Flots et Couplages

1. Flots et Coupes
2. Ford-Fulkerson, Dinic, Edmonds-Karp
3. Applications des flots
4. Couplages
5. Mariages Stables

Définitions utiles

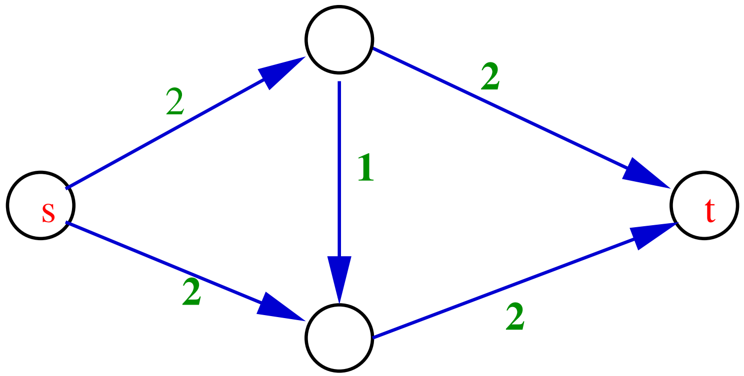
Réseau de transport: Graphe orienté $G = (X, A)$, deux sommets particuliers s, et t. s n'a pas d'arc entrant et t n'a pas d'arc sortant.

Pour chaque arc a, un entier $c(a)$ appelé sa *capacité*.

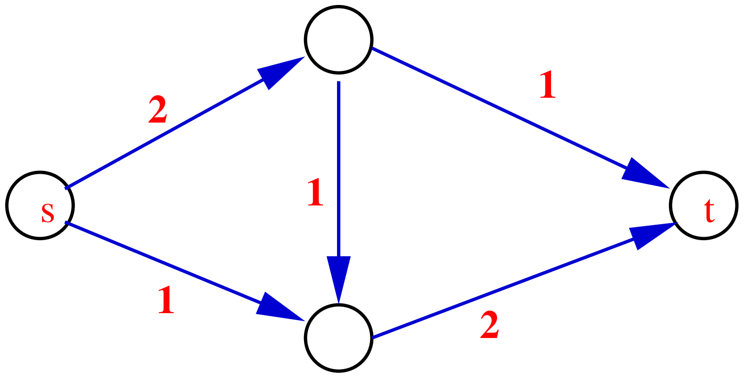
Un flot est une application ϕ de A dans $\mathcal{N}$ telle que :

1. Pour chaque arc a $\phi(a) \leq c(a)$
2. Pour chaque sommet autre que s et t la somme des flots entrants est égale à celle des flots sortants
3. La valeur du flot est égale à la somme des flots sortants de s

Problème : trouver un flot de valeur maximale.



Un reseau et ses capacite's



Un flot sur le graphe

Un flot saturé non maximal

Un flot saturé est un flot tel que sur tout chemin de s à t il existe un arc ayant un flot égal à sa capacité.

Coupes

Pour réseau de transport $G = (X, A, s, t)$, une *coupe* est donnée par une partition de X en deux sous-ensembles disjoints $X = Y \cup Z$ tels que $s \in Y$ et $t \in Z$. La capacité de la coupe est égale à la somme des capacités des arcs qui ont une origine dans Y et une extrémité dans Z.

Pour un flot ϕ et une coupe Y, Z on note :

$$\phi(Y, Z) = \sum_{or(u) \in Y, ext(u) \in Z} \phi(u)$$

De plus on introduit

$$\Delta(Y, Z) = \phi(Y, Z) - \phi(Z, Y)$$

Avec cette notation la valeur du flot à maximiser est : $V_\phi = \phi(\{s\}, X \setminus \{s\})$

Lemme 1 Pour toute coupe (Y, Z) et tout flot ϕ on a : $\Delta(Y, Z) = V_\phi$.

