LABORATORIO DI RICERCA OPERATIVA 2022-2023

Esercitazione 8

Lezione 8

1. Algoritmo di Ford-Fulkerson

Riepilogo Lez. 7

$$
\begin{array}{llllll}
max & & & v & & \\
\sum_{j|(s,j)\in E} f_{sj} & & & -v & = & 0 \\
\sum_{j|(i,j)\in E} f_{ij} & - & \sum_{j|(j,i)\in E} f_{ji} & & = & 0 \quad i \in V, i \neq S,T \\
 & - & \sum_{i|(i,T)\in E} f_{T} & +v & = & 0 \\
 & 0 & \leq & f_{ij} & \leq & u_{ij} \quad (i,j) \in E
\end{array}
$$

Problema Max Flow

P cammino elementare $\quad S \rightarrow v_2 \cdots\cdots \bigcirc \cdots\cdots v_k \rightarrow T$

Se f è ammissibile con valore v e se

1. $u_{ij} - f_{ij} > 0 \quad \forall (i,j) \in P^+$ (archi in avanti non saturi)
2. $f_{ij} > 0 \quad \forall (i,j) \in P^-$ (archi all'indietro non scarichi)

allora P è un cammino aumentante su G rispetto f.

$$
\Delta^+ = \min_{(i,j)\in P^+}\{u_{ij} - f_{ij}\}, \quad \Delta^- = \min_{(i,j)\in P^-}\{f_{ij}\} \rightarrow \Delta = min\{\Delta^+, \Delta^-\} \quad f'_{ij} = \begin{cases} f_{ij} + \Delta & \forall (i,j) \in P^+ \\ f_{ij} - \Delta & \forall (i,j) \in P^- \\ f_{ij} & altrimenti \end{cases}
$$

è ammissibile; ad essa corrisponde un valore di flusso $v' = v + \Delta$.

Algoritmo di Ford-Fulkerson: basato sulla ricerca di cammini aumentanti

```
1. Inizializzazione
        Calcola una distribuzione ammissibile di flusso f
        ed il suo valore v

2. Iterazione
      (P,Δ)=cercaCamminoAumentante(f)
      if(P = ∅){
            return(f* = f, v* = v);
            exit;
      }
      else{ //esiste un cammino aumentante
            f=aggiornaFlussi(P,Δ)
            v = v + Δ;
            GOTO2
      }
```

Ricerca di un cammino aumentante → Algoritmo di visita di un grafo $\begin{cases} BFS \\ DFS \end{cases}$ → Albero di visita

Algoritmo di visita di un Grafo

forall($i \in V$ **)** $label[i] = 0;$ $pred[i] = 0$

$label[S] = 1$; $L=\{S\}$ /* visita S */
$pred[S] = \perp$ /*S non ha predecessori*/
while $(L \neq \emptyset)$ **{**
  $L = L - \{j\}$ /*Estrai un nodo j da L*/
  1) forall($(j,k) \in E$ **)**
    if($label[k] == 0$ **){** /*Visita gli adiacenti di j non ancora visitati*/
      $\boldsymbol{label[k] = 1}$
      $L = L \cup \{k\}$
      $pred[k] = j^+$ /*segnala che k è raggiungibile da j con l'arco diretto (j,k) */
    }
  2) forall($(k,j) \in E$**)**
    if($label[k] == 0$**){**
      $\boldsymbol{label[k] = 1}$
      $L = L \cup \{k\}$
      $pred[k] = j^-$ /*segnala che k è raggiungibile da j con un arco inverso (k,j)*/
    }
}

Come modificare l'algoritmo di visita per determinare un cammino aumentante?

Modifica dell'algoritmo di visita per determinare un cammino aumentante

Applichiamo la definizione di cammino aumentante (archi in avanti non saturi ed archi all'indietro non scarichi) ed eticchettiamo opportunamente i nodi visitati, sostituendo l'etichetta generica $label[j] = 1$, con

$incr[j]$ = massimo incremento di flusso che da S può giungere a j;

L'inizializzazione sarà

$incr[S] = +\infty$; $L = \{S\}$ $pred[S] = \perp$

La ricerca si può arrestare quando $T \in L$, anche se $L \neq \emptyset$

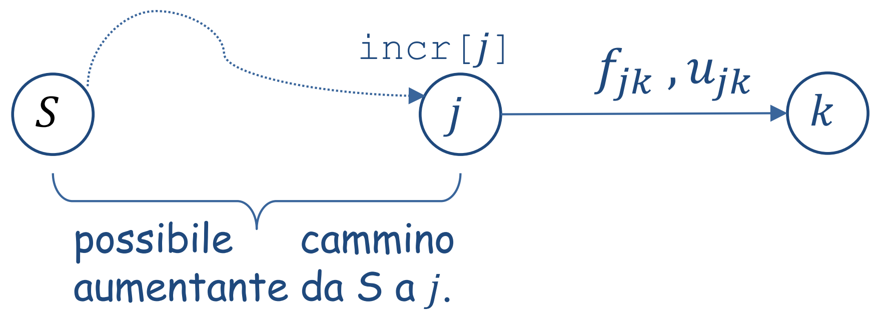
«Leggendo» ricorsivamente a ritroso il vettore $pred$, a partire da $pred[T]$ si ricostruisce il cammino aumentante. L'incremento di flusso lungo il cammino sarà $incr[T]$.

Algoritmo di visita di un Grafo per l'Algoritmo di F-F

1. Estrai un nodo j da L ($L = L - \{j\}$). Sia `incr[`j`]` la sua etichetta

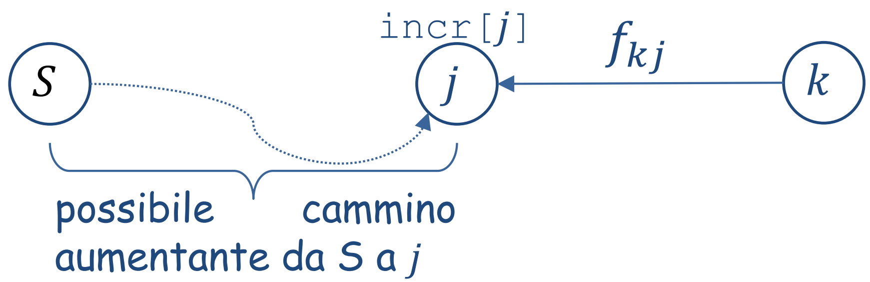
2. **forall(** $(j,k) \in E$)
   if(k `non è etichettato` **and** $u_{jk} - f_{jk} > 0$) {
   - a. `pred[`k`]=`j^+
   - b. `incr[`k`]=min{incr[`j`],`$u_{jk} - f_{jk}$`}`
   - c. $L = L \cup \{k\}$

