Guida esercizi Ricerca Operativa

Esercizi di Modellazione

Viene dato un contesto reale come testo, dove è necessario ottimizzare i risultati. Per fare questo si deve ottenere un risultato del genere:

$$\max \quad 30x_I + 20x_{II} + 50x_{III}$$

$$\begin{array}{lll}
s.t. & & \\
& x_I \geq 200 & \text{(vincoli sulla domanda)} \\
& x_{II} \geq 200 & \\
& x_{III} \geq 150 & \\
& & \\
& 2x_I + 3x_{II} + 5x_{III} \leq 4000 & \text{(vincoli sui materiali)} \\
& 4x_I + 2x_{II} + 7x_{III} \leq 6000 & \\
& & \\
& x_I + \frac{1}{2}x_{II} + \frac{1}{3}x_{III} \leq 700 & \text{(vincoli sulla forza lavoro)} \\
& & \\
& x_i \in \mathbb{Z}_+, \forall\, i \in \{I, II, III\} & \text{(dominio)}
\end{array}$$

Dove:

- **Max:** indica che si tratta di una funzione di massimo, ovvero si cerca di massimizzare l'espressione che segue (vale viceversa per il minimo)
- **La funzione**: indica le variazioni che subisce il risultato finale, modificando i parametri principali inseriti.
- **I parametri x_n:** vengono decisi dopo aver letto l'esercizio, e rappresentano il valore, relativo agli "n" elementi del problema, che impatta sul risultato finale e sui quali verranno messi vincoli.
  Questi parametri possono essere anche **a multipli indici**, se vi sono "n" elementi del problema, con "j" loro elementi a loro volta coinvolti.
  Oppure possono essere (in aggiunta a quelli base) a valori **binari** (**booleani**) nel caso si debba verificare la presenza o meno (il verificarsi o meno) di determinate condizioni.
- **S.t.**: è una scrittura fissa.
- **I vincoli**: che sono tutte le disequazioni poste sotto alla funzione, sono appunto le limitazioni poste dalle condizioni del problema, e che andranno a porre i parametri scelti entro certi limiti (sono estratte direttamente dal testo del problema).
- **Dominio:** generalmente, dopo i vincoli, in basso, viene specificato il dominio all'interno del quale i parametri si trovano con una costruzione del tipo.

Problema di Programmazione Lineare con Simplesso

1) Parto da un problema in forma standard

$$\begin{array}{lcccccccccccc} \min & -13x_1 & - & 10x_2 & & & & & & & & & \\ s.t. & 3x_1 & + & 4x_2 & + & s_1 & & & & & = & 24 \\ & x_1 & + & 4x_2 & & & + & s_2 & & & = & 20 \\ & 3x_1 & + & 2x_2 & & & & & + & s_3 & = & 18 \\ & x_1 & , & x_2 & , & s_1 & , & s_2 & , & s_3 & \geq & 0 \end{array}$$

2) Scrivo la funzione obiettivo come equazione, con la variabile z

$$z = -13x_1 - 10x_2$$

3) Riscrivo la funzione come vincolo del problema

$$\begin{array}{cccccccccccc} -13x_1 & - & 10x_2 & & & & & & & - & z & = & 0 \\ 3x_1 & + & 4x_2 & + & s_1 & & & & & & & = & 24 \\ x_1 & + & 4x_2 & & & + & s_2 & & & & & = & 20 \\ 3x_1 & + & 2x_2 & & & & & + & s_3 & & & = & 18 \end{array}$$

4) Costruisco la matrice estesa del sistema

$$\begin{array}{c|cccccc|c} & x_1 & x_2 & s_1 & s_2 & s_3 & z & \bar{b} \\ \hline (R_0) & -13 & -10 & 0 & 0 & 0 & -1 & 0 \\ (R_1) & 3 & 4 & 1 & 0 & 0 & 0 & 24 \\ (R_2) & 1 & 4 & 0 & 1 & 0 & 0 & 20 \\ (R_3) & 3 & 2 & 0 & 0 & 1 & 0 & 18 \end{array}$$

5) Scegliendo ad esempio la base con x_1, x_2 e s_3, eseguo le operazioni sulla matrice per ottenere una matrice di permutazione nelle rispettive colonne (ovvero una matrice diagonale di 1 ed il resto 0)

