Les flots

Nous allons voir dans ce cours :

1. Les réseaux de transport;
2. Les flots, flot complet;
3. Flot maximum, algorithme de Ford-Fulkerson;
4. Graphe d'écart;
5. Flot maximal à coût minimal, Algorithme de Roy;
6. Flots canalisés, flots canalisés à coût minimal

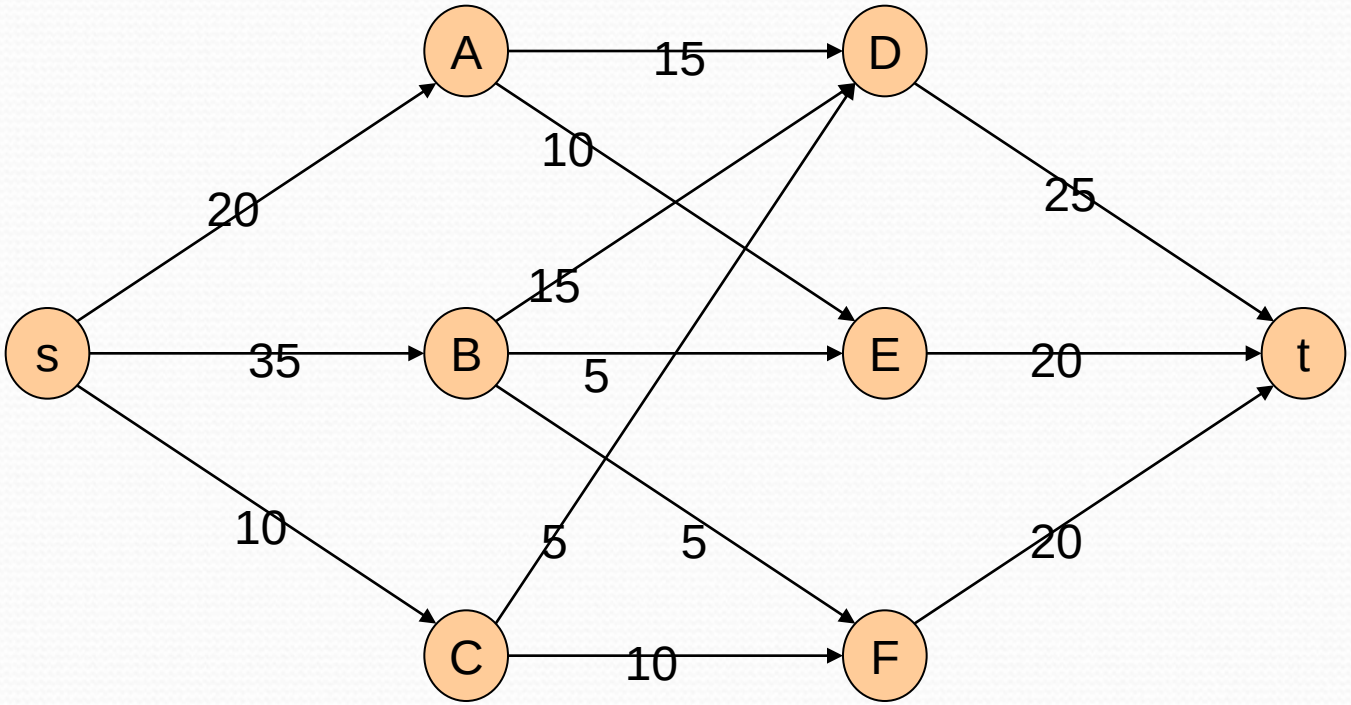
Les réseaux de transport

- **Exemple :** En trois dépôts A, B, C, on dispose respectivement de 20, 35 et 10 tonnes de marchandises. On a des demandes de 25, 20 et 20 tonnes aux destinations D, E et F. Il existe des possibilités de transport à l'aide de camions. Ces possibilités sont rapportées dans le tableau suivant :

	D	E	F
A	15	10	0
B	15	5	5
C	5	0	10

Problème : Déterminer un plan de transport permettant de transporter des origines aux destinations une quantité maximale.

Les réseaux de transport



- Réseau de transport :

  On appelle réseau de transport un graphe valué positivement sans boucle, ayant une racine s, un puits p et contenant l'arc (p,s) de valuation infinie. Les valuations des arcs sont appelées capacités.

Les flots

Définition : On appelle flot sur un réseau de transport G = (X, U) une application f : U dans R qui vérifie :

1) Les contraintes de capacité : pour tout arc (i,j) de U : $0 \leq f_{ij} \leq c_{ij}$;

2) Les contraintes de conservation (Kirchoff) pour tout sommet i de X, on a : $\sum_{j \in U^+(i)} f_{ij} = \sum_{k \in U^-(i)} f_{ki}$

<u>Lemme (généralisation des contraintes de Kirchoff) :</u> Si Y est un sous-ensemble de X, le flot sortant de Y est égal au flot entrant dans Y.

Flot complet

Définition : On dit qu'un flot est complet si tout chemin du réseau de transport allant de *s* à *p* contient au moins un arc saturé, c'est-à-dire un arc (i,j) tel que $f_{ij} = c_{ij}$.

Algorithme de recherche d'un flot complet :

(I) Marquer s.

(II) Soit i un sommet marqué non encore examiné; marquer j si j est un successeur non marqué de i avec $f_{ij} < c_{ij}$. La marque de j est +i.

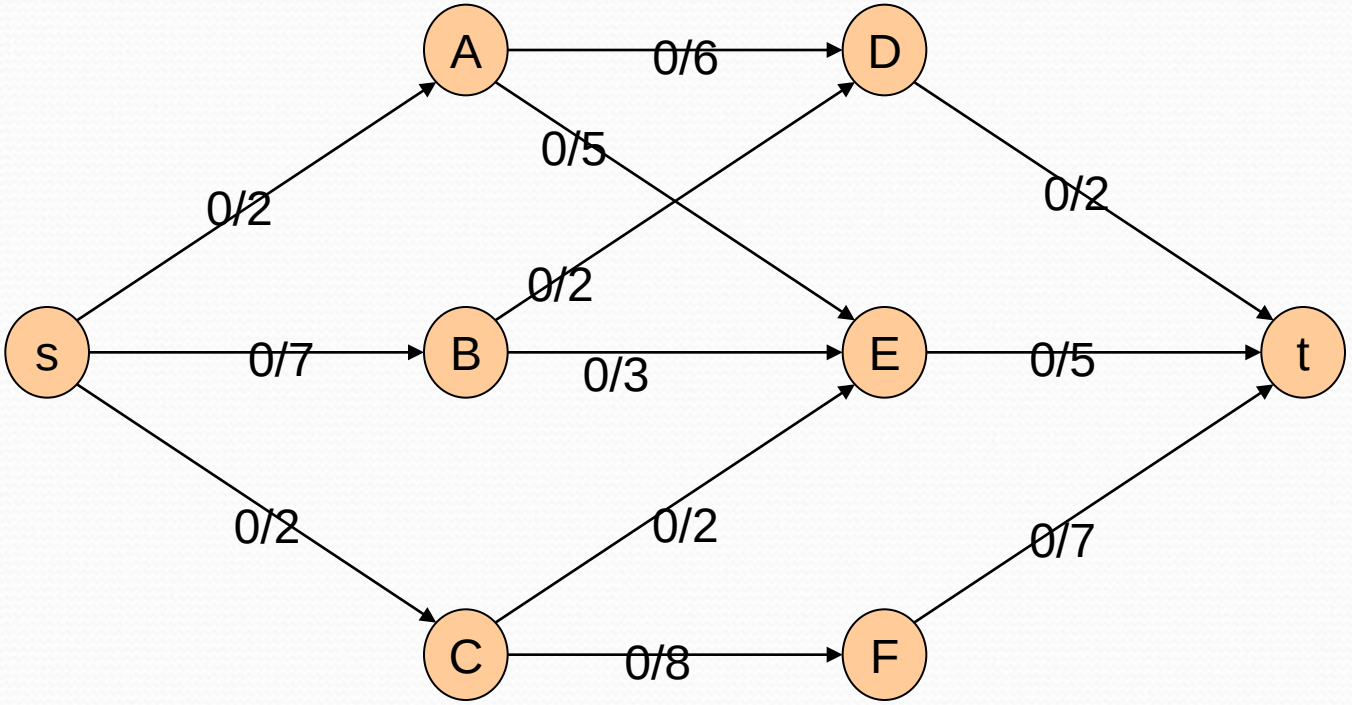
(III) Si p est marqué, aller en (IV).

Si tous les sommets marqués sont examinés, le flot est complet FIN.