   }

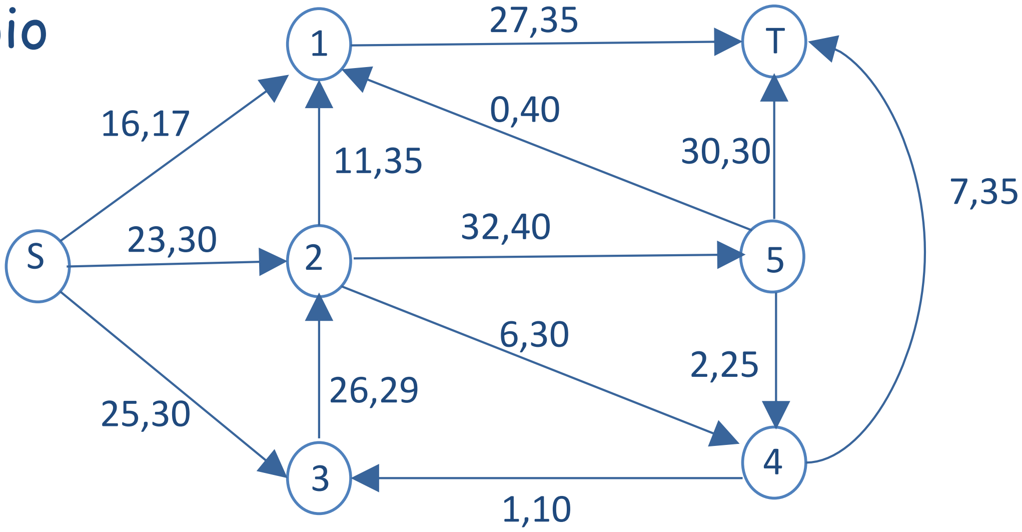


3. **forall(** $(k,j) \in E$)
   if(k `non è etichettato` **and** $f_{kj} > 0$) {
   - a. `pred[`k`]=`j^-
   - b. `incr[`k`]=min{incr[`j`],`f_{jk}`}`
   - c. $L = L \cup \{k\}$

   }



Esempio



$$v = \sum_{j|(S,j)\in E} f_{Sj}$$
$$= 16 + 23 + 25 = 64$$

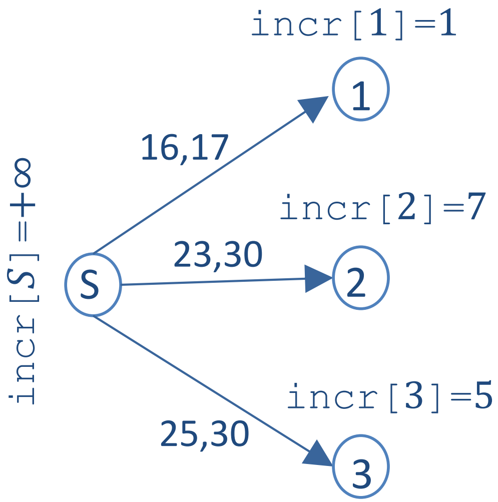
Visita DFS (Profondità)

incr[S]=+∞

$pred = [\perp \quad\quad\quad\quad\quad\quad\quad]$

$L = \{S\}$

Inizializzazione

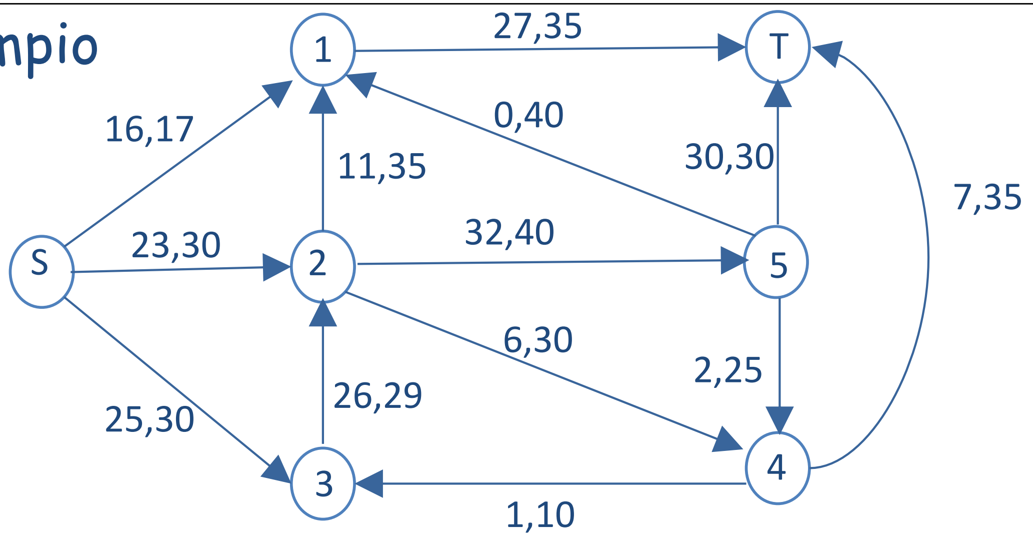


Etichetta gli adiacenti di S

$pred = [\perp \quad S^+ \quad S^+ \quad S^+ \quad\quad\quad]$

$L = \{1,2,3\}$

Esempio



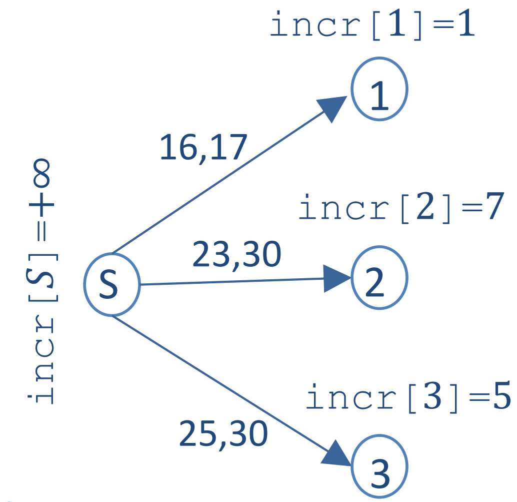
$$v = \sum_{j|(S,j)\in E} f_{Sj}$$
$$= 16 + 23 + 25 = 64$$

Visita DFS (Profondità)

La visita procede dal nodo che è in testa alla lista L ($nodo\ 1$): i nodi adiacenti ad 1 non etichettati (visitati) sono 5 e T:

