Graphe d'écart $G_e = (X_e, U_e)$

- On prend les mêmes sommets : $X_e = X$
- Pour tout arc $ij \in U$ de capacité c_{ij} sur lequel circule un flux f_{ij} :

  Si $f_{ij} < c_{ij}$ Alors

  on crée un arc ij dans $Ge(f)$ de capacité $c_{ij} - f_{ij}$

  Fin Si

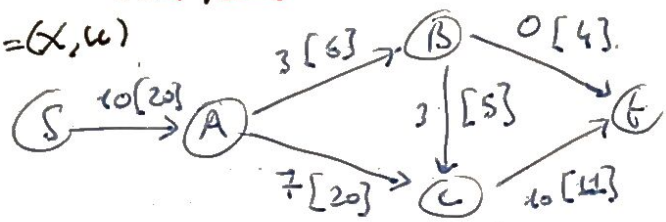
  Si $f_{ij} > 0$ Alors

  on crée un arc ji dans $Ge(f)$ de capacité f_{ij}

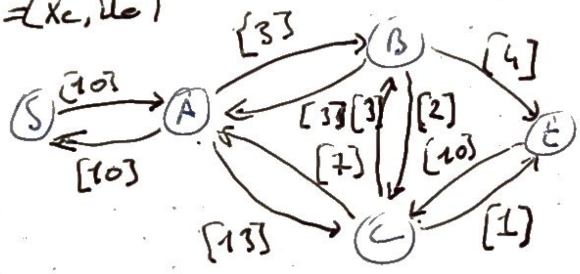
  Fin Si

Exemple

$G = (X, U)$



$Ge = (X_e, U_e)$



Dans le graphe Ge, la somme des capacités de 2 arcs opposés est égale à la capacité de l'arc dans G.

Pour un graphe où le flot est nul ($\forall ij \in U, f_{ij} = 0$) alors on a $Ge = G$.

Une st-chaîne dans G correspond à un st-chemin dans Ge

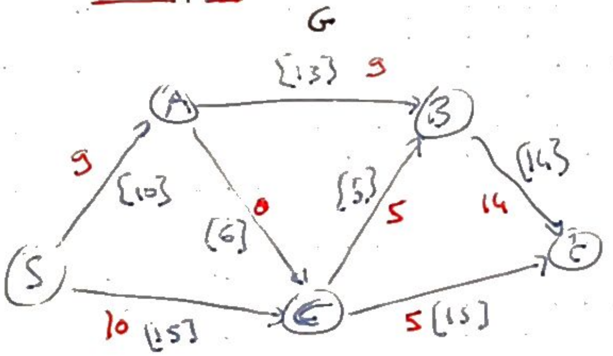
	G	Ge
si $f_{ij} < c_{ij}$	$i \to j$	$i \to j$
si $f_{ij} > 0$	$i \to j$	$i \leftarrow j$

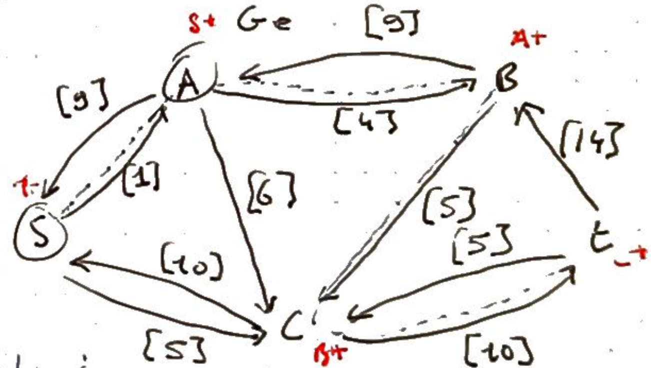
Une conséquence est l'algorithme de Roy.

* Trouver un flot compatible f
* Construire le graphe d'écart $Ge(f)$
* Tant qu'il existe un st-chemin augmentant dans $Ge(f)$, modifier f et $Ge(f)$ en utilisant ce chemin.

FTQ.

Exemple





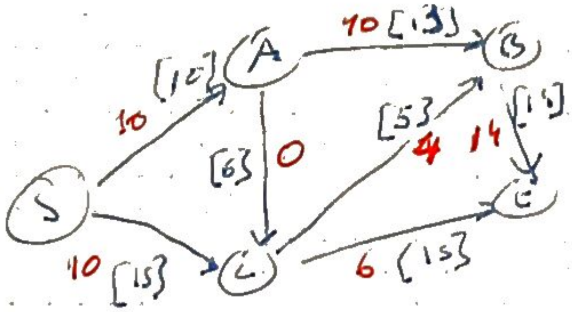
chemin augmentant :