Operazioni: $R_1 \leftarrow R_1/3$, $R_2 \leftarrow R_2 - R_1/3$, $R_3 \leftarrow R_3 - R_1$, $R_0 \leftarrow R_0 + 13/3R_1$

$$\begin{array}{c|cccccc|c} & x_1 & x_2 & s_1 & s_2 & s_3 & z & \bar{b} \\ \hline (R_0) & 0 & 22/3 & 13/3 & 0 & 0 & -1 & 104 \\ (R_1) & 1 & 4/3 & 1/3 & 0 & 0 & 0 & 8 \\ (R_2) & 0 & 8/3 & -1/3 & 1 & 0 & 0 & 12 \\ (R_3) & 0 & -2 & -1 & 0 & 1 & 0 & -6 \end{array}$$

Operazioni: $R_2 \leftarrow 3/8R_2$, $R_3 \leftarrow R_3 + 3/4R_1$, $R_1 \leftarrow R_1 - 1/2R_2$, $R_0 \leftarrow R_0 - 11/4R_2$

$$\begin{array}{c|cccccc|c} & x_1 & x_2 & s_1 & s_2 & s_3 & z & \bar{b} \\ \hline (R_0) & 0 & 0 & 21/4 & -11/4 & 0 & -1 & 71 \\ (R_1) & 1 & 0 & 1/2 & -1/2 & 0 & 0 & 2 \\ (R_2) & 0 & 1 & -1/8 & 3/8 & 0 & 0 & 9/2 \\ (R_3) & 0 & 0 & -5/4 & 3/4 & 1 & 0 & 3 \end{array}$$

6) Noto che corrisponde al sistema in forma canonica rispetto alla base scelta

$$\begin{array}{rcrcrlcrl} z & = & -71 & + & 21/4 & s_1 & - & 11/4 & s_2 \\ x_2 & = & 9/2 & + & 1/8 & s_1 & - & 3/8 & s_2 \\ s_3 & = & 3 & + & 5/4 & s_1 & - & 3/4 & s_2 \\ x_1 & = & 2 & - & 1/2 & s_1 & + & 1/2 & s_2 \end{array}$$

7) Valuto poi le condizioni di ottimalità e non illimitatezza, basandomi sullo schema sotto:

<u>Ottimalità</u>:

$$\begin{array}{rcl} x^*_{B_i} & = & \bar{b}_i \quad (i = 1 \ldots m) \\ x^*_{F_j} & = & 0 \quad (j = 1 \ldots n-m) \\ z^* & = & \bar{z}_B \end{array}$$

<u>Illimitatezza</u>:

$$(\bar{c}_h \;<\; 0) \;\wedge\; (\bar{a}_{ih} \;\leq\; 0, \;\forall\; i = 1 \ldots m)$$

<u>Schema</u>:

$$\begin{array}{c|cccccc|c} & x_1 & x_2 & s_1 & s_2 & s_3 & z & \bar{b} \\ \hline (R_0) & \bar{c}_{x_1} & \bar{c}_{x_2} & \bar{c}_{s_1} & \bar{c}_{s_2} & \bar{c}_{s_3} & -1 & -\bar{z}_B \\ (R_1) & 1 & 0 & \bar{a}_{1s_1} & \bar{a}_{1s_2} & 0 & 0 & \bar{b}_1 \\ (R_2) & 0 & 1 & \bar{a}_{2s_1} & \bar{a}_{2s_2} & 0 & 0 & \bar{b}_2 \\ (R_3) & 0 & 0 & \bar{a}_{3s_1} & \bar{a}_{3s_1} & 1 & 0 & \bar{b}_3 \end{array}$$

8) Nel caso una delle condizioni non sia verificata procedo con lo scegliere una variabile entrante per il cambio di base, con costo ridotto strettamente negativo:

$$x_h : \bar{c}_h < 0.$$

9) Ed una variabile uscente con:

$$t = \arg \min_{i=1\ldots m} \left\{ \frac{\bar{b}_i}{\bar{a}_{ih}} : \bar{a}_{ih} > 0 \right\}$$

