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

Nome: Cognome: Matricola:

1) L'azienda PisaDroni deve localizzare punti di lancio, detti alveari, per i propri droni, al fine di servire un insieme di clienti effettuando consegne di pacchi tramite droni. A ogni sito $i \in I = \{1, \dots, m\}$ candidato alla localizzazione di un alveare sono associati un costo fisso di apertura f_i e una capacità Q_i, che indica il numero massimo di pacchi che possono essere consegnati a partire da quell'alveare. Ogni cliente $j \in J = \{1, \dots, n\}$ ha una domanda d_j, che indica il numero di pacchi richiesti. Ogni cliente può essere servito da al più un alveare. In particolare, servire il cliente j dall'alveare i comporta un costo fisso di servizio c_{ij}. Se j non è servito, va pagata una penalità p_j.

Si formuli in termini di modello *PLI* il problema di decidere in quali siti candidati aprire un alveare, quali clienti assegnare agli alveari aperti, e quali clienti eventualmente non servire, rispettando i vincoli di linking (un cliente può essere servito solo da un alveare aperto), di capacità di servizio degli alveari (non eccedere Q_i per ogni alveare aperto i), e di assegnamento (ogni cliente può essere servito al massimo da un alveare o rimanere non servito). L'obiettivo è minimizzare il costo totale sostenuto dall'azienda, dato dalla somma dei costi di apertura degli alveari, dei costi di servizio per i clienti assegnati a un alveare, e delle penalità da pagare per i clienti non serviti.

SVOLGIMENTO

Dati di input

- $I = \{1, \dots, m\}$: insieme dei siti candidati all'apertura di un alveare
- $J = \{1, \dots, n\}$: insieme dei clienti da servire
- $f_i \geq 0$: costo di apertura di un alveare nel sito $i \in I$
- $d_j \geq 0$: domanda del cliente $j \in J$ (numero di pacchi da consegnare)
- $Q_i \geq 0$: capacità di servizio dell'alveare $i \in I$ (numero massimo di pacchi consegnabili)
- $p_j \geq 0$: penalità se il cliente $j \in J$ non viene servito
- $c_{ij} \geq 0$: costo di servizio del cliente j da parte dell'alveare i, $i \in I$, $j \in J$

Variabili decisionali

$$y_i = \begin{cases} 1, & \text{se viene aperto un alveare in } i \\ 0, & \text{altrimenti} \end{cases} \qquad \forall i \in I$$

$$x_{ij} = \begin{cases} 1, & \text{se il cliente } j \text{ è servito dall'alveare } i \\ 0, & \text{altrimenti} \end{cases} \qquad \forall i \in I, j \in J$$

$$z_j = \begin{cases} 1, & \text{se il cliente } j \text{ non è servito} \\ 0, & \text{altrimenti} \end{cases} \qquad \forall j \in J$$

Funzione obiettivo

Va minimizzato il costo totale, dato dalla somma dei costi di apertura degli alveari, dei costi di servizio per i clienti assegnati a un alveare, e delle penalità da pagare per i clienti non serviti:

$$\min \sum_{i \in I} f_i y_i + \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{j \in J} p_j z_j \tag{1}$$

Vincoli

- Ogni cliente è servito da un alveare oppure non è servito:

$$\sum_{i \in I} x_{ij} + z_j = 1, \qquad \forall j \in J \tag{2}$$

- Un cliente può essere servito solo da un alveare aperto:

$$x_{ij} \leq y_i, \qquad \forall i \in I,\ j \in J \tag{3}$$

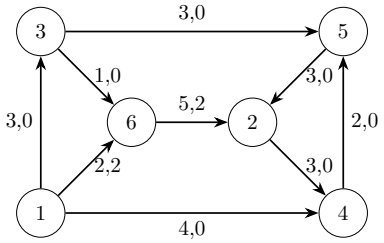
- La capacità degli alveari va rispettata:

$$\sum_{j \in J} d_j x_{ij} \leq Q_i, \qquad \forall i \in I \tag{4}$$

- Domini delle variabili:

$$y_i \in \{0,1\}, \forall i \in I \qquad x_{ij} \in \{0,1\}, \forall i \in I, j \in J, \qquad z_j \in \{0,1\}, \forall j \in J$$

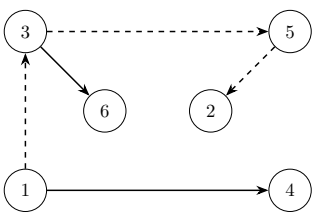
2) Si individui un flusso massimo dal nodo 1 al nodo 2 sulla rete in figura, utilizzando l'algoritmo di Edmonds e Karp a partire dal flusso indicato, di valore 2. Durante la ricerca di un cammino aumentante si visitino gli archi della stella uscente del nodo correntemente esaminato secondo l'ordine crescente 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 $(6,2)$ fosse pari a 3.

