Ricerca Operativa
a.a. 2018-2019

SAPIENZA
UNIVERSITÀ DI ROMA

Lavinia Amorosi
Lezione XVII – 3 aprile 2019

Enumerazione Totale e Enumerazione Implicita

Soluzione di un problema di PLI

NOTA

Tratteremo i problemi di PLI facendo l'assunzione (non restrittiva nelle applicazioni reali) che la regione ammissibile sia costituita da un numero finito di punti.

Come già osservato, tecnicamente ciò corrisponde ad assumere che il poliedro P_Q sia un politopo.

Per semplificare la trattazione del B&B, considereremo il caso $X = P_Q \cap \{0,1\}^n$:

Problema di OC		**Problema di PLI**
$\max c^T x$	(Q)	$\max c^T x$
$x \in X$		$x \in P_Q$
		$x \in \{0,1\}^n$

Osservazione

Non si tratta di un caso meno generale di $X = P_Q \cap Z^n$, perchè, dal punto di vista teorico, una variabile intera che assume un numero finito k di valori può essere sempre sostituita da k variabili binarie a valori 0/1 (una per ognuno dei k valori originali).

Soluzione di un problema di PLI

Quando il numero di soluzioni è finito (come per il problema di OC (Q)) è teoricamente possibile ricorrere a una tecnica di enumerazione totale per risolvere (Q).

ALGORITMO DI ENUMERAZIONE TOTALE

Indichiamo con x^* la migliore soluzione ammissibile corrente (cioè il migliore tra i vettori ammissibili valutati dall'algoritmo fino all'iterazione corrente).

Per ogni vettore $x \in \{0,1\}^n$ bisogna:

1. verificare l'ammissibilità ($x \in X$)
2. calcolare il valore della f.o. corrispondente a x;
3. valutare se x è migliore di x^*, nel qual caso $\rightarrow$ $x^* := x$.

Nota

Il numero totale di vettori binari in R^n è pari a 2^n.

L'enumerazione totale **è una tecnica non praticabile in generale**, ma si può utilizzare solo per problemi combinatori con un numero di soluzioni **molto basso** (nel caso binario ciò corrisponde a casi con poche variabili).

Soluzione di un problema di PLI

ENUMERAZIONE TOTALE

Tecnica **CORRETTA**

ma **INGENUA**
(esplorazione dell'**intero spazio** delle soluzioni intere)

BRANCH (RAMIFICA) & BOUND (LIMITA)

enumerazione INTELLIGENTE
(esplorazione **parziale** dello spazio delle soluzioni intere)

ENUMERAZIONE INTELLIGENTE

ESPLORAZIONE PARZIALE

ENUMERAZIONE IMPLICITA

Soluzione di un problema di PLI

ESPLORAZIONE PARZIALE

- **Viene analizzato direttamente solo un sottoinsieme delle soluzioni.**

  Al fine di ridurre il più possibile il tempo totale di calcolo, tale sottoinsieme **dovrebbe essere il più piccolo possibile** (di cardinalità $k \ll 2^n$).

  [**enum. esplicita** ↔ analisi diretta ↔ operazione computaz. **onerosa**].

- **Le altre soluzioni vengono analizzate in maniera "implicita".**

  In base alla verifica di alcune condizioni, durante la risoluzione del problema si può decidere quali soluzioni devono essere valutate direttamente e quali invece presentano caratteristiche che permettono di **"scartarle" senza mai analizzarle direttamente**.

  Al fine di ridurre il più possibile il tempo totale di calcolo, l'insieme delle soluzioni di questo tipo **dovrebbe essere il più grande possibile**.

  [**enum. implicita** ↔ analisi indiretta ↔ op. computaz. **non onerosa**].

(Papadimitriou e Steiglitz, 1982): "Ad ogni iterazione, il B&B cerca di costruire una dimostrazione che la soluzione corrente (la migliore fin qui) è proprio quella ottima basandosi su **partizioni successive** dell'insieme delle soluzioni ammissibili del problema".

Soluzione di un problema di PLI

Partizione: per una fissata variabile x_j, utilizzando l'informazione che essa può assumere solo valore 0 oppure 1, è possibile generare facilmente una (bi-)partizione dell'insieme delle soluzioni del problema (Q):

$(Q=Q^0)$ ($x_j \in \{0,1\}$)

$$\begin{aligned} &\max c^T x \\ &x \in P_Q \\ &x \in \{0,1\}^n \end{aligned}$$

(Q^1) ($x_j=0$)

$$\begin{aligned} &\max c^T x \\ &x \in P_Q \\ &x \in \{0,1\}^n \\ &x_j=0 \end{aligned}$$

(Q^2) ($x_j=1$)

$$\begin{aligned} &\max c^T x \\ &x \in P_Q \\ &x \in \{0,1\}^n \\ &x_j=1 \end{aligned}$$

(Papadimitriou e Steiglitz, 1982): "Ad ogni iterazione, il B&B cerca di costruire una dimostrazione che la soluzione corrente (la migliore fin qui) è proprio quella ottima basandosi su **partizioni successive** dell'insieme delle soluzioni ammissibili del problema".

Soluzione di un problema di PLI

Partizione: per una fissata variabile x_j, utilizzando l'informazione che essa può assumere solo valore 0 oppure 1, è possibile generare facilmente una (bi-)partizione dell'insieme delle soluzioni del problema (Q):

(Q=Q⁰) ($x_j \in \{0,1\}$)

$$\max c^T x \quad x \in P_Q \quad x \in \{0,1\}^n$$

(Q^1) ($x_j=0$)

$$\max c^T x \quad x \in P_Q \quad x \in \{0,1\}^n \quad x_j=0$$

(Q^2) ($x_j=1$)

$$\max c^T x \quad x \in P_Q \quad x \in \{0,1\}^n \quad x_j=1$$

Partizione: Dato un insieme S e la famiglia di insiemi $\{S_1,\dots,S_q\}$, $q \geq 2$, essa è una **partizione di S** se:

i. $S_i \cap S_j = \emptyset$ per ogni i,j in {1,…,q}

ii. $\bigcup_{i=1,\dots,q} S_i = S$

$S = P_Q \cap \{0,1\}^n$

$S_1 = P_Q \cap \{0,1\}^n \cap \{x \in R^n: x_j=0\}$

$S_2 = P_Q \cap \{0,1\}^n \cap \{x \in R^n: x_j=1\}$

Soluzione di un problema di PLI

Per un problema di minimo:

$$(Q=Q^0) \quad \begin{array}{l} \min c^Tx \\ x \in P_Q \\ x \in \{0,1\}^n \end{array}$$

$$(Q^1) \quad \begin{array}{l} \min c^Tx \\ x \in P_Q \\ x \in \{0,1\}^n \\ x_j=0 \end{array}$$

$$(Q^2) \quad \begin{array}{l} \min c^Tx \\ x \in P_Q \\ x \in \{0,1\}^n \\ x_j=1 \end{array}$$

Esempio (min)

$$(Q) \quad \begin{array}{r} \min \; -x_1 - x_2 \\ 2x_1 - 5x_2 \geq -5 \\ -2x_1 + 2x_2 \geq -1 \\ x_1, x_2 \in \{0,1\} \end{array}$$

$P_Q = \{x \in R^2 \colon 2x_1 - 5x_2 \geq -5,\ -2x_1 + 2x_2 \geq -1,\ 0 \leq x_1, x_2 \leq 1 \}$

Soluzione di un problema di PLI

Per un problema di minimo:

$$(Q=Q^0)\quad \min c^Tx \quad \left\{\begin{array}{l} x\in P_Q \\ x\in \{0,1\}^n \end{array}\right. \qquad \leftarrow X_Q$$

$$(Q^1)\quad \min c^Tx \quad \left.\begin{array}{l} x\in P_Q \\ x\in \{0,1\}^n \\ x_1=0 \end{array}\right\} X_Q^1 \qquad P_Q^1 = P_Q \cap \{x\in R^n: x_1=0\}$$

$$(Q^2)\quad \min c^Tx \quad \left.\begin{array}{l} x\in P_Q \\ x\in \{0,1\}^n \\ x_1=1 \end{array}\right\} X_Q^2 \qquad P_Q^2 = P_Q \cap \{x\in R^n: x_1=1\}$$

Esempio (min)

$$(Q)\quad \begin{array}{rl} \min & -x_1 - x_2 \\ & 2x_1 - 5x_2 \geq -5 \\ & -2x_1 + 2x_2 \geq -1 \\ & x_1, x_2 \in \{0,1\} \end{array}$$

$P_Q = \{x\in R^2: 2x_1 - 5x_2 \geq -5,\ -2x_1 + 2x_2 \geq -1,\ 0 \leq x_1,x_2 \leq 1\}$

L'insieme delle soluzioni ammissibili di (Q) è:

$S = X_Q = \{O = (0,0);\ A = (1,1);\ B = (0,1)\}$

$x_1 = 0 \swarrow \qquad \searrow x_1 = 1$

$S_1 = X_Q^1 = \{O = (\mathbf{0},0);\ B = (\mathbf{0},1)\}$ $\qquad S_2 = X_Q^2 = \{A = (\mathbf{1},1)\}$

La Tecnica Branch&Bound per la soluzione di problemi di OC (problemi di PLI)

Strategia del B&B

Consideriamo il problema (Q) e indichiamo con X_Q l'insieme delle sue soluzioni ammissibili, $X_Q = P_Q \cap \{0,1\}^n$:

$$\boxed{\begin{array}{l} \max c^T x \\ \quad x \in X_Q \end{array}} \quad (Q=Q^0)$$

Partendo da $X_{Q^0} = X_Q$, la tecnica B&B:

1. opera una **partizione sistematica** dell'insieme delle soluzioni del problema (Q) generando una sequenza di sottoinsiemi di soluzioni

$$\mathbf{X_{Q^0} = X_Q} \,,\ X_{Q^1} \,,\ X_{Q^2} \,,\ \dots \,,\ X_{Q^h} \,,\ \dots$$

che corrispondono ai **sotto-problemi interi**

1.

$$\mathbf{(Q^0) = (Q)} \,,\ (Q^1) \,,\ (Q^2) \,,\ \dots \,,\ (Q^h) \,,\ \dots$$

$$\boxed{(Q^h) \quad \begin{array}{l} \max c^T x \\ \quad x \in X_{Q^h} \end{array}}$$

Sottoproblema intero (Q^h)

Strategia del B&B

Consideriamo il problema (Q) e indichiamo con X_Q l'insieme delle sue soluzioni ammissibili, $X_Q = P_Q \cap \{0,1\}^n$:

$$\boxed{\begin{array}{l} \max c^T x \\ \quad x \in X_Q \end{array}} \quad (Q=Q^0)$$

Partendo da $X_{Q^0} = X_Q$, la tecnica B&B:

1. opera una **partizione sistematica** dell'insieme delle soluzioni del problema (Q) generando una sequenza di sottoinsiemi di soluzioni

$$\mathbf{X_{Q^0} = X_Q}\ ,\ X_{Q^1}\ ,\ X_{Q^2}\ ,\ \dots\ ,\ X_{Q^h}\ ,\ \dots$$

che corrispondono ai **sotto-problemi interi**

$$\mathbf{(Q^0) = (Q)}\ ,\ (Q^1)\ ,\ (Q^2)\ ,\ \dots\ ,\ (Q^h)\ ,\ \dots$$

$$\boxed{(Q^h) \qquad \begin{array}{l} \max c^T x \\ \quad x \in X_{Q^h} \end{array}}$$

Sottoproblema intero (Q^h)

2. risolve la corrispondente sequenza di problemi di PL che corrispondono ai rilassamenti continui dei problemi interi

$$\mathbf{(P_{Q^0}) = (P_Q)}\ ,\ (P_{Q^1})\ ,\ (P_{Q^2})\ ,\ \dots\ ,\ (P_{Q^h})\ ,\ \dots$$

$$\boxed{(P_{Q^h}) \qquad \begin{array}{l} \max c^T x \\ \quad x \in P_{Q^h} \end{array}}$$

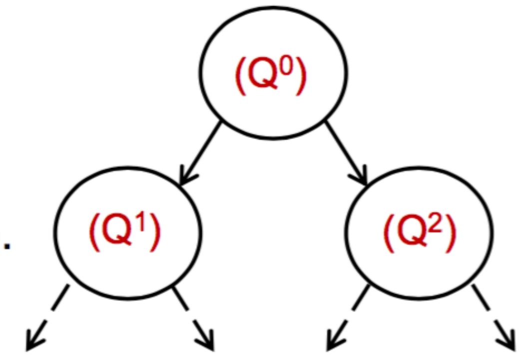
Rilassamento continuo di (Q^h)

con $X_{Q^h} = P_{Q^h} \cap \{0,1\}^n$ per ogni h.

Strategia del B&B

La sequenza dei problemi interi viene generata attraverso la costruzione di un **albero di ricerca** (detto albero del B&B):

- nel caso **$x \in \{0,1\}^n$** l'albero è **binario**;
- la **radice** dell'albero è associata al **problema (Q^0)**.

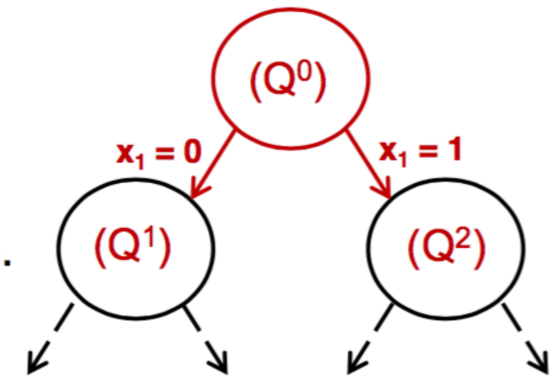


La tecnica B&B segue una strategia di tipo "Divide-and-Conquer", cioè "Dividi-e-Conquista", infatti:

Strategia del B&B

La sequenza dei problemi interi viene generata attraverso la costruzione di un **albero di ricerca** (detto albero del B&B):

- nel caso **$x \in \{0,1\}^n$** l'albero è **binario**;
- la **radice** dell'albero è associata al **problema (Q^0)**.



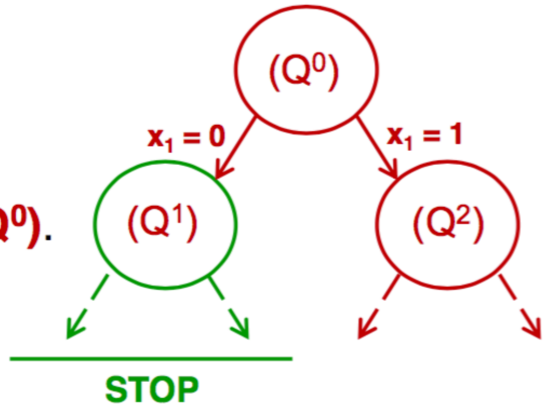
La tecnica B&B segue una strategia di tipo "Divide-and-Conquer", cioè "Dividi-e-Conquista", infatti:

- partiziona lo spazio delle soluzioni di (Q) per ottenere sotto-problemi "**più semplici da risolvere**" e in cui aumenta il numero di componenti del vettore x **fissate ad un valore intero** (Divide/Branch);

Strategia del B&B

La sequenza dei problemi interi viene generata attraverso la costruzione di un **albero di ricerca** (detto albero del B&B):

- nel caso **$x \in \{0,1\}^n$** l'albero è **binario**;
- la **radice** dell'albero è associata al **problema (Q^0)**.



