

RICERCA OPERATIVA - LM in Ingegneria Gestionale (a.a. 2025/26)

Nome:

Cognome:

Matricola:

1) L'azienda *FAST-SPORT* deve organizzare la spedizione di articoli sportivi dal proprio deposito centrale verso i clienti che hanno effettuato ordini online. Sia $G = (N, A)$ il grafo orientato che descrive la rete logistica di *FAST-SPORT*, in cui N è l'insieme dei nodi mentre A rappresenta i collegamenti tra i nodi. In particolare, il deposito centrale è situato nel nodo 1, mentre i clienti da servire corrispondono ai rimanenti nodi in N . Sia d_i il numero di articoli sportivi richiesti dal cliente i , per $i \in N \setminus \{1\}$, mentre u_{ij} indichi la massima quantità di articoli inviabili lungo il collegamento (i, j) , per ogni $(i, j) \in A$. Inoltre, sia H il massimo numero di collegamenti utilizzabili per la spedizione, stabilito dai manager dell'azienda. Infine, c_{ij} sia il costo unitario di invio lungo il collegamento (i, j) , per ogni $(i, j) \in A$.

Si formuli in termini di *PLI* il problema di decidere quale sottoinsieme di al più H collegamenti utilizzare per la spedizione, e come inviare gli articoli sportivi dal deposito centrale verso i nodi cliente lungo i collegamenti selezionati, in modo tale da soddisfare le richieste dei clienti e rispettare i vincoli di capacità associati ai collegamenti, minimizzando il costo totale della spedizione.

SVOLGIMENTO

Per descrivere il problema introduciamo le variabili di flusso x_{ij} , che indicano la quantità di articoli spediti lungo il collegamento $(i, j) \in A$. Introduciamo inoltre le variabili binarie

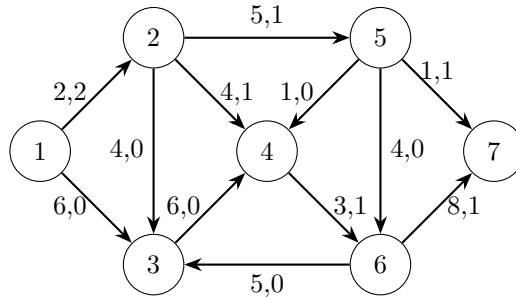
$$y_{ij} = \begin{cases} 1, & \text{se viene utilizzato il collegamento } (i, j) \\ 0, & \text{altrimenti} \end{cases} \quad (i, j) \in A.$$

Il problema può essere formulato come segue:

$$\begin{aligned} \min \quad & \sum_{(i,j) \in A} c_{ij} x_{ij} \\ & \sum_{(j,i) \in BS(i)} x_{ji} - \sum_{(i,j) \in FS(i)} x_{ij} = d_i \quad i \in N \setminus \{1\} \\ & \sum_{(j,1) \in BS(1)} x_{j1} - \sum_{(1,j) \in FS(1)} x_{1j} = - \sum_{i \in N \setminus \{1\}} d_i \\ & \sum_{(i,j) \in A} y_{ij} \leq H \\ & 0 \leq x_{ij} \leq u_{ij} y_{ij} \quad (i, j) \in A \\ & y_{ij} \in \{0, 1\} \quad (i, j) \in A \end{aligned}$$

Il primo blocco di vincoli garantisce il soddisfacimento della domanda dei clienti. Segue l'analogo vincolo di conservazione di flusso relativo al nodo deposito centrale. Si noti che tale vincolo potrebbe essere omissso in quanto linearmente dipendente dai vincoli di conservazione di flusso relativi ai nodi cliente. Il vincolo successivo garantisce che vengano utilizzati al più H collegamenti per effettuare la spedizione. Il penultimo blocco di vincoli assicura che solo i collegamenti selezionati possano essere utilizzati per la spedizione, fino alla relativa massima capacità. Infine la funzione obiettivo, da minimizzare, rappresenta il costo totale della spedizione.

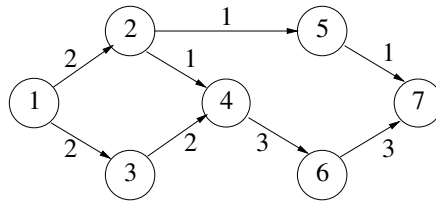
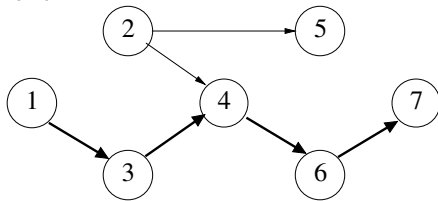
2) Si individui un flusso massimo dal nodo 1 al nodo 7 sulla rete in figura, utilizzando l'algoritmo di Edmonds e Karp a partire dal flusso indicato, di valore 2. Durante la ricerca di un cammino aumentante si visitino gli archi della stella uscente del nodo correntemente esaminato secondo l'ordine crescente dei rispettivi nodi testa (ad esempio, $(1, 2)$ è visitato prima di $(1, 3)$). Per ogni iterazione si riportino l'albero della visita, il cammino aumentante individuato con la relativa capacità, e il flusso ottenuto con il relativo valore. Al termine, si indichi il taglio di capacità minima restituito dall'algoritmo, specificando l'insieme dei nodi N_s , l'insieme dei nodi N_t e la capacità del taglio. Si discuta infine quale sarebbe il valore del flusso massimo nel caso in cui la capacità dell'arco $(5, 7)$ fosse pari a 2.



