



Chapitre 5 : Flot maximal dans un graphe

J.-F. Scheid

2011–2012

I. Définitions

- ① Graphe
- ② Graphe valué
- ③ Représentation d'un graphe (matrice d'incidence, matrice d'adjacence, successeurs/prédécesseurs)
- ④ Flot dans un graphe

II. Problème de flot optimal dans un graphe

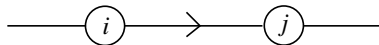
III. Algorithme de Ford-Fulkerson

1) Graphe

Graphe $G = (E, \Gamma)$

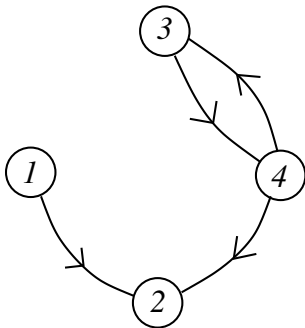
- E : ensemble fini des **sommets**
- Γ : ensemble fini de couples ordonnés (i, j) avec $i, j \in E$.
Les éléments de Γ sont appelés les **arêtes** du graphe

Notation :



Remarque : Les graphes considérés sont tous orientés : les arêtes (i, j) et (j, i) sont distinctes.

Exemple de graphe : $E = \{1, 2, 3, 4\}$
 $\Gamma = \{(1, 2), (3, 4), (4, 2), (4, 3)\}$



Exemples de modélisation par des graphes :

- réseau routier : les sommets sont les intersections des routes, les arêtes représentent les routes.
- cheminement dans un réseau informatique.
- Web modélisé par un graphe. Les sommets sont les pages Web et les arêtes sont les liens hypertexte entre ces différentes pages.

2) Graphe valué

$G = (E, \Gamma, c)$ est un **graphe valué** si (E, Γ) est un graphe auquel on associe une fonction $c : \Gamma \rightarrow \mathbb{R}$ appelée **capacité**.

La capacité de l'arête (i, j) est notée c_{ij} .

Notation :



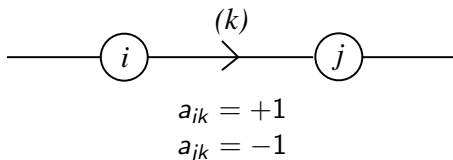
Exemple : La capacité c_{ij} représente par exemple la longueur du tronçon de route (i, j) , le nombre max. de voitures par unité de temps entre deux villes i et j , la bande passante maximale entre les serveurs i et j ...

3) Représentation d'un graphe

a) Matrice d'incidence sommet-arête

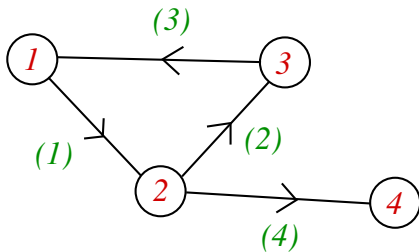
Soit un graphe **sans boucle** c-à-d sans arête (i, i) , avec n sommets et m arêtes. On définit A la **matrice d'incidence** de taille $n \times m$:

$$a_{ik} = \begin{cases} +1 & \text{si le sommet } i \text{ est l'extrémité } \mathbf{initiale} \text{ de l'arête } k \\ -1 & \text{si le sommet } i \text{ est l'extrémité } \mathbf{terminale} \text{ de l'arête } k \\ 0 & \text{sinon} \end{cases}$$



Exemple : $E = \{1, 2, 3, 4\}$

$\Gamma = \{(1, 2), (2, 3), (3, 1), (2, 4)\}$



Matrice d'incidence $A =$

	(1,2)	(2,3)	(3,1)	(2,4)
1	+1	0	-1	0
2	-1	+1	0	+1
3	0	-1	+1	0
4	0	0	0	-1

b) Matrice d'adjacence sommet-sommet

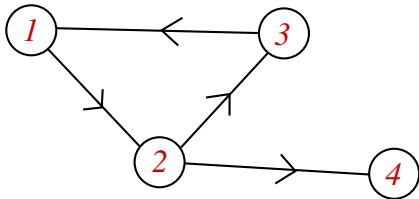
Matrice booléenne A de taille $n \times n$ (n sommets)

$$a_{ij} = \begin{cases} 1 & \text{si l'arête } (i, j) \text{ existe dans le graphe} \\ 0 & \text{sinon} \end{cases}$$

