Max-Flow Min-Cut

1. Reti di Trasporto
2. Flussi
3. Tagli
4. Max-Flow Min-Cut

Melissa Sordi 16/04/2021

1.Reti di Trasporto

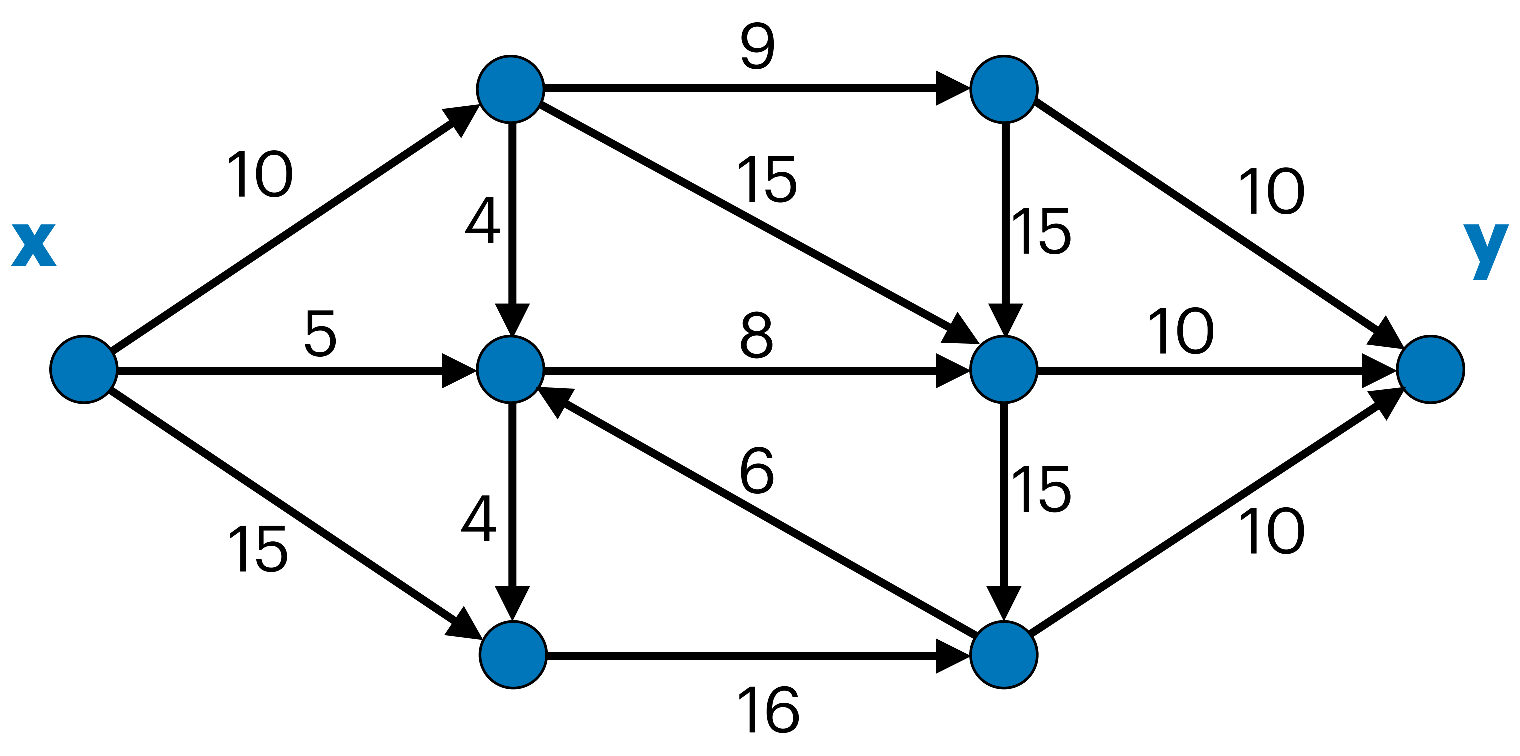
Iniziamo presentando gli elementi di base di questo argomento.

Definizione: Una ***rete*** $N := N(x, y)$ è un digrafo (grafo orientato) D , con $x, y \in V(D)$ vertici distinti, chiamati rispettivamente **sorgente** e **pozzo**, a cui è associata una funzione c reale non negativa definita sui lati di D. Questa funzione viene detta **funzione di capacità**. Per questo diremo che ogni lato $e \in E(D)$ ha **capacità** $c(e) \geq 0$.

Le reti sono utili per schematizzare diversi tipi di problemi.

L'idea è quella di voler mandare dei materiali da x a y attraverso questa rete di trasporto, la capacità di un arco indica la quantità di materiale che può essere trasportata tramite esso.
Un arco può avere capacità infinita, questo vuol dire che non c'è un limite alla quantità di materiale che può trasportare, anche se è difficile che questo accada in situazioni reali.

ESEMPIO 1:



2.Flussi

Chiamiamo I il sottoinsieme di V che contiene tutti i nodi di D , tranne la sorgente x e il pozzo y.
Questi nodi vengono chiamati **nodi intermedi**.

Definizione: Un (x, y)-**flusso** o semplicemente flusso in N è una funzione f a valori reali definita su $E(D)$ che soddisfa:

- Per ogni $e \in E(D)$ $\qquad 0 \leq f(e) \leq c(e)$ (1)
- Per ogni $v \in I$ $\qquad \sum_{\text{e entrante in v}} f(e) = \sum_{\text{e uscente da v}} f(e)$ cioè il flusso entrante in un nodo v, deve essere uguale al flusso uscente da esso. (2)

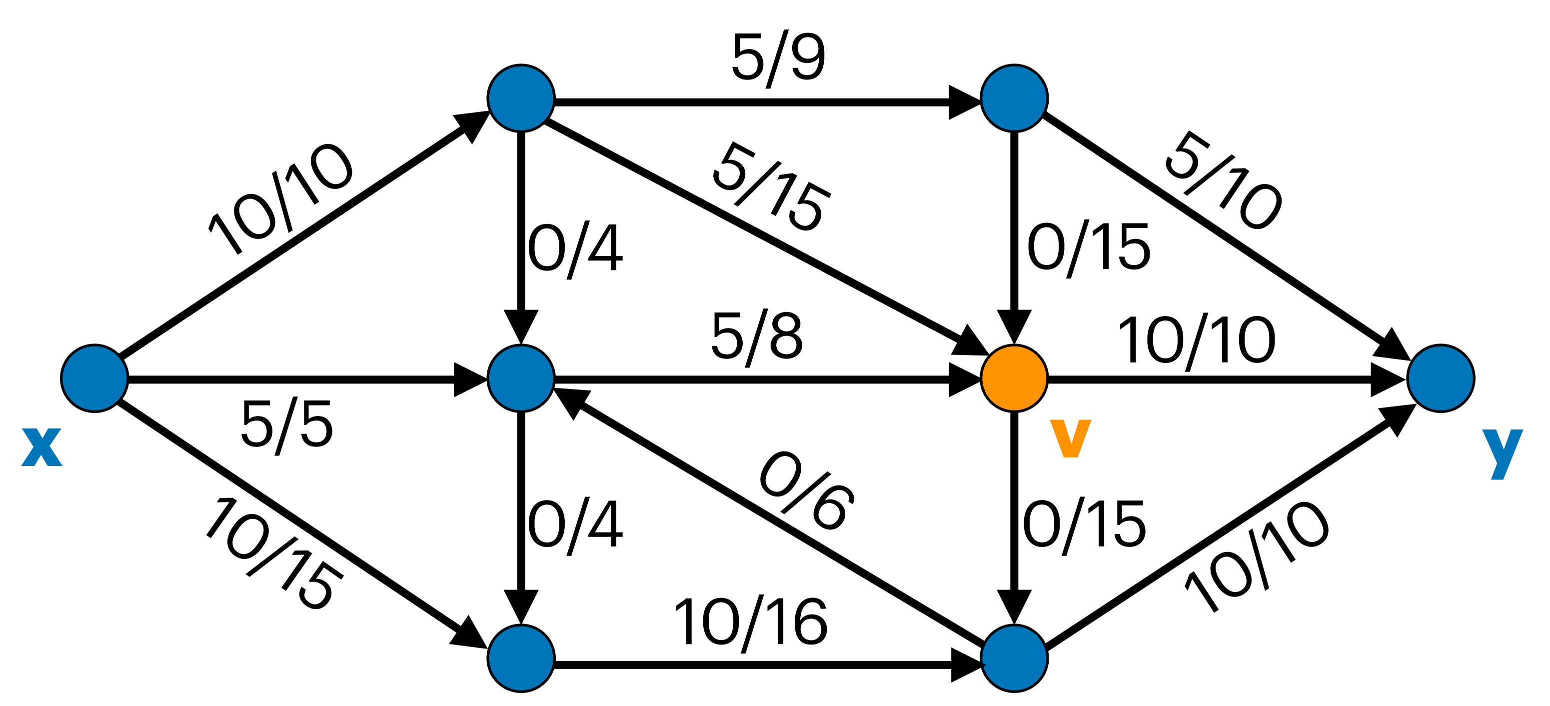
Per comodità chiameremo $f^+(v) = \sum_{\text{e uscente da v}} f(e)$ (flusso uscente) e $f^-(v) = \sum_{\text{e entrante in v}} f(e)$ (flusso entrante)

Possiamo allora riscrivere (2) in questo modo:

$$f^+(v) = f^-(v) \qquad \forall v \in I$$

Questa condizione è conosciuta come **condizione di conservazione**.

Notiamo che ogni rete ha almeno un flusso ben definito infatti la funzione $f(e) := 0 \ \ \forall e \in E(D)$ soddisfa entrambe le condizioni.
Vediamo un esempio non banale di flusso sulla rete vista in precedenza.
ESEMPIO 2:



Flusso entrante in $v = 5 + 5 + 0 = 10.$

Flusso uscente in $v = 0 + 10 = 10.$

Definizione: Sia f un (x, y)-flusso. Il **valore** di f è:

$$val(f) = f^{+}(x) - f^{-}(x)$$

Nell'*ESEMPIO 2* si ha $val(f) = 10 + 5 + 10 = 25.$

Dalla condizione (2) possiamo dedurre:

$$val(f) = f^+(x) - f^-(x) = f^-(y) - f^+(y)$$

Sia $S \subset V(D)$ un sottoinsieme di vertici di una rete N e un flusso f di N, allora:

$$f^+(S) - f^-(S) = \sum_{e \text{ uscente da } s} f(e) - \sum_{e \text{ entrante in } s} f(e) \qquad \forall s \in S$$

viene chiamato **flusso netto uscente** da S. Mentre $f^-(S) - f^+(S)$ viene chiamato **flusso netto entrante** in S.

Il valore di un flusso f può essere espresso tramite un qualsiasi sottoinsieme X di $V(D)$ tale che $x \in X$ e $y \in V \backslash X$, come si mostra ora.

Proposizione2.1: per ogni flusso f in una rete $N(x, y)$, e per ogni sottoinsieme X di V tale che $x \in X$ e $y \in V \backslash X$ si ha che

$$val(f) = f^+(X) - f^-(X)$$

Dimostrazione: Dalla definizione di flusso e del valore associato ad esso sappiamo che:

$$f^+(v) - f^-(v) = \begin{cases} val(f) & se \;\; v = x \\ 0 & se \;\; v \in X \backslash \{x\} \end{cases}$$

Sommando su tutti gli $v \in X$ si ha:

$$\sum_{v \in X} \left[f^+(v) - f^-(v)\right] = val(f) + 0 = val(f)$$

e quindi:

$$f^+(X) - f^-(X) = val(f) \qquad \square$$

Definizione: Un flusso f in una rete N è detto **flusso massimo** se non esiste un flusso in N con valore maggiore del valore di f.

Una rete $N(x, y)$ in cui tutti gli archi sono di capacità infinita, ammette flussi di valore arbitrariamente alto.
Tali reti sono però difficili da riprodurre nella pratica, quindi possiamo ridurci a trattare le reti che ammettono flussi massimi.

PROBLEMA: Maximum Flow

Input: Una rete $N(x, y)$.

Output: Flusso massimo da x a y in N.