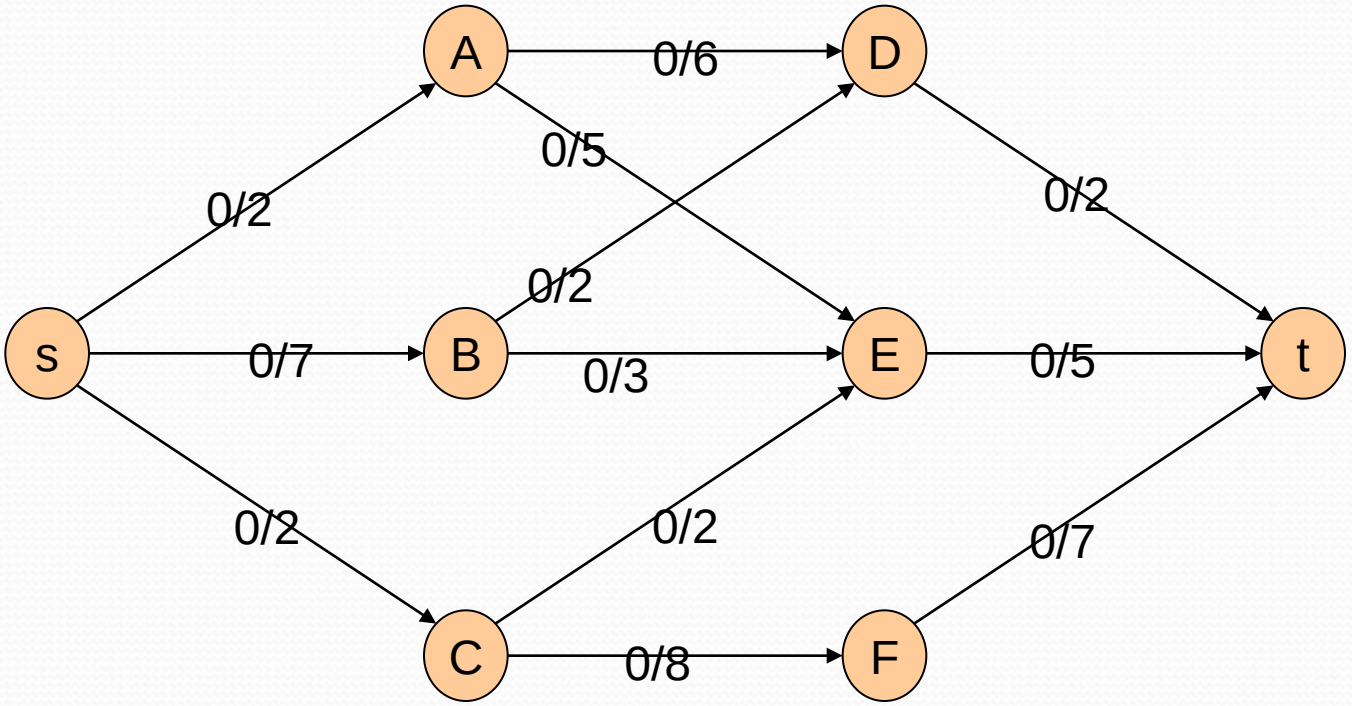
Sinon aller en (II).

(IV) Améliorer le flot. Effacer les marques (sauf celle de s) et aller en (I).

