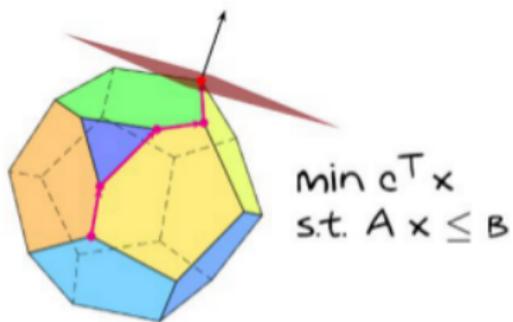


Flots

- Réseaux et Flots



Réseaux et flots



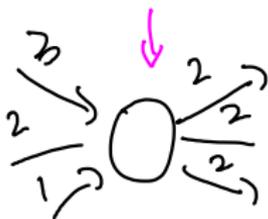
Réseau: Graphe orienté simple $D = (V, A)$ et fonction de capacité $u : A \rightarrow \mathbb{R}_{\geq 0}$.



$f : A \rightarrow \mathbb{R}_{\geq 0}$ est appelée **flot de s à t** , si

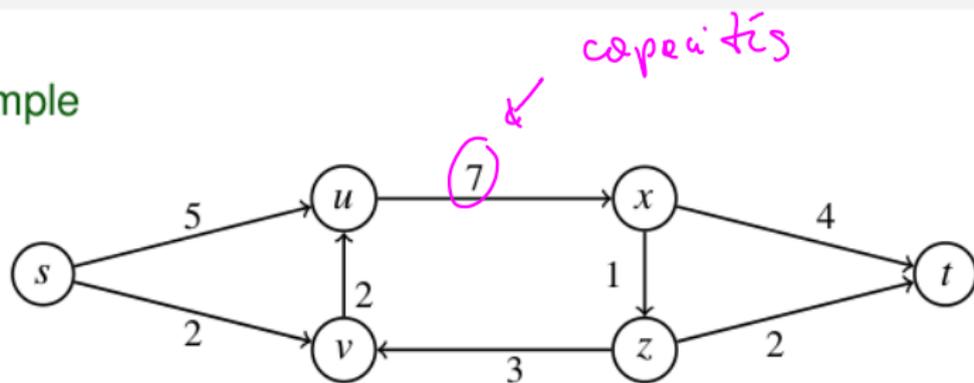
$$\sum_{e \in \delta^{out}(v)} f(e) = \sum_{e \in \delta^{in}(v)} f(e), \text{ pour tout } v \in V - \{s, t\}, \quad (1)$$

où $s, t \in V$. Le flot est admissible, si $f(e) \leq u(e)$ pour tout $e \in A$.



