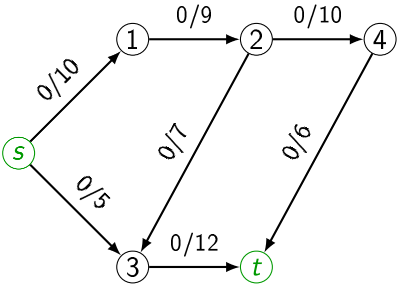


Graphe de capacité et de flot

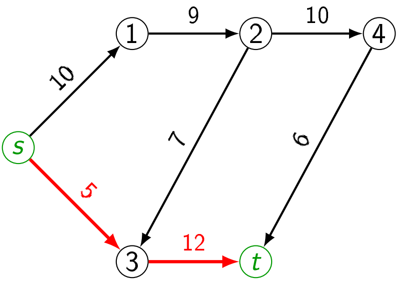
Application sur un exemple

Soit le graphe de capacité suivant. On cherche à trouver le flot maximum sur ce graphe.

Au départ, le flot sur chaque arc est à 0.



Graphe de capacité et de flot

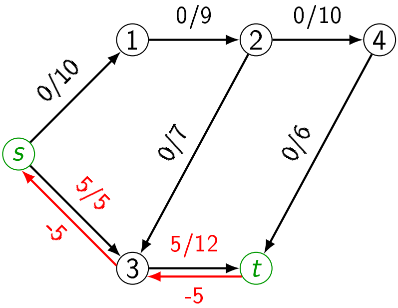


Graphe résiduel (capacité - flot) n°0.

Plus court chemin p_0 en rouge

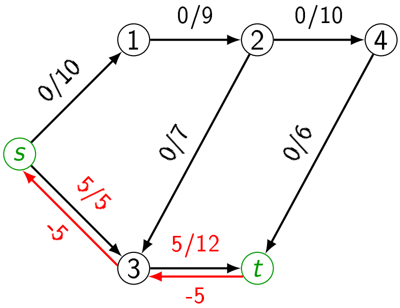
Capacité du chemin $c(p_0) = 5$

Application sur un exemple

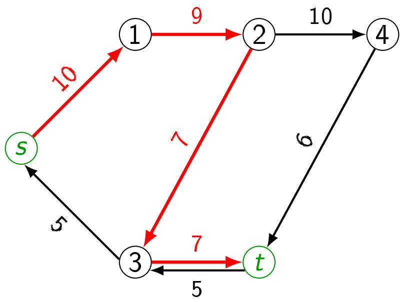


Ajout de $c(p_0)$ dans le sens des flèches et retrait dans l'autre sens.

Application sur un exemple



Ajout de $c(p_0)$ dans le sens des flèches et retrait dans l'autre sens.

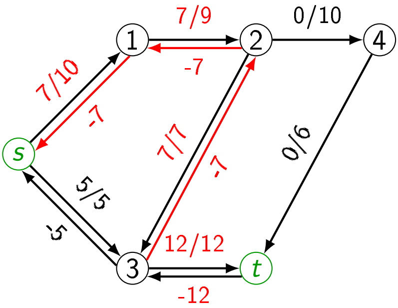


Graphe résiduel n°1.

Plus court chemin p_1 en rouge

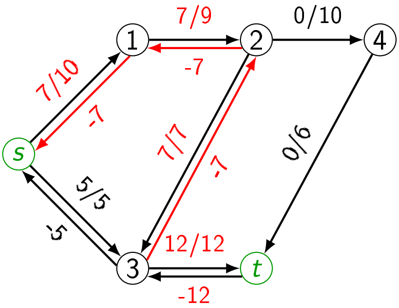
Capacité du chemin $c(p_1) = 7$

Application sur un exemple

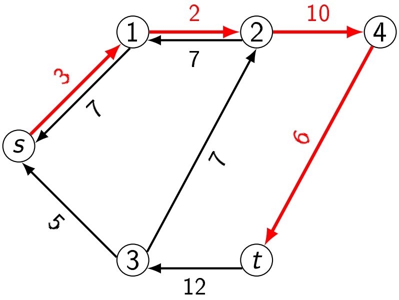


Ajout de $c(p_1)$ dans le sens des flèches et retrait dans l'autre sens.

Application sur un exemple



Ajout de $c(p_1)$ dans le sens des flèches et retrait dans l'autre sens.