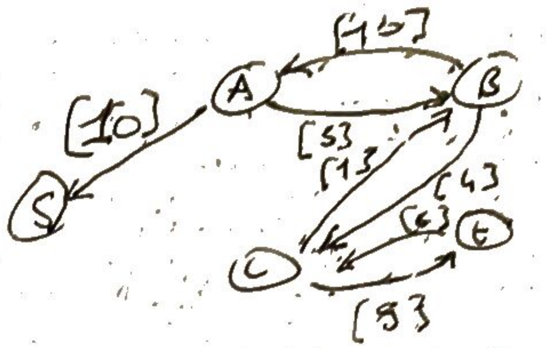
$S \longrightarrow A \longrightarrow B \longrightarrow C \longrightarrow t$

1 4 5 10 ← marge

min = 1 on enlève 1 à tous les arcs → du chemin augmentant et on rajoute 1 à l'arc opposé.

On change en conséquence le graphe G





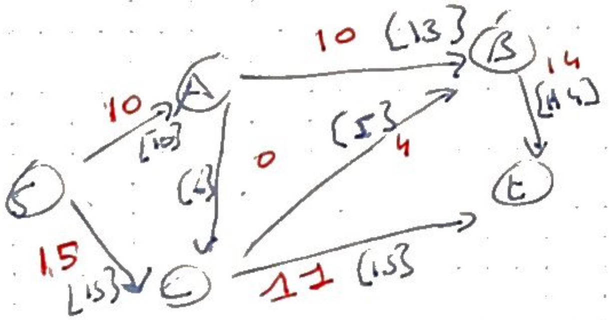
Puis, on recherche un chemin augmentant dans Gc

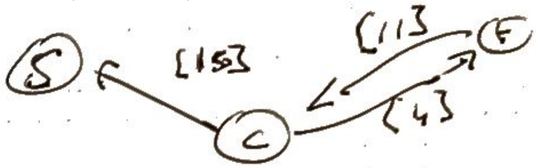
par exemple $S \xrightarrow[5]{} C \xrightarrow[10]{} E$ min = 5

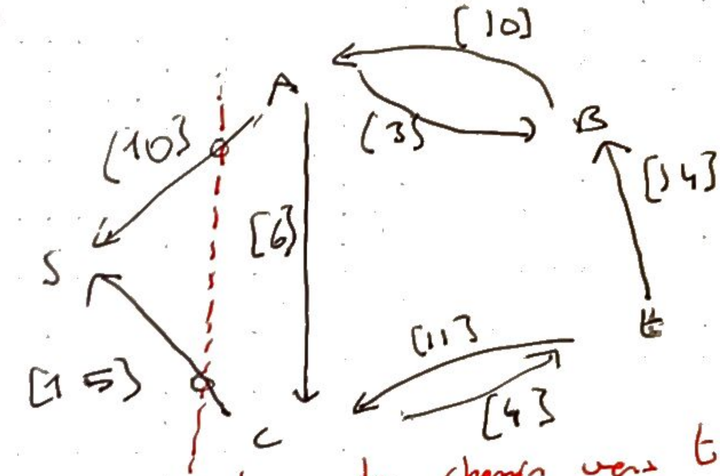
marge

on enlève 5 au flèche → et

on rajoute 5 au arc opposé







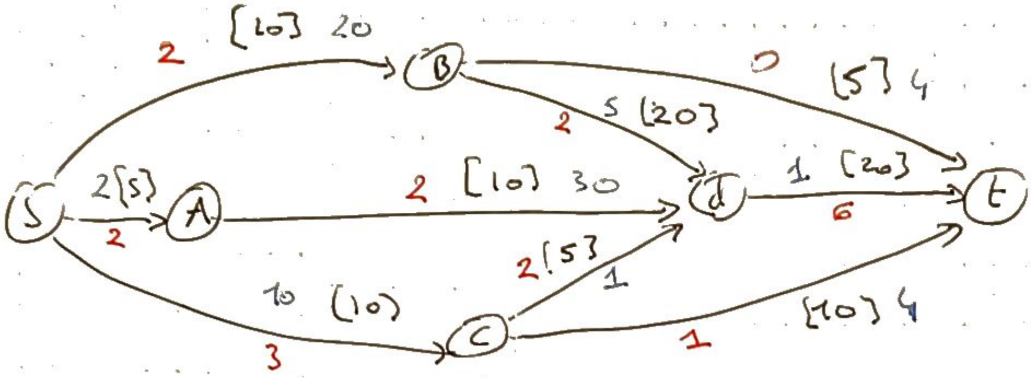
plus de chemin vers t.

⇒ flot maximal

de 25

On se pose le problème de trouver un flot maximal de coût minimal.

On considère que le coût d'un flot est la somme des coûts d'utilisation de chaque arc.



