Esercitazione 6
Ancora sul Network Flow

Problema 14 (appello 28/09/2015)

Un'importante azienda di sviluppo software ha n progetti da portare a termine entro la fine dell'anno. Il manager dell'azienda stima che ogni progetto $i \in [n]$ necessiti di un certo numero $d_i \in \mathbb{N}$ di ore di lavoro da parte dei programmatori. In tutto l'azienda impiega m programmatori e ogni programmatore $j \in [m]$, in base alle sue competenze, è in grado di lavorare su un sottoinsieme $P_j \subseteq [n]$ dei progetti. Considerando che fino alla fine dell'anno ogni programmatore potrà svolgere k ore di lavoro in tutto, il manager deve decidere se è possibile portare a termine tutti i progetti con il personale attualmente impiegato oppure se bisogna assumere urgentemente nuovi programmatori.

Descrivere un algoritmo che prende in input $\{d_1, \cdots, d_n\}$, $\{P_1, \cdots, P_k\}$ e k e fornisce una risposta in tempo polinomiale.

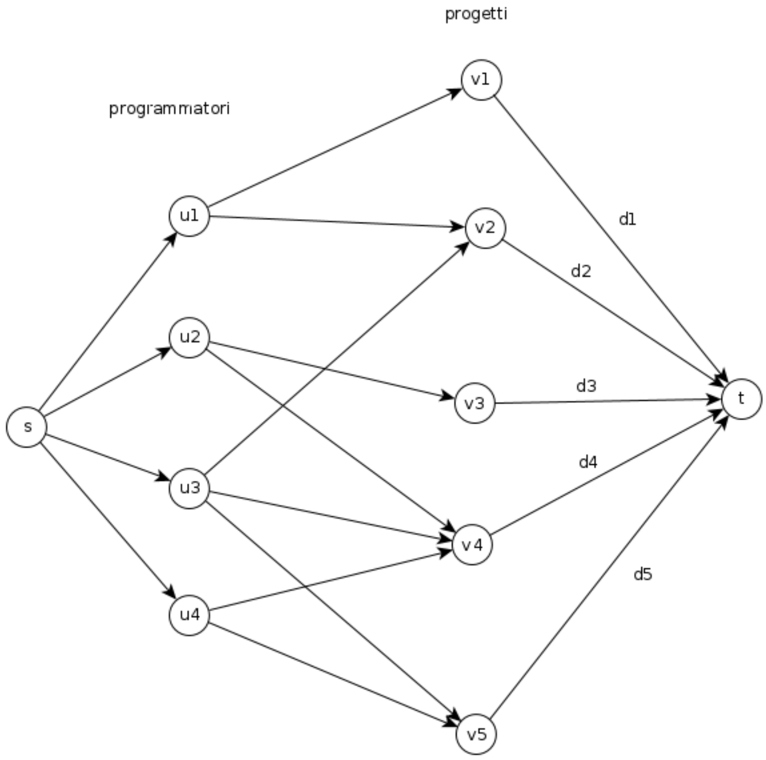
Soluzione Al fine di trovare un'assegnazione di programmatori a progetti che, rispetti i vincoli imposti dal problema e permette di completare tutti i progetti, effettueremo una riduzione al problema *max-flow* su una rete costruita a partire da un'istanza del problema di partenza.

Nella nostra rete ci sarà un nodo u_i per ogni programmatore i ed un nodo v_j per ogni progetto j, sono inoltre presenti un nodo sorgente s e un nodo target t. Per quanto riguarda gli archi ci sarà un arco diretto dal nodo s verso ogni nodo programmatore con capacità k. Ogni nodo programmatore è collegato con un arco diretto di capacità k ai nodi che rappresentano i progetti a cui lui può lavorare. Infine tutti i nodi progetto sono collegati al nodo t con un arco diretto con capacità pari al loro tempo di completamento.