Variante pour un graphe valué par $\{c_{ij}\}$:

$$a_{ij} = \begin{cases} c_{ij} & \text{si l'arête } (i, j) \text{ existe dans le graphe} \\ 0 & \text{sinon} \end{cases}$$

Exemple : $E = \{1, 2, 3, 4\}$
 $\Gamma = \{(1, 2), (2, 3), (3, 1), (2, 4)\}$



Matrice d'adjacence $A =$

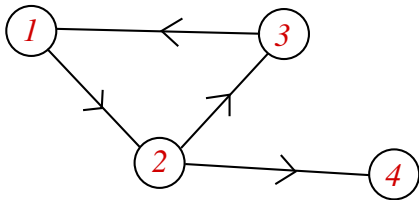
	1	2	3	4
1	0	1	0	0
2	0	0	1	1
3	1	0	0	0
4	0	0	0	0

c) Listes d'adjacence : successeurs et prédécesseurs

Pour chaque sommet i du graphe, on définit

- la liste de ses **successeurs** $S(i)$: liste des sommets j tq l'arête (i, j) existe dans le graphe.
- la liste de ses **prédécesseurs** $P(i)$: liste des sommets j tq l'arête (j, i) existe dans le graphe.

Exemple : $E = \{1, 2, 3, 4\}$
 $\Gamma = \{(1, 2), (2, 3), (3, 1), (2, 4)\}$



sommet	successeur S	prédécesseur P
1	2	3
2	3, 4	1
3	1	2
4	–	2

4) Flot dans un graphe

Problèmes de circulation d'objets (voiture, information ...) dans un réseau (routier, informatique ...).

Définition

Soit $G = (E, \Gamma, c)$ un graphe valué comportant un seul sommet source s et un seul sommet puits t .

- Un **flot** de s à t est une fonction $f : \Gamma \rightarrow \mathbb{R}$ tq

$$\sum_{i \in P(j)} f_{ij} = \sum_{k \in S(j)} f_{jk} \quad \text{où } f_{ij} \stackrel{\text{def}}{=} f(i, j)$$

pour tout sommet $j \neq s, t$. On dit qu'il y a conservation du flux au sommet j ("ce qui rentre égale ce qui sort").

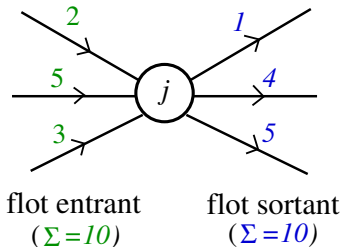
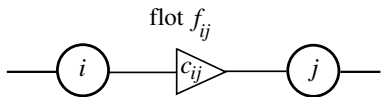
La valeur $f_{ij} \stackrel{\text{def}}{=} f(i, j)$ est le **flot dans l'arête** (i, j) .

Définition (suite)

- Le flot est dit **réalisable** si pour toute arête $(i, j) \in \Gamma$, on a

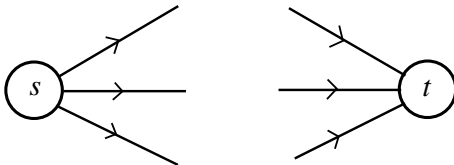
$$0 \leq f_{ij} \leq c_{ij}$$

- La quantité $v = \sum_{i \in P(t)} f_{it}$ est la **valeur du flot** de s à t .