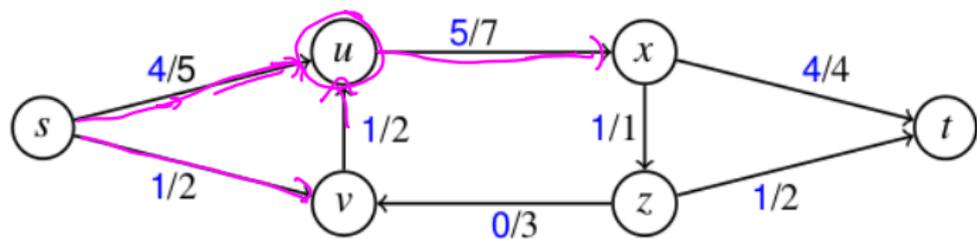
Réseaux et flots

Exemple



Réseaux et flots

Exemple : Réseau et flot de s à t



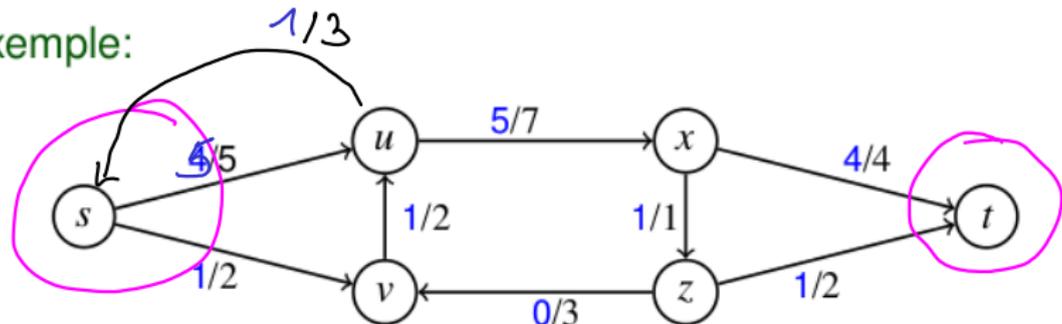
Valeur du flot

La *valeur* de f est définie comme

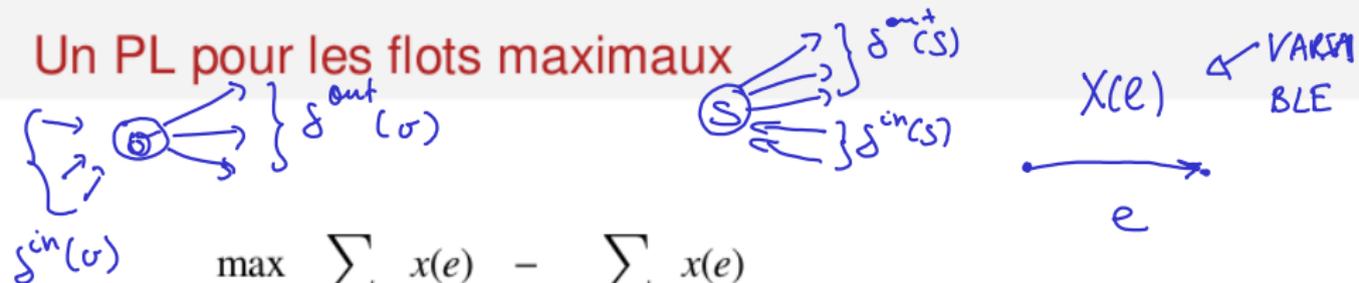
$$\text{value}(f) = \sum_{e \in \delta^{\text{out}}(s)} f(e) - \sum_{e \in \delta^{\text{in}}(s)} f(e).$$

Le *problème du flot maximal de s à t* consiste à déterminer un flot valeur maximal de s à t qui est admissible.

Exemple:



Un PL pour les flots maximaux



$$\max \sum_{e \in \delta^{out}(s)} x(e) - \sum_{e \in \delta^{in}(s)} x(e)$$

$$\sum_{e \in \delta^{out}(v)} x(e) = \sum_{e \in \delta^{in}(v)} x(e), \text{ pour tout } v \in V - \{s, t\}$$

$$x(e) \leq u(e), \text{ pour tout } e \in A$$

$$x(e) \geq 0, \text{ pour tout } e \in A$$

Capacités sont respectées

Remarque

Si le PL est borné et les capacités sont des nombres entiers, le PL a une solution optimale intégrale, lorsque la *matrice d'incidence sommets-arcs d'un graphe orienté* est totalement unimodulaire.

Un PL pour les flots maximaux

σ



arrêtes sortant.

$|A| \times |V|$
 $\{Bx = 0\}$



$$\max \sum_{e \in \delta^{out}(s)} x(e) - \sum_{e \in \delta^{in}(s)} x(e)$$

$$\sum_{e \in \delta^{out}(v)} x(e) = \sum_{e \in \delta^{in}(v)} x(e), \text{ pour tout } v \in V - \{s, t\}$$

$$x(e) \leq u(e), \text{ pour tout } e \in A$$

$$x(e) \geq 0, \text{ pour tout } e \in A$$



Remarque

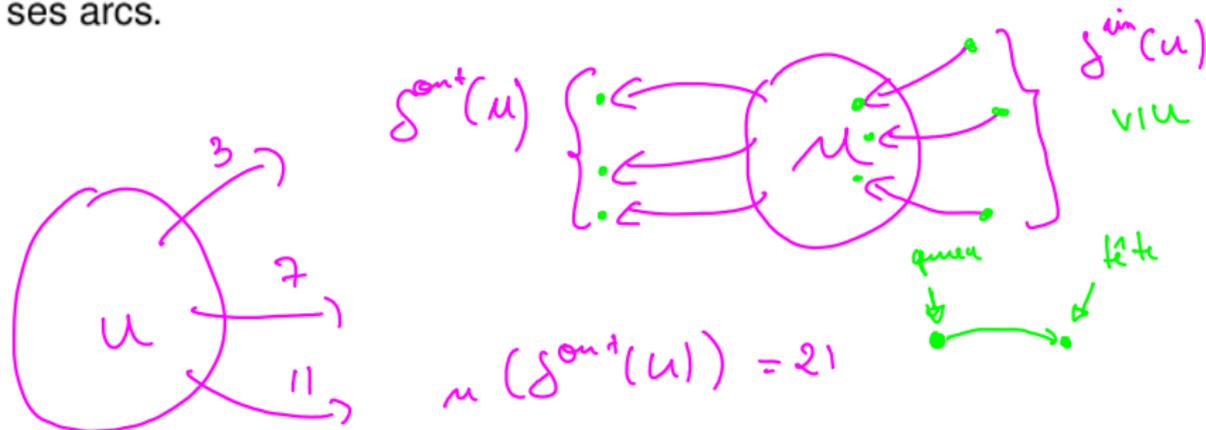
Si le PL est borné et les capacités sont des nombres entiers, le PL a une solution optimale intégrale, lorsque la *matrice d'incidence sommets-arcs d'un graphe orienté* est totale-ment unimodulaire.

Coupes

Pour $U \subseteq V$, $\delta^{in}(U)$ dénote les arcs qui entrent dans U et $\delta^{out}(U)$ dénote les arcs sortant de U .

Des ensembles d'arcs de la forme $\delta^{out}(U)$ sont appelés une coupe de $D = (V, A)$ $u: A \rightarrow \mathbb{R}_+$

La capacité d'une coupe $u(\delta^{out}(U))$ est la somme des capacités de ses arcs.