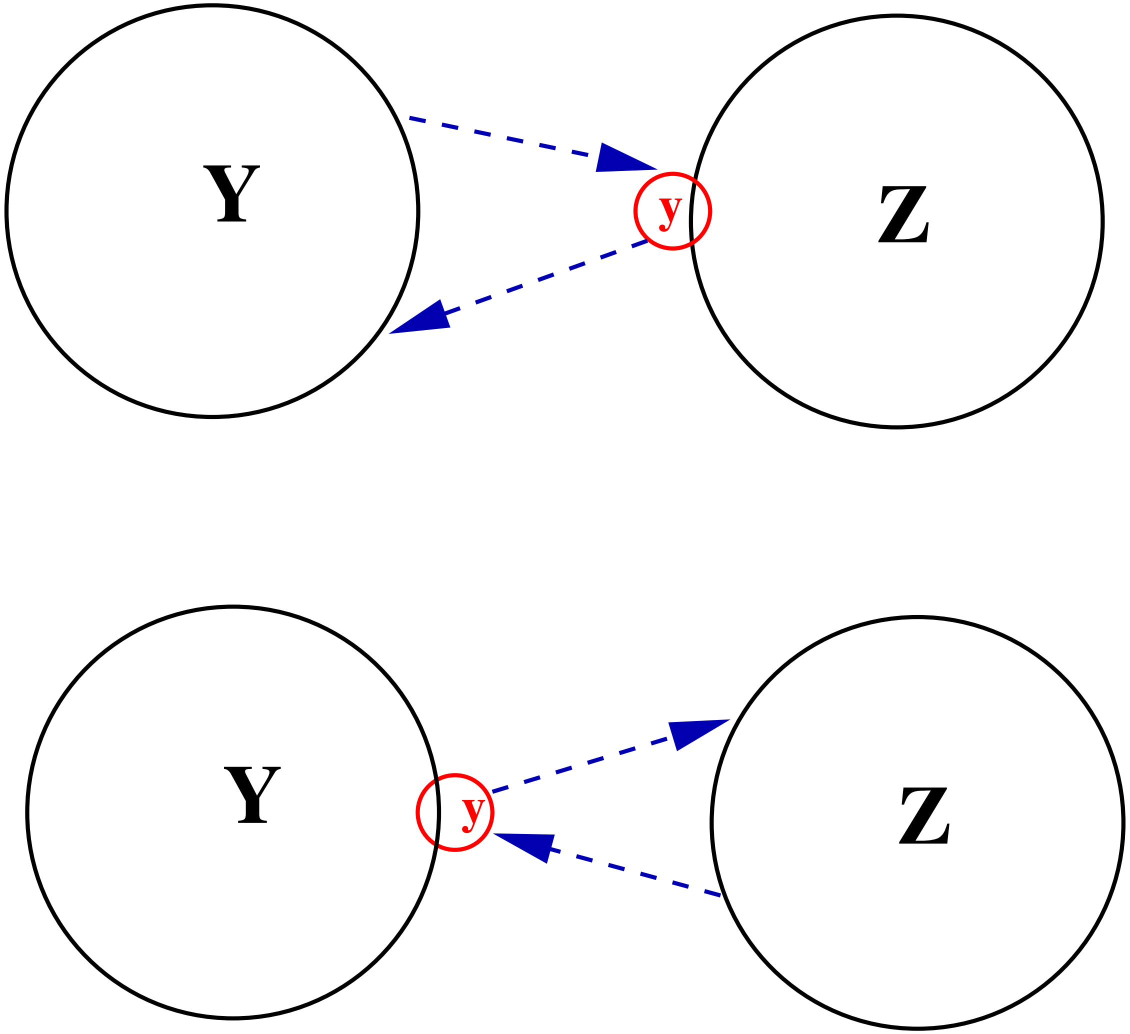
- Preuve par récurrence sur $|Y|$
- C'est immédiat si $|Y| = 1$
- Lorsqu'on augmente Y du sommet y on obtient :

$$\phi(Y \cup y, Z \setminus y) \;=\; \phi(Y, Z) + \phi(y, Z) - \phi(Y, y)$$

$$\phi(Z \setminus y, Y \cup y) \;=\; \phi(Z, Y) + \phi(Z, y) - \phi(y, Y)$$

Ce qui donne :

$$\Delta(Z \setminus y, Y \cup y) \;=\; \Delta(Z, Y) + \phi(y, Z) + \phi(y, Y) - \phi(Y, y) - \phi(Z, y) = \Delta(Y, Z)$$



Conservation du flot

Lemme 2 Pour toute coupe (Y, Z) et tout flot ϕ on a : $V_\phi \leq C(Y, Z)$

Preuve

- Par le Lemme 1 on a :

$$V_\phi \;=\; \Delta(Y, Z) \;=\; \phi(Y, Z) - \phi(Z, Y)$$

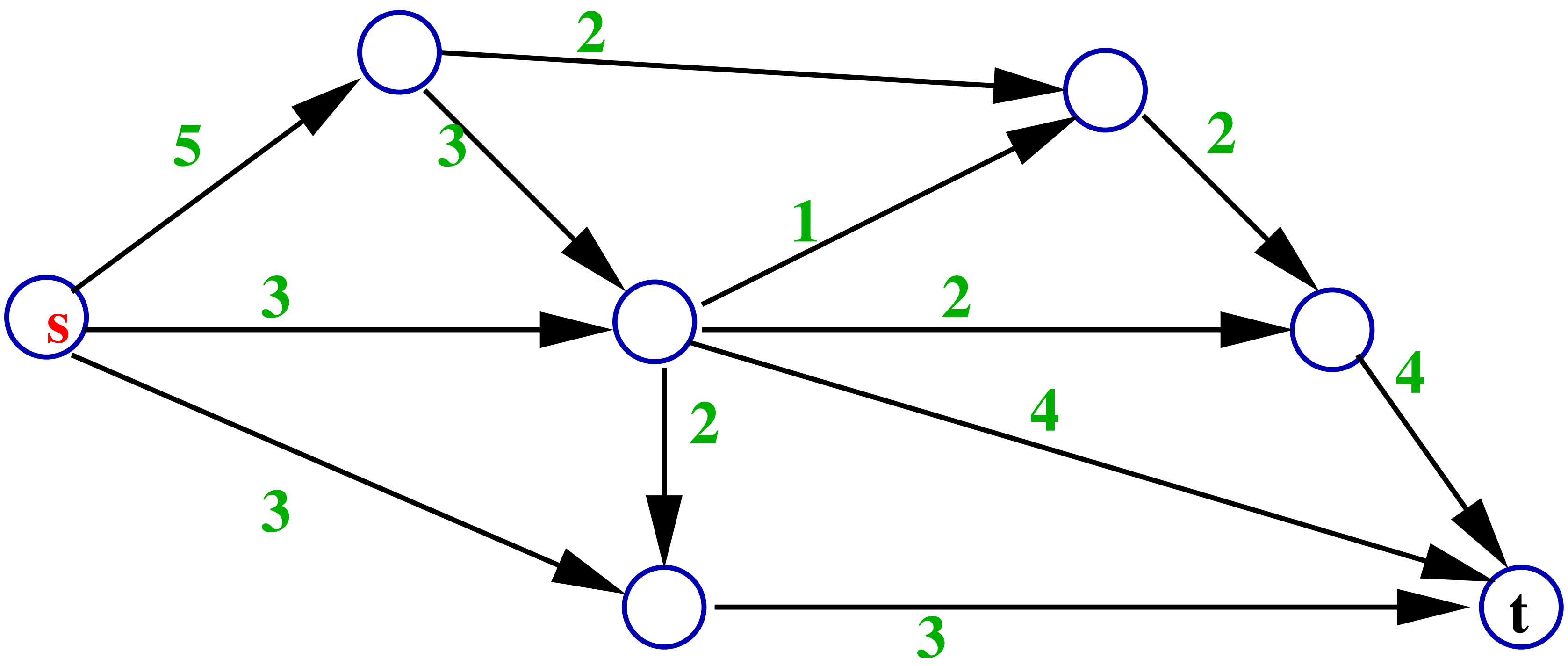
- Soit :