Riassumendo definiamo la nostra rete $G = (V, E)$ nel seguente modo:

- per i nodi abbiamo: $V = \{s, t\} \cup \{u_1, \cdots, u_n\} \cup \{v_1, \cdots, v_m\}$
- per gli archi si ha che E include:
  - per $i = 1, \cdots, n$, si hanno gli archi (s, u_i) con capacità k
  - per $i = 1, \cdots, n$ e $j = 1, \cdots, m$ si ha l'arco (u_i, v_j) con capacità k se e solo se il programmatore i può lavorare sul progetto j
  - per $j = 1, \cdots, m$ si hanno gli archi (d_j, t) con capacità d_j

Data un'istanza in cui si hanno 4 programmatori e 5 progetti in cui $P_1 = \{1, 2\}, P_2 = \{3, 4\}, P_3 = \{2, 4, 5\}, P_4 = \{4, 5\}$, (non esplicitiamo i valori k e d), la funzione di riduzione genera la rete riportata nella figura seguente. (Gli archi senza valore associato hanno capacità k).



Ora che abbiamo definito come opera la funzione di riduzione facciamo vedere come possiamo trovare una soluzione al problema dato trovando un flusso massimo su G che satura il taglio composto da tutti i nodi tranne t. Al fine di alleggerire la notazione poniamo $D = \sum_{i=1}^{m} d_i$.

Claim. $\exists$ un'assegnazione di programmatori a progetti ammissibile $\iff$ $\exists$ un flusso massimo f^* su G tale che $v(f^*) = D$.

Dimostriamo l'affermazione precedente:

- $\Longrightarrow$

  In questo caso facciamo vedere come a partire da un'assegnazione ammissibile di programmatori a progetti è possibile definire un flusso massimo ammissibile su G di valore D e quindi che satura il taglio composto da tutti i nodi eccetto t.

  Sia $x_{i,j}$ il numero di ore in cui il programmatore i è stato assegnato al progetto j nell'assegnazione in nostro possesso, poniamo $f(u_i, v_j) = x_{i,j}$. Il vincolo di capacità su tale arco risulta essere rispettato in quanto, questa era posta a k e dato che l'assegnazione è ammissibile, sicuramente non avremo che un programmatore è stato assegnato ad un progetto per più di k ore.

  Per ogni nodo u_i si avrà che $f^{out}(u_i) = \sum_{j \in P_i} x_{i,j} \leq k$, quindi è possibile rispettare il vincolo di conservazione sul flusso ponendo $f(s, u_i) = f^{out}(u_i)$, il vincolo sulle capacità degli archi uscenti da s è rispettato in quanto questa era stata posta a k.

  Per ogni nodo v_j si avrà che $f^{in}(v_j) = \sum_{i=1}^{n} x_{i,j}$, tale valore sarà pari a d_j in quanto l'assegnazione è ammissibile e quindi ha assegnato ad ogni progetto il numero di ore lavoro necessarie per il suo completamento, è possibile allora rispettare il vincolo di conservazione ponendo $f(v_j, t) = d_j$, il vincolo sulla capacità degli archi entranti in t risulta essere rispettato ed inoltre si ha che tutti tali archi sono saturi dandoci un valore totale di flusso D.

- $\Longleftarrow$

  Mostriamo ora come da un flusso massimo ammissibile di valore D è possibile ricavare un'assegnazione di programmatori a progetti ammissibile. Dopo aver fatto "girare" un algoritmo di $max - flow$ sulla nostra rete avremo che alcuni archi del tipo (u_i, v_j) avranno flusso maggiore di 0, indichiamo tale quantità di flusso con $x_{i,j}$. Costruiamo l'assegnazione richiesta assegnando il programmatore i al progetto j per $x_{i,j}$

ore. Banalmente i programmatori saranno assegnati solo a progetti per cui sono competenti in quanto nella rete erano collegati solo a nodi relativi a tali progetti.

Andiamo ora a mostrare che ogni lavoratore non lavorerà per più di k ore. Osserviamo che essendo la capacità sugli archi (s, u_i) pari a k abbiamo che per ogni generico programmatore u_i, $f^{in}(u_i) \leq k$ e dato che il flusso è ammissibile avremo che $f^{out}(u_i) = f^{in}(u_i)$, in definitiva ogni programmatore non sarà assegnato per più di k ore.