Exemple

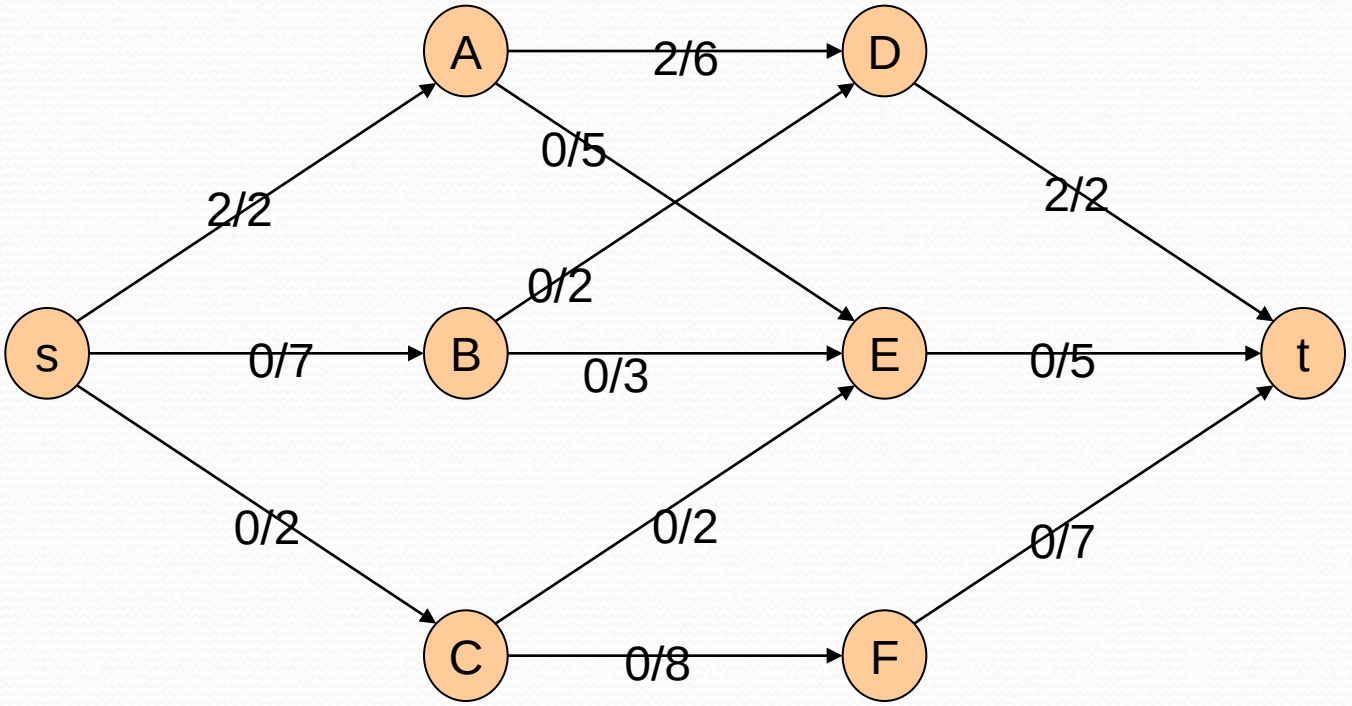


Exemple

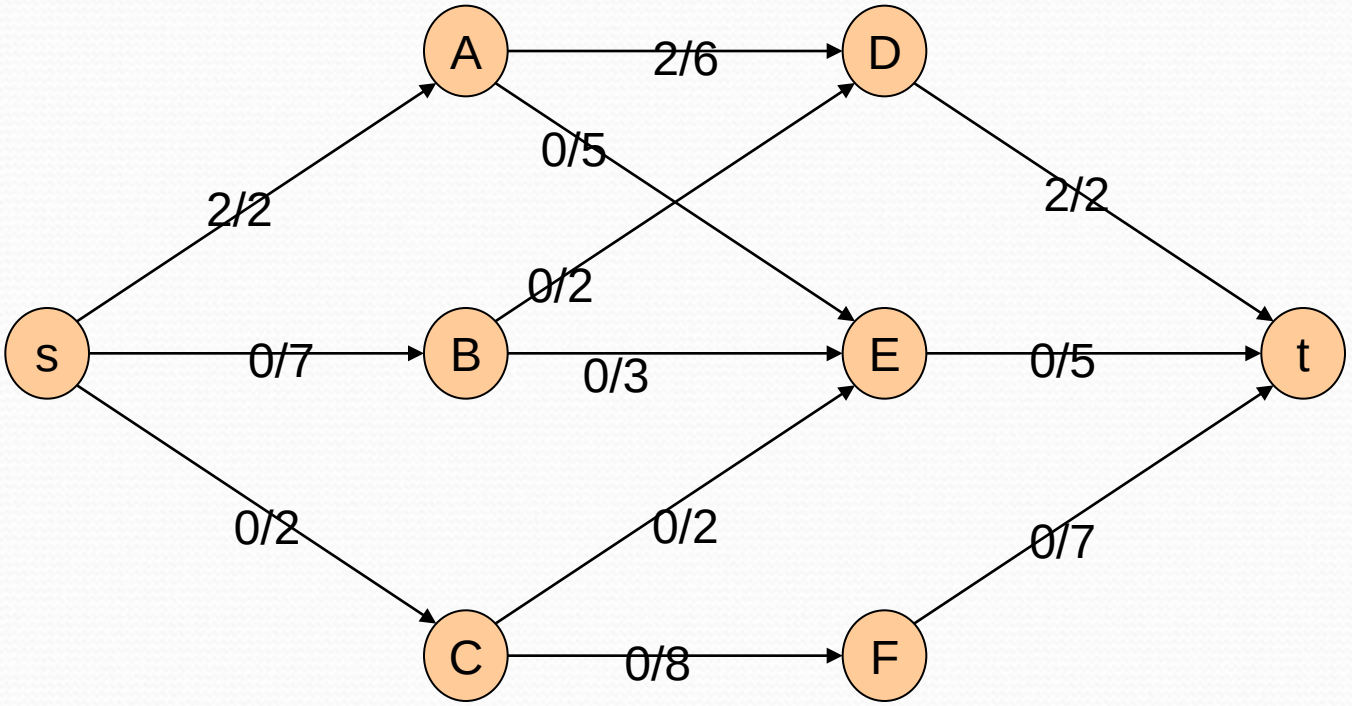


P = ((s,A),(A,D),(D,t)), $\min_{a \in P} u(a)-f(a)=2$

Exemple

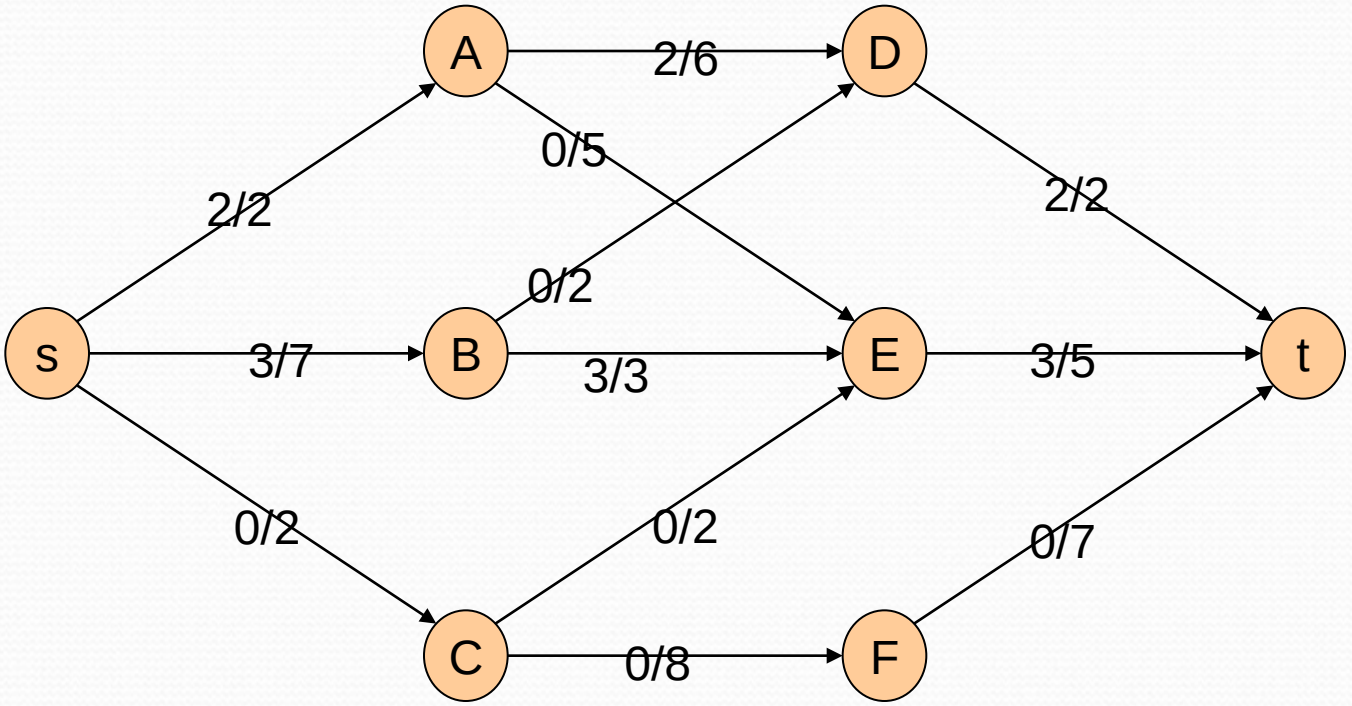


Exemple

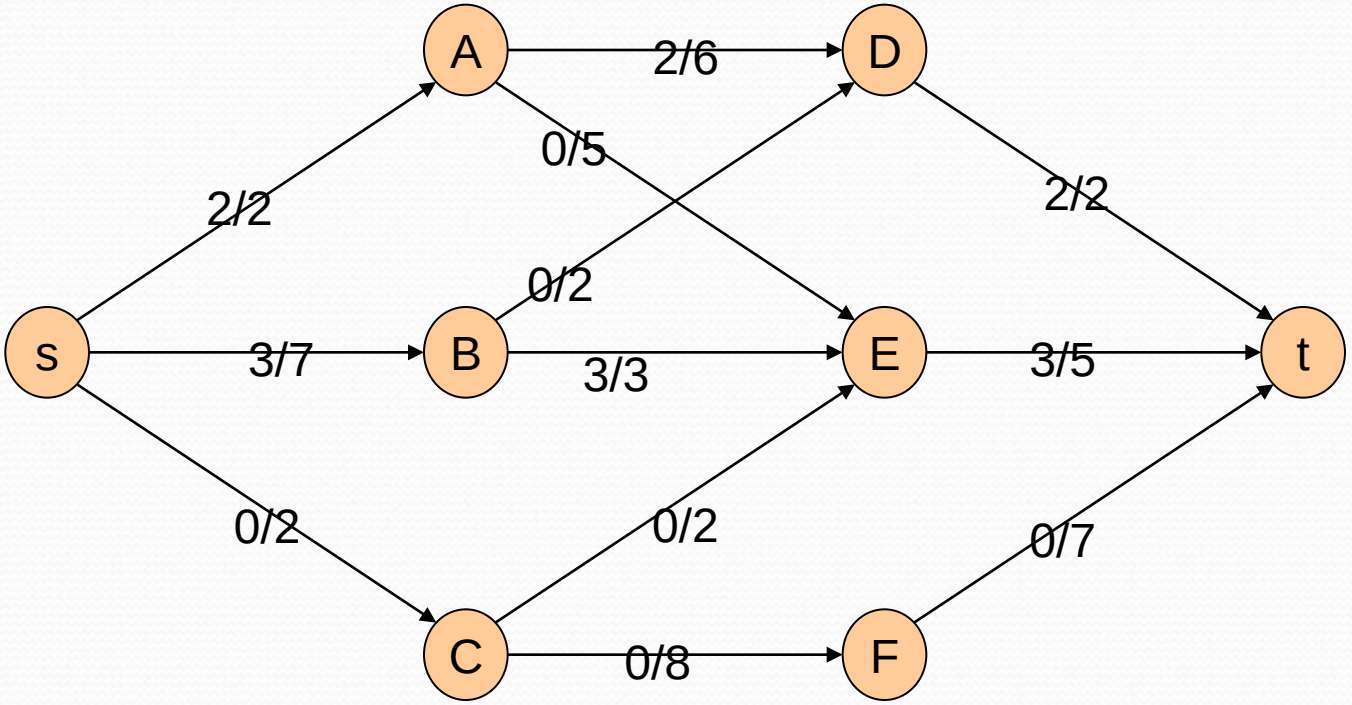


P = ((s,B),(B,E),(E,t)), $\min_{a \in P} u(a)-f(a)=3$

Exemple

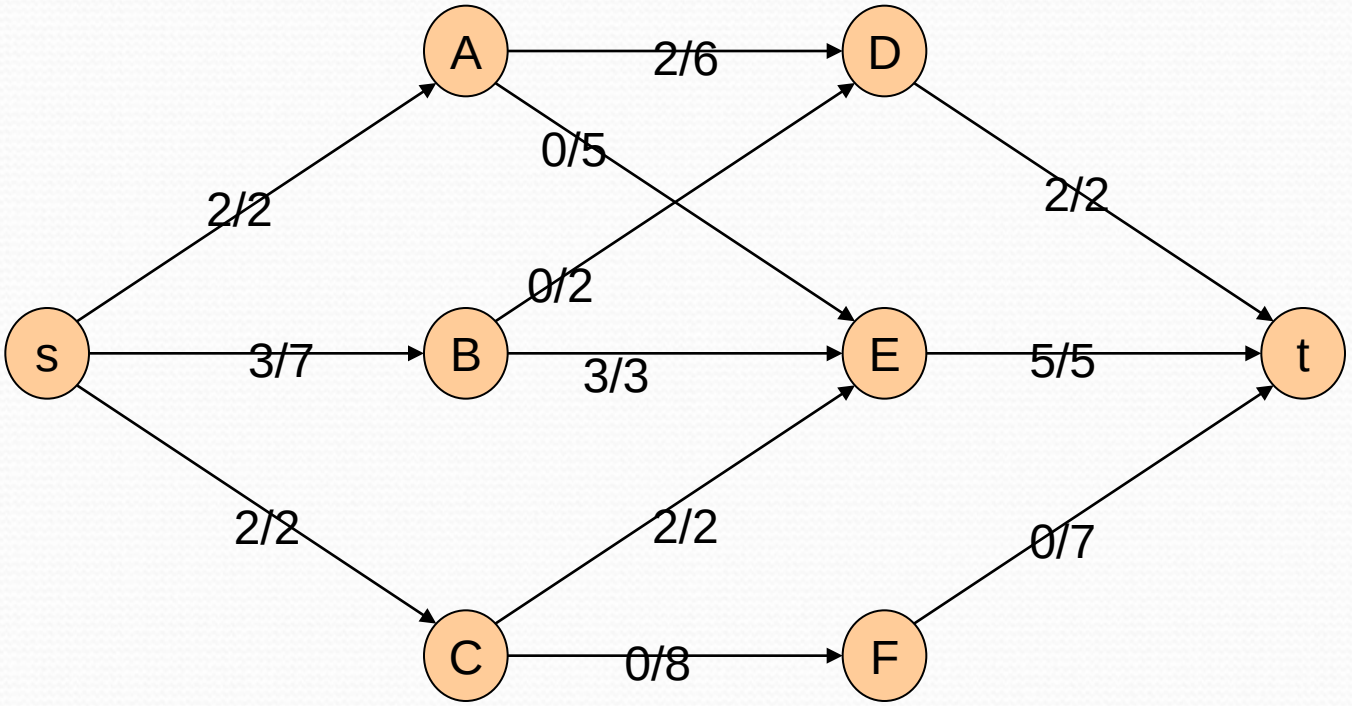


Exemple



P = ((s,C),(C,E),(E,t)), $\min_{a \in P} u(a)-f(a)=2$

Exemple



value(*f*) = 7

Flot maximal
Algorithme de Ford-Fulkerson

(I) Marquer l'entrée s par *.

(II) Soit i un sommet marqué non examiné.

Étudier tous les successeurs j de i :

Marquer j s'il est non marqué et si $f_{ij} < c_{ij}$ par +i.

Étudier tous les prédécesseurs k de i:

Marquer k s'il est non marqué et si $f_{ki} > 0$ par -i.