$$V_\phi \;\leq\; \phi(Y, Z) \;\leq\; c(Y, Z)$$

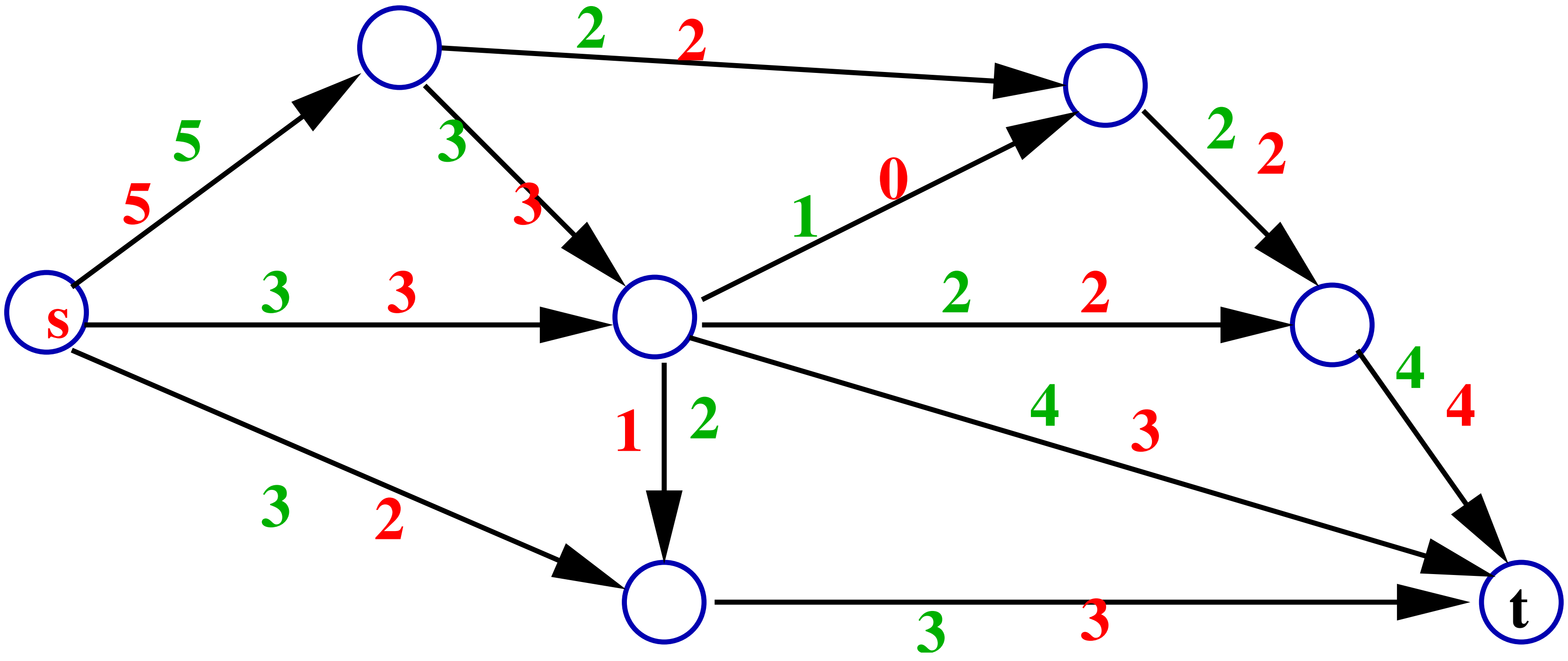
Graphe des augmentations

On associe à un réseau de transport $G = (X, s, t, A, c)$ et à un flot ϕ sur G un graphe G_ϕ donné par :