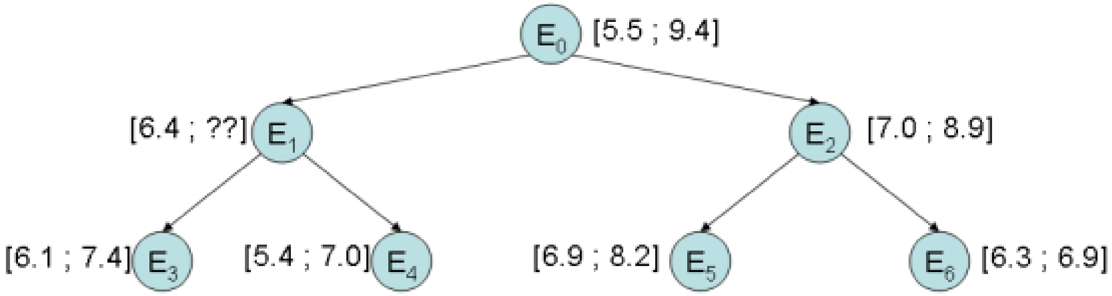
10) Aggiorno quindi la base corrente, eliminando la colonna della variabile uscente e sostituendola con la colonna entrante.

11) È possibile utilizzare un accorgimento tramite il “metodo del Pivot” per velocizzare leggermente il cambio di base.

12) **Reitero l’intero procedimento**

Problemi di ottimizzazione con Branch-and-Bound

1. Si parte sapendo se un problema è di massimo o di minimo (es. massimo) e le coppie $[SA; UB]$:



2. Si tenta di trovare un range per la soluzione ottima attuando un ulteriore Branch sul nodo E_5 e chiudendo il resto dei nodi:

3. Mi viene garantito dall'ipotesi che faccio che E_8 è non ammissibile, quindi il nodo è <u>chiuso</u>.
4. Di seguito posso <u>chiudere</u> anche i nodi E_6 in quanto i loro UB risultano minori dell'SA di E_2.
5. Poi posso dire che per ottenere una soluzione ottima presente in E_7 il valore della sua soluzione ammissibile dovrà essere $\geq$ del valore più alto promesso (UB) di E_3. Nello specifico sarà $[7.4, 8.2]$ dove 8.2 è il valore massimo promesso generale.
6. Essendo stato surclassato da E_7 posso <u>chiudere</u> anche E_3.

Problemi di cammino minimo

Spesso negli esercizi viene chiesto di scegliere quale algoritmo applicare e perché, in generale si seguono queste regole:

1) **Algoritmo label correcting:** Buono, ma non applicabile con cicli negativi.
2) **Algoritmo di Bellman-Ford:** Buono ed applicabile con cicli negativi. (ma probabilmente più lungo del precedente)
3) **Algoritmo di Dijkstra:** Il più efficiente, ma non applicabile con costi negativi.

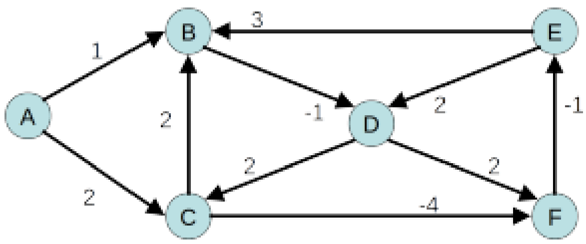


Figure 13: Grafo per l'Esempio 8.

Svolgimento

Eseguo l’iterazione del procedimento scelto, tramite tabelle diverse:

- Per l’Algoritmo label correcting

Inizializzazioni: $\pi_s = 0$; $\pi_1 = \pi_2 = \pi_3 = +\infty$;
Iterazione 1: arco $(s, 1)$: $\pi_1 = 2$; $p(1) = s$.
Iterazione 2: arco $(1, 2)$: $\pi_2 = 3$; $p(2) = 1$.
Iterazione 3: arco $(2, 3)$: $\pi_3 = 2$; $p(3) = 2$.
Iterazione 4: arco $(3, 1)$: $\pi_1 = 1$; $p(1) = 3$.
Iterazione 5: arco $(1, 2)$: $\pi_2 = 2$; $p(2) = 1$.
Iterazione 6: arco $(2, 3)$: $\pi_3 = 1$; $p(3) = 2$.
Iterazione 7: arco $(3, 1)$: $\pi_1 = 0$; $p(1) = 3$.
Iterazione 8: arco $(1, 2)$: $\pi_2 = 1$; $p(2) = 1$.
Iterazione 9: arco $(2, 3)$: $\pi_3 = 0$; $p(3) = 2$.
Iterazione 10: arco $(3, 1)$: $\pi_1 = -1$; $p(1) = 3$.
...