3.Tagli

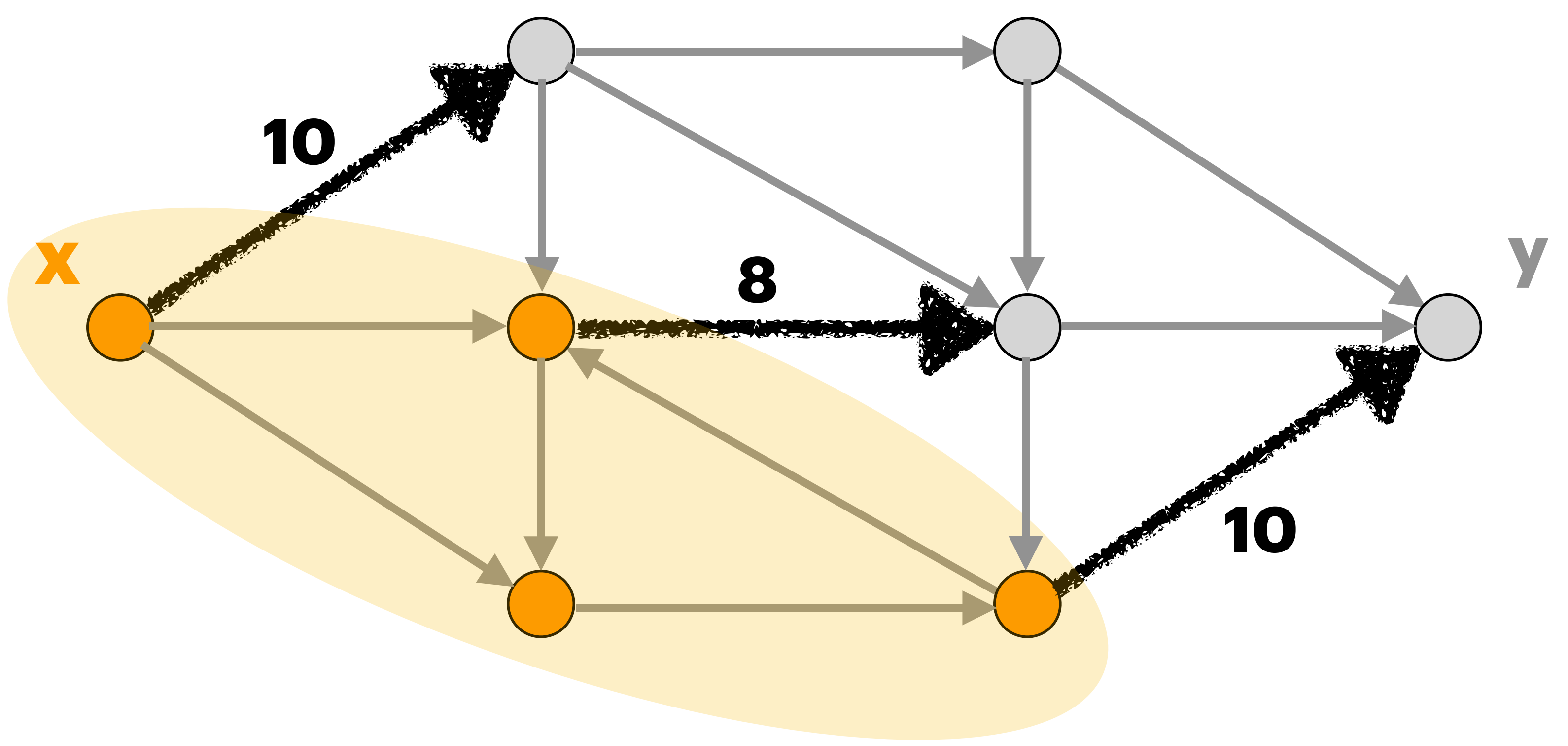
Definizione: Sia D un digrafo. Un (x, y)-**taglio** è una partizione dei nodi di $V(D)$ in due insiemi X e $V \backslash X$ tali che $x \in X$ e $y \in V \backslash X$. Indicheremo l'insieme dei lati che vanno da X a $V \backslash X$ con $\partial^+(X)$.

Un **taglio** in una rete $N(x, y)$ è un (x, y)-taglio nel suo digrafo sottostante e lo indicheremo con $K := \partial^+(X)$.

La **capacità di un taglio** è la somma delle capacità degli archi che appartengono a K:

$$cap(K) = \sum_{e \in K} c(e)$$

ESEMPIO 3:



In arancione abbiamo rappresentato i nodi di X

Con le frecce marcate gli archi di $\partial^+(X)$

$cap(K) = 10 + 8 + 10 = 28$

Flussi e tagli sono correlati tra loro, come vedremo nel seguente Teorema.

Teorema3.1: *Per qualsiasi flusso f e qualsiasi taglio $K := \partial^+(X)$ in una rete N si ha:*

$$val(f) \leq cap(K)$$

Inoltre l'uguaglianza vale se ogni arco a in $\partial^+(X)$ è tale che $f(a) = c(a)$, e in tal caso diremo che a è f-**saturo**, e ogni arco a in $\partial^-(X)$ è tale che $f(a) = 0$, e in tal caso diremo che a è f-**nullo**.

Dimostrazione:

Da (1) sappiamo che:

$$0 \leq f(e) \leq c(e) \quad \forall e \in E(D)$$

quindi:

$$f^+(X) \leq c^+(X) \quad \text{e} \quad f^-(X) \geq 0$$

Applicando la Proposizione2.1 precedente:

$$val(f) = f^+(X) - f^-(X) \leq c^+(X) = cap(K)$$

Se $f(a) = c(a) \;\; \forall a \in \partial^+(X)$ abbiamo $f^+(X) = c^+(X)$ e se $f(a) = 0 \;\; \forall a \in \partial^-(X)$ allora $f^-(X) = 0$.

E quindi risulterà: $$val(f) = cap(K). \qquad \square$$

Definizione: Un taglio K in una rete N è un **taglio minimo** se nessun taglio in N ha capacità minore di K.

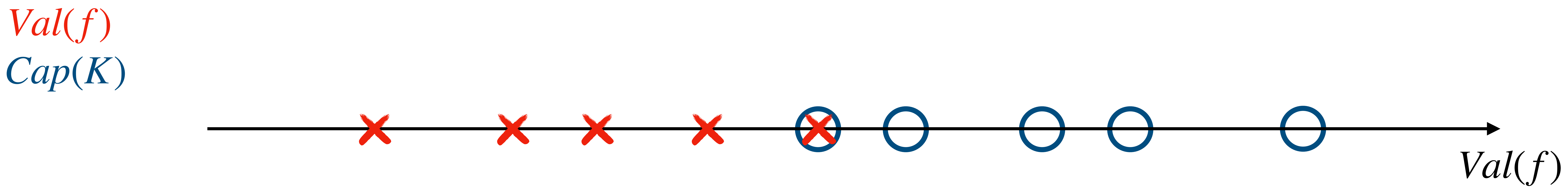
Corollario3.2: Sia f un flusso e K un taglio, se $val(f) = cap(K)$ allora f è un flusso massimo e K un taglio minimo.

Dimostrazione: Sia f^* un flusso massimo e K^* un taglio minimo, dal Teorema3.1 abbiamo:

$$val(f) \leq val(f^*) \leq cap(K^*) \leq cap(K)$$

Ma, per ipotesi $val(f) = cap(K)$, quindi $val(f) = val(f^*)$ e $cap(K) = cap(K^*)$. Ne deduciamo che f è un flusso massimo e K è un taglio minimo. $\square$

Intuitivamente...

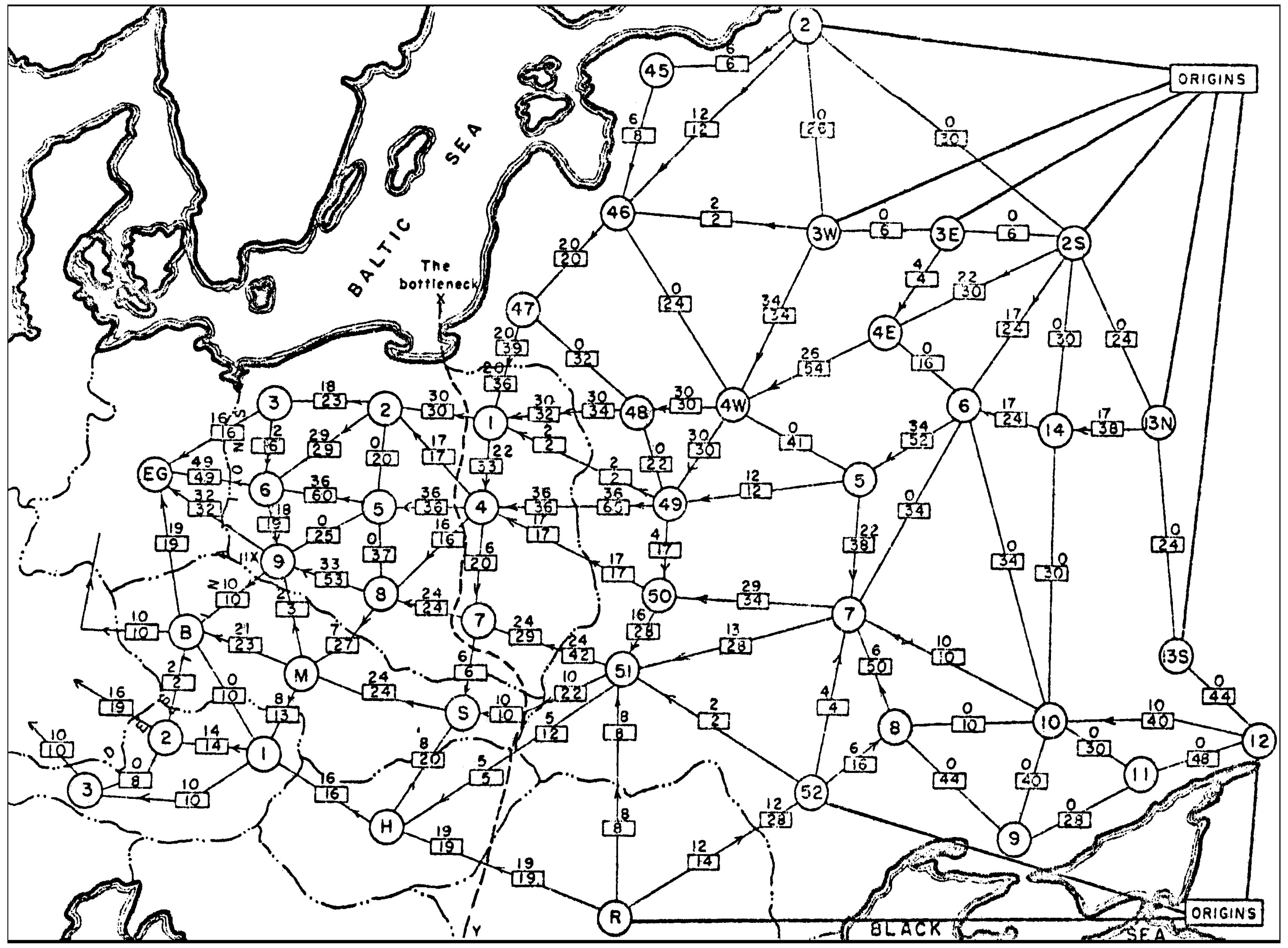


4.Max-Flow Min-Cut

LA STORIA

Il problema del massimo flusso minimo taglio fu introdotto negli anni '30.
Durante la Guerra Fredda i sovietici volevano massimizzare l'apporto di rifornimenti ai paesi dell'URSS mentre gli statunitensi volevano determinare il "minimo taglio" da effettuare alla rete ferroviaria sovietica per sconnettere Mosca dal resto dell'URSS.

Studi sovietici:
Leonid V. Kantorovich in *"Metodi matematici di organizzazione e pianificazione della produzione"* (1939).
Studi statunitensi:
L.R.Ford Jr., D.R. Fulkerson in *"Maximal flow trough a network"* (1954).
T.E. Harris, Gen F.S. Ross in *"Fondamentals of a method for evaluating network capacities"* (1955)



Rete Ferroviaria Sovietica

IL TEOREMA

Enunciato: In ogni rete, il valore del flusso massimo f^* eguaglia quello del taglio minimo K^*:

$$val(f^*) = cap(K^*).$$

Dobbiamo far vedere che anche l'implicazione inversa a quella del Corollario3.2 è valida e per farlo abbiamo bisogno di alcuni risultati preliminari.

Sia f un flusso in una reta $N := N(x, y)$ dove ad ogni x-cammino (non necessariamente diretto) P in N associamo un intero non negativo $\epsilon(P)$ così definito:

$$\epsilon(P) := min\{\epsilon(e) : e \in E(P)\}$$

dove

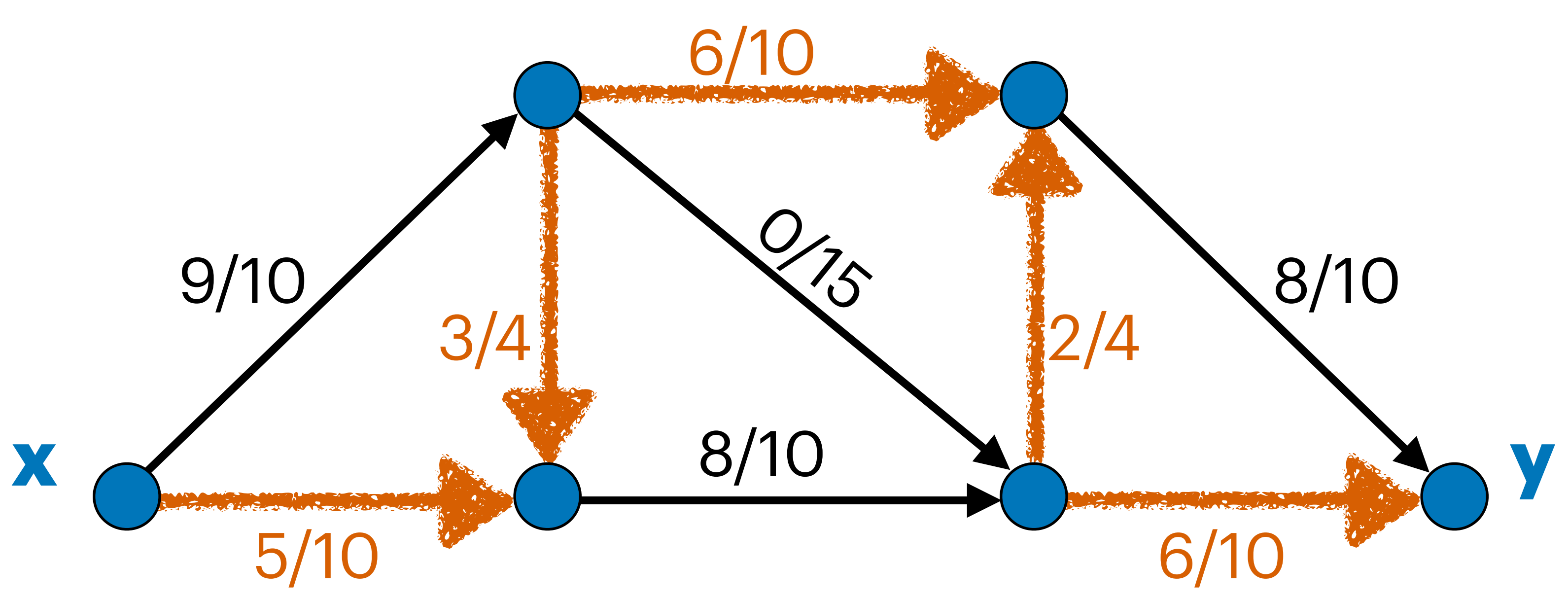
$$\epsilon(e) := \begin{cases} c(e) - f(e) & \text{se } e \text{ è un arco in avanti di } P \\ \\ f(e) & \text{se } e \text{ è un arco al rovescio di } P \end{cases}$$