La tecnica B&B segue una strategia di tipo "Divide-and-Conquer", cioè "Dividi-e-Conquista", infatti:

- partiziona lo spazio delle soluzioni di (Q) per ottenere sotto-problemi "**più semplici da risolvere**" e in cui aumenta il numero di componenti del vettore x **fissate ad un valore intero** (Divide/Branch);

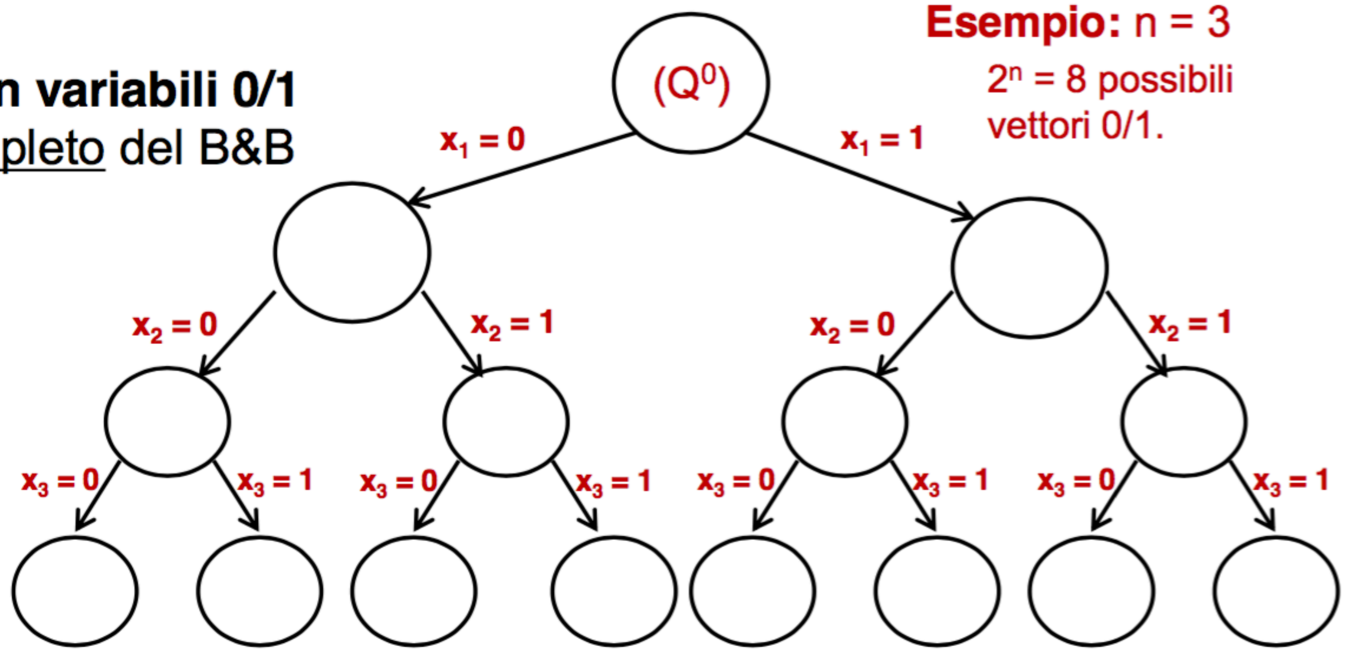
- per ogni sottoproblema (Q^h) risolve il rilassamento continuo (P_{Q^h}) e dal risultato è in grado di distinguere i casi in cui è necessario partizionare ulteriormente l'insieme delle soluzioni da quelli in cui, invece, non è necessario procedere oltre nell'analisi. In ogni caso **acquisisce maggiore informazione** sulla struttura della soluzione ottima (Conquer/Bound).

Strategia del B&B

NOTA

Per un problema con **n variabili 0/1** l'albero di ricerca <u>completo</u> del B&B ha **2^n foglie**.

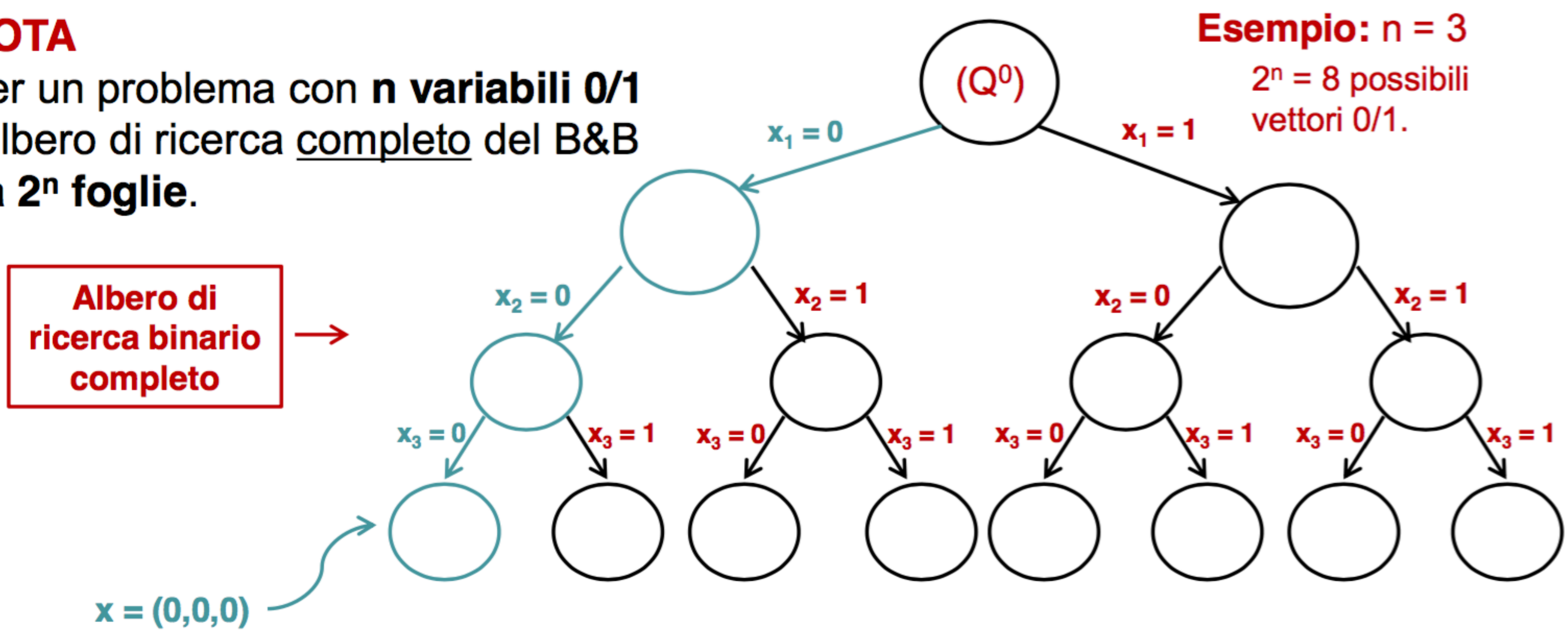
Albero di ricerca binario completo →



Strategia del B&B

NOTA

Per un problema con **n variabili 0/1** l'albero di ricerca <u>completo</u> del B&B ha **2^n foglie**.



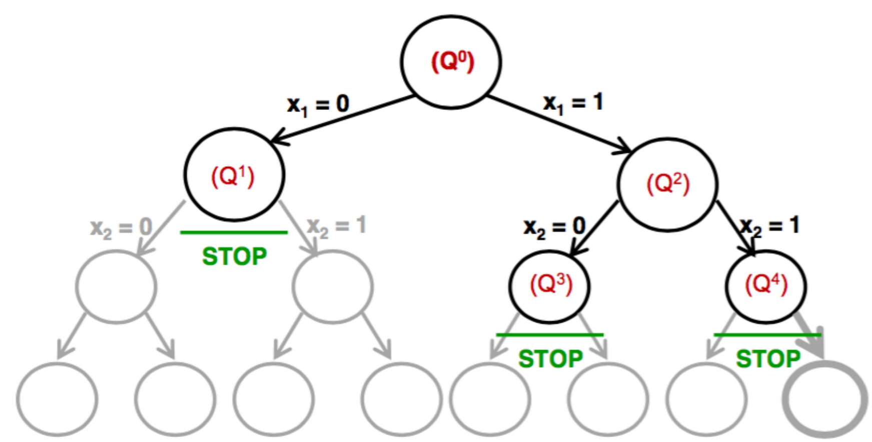
Strategia del B&B

NOTA

Esempio: n = 3

Nell'albero di ricerca del B&B il nodo radice può essere **sempre etichettato con 0**, mentre i successivi nodi verranno **etichettati progressivamente con 1,2,...** seguendo l'ordine in cui essi vengono generati e analizzati dall'algoritmo.

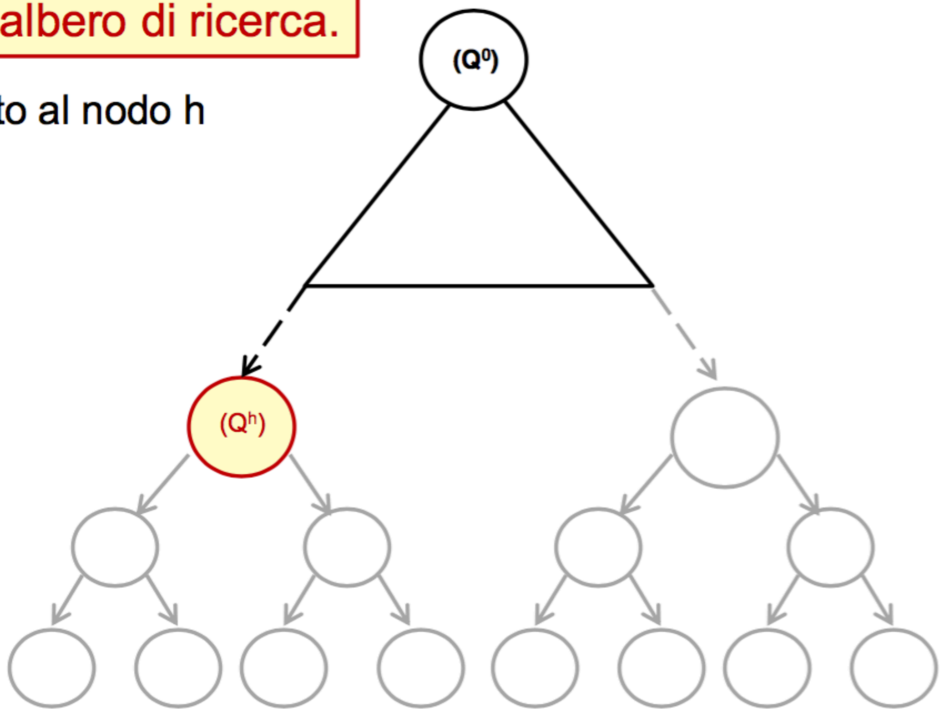
ENUMERAZIONE IMPLICITA (PARZIALE)



Branching

Consideriamo il generico nodo h dell'albero di ricerca.

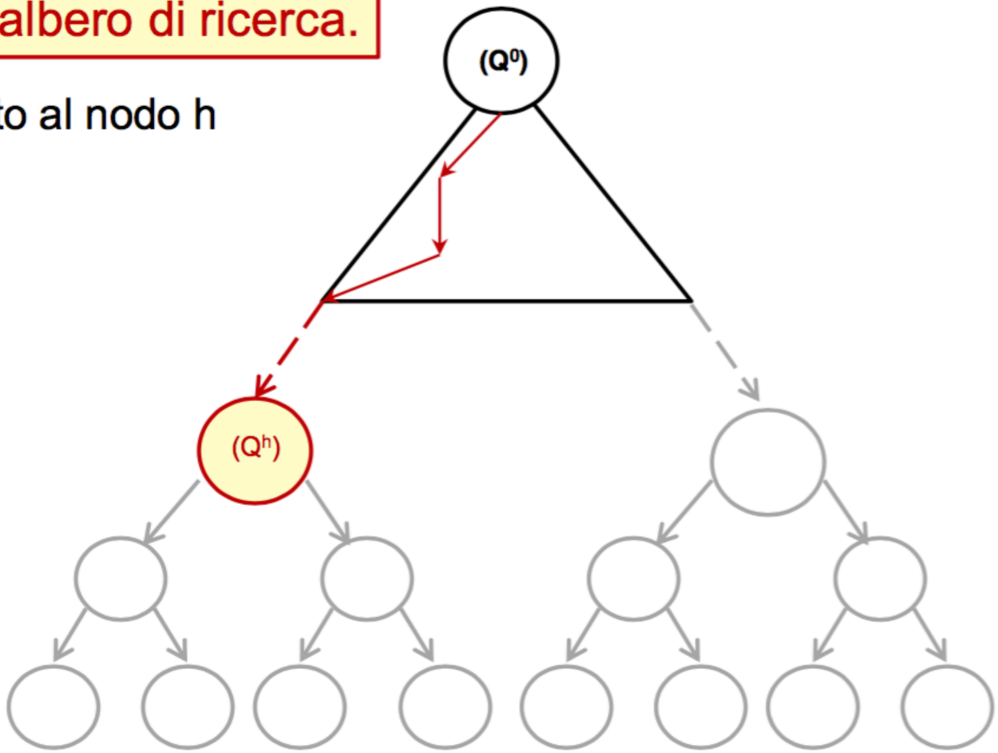
(Q^h)	sotto-problema (intero) associato al nodo h
X_{Q^h}	regione ammissibile di (Q^h)
(P_{Q^h})	rilassamento continuo di (Q^h)



Branching

Consideriamo il generico nodo h dell'albero di ricerca.

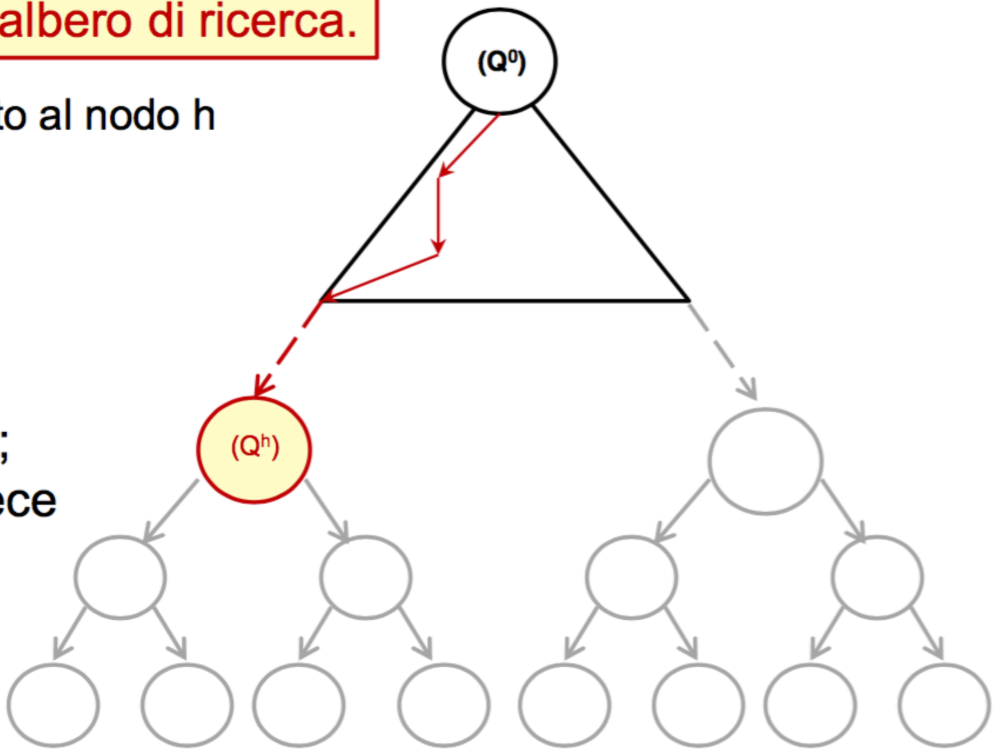
(Q^h)	sotto-problema (intero) associato al nodo h
X_{Q^h}	regione ammissibile di (Q^h)
(P_{Q^h})	rilassamento continuo di (Q^h)



Branching

Consideriamo il generico nodo h dell'albero di ricerca.