Infine mostriamo come ad ogni progetto vengono assegnati programmatori per un numero totale di ore necessario a portarlo a termine. Essendo nell'ipotesi che il flusso f^* è di valore D abbiamo che tutti gli archi del tipo (v_j, t) sono saturi, quindi possiamo affermare che per ogni j, $f^{out}(v_j) = d_j$, dato che il flusso è ammissibile si ha anche $f^{in}(v_j) = d_j$ assicurandoci così che tutti i progetti abbiano ricevuto il numero di ore necessarie. □

Quindi ricapitolando, avendo mostrato che il problema è riducibile ad un problema di max-flow, al fine di trovare una soluzione, possiamo trasformare l'istanza data in un'istanza di max-flow, trovare un flusso massimo sulla rete ottenuta, e da tale flusso ricavare la soluzione per il problema di partenza come descritto nel secondo punto della dimostrazione precedente.

Diamo infine una veloce analisi sulla complessità della soluzione proposta mostrando come questa risulti polinomiale nella dimensione dell'istanza del problema di partenza (come richiesto):

Per quanto riguarda la rete G ottenuta si ha che: $|V| = n+m+2$ mentre $|E| = n+m+\sum_{i=1}^{n} |P_i|$, osservando che ogni programmatore può lavorare al più in tutti i progetti abbiamo che $\sum_{i=1}^{n} |P_i| \leq nm$ quindi possiamo concludere che G ha dimensione totale $O(nm)$, tale complessità risulta essere anche la complessità di costruzione della rete. Una volta costruita la rete possiamo applicare l'algoritmo *max-flow* di Edmond-Karp (quello che seleziona ogni volta il cammino aumentante di lunghezza minima). Tale algoritmo in una rete connessa ha complessità $O(|E|^2|V|)$ quindi nel nostro caso si ha complessità $O((nm)^2(n+m))$. Quindi in definitiva, considerando sia la complessità di costruzione della rete che la complessità per il calcolo del flusso massimo possiamo affermare di riuscire a trovare una soluzione al problema di partenza in tempo totale $O((nm)^2(n+m))$.

Problema 15 (appello 18/06/2014)

Sia $C = \{C_1, \cdots, C_n\}$ l'insieme dei corsi impartiti in un Corso di Laurea, ciascuno insegnato da un docente distinto (ossia nel Corso di Laurea insegnano n docenti). Per $i = 1, \cdots, n$, la durata del corso C_i è di d_i ore; indichiamo con $D = \{d_1, \cdots, d_n\}$ l'insieme delle durate dei corsi.

I corsi vengono impartiti nell'insieme di giorni $G = \{g_1, \cdots, g_k\}$, per 4 ore al giorno.

Ciascun docente può richiedere di non essere impiegato in alcune ore di taluni giorni: ad esempio, il docente del corso di Algoritmi richiede di non tenere le sue lezioni né il giorno g_1 alla seconda ora, né il giorno g_4 alla terza ora. Formalmente, questo viene espresso associando al docente i, $1 \leq i \leq n$, un insieme L_i di coppie: se $(g_j, o_h) \in L_i$, allora il docente del corso C_i non può tenere le sue lezioni il giorno $g_j \in G$ all'ora $o_h \in \{1, 2, 3, 4\}$.

Una sola aula è a disposizione del Corso di Laurea.

Un *orario* è un assegnazione $o : G \times \{1, 2, 3, 4\} \to C$ che specifica in quali giorni e ore è insegnato ciascun corso.

Un orario è *ammissibile* se rispetta i vincoli seguenti:

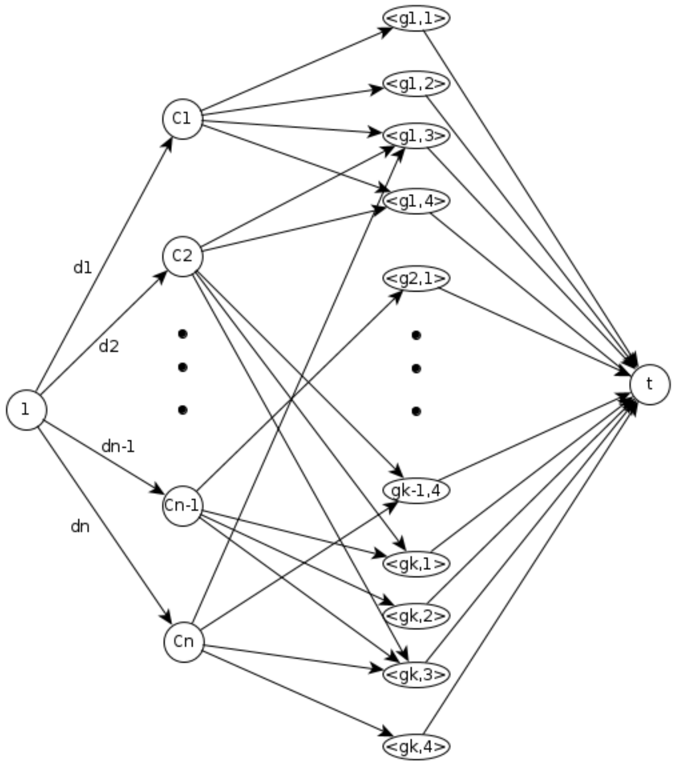
- le richieste di ogni docente (espresse tramite gli insiemi L_i) vengono rispettate;
- in ciascuna ora di ciascun giorno, l'aula è assegnata ad al più un corso;
- per $i = 1, \cdots, n$, al corso C_i sono assegnate esattamente d_i ore.

Definire un algoritmo polinomiale che, dati C, D, G e $\{L_i : 1 \leq i \leq n\}$, determina se esiste un orario ammissibile e, in caso affermativo, lo restituisce. Presentare anche l'analisi di complessità dell'algoritmo proposto.

Soluzione Come nell'esercizio precedente andremo a ridurre il problema ad un problema di *max-flow*. Data un'istanza del problema definiamo la rete $G = (V, E)$ nel seguente modo: c'è un nodo C_i per ogni corso i ed un nodo $\langle c_j, l\rangle$ per ogni combinazione *giorno-ora* , sono inoltre presenti un nodo sorgente s ed un nodo target t. Aggiungiamo un arco dal nodo s verso ogni nodo corso C_i ponendo capacità pari a d_i, colleghiamo ogni nodo *corso* con i nodi che rappresentano le combinazioni *giorno-ora* in cui il docente del corso è disposto a fare lezione, infine colleghiamo ogni nodo *giorno-ora* al nodo t con un arco diretto di capacità 1. Riassumendo definiamo la nostra rete $G = (V, E)$ nel seguente modo:

- per i nodi abbiamo: $V = \{s, t\} \cup \{C_1, \cdots, C_n\} \cup \{\langle g_1, 1\rangle, \cdots \langle g_k, 4\rangle\}$ dove il generico nodo $\langle g_j, l\rangle$ sta ad indicare giorno j ora l
- per gli archi si ha che E include:
  - per $i = 1, \cdots, n$, si hanno gli archi (s, C_i) con capacità d_i
  - per $i = 1, \cdots, n$, $j = 1, \cdots, k$ e $l = 1, \cdots, 4$ si ha l'arco $(C_i, \langle g_j, l\rangle)$ con capacità 1 se e soltanto se la coppia $(g_j, l) \notin L_i$
  - per $j = 1, \cdots, k$ e $l = 1, \cdots, 4$ si hanno gli archi $(\langle g_j, l\rangle, t)$ con capacità 1

Nella figura seguente è rappresentata uno schema della rete appena descritto (gli archi senza valore associato hanno capacità 1):



Ora che abbiamo definito come opera la funzione di riduzione facciamo vedere come possiamo trovare una soluzione al problema dato trovando un $max - flow$ su G che satura il taglio composto dal solo nodo s e quindi di valore pari al numero totale di ore da svolgere.

Al fine di alleggerire la notazione poniamo $Z = \sum_{i=1}^{n} d_i$.

Claim. $\exists$ un orario ammissibile $\iff$ $\exists$ un flusso massimo f^* su G tale che $v(f^*) = Z$
Dimostriamo l'affermazione precedente:

- $\Longrightarrow$

  In questo caso facciamo vedere come a partire da un orario ammissibile è possibile definire un flusso massimo ammissibile su G di valore Z e quindi che satura il taglio composto dal solo nodo s. Senza perdita di generalità possiamo dire che l'orario in nostro possesso è costituito da un insieme di triple della forma : $\langle c_i, g_j, l\rangle$ dove la tripla riportata indica che al corso i è stata assegnata l'aula nel giorno j all'ora l.