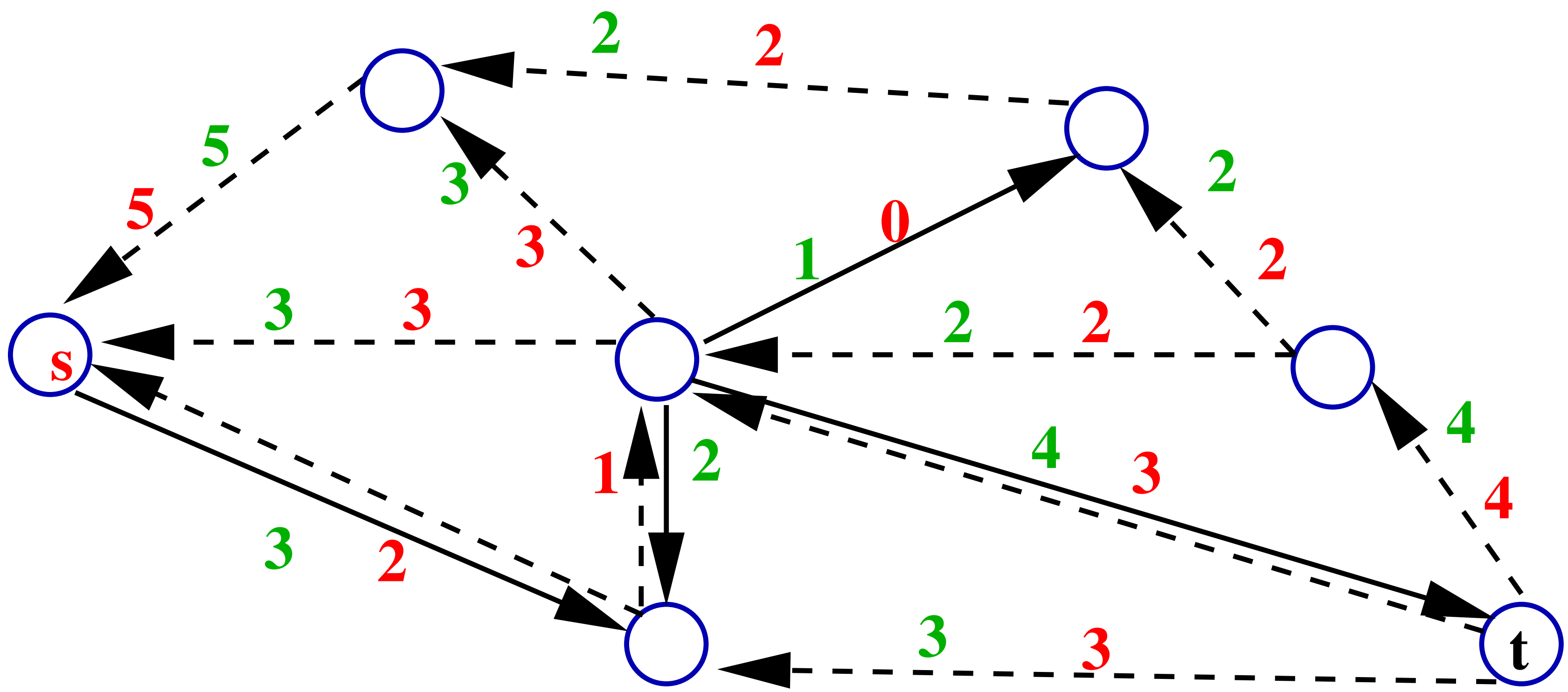
- Les sommets de G_ϕ sont ceux de G
- Pour tout arc $a = (x, y)$ de G on construit un arc entre x et y dans G_ϕ si $\phi(a) < c(a)$
- Pour tout arc $a = (x, y)$ de G on construit un arc entre y et x dans G_ϕ si $\phi(a) > 0$



Un graphe et ses capacités



Un flot sur ce graphe



Le graphe des augmentations

Lemme 3 S'il existe un chemin joignant s à t dans G_ϕ le flot peut être amélioré:

Preuve

- déterminer le minimum de l'ensemble des $c(u) - \phi(u)$ (arcs dans le même sens que dans G) et des $\phi(u)$ (arcs en sens contraire) sur le chemin
- Il suffit d'ajouter un flot égal à ce minimum sur les arcs dans le bon sens et de retrancher cette valeur sur les arcs en sens opposé

Lemme 4 S'il n'existe pas de chemin joignant s à t dans G_ϕ alors il existe une coupe (Y, Z) telle que :

$$c(Y, Z) \; = \; V_\phi$$

Preuve:

- Y est formé de tous les sommets atteignables à partir de s dans le graphe des augmentations
- On vérifie que la capacité de la coupe est égale au flot V_ϕ