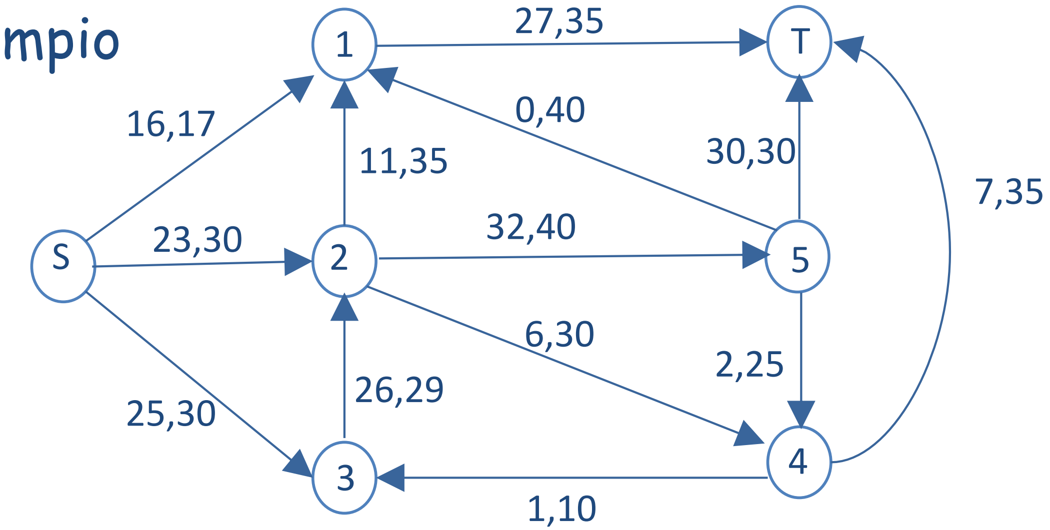
a) 5 non è etichettabile (raggiungibile) a partire dal nodo 1 perché $f_{51} = 0$.

b) T è etichettabile (raggiungibile) dal nodo 1 perché $u_{1T} - f_{1T} = 35 - 27 > 0$



$pred = [\perp \quad S^+ \quad S^+ \quad S^+ \qquad\qquad]$ $\quad L = \{1,2,3\}$

Esempio



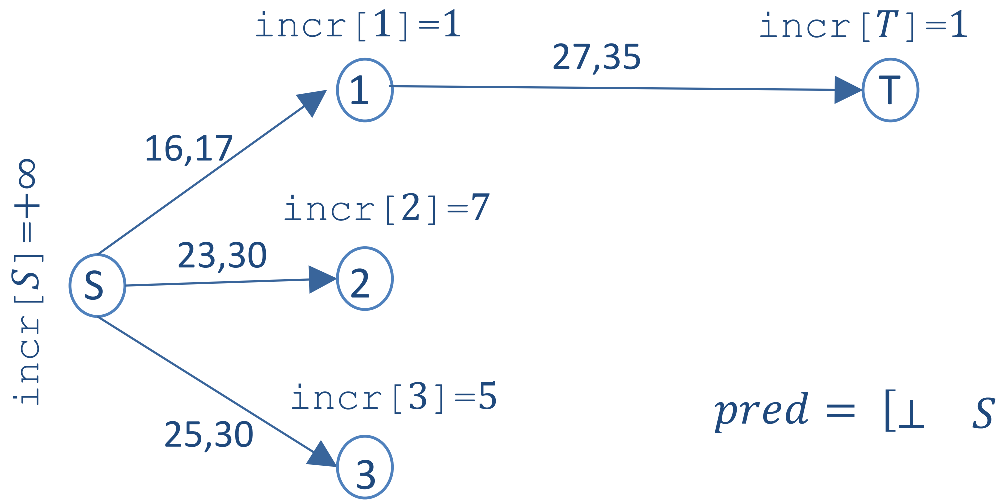
$$v = \sum_{j|(S,j)\in E} f_{Sj}$$
$$= 16 + 23 + 25 = 64$$

Visita DFS (Profondità)

La visita procede dal nodo che è in testa alla lista L ($nodo\ 1$): i nodi adiacenti ad 1 non etichettati (visitati) sono 5 e T:

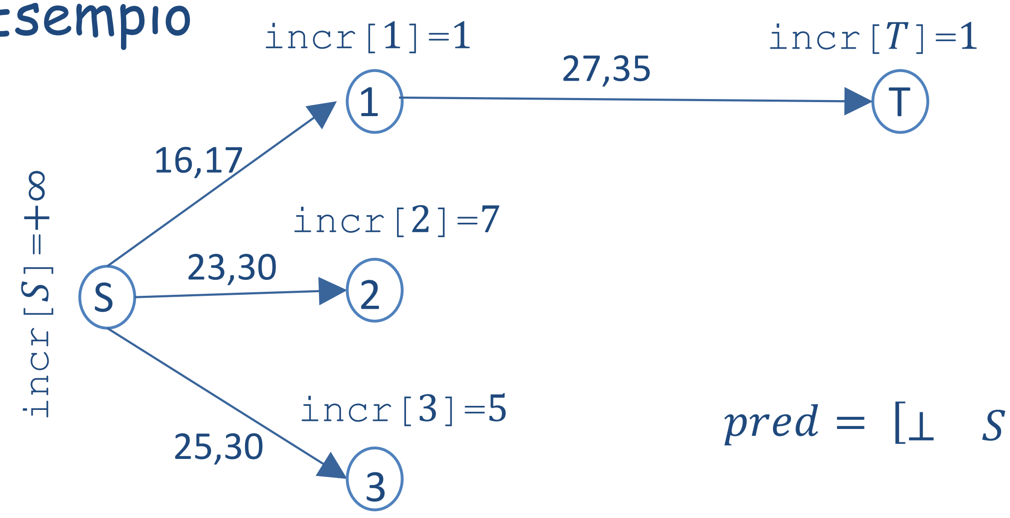
a) 5 non è etichettabile (raggiungibile) a partire dal nodo 1 perché $f_{51} = 0.$

b) T è etichettabile (raggiungibile) dal nodo 1 perché $u_{1T} - f_{1T} = 35 - 27 > 0$



$pred = [\perp \quad S^+ \quad S^+ \quad S^+ \qquad\qquad 1^+]$ $\quad L = \{T, 2{,}3\}$