Coupes

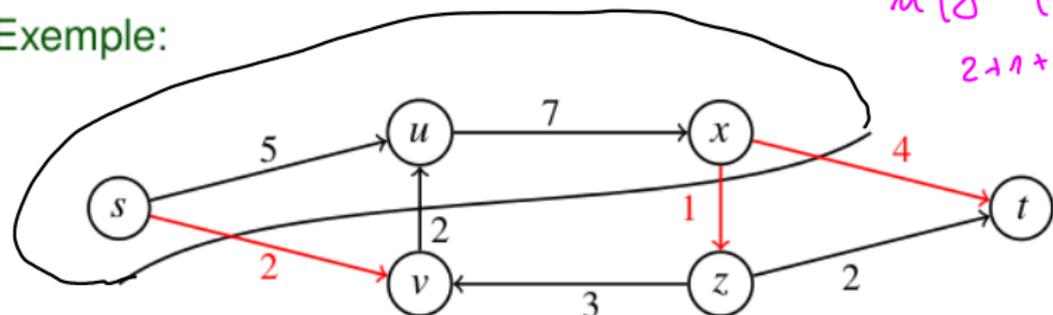
Pour $U \subseteq V$, $\delta^{in}(U)$ dénote les arcs qui entrent dans U et $\delta^{out}(U)$ dénote les arcs sortant de U .

Des ensembles d'arcs de la forme $\delta^{out}(U)$ sont appelés une *coupe* de D .

Si $s \in U$ et $t \notin U$, $\delta^{out}(U)$ est une coupe $s-t$

La *capacité d'une coupe* $u(\delta^{out}(U))$ est la somme des capacités de ses arcs.

Exemple:

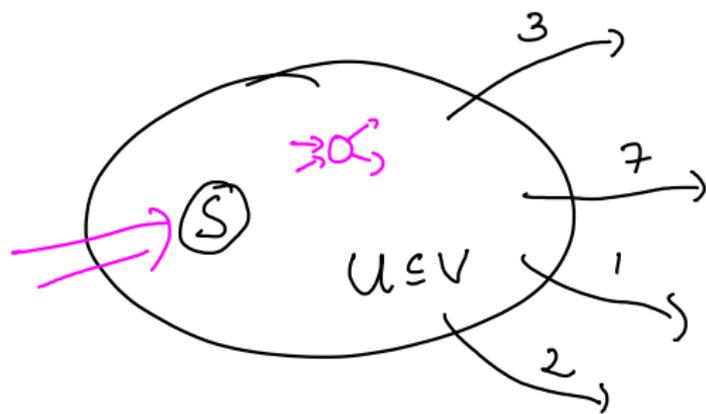


$$u(\delta^{out}(U)) = 2 + 1 + 4 = 7$$

Fonction d'excès

Pour tout $f : A \rightarrow \mathbb{R}$, la fonction d'excès; $exces_f : 2^V \rightarrow \mathbb{R}$ est définie par

$$exces_f(U) = \sum_{e \in \delta^{in}(U)} f(e) - \sum_{e \in \delta^{out}(U)} f(e).$$



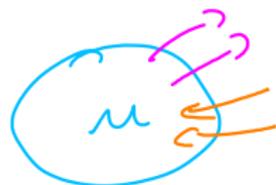
Idea: Flot
max. est ≤ 13

But: Confirmer notre intuition.

Fonction d'excès

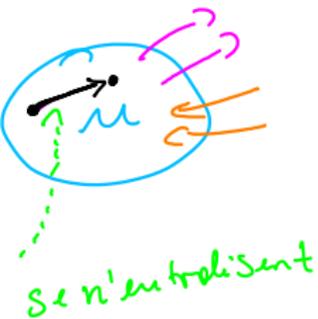
Pour tout $f : A \rightarrow \mathbb{R}$, *la fonction d'excès*; $exces_f : 2^V \rightarrow \mathbb{R}$ est définie par

$$exces_f(U) = \sum_{e \in \delta^{in}(U)} f(e) - \sum_{e \in \delta^{out}(U)} f(e).$$



Fonction d'excès

Soit $D = (V, A)$ un graphe orienté, soit $f : A \rightarrow \mathbb{R}$ et soit $U \subseteq V$,
alors



$$\begin{aligned} \text{exces}_f(U) &= \sum_{v \in U} \text{exces}_f(v) = \sum_{e \in \mathcal{E}^{\text{in}}(U)} f(e) - \sum_{e \in \mathcal{E}^{\text{out}}(U)} f(e) \\ &= \text{exces}_f(U) \end{aligned}$$

(Note: The diagram includes a green arrow pointing from the term $f(e)$ in the second sum to the term $f(e)$ in the first sum, and a green arrow pointing from the term $f(e)$ in the first sum to the term $f(e)$ in the final result.)