Théorème de Ford-Fulkerson

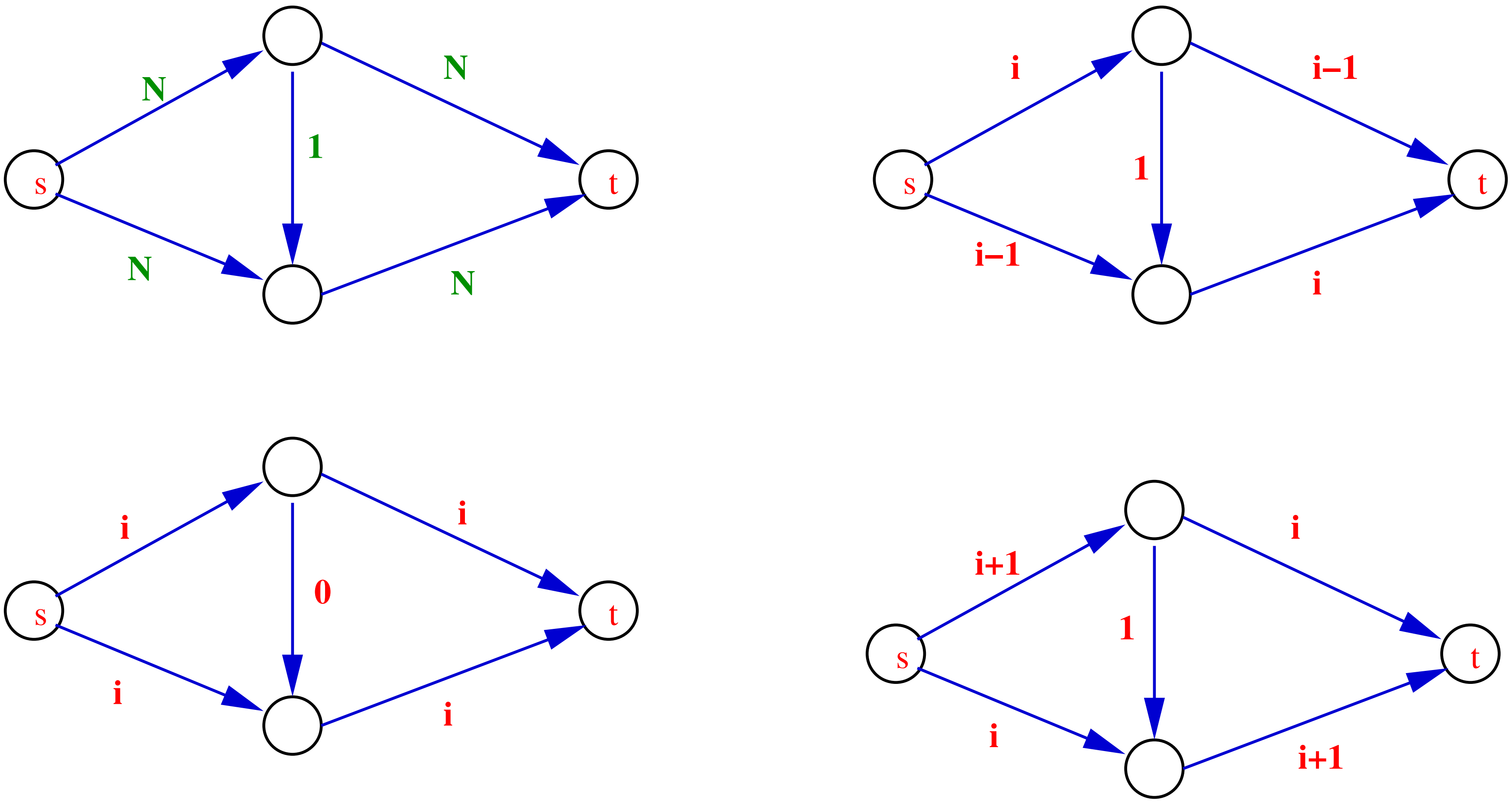
Le flot maximal est égal à la capacité de la coupe minimale:

- Le lemme 2 montre que tout flot est inférieur ou égal à la capacité de toute coupe
- Les lemmes 3 et 4 montrent qu'il existe un flot égal à une coupe; c'est donc le flot maximal et la coupe minimale

Méthode de Ford Fulkerson

On calcule un flot quelconque (par exemple nul)

- Tant qu'il exite un chemin dans le graphe des augmentations on augmente le flot
- On obtient ainsi le flot maximal
- La coupe minimale est obtenue considérant l'ensemble des sommets atteignables à partir de s dans le graphe des augmentations



Un calcul très lent

Algorithme de Dinic, Edmonds et Karp

On calcule un flot saturé

- Tant qu'il exite un chemin dans le graphe des augmentations on augmente le flot
- On choisit le premier chemin obtenu par recherche en largeur
- C'est ainsi le plus court chemin

Analyse de l'algorithme de Dinic

- La longueur du chemin dans le graphe des améliorations est faiblement croissante au cours de l'algorithme
- Le coût de l'alogorithme est en $O(nm^2)$ opérations.

Exemples d'application des algorithmes de flots

1. Construction de chemins disjoints (voir exercices)
2. Optimisation de la distribution de denrées
3. Détermination du profit maximal pour certains problèmes d'ordonnacement
4. Coloration de cartes

Distribution de denrées

- Une denrée est disponible en des centres de distribution $A_1, A_2, \ldots, A_p$, la quantité disponible en A_i est notée a_i
- Des demandes sont effectuées par des clients $B_1, B_2, \ldots, B_q$ la quantité demandée par B_i est notée b_i.
- Le client B_i ne peut être livré que par un sous-ensemble de l'ensemble des centres de distributions.
- Il s'agit de déterminer l'approvisionnement maximal de l'ensemble des clients.

Solution

Construire un graphe ayant pour sommets les A_i les B_j et deux nouveaux sommets s et t

Relier s à chacun des sommets A_i par un arc de capacité a_i.

Relier chacun des sommets B_j à t par un arc de capacité b_j.

Relier A_i à B_j chaque fois que le centre de dsitribution A_i est à même de livrer de la marchandise à B_j, la capacité de cet arc est aussi grande que l'on veut (choisir par exemple $max(a_i, b_j)$).