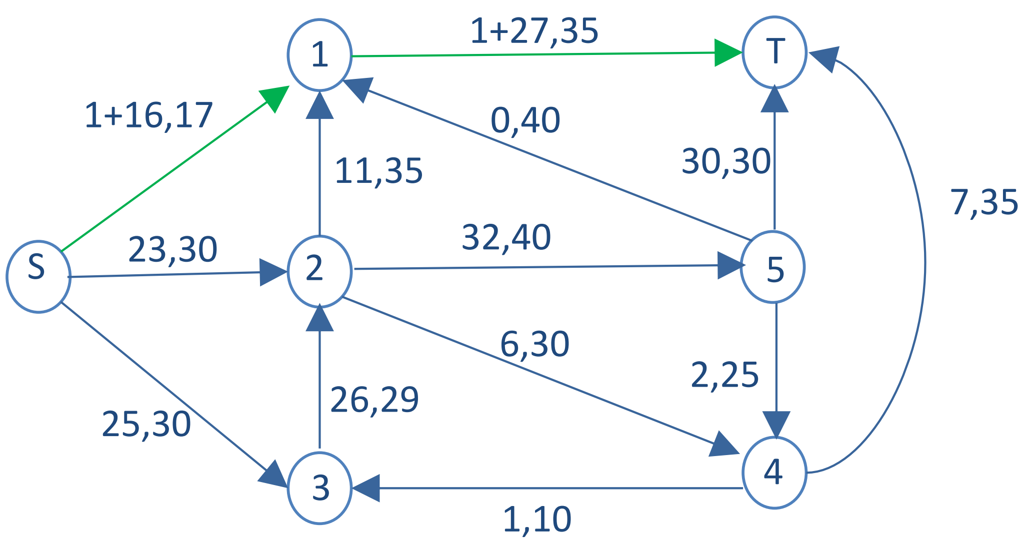
Esempio



$pred = [\perp \quad S^+ \quad S^+ \quad S^+ \qquad\qquad 1^+] \quad L = \{T, 2{,}3\}$

T è stato etichettato → è stato determinato un cammino aumentante con $\Delta = incr[T] = 1$

Si aggiornano i flussi solo sugli archi del cammino aumentante e si reitera



$v' = v + \Delta = 64 + 1$
$= 65$

Una forma Tabellare per l'Algoritmo di Ford-Fulkerson

Ogni iterazione dell'Algoritmo di F-F può essere, equivalentemente, rappresentata da una tabella con n+2 colonne (una per ciascun nodo + 2 di supporto) ed un numero di righe al più pari ad n-1.

Una riga di tale tabella rappresenta la propagazione dell'etichetta di un nodo j ai suoi adiacenti, secondo le regole precedentemente esposte.

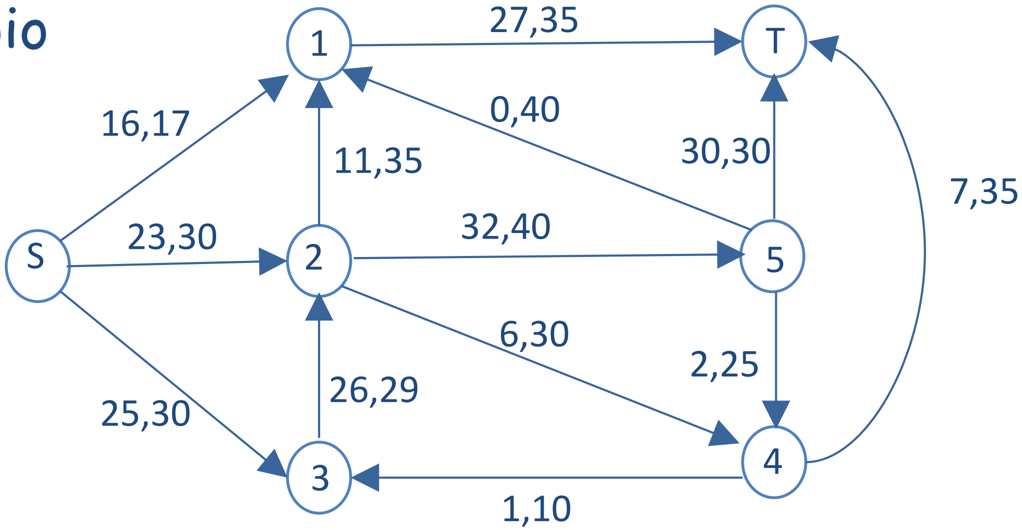
Inizializzazione

	S	1	j		k		T	L
	$[\perp, +\infty]$							$L = \{S\}$
S			$[pred[j], incr[j]]$					$L = L \cup \{j\}$
j					$[pred[k], incr[k]]$			$L = L \cup \{k\}$

Nodi estratti da L che propagano l'etichetta.

Etichetta propagata dal nodo j al nodo k

Esempio

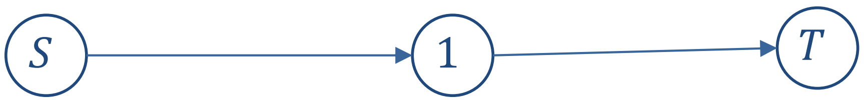


$$v = \sum_{j|(S,j)\in E} f_{Sj}$$
$$= 16 + 23 + 25 = 64$$

Ricerca di cammini aumentanti con visita DFS

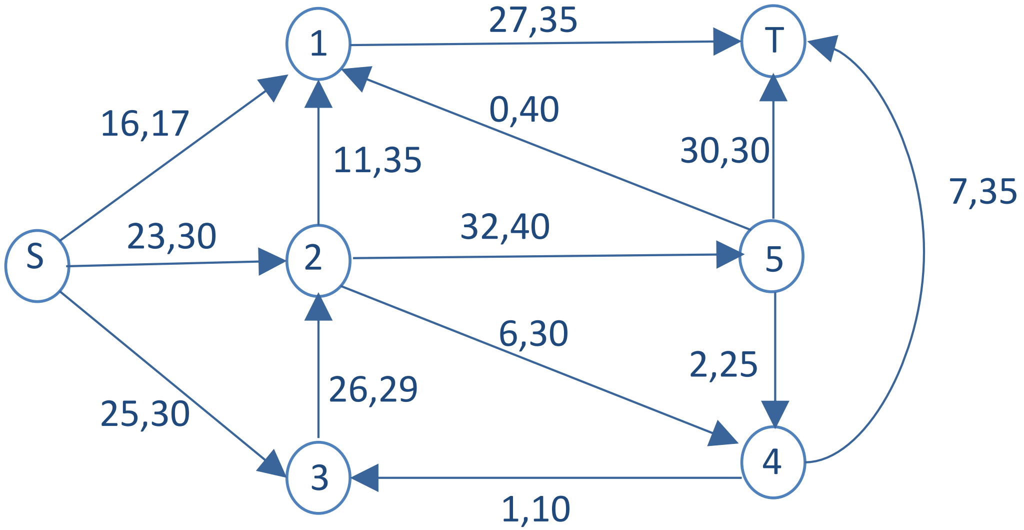
	S	1	2	3	4	5	6	T	L
	[⊥, +∞]								S
S		[S+,1]	[S+,7]	[S+,5]					1,2,3
1								[1+,1]	T,2,3