Ovvero scorro ogni nodo, a partire da quello selezionato per la partenza, e in base al valore dato all’arco aggiorno l’etichetta del nodo. Termino appena sono soddisfatti i vincoli duali (ho trovato il percorso minimo), oppure trovo un ciclo negativo.

- Per l’Algoritmo di Bellman-Ford

iterazione	nodo A	nodo B	nodo C	nodo D	nodo E	nodo F	Aggiornati
$h = 0$	$0_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	A
$h = 1$	$0_{(\wedge)}$	$+1_{(A)}$	$+2_{(A)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	B, C
$h = 2$	$0_{(\wedge)}$	$+1_{(A)}$	$+2_{(A)}$	$0_{(B)}$	$+\infty_{(\wedge)}$	$-2_{(C)}$	D, F
$h = 3$	$0_{(\wedge)}$	$+1_{(A)}$	$+2_{(A)}$	$0_{(B)}$	$-3_{(F)}$	$-2_{(C)}$	E
$h = 4$	$0_{(\wedge)}$	$0_{(E)}$	$+2_{(A)}$	$-1_{(E)}$	$-3_{(F)}$	$-2_{(C)}$	B, D
$h = 5$	$0_{(\wedge)}$	$0_{(E)}$	$+1_{(D)}$	$-1_{(E)}$	$-3_{(F)}$	$-2_{(C)}$	C
$h = 6$	$0_{(\wedge)}$	$0_{(E)}$	$+1_{(D)}$	$-1_{(E)}$	$-3_{(F)}$	$-3_{(C)}$	F

Anche qui, parto da un nodo (A in questo caso) ed aggiorno le varie etichette dei nodi (segnandolo in tabella). Ogni iterazione corrisponde ad un passo successivo nel grafo, e devo ricordare di (ovviamente) aggiornare solo dove necessario (costo etichetta migliorato/più basso). Le iterazioni si fermano appena se ne incontra una che non aggiorna, o (generalmente) ad h=6, oppure un limite posto dal problema.

- Per l’algoritmo di Dijkstra

it.	nodo A	nodo B	nodo C	nodo D	nodo E	nodo F	S	$\hat{v}$
0	$0_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	A, B, C, D, E, F	
1	$*$	$7_{(A)}$	$1_{(A)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	$+\infty_{(\wedge)}$	B, C, D, E, F	A
2		$6_{(C)}$	$*$		$3_{(C)}$	$8_{(C)}$	B, D, E, F	C
3		$5_{(E)}$		$8_{(E)}$	$*$		B, D, F	E
4		$*$		$-$		$6_{(B)}$	D, F	B
5					$\times$	$*$	D	F
6				$*$			$\emptyset$	D

$*$: etichetta fissata.
$-$: etichetta controllata ma non aggiornata.
$\times$: etichetta non controllata perché il nodo è già fissato.

L’algoritmo di Dijkstra funziona in maniera simile al precedente, solo che invece di aggiornare tutte le etichette, a partire dalla seconda iterazione comincia a fissare (per ogni iterazione) un nodo alla volta, ed aggiornare tutte le etichette possibili da quel nodo. Ogni nodo viene fissato una sola volta e (come si può vedere dalla tabella) viene rimosso dall’elenco di quelli da fissare. I nodi fissati sono quelli per cui si ha la certezza che non si avrà un valore migliore di quello attuale. L’algoritmo si arresta nel momento in cui tutti i nodi sono stati fissati una volta.

Esercizio Primale-Duale

Viene generalmente chiesto di enunciare le condizioni di complementarietà Duale-Primale:

$$\min\{c^T x : x \geq 0, Ax \geq b\}$$
$$\max\{u^T b : u \geq 0, u^T A \leq c^T\}$$

$$\begin{array}{c} x \text{ e } u \text{ ottime} \\ \text{primale e duale (risp.)} \end{array} \iff \begin{array}{cl} Ax \geq b \wedge x \geq 0 & \textit{(ammissibilità primale)} \\ u^T A \leq c^T \wedge u \geq 0 & \textit{(ammissibilità duale)} \\ \left.\begin{array}{c} u^T(Ax - b) = 0 \\ (c^T - u^T A)x = 0 \end{array}\right\} & \textit{(ortogonalità)} \end{array}$$

E poi di dimostrarle applicandole ad un problema particolare, data una soluzione proposta. Per fare ciò bisogna innanzitutto verificare le condizioni di ammissibilità primale del problema dato per la soluzione data. Questo viene fatto sostituendo alle variabili nei vincoli del problema, i valori proposti dalla soluzione, nel caso questa operazione dia risultati accettabili passo alla fase successiva.