Chercher le flot maximal dans ce graphe

Problème de choix d'objets à emporter

- On peut emmener dans une navette spatiale un certain nombre d'instruments $A_1, A_2, \ldots, A_p$, le coût du transport de A_i est c_i
- Des commandes de réalisations d'expériences dans l'espace sont faites on les dénote par $B_1, B_2, \ldots, B_q$; le gain réalisé si l'on effectue l'expérience B_j est noté b_j
- La réalisation de B_i nécessite d'avoir emporté un sous-ensemble de l'ensemble des instruments
- Il s'agit de déterminer quels sont les instruments à emporter et quelles sont les expériences à réaliser afin de rendre le bénéfice de l'opération maximal

Solution

Construire un graphe ayant pour sommets les A_i les B_j et deux nouveaux sommets s et t

Relier s à chacun des sommets B_j par un arc de capacité b_j

Relier chacun des sommets A_i à t par un arc de capacité c_i

Relier B_j à A_i chaque fois que l'expérience B_j nécessite l'instrument A_i, la capacité de cet arc est non bornée

Chercher la coupe de capacité minimale dans ce graphe

Explications

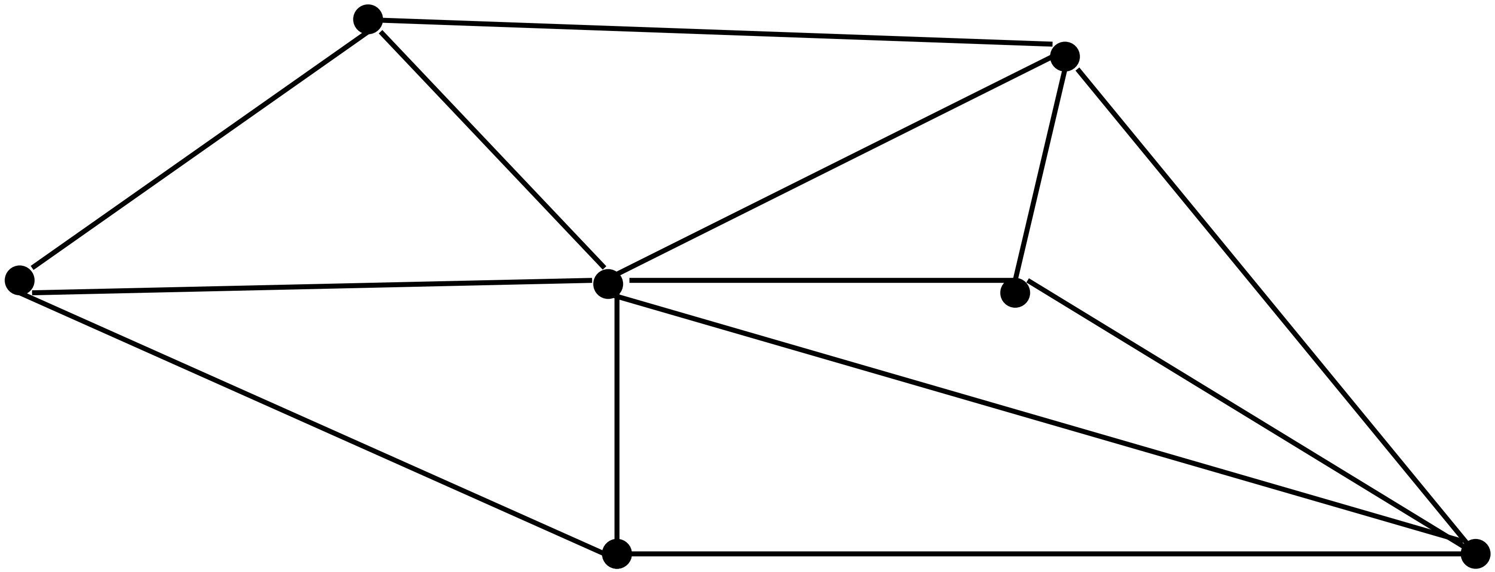
- On interprète une coupe de valeur finie Y, Z en considérant les b_j dans Y comme les expériences faites et les $A_i \in Y$ comme les objets emportés
- Tous les objets nécessaires à une expérience sont emportés, sinon valeur de la coupe est infinie
- La valeur de la coupe est égale à la somme des coûts des objets emportés et des bénéfices des expériences non réalisées
- Ainsi le bénéfice est la somme de tous les b_j diminuée de la valeur de la coupe