T è stato etichettato, è stato determinato un cammino aumentante: $\Delta = incr[T] = 1$

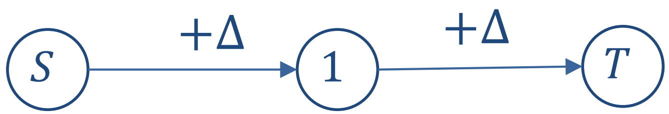


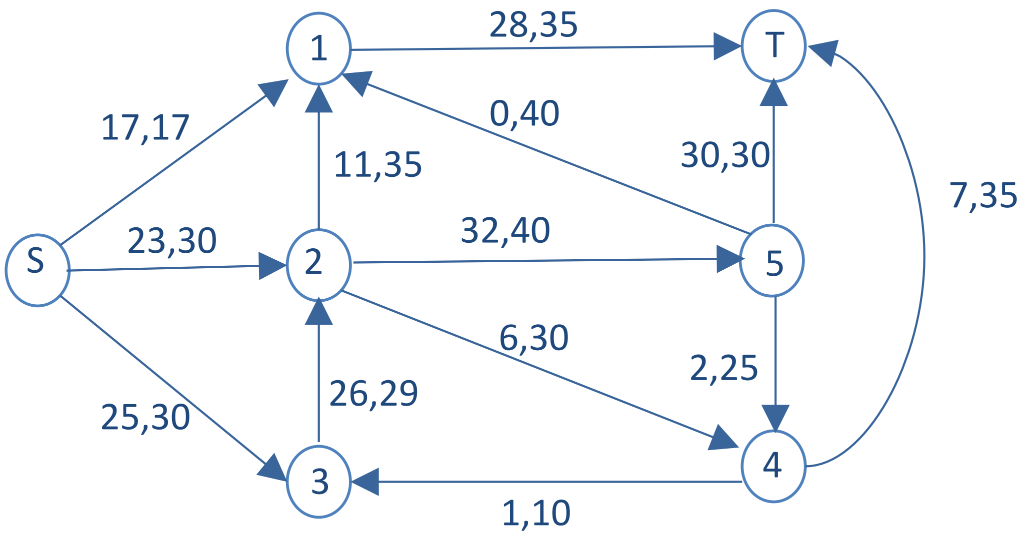
Leggendo $pred[1]$ Leggendo $pred[\mathrm{T}]$

Esempio



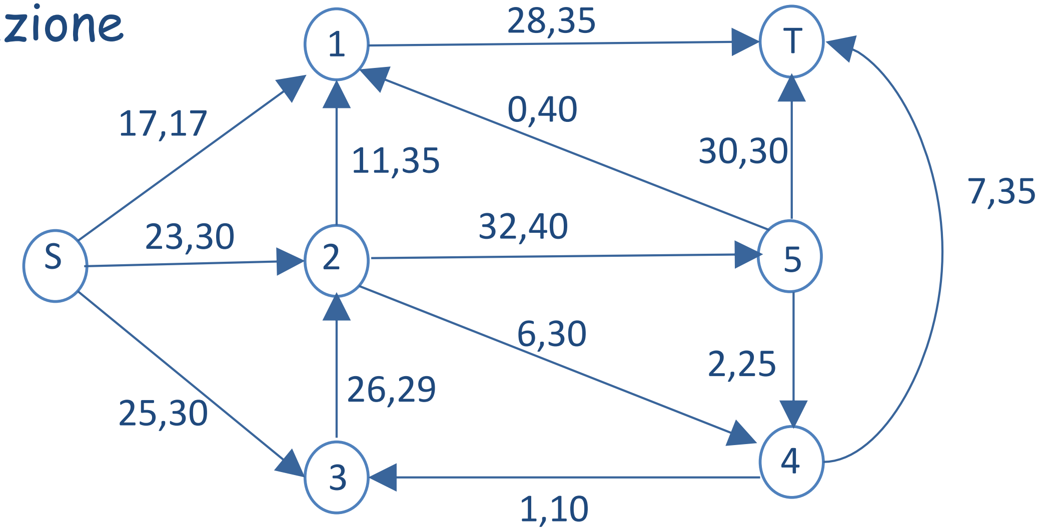
Aggiornamento soluzione





$$v = v + \Delta = 64 + 1 = 65$$

Nuova iterazione



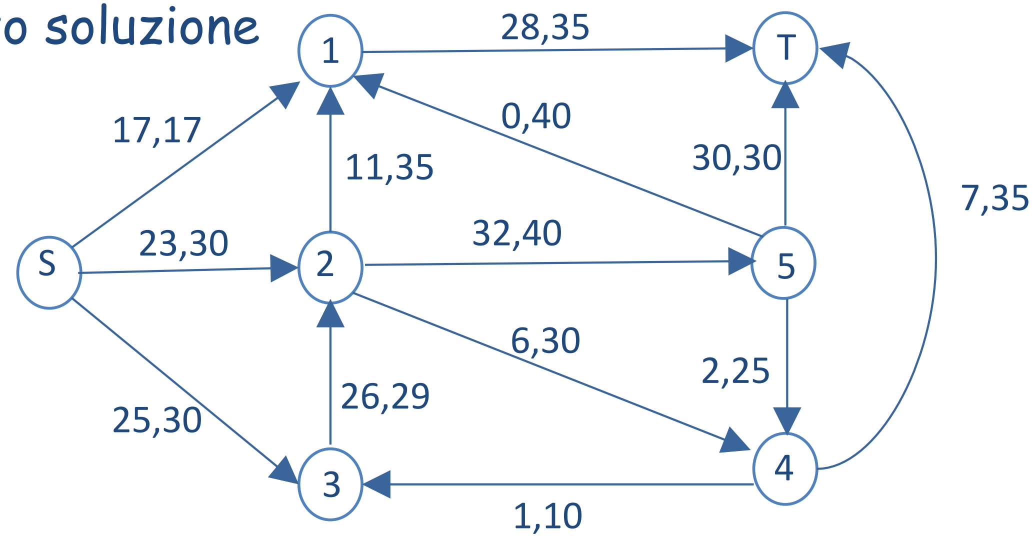
$v = 65$

	S	1	2	3	4	5	6	T	L
	[⊥, +∞]								S
S			[S+,7]	[S+,5]					2,3
2		[2+,7]			[2+,7]	[2+,7]			1,4,5,3
1								[1+,7]	T,4,5,3

