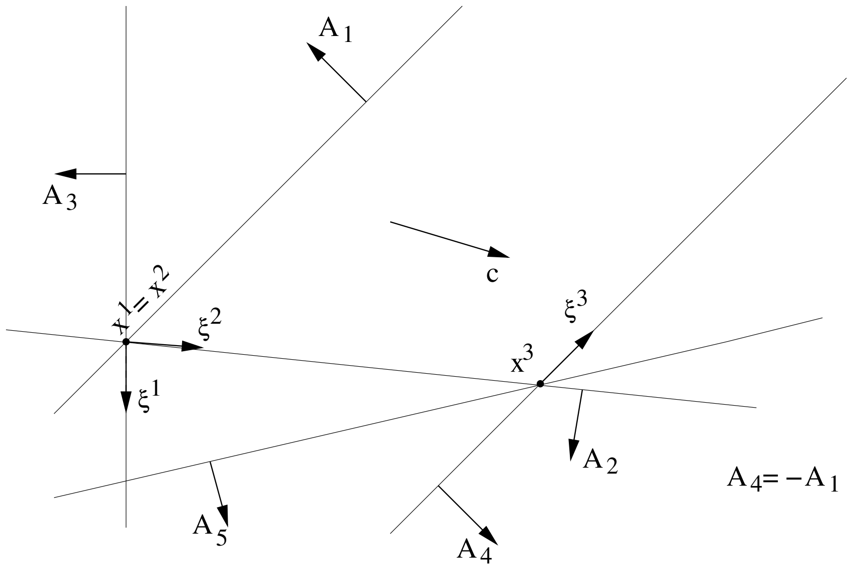
RICERCA OPERATIVA (a.a. 2022/23)

Nome: Cognome: Matricola:

1) Si risolva geometricamente il problema di *PL* in figura per mezzo dell'algoritmo del Simplesso Primale, partendo dalla base $B = \{1,3\}$. Per ogni iterazione si riportino la base, la soluzione di base primale e la direzione di spostamento (riportandoli direttamente sulla figura), il segno delle variabili duali in base e gli indici uscente ed entrante, giustificando le risposte. Si discuta inoltre la degenerazione, sia primale che duale, delle soluzioni di base visitate dall'algoritmo.



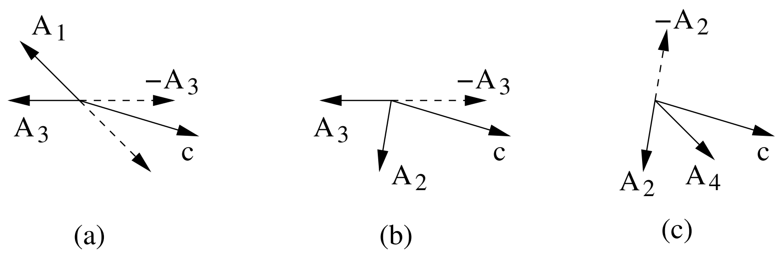
SVOLGIMENTO

it. 1) $B = \{1,3\}$: $y_1 < 0$, $y_3 < 0$ in quanto c appartiene al cono finitamente generato da $-A_1$ e $-A_3$, come mostrato in (a), pertanto $h = \min\{1,3\} = 1$ per la regola anticiclo di Bland. La soluzione di base primale è degenere in quanto $I(x^1) = \{1,2,3\}$, mentre quella duale è non degenere perché nessuna delle variabili duali in base ha valore zero (c è interno al cono generato da $-A_1$ e $-A_3$). Il massimo passo di spostamento lungo la direzione ξ^1 si ottiene in corrispondenza del vincolo 2, attivo ma non in base. Pertanto $k = 2$ e si esegue un cambio di base degenere.

it. 2) $B = \{2,3\}$: $y_2 > 0$, $y_3 < 0$ in quanto c appartiene al cono finitamente generato da A_2 e $-A_3$, come mostrato in (b), pertanto $h = 3$. La soluzione di base primale è sempre degenere, mentre quella duale è non degenere. Il massimo passo di spostamento lungo la direzione ξ^2 si ottiene in corrispondenza dei vincoli 4 e 5, quindi $k = \min\{4,5\} = 4$ per la regola anticiclo di Bland.