Coloration des pays d’une carte de géographie

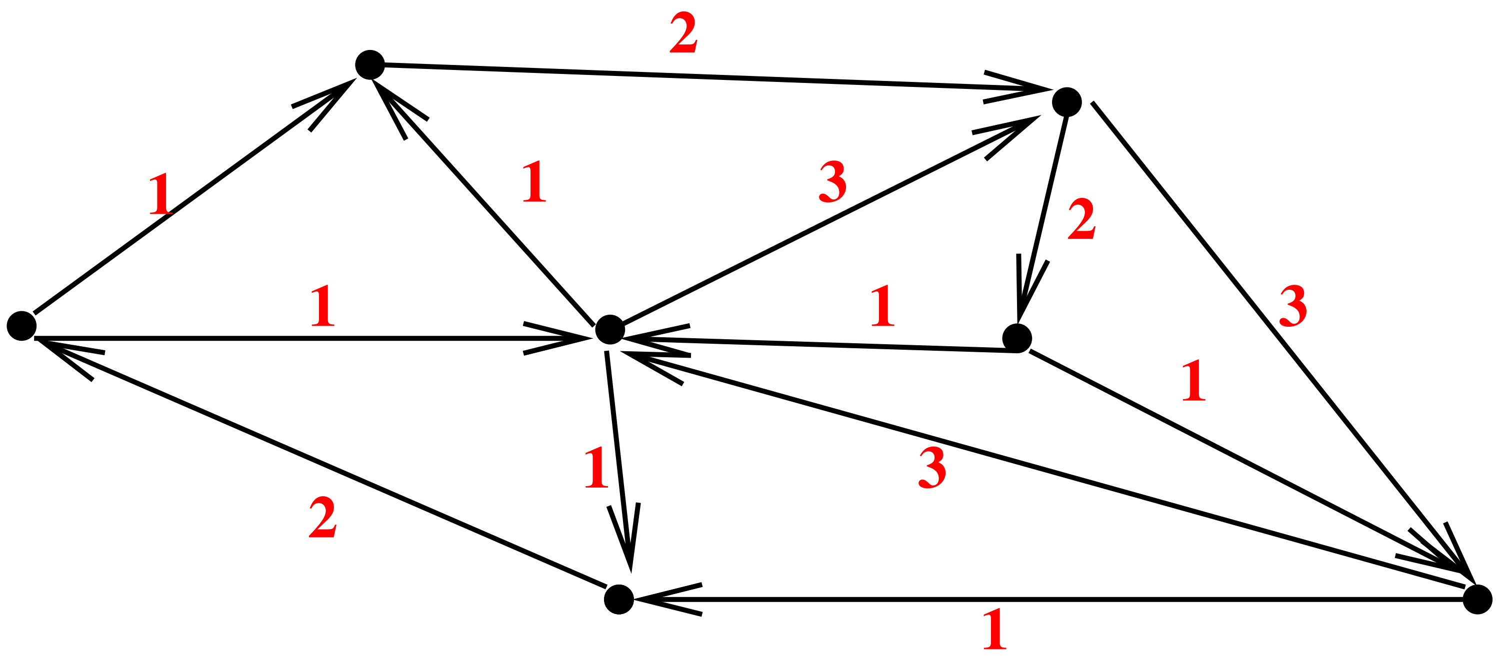
- On veut colorier les pays dans une carte de géographie de façon telle que deux pays voisins aient des couleurs distinctes et ceci avec un nombre minimal de couleurs
- On considère le graphe dont les sommets sont les points situés sur au moins deux frontières
- Les arêtes sont déterminées par les frontières

Solution

- Flots dans Z/nZ
- On se donne une orientation des arêtes du graphe
- On attribue des valeurs dans Z/nZ aux arêtes, dont aucune n'est nulle.
- On respecte la condition que pour chaque sommet la somme des valeurs des arcs entrants est égale à la somme des valeurs des arcs sortants (dans Z/nZ)
- La coloration s'obtient à partir du flot en attribuant une couleur quelconque à l'une des faces et en ajoutant la valeur du flot sur un arc à la couleur de la face de l'autre coté de l'arc ou en retranchant suivant le sens de parcours de l'arc