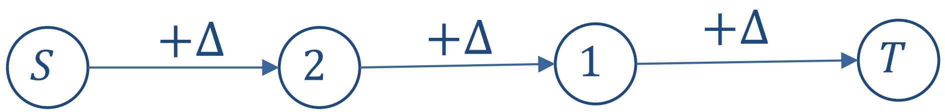
T è stato etichettato, è stato determinato un cammino aumentante: $\Delta = incr[T] = 7$

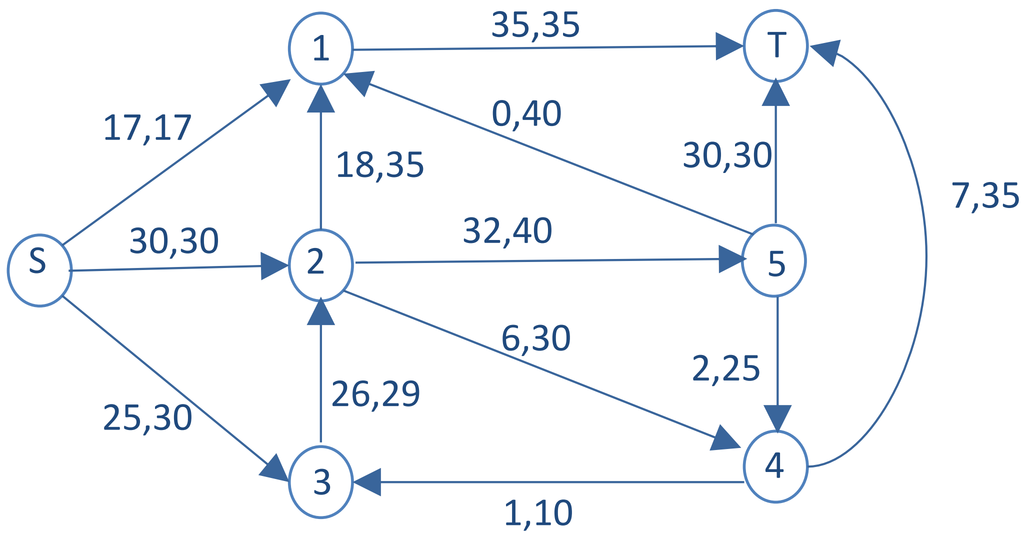
Leggendo $pred[2]$ Leggendo $pred[1]$ Leggendo $pred[T]$