  Per gli archi del tipo $(C_i, \langle g_j, l\rangle)$ poniamo $f(C_i, \langle g_j, l\rangle) = 1$ se nell'orario in nostro possesso abbiamo la tripla $\langle c_i, g_j, l\rangle$, 0 altrimenti; il vincolo sulle capacità è banalmente rispettato in quanto queste erano state poste ad 1. Dato che l'orario è ammissibile avremo che per ogni *giorno-ora* l'aula è stata assegnata ad al più un corso, quindi possiamo affermare che per ogni nodo $\langle g_j, l\rangle$ si ha $f^{in}(\langle g_j, l\rangle) \leq 1$ e di conseguenza è possibile rispettare il vincolo di conservazione del flusso ponendo $f(\langle g_j, l\rangle, t) = f^{in}(\langle g_j, l\rangle)$ in quanto le capacità di tali archi erano state poste ad 1.

Infine visto che l'orario è ammissibile si ha che ad ogni corso è stata assegnata l'aula per un numero di ore pari a quello richiesto, quindi per ogni nodo C_i avremo che $f^{out}(C_i) = d_i$, per rispettare il vincolo di conservazione su tali nodi poniamo $f(s, C_i) = d_i$ e avremo che il vincolo di capacità sarà rispettato per tutti gli archi uscenti da s ed inoltre tali archi saranno tutti saturi, abbiamo così definito un flusso massimo ammissibile di valore Z.

- $\Longleftarrow$

  Mostriamo ora come da un flusso massimo ammissibile di valore Z è possibile ricavare un orario ammissibile. Dopo aver fatto "girare" un algoritmo di $max - flow$ sulla nostra rete avremo che alcuni archi del tipo $(C_i, \langle g_j, l\rangle)$ avranno flusso 1 mentre altri 0. Costruiamo l' orario richiesto aggiungendo la tripla $\langle C_i, g_j, l\rangle$ se e solo se si ha che $f(C_i, \langle g_j, l\rangle) = 1$. Mostriamo che l'orario così ottenuto è ammissibile: Innanzitutto le richieste dei docenti sono rispettate in quanto i loro corsi sono collegati solo alle combinazioni *giorno-ora* in cui sono disposti a fare lezione.

  Nell'orario costruito avremo che per ogni *giorno-ora* l'aula sarà assegnata ad al più un corso, infatti per la capacità sugli archi $(\langle g_j, l\rangle, t)$ si ha che per ogni nodo $\langle g_j, l\rangle$, $f^{out}(\langle g_j, l\rangle) \leq 1$ e dato che il flusso essendo ammissibile rispetta il vincolo di conservazione si ha anche $f^{in}(\langle g_j, l\rangle) \leq 1$.

  Mostriamo infine che ad ogni corso vengono assegnate il numero di ore totali necessarie: essendo nell'ipotesi che il flusso satura il taglio composto solo da s, avremo che per ogni nodo C_i, $f^{in}(C_i) = d_i$ ed essendo rispettato il vincolo di conservazione, tale quantità di flusso è stata ripartita in d_i archi con flusso 1 e quindi d_i ore di lezione. $\square$

Quindi ricapitolando, avendo mostrato che il problema è riducibile ad un problema di max-flow, al fine di trovare una soluzione, possiamo trasformare l'istanza data in un'istanza di max-flow, trovare un flusso massimo sulla rete ottenuta, e da tale flusso ricavare la soluzione per il problema di partenza come descritto nel secondo punto della dimostrazione precedente.