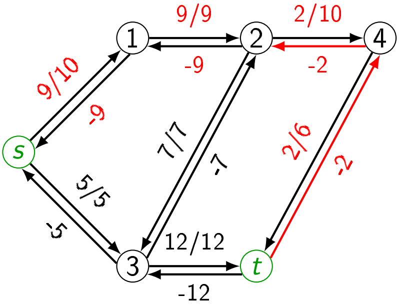


Graphe résiduel n°2.

Plus court chemin p_2 en rouge

Capacité du chemin $c(p_2) = 2$

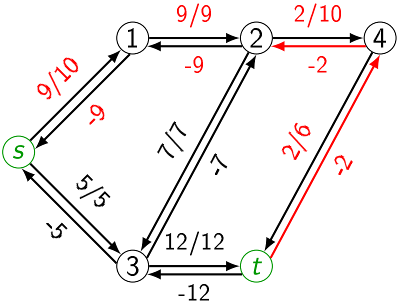
Application sur un exemple



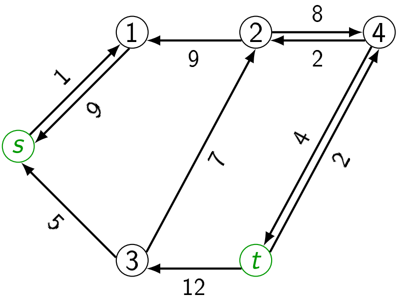
Ajout de $c(p_2)$ dans le sens des flèches et

retrait dans l'autre sens.

Application sur un exemple



Ajout de $c(p_2)$ dans le sens des flèches et retrait dans l'autre sens.



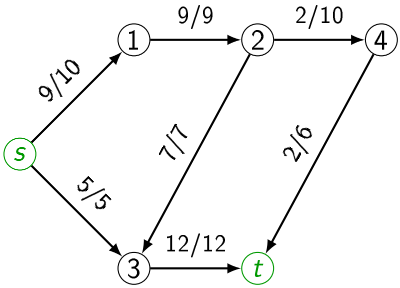
Graphe résiduel n°3.

Pas de plus court chemin

Fin de l'algorithme

Application sur un exemple

Résultats : Graphe de flot (valeurs positives) et valeur du flot total.



$$\text{Flot total} = c(p_0) + c(p_1) + c(p_2) = 14$$

Plan

Flots

Définitions

Algorithme d'Edmonds-Karp

Exemple

Complexité

Couplage

Définitions

Couplage biparti maximal

Motivation

Couplage biparti maximal

Complexité

L'algorithme d'Edmonds-Karp a une complexité en $O(|S||A|^2)$

Plan

Plan

Flots

Définitions

Algorithme d'Edmonds-Karp

Exemple

Complexité

Couplage

Définitions

Couplage biparti maximal

Motivation

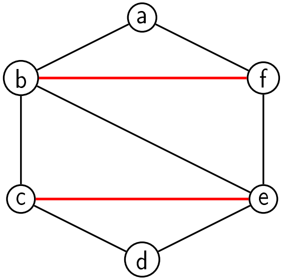
Couplage biparti maximal

Couplage

Un couplage (ou appariement, matching) d'un graphe est un ensemble d'arêtes de ce graphe qui n'ont pas de sommets en commun

Couplage

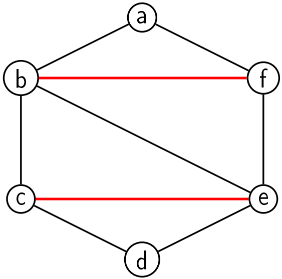
Un couplage (ou appariement, matching) d'un graphe est un ensemble d'arêtes de ce graphe qui n'ont pas de sommets en commun



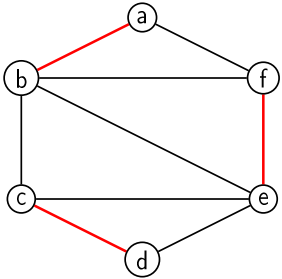
Couplage **maximal** : toute arête du graphe possède au moins une extrémité commune avec une arête du couplage

Couplage

Un couplage (ou appariement, matching) d'un graphe est un ensemble d'arêtes de ce graphe qui n'ont pas de sommets en commun