Aggiornamento soluzione



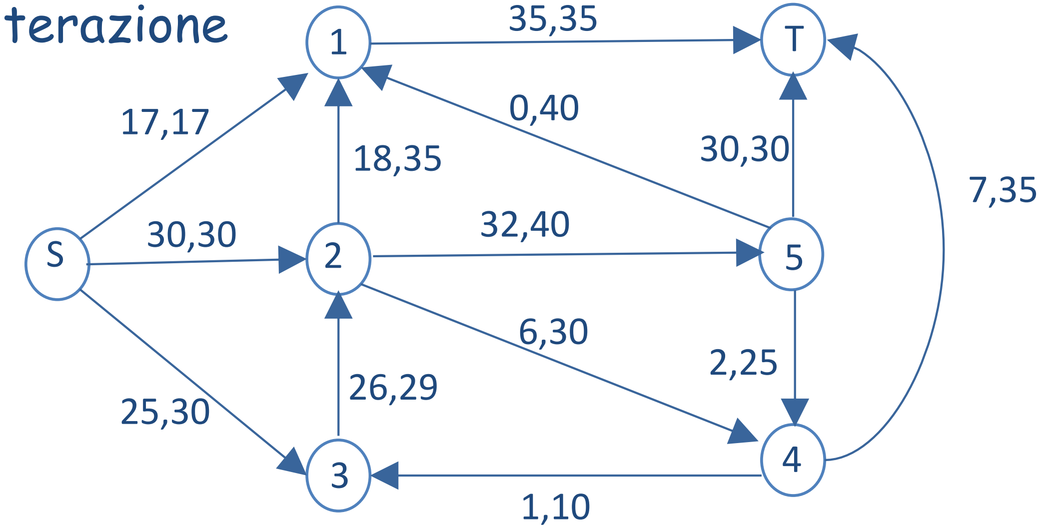
$v = 65$





$v = 72$

Nuova iterazione

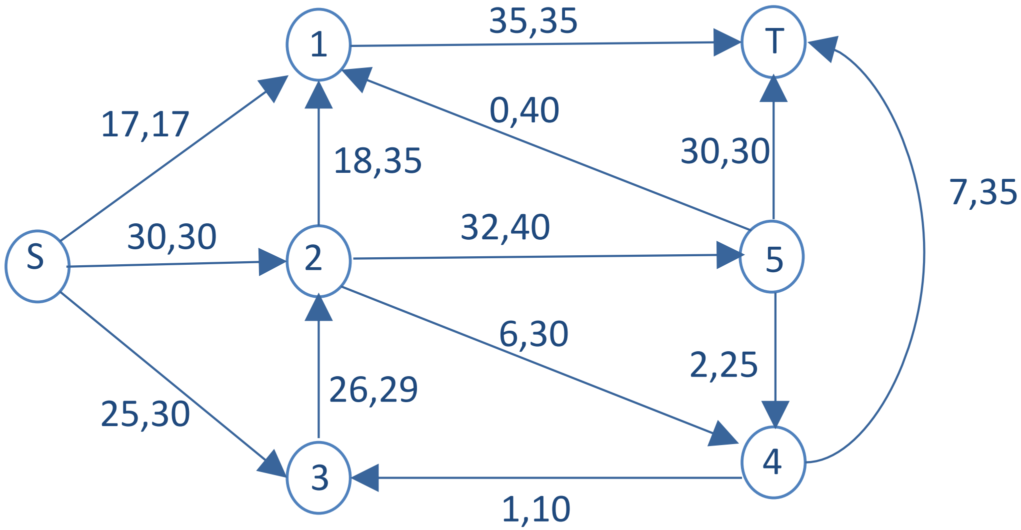


$v = 72$

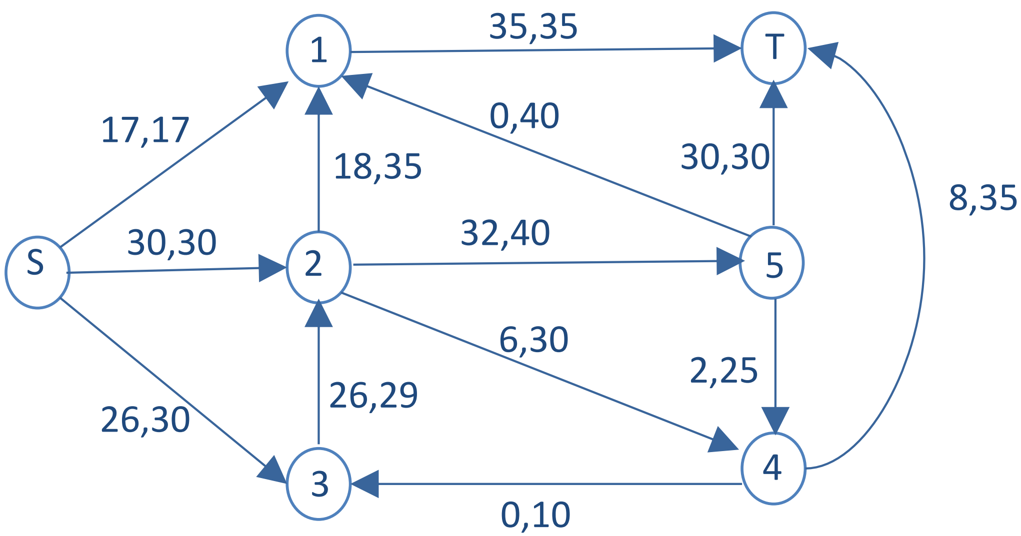
	S	1	2	3	4	5	6	T	L
	[⊥, +∞]								S
S				[S+,5]					3
3			[3+,3]		[3-,1]				2,4
2		[2+, 3]				[2+,3]			1,5,4
1									5,4
5									4
4								[4+,1]	T

T è stato etichettato, è stato determinato un cammino aumentante: $\Delta = incr[T] = 1$

Aggiornamento soluzione

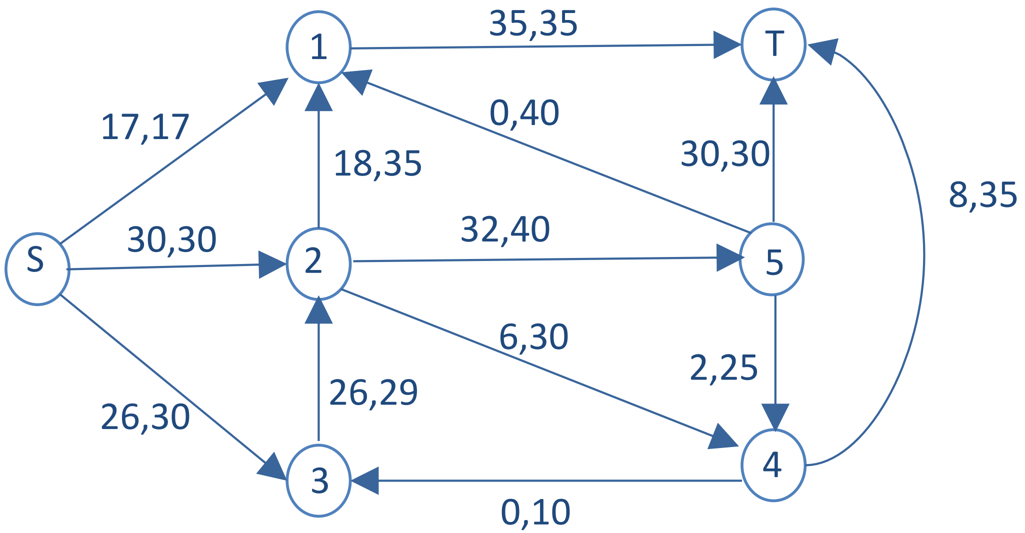


$v = 72$



$v = 73$

Nuova Iterazione



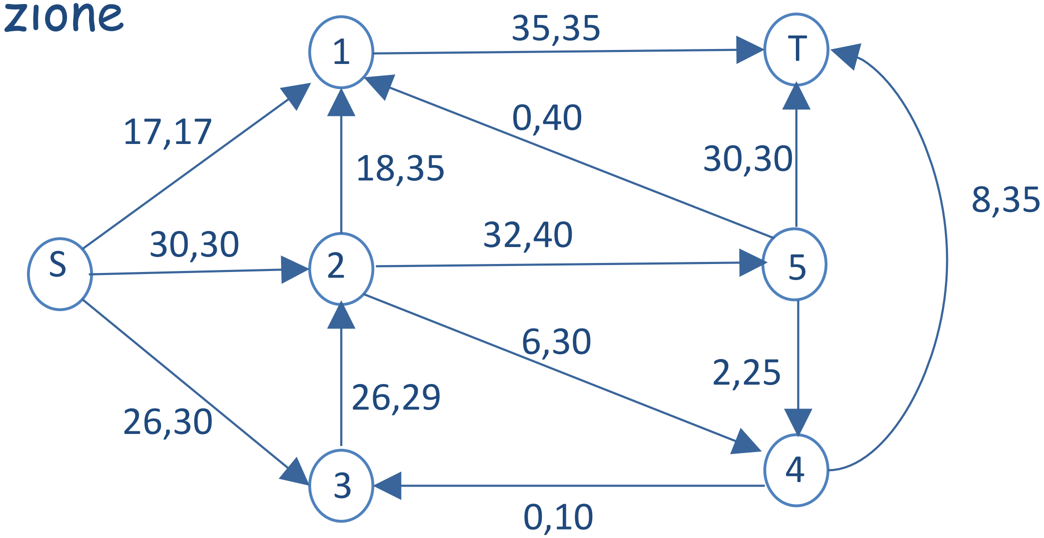
$v = 73$

	S	1	2	3	4	5	6	T	L
	$[\perp, +\infty]$								S
S				[S+,4]					3
3			[3+,3]						2
2		[2+, 3]			[2+,3]	[2+,3]			1,4,5
1									4,5
4								[4+,3]	T,5

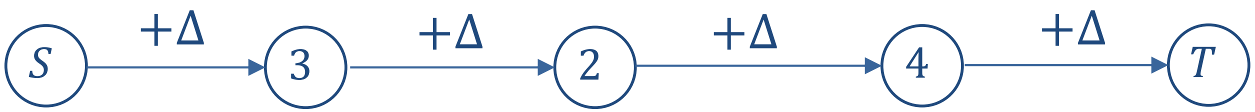
T è stato etichettato, è stato determinato un cammino aumentante: $\Delta = incr[T] = 3$