(Q^h)	sotto-problema (intero) associato al nodo h
X_{Q^h}	regione ammissibile di (Q^h)
(P_{Q^h})	rilassamento continuo di (Q^h)



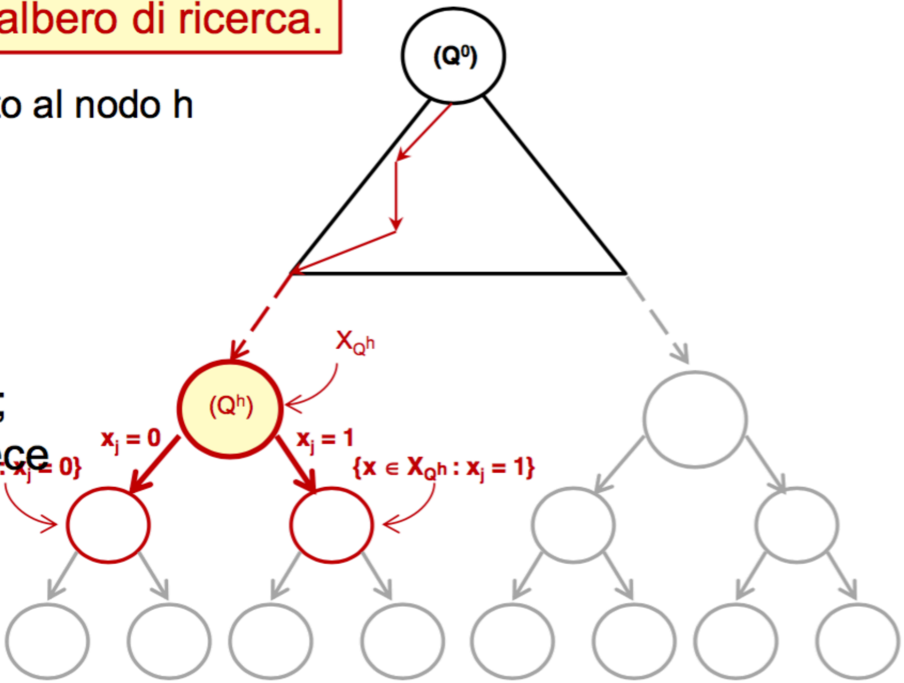
NOTA In (Q^h) solo alcune delle variabili binarie sono ancora incognite; alcune variabili binarie sono state invece fissate (al valore 0 o 1) durante la visita dell'albero dalla radice fino al nodo h.

Branching

Consideriamo il generico nodo h dell'albero di ricerca.

(Q^h)	sotto-problema (intero) associato al nodo h
X_{Q^h}	regione ammissibile di (Q^h)
(P_{Q^h})	rilassamento continuo di (Q^h)

NOTA In (Q^h) solo alcune delle variabili binarie sono ancora incognite; alcune variabili binarie sono state invece fissate (al valore 0 o 1) durante la visita dell'albero dalla radice fino al nodo h.



Branching al nodo h risp. alla variabile x_j

Una operazione di branching al nodo h **risp. a x_j** consiste nel partizionare la regione X_{Q^h} in due sotto-regioni tali che i vettori ammissibili appartenenti all'una differiscano da quelli dell'altra per la sola componente x_j.

Il branching:

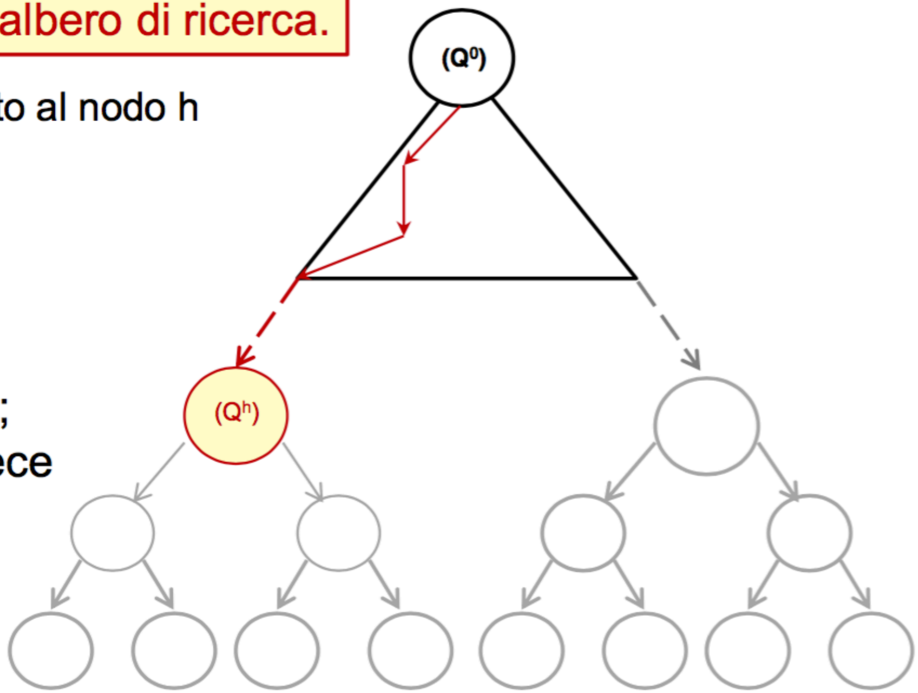
- **genera 2 nodi 'figli di h'**
- **fissa il valore della variabile x_j** (il numero delle incognite diminuisce di 1)

Branching

Consideriamo il generico nodo h dell'albero di ricerca.

(Q^h)	sotto-problema (intero) associato al nodo h
X_{Q^h}	regione ammissibile di (Q^h)
(P_{Q^h})	rilassamento continuo di (Q^h)

NOTA In (Q^h) solo alcune delle variabili binarie sono ancora incognite; alcune variabili binarie sono state invece fissate (al valore 0 o 1) durante la visita dell'albero dalla radice fino al nodo h.



Al nodo h è associata anche una **"soluzione parziale"** x^h del problema originale (Q).

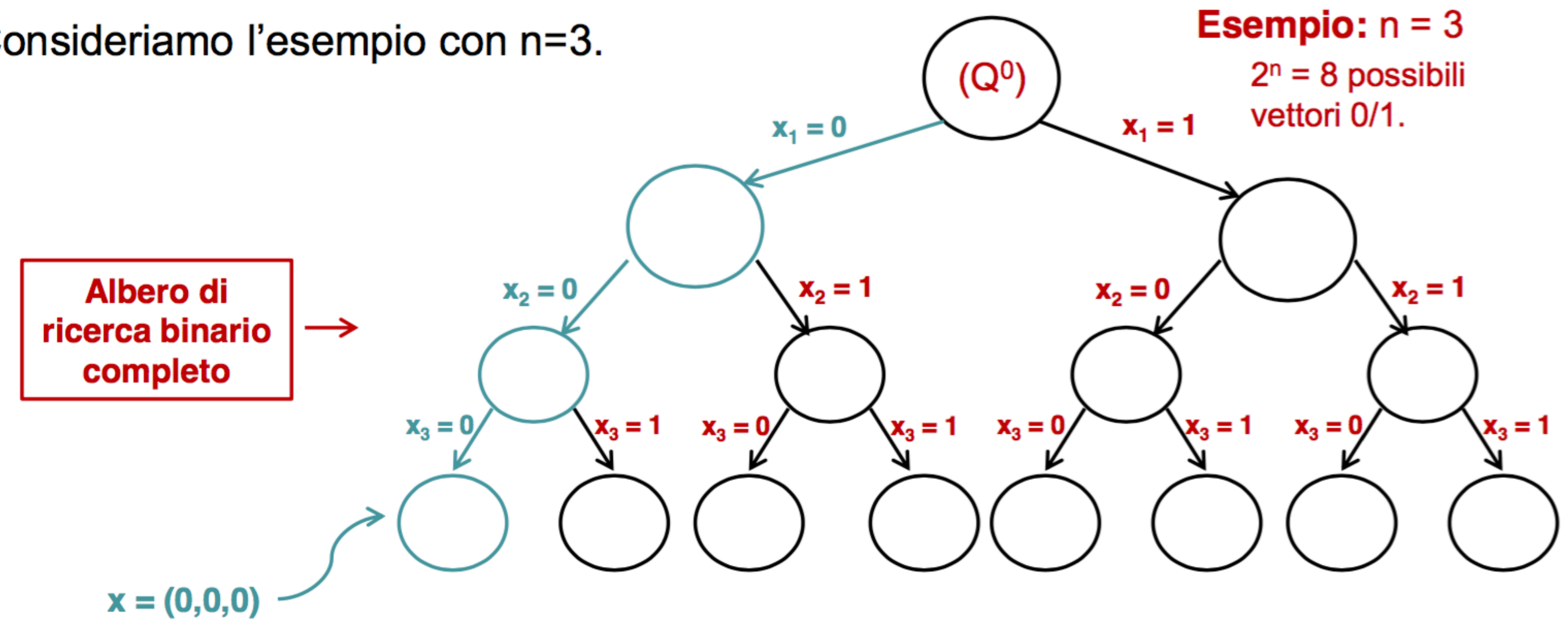
Il branching:

- **genera 2 nodi 'figli di h'**
- **fissa il valore della variabile x_j**

(il numero delle incognite diminuisce di 1)

Strategia del B&B

Consideriamo l'esempio con n=3.



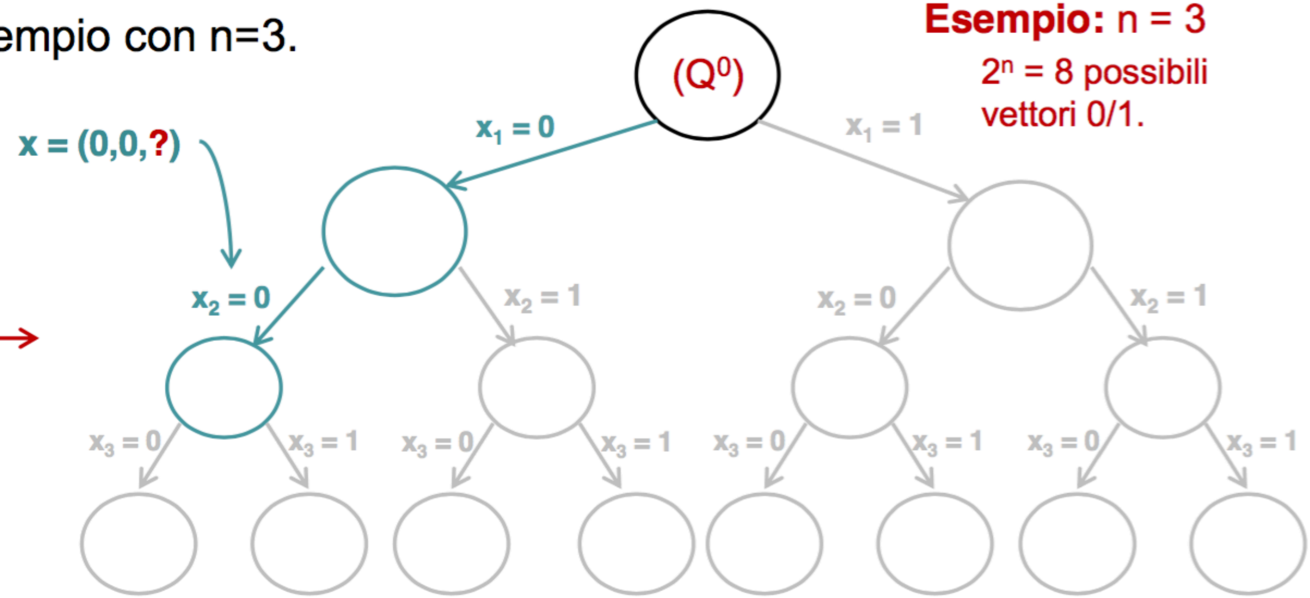
Strategia del B&B

Consideriamo l'esempio con n=3.

Esempio: n = 3

$2^n = 8$ possibili vettori 0/1.

Albero di ricerca binario completo →



Fissando x_1=0 e x_2=0 si ottiene una soluzione parziale (0,0,$\mathbf{x_3}$) con x_3 incognita che può essere 'completata' generando la soluzione (0,0,**0**) oppure (0,0,**1**).

Soluzioni parziali e completamenti

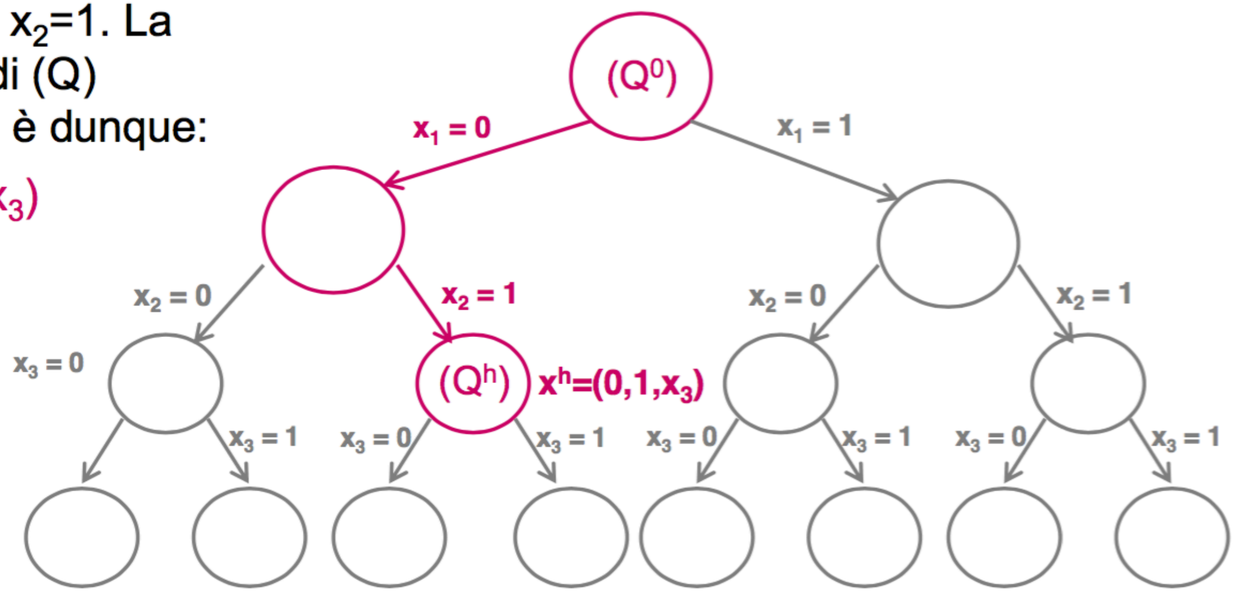
Soluzione parziale

Definizione (soluzione parziale x^h)
Un vettore $(x_1,...,x_j,...,x_n)$ è una soluzione parziale per il problema (Q) se una parte delle componenti è fissata (al valore 0 oppure a 1), mentre le restanti sono ancora incognite.

Esempio: n=3.