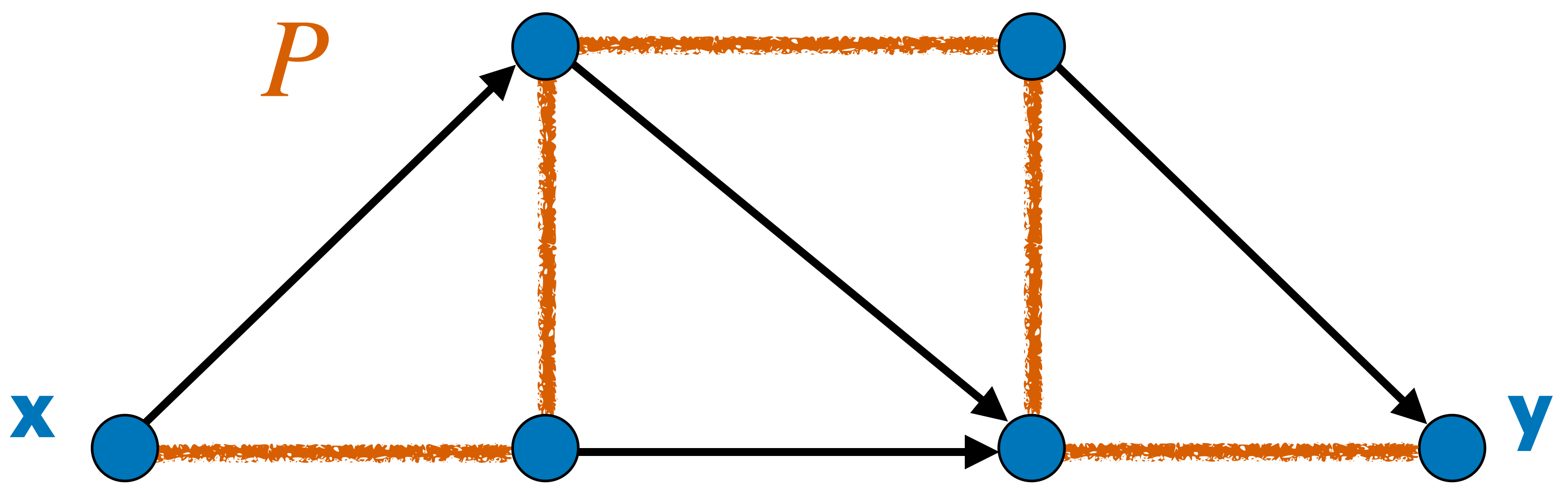
$\epsilon(P)$ è la quantità massima che possiamo aggiungere al flusso f lungo P in modo tale da non violare i vincoli dati da (1).

Il cammino P viene detto f**-saturo** se $\epsilon(P) = 0$ e f**-insaturo** se $\epsilon(P) > 0$.

Un **cammino** f**-aumentante** è un (x, y)-cammino f-insaturo.

ESEMPIO 4:





In questo caso abbiamo preso un cammino P non diretto.

E abbiamo $\epsilon(P) := min\{\epsilon(e) : e \in E(P)\} = min\{(10-5),\ 3, (10-6), 2, (10-6)\} = min\{5, 3, 4, 2, 4\} = 2.$

L'esistenza di un cammino f-aumentante P implica che f non è un flusso massimo. Inviando un flusso aggiuntivo pari a $\epsilon(P)$ lungo P, si ottiene un nuovo flusso f' di valore maggiore di f.
Più precisamente $f' : E \to \mathbb{R}$ tale che:

$$f'(e) := \begin{cases} f(e) + \epsilon(P) & \text{se } e \text{ è un arco in avanti di } P \\ f(e) - \epsilon(P) & \text{se } e \text{ è un arco al rovescio di } P \\ f(e) & \text{altrimenti} \end{cases}$$

Proposizione4.1: Sia f un flusso in una rete N. Se esiste un cammino f-aumentante, allora f non è un flusso massimo.
Più precisamente, la funzione f' definita in precedenza è un flusso in N e che $val(f') = val(f) + \epsilon(P)$.

Dimostrazione:

Iniziamo mostrando che f' è un flusso.

Sia a un arco in avanti di P. Allora $f'(a) = f(a) + \epsilon(P)$.

Per la stima dal basso notiamo che $\epsilon(P) > 0$ (altrimenti P non sarebbe un cammino f-aumentante) quindi si ha:

$$f'(a) = f(a) + \epsilon(P) > 0$$

essendo f un flusso.

Per la stima dall'alto invece il caso peggiore si ha per $\epsilon(P) = c(a) - f(a)$, quindi avremmo:

$$f'(a) = f(a) + \epsilon(P) \leq f(a) + (c(a) - f(a)) = c(a)$$

quindi f' rispetta la proprietà (1) del flusso per ogni arco in avanti di P.

Adesso invece prendiamo r, arco al rovescio di P. Allora $f'(r) = f(r) - \epsilon(P)$.

Per la stima dal basso notiamo che nel caso peggiore $\epsilon(P) = f(r)$ e quindi:

$$f'(r) = f(r) - \epsilon(P) \geq f(r) - f(r) = 0$$

Per la stima dall'alto invece utilizziamo che $\epsilon(P) > 0$, quindi si ha:

$$f'(r) = f(r) - \epsilon(P) \leq f(r) \leq c(r)$$

essendo f un flusso. Quindi f' rispetta la proprietà (1) del flusso anche per tutti gli archi al rovescio di P.

Ora dobbiamo far vedere che f' verifica anche la proprietà di conservazione del flusso (2).
Nei nodi che non appartengono a P abbiamo:

$$\sum_{\text{e entrante in v}} f'(e) = \sum_{\text{e entrante in v}} f(e) = \sum_{\text{e uscente da v}} f(e) = \sum_{\text{e uscente da v}} f'(e) \qquad \forall v \in V(D) \setminus V(P)$$

essendo f un flusso.
Per i vertici di D che fanno parte di P possiamo notare che, essendo P un cammino, ci saranno esattamente due archi di P incidenti in ogni nodo intermedio di P.
Preso $v \in P$ Indichiamo con $a1$ e $a2$ i due archi di P incidenti in v, con I l'insieme degli archi entranti in v e con J l'insieme degli archi uscenti da v. Ovviamente si ha: $v \in V(P) \subset V(D)$, $I \subset E(D)$ e $J \subset E(D)$.
Se $a1$ e $a2$ sono entrambi archi in avanti di P allora saranno uno entrante e uno uscente.
Supponiamo $a1 \in I$ e $a2 \in J$, avremo quindi:

$$\sum_{e \in I} f'(e) = \sum_{e \in I \setminus \{a1\}} f(e) + f(a1) + \epsilon(P) = \sum_{e \in I} f(e) + \epsilon(P)$$

mentre:

$$\sum_{e \in J} f'(e) = \sum_{e \in J \setminus \{a2\}} f(e) + f(a2) + \epsilon(P) = \sum_{e \in J} f(e) + \epsilon(P)$$

per la proprietà di conservazione di f le due espressioni risultano essere uguali.

Se $a1$ e $a2$ sono entrambi archi al rovescio di P ci troviamo in una situazione analoga alla precedente.
Se invece $a1$ è un arco in avanti di P e $a2$ è un arco al rovescio di P i due archi saranno o entrambe uscenti o entrambi entranti in v.
Supponiamo siano entrambe entranti, allora avremo che:

$$\sum_{e\in I} f'(e) = \sum_{e\in I\setminus\{a1,a2\}} f(e) + (f(a1) + \epsilon(P)) + (f(a2) - \epsilon(P)) = \sum_{e\in I} f(e) + \epsilon(P) - \epsilon(P) = \sum_{e\in I} f(e)$$

mentre:

$$\sum_{e\in J} f'(e) = \sum_{e\in J} f(e)$$

per la proprietà di conservazione di f le due espressioni risultano essere uguali.
Quindi f' soddisfa entrambe le proprietà (1) e (2).

Ora non ci resta che calcolare il valore di f'.
Sappiamo che:

$$val(f') = f'^{+}(x) - f'^{-}(x).$$

Essendo P un x-cammino solo uno degli archi incidenti in x sarà un arco di P, supponiamo che sia un arco in avanti. Allora avremo $f'^{-}(x) = f^{-}(x)$.

Chiamiamo l'arco in avanti a, I l'insieme degli archi entranti in x e J l'insieme degli archi uscenti da x . Si avrà che:

$$val(f') = f'^+(x) - f'^-(x) = \sum_{e \in J} f'(e) - \sum_{e \in I} f(e) = \sum_{e \in J \setminus \{a\}} f(e) + f'(a) - \sum_{e \in I} f(e)$$

$$= \sum_{e \in J \setminus \{a\}} f(e) + f(a) + \epsilon(P) - \sum_{e \in I} f(e) = \sum_{e \in J} f(e) - \sum_{e \in I} f(e) + \epsilon(P)$$

$$= val(f) + \epsilon(P).$$

Da questo risultato si deduce anche che f non è un flusso massimo, infatti f' è un flusso con $val(f') > val(f)$. $\square$

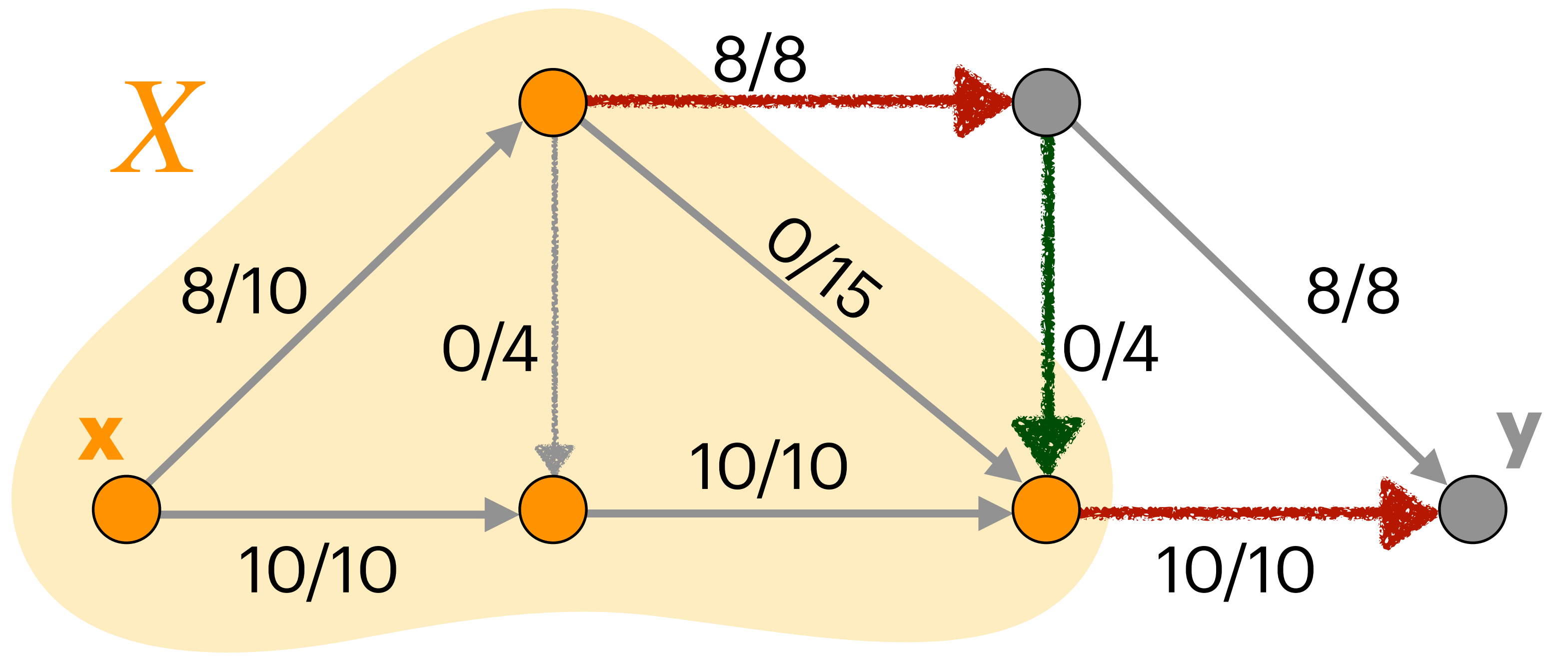
Possiamo chiamare f' **flusso aumentato basato su** P.