Remarque (rappels) :

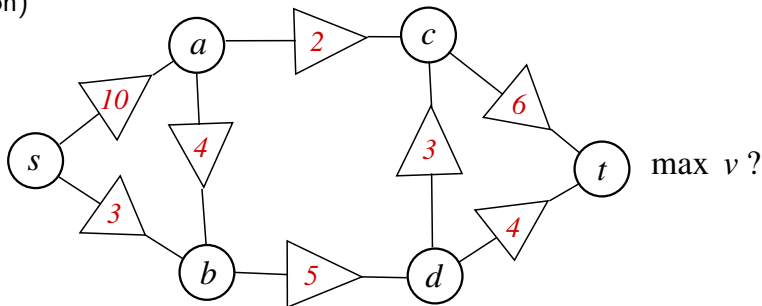
- $S(i)$: ensemble des sommets j **successeurs** du sommet i c-à-d tq l'arête (i, j) existe dans le graphe.
- $P(i)$: ensemble des sommets j **prédécesseurs** du sommet i c-à-d tq l'arête (j, i) existe dans le graphe.
- Une **source** s (resp. un **puits** t) est un sommet ne possédant pas de prédécesseur (resp. de successeur).



II. Problème de flot maximal dans un graphe

1) Introduction

On veut par ex. trouver le trafic maximal entre deux villes d'un réseau routier dont on connaît la capacité (nb de voiture par heure sur chaque tronçon)



Problème de flot maximal

Etant donné un graphe valué possédant une seule source et un seul puits, trouver un flot réalisable maximal (i.e. dont la valeur est maximale).

Flot maximal et programmation linéaire

$$\max_{f_{ij}, v} [F = v]$$

conservation des flux
en chaque sommet :

$$\left\{ \begin{array}{l} \sum_{k \in S(s)} f_{sk} - v = 0 \quad (\text{source } s) \\ - \sum_{i \in P(j)} f_{ij} + \sum_{k \in S(j)} f_{jk} = 0, \quad \forall j \neq s, t \\ - \sum_{i \in P(t)} f_{it} + v = 0 \quad (\text{puits } t) \end{array} \right.$$

respect des capacités : $f_{ij} \leq c_{ij}$ pour toute arête $(i, j) \in \Gamma$

contrainte de signe : $f_{ij} \geq 0$ pour toute arête $(i, j) \in \Gamma$

Remarque : les inconnues sont les f_{ij} et la valeur v du flot.

Flot maximal et programmation linéaire

Ecriture matricielle (n sommets et m arêtes)

$$\begin{aligned} \max_{\mathbf{f}, \mathbf{v}} [F = v] \\ \left\{ \begin{array}{l} A\mathbf{f} + \mathbf{v} = 0 \\ \mathbf{f} \leq \mathbf{c} \\ \mathbf{f} \geq 0 \end{array} \right. \end{aligned}$$

A est la matrice d'incidence du graphe, de taille $n \times m$,

$$\mathbf{f} = \begin{pmatrix} (f_{sk})_{k \in S(s)} \\ \vdots \\ f_{ij} \\ \vdots \\ (f_{it})_{i \in P(t)} \end{pmatrix} \in \mathbb{R}^m; \quad \mathbf{v} = \begin{pmatrix} -v \\ 0 \\ \vdots \\ 0 \\ +v \end{pmatrix} \in \mathbb{R}^n$$

II. Problème de flot maximal dans un graphe

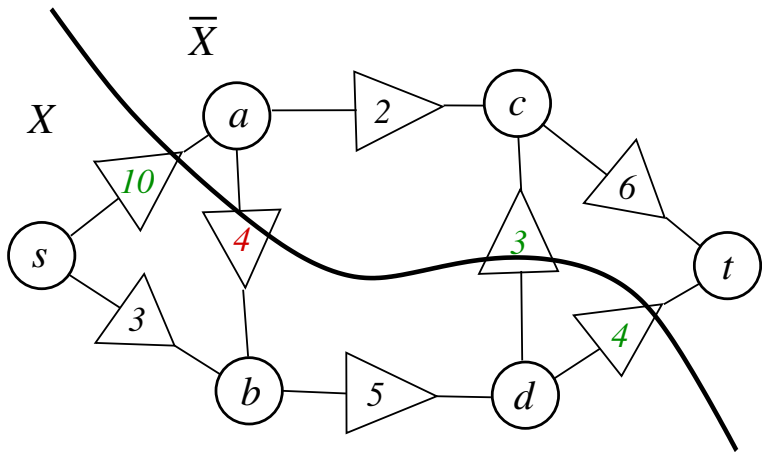
2) Théorème de Ford-Fulkerson

Définition

Une **coupe** d'un graphe valué $G = (E, \Gamma, c)$ possédant un seul sommet source s et un seul sommet puits t , est une partition des sommets notée (X, \overline{X}) telle que :

- $E = X \cup \overline{X}$
- $X \cap \overline{X} = \emptyset$
- $s \in X$ et $t \in \overline{X}$

La **capacité de la coupe** est définie par
$$c(X, \overline{X}) = \sum_{\substack{i \in X \\ j \in \overline{X}}} c_{ij}$$



Capacité de la coupe $c(X, \bar{X}) = 10 + 3 + 4 = 17$.