Dualité faible

Dualité faible

Soit f un flot admissible de s à t et soit $\delta^{out}(U)$ une coupe de $s \rightarrow t$,
alors $value(f) \leq u(\delta^{out}(U))$.

$$\begin{aligned}value(f) &= -\text{exces}(f) \\ &= \sum_{v \in U} -\text{exces}(v)\end{aligned}$$

$$= -\text{exces}(U) = \sum_{e \in \delta^{out}(U)} f(e) - \sum_{e \in \delta^{in}(U)} f(e)$$

$$\leq \sum_{e \in \delta^{out}(U)} f(e) \leq u(\delta^{out}(U))$$

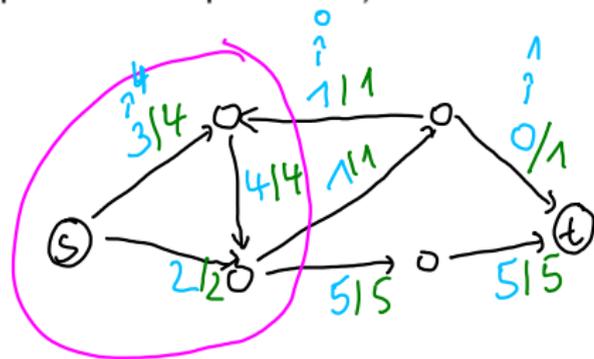


Le Graphe résiduel

Soit $f : A \rightarrow \mathbb{R}$ et $u : A \rightarrow \mathbb{R}$ où $0 \leq f \leq u$. Considérons l'ensemble d'arcs

$$A_f = \{a \mid a \in A, f(a) < u(a)\} \cup \{a^{-1} \mid a \in A, f(a) > 0\}.$$

Le graphe orienté $D(f) = (V, A_f)$ est appelé graphe résiduel de f (pour des capacités u).



Capacité = 6
nouvelle valeur
= 6
= 1 opt!

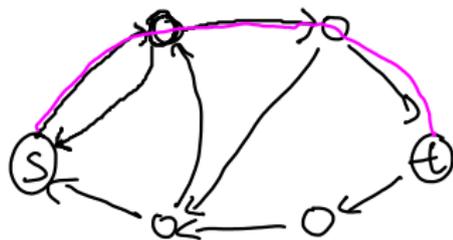
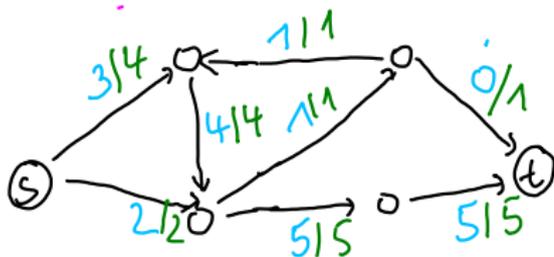
Le Graphe résiduel



Soit $f : A \rightarrow \mathbb{R}$ et $u : A \rightarrow \mathbb{R}$ où $0 \leq f \leq u$. Considérons l'ensemble d'arcs

$$A_f = \{a \mid a \in A, f(a) < u(a)\} \cup \{a^{-1} \mid a \in A, f(a) > 0\}.$$

Le graphe orienté $D(f) = (V, A_f)$ est appelé **graphe résiduel** de f (pour des capacités u).
Chemin orienté de s à t !



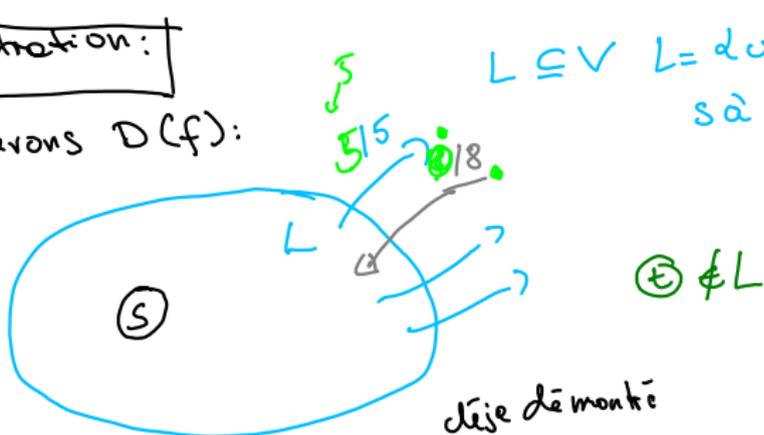
→
Graphe résiduel !

Théorème

Soit f un flot admissible de s à t et supposons qu'il n'y a pas de chemin de s à t dans $D(f)$, alors f est de valeur maximale.

Démonstration:

Considérons $D(f)$:



$$u(D^{\text{out}}(L)) \stackrel{d}{=} -\text{exces}_f(L) \stackrel{d}{=} \text{exces}(s) = \text{valeur de } f.$$



Marches non-orientées

Une *marche non-orientée* est une séquence de la forme $P = (v_0, v_1, \dots, v_{m-1}, v_m)$, où $(v_{i-1}, v_i) \in A$ ou $(v_i, v_{i-1}) \in A$. Si les sommets v_0, \dots, v_m sont tous différents, alors P est un *chemin non-orienté*.