$2x_1 + 3x_2 - x_3 = 0 + 12 - 8 = 4$ (OK)
$x_1 + x_2 = 0 + 4 \leq 4$ (OK)
$0 \leq 0, 4 \geq 0, 8 \geq 0$ (domini OK)

La fase successiva prevede il passaggio dal primale al duale (o viceversa), e viene fatto seguendo queste regole:

Primale $(\min c^T x)$	**Duale** $(\max u^T b)$
$a_i^T x \geq b_i$	$u_i \geq 0$
$a_i^T x \leq b_i$	$u_i \leq 0$
$a_i^T x = b_i$	u_i libera
$x_j \geq 0$	$u^T A_j \leq c_j$
$x_j \leq 0$	$u^T A_j \geq c_j$
x_j libera	$u^T A_j = c_j$

Applico infine le condizioni citate in partenza per verificare il risultato ottenuto.

- Il primo vincolo primale è di uguaglianza: non ci sono condizioni di complementarietà con la relativa variabile duale u_1
- $u_2(x_1 + x_2 - 4) = 0 \rightarrow u_2(0) = 0 \rightarrow //$

- $(2u_1 + u_2 - 1)x_1 = 0 \rightarrow (2u_1 + u_2 - 1)0 = 0 \rightarrow //$
- $(3u_1 + u_2 - 1)x_2 = 0 \rightarrow (3u_1 + u_2 - 1)4 = 0 \rightarrow 3u_1 + u_2 - 1 = 0$ (prima condizione)
- $(-u_1 - 0)x_3 = 0 \rightarrow (-u_1)8 = 0 \rightarrow u_1 = 0$ (seconda condizione)

Mettiamo a sistema le due condizioni trovate:

$$\begin{cases} 3u_1 + u_2 - 1 = 0 \\ u_1 = 0 \end{cases} \Rightarrow \begin{cases} u_2 = 1 \\ u_1 = 0 \end{cases}$$

La soluzione duale trovata:

- soddisfa i tre vincoli duali ($2u_1 + u_2 = 1 \leq 1$, $3u_1 + u_2 = 1 \geq 1$, $-u_1 = 0 \geq 0$);
- soddisfa i vincoli di dominio (u_1 libera, $u_2 = 1 \geq 0$)
- è in scarti complementari con la soluzione primale data (per costruzione).

Pertanto, le due soluzioni sono ottime per i rispettivi problemi primale e duale.

Aggiunte importanti riguardanti la teoria o procedimenti particolari

Passaggio in forma standard

$$\begin{array}{lll} \min & c_1x_1 + c_2x_2 + \ldots + c_nx_n & \\ \text{s.t.} & a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \quad = b_i & (i = 1 \ldots m) \\ & x_i \in \mathbb{R}_+ & (i = 1 \ldots n) \end{array}$$

Dove:

1. La funzione obiettivo è di minimo e senza costanti additive o moltiplicative.
2. Tutte le variabili sono positive o nulle.
3. Tutti i vincoli sono delle equazioni.
4. I termini noti b_i sono tutti positivi o nulli.

Come fare:

1. Si moltiplicano per -1 le funzioni di massimo, le costanti additive e moltiplicative positive possono essere trascurate, le costanti moltiplicative negative possono essere eliminate cambiando il verso di ottimizzazione.
2. Si effettuano sostituzioni di variabili per le variabili libere o negative.
3. Si aggiunge una variabile positiva di slack per i vincoli $\leq$ e si sottrae per i vincoli $\geq$.
4. Si moltiplicano per -1 i vincoli con termine noto negativo.

Regola anti-ciclo di Bland: tra tutte le variabili candidate al cambio di base, scegliere sempre quella con indice minimo.