Diamo infine una veloce analisi sulla complessità della soluzione proposta mostrando come questa risulti polinomiale nella dimensione dell'istanza del problema di partenza (come richiesto):

Per quanto riguarda la rete G ottenuta si ha che: $|V| = n + 4k + 2$ mentre $|E| = n + 4k + (4nk - \sum_{i=1}^{n} |L_i|)$, osservando che gli insiemi L_i potrebbero anche essere vuoti abbiamo che $E = O(nk)$ e quindi G ha dimensione totale $O(nk)$. Per la costruzione della rete bisogna prestare attenzione al modo in cui gli archi tra corsi e *giorni-ore* vengono aggiunti, in particolare al momento di collegare un generico nodo C_i ai nodi $\langle g_j, l\rangle$ bisogna verificare che tale coppia non appartenga ad L_i, supponendo di avere gli insiemi L rappresentati tramite semplici liste tale test può essere effettuato in tempo $O(k)$ dandoci cosi una complessità di costruzione del grafo pari a $O(nk^2)$. (Tale complessità potrebbe essere migliorata rappresentando gli insiemi L con una struttura dati alternativa?)

Una volta costruita la rete possiamo applicare l'algoritmo *max-flow* di Edmond-Karp. Tale algoritmo in una rete connessa ha complessità $O(|E|^2|V|)$ quindi nel nostro caso si ha complessità $O((nk)^2(n+k))$. In definitiva, considerando sia la complessità di costruzione della rete che la complessità per il calcolo del flusso massimo possiamo affermare di riuscire a trovare una soluzione al problema di partenza in tempo totale $O((nk)^2(n+k))$.

Problema 16

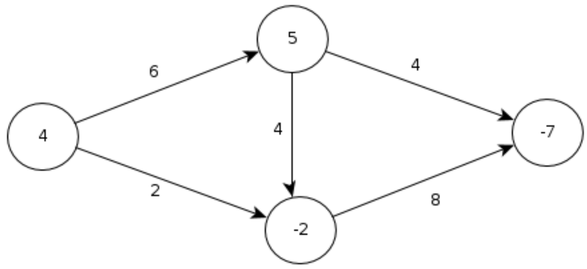
In questo problema, consideriamo:

- un grafo $G = (V, E)$ con archi di capacità $c : E \to \mathbb{Z}$
- una funzione $b : V \to \mathbb{Z}$, che possiamo interpretare come offerta ($b(v) > 0$) o domanda ($b(v) < 0$) di una determinata merce, e per la quale assumiamo $\sum_{v \in V} b(v) = 0$, ovvero c'è equilibrio complessivo tra domanda e offerta.

Ci chiediamo se esiste una modalità di trasferimento della merce attraverso gli archi del grafo che soddisfi per ogni arco il limite posto dalla sua capacità e inoltre faccia sì che avvenga un trasferimento perfetto della merce, che soddisfi tutta la domanda. Formalmente ci chiediamo se esiste un flusso f tale che:

- per ogni nodo $v \in V$: $\sum_{e \in O(v)} f(e) - \sum_{e \in I(v)} f(e) = b(v)$ (la differenza tra flusso uscente ed entrante è pari al valore $b(v)$)
- per ogni $e \in E$, $0 \leq f(e) \leq c(e)$

Nella figura seguente è riportata un'istanza d'esempio, all'interno di ogni nodo è riportato il valore b associato:



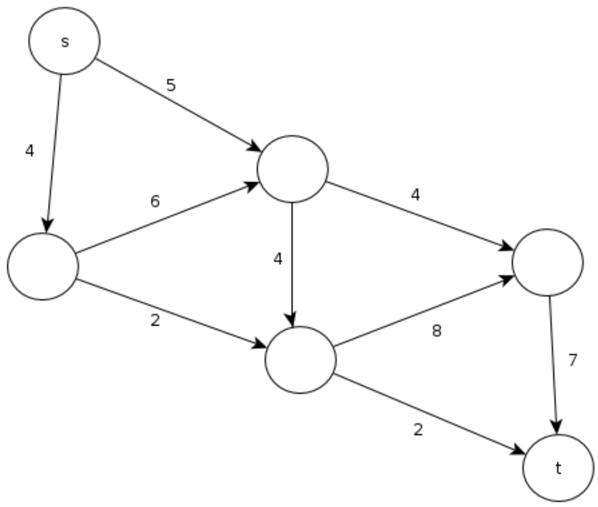
Soluzione Per la prima condizione posta sul flusso da trovare, vogliamo che per i nodi offerta il flusso entrante sia minore di $b(v)$ unita rispetto al flusso uscente, mentre per i nodi domanda il flusso uscente deve essere minore di $b(v)$ unita rispetto a quello entrante. Nel classico problema di $max - flow$ ci sono solo 2 nodi nella rete che non rispettano il vincolo di conservazione sul flusso: s e t. In particolare s è in grado di generare flusso (come i nodi offerta nel nostro problema) mentre t è in grado di assorbire flusso (come i nodi domanda del nostro problema). L'idea è quella di ridurre il problema ad un problema di *max-flow* "spostando" tutta l'offerta presente sulla rete su un unico nodo: la sorgente, e tutta la domanda su un ulteriore unico nodo: la destinazione.