(III) Si p est marqué, aller en (IV)

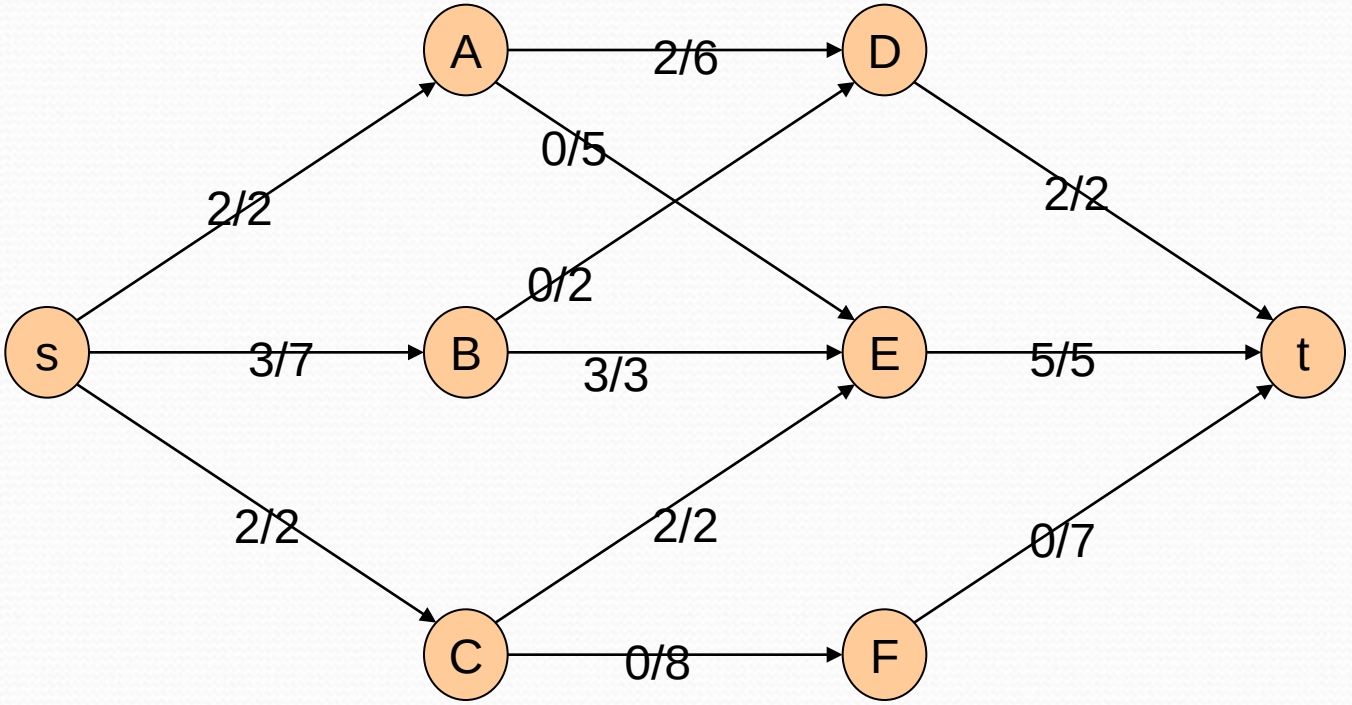
S'il reste des sommets marqués non examinés, aller en (II).

Sinon le flot est optimal, FIN.

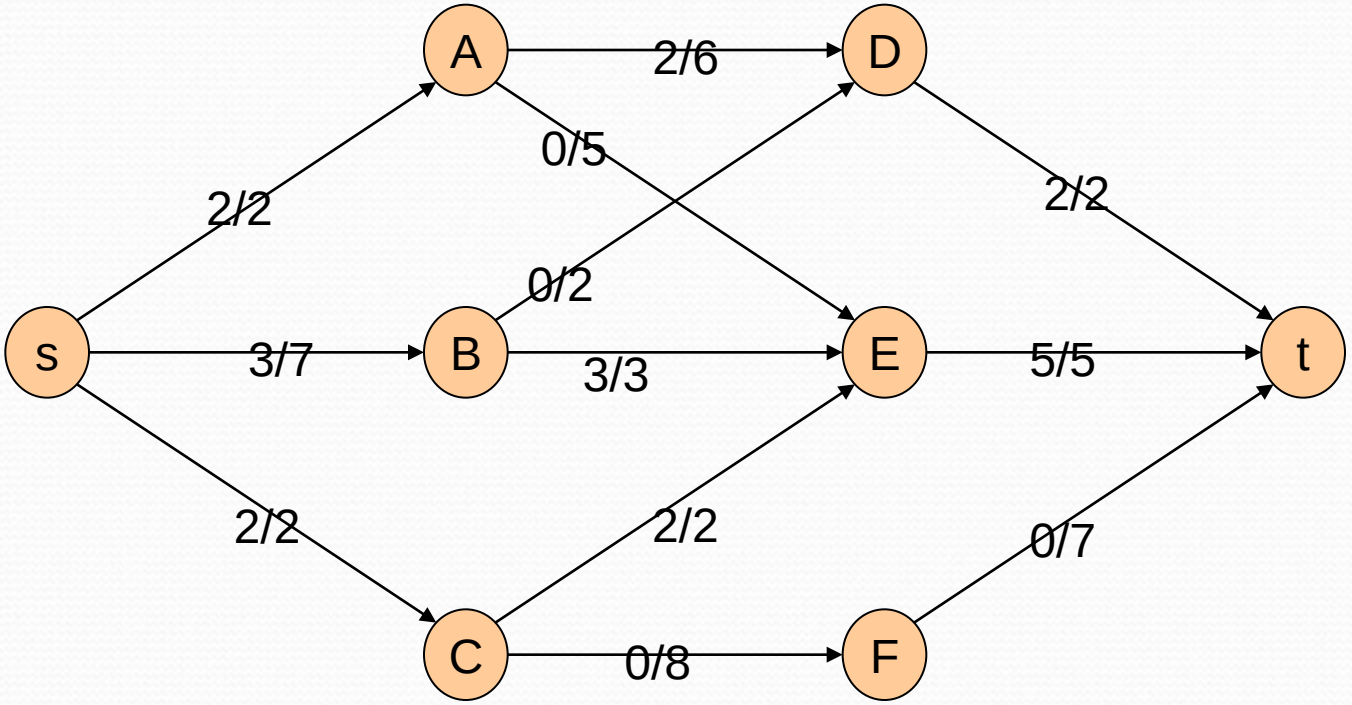
(IV) Améliorer le flot à l'aide de la chaîne améliorante ayant permis de marquer p.

Effacer les marques, sauf celle de s, et aller en (I).

Exemple (précédent)

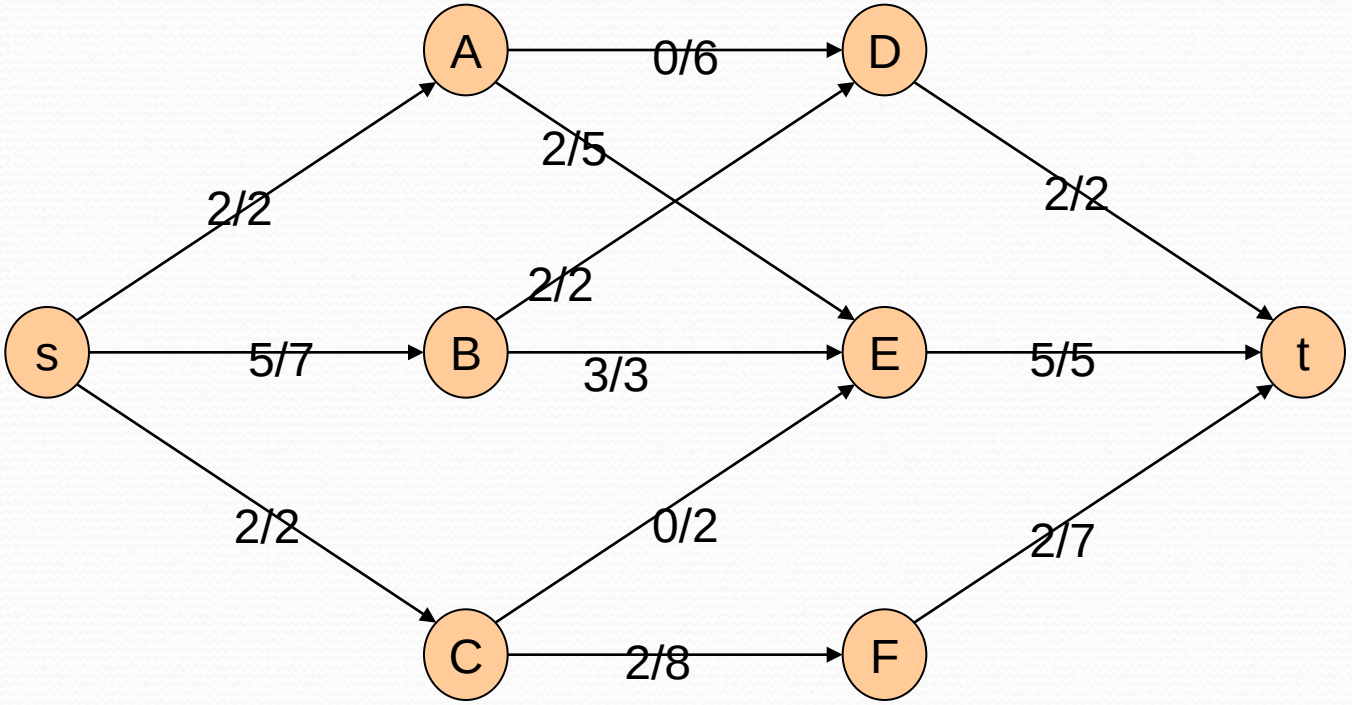


Exemple (précédent)