Vediamo f' nella rete che abbiamo rappresentato nell'esempio4. (Ricordiamo che $\epsilon(P) = 2$)

$val(f) = 14$ $\qquad$ $val(f') = 16$

Proposizione4.2: Sia f un flusso in una rete $N := N(x, y)$. Supponiamo che non ci sia più nessun cammino f-aumentante in N. Sia X l'insieme di tutti i vertici raggiungibili da x attraverso cammini f-insaturi, e poniamo $K := \partial^+(X)$. Allora $val(f) = cap(K)$, quindi f è un flusso massimo e K è un taglio minimo, in N.

Prima della dimostrazione possiamo vedere un esempio.
ESEMPIO5:
Dopo aver aumentato tutti i cammini ancora f-insaturi della rete rappresentata qui sotto, ci troviamo in questa situazione:



$\partial^+(X)$
$\partial^-(X)$

Dimostrazione: Chiaramente $x \in X$, mentre $y \in V \backslash X$ perché non esiste un cammino f-aumentante da x a y.
Quindi K è un taglio in N.
Consideriamo un arco $a \in \partial^+(X)$ con coda u e testa v.
Visto che $u \in X$ esiste un cammino f-insaturo (x, u) che chiameremo Q. Se $f(a) \leq c(a)$ fosse minore di $c(a)$ allora potremmo aggiungere l'arco a al cammino Q e ottenere un f-insaturo (x, v)-cammino. Ma $v \in V \backslash X$ quindi non può esistere un tale cammino. Quindi $f(a) = c(a)$.

Possiamo fare lo stesso tipo di ragionamento con $a \in \partial^-(X)$.
In questo caso avremmo $v \in X$ e $u \in V \backslash X$, esiste quindi un cammino Q f-insaturo da x fino a v. Se $f(a) \geq 0$ fosse maggiore di 0 allora potremmo aggiungere a a Q ottenendo un cammino f-insaturo da x fino a u, ma questo non è possibile perché sappiamo che $u \in V \backslash X$. Quindi $f(a) = 0$.
Dal Teorema3.1 sappiamo che $val(f) = cap(K)$ e dal Corollario3.2 concludiamo che f è un flusso massimo e K un taglio minimo. $\square$

The Max-Flow Min-Cut Theorem: In ogni rete, il valore del flusso massimo f eguaglia quello del taglio minimo K:

$$val(f) = cap(K).$$

Dimostrazione: Sia f un flusso massimo. Dalla Proposizione4.1 si deduce che non può esserci un cammino f-aumentante. A questo punto non ci resta che utilizzare la Proposizione4.2 per arrivare alla tesi. $\square$

L' ALGORITMO DI FORD E FULKERSON

Come conseguenza della Proposizione4.1 e della Proposizione4.2 si ha il seguente risultato:

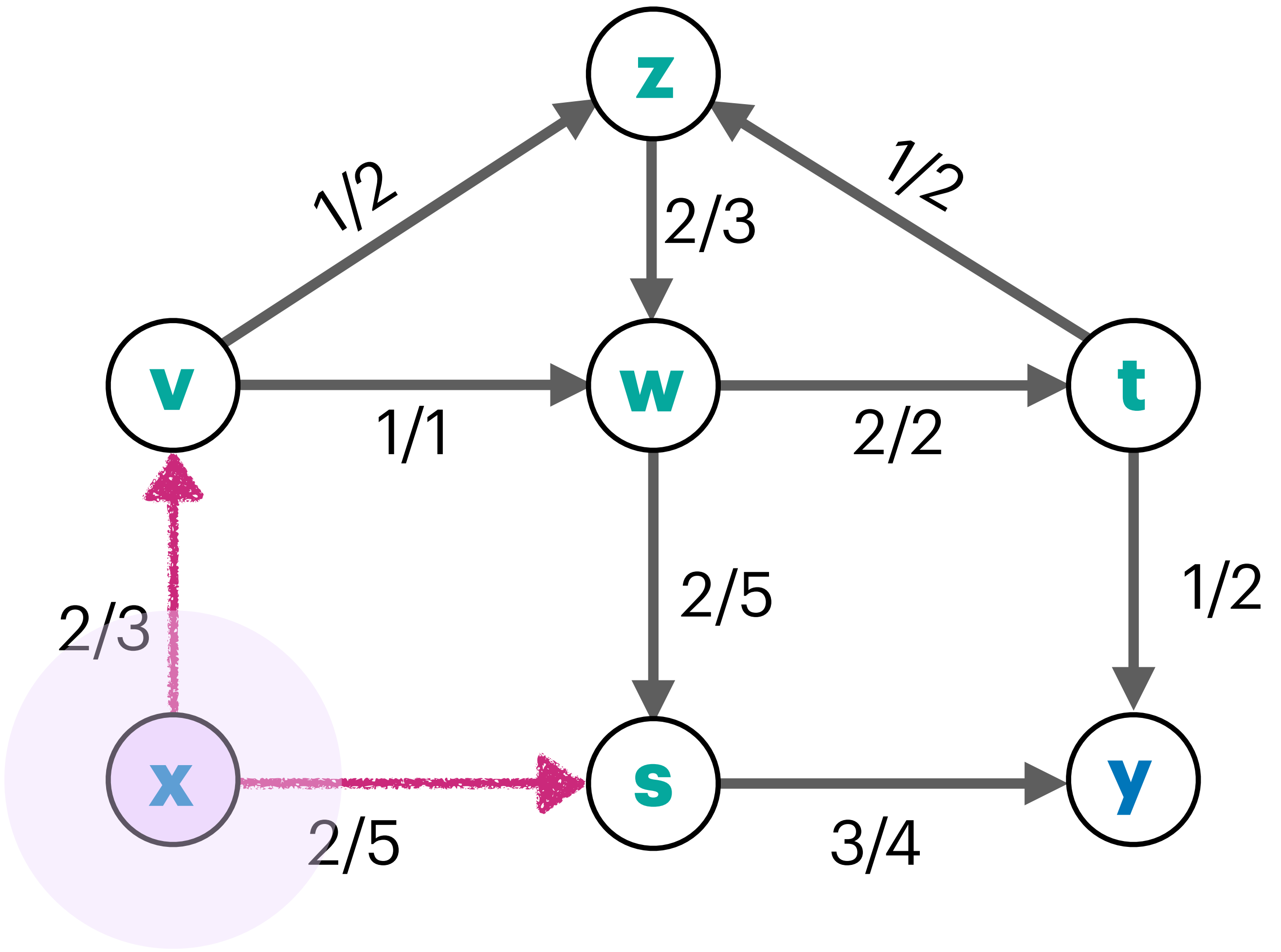
Teorema4.2: Un flusso f in una rete è un flusso massimo se e solo se non esiste un cammino f-aumentante.

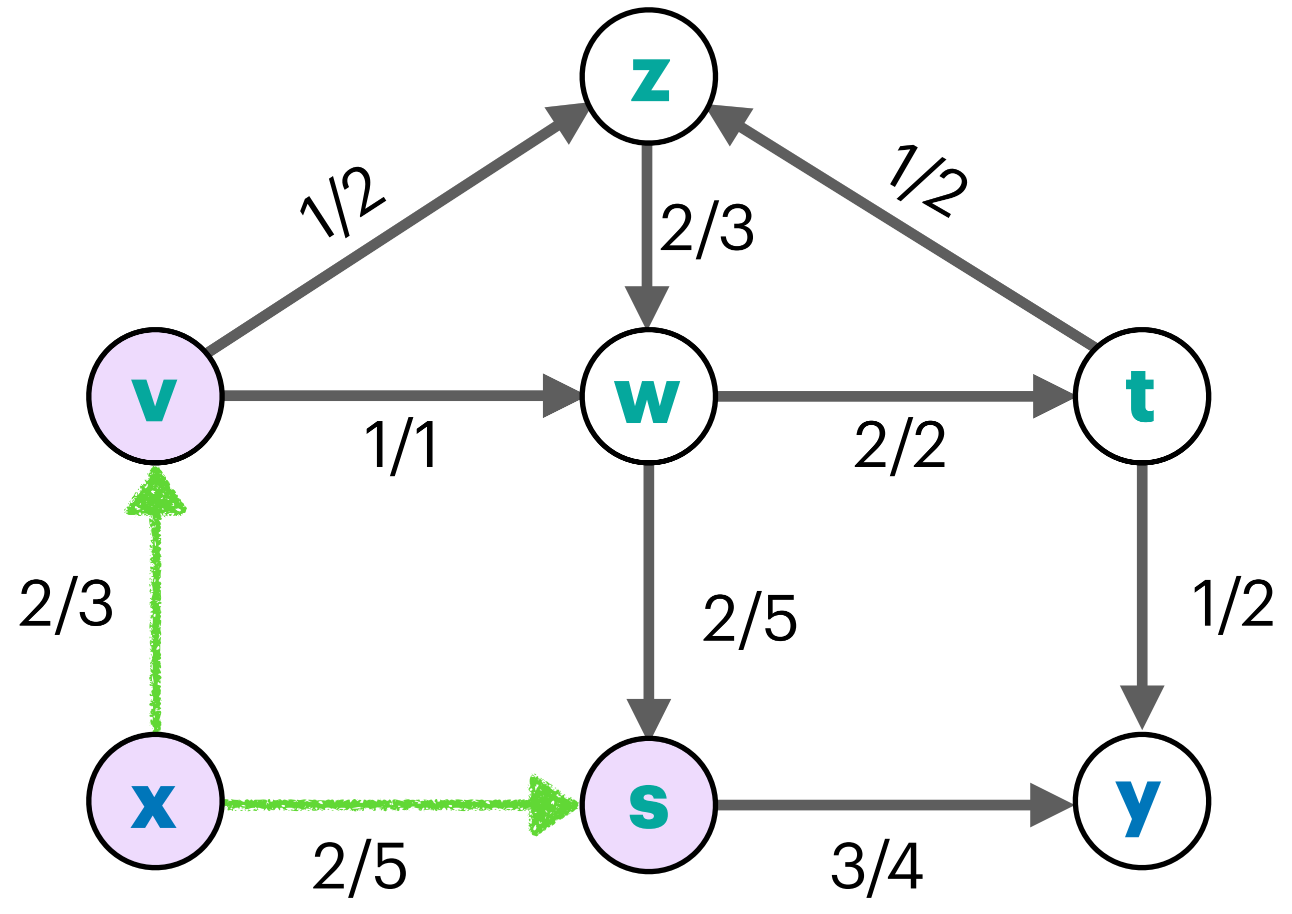
Su questo concetto si basa l'algoritmo di Ford e Fulkerson.

A partire da un flusso noto f, come ad esempio lo zero flusso ($f(e) = 0\ \forall e \in E(D)$), cerchiamo un cammino f-aumentante tramite un algoritmo di ricerca di alberi.

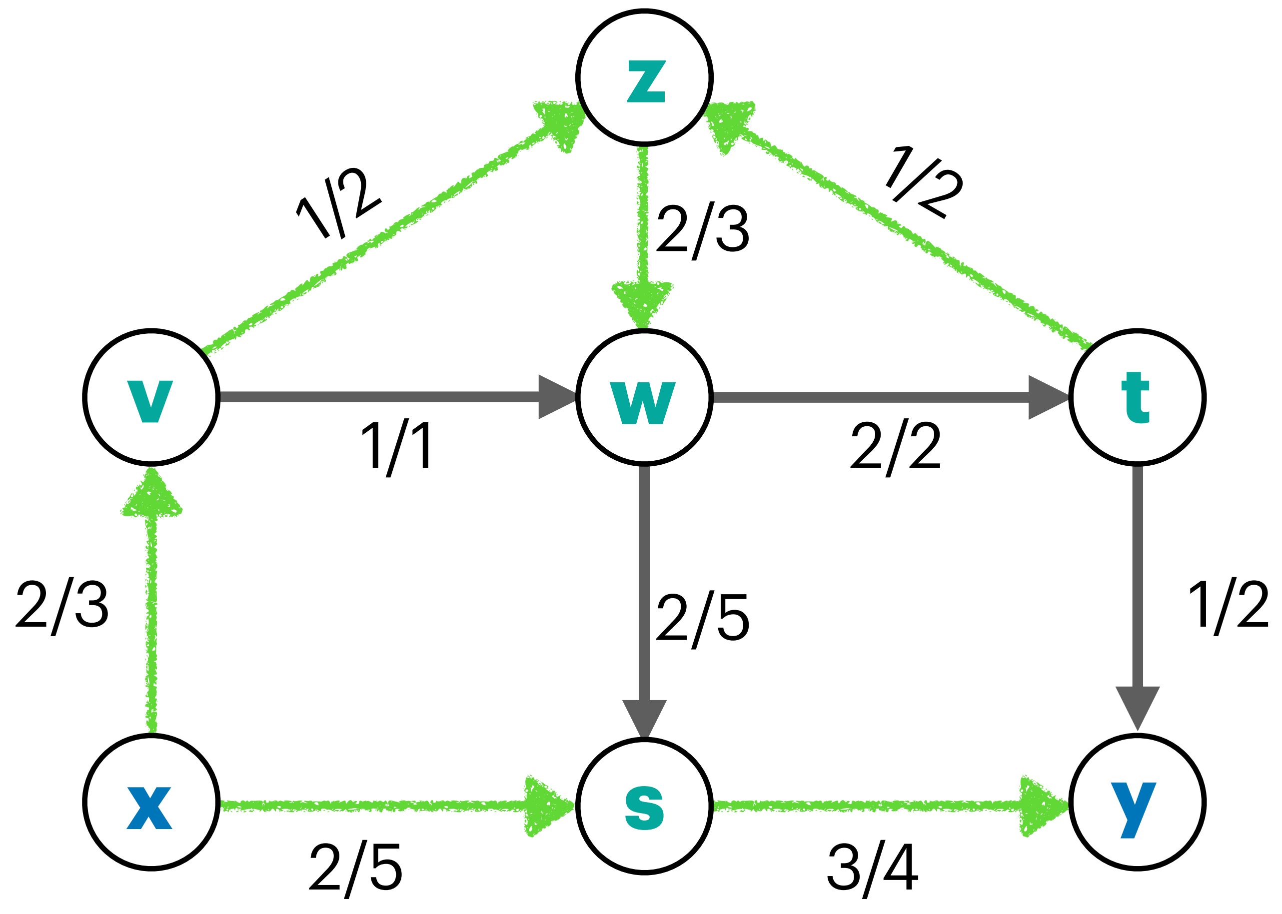
Un x-albero T è f-**insaturo** se, per ogni vertice v di T, il cammino xTv è f-insaturo.