Al nodo h sono fissate le componenti $x_1=0$ e $x_2=1$. La soluzione parziale di (Q) associata al nodo h è dunque:

$x^h=(\mathbf{0},\mathbf{1},x_3)$



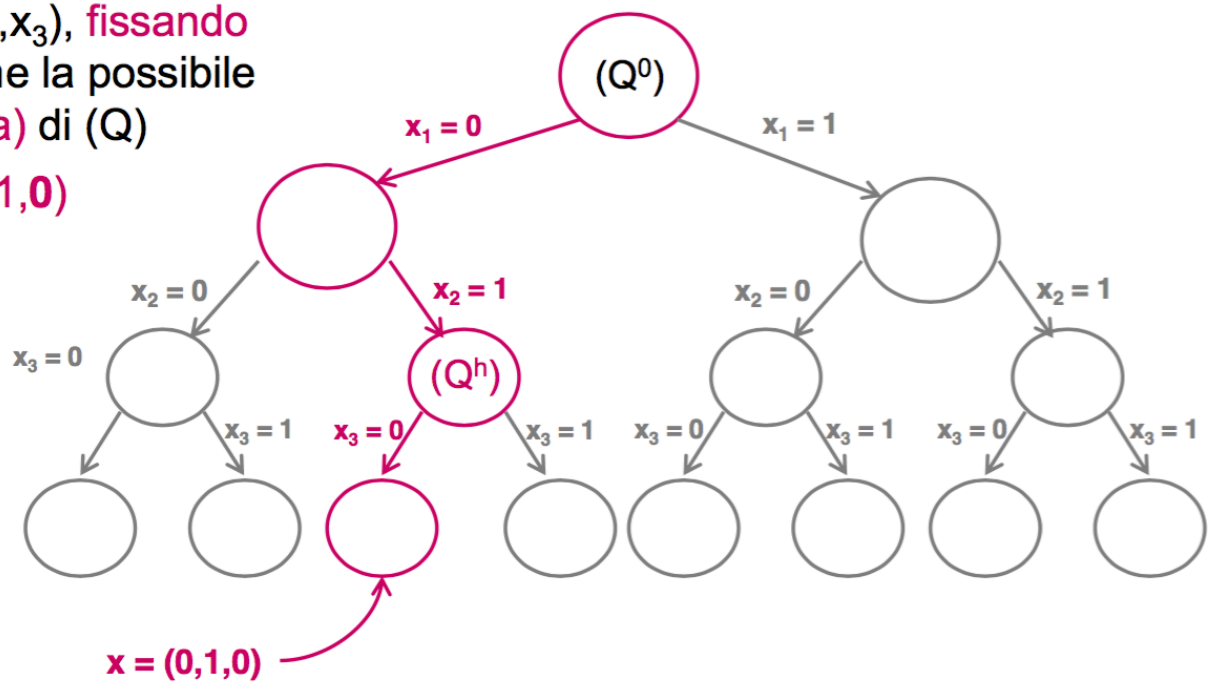
Soluzione parziale

Definizione (soluzione parziale x^h)
Un vettore $(x_1,\dots,x_j,\dots,x_n)$ è una soluzione parziale per il problema (Q) se una parte delle componenti è fissata (al valore 0 oppure a 1), mentre le restanti sono ancora incognite.

Esempio: n=3.

NOTA
A partire da $x^h=(0,1,x_3)$, fissando anche $x_3=0$ si ottiene la possibile soluzione (completata) di (Q)

$x = (0,1,\mathbf{0})$



Soluzione parziale

Definizione (soluzione parziale x^h)
Un vettore $(x_1,\ldots,x_j,\ldots,x_n)$ è una soluzione parziale per il problema (Q) se una parte delle componenti è fissata (al valore 0 oppure a 1), mentre le restanti sono ancora incognite.

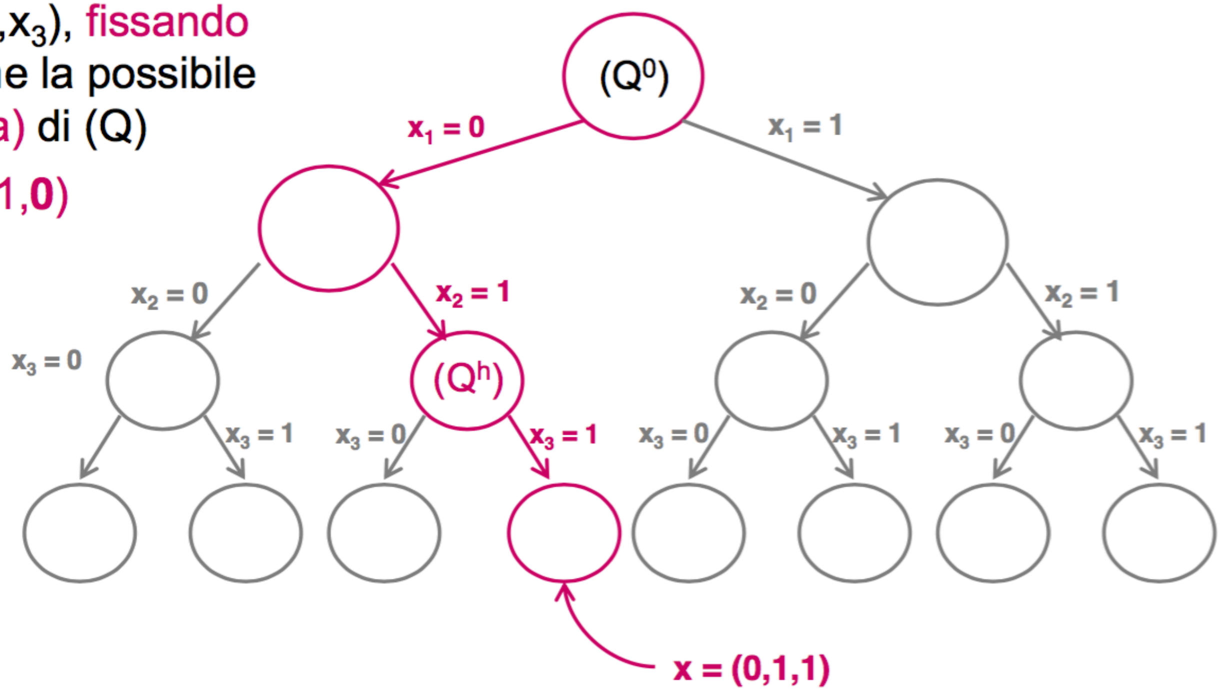
Esempio: n=3.

NOTA
A partire da $x^h=(0,1,x_3)$, fissando anche $x_3=0$ si ottiene la possibile soluzione (completa) di (Q)

$$x = (0,1,\mathbf{0})$$

Fissando $x_3=1$ si ottiene invece:

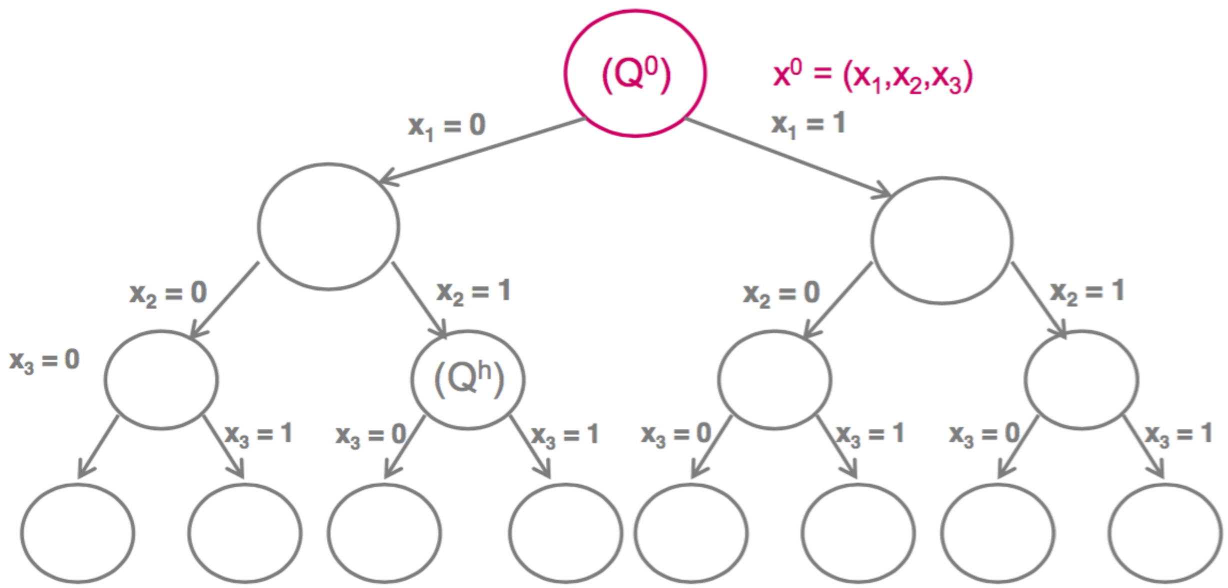
$$x = (0,1,\mathbf{1}).$$



Soluzione parziale

NOTA

Al nodo radice h=0 tutte le componenti del vettore x sono incognite (nessuna componente è stata ancora fissata).

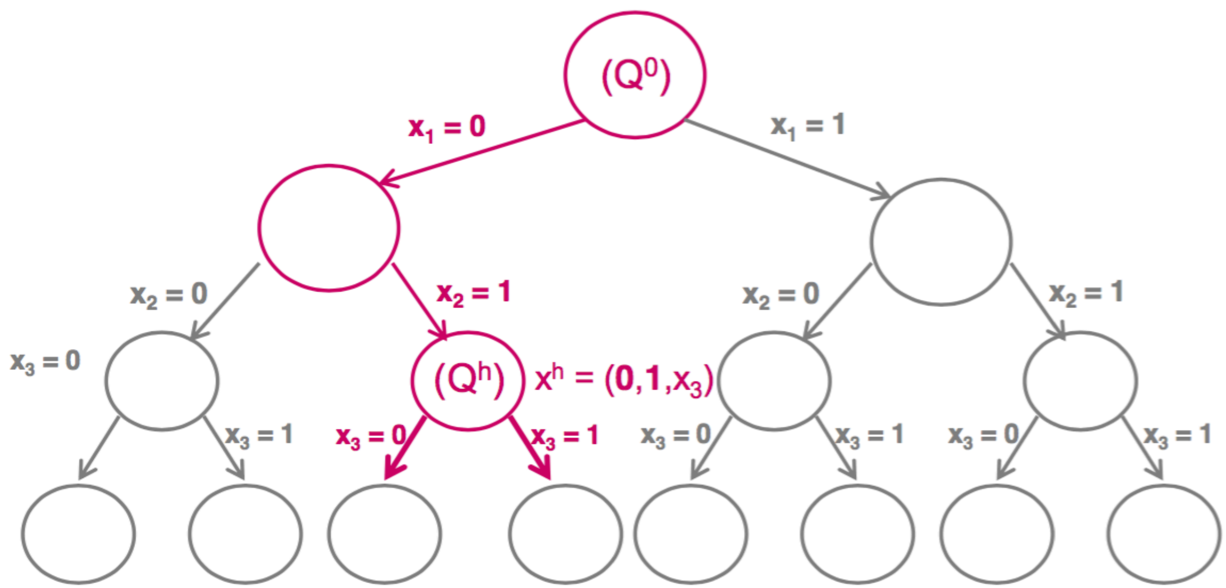


Soluzione parziale

NOTA

Al nodo radice h=0 tutte le componenti del vettore x sono incognite (nessuna componente è stata ancora fissata).

Una operazione di branching al generico nodo h fissa (a 0 o a 1) una nuova componente nel vettore della soluzione parziale x^h.



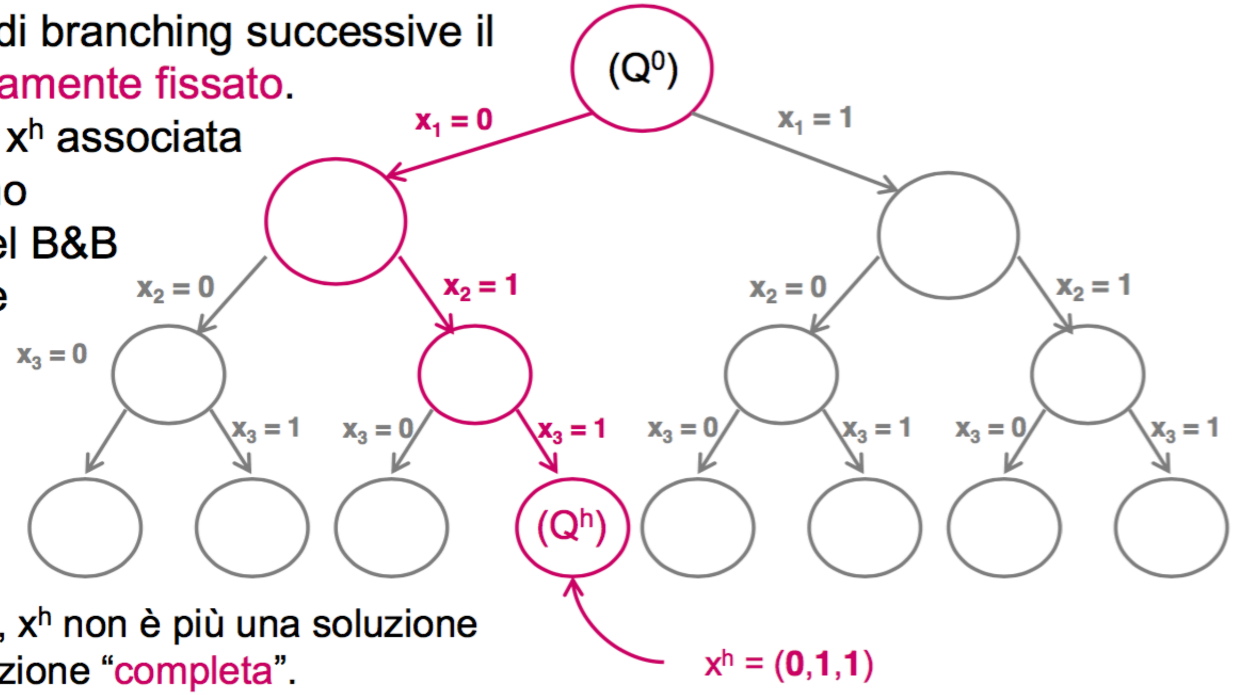
Soluzione parziale

NOTA

Al nodo radice h=0 tutte le componenti del vettore x sono incognite (nessuna componente è stata ancora fissata).

Una operazione di branching al generico nodo h fissa (a 0 o a 1) una nuova componente nel vettore della soluzione parziale x^h.

Dopo n operazioni di branching successive il vettore x è completamente fissato. Perciò la soluzione x^h associata a un nodo dell'ultimo livello dell'albero del B&B (foglia dell'albero) è una delle soluzioni possibili (vettori binari in $\{0,1\}^n$).



NOTA

In effetti, in una foglia, x^h non è più una soluzione parziale, ma una soluzione "completa".