Algoritmo di Dijkstra e la sua complessità

L'algoritmo di Dijkstra determina i valori ottimi delle etichette π_j, mantenendo ad ogni iterazione un insieme di nodi che sono già stati fissati (ottimi) ed uno di nodi ancora da fissare. Ad ogni iterazione un nodo viene passato da un insieme all'altro e l'algoritmo termina quando non vi sono più nodi da fissare. Ad ogni iterazione viene aggiornata l'etichetta π dei rispettivi nodi che lo necessitano. L'algoritmo di Dijkstra può lavorare utilizzando l'heap e tra le operazioni che possono essere effettuate su un heap, quelle che interessano l'algoritmo di Dijkstra sono le seguenti:

- Create: crea un heap vuoto, usata in fase di inizializzazione
- Insert: aggiunge un elemento allo heap, usata in fase di inizializzazione
- Find-min: restituisce l'elemento minimo, usata al passo 2) per selezionare $\hat{v}$
- Delete-min: elimina l'elemento minimo dall'heap, usata al passo 2) per eliminare da $\bar{S}$ l'elemento $\hat{v}$
- Decrease-key: diminuisce il valore di un elemento, usata al passo 3) ogni volta che si migliora un'etichetta

Le varie complessità, derivanti dal modo in cui l'heap è implementato:

- Create: complessità $O(1)$
- Insert: complessità $O(\log|N|)$
- Find-min: complessità $O(1)$
- Delete-min: complessità $O(\log|N|)$
- Decrease-key: complessità $O(\log|N|)$

La proprietà dell'algoritmo dice:

Dato un grafo pesato $G = (N, A)$ e un nodo origine $s \in N$, l'algoritmo di Dijkstra risolve il problema del cammino minimo dall'origine s verso tutti gli altri nodi in tempo $O(|N|^2)$.

Problema dello Zaino 0/1

Si parte da un problema del genere:

$$\begin{array}{rrlrlrlrlrlrlll}
\max & 36 & x_1 & + & 15 & x_2 & + & 3 & x_3 & + & 5 & x_4 & + & 11 & x_5 & + & 30 & x_6 & & \\
s.t. & 12 & x_1 & + & 6 & x_2 & + & 2 & x_3 & + & 3 & x_4 & + & 5 & x_5 & + & 9 & x_6 & \leq & 17 \\
 & & x_1 & , & & x_2 & , & & x_3 & , & & x_4 & , & & x_5 & , & & x_6 & \in & \{0,1\}
\end{array}$$

Si calcolano i valori dei rapporti p_i/w_i e ordiniamo in maniera decrescente:

$$\frac{30}{9} \geq \frac{36}{12} \geq \frac{15}{6} \geq \frac{11}{5} \geq \frac{5}{3} \geq \frac{3}{2}$$

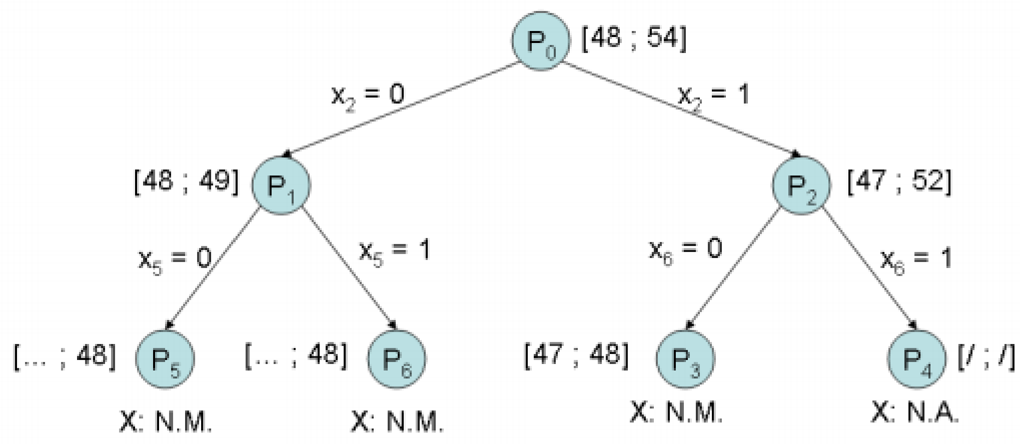
L'ordine delle variabili è quindi:

$$x_6 \to x_1 \to x_2 \to x_5 \to x_4 \to x_3$$

Riordino il problema:

$$\begin{array}{rrlrlrlrlrlrlll}
\max & 30 & x_6 & + & 36 & x_1 & + & 15 & x_2 & + & 11 & x_5 & + & 5 & x_4 & + & 3 & x_3 & & \\
s.t. & 9 & x_6 & + & 12 & x_1 & + & 6 & x_2 & + & 5 & x_5 & + & 3 & x_4 & + & 2 & x_3 & \leq & 17 \\
 & & x_6 & , & & x_1 & , & & x_2 & , & & x_5 & , & & x_4 & , & & x_3 & \in & \{0,1\}
\end{array}$$