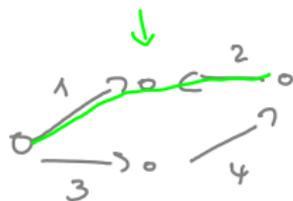


$D(f)$ et marches non-orientées

Tout chemin orienté P dans $D(f)$ induit un chemin non-orienté dans D . Définissons pour un tel chemin P le vecteur $\chi^P \in \{0, \pm 1\}^A$ comme

$$\chi^P(a) = \begin{cases} 1 & \text{si } P \text{ traverse } a, \\ -1 & \text{si } P \text{ traverse } a^{-1}, \\ 0 & \text{si } P \text{ ne traverse ni } a \text{ ni } a^{-1}. \end{cases} \quad (2)$$

chemin non-orienté.



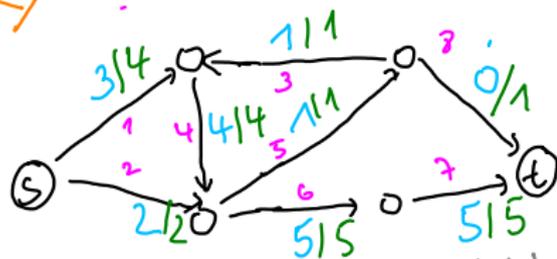
$$\chi^P(a) = \begin{pmatrix} 1 \\ -1 \\ 0 \\ 0 \end{pmatrix}$$

$D(f)$ et marches non-orientées

Tout chemin orienté P dans $D(f)$ induit un chemin non-orienté dans D . Définissons pour un tel chemin P le vecteur $\chi^P \in \{0, \pm 1\}^A$ comme

$$\chi^P(a) = \begin{cases} 1 & \text{si } P \text{ traverse } a, \\ -1 & \text{si } P \text{ traverse } a^{-1}, \\ 0 & \text{si } P \text{ ne traverse ni } a \text{ ni } a^{-1} \end{cases} \quad (2)$$

*nouveau flux !
f + χ^P !!*

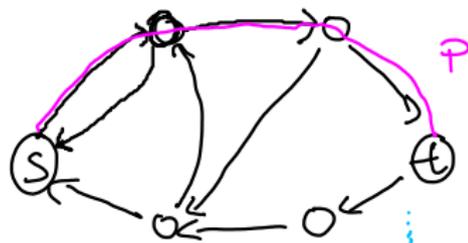


$$\chi^P = \begin{pmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$f = \begin{pmatrix} 3 \\ 2 \\ 1 \\ 4 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 5 \\ 5 \\ 0 \end{pmatrix}$$

chemin orienté de s à t !!



L'algorithme de Ford et Fulkerson

L'algorithme de Ford et Fulkerson

Commencer avec $f = 0$. Puis appliquer itérativement *l'algorithme d'augmentation de flot* suivant:

Soit P un chemin orienté de s à t dans $D(f)$. Soit $f \leftarrow f + \epsilon x^P$, où ϵ est aussi grand que possible tout en gardant $0 \leq f \leq u$.

Quiz

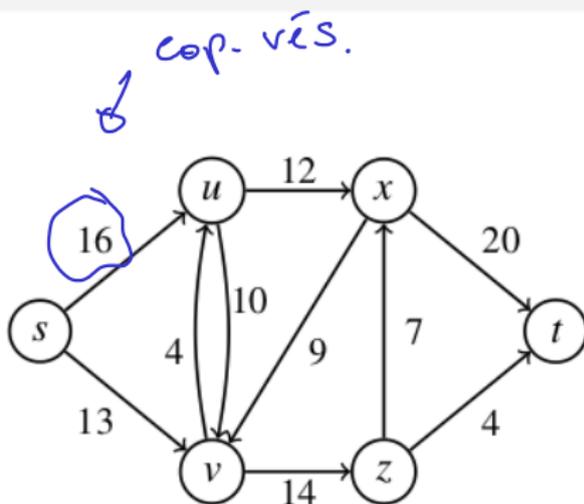
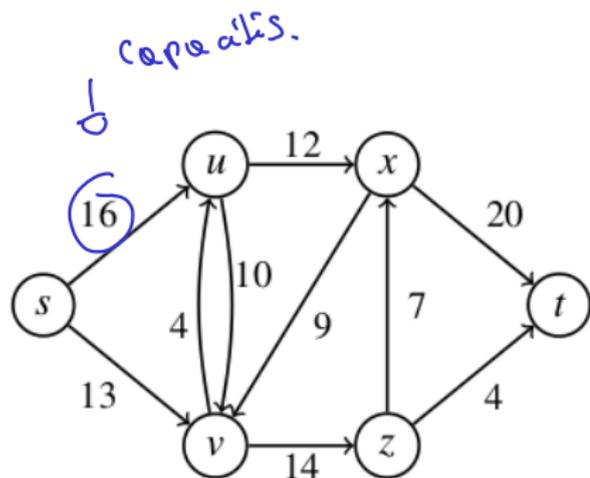
Définissez une *capacité résiduelle* pour $D(f)$. Déterminez ensuite la valeur maximale de ϵ telle que $0 \leq f \leq u$.