SVOLGIMENTO

Per ogni iterazione (tranne l'ultima) viene riportato l'albero della visita, in cui viene evidenziato il cammino aumentante P individuato (archi in neretto). Vengono inoltre riportati la capacità di P , il flusso ottenuto in seguito all'invio, lungo P , di una quantità di flusso pari alla capacità del cammino aumentante, e il corrispondente valore del flusso.

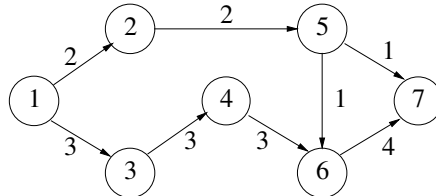
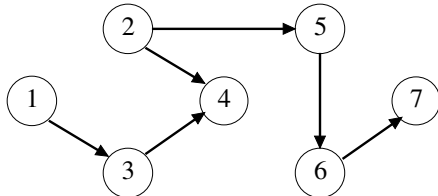
Iterazione 1:



$$P = (1, 3, 4, 6, 7),$$

$$\theta(P, x) = 2, v = 4$$

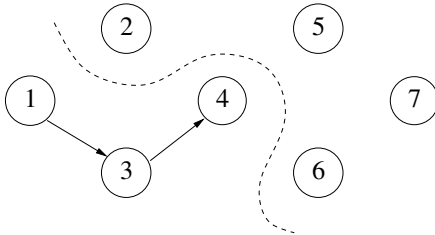
Iterazione 2



$$P = (1, 3, 4, 2, 5, 6, 7),$$

$$\theta(P, x) = 1, v = 5$$

Iterazione 3:



Non esistendo cammini aumentanti, l'ultimo flusso individuato, di valore $v = 5$, è massimo, e $N_s = \{1, 3, 4\}$, $N_t = \{2, 5, 6, 7\}$ è un taglio di capacità minima che separa il nodo 1 dal nodo 7:

Se la capacità dell'arco $(5, 7)$ fosse pari a 2, il taglio ($N_s = \{1, 3, 4\}$, $N_t = \{2, 5, 6, 7\}$) continuerebbe a essere un taglio di capacità minima, con capacità pari a 5. Pertanto il valore del flusso massimo resterebbe invariato.

3) Si applichi alla seguente istanza del problema dello zaino binario

$$\begin{array}{rcccccc} \max & 9x_1 & +12x_2 & +10x_3 & +5x_4 & +x_5 & \\ & 3x_1 & +3x_2 & +4x_3 & +3x_4 & +2x_5 & \leq 11 \\ & x_1, & x_2, & x_3, & x_4, & x_5 & \in \{0,1\} \end{array}$$

l'algoritmo Branch and Bound, utilizzando il rilassamento continuo per determinare la valutazione superiore, l'euristica Greedy CUD per determinare la valutazione inferiore, eseguendo il branching sulla variabile frazionaria della soluzione ottima del rilassamento continuo, visitando l'albero di enumerazione in modo breadth-first e, tra i figli di uno stesso nodo, visitando per primo quello in cui la variabile frazionaria è fissata a 1. Per ogni nodo dell'albero si riportino le soluzioni ottenute dal rilassamento e dall'euristica (se vengono eseguiti), con le corrispondenti valutazioni superiore e inferiore. Si indichi poi se viene effettuato il branching, e come, o se il nodo viene chiuso e perché. Giustificare tutte le risposte.

SVOLGIMENTO

Indichiamo con x^* la soluzione ottenuta dal rilassamento e con $\bar{x}$ quella ottenuta dall'euristica. Indichiamo inoltre con $\bar{z}$ la valutazione superiore ottenuta a ogni nodo (ossia $\bar{z} = c^T x^*$), con $\underline{z}$ la valutazione inferiore ottenuta a ogni nodo (ossia $\underline{z} = c^T \bar{x}$) e con z la migliore delle valutazioni inferiori determinate. L'ordinamento delle variabili per Costo Unitario Decrescente è il seguente: x_2, x_1, x_3, x_4, x_5 .

Inizializzazione: La coda Q viene inizializzata inserendovi il solo nodo radice dell'albero delle decisioni, corrispondente a non aver fissato alcuna variabile; inoltre, si pone $z = -\infty$.

Nodo radice: $x^* = [1, 1, 1, 1/3, 0]$, $\bar{z} = 32 + 2/3$, $\bar{x} = [1, 1, 1, 0, 0]$, $\underline{z} = 31$. Poiché $\underline{z} > z = -\infty$, $z = 31$. Siccome $\bar{z} > z$, si esegue il branching sulla variabile frazionaria x_4 .

$x_4 = 1$: $x^* = [1, 1, 1/2, 1, 0]$, $\bar{z} = 31$, $\bar{x} = [1, 1, 0, 1, 1]$, $\underline{z} = 27$. Siccome $\bar{z} \leq z$, il nodo viene chiuso dalla valutazione superiore.

$x_4 = 0$: $x^* = [1, 1, 1, 0, 1/2]$, $\bar{z} = 31 + 1/2$, $\bar{x} = [1, 1, 1, 0, 0]$, $\underline{z} = 31$. Poiché i costi sono interi, la valutazione superiore può essere arrotondata per difetto al valore 31, e pertanto anche questo nodo può essere chiuso dalla valutazione superiore, poiché $\bar{z} \leq z$.

L'algoritmo termina in quanto Q è vuota, restituendo la soluzione ottima $x = [1, 1, 1, 0, 0]$, di valore $z = 31$.