La funzione di riduzione opera nel seguente modo: aggiunge un nodo s che, viene collegato a tutti i nodi $v : b(v) > 0$ con un arco diretto di capacità $b(v)$, si aggiunge inoltre un nodo t e si collegano tutti i nodi $v : b(v) < 0$ a tale nodo t con un arco diretto di capacità $-b(v)$, infine ad ogni nodo non è più associato alcun valore b.

Chiamando la rete ottenuta $G' = (V', E')$ formalmente abbiamo che:

- $V' = V \cup \{s, t\}$
- per gli archi si ha che E' include :
  - l'insieme E con capacità invariate
  - per ogni $v : b(v) > 0$ un arco (s, v) con capacità $b(v)$
  - per ogni $v : b(v) < 0$ un arco (v, t) con capacità $-b(v)$

Nella figura seguente è rappresentata la rete ottenuta a partire dall'istanza di esempio:



Ora che abbiamo definito come opera la funzione di riduzione facciamo vedere come possiamo trovare una soluzione al problema dato trovando un $max - flow$ su G' che satura il taglio composto dal solo nodo s (o equivalentemente quello composto da tutti i nodi eccetto t).

Claim. $\exists$ un trasferimento di merci ammissibile $\iff$ $\exists$ un flusso massimo f^* su G' tale che $v(f^*) = \sum_{v \in V : b(v) > 0} b(v)$

Dimostriamo l'affermazione precedente:

- $\Longrightarrow$

  Mostriamo come a partire da un trasferimento ammissibile si arriva ad un flusso massimo su G' di valore $\sum\limits_{v \in V : b(v) > 0} b(v)$

  Per gli archi in E' che si trovano anche in E lasciamo il flusso invariato e banalmente il vincolo sulle capacità è rispettato in quanto quest'ultime in E' sono rimaste invariate.

  Per gli archi (s, v_i) poniamo $f(s, v_i) = b(v_i)$, sui nodi v_i in questione il vincolo di conservazione è ora rispettato in quanto sono state fornite $b(v_i)$ unità di flusso in più in ingresso, inoltre per tutti gli archi uscenti da s avremo, il vincolo di capacità rispettato e che tali archi sono tutti saturi.

  Infine per gli archi (v_j, t) poniamo $f(v_j, t) = -b(v_j)$, sui nodi v_j in questione il vincolo di conservazione è ora rispettato in quanto sono state fornite $b(v)$ unità di flusso in più in uscita, inoltre per tutti gli archi entranti in t avremo, il vincolo di capacità rispettato e che tali archi sono tutti saturi.

- $\Longleftarrow$

  Mostriamo ora, come a partire da un flusso massimo ammissibile su G' che, satura tutti gli archi uscenti da s e tutti quelli entranti in t, otteniamo un trasferimento di merci ammissibile.

  Applichiamo lo stesso flusso trovato su G' alla sola rete G quindi per tutti i nodi v_i offerta ci saranno $b(v_i)$ unità di flusso in meno in ingresso (come richiesto nel trasferimento cercato); per tutti i nodi v_j domanda ora ci saranno $b(v_j)$ unità di flusso in meno in uscita (come richiesto nel trasferimento cercato).

  Per quanto riguarda le capacità sugli archi banalmente queste risultano essere rispettate in quanto sono invariate rispetto alla rete G' su cui è stato trovato il flusso massimo. In definitiva a partire da un flusso massimo ammissibile si è in grado di derivare un trasferimento di merci ammissibile. $\square$