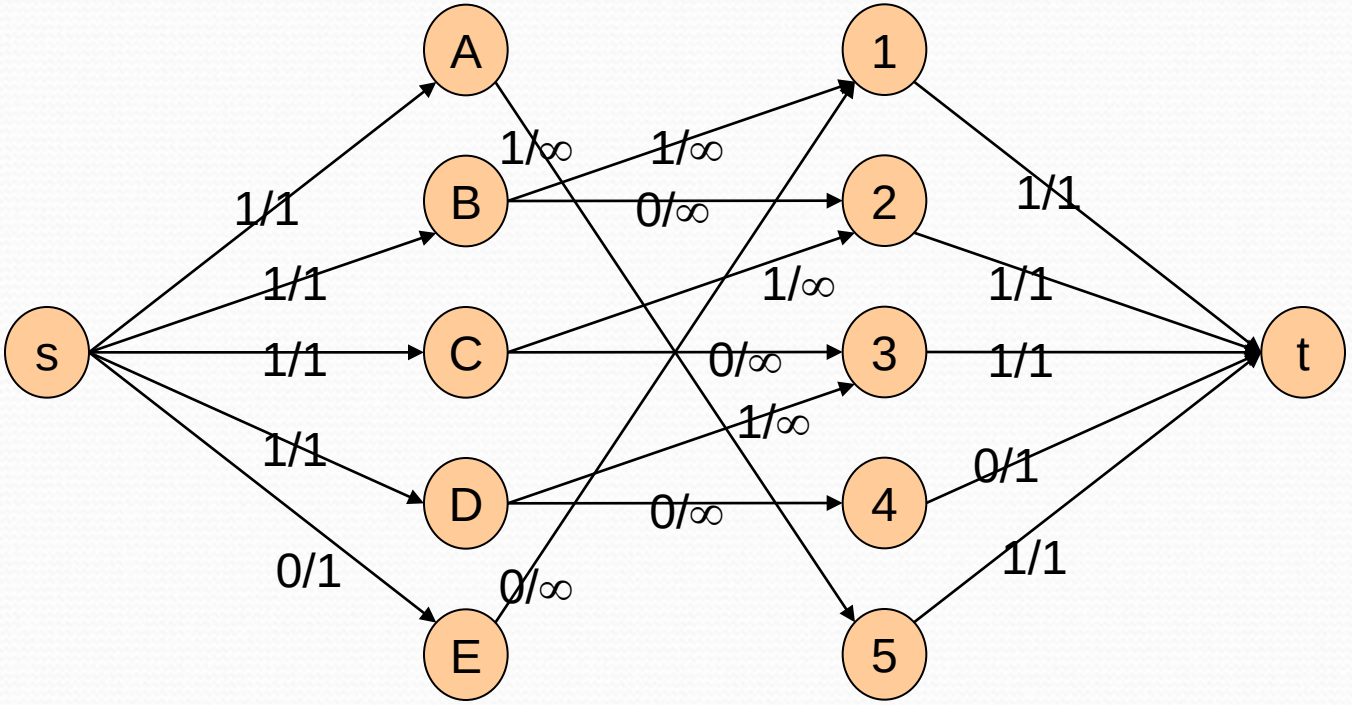
P = ((s,B),(B,D),(D,A),(A,E),(E,C),(C,F),(F,t)),
$\min_{a \in P} u_f(a) = 2$

Exemple (précédent)



value(*f*) = 9, *f* is maximum

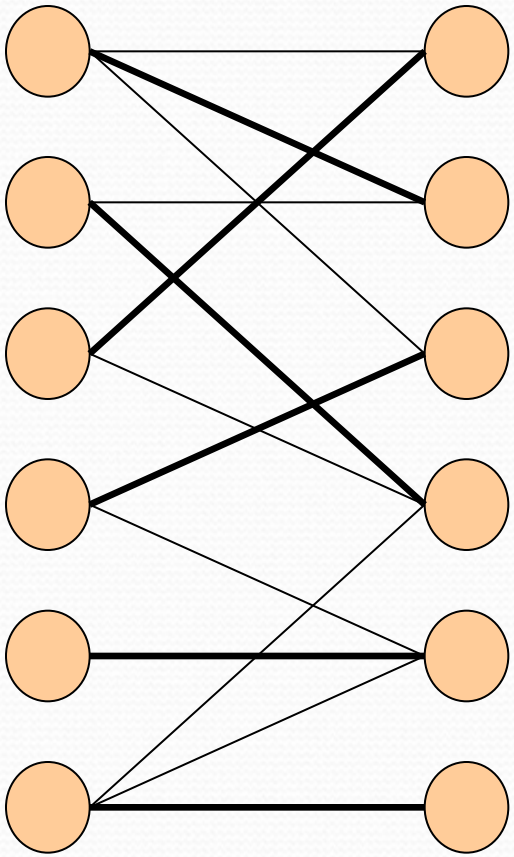
Un autre problème de flot max…



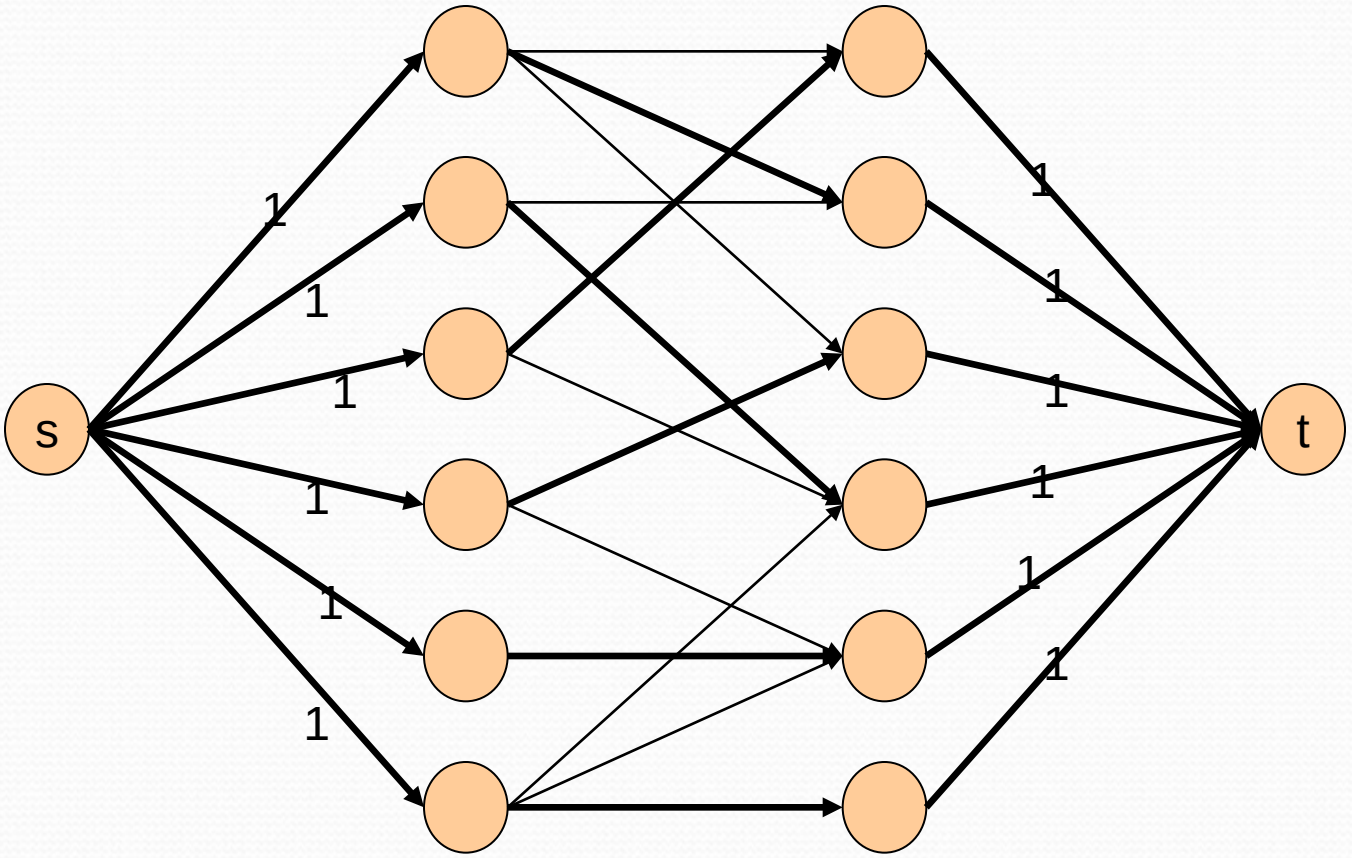
Transform the above complet flow to a maximum one