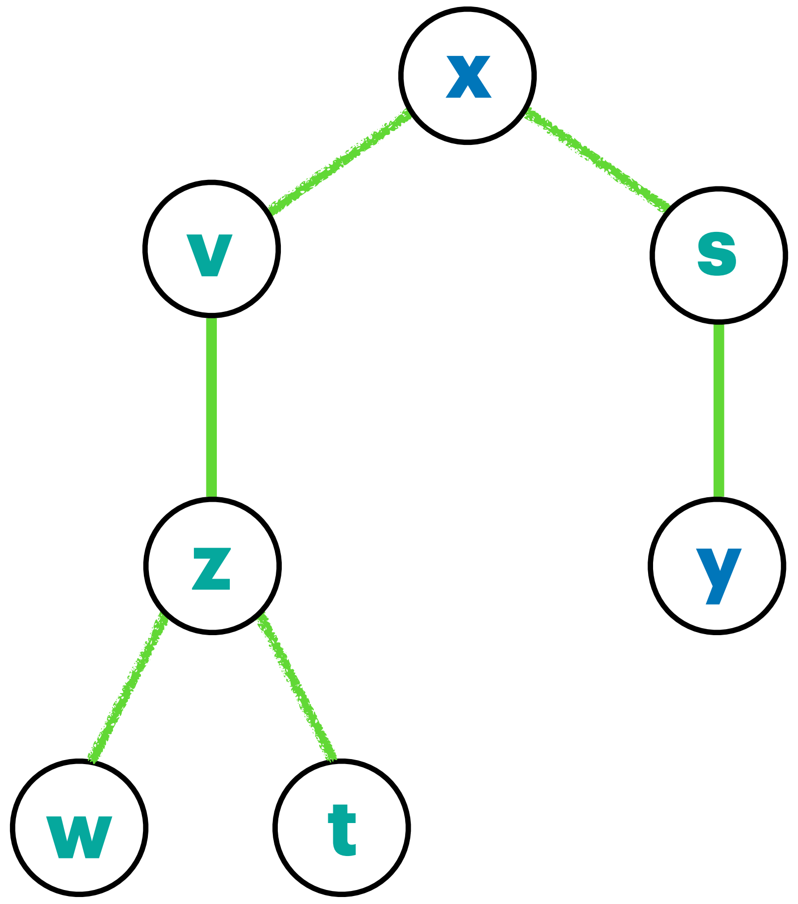
Inizialmente, l'albero T conterrà solo la sorgente x. In qualsiasi fase dell'algoritmo, se esiste un arco a f-insaturo ($f(a) < c(a)$) in $\partial^+(X)$, dove $X = V(T)$, aggiungeremo sia a che la sua testa a T.





Allo stesso modo, se esiste un arco e f-positivo ($f(e) > 0$) in $\partial^-(X)$, aggiungeremo sia e che la sua coda a T. Continueremo così ad aggiungere archi e vertici a T, fino ad arrivare ad un albero f-insaturo massimale.





Se l'albero T raggiunge il pozzo y, allora grazie ad una funzione che salva il predecessore di ogni nodo di T otterremmo un cammino f-aumentante P da x a y, e possiamo quindi calcolare $\epsilon(P)$ e sostituire f con il nuovo flusso f' definito come in precedenza.

Se T non raggiunge il pozzo, ed è un f-insaturo albero massimale, ogni arco in $\partial^+(X)$ è f-saturo e ogni arco in $\partial^-(X)$ è f-zero. Possiamo quindi concludere, in virtù del Teorema3.1 e del Corollario3.2, che il flusso f è un flusso massimo e il taglio $\partial^+(X)$ è un taglio minimo.

Vediamo ora lo pseudocodice dell'algoritmo e un esempio per capirne il funzionamento.

MAX-FLOW MIN-CUT

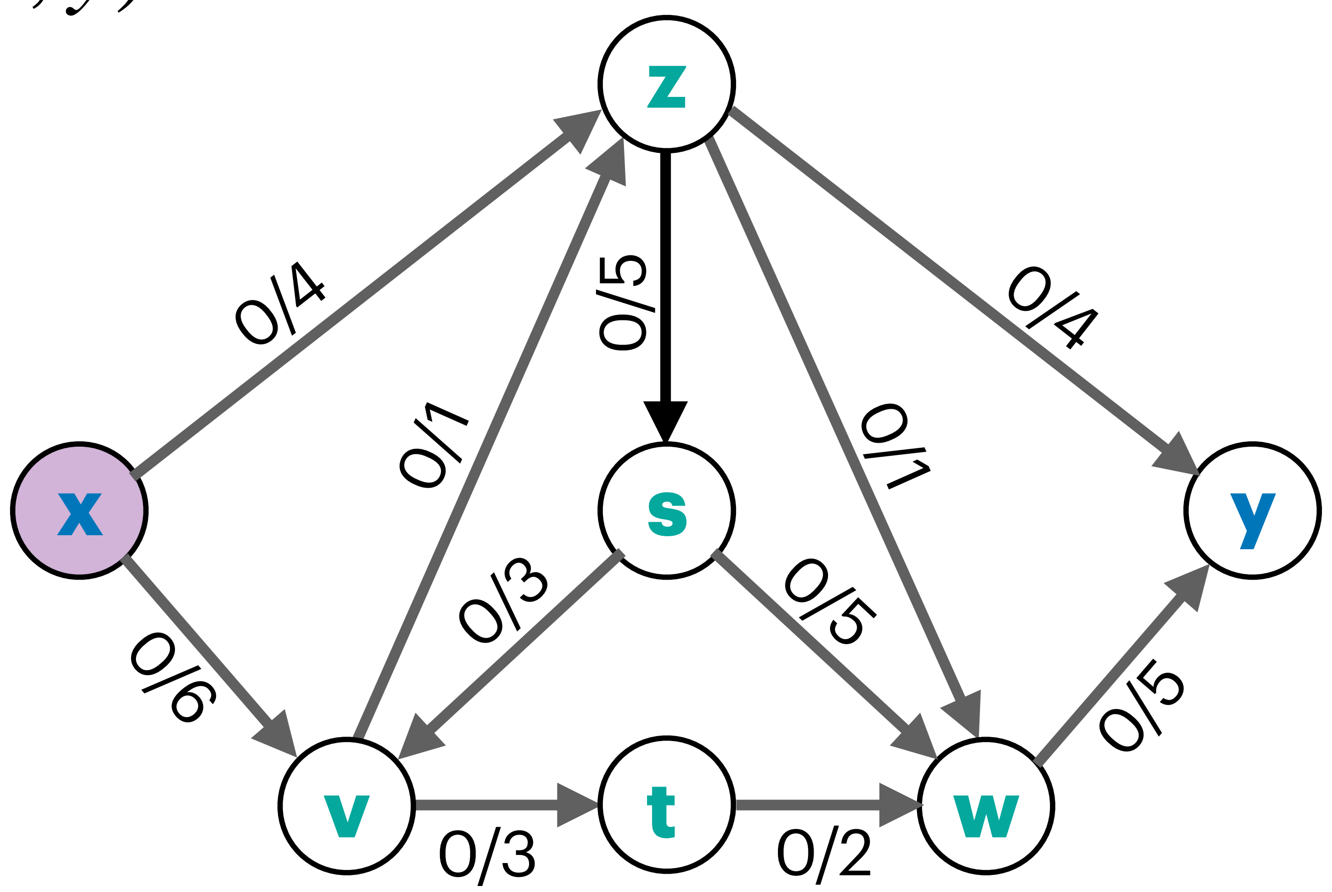
<u>INPUT:</u> una rete $N := N(x, y)$ e un flusso f in N.

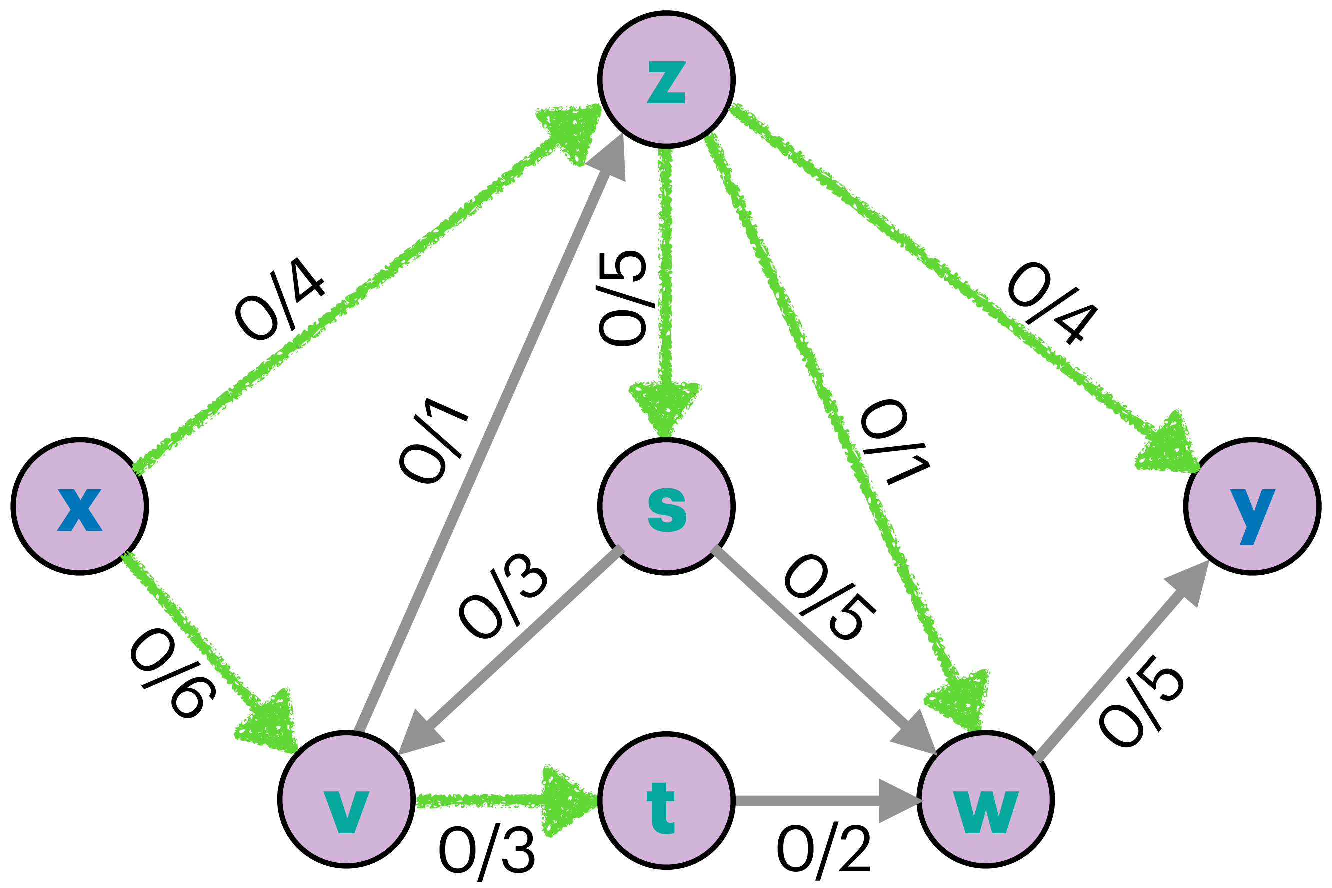
<u>OUTPUT:</u> flusso massimo f e taglio minimo $\partial^+(X)$ in N.