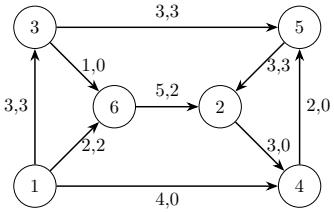


SVOLGIMENTO

Per ogni iterazione viene riportato l'albero della visita, in cui viene evidenziato il cammino aumentante P individuato (linee tratteggiate). Viene inoltre riportato il flusso ottenuto in seguito all'invio, lungo P, di una quantità di flusso pari alla capacità del cammino aumentante.

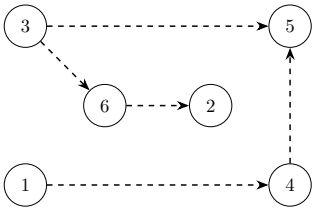
Iterazione 1:

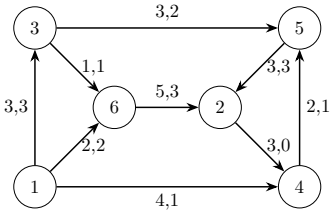




$\theta(P,x) = 3, \quad v = 5$

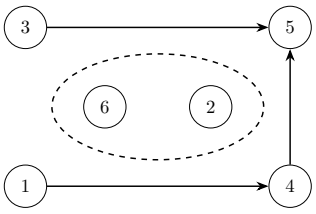
Iterazione 2:





$\theta(P,x) = 1, \quad v = 6$

Iterazione 3:



Non esistendo cammini aumentanti, il flusso corrente è massimo.
Inoltre, il taglio individuato dall'algoritmo, definito da $N_s = \{1,3,4,5\}$ e $N_t = \{2,6\}$, è di capacità minima: $u(N_s, N_t) = u_{16} + u_{36} + u_{52} = 2 + 1 + 3 = 6 = v$.

Se la capacità dell'arco $(6,2)$ fosse pari a 3, il flusso individuato dall'algoritmo continuerebbe a essere ammissibile. Pertanto, non esistendo cammini aumentanti dalla sorgente al pozzo, continuerebbe a essere un flusso massimo. Il valore del flusso massimo sarebbe quindi invariato, ovvero uguale a 6.

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

$$\begin{array}{rrrrrrrl} \max & 12x_1 & +8x_2 & +10x_3 & +5x_4 & +x_5 & & \\ & 3x_1 & +3x_2 & +4x_3 & +3x_4 & +2x_5 & \leq & 11 \\ & x_1, & x_2, & x_3, & x_4, & x_5 & \in & \{0,1\} \end{array}$$

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.

Al termine, si discuta se la soluzione ottima individuata resterebbe ottima anche nel caso in cui la capacità dello zaino valesse 10.

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. Le variabili sono già ordinate per Costo Unitario Decrescente.

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: $x^* = [1, 1, 1, 1/3, 0]$, $\bar{z} = 31 + 2/3$, $\bar{x} = [1, 1, 1, 0, 0]$, $\underline{z} = 30$. Poiché $\underline{z} > z = -\infty$, $z = 30$. Siccome $\bar{z} > z$, si esegue il branching sulla variabile frazionaria x_4.

$x_4 = 1$: $x^* = [1, 1, 1/2, 1, 0]$, $\bar{z} = 30$, $\bar{x} = [1, 1, 0, 1, 1]$, $\underline{z} = 26$. Siccome $\bar{z} \leq z$, il nodo viene chiuso dalla valutazione superiore.

$x_4 = 0$: $x^* = [1, 1, 1, 0, 1/2]$, $\bar{z} = 30 + 1/2$, $\bar{x} = [1, 1, 1, 0, 0]$, $\underline{z} = 30$. Poiché i costi sono interi, la valutazione superiore può essere arrotondata per difetto al valore 30, 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 $z = 30$.

Nel caso in cui la capacità dello zaino fosse pari a 10, $x = [1, 1, 1, 0, 0]$ sarebbe una soluzione ammissibile, e pertanto il suo valore, ovvero 30, rappresenterebbe una valutazione inferiore del valore ottimo dell'istanza. Si osservi che l'istanza del problema con capacità 11 è un rilassamento dell'istanza con capacità 10: il suo valore ottimo, quindi, ovvero 30, valuta per eccesso il valore ottimo dell'istanza con capacità 10. Poiché la valutazione inferiore coincide con quella superiore, segue che $x = [1, 1, 1, 0, 0]$ è soluzione ottima anche per l'istanza con capacità 10.