$v(f) = 7$.

en noir, les capacités [ ]

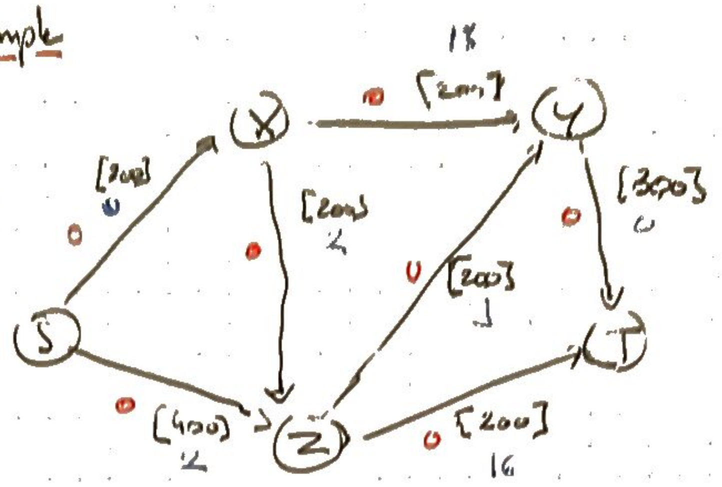
en bleu, les couts d'utilisation (peut représenter une distance, un prix, ...)

en rouge un flot.

On veut maximiser $v(f)$ en minimisant le coût.

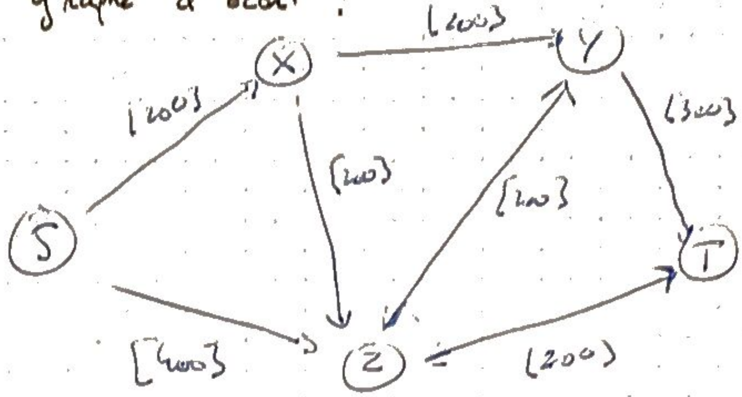
L'idée est de construire un graphe d'écart G_e de flot et un graphe d'écart G_c de coût. Puis on cherche un chemin minimal de coût dans le graphe d'écart de coût grâce à un algo de plus court chemin (Dijkstra, Bellman, Ford..). Le chemin le plus court de s à t trouvé est le chemin augmentant dans G_e. On applique le même principe que l'algo de Roy : on modifie f et $G_e(f)$ en utilisant ce chemin.

Exemple

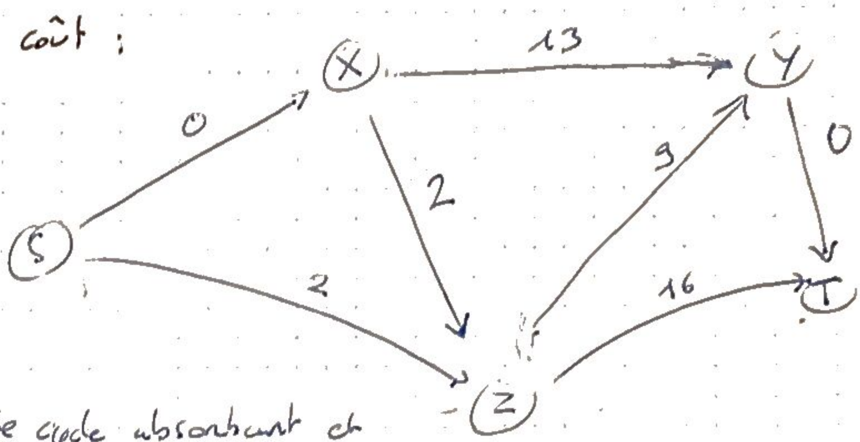


$v(f) = 0$

graphe d'écart :



graphe de coût :



Il n'y a pas de cycle absorbant et
$\forall ij \in U$ $l_{ij} \geq 0$ $\Rightarrow$ on peut appliquer Dijkstra.

On trouve $S \longrightarrow X \longrightarrow Z \longrightarrow Y \longrightarrow T$

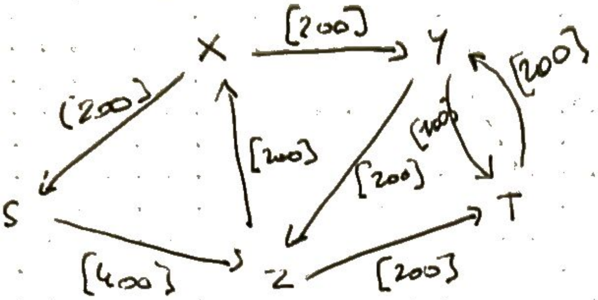
comme chemin optimal de S à T.