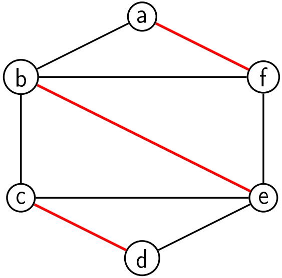


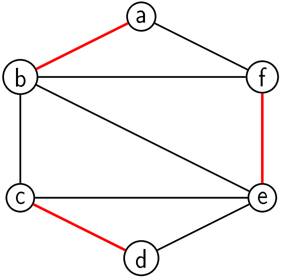
Couplage **maximal** : toute arête du graphe possède au moins une extrémité commune avec une arête du couplage



Couplage **maximum** : couplage contenant le plus grand nombre possible d'arêtes

Couplage





Couplage **parfait** : tout sommet du graphe appartient à exactement une arête du couplage

Plan

Organisation d'emploi du temps

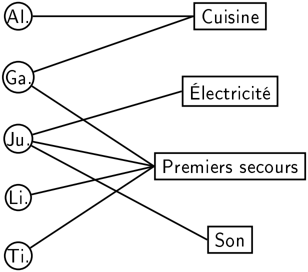
On cherche des bénévoles pour un festival. Les bénévoles donnent leurs compétences sur les tâches :

- Alaric : cuisine
- Jules : électricité, premier secours et son
- Gaspard : cuisine et premiers secours
- Lily : premiers secours
- Timothée : premiers secours
  bricolage, électricité et cuisine

Organisation d'emploi du temps

On cherche des bénévoles pour un festival. Les bénévoles donnent leurs compétences sur les tâches :

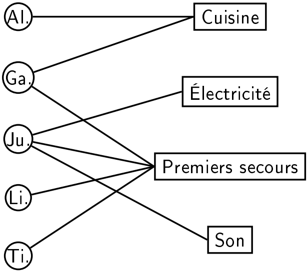
- Alaric : cuisine
- Jules : électricité, premier secours et son
- Gaspard : cuisine et premiers secours
- Lily : premiers secours
- Timothée : premiers secours
  bricolage, électricité et cuisine



Organisation d'emploi du temps

On cherche des bénévoles pour un festival. Les bénévoles donnent leurs compétences sur les tâches :

- Alaric : cuisine
- Jules : électricité, premier secours et son
- Gaspard : cuisine et premiers secours
- Lily : premiers secours
- Timothée : premiers secours
  bricolage, électricité et cuisine



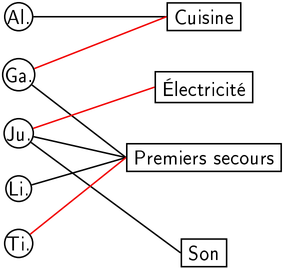
Chaque bénévole ne peut tenir qu'un poste (qui n'a besoin que d'une personne), comment optimiser la répartition ?

Couplage biparti maximal

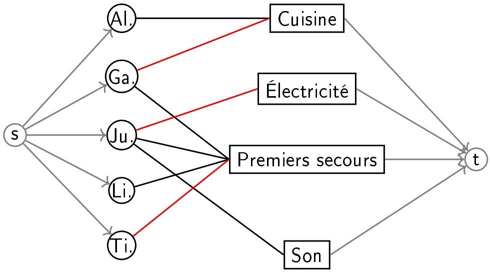
Un graphe biparti est un graphe $G = (S, A)$ où l'ensemble des sommets S peut être partitionné en deux ensemble S_1, S_2 tels que :

1. $S_1 \cup S_2 = S$
2. $S_1 \cap S_2 = \emptyset$
3. Il n'y a aucune arête entre deux sommets de S_1 ni entre deux sommets de S_2

Le problème des affectations peut-être représenté par un graphe biparti :



Le problème des affectations peut-être représenté par un graphe biparti :



Qui peut lui même se changer en problème de flot maximum en rajoutant :

- une source qui mène à chacun des sommets de S_1 et un puits accessible depuis chacun des sommets de S_2
- des capacités de 1 sur chaque arête.

On trouve le couplage max en calculant le flot maximum.

Plan

Pour aller plus loin...

La semaine prochaine, nous passerons à un thème différent mais si vous êtes intéressés par les problèmes de couplage, en voici d'autres...

- Problème des colocataires : mariages entre membres d'une même population
- Recherche d'un couplage parfait grâce à l'algorithme hongrois