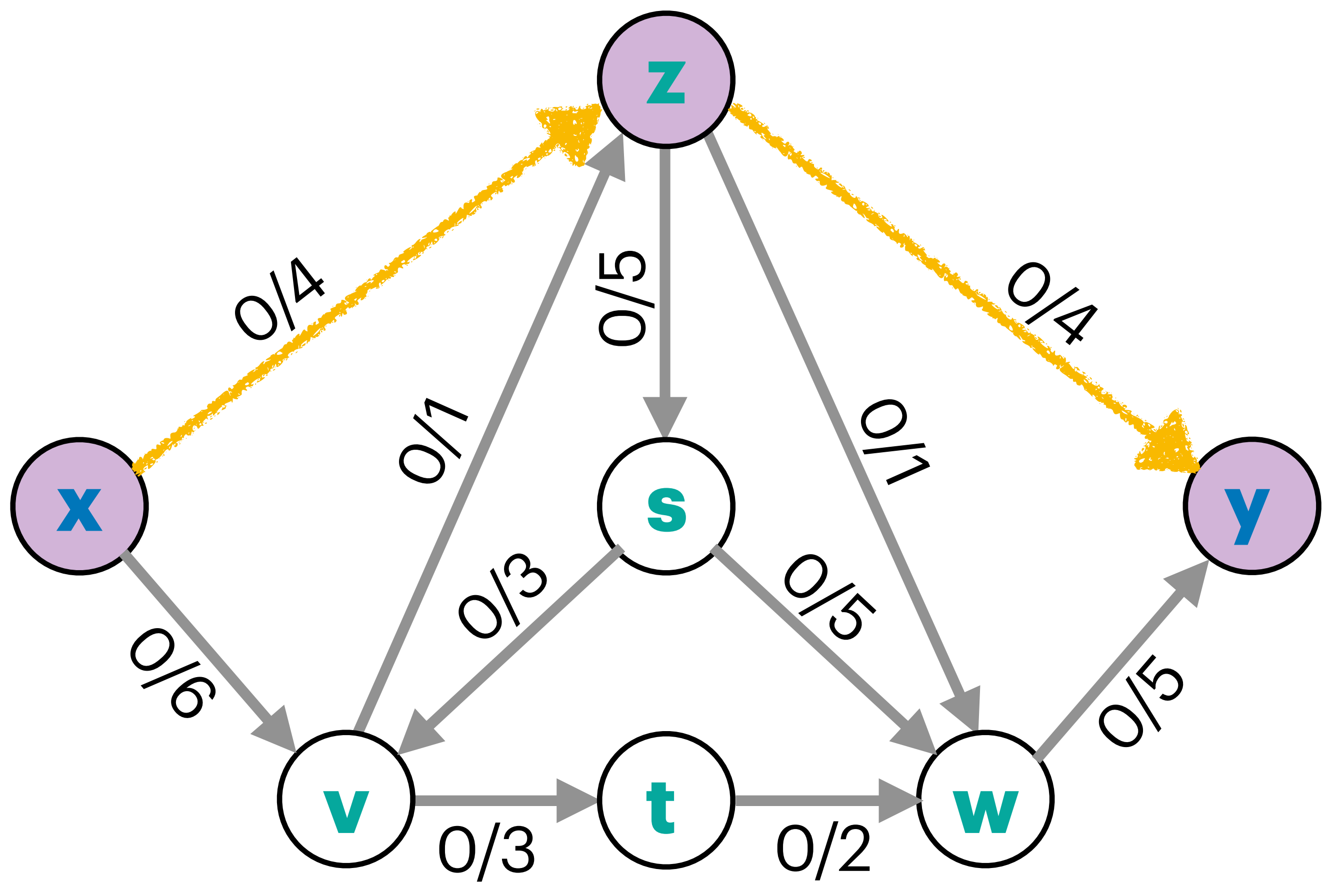
1. Impostare $X := \{x\}$, $p(v) := \emptyset \ \forall v \in V$.
2. Finché ci sono archi f-insaturi $a := (u, v)$ o archi f-positivi $e := (v, u)$ con $u \in X$ e $v \in V \setminus X$:
3. $\quad X = X \cup \{v\}$
4. $\quad p(v) = u$
5. Fine
6. Se $y \in X$ allora:
7. $\quad$ Calcolare $\epsilon(P) := min\{\epsilon(a) : a \in A(P)\}$ dove P è un (x, y)-cammino in T ottenuto
8. $\quad$ utilizzando la funzione predecessore p.
9. $\quad$ Per ogni arco in avanti a di P:
10. $\quad\quad f(a) = f(a) + \epsilon(P)$
11. $\quad$ Per ogni arco al rovescio e di P:
12. $\quad\quad f(e) = f(e) - \epsilon(P)$
13. $\quad$ Si torna al punto 1.
14. Fine
15. Restituisce $(f, \partial^+(X))$