Completamento

Definizione (completamento di x^h)
Sia x^h una soluzione parziale del problema (Q). Si definisce completamento di x^h ogni vettore binario $(x_1,\ldots,x_j,\ldots,x_n)$ tale che:

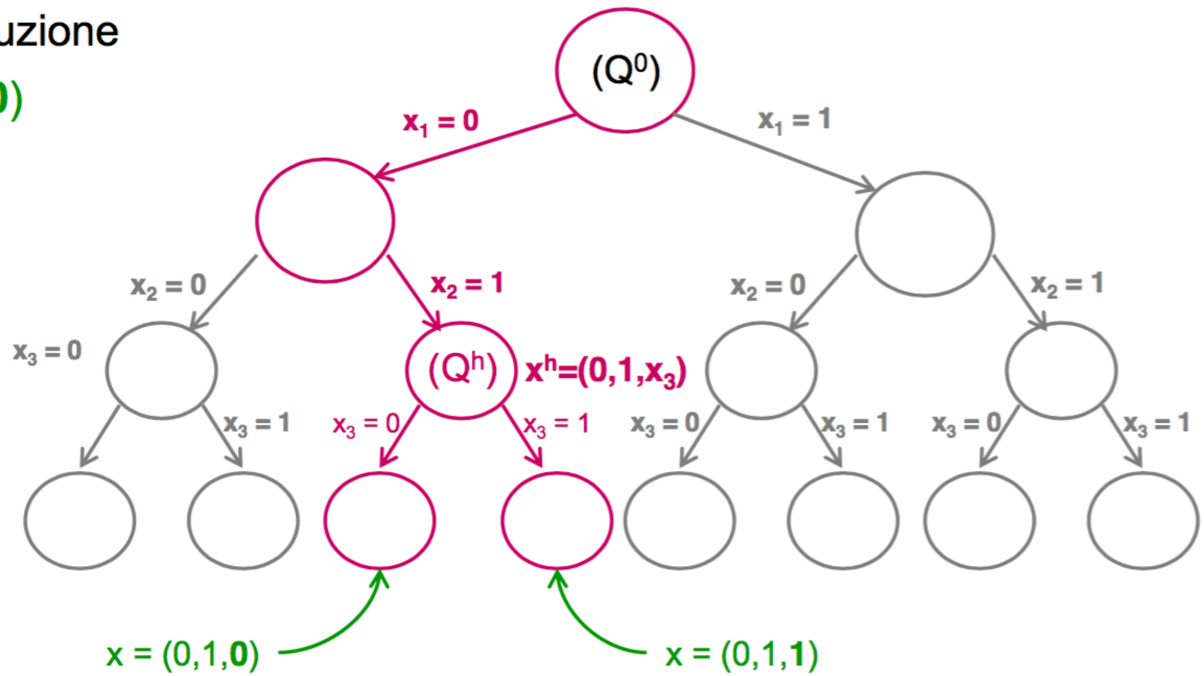
$$x_i = x_i^h \text{ per ogni i fissata in } x^h.$$

Esempio: n=3.

Un possibile completamento di $x^h=(0,1,\mathbf{x_3})$ è la soluzione

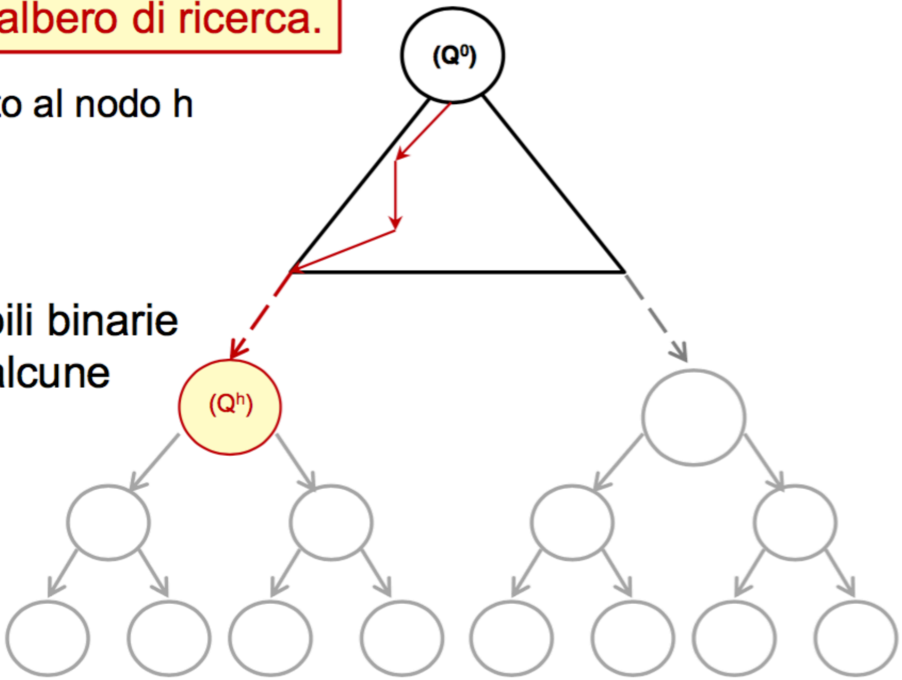
$$x=(0,1,\mathbf{0})$$

l'altro è $x=(0,1,\mathbf{1})$.



Branching

Consideriamo il generico nodo h dell'albero di ricerca.



(Q^h) sotto-problema (intero) associato al nodo h

X_{Q^h} regione ammissibile di (Q^h)

(P_{Q^h}) rilassamento continuo di (Q^h)

NOTA In (Q^h) solo alcune delle variabili binarie sono ancora incognite del problema; alcune variabili binarie sono state invece fissate (al valore 0 o 1) durante la visita dell'albero dalla radice fino al nodo h.

Al nodo h è associata anche una **"soluzione parziale"** x^h del problema originale (Q).

Il branching:

- **genera 2 nodi 'figli di h'**
- **fissa il valore della variabile x_j**

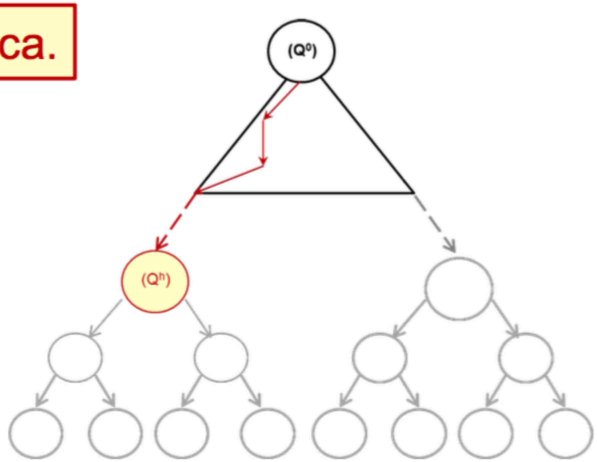
(il numero delle incognite diminuisce di 1)

Analisi al nodo h dell’albero del B&B

Problemi ristretti

Consideriamo il generico nodo h dell'albero di ricerca.

(Q^h)	sotto-problema (intero) associato al nodo h
X_{Q^h}	regione ammissibile di (Q^h)
(P_{Q^h})	rilassamento continuo di (Q^h)



Al nodo h sono definiti due problemi **ristretti**:

Problema intero ristretto

$$(Q^h) \qquad z_{X_{Q^h}} = \max c^T x \quad x \in X_{Q^h}$$

Rilassamento ristretto

$$(P_{Q^h}) \qquad z_{P_{Q^h}} = \max c^T x \quad x \in P_{Q^h}$$

Rilassamento di (Q^h)

con $X_{Q^h} = P_{Q^h} \cap \{0,1\}^n$.

Problemi ristretti

Nel caso di **max** il problema (PR) è un rilassamento di (P) se:

1. $X \subseteq Y$
2. $g(x) \geq f(x)$ per ogni $x \in X$.

Nel nostro caso il problema (P_{Q^h}) è un rilassamento di (Q^h):

1. $X_{Q^h} = P_{Q^h} \cap \{0,1\}^n \subseteq P_{Q^h}$
2. $z_{P_{Q^h}}(x) = z_{X_{Q^h}}(x)$ per ogni $x \in X$.

Proprietà 1 Se (P_{Q^h}) è non ammissibile $\rightarrow$ anche (Q^h) è non ammissibile

Corollario 1 Se $\overline{x}^h$ è ottima per (P_{Q^h}) e ammissibile per (Q^h) (a componenti intere) $\rightarrow$ $\overline{x}^h$ è ottima anche per (Q^h)

Problema intero ristretto

(Q^h)
$$z_{X_{Q^h}} = \max c^T x \quad x \in X_{Q^h}$$

Rilassamento ristretto

(P_{Q^h})
$$z_{P_{Q^h}} = \max c^T x \quad x \in P_{Q^h}$$

Rilassamento di (Q^h)

con $X_{Q^h} = P_{Q^h} \cap \{0,1\}^n$.

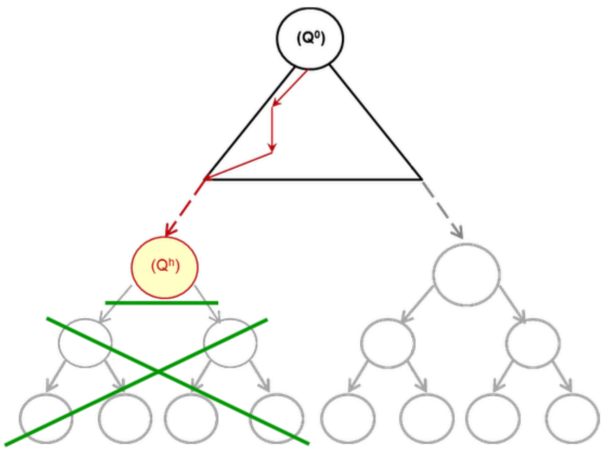
Analisi al nodo h

Tenendo conto della **Proprietà 1** e del **Corollario 1**, possiamo elencare e analizzare i seguenti **4 casi** che si possono presentare in corrispondenza di un nodo h dell'albero del B&B.

Indichiamo con x* la migliore soluzione ammissibile di (Q), e quindi **intera**, individuata dal B&B fino al nodo h (x* è la soluzione corrente).

1 **[INFEASIBILITY]**
(P_{Q^h}) è non ammissibile $\xrightarrow{\textbf{Proprietà 1}}$ (Q^h) è non ammissibile
[conclusione **implicita**]

Tutti i **sottoproblemi** di (Q^h) non sono ammissibili.
Tutti i **completamenti** della soluzione parziale x^h non sono ammissibili per (Q).



Analisi al nodo h

Tenendo conto della **Proprietà 1** e del **Corollario 1**, possiamo elencare e analizzare i seguenti **4 casi** che si possono presentare in corrispondenza di un nodo h dell'albero del B&B.

Indichiamo con x^* la migliore soluzione ammissibile di (Q), e quindi **intera**, individuata dal B&B fino al nodo h (x^* è la soluzione corrente).

1 **[INFEASIBILITY]**
(P_{Q^h}) è non ammissibile $\xrightarrow{\text{Proprietà 1}}$ (Q^h) è non ammissibile [conclusione **implicita**]

Tutti i **sottoproblemi** di (Q^h) non sono ammissibili.
Tutti i **completamenti** della soluzione parziale x^h non sono ammissibili per (Q).

2 **[SOLVING]**
(P_{Q^h}) è ammissibile e la sua soluzione ottima $\bar{x}^h$ **è a componenti intere** $\xrightarrow{\text{Corollario 1}}$ $\bar{x}^h$ è **ottima** anche **per** (Q^h) [conclusione **implicita**]

$\bar{x}^h$ è una soluzione "candidata" ad essere ottima per (Q).
Tutti i completamenti di x^h sono non migliori di $\bar{x}^h$.

Analisi al nodo h

3

[BOUNDING]

(P_{Q^h}) **è ammissibile** ma

la sua soluzione ottima $\overline{x}^h$

non è a componenti intere

Analisi al nodo h

3

[BOUNDING]

(P_{Q^h}) **è ammissibile** ma

la sua soluzione ottima $\overline{x}^h$

non è a componenti intere

e si ha: $\mathbf{z(x^*) \geq z(\overline{x}^h) = z_{PQ^h}}$

x* è la migliore soluzione intera corrente

Il problema è di max.

Tutte le soluzioni intere che potrebbero essere individuate procedendo con l'analisi nel sottoalbero radicato nel nodo h sono sicuramente non migliori di x*.

[conclusione **implicita**]

Analisi al nodo h

3 **[BOUNDING]**

(P_{Q^h}) **è ammissibile** ma

la sua soluzione ottima $\overline{x}^h$ non è a componenti intere

x* è la migliore soluzione intera corrente

e si ha: $z(x^*) \geq z(\overline{x}^h) = z_{P_{Q^h}}$

Tutte le soluzioni intere che potrebbero essere individuate procedendo con l'analisi nel sottoalbero radicato nel nodo h sono sicuramente non migliori di x*.

[conclusione **implicita**]

La soluzione ottima di (Q^h) e tutti i completamenti di x^h sono non migliori di x*.

Il problema è di max.

$$z_{X_{Q^h}} \leq z_{P_{Q^h}} = z(\overline{x}^h) \leq z(x^*)$$

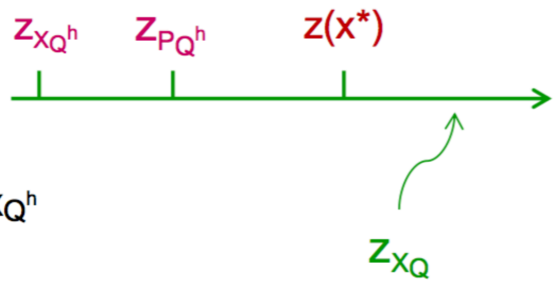
max

$z_{P_{Q^h}}$ è un upper bound per $z_{X_{Q^h}}$

La condizione di bounding corrisponde a un upper bound non utilizzabile perchè la soluzione x* (intera) è migliore della soluzione $\overline{x}^h$.

NOTA

Questo caso è possibile perchè $\overline{x}^h$ è la soluzione ottima di un problema rilassato **ristretto** a P_{Q^h}.



Il valore $z_{P_{Q^h}}$

è un upper bound per $z_{X_{Q^h}}$

ma non lo è per z_{X_Q}.

Analisi al nodo h

3 **[BOUNDING]**

(P_{Q^h}) **è ammissibile** ma

la sua soluzione ottima $\overline{x}^h$

non è a componenti intere

x* è la migliore soluzione intera corrente

e si ha: $z(x^*) \geq z(\overline{x}^h) = z_{P_{Q^h}}$

Tutte le soluzioni intere che potrebbero essere individuate procedendo con l'analisi nel sottoalbero radicato nel nodo h sono sicuramente non migliori di x*.

[conclusione **implicita**]

Il problema è di max.

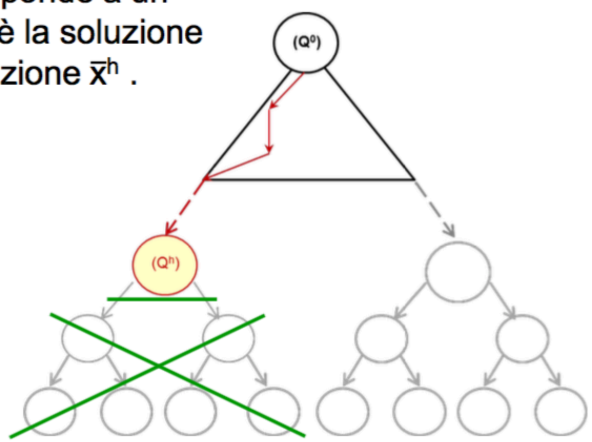
$$z_{X_{Q^h}} \leq z_{P_{Q^h}} = z(\overline{x}^h) \leq z(x^*)$$

max

$z_{P_{Q^h}}$ è un upper bound per $z_{X_{Q^h}}$

La soluzione ottima di (Q^h) e tutti i completamenti di x^h sono non migliori di x*.

La condizione di bounding corrisponde a un upper bound non utilizzabile perchè la soluzione x* (intera) è migliore della soluzione $\overline{x}^h$.



Analisi al nodo h

Se non si verifica uno dei casi precedenti, allora si verifica necessariamente il caso seguente.

4 **[BRANCHING]**

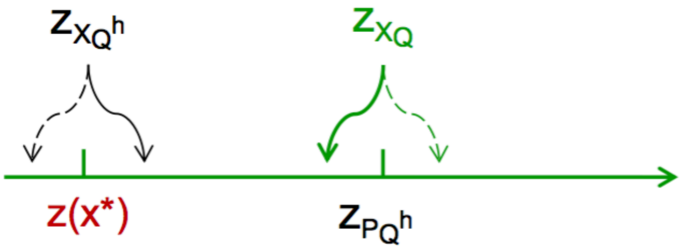
(P_{Q^h}) **è ammissibile** ma

la sua soluzione ottima $\overline{x}^h$

non è a componenti intere

e si ha: $z(x^*) < z(\overline{x}^h) = z_{P_{Q^h}}$

?