Le problème de couplage maximal

- Transformer ce problème à un problème de flot max

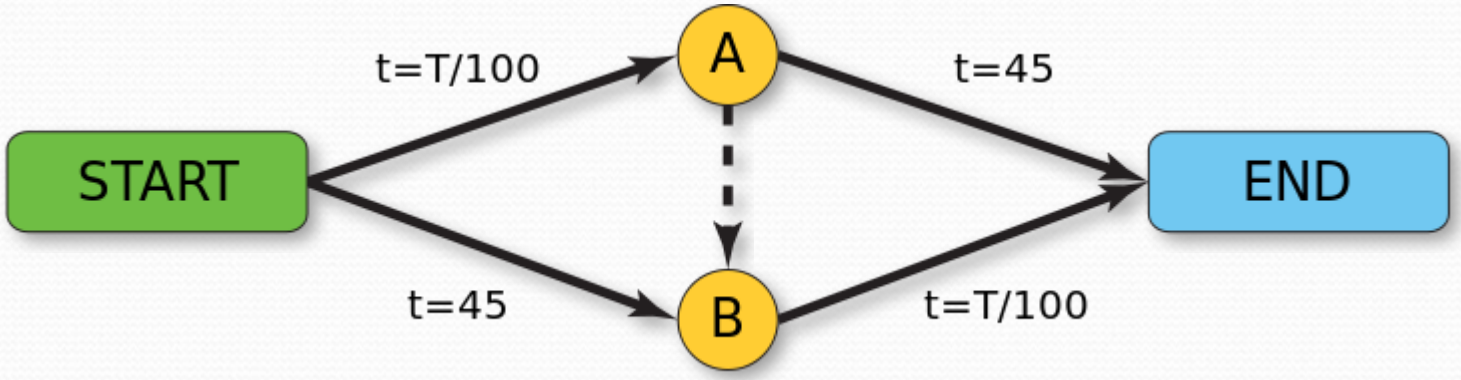


Le problème de couplage maximal



Le paradox de Braess

Le paradox de Braess montre que l'ajout d'un arc dans un réseau peut empirer le résultat attendu



Flot maximal versus coupe minimale (I)

Définition de la coupe minimale :

Une coupe est un ensemble de sommets contenant *s* et ne contenant pas *p*. La capacité d'une coupe est la somme des capacités des arcs sortant de cette coupe.

Théorème de Ford – Fulkerson : ***La capacité minimale d'une coupe est égale au flot maximal.***

Flot maximal versus coupe minimale (II)

Corollaire 1 : ***Une C.N.S pour qu'un flot soit maximal est qu'il n'existe pas de chaîne améliorante.***

Corollaire 2 : ***L'algorithme de Ford-Fulkerson construit un flot maximal.***

Complexité de l'algorithme de Ford-Fulkerson

Algorithme du graphe d'écart

Définition du graphe d'écart $G_e(f)$:

à tout arc (i, j) de G, correspond deux arcs : un arc (i, j) de $G_e(f)$ de capacité $c_{ij} - f_{ij}$ et un arc (j, i) de $G_e(f)$ de capacité f_{ij} (si la capacité d'un arc est nulle dans $G_e(f)$, on ne représente pas l'arc, car il est inutile).

Algorithme du graphe d'écart :

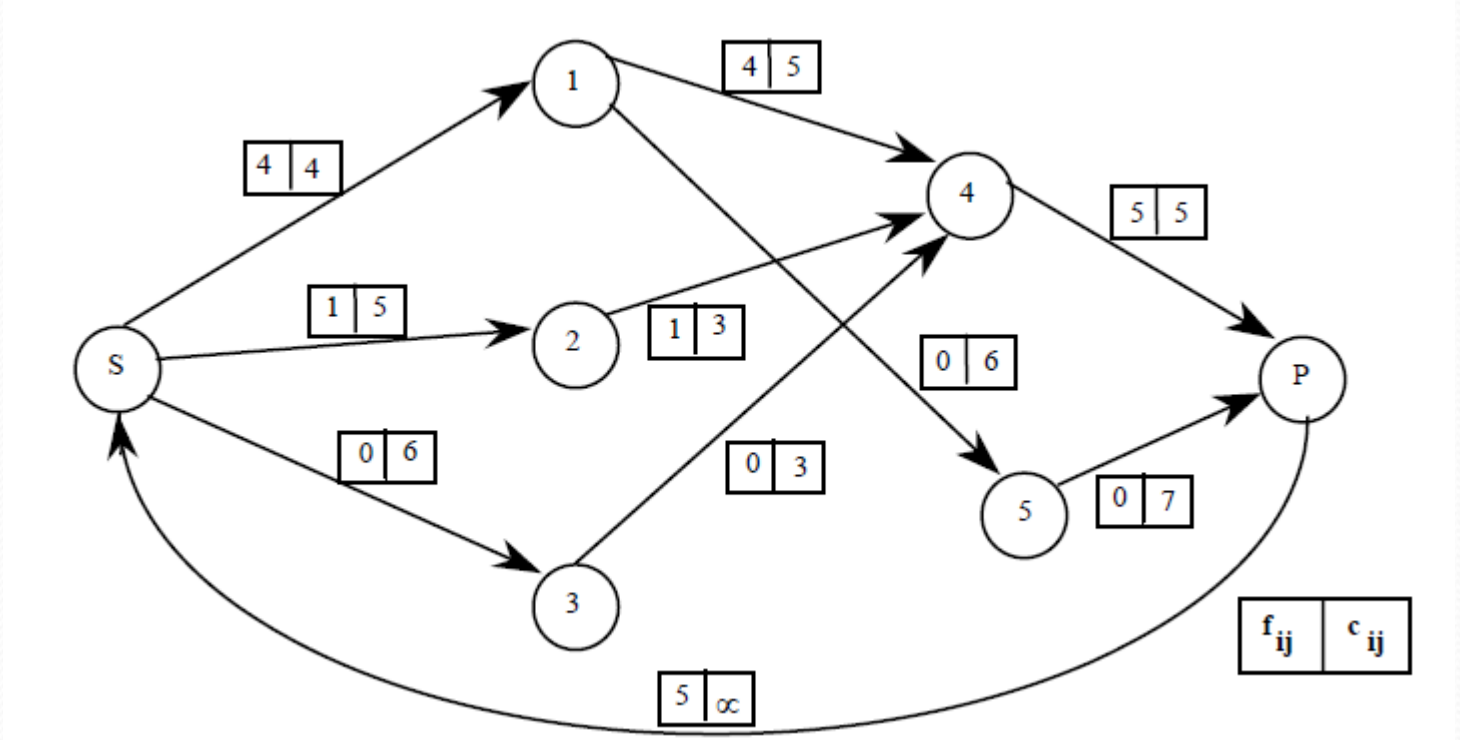
(I) Initialiser f sur tous les arcs (par exemple, f = 0, si b(u)= 0, pour tout u).

(II) Construire $G_e(f)$ et chercher un chemin de s à p dans $G_e(f)$.

Aller en (III) si un chemin a été trouvé sinon FIN.

(III)Améliorer le flot f le long du chemin obtenu et retourner en (II).

Un exemple



Le problème du flot maximal à coût minimal

Algorithme de ROY :

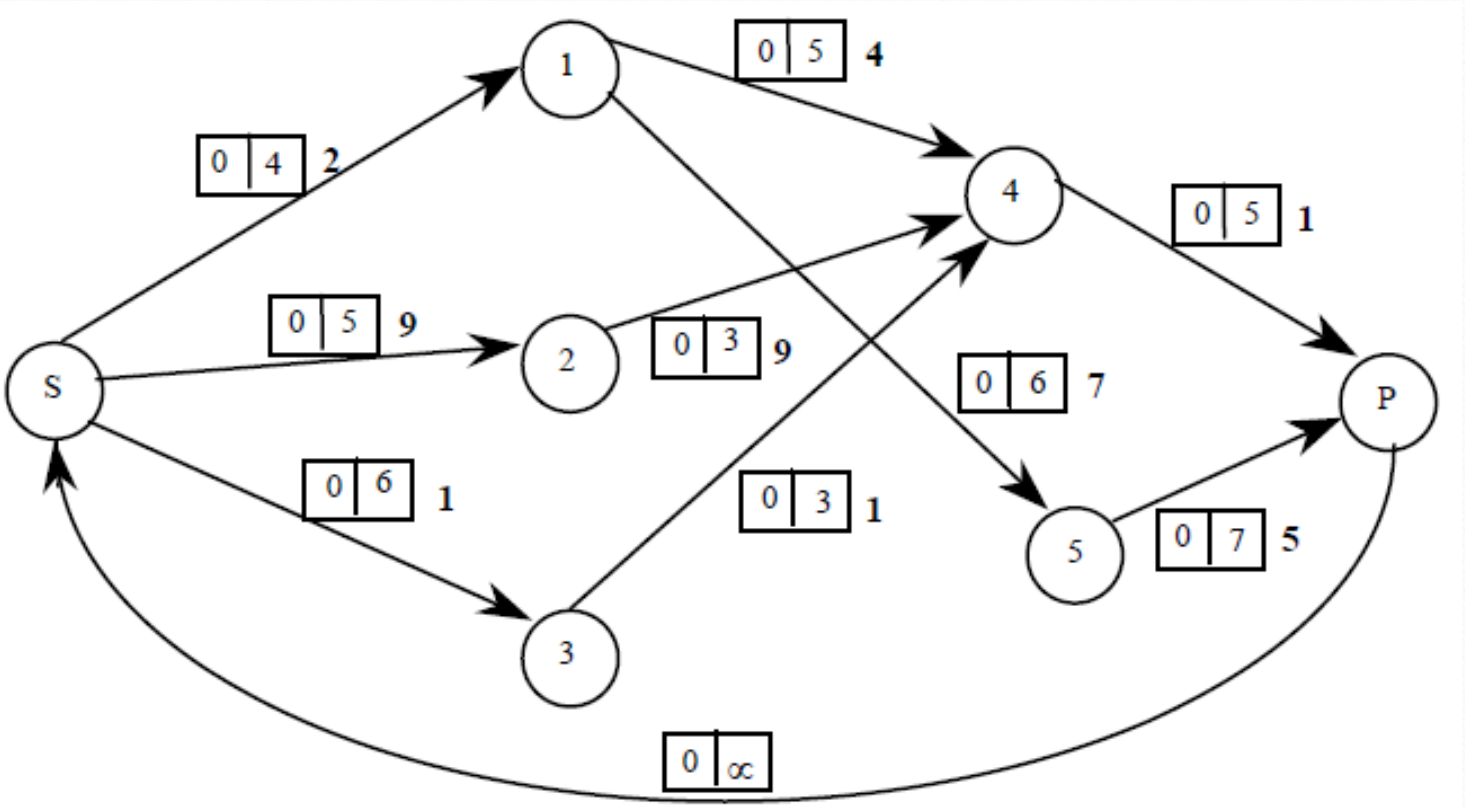
(I) Poser f = 0.