Supposons:

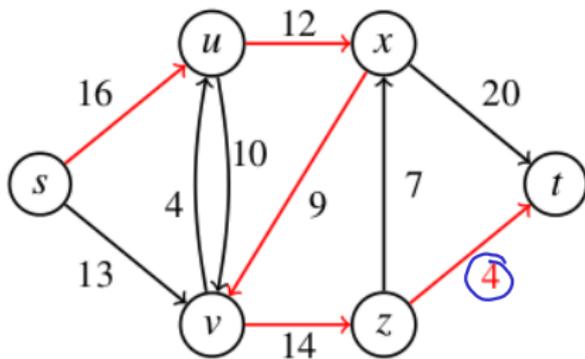
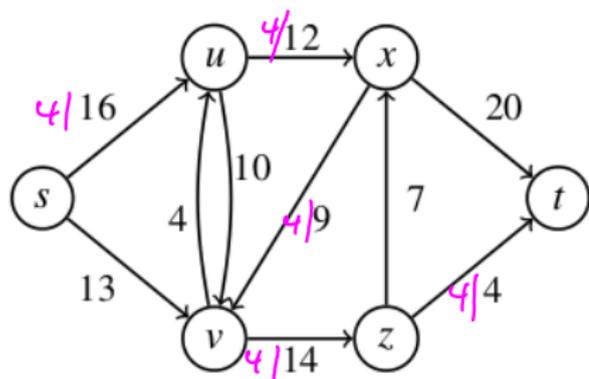
D n'a pas



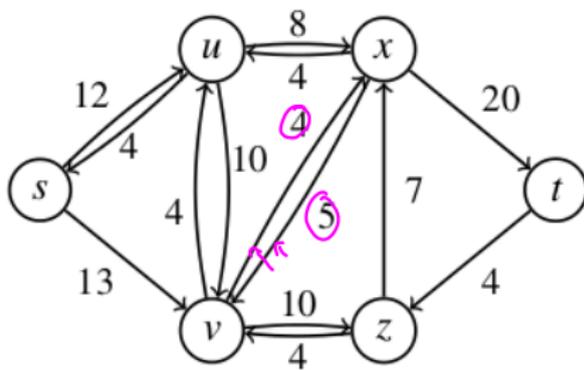
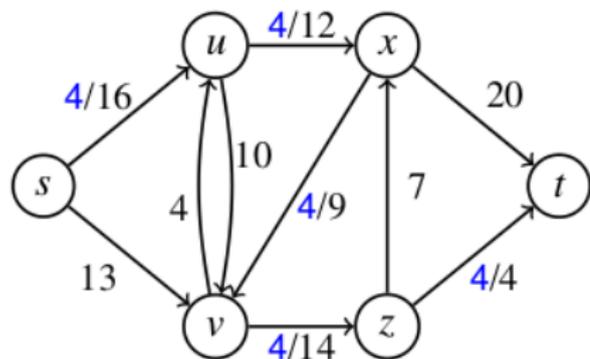
Algorithme de Ford-Fulkerson : exemple



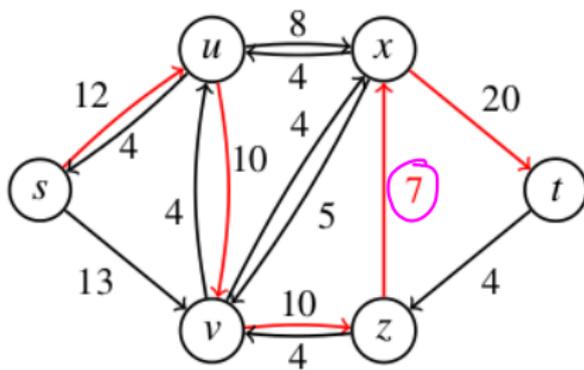
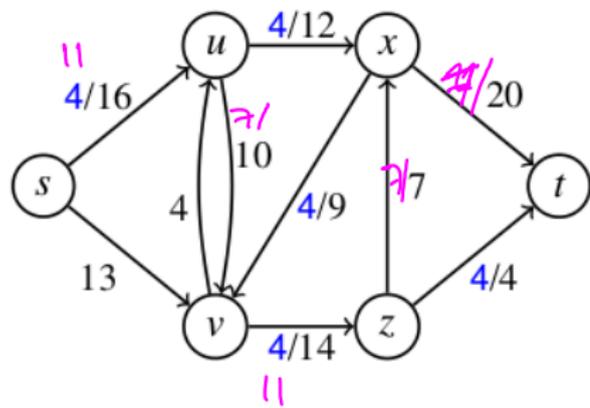
Algorithme de Ford-Fulkerson : exemple



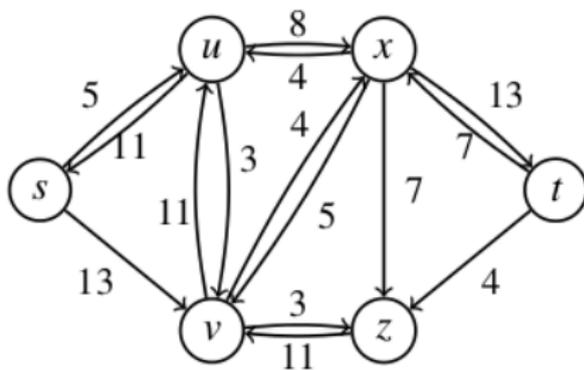
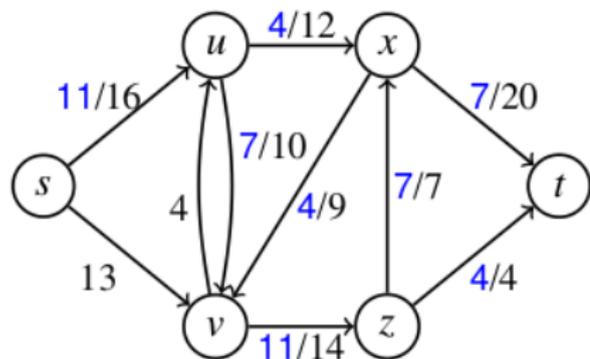
Algorithme de Ford-Fulkerson : exemple