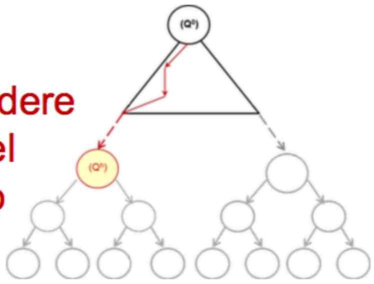


NOTA

In questo caso, al nodo h, non è possibile giungere a conclusioni implicite sui completamenti possibili di x^h.

Rimane aperta la possibilità che l'ottimo di (Q) sia tra $z(x^*)$ e $z_{P_{Q^h}}$, oppure che sia maggiore di $z_{P_{Q^h}}$.

È necessario procedere nell'esplorazione del sottoalbero radicato nel nodo h.



Analisi al nodo h

Se non si verifica uno dei casi precedenti, allora si verifica necessariamente il caso seguente.

4

[BRANCHING]

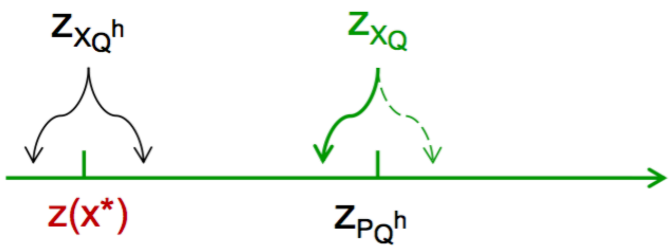
(P_{Q^h}) **è ammissibile** ma

la sua soluzione ottima $\overline{x}^h$

non è a componenti intere

e si ha: $z(x^*) < z(\overline{x}^h) = z_{PQ^h}$

È necessario procedere nella visita dell'albero del B&B attraverso una operazione di branching al nodo h risp. ad una delle componenti x_j non ancora fissate nella soluzione parziale x^h ad esso associata.

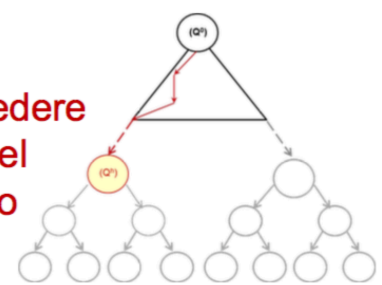


NOTA

In questo caso, al nodo h, non è possibile giungere a conclusioni implicite sui completamenti possibili.

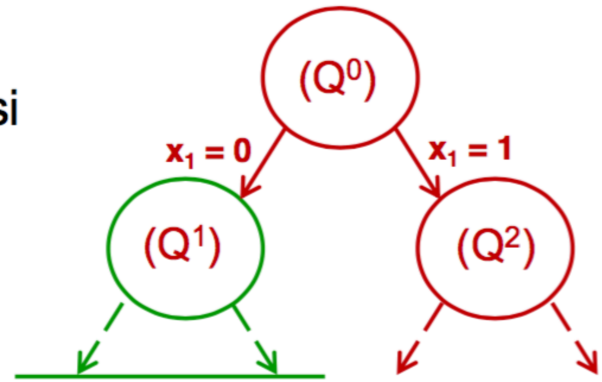
Rimane aperta la possibilità che l'ottimo di (Q) sia tra $z(x^*)$ e z_{PQ^h}, oppure che sia maggiore di z_{PQ^h}.

È necessario procedere nell'esplorazione del sottoalbero radicato nel nodo h.



Terminazione

Nei casi 1, 2, 3, la visita dell'albero del B&B si arresta al nodo h (**conclusioni implicite**), dato che – per motivi diversi – in ciascuno di questi casi è possibile trarre conclusioni sui completamenti di x^h rispetto al problema originale (Q).



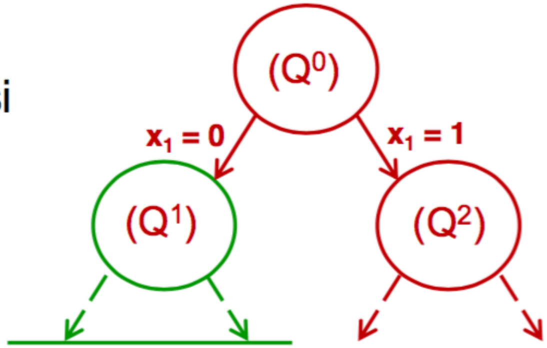
STOP:
Ogni sottoproblema è non ammissibile.
Tutti i completamenti della soluzione parziale x^h non sono ammissibili per (Q).

1 **[INFEASIBILITY]**

$X_{Q^h} = \emptyset$ ➡ tutti i sottoinsiemi di X_{Q^h} sono vuoti

Terminazione

Nei casi 1, 2, 3, la visita dell'albero del B&B si arresta al nodo h (**conclusioni implicite**), dato che – per motivi diversi – in ciascuno di questi casi è possibile trarre conclusioni sui completamenti di x^h rispetto al problema originale (Q).



STOP:
È stata individuatauna <u>possibile</u> soluzione ottima di (Q) e tutti i completamenti di x^h sono non migliori di x^h

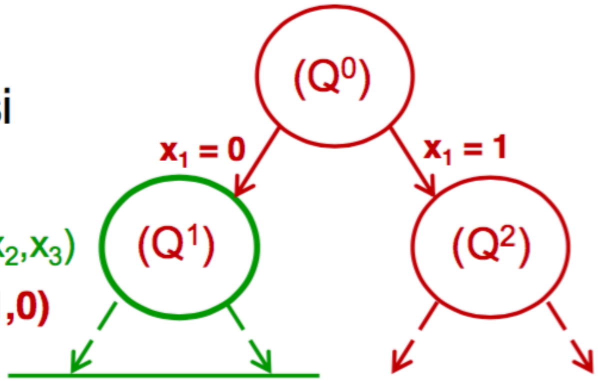
2 **[SOLVING]**

$\overline{x}^h$ è a componenti intere

$\overline{x}^h$ è il **completamento ottimo** di x^h

Terminazione

Nei casi 1, 2, 3, la visita dell'albero del B&B si arresta al nodo h (**conclusioni implicite**), dato che – per motivi diversi – in ciascuno di questi casi è possibile trarre conclusioni sui completamenti di x^h rispetto al problema originale (Q).



$x^h = (\mathbf{0}, x_2, x_3)$

Soluzione parziale x^h ↗ $\bar{x}^h = (0,1,0)$

Soluzione ottima del rilassamento ristretto (P_{Q^h}). ↗

STOP:
È stata individuatauna <u>possibile</u> soluzione ottima di (Q) e tutti i completamenti di x^h sono non migliori di x^h

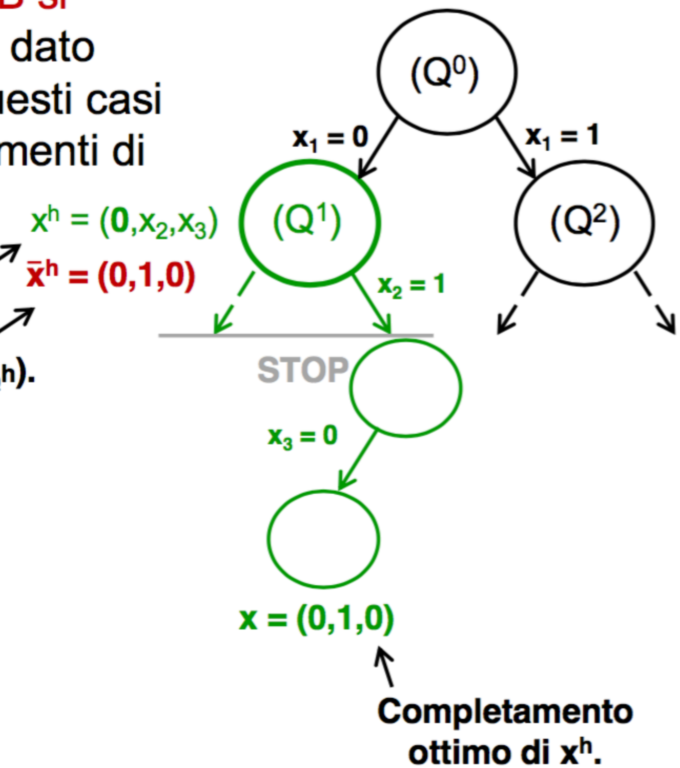
2 **[SOLVING]**

$\bar{x}^h$ è a componenti intere

➡ $\bar{x}^h$ è il **completamento ottimo** di x^h

Terminazione

Nei casi 1, 2, 3, la visita dell'albero del B&B si arresta al nodo h (**conclusioni implicite**), dato che – per motivi diversi – in ciascuno di questi casi è possibile trarre conclusioni sui completamenti di x^h rispetto al problema originale (Q).



2 **[SOLVING]**

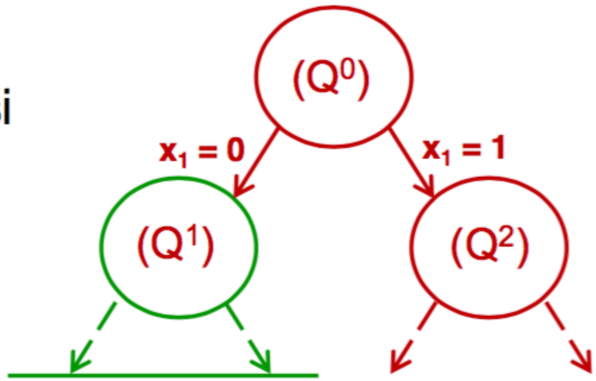
$\bar{x}^h$ è a componenti intere

→ $\bar{x}^h$ è il **completamento ottimo** di x^h

NOTA Se x^h fornisce un valore della f.o. migliore di quello fornito da x^*, allora x^h diventerà il nuovo valore x^*. Siccome è soluzione ottima del **sottoproblema** (Q^1), **non possiamo concludere che è ottima per (Q)**, <u>ma solo memorizzarla come **candidata** ad essere ottima</u>.

Terminazione

Nei casi 1, 2, 3, la visita dell'albero del B&B si arresta al nodo h (**conclusioni implicite**), dato che – per motivi diversi – in ciascuno di questi casi è possibile trarre conclusioni sui completamenti di x^h rispetto al problema originale (Q).



STOP:
La soluzione ottima di (Q^h) e quella di ogni sottoproblema intero di (Q^h) sono non migliori di x^*.

3 **[BOUNDING]**

➡ **Ogni completamento di x^h è non migliore di x^*.**

Terminazione

NOTA 1

In tutti i casi 1, 2, 3, il nodo h si dice **TERMINATO** (**SONDATO**/**FATHOMED**), PER INFEASIBILITY, PER SOLVING, o PER BOUNDING, rispettivamente.

NOTA 2

Se il nodo h è una foglia, tutte le componenti del vettore x sono state necessariamente fissate a un valore intero 0/1 e, dunque un nodo foglia è **naturalmente TERMINATO** (sempre "per solving") appena viene raggiunto (sono state fissate tutte le variabili e non è possibile effettuare ulteriori operazioni di branching).

NOTA 3

Dopo ogni operazione di branching al nodo h, il numero di componenti fissate nella soluzione parziale x^h aumenta esattamente di 1.

Terminazione

L'algoritmo B&B si arresta quando tutti i nodi sono stati terminati (in un numero finito di passi).

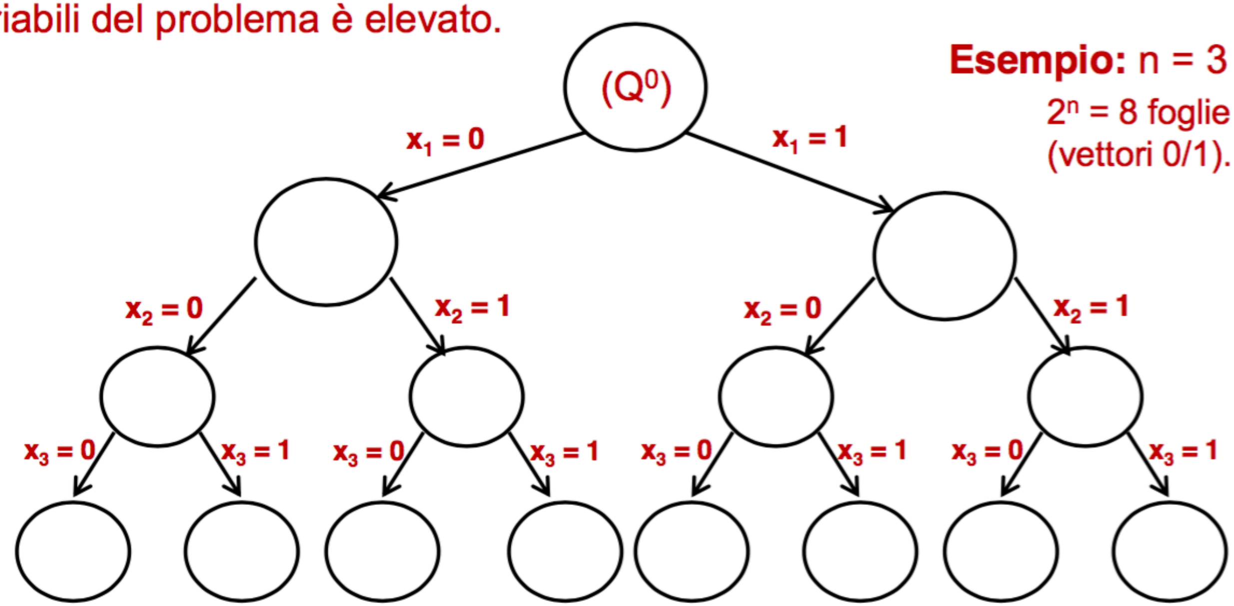
Nel caso peggiore ciò avviene solo quando tutte le foglie dell'albero di ricerca vengono raggiunte. In questo caso solo il numero di operazioni di branching è pari a 2^n-1. Per ciascuna è poi necessario risolvere i problemi associati ai nuovi nodi generati.

L'algoritmo B&B risolve (Q) in maniera **esatta**, ma ha complessità di calcolo **eccessiva** quando il numero n delle variabili del problema è elevato.

Esempio: n = 3

2^n = 8 foglie (vettori 0/1).