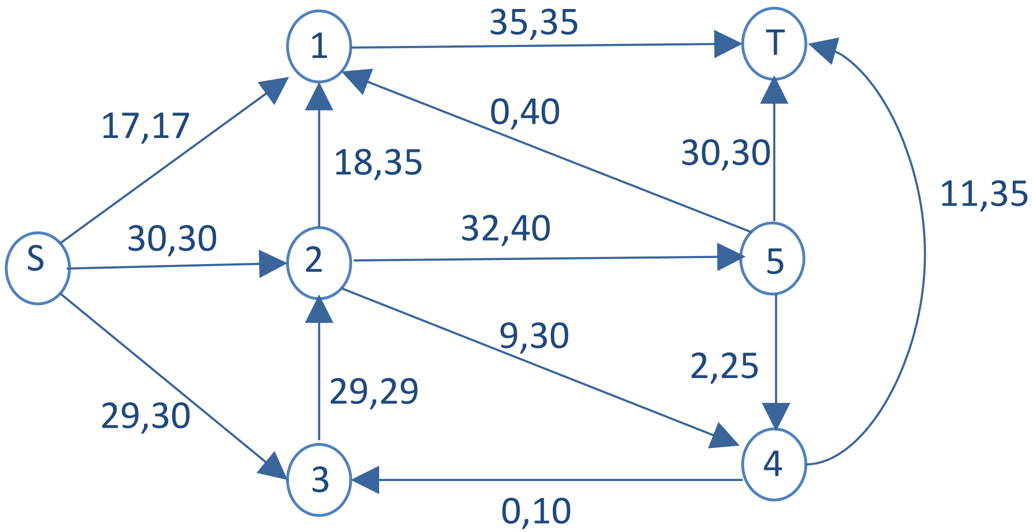
$S \to 3 \to 2 \to 4 \to T$

Aggiornamento soluzione



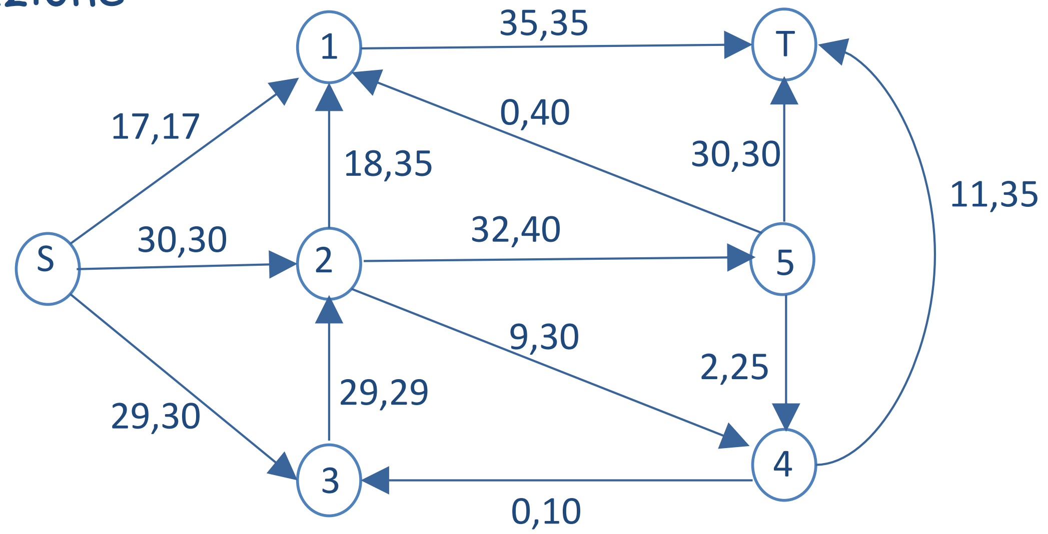
$v = 73$





$v = 76$

Nuova iterazione



$v = 76$

	S	1	2	3	4	5	6	T	L
	[⊥, +∞]								S
S				[S+,1]					3
3									

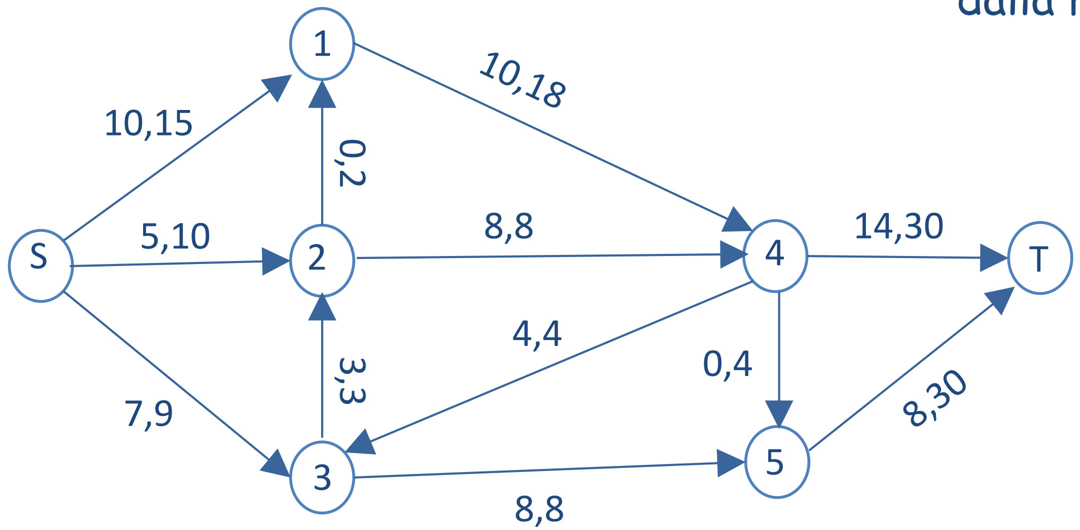
Non è più possibile etichettare il terminale perché non esistono cammini aumentanti rispetto al flusso corrente (condizione sufficiente di ottimalità e criterio di arresto per l'algoritmo di F.F): il flusso corrente è quello ottimo.

$$v^* = 76$$

$f_{ij}^* =$ flusso sugli archi nell'ultimo grafo

Esercizio

E' assegnato il problema di MaxF, definito dalla rete di seguito riportata.



1. Scrivere la formulazione matematica del problema in esame
2. Verificare che la distribuzione di flusso assegnata sia ammissibile e calcolarne il valore
3. Determinare il valore del flusso che è possibile trasferire da S a T, applicando l'Algoritmo di F.F. con visita in ampiezza.