<u>ESEMPIO MAX-FLOW MIN-CUT:</u>

$N(x, y)$







$X = \{x\}$

$p(v) = \varnothing \quad \forall v \in V(D)$

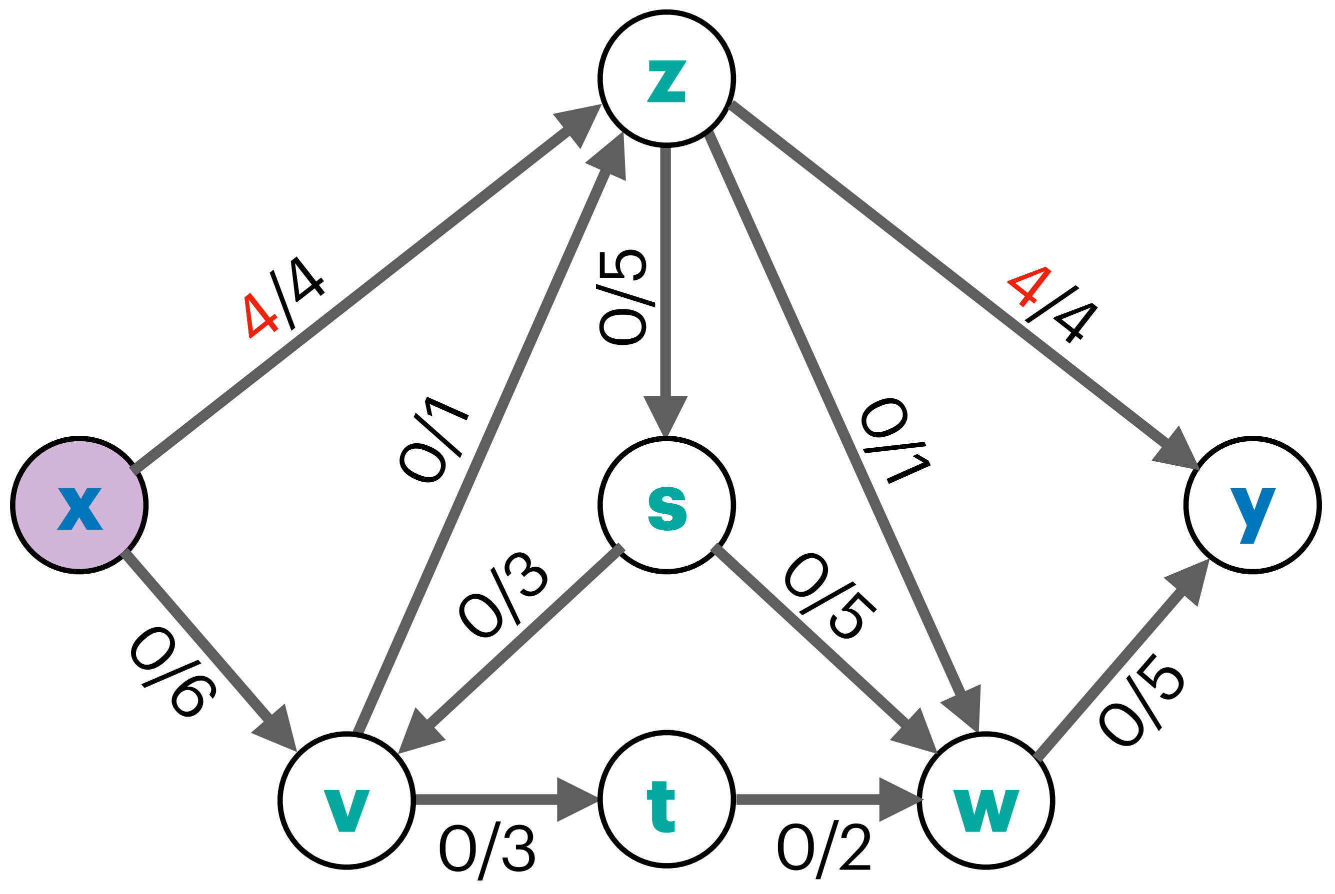
$X = \{x, z, s, w, y, v, t\},$

Albero f-insaturo massimale

$p(y) = z \quad p(z) = x$

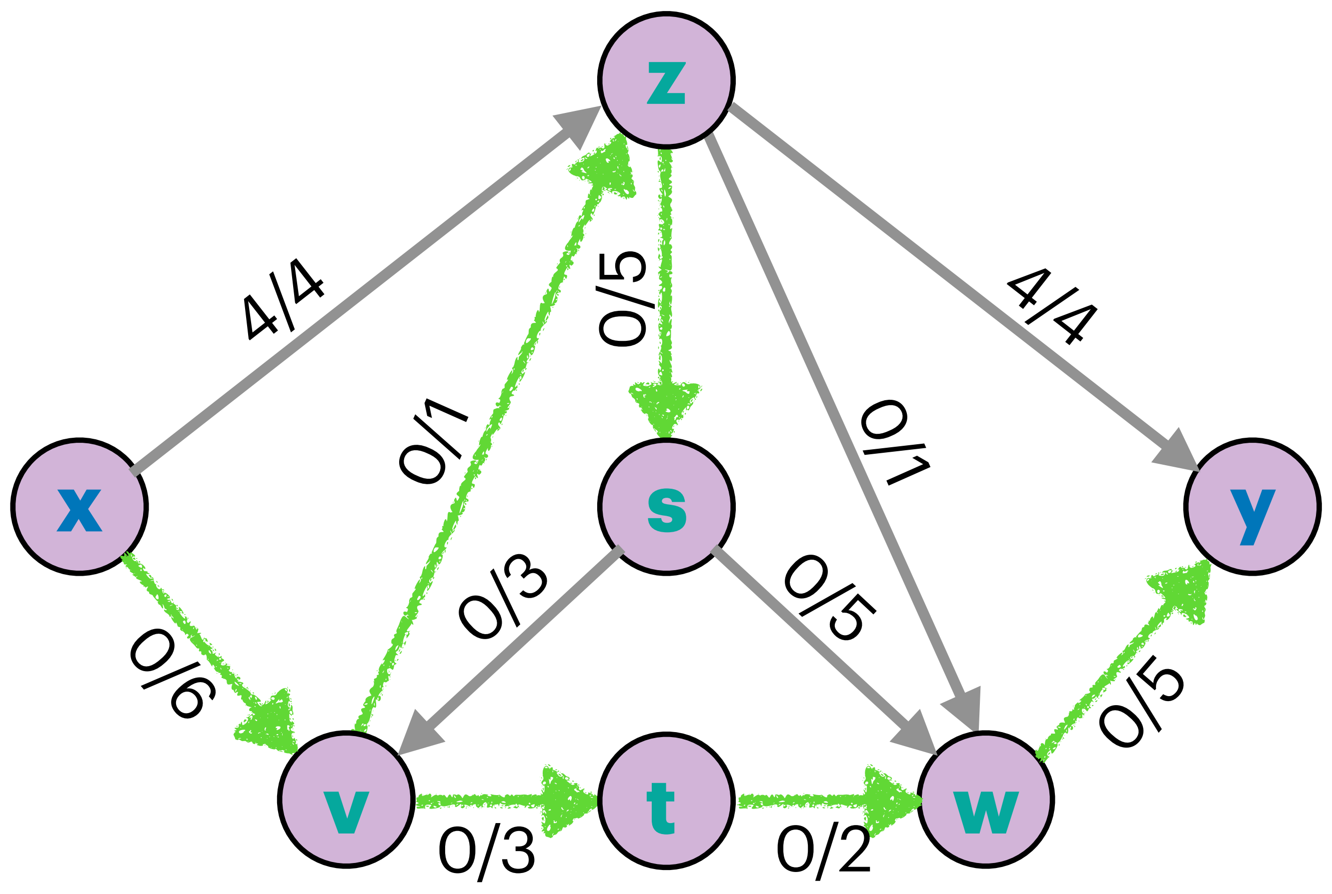
f-cammino insaturo

$\epsilon(P) = min\{4{,}4\} = 4$



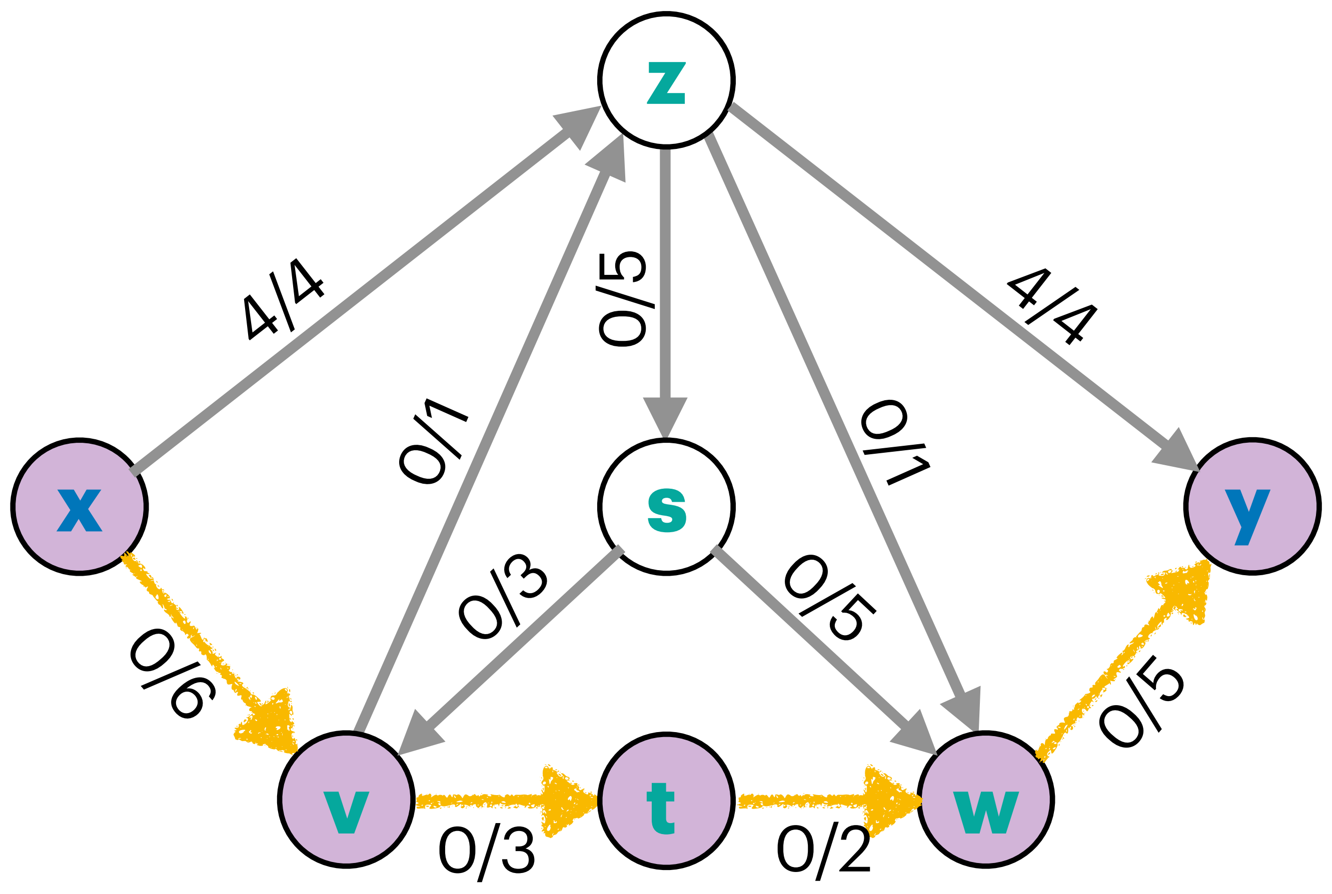
$X = \{x\}$

$p(v) = \varnothing \quad \forall v \in V(D)$



$X = \{x, v, t, w, y, z, s\}$,

Albero f-insaturo massimale

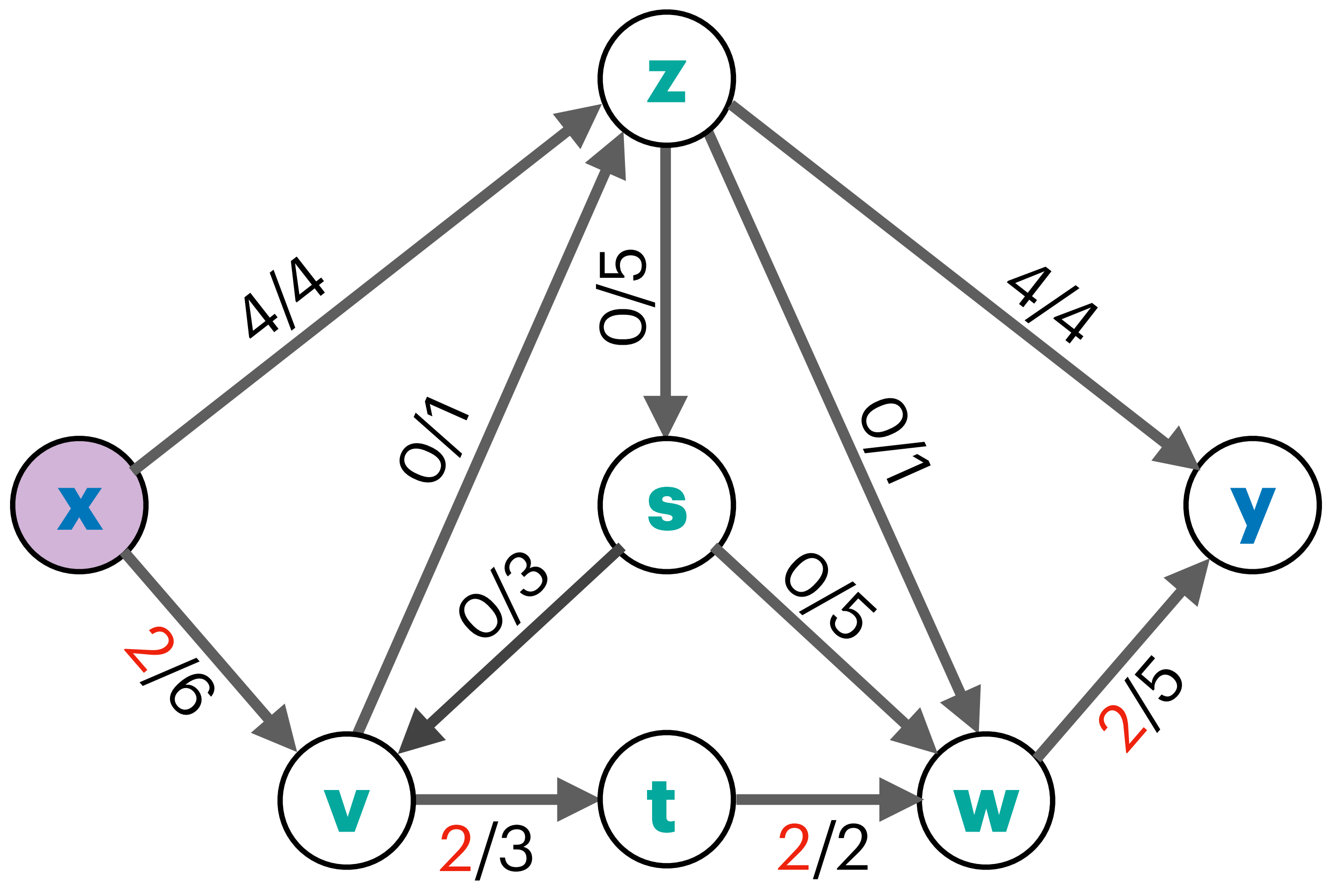


$p(y) = w \quad p(w) = t \quad p(t) = v$

$p(v) = x$

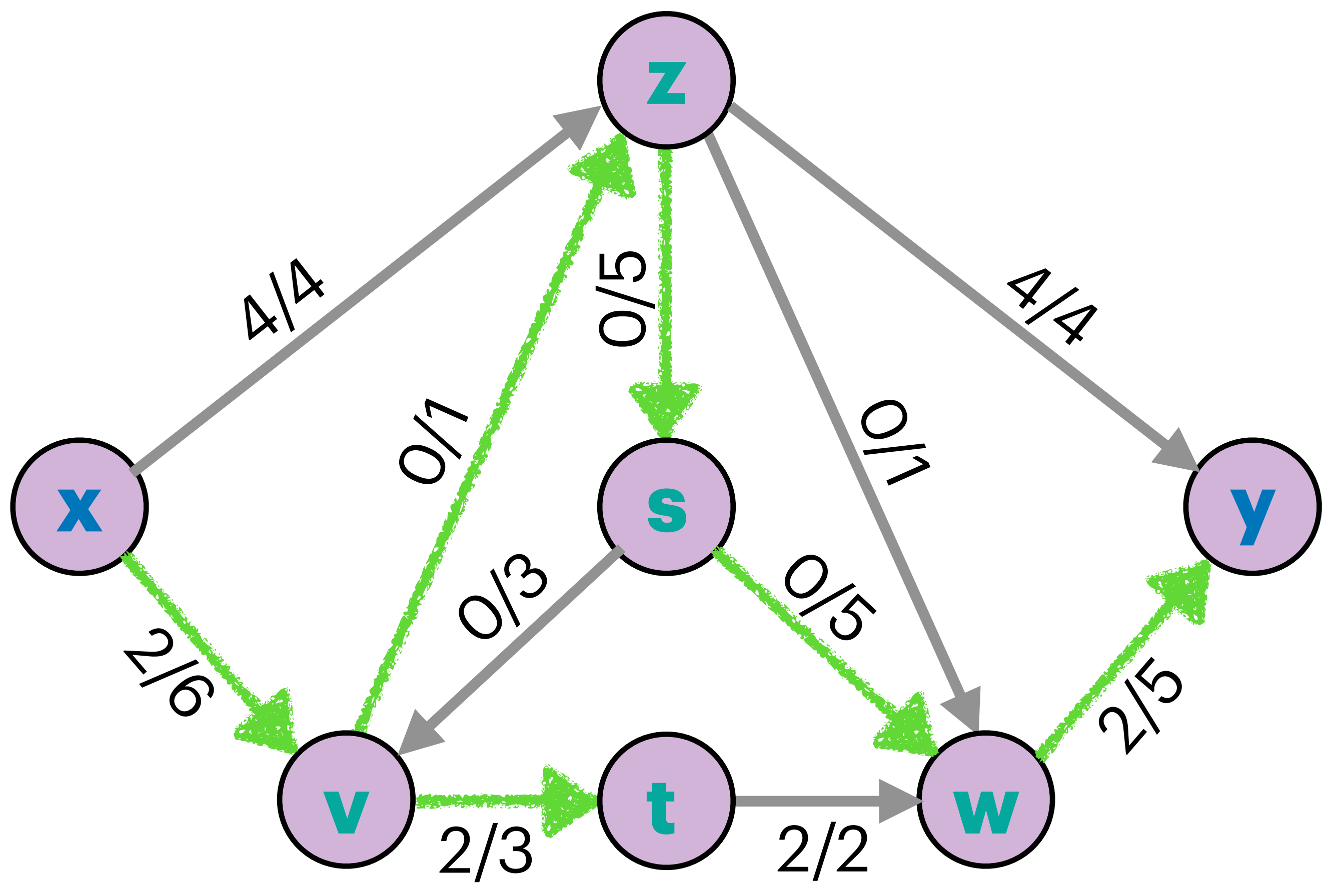
f-cammino insaturo

$\epsilon(P) = min\{6,3,2,5\} = 2$



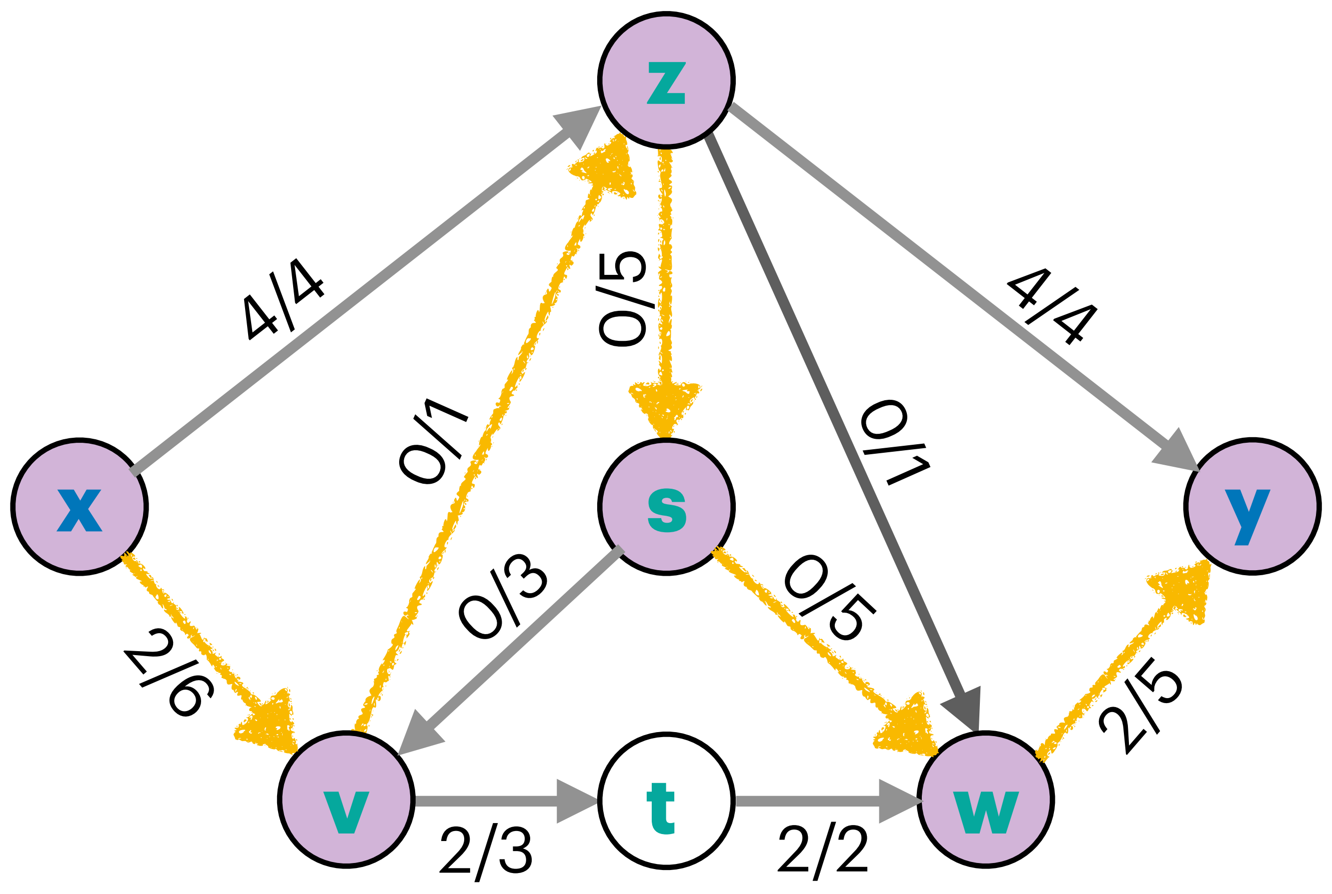
$X = \{x\}$

$p(v) = \emptyset \quad \forall v \in V(D)$



$X = \{x, v, t, z, s, w, y\},$

Albero f-insaturo massimale

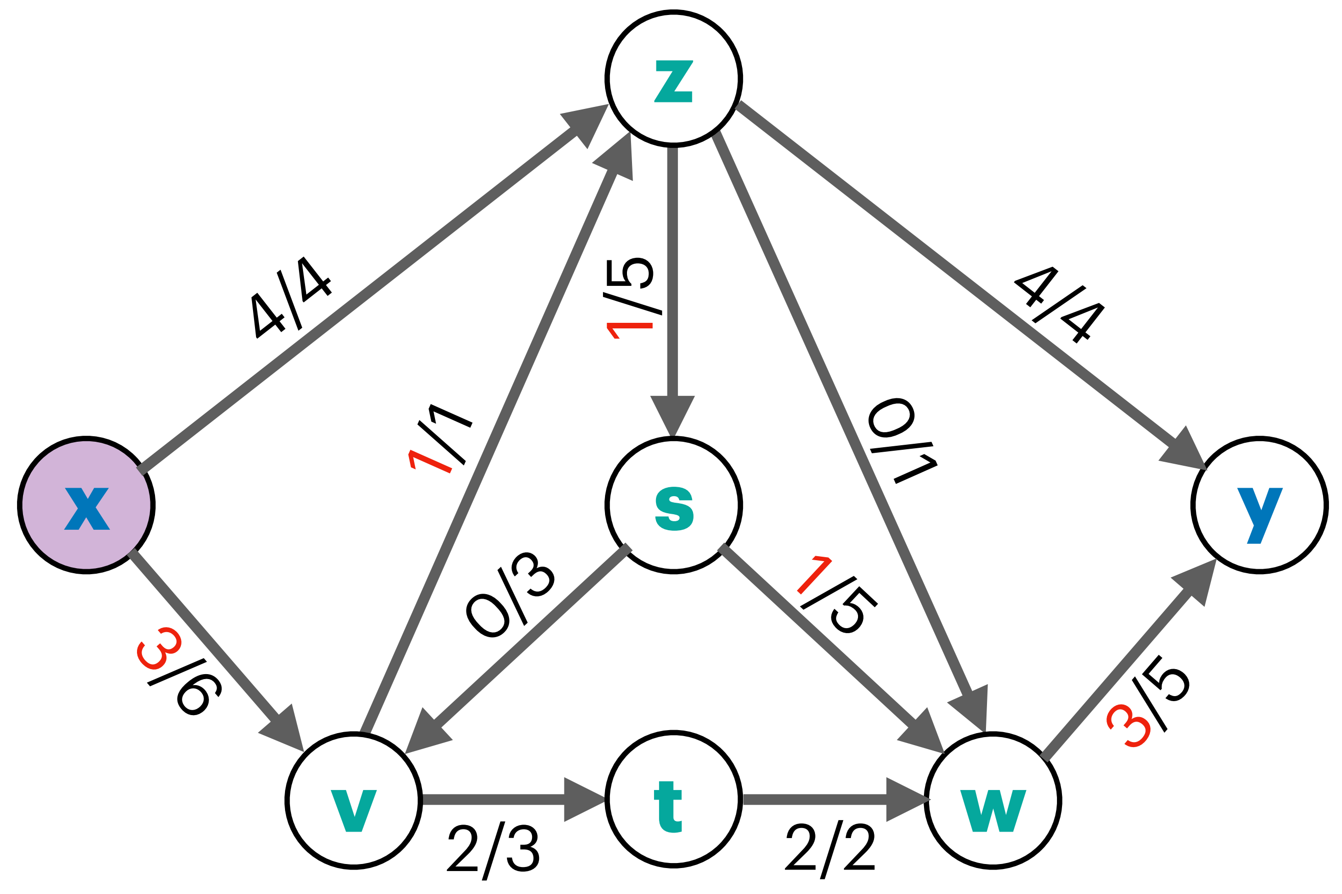


$p(y) = w \quad p(w) = t \quad p(t) = v$