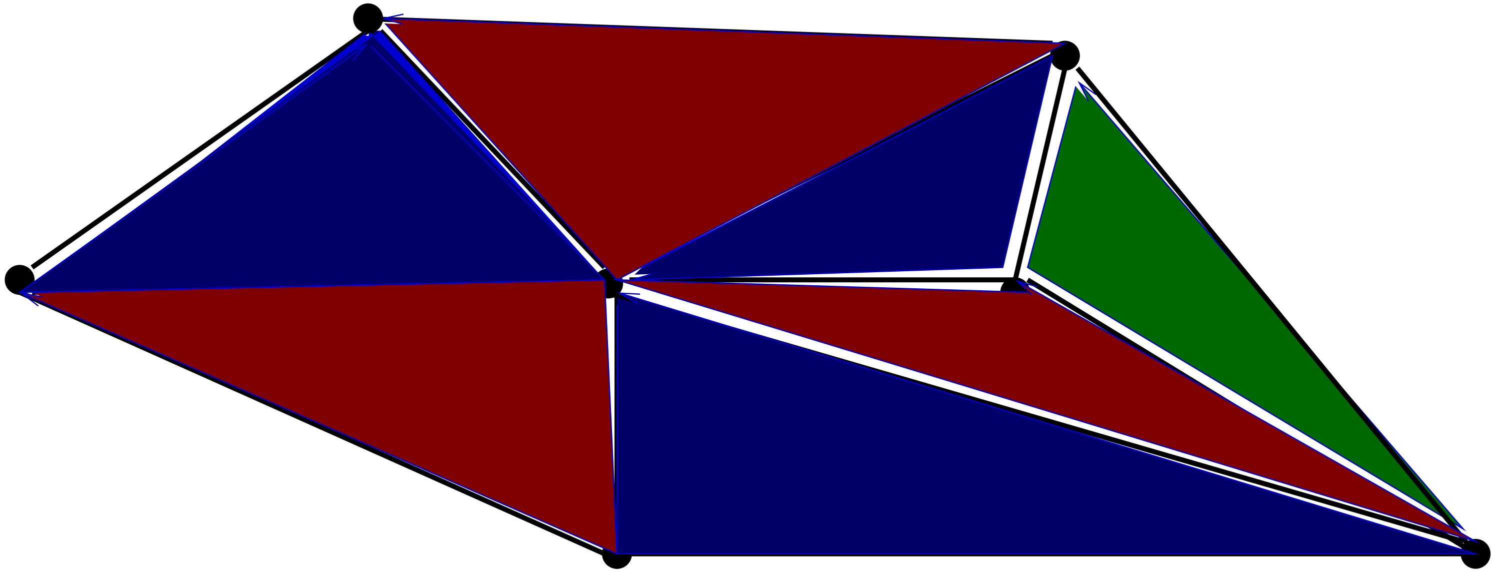


Carte à colorier



Un flot jamais nul dans $\frac{\mathbb{Z}}{4\mathbb{Z}}$

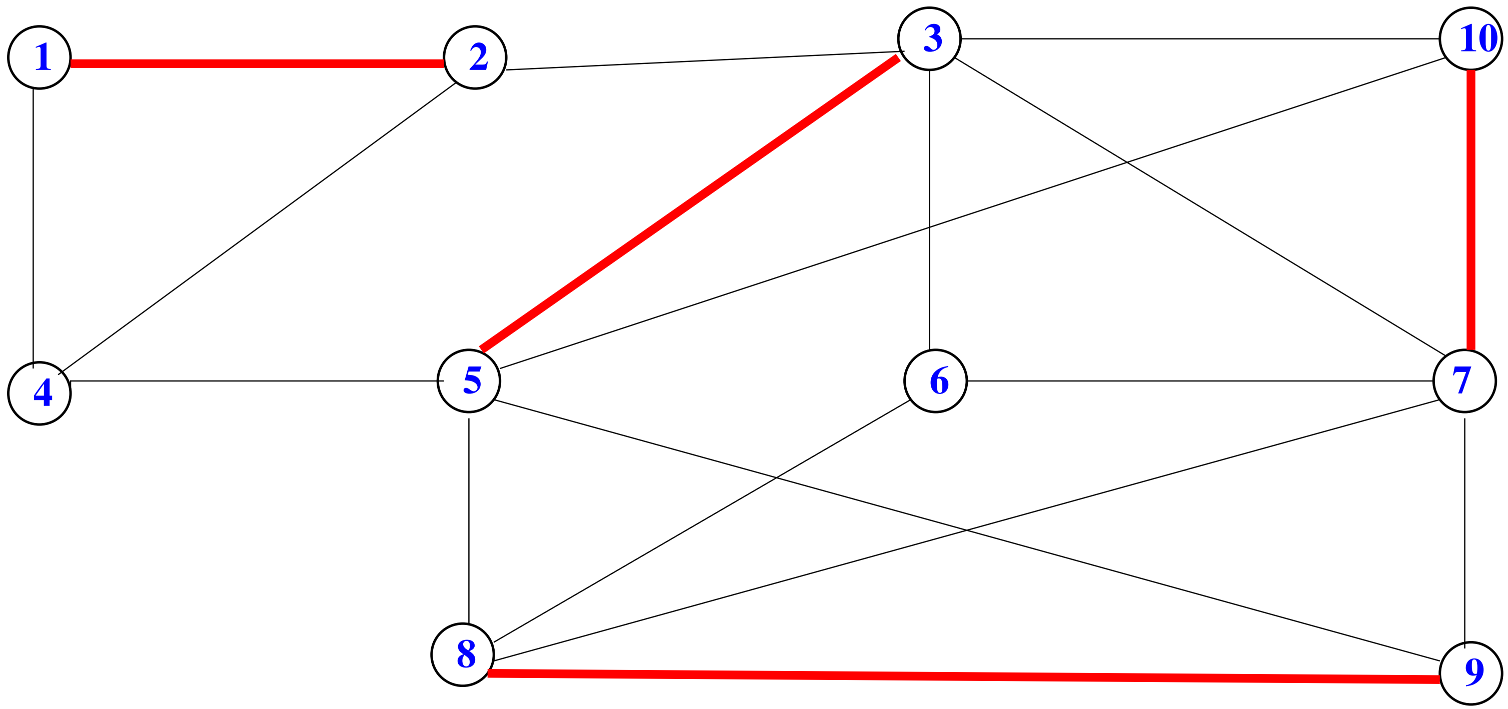
blanc -> bleu -> rouge -> vert



Carte coloriée

Couplages dans un graphe symétrique

- On se donne un graphe symétrique (non-orienté), un couplage est un ensemble d'arêtes sans extrémité communes
- On veut déterminer le couplage qui a un nombre maximum d'arêtes



Un couplage non maximal

Couplages dans un graphe biparti

- On suppose que le graphe $G = (X, E)$ est biparti c'est à dire qu'il existe une partition des sommets en deux sous ensembles Y et Z tels que toute arête a une extrémité dans Y et l'autre dans Z
- Le problème se ramène alors à un problème de flot sur un réseau construit à partir de G
- 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

Théorème du couplage maximal

Un couplage est maximal si et seulement s'il n'existe pas de chemin alternant joignant deux sommets non couplés

Mariages stables

- On se donne deux ensembles $X = \{x_1, x_2, \ldots, x_p\}$ et $Y = \{y_1, y_2, \ldots, y_p\}$
- Pour chaque $i = 1, \ldots, p$ une liste ordonnée F_i d'éléments de Y
- On note $\prec_x$ la relation de préference de l'individu x
- Pour chaque $j = 1, \ldots, q$ une liste ordonnée G_j d'éléments de X
- La recherche d'un couplage maximal consiste à travailler dans le graphe biparti (X, Y, E) où les arêtes sont tous les couples $\{x, y\}$ tels que y figure dans la liste F_x et x dans la liste G_y
- On note $\alpha(x)$ la personne couplée avec x et $\beta(y)$ la personne couplée avec y

Mariages stables

- Un couplage C est stable si et seulement si pour tout couple x, y qui n'est pas dans C on a

$$\beta(y) \prec_y x \qquad \text{ou} \qquad \alpha(x) \prec_x y$$

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

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

- Le couplage $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ n'est pas stable en raison de x_1 et y_2 qui se préfèrent à y_1 et x_2 respectivement.
- Par contre 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.