On peut comparer la valeur d'un flot avec la capacité d'une coupe du graphe.

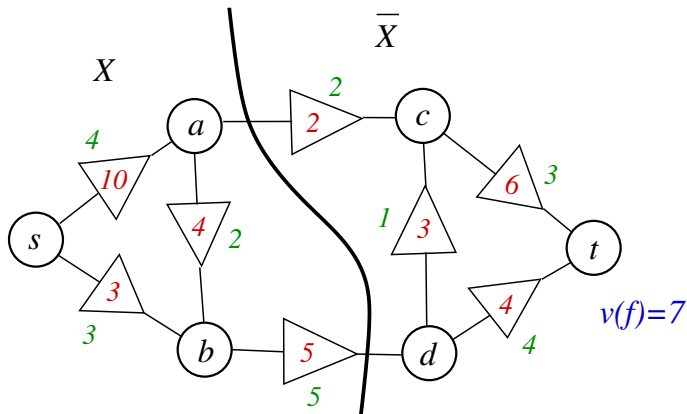
Théorème de Ford-Fulkerson

Soit $G = (E, \Gamma, c)$ un graphe valué. Pour tout flot réalisable f et toute coupe (X, \bar{X}) , on a

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

où $v(f)$ est la valeur du flot f .

Le théorème de Ford-Fulkerson permet de savoir si un flot est maximal ou non. Par exemple :



$v(f) = 7$ et $c(X, \bar{X}) = 7 \Rightarrow$ flot maximal.

Démonstration du théorème de Ford-Fulkerson

Convention : si l'arête (i, j) n'existe pas dans le graphe, on pose $f_{ij} = 0$.

$$\Rightarrow v(f) = \sum_{j \in S(s)} f_{sj} = \sum_{j \in E} f_{sj}.$$

On montre que

$$v(f) = \sum_{\substack{i \in X \\ j \in \bar{X}}} f_{ij} - \sum_{\substack{i \in X \\ k \in \bar{X}}} f_{ki}$$

flot de X à \bar{X} flot de \bar{X} à X

$$\Rightarrow v(f) \leq \sum_{\substack{i \in X \\ j \in \bar{X}}} c_{ij} = c(X, \bar{X})$$



II. Problème de flot maximal dans un graphe

3) Coupe minimale

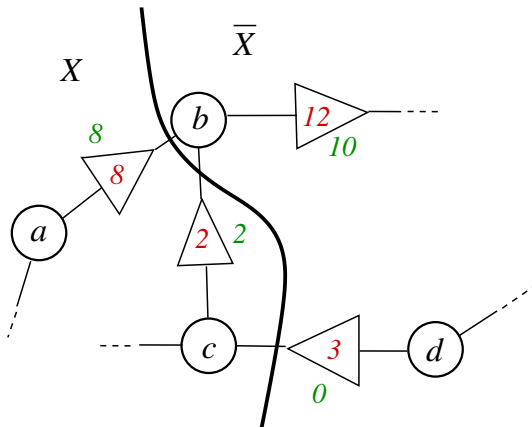
Le Théorème de Ford-Fulkerson admet un corollaire qui donne une condition suffisante pour avoir un flot maximal.

On dit qu'une arête $(i, j) \in \Gamma$ est **saturée** si $f_{ij} = c_{ij}$ et qu'une arête $(j, i) \in \Gamma$ est **insaturée** si $f_{ji} = 0$

Définition

Une coupe (X, \bar{X}) est dite **minimale** pour f si toute arête de X vers \bar{X} est saturée et toute arête de \bar{X} vers X est insaturée.