$p(v) = x$

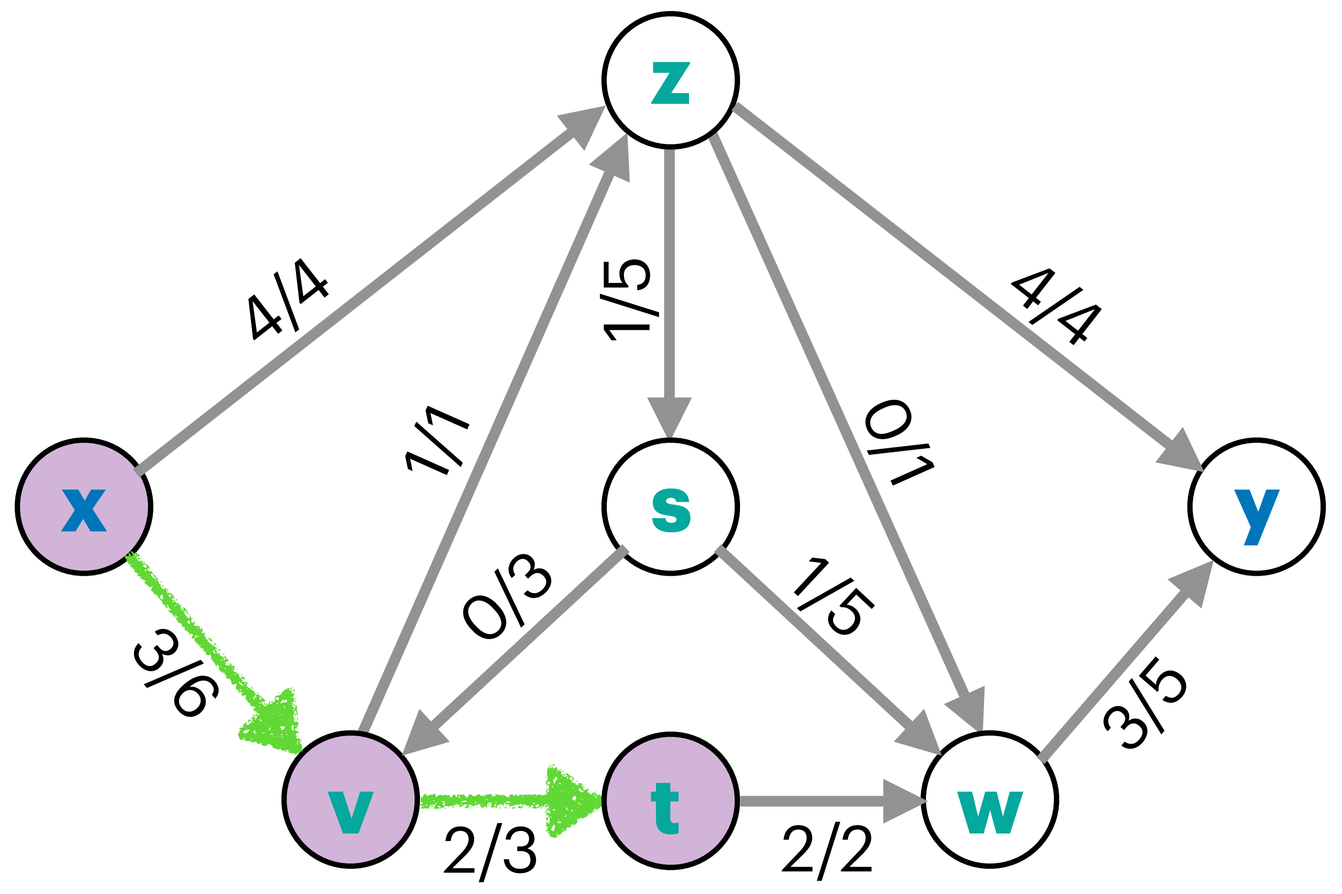
f-cammino insaturo

$\epsilon(P) = min\{2,1,5,5,2\} = 1$



$X = \{x\}$

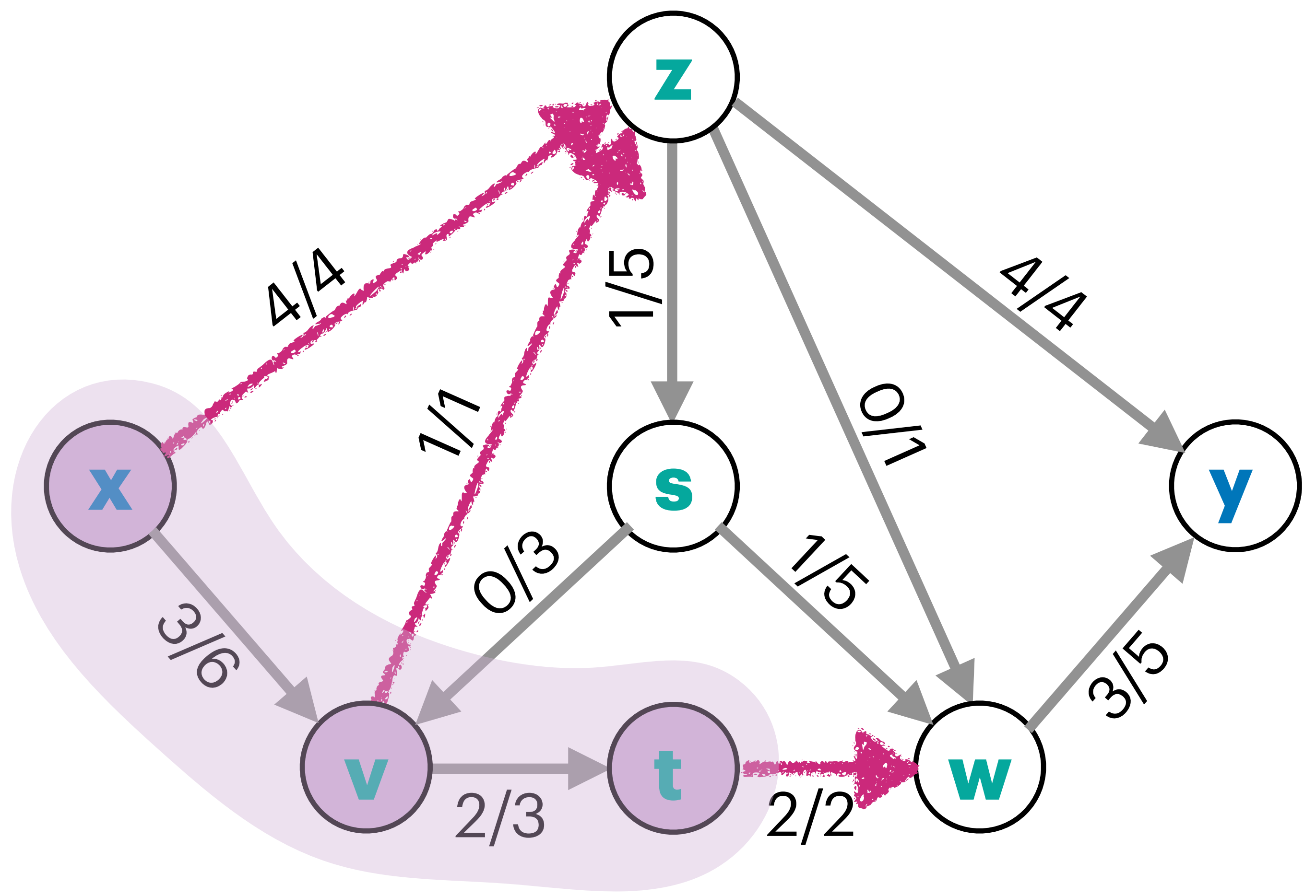
$p(v) = \varnothing \quad \forall v \in V(D)$



$X = \{x, v, t\}$

Albero f-insaturo massimale

$y \notin X$ $\;f$ è il flusso massimo.



$\partial^+(X) = \{(x, z); (v, z); (t, w)\}$ è il taglio minimo.

Il costo dell'algoritmo è di $O(m \times val(f))$ con f flusso massimo.

COMPLESSITA' COMPUTAZIONALE

METODO	COMPL. COMPUT.
Algoritmo di Ford-Fulkerson	$O(mval(f^*))$
Algoritmo di Edmonds-Karp	$O(nm^2)$
Algoritmo di Dinic con flusso bloccante	$O(n^2m)$
Algoritmo MPM	$O(n^3)$
Algoritmo di Dinic	$O(nmlog(n^2/m))$
Algoritmo KRT	$O\left(nmlog\left(\frac{m}{nlog(n)}n\right)\right)$
Algoritmo binario con flusso bloccante	$O(m(min\{n^{\frac{2}{3}}, \sqrt{n}\})log(\frac{n^2}{m})log(max(c)))$
Algoritmo di James B. Orlin e KRT	$O(nm)$