Albero di ricerca binario completo →



Algoritmo B&B: pseudocodice

BRANCH&BOUND PER PLI 0/1 (ENUMERAZIONE IMPLICITA)

Input: Problema (Q) di PLI (di massimo) a variabili Booleane: max $\{c^T x, x \in X\}$
con $P = \{x \in R^n, Ax = b, x \geq 0\}$ e $X = P \cap \{0,1\}^n$

Output: $x^* \in X$, soluzione ottima di (Q)

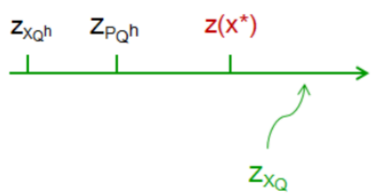
INIZIO
- $h := 0$, $z^* = -\infty$
- $A = \{x^0\}$ (insieme delle soluzioni parziali - nodi attivi dell'albero del B&B)
- FINCHÉ $A \neq \emptyset$
- INIZIO
  - **Selezionare una x^h in A**
  - Sia (Q^h) il problema (intero) ristretto a $X^h = \{x \in X : \underbrace{x \cap x^h = x^h}\}$
  - Sia (P_Q^h) il rilassamento continuo di (Q^h)
  - Risolvere (P_Q^h)
  - SE (P_Q^h) è non ammissibile
  - INIZIO
    - Eliminare x^h da A [INFEASIBILITY]
  - FINE
  - SE per la soluzione ottima $\bar{x}^h$ di (P_Q^h) si ha $z^* \geq z(\bar{x}^h)$
  - INIZIO
    - Eliminare x^h da A [BOUNDING]
  - FINE

Scelta del prossimo nodo h (soluzione parziale x^h) da analizzare

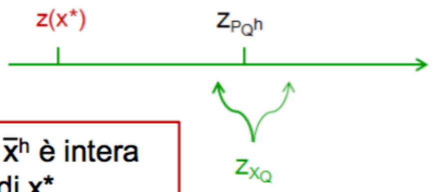
Completamenti (incogniti) di x^h

Risoluzione di un problema di PL

Decisioni implicite: si elimina l'intero sottoalbero radicato in h



Algoritmo B&B: pseudocodice



SE per la soluzione ottima $\bar{x}^h$ di (P_Q^h) si ha $\boxed{z^* < z(\bar{x}^h)}$

INIZIO

    SE $\bar{x}^h$ è a **componenti intere**

    INIZIO

        $x^* = \bar{x}^h$

        $z^* = z(\bar{x}^h)$

        Eliminare x^h da A [SOLVING]

    FINE

    ALTRIMENTI [BRANCHING]

    INIZIO

        Fissare una ulteriore componente in x^h

        Inserire in A le due nuove soluzioni parziali così generate

        Eliminare x^h da A [BRANCHING]

    FINE

FINE

FINE

SE $z^* > -\infty$

INIZIO

    Stampare x^* e z^*

FINE

FINE

Nota: in questo caso $\bar{x}^h$ è intera e certamente meglio di x*

Il branching al nodo h si esegue se $\bar{x}^h$ è frazionaria e $z^* < z(\bar{x}^h)$.

Scelta della componente incognita da fissare nel branching al nodo h.

Se A= ∅ e z* non è mai stato aggiornato → (Q) è non ammissibile

Algoritmo B&B: scelte

NOTA 1

Lo pseudocodice visto descrive i passi fondamentali della tecnica del Branch&Bound.

NOTA 2

Nello pseudocodice ci sono due punti relativi a scelte che possono essere implementate in modi diversi dando luogo a versioni diverse dell'algoritmo.

1) Scelta della componente incognita da fissare nel branching al nodo h

Nel caso di variabili 0/1, quando al nodo h si deve eseguire una operazione di branching è necessario **decidere quale componente x_j** ancora incognita della soluzione parziale x^h deve essere fissata (risp. a 0 e a 1 nei due sottoproblemi generati).

Generalmente si fissa una delle componenti che nella soluzione ottima di P_{Q^h} assume valore frazionario, tendenzialmente quella con valore più vicino ad un intero.

Esempio

Se $x^h = (x_1=0,\ x_2=1,\ x_3,\ x_4,\ x_5)$

e $\overline{x}^h = (x_1=0,\ x_2=1,\ x_3=0,\ x_4=0.4,\ \mathbf{x_5=0.9})$

→ **si fissa x_5**

Algoritmo B&B: scelte

NOTA 1

Lo pseudocodice visto descrive i passi fondamentali della tecnica del Branch&Bound.

NOTA 2

Nello pseudocodice ci sono due punti relativi a scelte che possono essere implementate in modi diversi dando luogo a versioni diverse dell'algoritmo.

2) Scelta del prossimo nodo h (soluzione parziale x^h) da analizzare

Alla generica iterazione del B&B esistono più nodi dell'albero di ricerca che sono stati generati attraverso una precedente operzione di Branching e che attendono di essere analizzati – sono tutti i nodi non terminati (o nodi attivi).

Si possono seguire diverse **tecniche di esplorazione** dell'albero di ricerca del del B&B. Le più utilizzate sono le seguenti:

Depth-first search (**Breadth-first** search)

Best-first search

Depth-forward/best-back search

Altre tecniche di raffinamento del B&B

È possibile implementare alcune tecniche per raffinare ulteriormente la ricerca delle soluzioni di (Q) con il B&B e migliorare la performance dell'algoritmo.

L'obiettivo è **potenziare le capacità del B&B di terminare precocemente** i nodi dell'albero di ricerca.

Arrotondamento

Prima di effettuare un branching al nodo h si "prova" ad arrotondare (o troncare) la soluzione del problema rilassato ristretto (P_{Q^h}) senza perdere l'ammissibilità della soluzione rispetto ai vincoli di (P_{Q^h}).

NOTA

Non è detto che si ottenga sempre una soluzione ammissibile per (Q) e, comunque, non si ha alcuna garanzia di ottimalità.

Trattandosi, però, di una **operazione poco dispendiosa, è utile tentare** per cercare di individuare quante più possibili soluzioni a componenti intere.

Altre tecniche di raffinamento del B&B

È possibile implementare alcune tecniche per raffinare ulteriormente la ricerca delle soluzioni di (Q) con il B&B e migliorare la performance dell'algoritmo.

L'obiettivo è **potenziare le capacità del B&B di terminare precocemente** i nodi dell'albero di ricerca.

Eliminazione "per bound" dei padri

Ogni volta che si aggiorna la migliore soluzione corrente x* è possibile terminare tutti i nodi attivi h che hanno nodo padre p(h) tale che:

- $z(P_Q^{p(h)}) < z(x^*)$ (nel caso di max)
- $z(P_Q^{p(h)}) > z(x^*)$ (nel caso di min)

Infatti in questi casi ogni completamento della soluzione parziale $x^{p(h)}$ è peggiore di x*. Posso eliminare i nodi-figlio di p(h) dalla lista dei nodi attivi.

NOTA

Si tratta di una terminazione del tutto analoga a quella "per bounding", ma eseguita sul padre di ogni nodo h attivo ("parent bound termination") alla luce dell'aggiornamento di x*.

Risoluzione di un problema di Knapsack binario con il B&B

Esempio

Problema di knapsack binario

(Q^0)

$$\max 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^0})

$$\max 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in \mathbf{[0,1]}$$

Iterazione 1: Nodo $h=0$: problemi (Q^0) e (P_{Q^0})

Soluzione ottima di (P_{Q^0}):

$\bar{x}^0 = (x_1=1, x_2=1, \mathbf{x_3=2/5}, x_4=0)$

$z(\bar{x}^0) = 21.4$

x_3 è frazionaria

valutazione (u.b.)

NOTA

Il valore ottimo di (Q^0) è certamente intero, perciò si può considerare il bound (più stretto) seguente (valutazione):

$$z(P_{Q^0}) \geq \lfloor z(P_{Q^0}) \rfloor \geq z(Q^0)$$

Esempio

Problema di knapsack binario

$$(Q^0) \quad \max\ 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

$$(P_{Q^0}) \quad \max\ 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

Iterazione 1: Nodo $h=0$: problemi (Q^0) e (P_{Q^0}) $\boxed{v_0 = 21}$

Soluzione ottima di (P_{Q^0}):

$\bar{x}^0 = (x_1=1,\ x_2=1,\ \mathbf{x_3=2/5},\ x_4=0)$

x_3 è frazionaria

$z(\bar{x}^0) = 21.4$

$\lfloor z(\bar{x}^0) \rfloor = \mathbf{21}$ ← **valutazione** (u.b.)

NOTA

Il valore ottimo di (Q^0) è certamente intero, perciò si può considerare il bound (più stretto) seguente (valutazione):

$$z(P_{Q^0}) \geq \lfloor z(P_{Q^0}) \rfloor \geq z(Q^0)$$

Esempio

Problema di knapsack binario

$$(Q^0)\quad \begin{aligned} \max\ & 10x_1 + 9x_2 + 6x_3 + 4x_4 \\ & 6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15 \\ & x_1, x_2, x_3, x_4 \in \{0,1\} \end{aligned}$$

Rilassamento continuo

$$(P_{Q^0})\quad \begin{aligned} \max\ & 10x_1 + 9x_2 + 6x_3 + 4x_4 \\ & 6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15 \\ & x_1, x_2, x_3, x_4 \in [0,1] \end{aligned}$$

Iterazione 1: Nodo $h=0$: problemi (Q^0) e (P_{Q^0}) $\quad \boxed{v_0 = \mathbf{21}}$ $\quad \boxed{z(x^*) = \mathbf{19}}$

Soluzione ottima di (P_{Q^0}):

$\overline{x}^0 = (x_1=1,\ x_2=1,\ \mathbf{x_3=2/5},\ x_4=0)$

$z(\overline{x}^0) = 21.4$ $\quad$ x_3 è frazionaria

$\lfloor z(\overline{x}^0) \rfloor = \mathbf{21}$ $\longleftarrow$ **valutazione** (u.b.)

In questo caso è possibile ottenere una soluzione iniziale per (Q^0) se **troncando** la soluzione $\overline{x}^0$ si ottiene una soluzione **ammissibile**.

Si ottiene così **la** (prima) **migliore soluzione intera corrente**

$$x^* = (x_1=1,\ x_2=1,\ x_3=0,\ x_4=0) \text{ con } \mathbf{z(x^*) = 19}.$$

Esempio

Problema di knapsack binario

(Q^0)

$$\max\ 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^0})

$$\max\ 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

Meglio di così NO Peggio di così NO

Iterazione 1: Nodo $h=0$: problemi (Q^0) e (P_{Q^0}) $v_0 = \mathbf{21}$ $z(x^*) = \mathbf{19}$

Si ha: $\mathbf{21} = \lfloor z(\bar{x}^h) \rfloor > z(x^*) = \mathbf{19}$

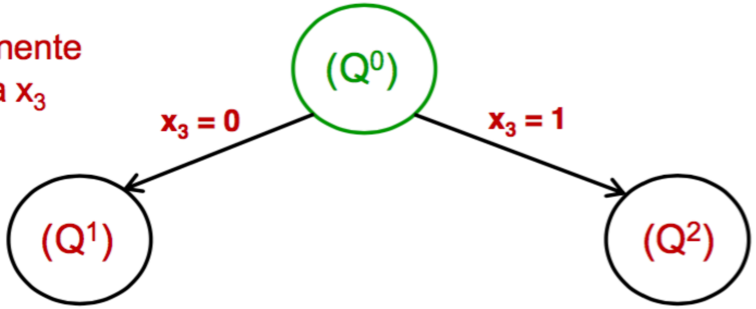
$\bar{x}^0 = (x_1=1,\ x_2=1,\ \mathbf{x_3=2/5},\ x_4=0)$

Valutazione al nodo 0 (u.b. per l'ottimo)

migliore soluzione corrente ammissibile intera

Branching

sulla componente frazionaria x_3



NOTA
I nodi "**attivi**" associati ai problemi (Q^1) e (Q^2) vengono memorizzati in una lista per poi essere estratti secondo l'ordine di esplorazione dell'albero del B&B.

Esempio

Problema di knapsack binario

(Q^0)

$$\max\ 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^0})

$$\max\ 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

Meglio di così NO Peggio di così NO

Iterazione 1: Nodo $h=0$: problemi (Q^0) e (P_{Q^0}) $\boxed{v_0 = 21}$ $\boxed{z(x^*) = 19}$

Si ha: $21 = \lfloor z(\bar{x}^h) \rfloor > z(x^*) = 19$ $x^0 = (x_1=1,\ x_2=1,\ x_3=2/5,\ x_4=0)$

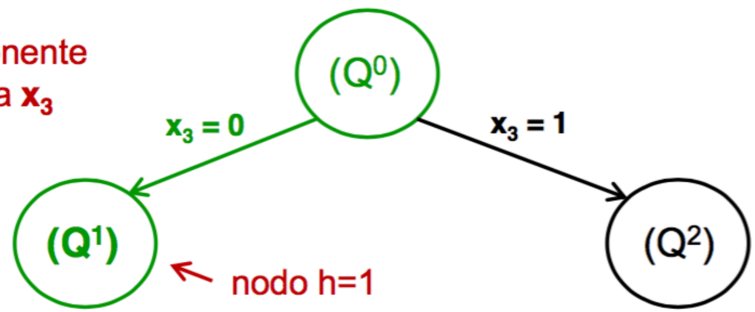
valutazione (u.b.)

migliore soluzione corrente ammissibile intera

Branching

sulla componente frazionaria x_3

Si valuta uno dei nodi attivi, ad esempio **h=1.**



Esempio

Problema di knapsack binario

(Q^1)

$$\max 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^1})

$$\max 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

$\mathbf{x_3 = 0}$

Iterazione 2: Nodo $h=1$: problemi (Q^1) e (P_{Q^1})

$v_1 = \mathbf{21}$

$z(x^*) = \mathbf{19}$

Soluzione ottima di (P_{Q^1}):

$\bar{x}^1 = (x_1=1,\ x_2=1,\ \mathbf{x_3=0},\ \mathbf{x_4=1/2})$

$z(\bar{x}^1) = \mathbf{21}$

valutazione (u.b.)

NOTA

La variabile x_4 prima era intera e ora è frazionaria.

Ciò può accadere per le variabili che rimangono non fissate (incognite).

Troncando tale soluzione si ottiene la soluzione ammissibile per (Q):

$$x = (x_1=1,\ x_2=1,\ x_3=0,\ x_4=0) \text{ con } z(x) = 19$$

che è proprio la **migliore soluzione intera corrente x*** già individuata precedentemente.

Esempio

Problema di knapsack binario

(Q^1)

$$\max\ 10x_1 + 9x_2 + 6x_3 + 4x_4$$

$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^1})

$$\max\ 10x_1 + 9x_2 + 6x_3 + 4x_4$$

$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

$\mathbf{x_3 = 0}$

Iterazione 2: Nodo $h=1$: problemi (Q^1) e (P_{Q^1}) $\boxed{v_1 = \mathbf{21}}$ $\boxed{z(x^*) = \mathbf{19}}$

Soluzione ottima di (P_{Q^1}): $\bar{x}^1 = (x_1=1,\ x_2=1,\ \mathbf{x_3=0},\ \mathbf{x_4=1/2})$

$z(\bar{x}^1) = \mathbf{21}$

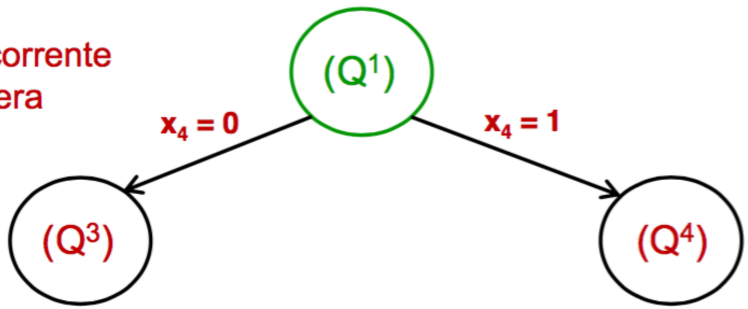
Si ha: $\mathbf{21} = \lfloor z(\bar{x}^h) \rfloor > z(x^*) = \mathbf{19}$

valutazione (u.b.)

migliore soluzione corrente ammissibile intera

branching

sulla componente frazionaria $\mathbf{x_4}$

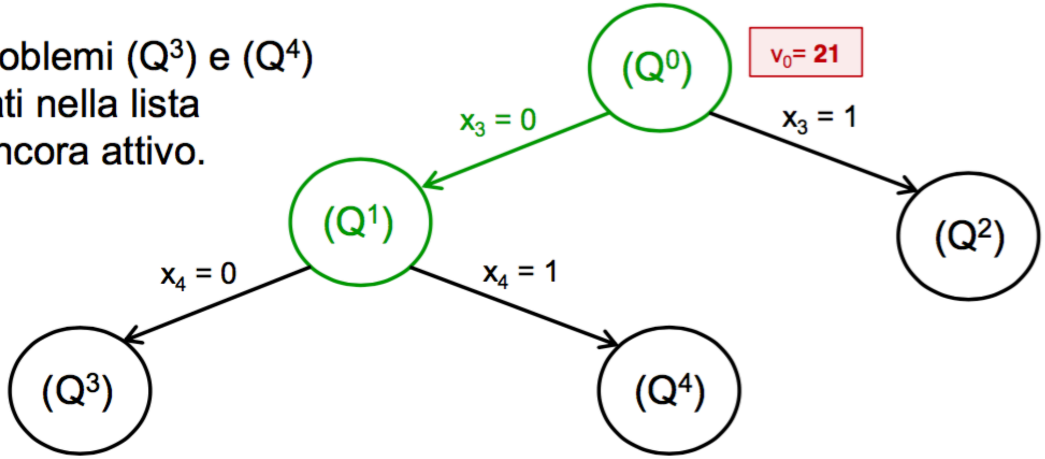


Esempio

Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

NOTA
I nodi associati ai problemi (Q^3) e (Q^4) vengono memorizzati nella lista insieme al nodo 2 ancora attivo.