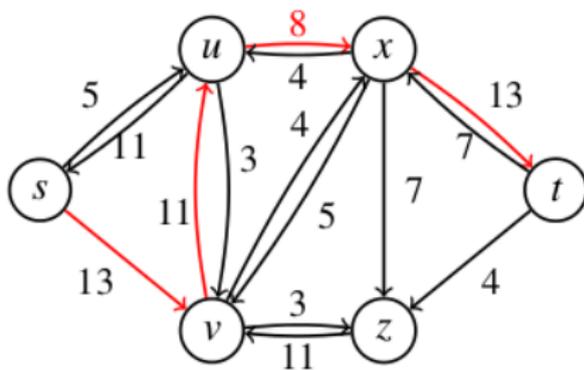
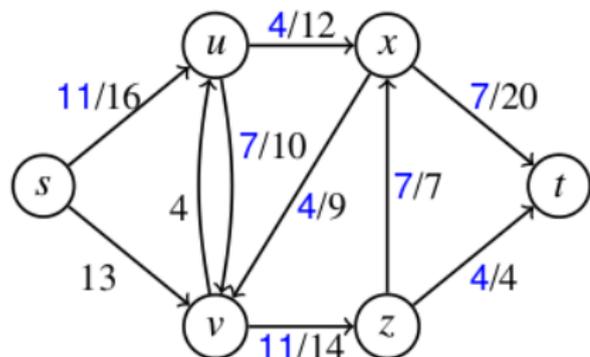
Algorithme de Ford-Fulkerson : exemple



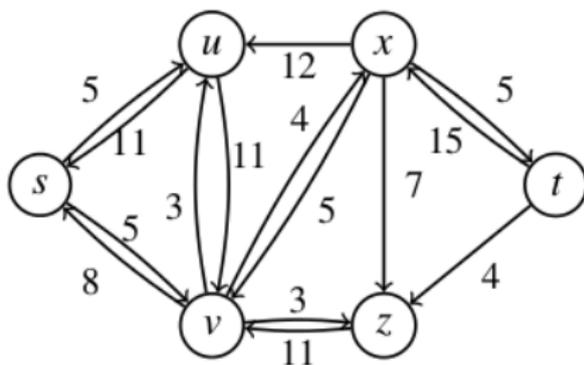
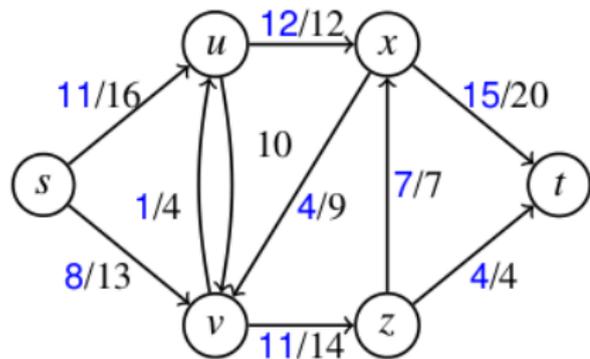
Algorithme de Ford-Fulkerson : exemple



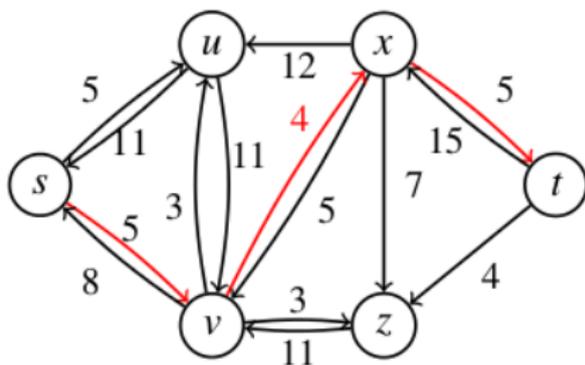
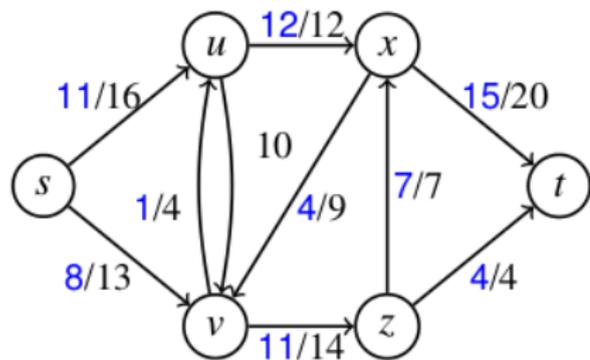
Algorithme de Ford-Fulkerson : exemple