Coupe minimale (X, \bar{X})



- arêtes (a, b) et (c, b) **saturées**
- arête (d, c) **insaturée**

Proposition

S'il existe une coupe minimale pour un flot f , alors ce flot est maximal.

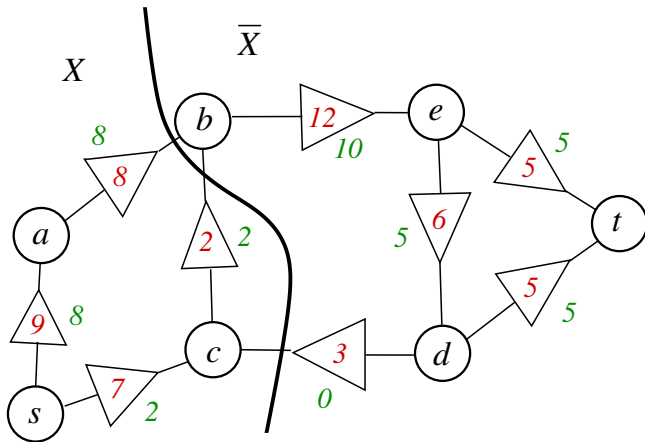
Preuve. A partir de la formule établie dans le th. de Ford-Fulkerson :

$$v(f) = \sum_{\substack{i \in X \\ j \in \bar{X}}} \overbrace{f_{ij}}^{=c_{ij}} - \sum_{\substack{i \in X \\ k \in \bar{X}}} \overbrace{f_{ki}}^{=0} = c(X, \bar{X})$$

$\Rightarrow v(f)$ est maximal.



Coupe minimale / flot maximal



\Rightarrow flot maximal $v(f) = c(X, \bar{X}) = 10$

III. Algorithme de Ford-Fulkerson

1) Condition nécessaire et suffisante de flot maximal

Définition 1.

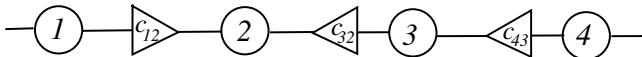
Une chaîne d'un graphe est une suite de sommets

$$C = (i_1, i_2, \dots, i_p, i_{p+1}, \dots, i_q)$$

reliés entre eux par des arêtes c'est-à-dire tels que

$$\begin{array}{ccc} (i_p, i_{p+1}) \in \Gamma & \text{ou} & (i_{p+1}, i_p) \in \Gamma \\ \text{(arête **directe**)} & & \text{(arête **inverse**)} \end{array}$$

Une chaîne ne tient pas compte de l'orientation des arêtes reliant les sommets.

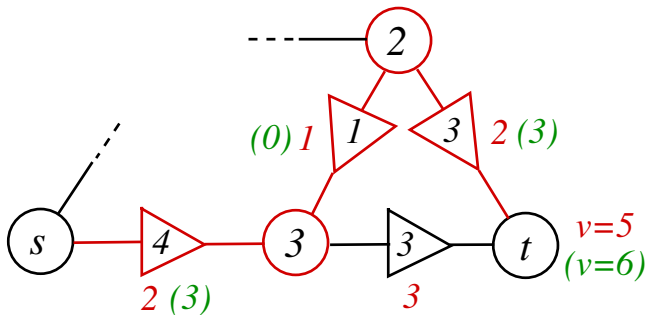


Définition 2.

Soit $G = (E, \Gamma, c)$ un graphe valué possédant une seule source s et un seul puits t . Une chaîne $\mathcal{C} = (s, i_1, i_2, \dots, i_p, i_{p+1}, \dots, i_q, t)$ est dite **améliorante** pour un flot réalisable f donné si :

- $f(i_p, i_{p+1}) < c(i_p, i_{p+1})$ si $(i_p, i_{p+1}) \in \Gamma$ (arête directe)
- $f(i_{p+1}, i_p) > 0$ si $(i_{p+1}, i_p) \in \Gamma$ (arête inverse)

Remarque : ce qui compte ici, ce sont les inégalités strictes.