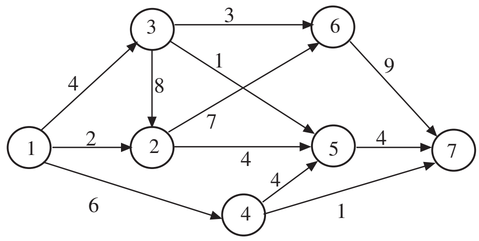
it. 3) $B = \{2,4\}$: $y_2 < 0$, $y_4 > 0$ in quanto c appartiene al cono finitamente generato da $-A_2$ e A_4, come mostrato in (c). Si ha quindi $h = 2$. La soluzione di base primale è degenere in quanto $I(x^3) = \{2,4,5\}$, mentre quella duale è non degenere perché nessuna delle variabili duali in base ha valore zero. La direzione di spostamento scelta dall'algoritmo è ξ^3. Poiché $A_i\xi^3 \leq 0$ per $i \in N = \{1,3,5\}$, ξ^3 è una direzione di crescita illimitata. STOP.

Il problema primale è superiormente illimitato, e di conseguenza il suo problema duale è vuoto.



2) Si individui un albero dei cammini minimi di radice 1, sul grafo in figura, utilizzando l'algoritmo di Dijkstra. Per ogni iterazione si forniscano il nodo selezionato u, i vettori dei predecessori e delle etichette, e l'insieme dei nodi candidati Q. Al termine si disegni l'albero dei cammini minimi individuato.

Si discuta quindi se l'algoritmo di Dijkstra sia il più appropriato, per l'istanza proposta, dal punto di vista della complessità computazionale in tempo. In caso contrario, quale algoritmo sarebbe teoricamente più conveniente, tra quelli studiati? Giustificare le risposte.

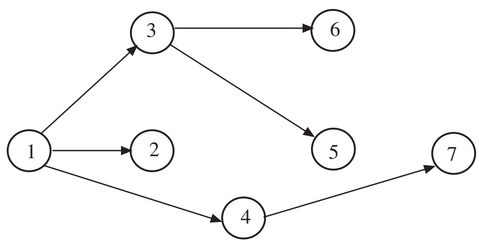


SVOLGIMENTO

$M = (n-1)C_{max} + 1 = 6 \times 9 + 1 = 55.$

Iter.	u	$d[1]$	$d[2]$	$d[3]$	$d[4]$	$d[5]$	$d[6]$	$d[7]$	$p[1]$	$p[2]$	$p[3]$	$p[4]$	$p[5]$	$p[6]$	$p[7]$	Q
0		0	55	55	55	55	55	55	nil	1	1	1	1	1	1	$\{1\}$
1	1	0	2	4	6	55	55	55	nil	1	1	1	1	1	1	$\{2,3,4\}$
2	2	0	2	4	6	6	9	55	nil	1	1	1	2	2	1	$\{3,4,5,6\}$
3	3	0	2	4	6	5	7	55	nil	1	1	1	3	3	1	$\{4,5,6\}$
4	5	0	2	4	6	5	7	9	nil	1	1	1	3	3	5	$\{4,6,7\}$
5	4	0	2	4	6	5	7	7	nil	1	1	1	3	3	4	$\{6,7\}$
6	6	0	2	4	6	5	7	7	nil	1	1	1	3	3	4	$\{7\}$
7	7	0	2	4	6	5	7	7	nil	1	1	1	3	3	4	$\emptyset$

L'albero ottimo individuato è mostrato in figura.



Si osservi che il grafo non contiene cicli orientati. Pertanto, l'algoritmo più conveniente dal punto di vista della complessità computazionale in tempo, tra quelli studiati, è l'algoritmo per reti acicliche, avente complessità in tempo $O(m)$, mentre la complessità in tempo dell'algoritmo di Dijkstra è $O(n^2)$. Si osservi anche che il grafo non è ben numerato (si consideri a riguardo l'arco $(3, 2)$). Per applicare l'algoritmo per reti acicliche è pertanto necessario rinumerare i nodi del grafo in accordo a una buona enumerazione. A tal fine, è sufficiente invertire la numerazione dei nodi 2 e 3.