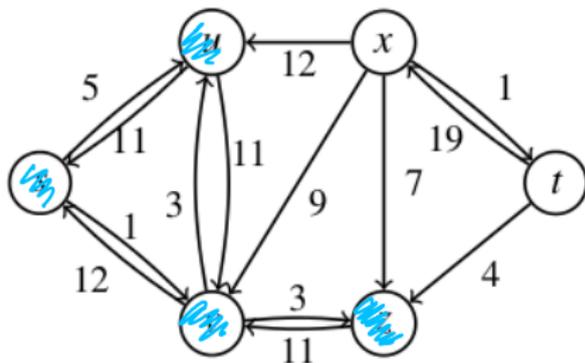
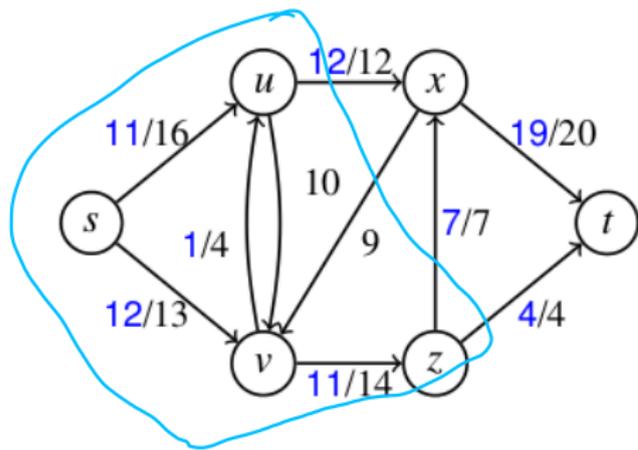
Algorithme de Ford-Fulkerson : exemple



Algorithme de Ford-Fulkerson : exemple



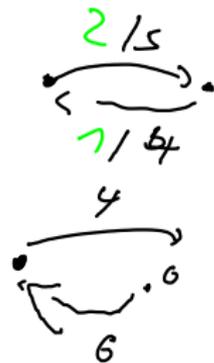
Algorithme de Ford-Fulkerson : exemple



$$\text{cap}(s^{\text{out}}(U)) = 23$$

$$f_{\text{tot}} = 23$$

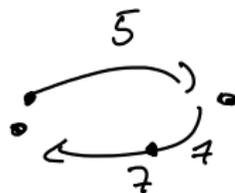
L



Dualité forte

Theorem (Théorème du flot max et de la coupe min, dualité forte)

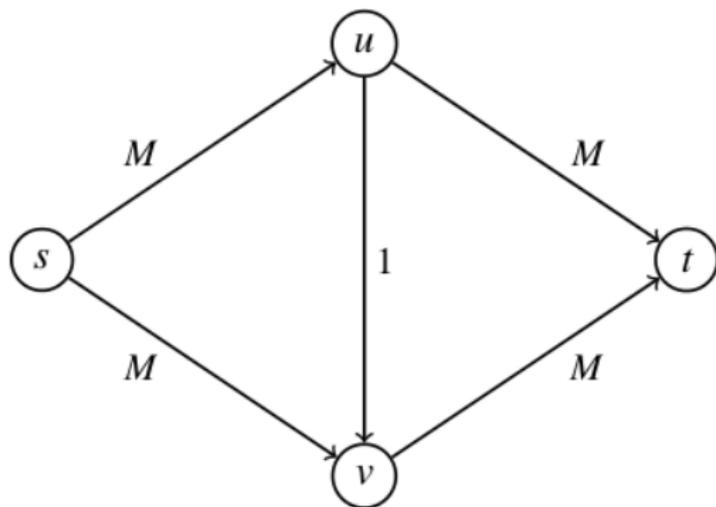
La valeur maximale d'un flot admissible de s à t est égale à la capacité d'une coupe minimale de $s - t$.



Corollary (Théorème d'intégralité)

Si $u(a) \in \mathbb{N}$ pour tout $a \in A$, alors il existe un flot maximal entier ($f(a) \in \mathbb{N}$ pour tout $a \in A$).

L'algorithme de Ford-Fulkerson s'exécute-t-il en temps polynômial?



Lemme

Soient $D = (V, A)$ un graphe orienté, $s, t \in V$ et $\mu(D)$ la longueur d'un plus court chemin de s à t . Soit $a(D)$ l'ensemble d'arcs contenu dans au moins un des plus courts chemins de s à t .

Lemme

Soit $D = (V, A)$ un graphe orienté et $s, t \in V$. Définissons $D' = (V, A \cup a(D)^{-1})$. Alors $\mu(D) = \mu(D')$ et $a(D) = a(D')$.

Algorithme en temps polynômial

Théorème

Si à chaque itération nous choisissons un plus court chemin de s à t dans $D(f)$ comme chemin d'augmentation de flot, le nombre d'itérations est au plus $|V| \cdot |A|$.

Algorithme en temps polynômial

Théorème

Si à chaque itération nous choisissons un plus court chemin de s à t dans $D(f)$ comme chemin d'augmentation de flot, le nombre d'itérations est au plus $|V| \cdot |A|$.