L'algorithme de Ford-Fulkerson permet de trouver un flot maximal par recherche de chaînes améliorantes. Il est basé sur le résultat suivant :

Théorème

Un flot réalisable est maximal si et seulement s'il n'existe pas de chaîne améliorante.

Preuve.

i) Condition nécessaire

Soit f un flot réalisable maximal. On suppose par l'absurde qu'il existe une chaîne améliorante $\mathcal{C} = (s, i_1, i_2, \dots, i_p, i_{p+1}, \dots, i_q, t)$. On note

$$\varepsilon_1 = \min\{c(i_p, i_{p+1}) - f(i_p, i_{p+1}) \text{ tel que } (i_p, i_{p+1}) \in \Gamma \text{ (arête directe)}\}$$

$$\varepsilon_2 = \min\{f(i_{p+1}, i_p) \text{ tel que } (i_{p+1}, i_p) \in \Gamma \text{ (arête inverse)}\}$$

$$\rightarrow \boxed{\varepsilon = \min\{\varepsilon_1, \varepsilon_2\} > 0}$$

ε représente l'amélioration qu'on peut apporter au flot.

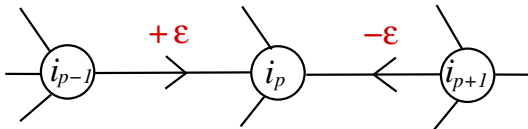
Nouveau flot f' qui coïncide avec f en dehors de la chaîne améliorante.
Sur les arêtes de la chaîne :

- Si $(i_p, i_{p+1}) \in \Gamma$ (arête directe) alors

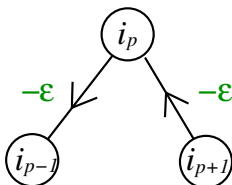
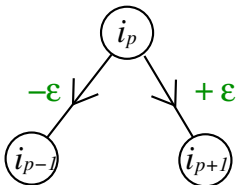
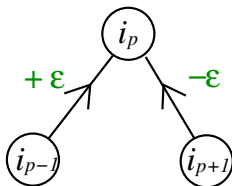
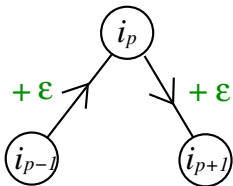
$$f'(i_p, i_{p+1}) = f(i_p, i_{p+1}) + \varepsilon$$

- Si $(i_{p+1}, i_p) \in \Gamma$ (arête inverse) alors

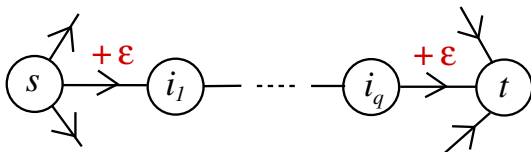
$$f'(i_p, i_{p+1}) = f(i_p, i_{p+1}) - \varepsilon$$



- Le nouveau flot f' est bien **réalisable**. En particulier, il y a conservation des flux en chaque sommet : 4 cas possibles



- Le nouveau flot f' est augmenté de $+\varepsilon$ quand il arrive au puits t :
 $v(f') = v(f) + \varepsilon$.



\Rightarrow le flot f n'est pas maximal \Rightarrow contradiction.



ii) Condition suffisante

On suppose qu'il n'existe pas de chaîne améliorante. On va montrer que le flot est maximal en trouvant une coupe (X, \overline{X}) telles que $v(f) = c(X, \overline{X})$ (Th. Ford-Fulkerson).

Construction de la coupe (X, \overline{X})

X est l'ensemble des sommets qui sont **marqués** de la façon suivante :

- 1 On marque la source s
- 2 A partir de tous les sommets i marqués, marquer tous les sommets j non encore marqués tels que

$$f(i, j) < c(i, j) \text{ (arête directe)} \quad \text{ou} \quad f(j, i) > 0 \text{ (arête inverse)}$$

- 3 Recommencer en 2) jusqu'à ce qu'il n'y ait plus de marquage possible.

\Rightarrow A l'issue du marquage, on a $v(f) = c(X, \overline{X})$.



Remarque : le puits t ne peut pas être marqué sinon il y aurait une chaîne améliorante.