(II) Construire $G_e(f)$ en introduisant un coût d_{ij} sur l'arc (i, j) et un coût - d_{ij} sur l'arc (j, i).

(III) Chercher un chemin de coût minimal dans $G_e(f)$ allant de s à p. S'il n'y a pas de chemin de s à p, le flot est maximal de coût minimal alors FIN.

(IV) Sinon améliorer le flot f et retourner en (II).

Un exemple



Les problèmes de flots canalisés à coût minimal

Définition : Un flot canalisé est une application f de U dans N (ensemble des entiers) satisfaisant les contraintes de Kirchoff et les contraintes de bornes et de capacités.

Théorème d'HOFFMAN :

Une C.N.S pour qu'il existe un flot canalisé dans le réseau G = (X, U, b, c) est que : pour tout $Y \subseteq X$, la somme des bornes des arcs entrants dans Y est inférieure ou égale à la somme des capacités des arcs sortants de Y, c'est-à-dire :

$$\sum_{u \in U^-(Y)} b(u) \leq \sum_{u \in U^+(Y)} c(u)$$

Les flots compatibles

Algorithme de recherche d'un flot compatible :

(I) Partir du flot f nul.

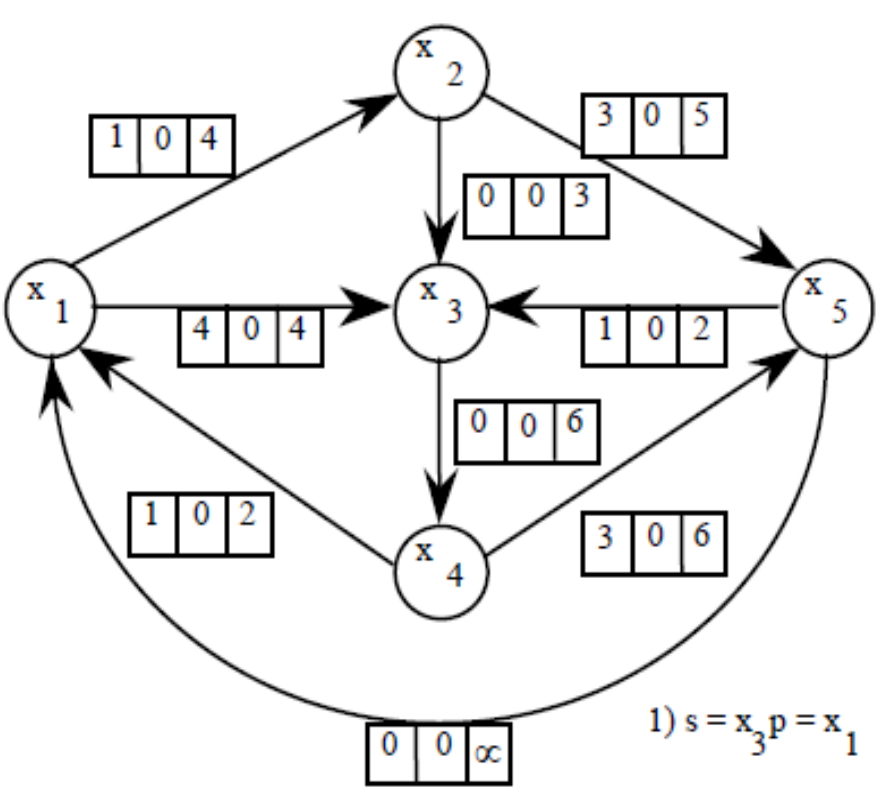
(II) Chercher un arc u tel que $f_u < b(u)$

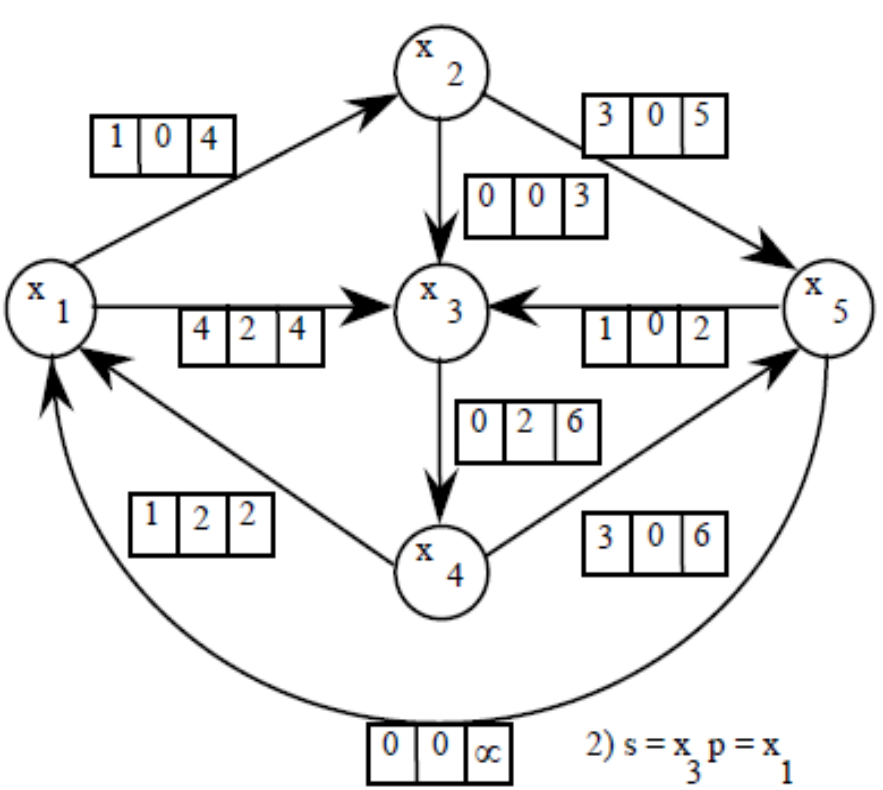
- si un tel arc n'existe pas alors FIN 1;
- sinon poser s = extrémité terminale de u et p = extrémité initiale de u.

(III) Chercher une chaîne de s à p dont les arcs dans le sens + sont insaturés et les arcs dans le sens - ont un flux strictement supérieur à la borne; si ce n'est pas possible alors FIN 2.

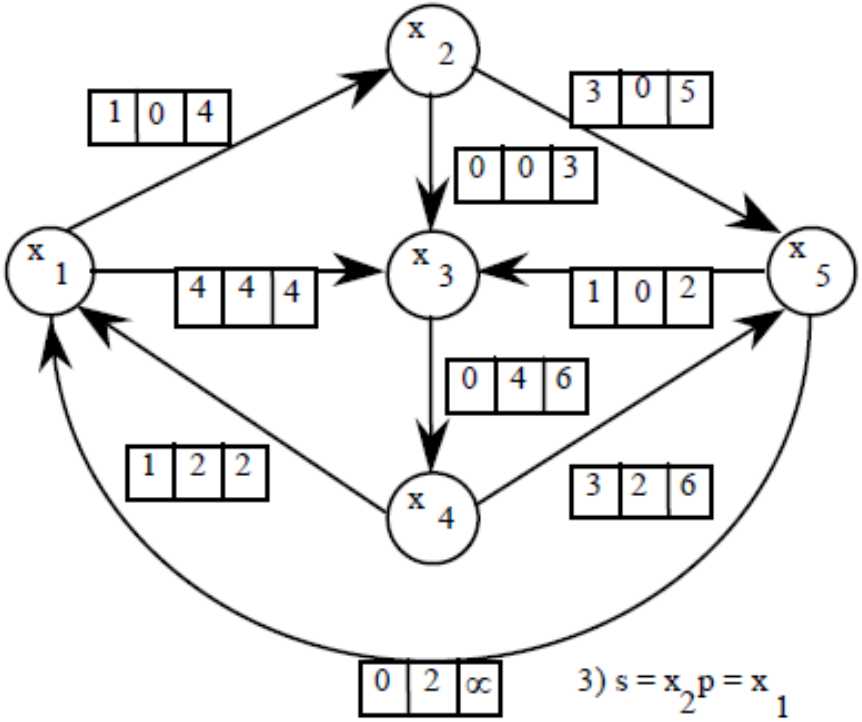
(IV) Utiliser cette chaîne pour améliorer le flot en cherchant à satisfaire la contrainte de borne sur l'arc u et retourner en (ii).