$S \longrightarrow X \longrightarrow Z \longrightarrow Y \longrightarrow T$

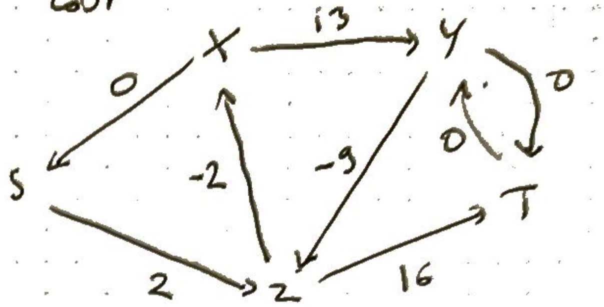
marge	200	200	200	300

$\Rightarrow$ min = 200

Gc



coût



* longueur négative => pas Dijkstra
* pas d'ordre topologique => pas Bellman

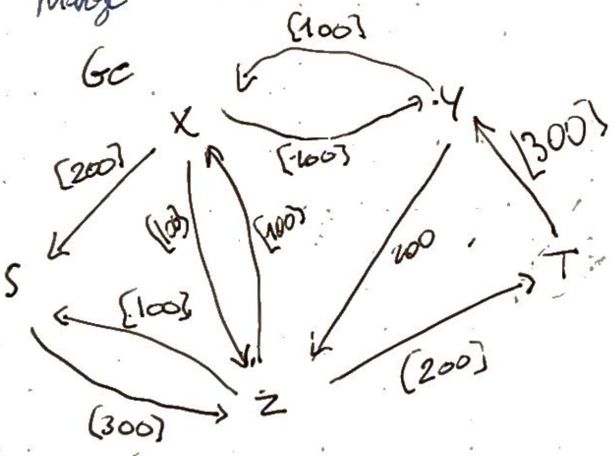
il reste Ford pour trouver le plus court chemin.

	S	X	Y	Z	T	
π	0	$+\infty$	$+\infty$	2	$+\infty$	étape 1
pred	S	$\emptyset$	$\emptyset$	S	$\emptyset$	
π	0	0	13	2	13	étape 2 fini $\forall i,j \;\; \pi_j \leq \pi_i + l_{ij}$
pred	S	Z	X	S	Y	

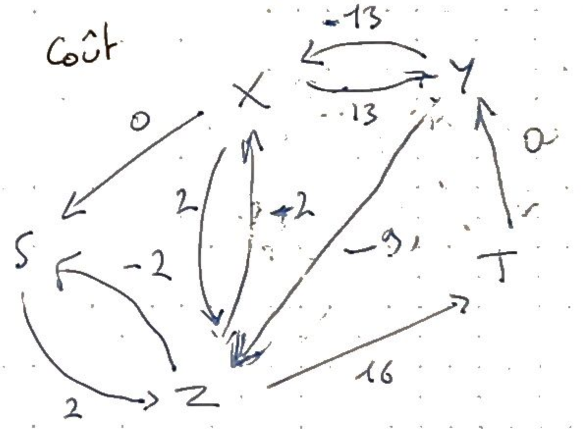
$S \xrightarrow[400]{} Z \xrightarrow[200]{} X \xrightarrow[200]{} Y \xrightarrow[100]{} T$ nouveau chemin augmentant. min = 100

marge

Gc



Coût



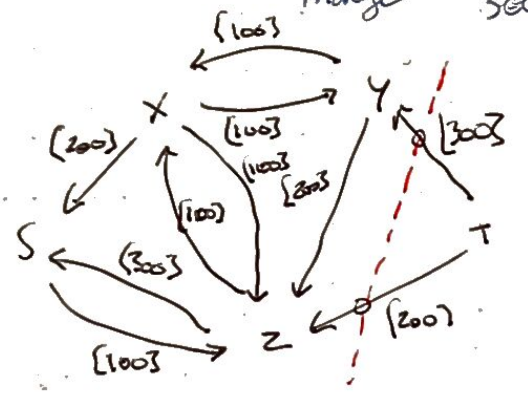
De nouveau, on est obligé de faire Ford.

	S	X	Y	Z	T	
π	0	$+\infty$	$+\infty$	2	$+\infty$	étape 1
pred	S	ϕ	ϕ	S	ϕ	

π	0	0	13	2	18	étape 2	fini
pred	S	Z	X	S	Z		

$S \longrightarrow Z \longrightarrow T$ nouveau chemin augmentant

marge 300 200 min = 200



plus de chemin possible vers T

$v(f) = 500$ flt max. et on obtient le graphe G :