III. Algorithme de Ford-Fulkerson

2) Algorithme de Ford-Fulkerson

- Initialisation par un flot initial réalisable ($f = 0$)
- Tant que le flot n'est pas maximal
 - Marquage de la source s
 - Tant qu'on marque des sommets
 - Pour tout sommet marqué i
 - Marquer les sommets j non marqués tq
 $f(i,j) < c(i,j)$ ou $f(j,i) > 0$
 - Fin pour
 - Fin Tant que
- Si le puits t n'est pas marqué alors le flot est maximal
 Sinon amélioration du flot

Fin Tant que

III. Algorithme de Ford-Fulkerson

Amélioration du flot

- trouver une chaîne qui a permis de marquer t et calculer $\varepsilon = \min(\varepsilon_1, \varepsilon_2) > 0$ avec

$$\varepsilon_1 = \min \{c(i_p, i_{p+1}) - f(i_p, i_{p+1}) \text{ avec } (i_p, i_{p+1}) \in \Gamma \text{ (arête directe)}\}$$

$$\varepsilon_2 = \min \{f(i_{p+1}, i_p) \text{ avec } (i_{p+1}, i_p) \in \Gamma \text{ (arête inverse)}\}$$

- trouver le nouveau flot f' :
 - Si $(i_p, i_{p+1}) \in \Gamma$ (arête directe) alors $f'(i_p, i_{p+1}) = f(i_p, i_{p+1}) + \varepsilon$
 - Si $(i_{p+1}, i_p) \in \Gamma$ (arête inverse) alors $f'(i_p, i_{p+1}) = f(i_p, i_{p+1}) - \varepsilon$

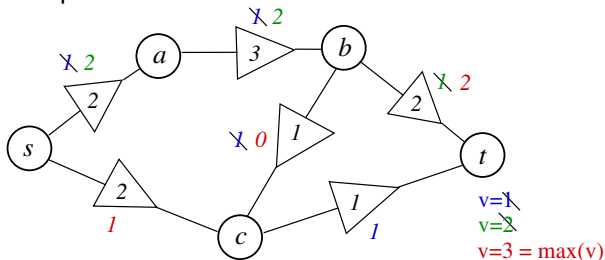
III. Algorithme de Ford-Fulkerson

Remarque pratique

Dans la pratique, on utilise plusieurs tableaux

- $\mathbb{E} = \{\text{sommets marqués mais non complètement examinés}\}$
- tableau `orig` qui indique l'origine du marquage :
 - arête directe $(i_p, i_{p+1}) \in \Gamma \rightarrow \text{orig}(i_{p+1}) = i_p$
 - arête inverse $(i_{p+1}, i_p) \in \Gamma \rightarrow \text{orig}(i_p) = -i_{p+1}$
- tableau ε : amélioration possible du flot pour chaque sommet de \mathbb{E} .

Exemple. Marquage pile profondeur : on explore le sommet le plus éloigné du sommet de départ.



Etape 1 :

\mathbb{E}	s	a	b	c	t
orig	—	s	a	b	c
ε	∞	2	2	1	$\varepsilon = 1$

Etape 2

\mathbb{E}	s	a	b	t
orig	—	s	a	b
ε	∞	1	1	$\varepsilon = 1$

Etape 3

\mathbb{E}	s	c	b	a	t
orig	—	s	$-c$	$-b$	b
ε	∞	2	1	1	$\varepsilon = 1$

III. Algorithme de Ford-Fulkerson

3) Complexité

n sommets et m arêtes.

- $\mathcal{O}(n)$ opérations pour l'amélioration du flot.
- $\mathcal{O}(m)$ opérations pour la recherche d'une chaîne améliorante.

⇒ $\mathcal{O}(nm)$ opérations pour l'algorithme de Ford-Fulkerson.

Remarque : en fait, le coût est en $\mathcal{O}(C_{\max}nm)$ où C_{\max} est le maximum des capacités des arêtes. En effet la capacité d'une coupe est au plus $C_{\max} \times (n - 1)$ et dans le pire des cas le flot augmente d'une seule unité à chaque fois. Il y a donc au plus $C_{\max}n$ améliorations.