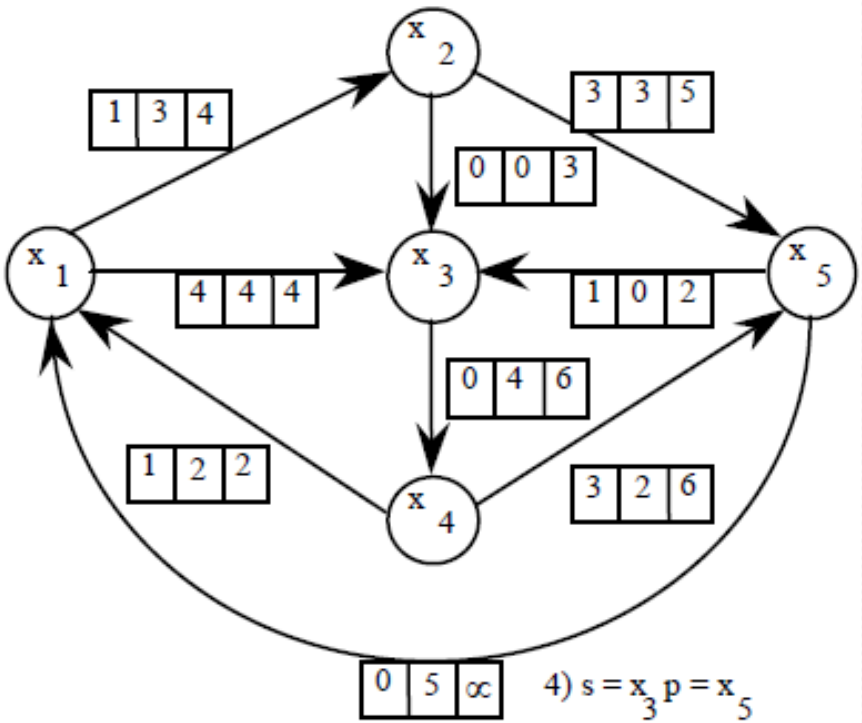
Un exemple (I)


x 2
x 1
x 3
x 5
x 4
1 0 4
3 0 5
0 0 3
4 0 4
1 0 2
0 0 6
1 0 2
3 0 6
0 0 ∞
1) s = x3 p = x1

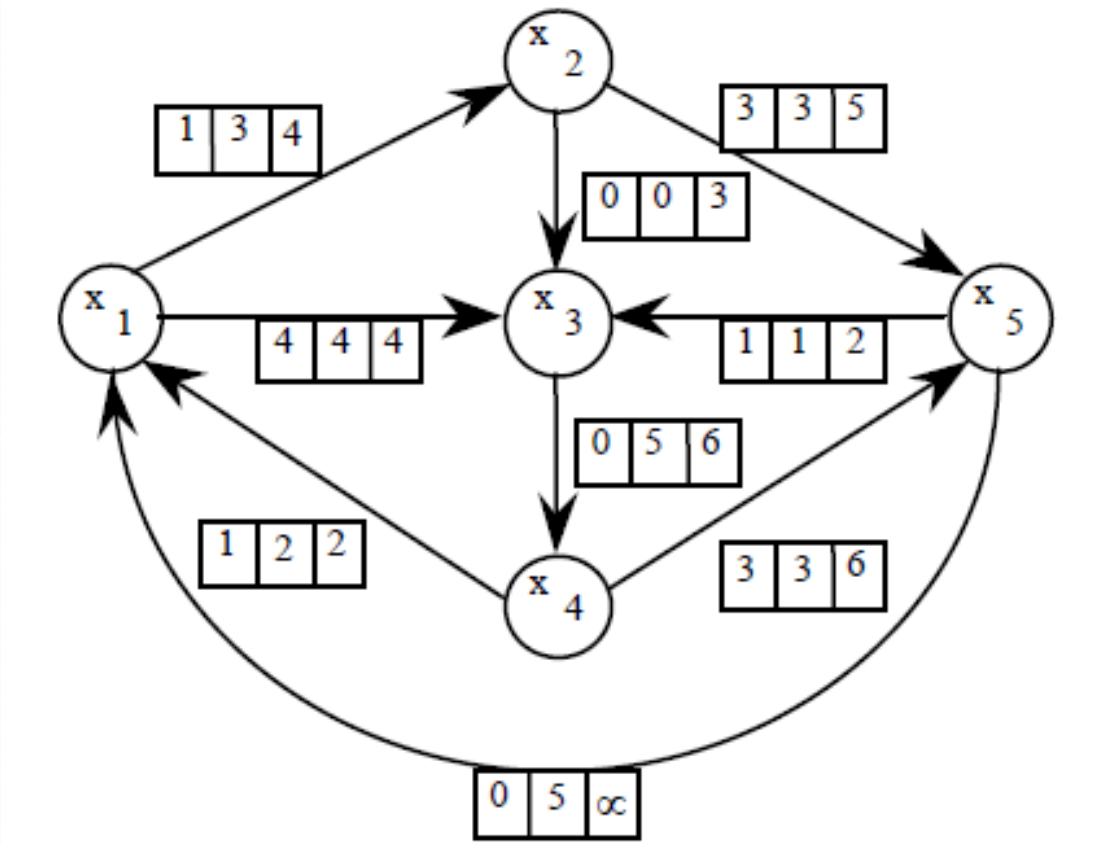

x 2
x 1
x 3
x 5
x 4
1 0 4
3 0 5
0 0 3
4 2 4
1 0 2
0 2 6
1 2 2
3 0 6
0 0 ∞
2) s = x3 p = x1

Un exemple (II)

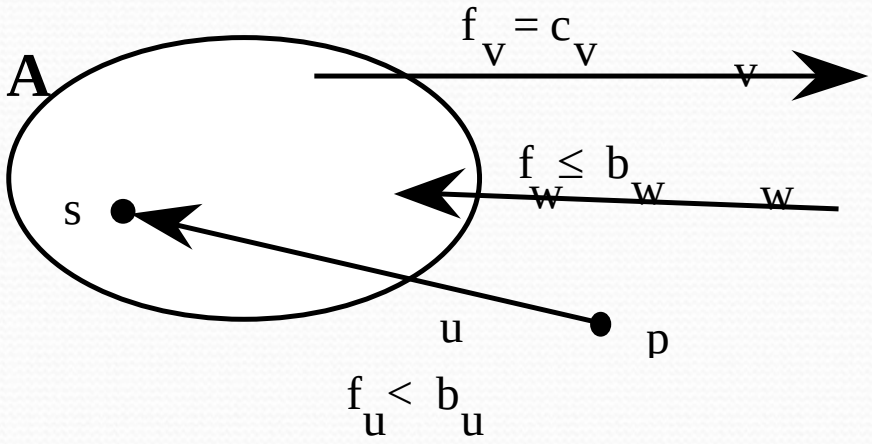




Un exemple (III)



Validité de l'algorithme



Cas de non-existence d'un flot compatible (FIN2)

Decomposition des flots

Décomposition d'un flot sur une base de circuits

Proposition

Une condition nécessaire et suffisante pour qu'un flot f soit positif est qu'il existe une famille de circuits $\mu 1, \mu 2, ..., \mu p$ et des coefficients $\lambda 1, \lambda 2, ..., \lambda p$ entiers positifs tels que :

$f = \lambda 1\mu 1 + \lambda 2\mu 2 + ... + \lambda p\mu p$.

Graphe d'écart d'un flot canalisé.

Soit $f0$ un flot compatible et $Ge(f0)$ le graphe d'écart associé. Soit φ un flot sur $Ge(f0)$ on définit un nouveau flot f compatible sur G par la formule notée $f = f0 \oplus \varphi$ telle que : $(f)u = (f0)u + (\varphi)u+ - (\varphi)u-$

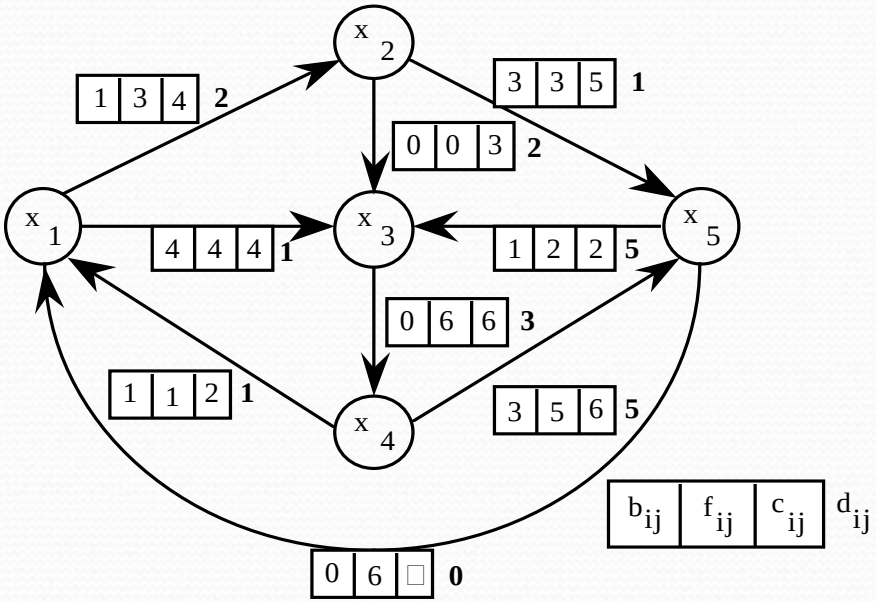
Et le coût du flot f sur G est égal au coût du flot $f0$ sur G plus le coût du flot φ sur le graphe d'écart.

Flot canalisé à coût minimal

Théorème d'optimalité

Un flot $f0$ compatible est de coût minimal si et seulement si Ge($f0$) ne contient pas de circuit de coût strictement négatif.

Exemple



problème de flot canalisé à coût minimal.

Preuve de l'algorithme de Roy

Proposition

l'algorithme de ROY détermine un flot maximal de coût minimal.

Preuve par récurrence: On montre par récurrence que tous les flots f^k construits au cours de l'algorithme de ROY sont de coût minimal parmi les flots de valeur vk. Il résultera que le dernier flot sera maximal à coût minimal.