3) Si risolva la seguente istanza del problema dello zaino binario

$$\begin{array}{llllllll} \max & 4x_1 & +16x_2 & +8x_3 & +14x_4 & +2x_5 & & \\ & 3x_1 & +3x_2 & +2x_3 & +4x_4 & +2x_5 & \leq & 8 \\ & x_1, & x_2, & x_3, & x_4, & x_5 & \in & \{0,1\} \end{array}$$

mediante l'algoritmo Branch and Bound, utilizzando il rilassamento continuo per determinare una valutazione superiore, l'euristica Greedy CUD per determinare una valutazione inferiore, eseguendo il branching sulla variabile frazionaria della soluzione ottima del rilassamento continuo, e visitando l'albero di enumerazione in modo breadth-first (tra i figli di uno stesso nodo, si visiti per primo quello in cui la variabile frazionaria è fissata a 0). 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 inoltre se viene effettuato il branching, e come, o se il nodo viene chiuso e perché. Si interrompa l'esecuzione dell'algoritmo dopo aver visitato cinque nodi dell'albero (incluso il nodo radice), riportando il gap assoluto finale, ovvero la differenza tra la miglior valutazione superiore e la miglior valutazione inferiore disponibili nel momento in cui l'esplorazione viene interrotta.

SVOLGIMENTO

Indichiamo con x^* la soluzione ottima del rilassamento continuo e con $\bar{x}$ la soluzione restituita 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 CUD (Costo Unitario Decrescente) delle variabili è: x_2, x_3, x_4, x_1, 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: $x^* = [0, 1, 1, 3/4, 0]$, $\bar{z} = 34 + 1/2$, $\bar{x} = [1, 1, 1, 0, 0]$, $\underline{z} = 28$. Poiché $\underline{z} > z = -\infty$, $z = 28$. Siccome $\bar{z} > z$, si esegue il branching sulla variabile frazionaria x_4.

$x_4 = 0$: $x^* = [1, 1, 1, 0, 0]$, $\bar{z} = 28$. Poiché la soluzione del rilassamento continuo è intera, si ha $\underline{z} = 28$. Dato che $\bar{z} = \underline{z}$, il nodo viene chiuso per ottimalità (avrebbe potuto anche essere chiuso dalla valutazione superiore).

$x_4 = 1$: $x^* = [0, 1, 1/2, 1, 0]$, $\bar{z} = 34$, $\bar{x} = [0, 1, 0, 1, 0]$, $\underline{z} = 30$. Poiché $\underline{z} > z$, si pone $z = 30$. Inoltre, poiché $\bar{z} > z$, si esegue il branching sulla variabile frazionaria x_3.

$x_4 = 1, x_3 = 0$: $x^* = [1/3, 1, 0, 1, 0]$, $\bar{z} = 31 + 1/3$, $\bar{x} = [0, 1, 0, 1, 0]$, $\underline{z} = 30$, quindi z non cambia. Poiché $\bar{z} > z$, si esegue il branching sulla variabile frazionaria x_1.

$x_4 = x_3 = 1$: $x^* = [0, 2/3, 1, 1, 0]$, $\bar{z} = 32 + 2/3$, $\bar{x} = [0, 0, 1, 1, 1]$, $\underline{z} = 24$. Poiché $\bar{z} > z$, si esegue il branching sulla variabile frazionaria x_2.

Poiché sono stati visitati cinque nodi, l'esecuzione dell'algoritmo viene interrotta. Nel momento dell'interruzione, la miglior valutazione inferiore disponibile è data da $z = 30$. La miglior valutazione superiore disponibile è invece pari a $\max\{\ z\ ,\ \max\ \{\ \bar{z}(P')\ :\ P' \in Q'\ \}\ \}$, dove Q' è l'insieme dei predecessori immediati dei nodi in Q. In questo caso Q' contiene sia il nodo $x_4 = 1, x_3 = 0$ che il nodo $x_4 = x_3 = 1$. Pertanto, la miglior valutazione superiore disponibile è $32 + 2/3$, che può essere arrotondata a 32 essendo tutti i coefficienti di costo interi, e di conseguenza il gap assoluto finale è 2.