Ottenendo il seguente albero di Branch-and-Bound:



$\boxed{\text{Nodo P0}}$:

$x_6 = 1 \quad \bar{W} = 8$

$x_1 = \frac{8}{12} \quad \bar{W} = 0 \quad x_1$ critica

$UB_0 = 30 + \frac{8}{12}36 = 54$

$SA_0 = 30_{(x_6)} + 15_{(x_2)} + 3_{(x_3)} = 48$

$\boxed{\text{Nodo P1}}$:

$x_1 = 0 \quad \bar{W} = 17$

$x_6 = 1 \quad \bar{W} = 8$

$x_2 = 1 \quad \bar{W} = 2$

$x_5 = \frac{2}{5} \quad \bar{W} = 0 \quad x_5$ critica

$UB_1 = 0 + 30 + 15 + \frac{2}{5}11 = 49.4$

$SA_1 = 30_{(x_6)} + 15_{(x_2)} + 3_{(x_3)} = 48$

$\boxed{\text{Nodo P2}}$:

$x_1 = 1 \quad \bar{W} = 5$

$x_6 = \frac{5}{9} \quad \bar{W} = 0 \quad x_6$ critica

$UB_2 = 36 + \lfloor \frac{5}{9}30 \rfloor = 36 + \lfloor 16,\bar{6} \rfloor = 52$

$SA_2 = 36 + 11_{(x_5)} = 47$

Adottando una strategia Best Bound First, effettuiamo il branch del nodo P_2.

$\boxed{\text{Nodo P3}}$:

$x_1 = 1 \quad \bar{W} = 5$

$x_6 = 0 \quad \bar{W} = 5$

$x_2 = \frac{5}{6} \quad \bar{W} = 0 \quad x_2$ critica

$UB_3 = 36 + \lfloor \frac{5}{6}15 \rfloor = 48$

$SA_3 = 36 + 11_{(x_5)} = 47$

$UB_3 = 48 \geq 48$ (valore della migliore soluzione disponibile): chiudo P_3 perché non migliorante.

$\boxed{\text{Nodo P4}}$:

$x_1 = 1 \quad \bar{W} = 5$

$x_6 = 1 \quad \bar{W} = -4 \quad$ non ammissibile

Chiudo P_4 per non ammissibilità.

Rimane aperto ancora il nodo P_1, sul quale faccio branch.

$\boxed{\text{Nodo P5}}$:

$$
\begin{array}{lll}
x_1 = 0 & \bar{W} = 17 & \\
x_5 = 0 & \bar{W} = 17 & \\
\hline
x_6 = 1 & \bar{W} = 8 & \\
x_2 = 1 & \bar{W} = 2 & \\
x_4 = \frac{2}{3} & \bar{W} = 0 & x_5 \text{ critica}
\end{array}
$$

$UB_5 = 0 + 0 + 30 + 15 + \lfloor \frac{2}{3} 5 \rfloor = 48$

$UB_5 = 48 \geq 48$ (valore della migliore soluzione disponibile): chiudo P_5 perché non migliorante (e non valuto la soluzione ammissibile corrispondente, che non migliorerà sicuramente quella corrente).

$\boxed{\text{Nodo P6}}$:

$$
\begin{array}{lll}
x_1 = 0 & \bar{W} = 17 & \\
x_5 = 1 & \bar{W} = 12 & \\
\hline
x_6 = 1 & \bar{W} = 3 & \\
x_2 = \frac{3}{6} & \bar{W} = 0 & x_5 \text{ critica}
\end{array}
$$

$UB_6 = 0 + 11 + 30 + \lfloor \frac{3}{6} 15 \rfloor = 48$

$UB_6 = 48 \geq 48$ (valore della migliore soluzione disponibile): chiudo P_6 perché non migliorante (e non valuto la soluzione ammissibile corrispondente, che non migliorerà sicuramente quella corrente).

A questo punto tutti i nodi sono chiusi e il valore ottimo della funzione obiettivo è 48. Una soluzione ottima è quella trovata come soluzione ammissibile al nodo P_0, e cioè: $x_6 = x_2 = x_3 = 1$, $x_1 = x_4 = x_5 = 0$.