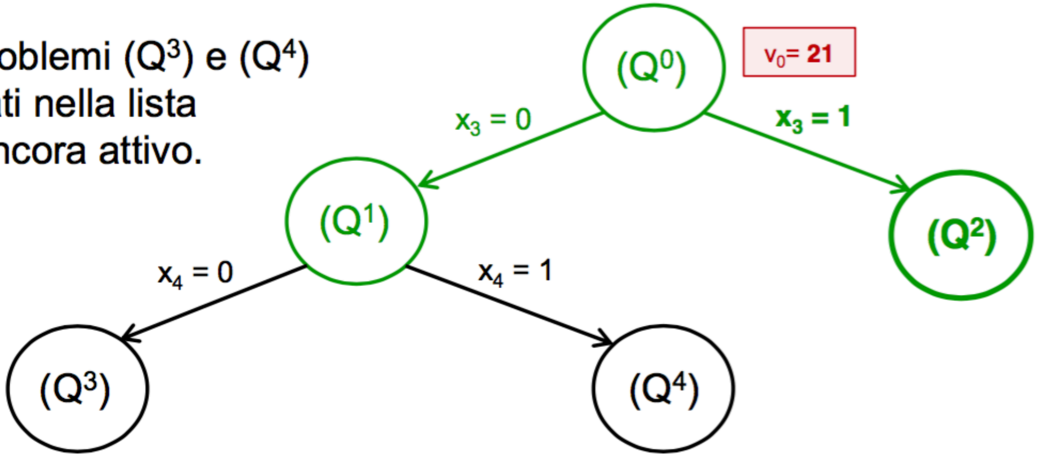
z(x*) = 19

Esempio

Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

NOTA
I nodi associati ai problemi (Q^3) e (Q^4) vengono memorizzati nella lista insieme al nodo 2 ancora attivo.



Si valuta uno dei nodi "attivi", ad esempio il nodo **h=2**.

Esempio

Problema di knapsack binario

(Q^2)

$$\max\ 10x_1 + 9x_2 + \mathbf{6} + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq \mathbf{10}$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^2})

$$\max\ 10x_1 + 9x_2 + \mathbf{6} + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq \mathbf{10}$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

$\mathbf{x_3 = 1}$

Iterazione 3: Nodo $h=2$: problemi (Q^2) e (P_{Q^2}) — $v_2 = \mathbf{21}$ — $z(x^*) = \mathbf{19}$

Soluzione ottima di (P_{Q^2}):

$\bar{x}^2 = (x_1=1,\ \mathbf{x_2=4/7},\ \mathbf{x_3=1},\ x_4=0)$

$z(\bar{x}^2) = 21.143$ — x_2 è frazionaria

$\lfloor z(\bar{x}^2) \rfloor = \mathbf{21}$ ← valutazione (u.b.)

Troncando tale soluzione si ottiene la soluzione ammissibile per (Q^2):

$x = (x_1=1,\ x_2=0,\ x_3=1,\ x_4=0)$ con $\mathbf{z(x) = 16} < 19$

che **non è migliore** della x^* corrente.

Esempio

Problema di knapsack binario | **Rilassamento continuo**

(Q^2)

$$\max 10x_1 + 9x_2 + \mathbf{6} + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \le \mathbf{10}$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

(P_{Q^2})

$$\max 10x_1 + 9x_2 + \mathbf{6} + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \le \mathbf{10}$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

$\mathbf{x_3 = 1}$

Iterazione 3: Nodo h=2: problemi (Q^2) e (P_{Q^2}) $\boxed{v_2 = \mathbf{21}}$ $\boxed{z(x^*) = \mathbf{19}}$

Soluzione ottima di (P_{Q^2}): $\bar{x}^2 = (x_1=1, \mathbf{x_2=4/7}, \mathbf{x_3=1}, x_4=0)$

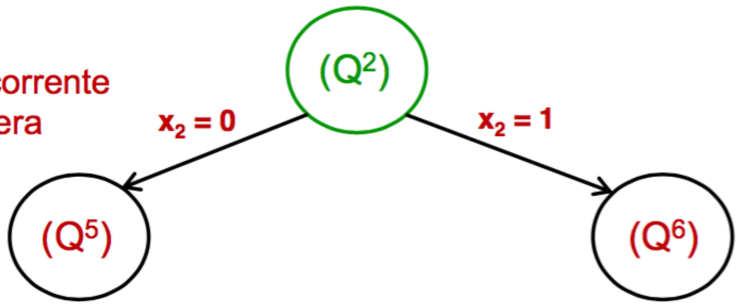
$z(\bar{x}^2) = 21.143$ x_2 è frazionaria

$\lfloor z(\bar{x}^2) \rfloor = \mathbf{21}$ ← valutazione (u.b.)

Si ha: $\mathbf{21} = \lfloor z(\bar{x}^h) \rfloor \mathbf{>} z(x^*) = \mathbf{19}$

valutazione (u.b.)

migliore soluzione corrente ammissibile intera



branching

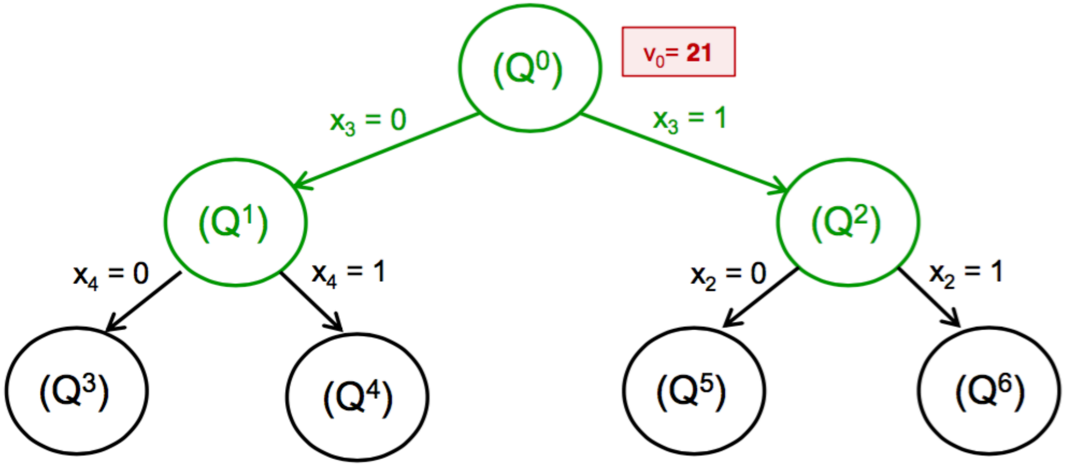
sulla componente frazionaria $\mathbf{x_2}$

Esempio

Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2



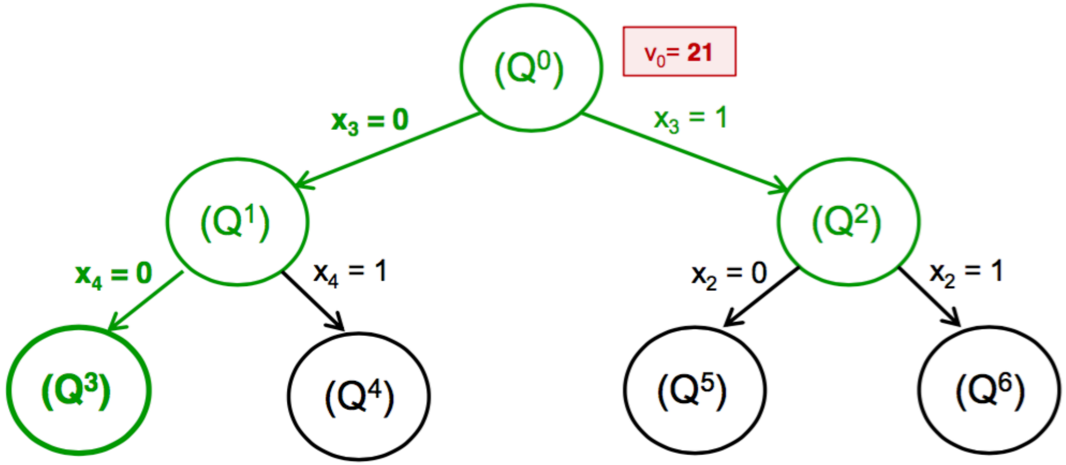
$z(x^*) = 19$

Esempio

Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2



- Dei quattro nodi “attivi” analizziamo ora il nodo **h=3**.

Esempio

Problema di knapsack binario

(Q^3)

$$\max 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^3})

$$\max 10x_1 + 9x_2 + 6x_3 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 15$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

$\mathbf{x_3 = 0}$

$\mathbf{x_4 = 0}$

Iterazione 4: Nodo h=3: problemi (Q^3) e (P_{Q^3}) $\boxed{v_3 = 19}$ $\boxed{z(x^*) = 19}$

Soluzione ottima di (P_{Q^3}): $\bar{x}^3 = (x_1=1, x_2=1, x_3=0, x_4=0)$

$z(\bar{x}^3) = 19$

La soluzione ottima di $(P_Q{}^3)$ è a **componenti intere**.

Si ha: $19 = v_3 = z(\bar{x}^3) = 19$

valutazione (u.b.)

migliore soluzione corrente ammissibile intera

Il nodo h=3 è sondato per ottimalità **[solving]**

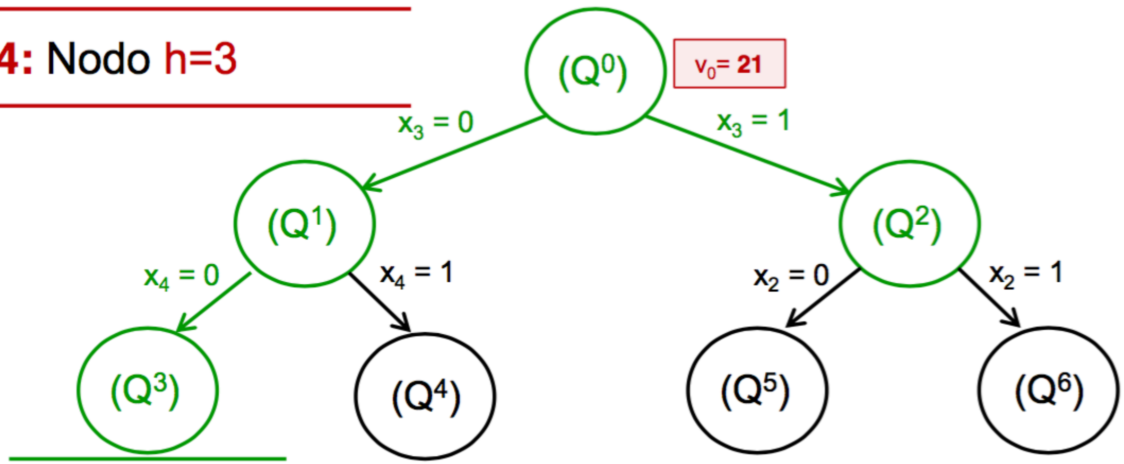
Esempio

Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2

Iterazione 4: Nodo h=3



[solving]:

$\bar{x}^3 = (x_1=1, x_2=1, \mathbf{x_3=0}, \mathbf{x_4=0})$

$\mathbf{z(\bar{x}^3) = 19}$

$\boxed{\mathbf{z(x^*) = 19}}$

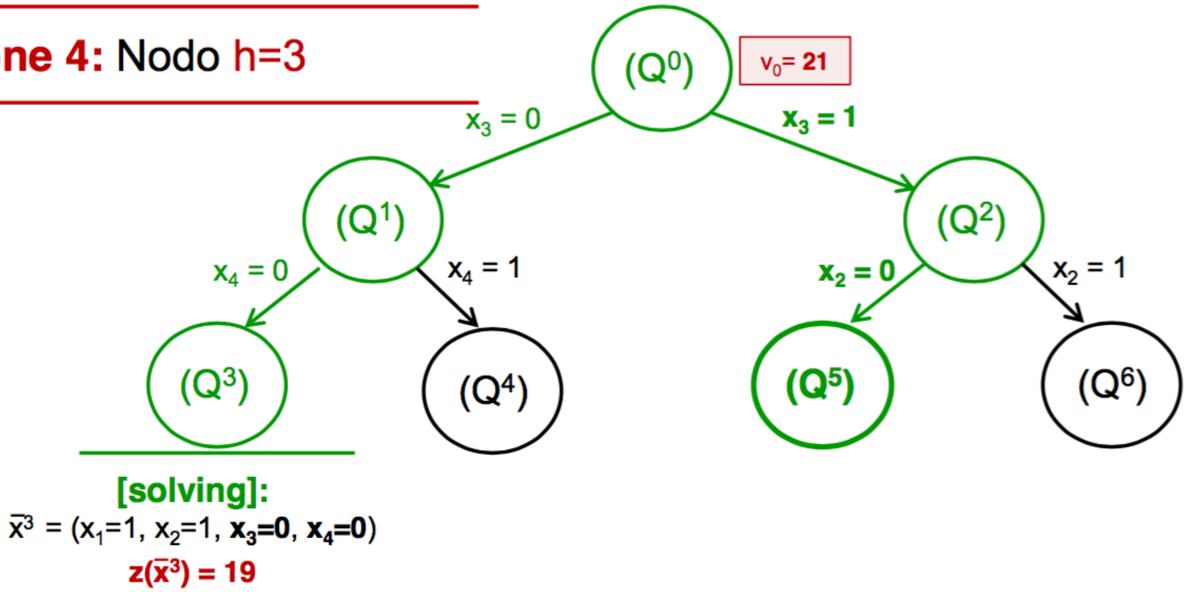
Esempio

Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2

Iterazione 4: Nodo h=3



Dei tre nodi "attivi" analizziamo ora il nodo **h=5**.

Esempio

Problema di knapsack binario | **Rilassamento continuo**

(Q^5)

$$\max 10x_1 + 9x_2 + 6 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \le 10$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

$$\max 10x_1 + 9x_2 + 6 + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \le 10$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

(P_{Q^5})

$x_3 = 1$

$x_2 = 0$

Iterazione 6: Nodo h=5: problemi (Q^5) e (P_{Q^5}) $\boxed{v_5 = 20}$ $\boxed{z(x^*) = 20}$

Soluzione ottima di (P_{Q^5}): $\overline{x}^5 = (x_1=1, x_2=0, x_3=1, x_4=1)$

$z(\overline{x}^5) = 20$

La soluzione ottima di $(P_Q{}^5)$ è a **componenti intere**.

Si ha: $20 = v_5 = z(\overline{x}^5) = 20$

valutazione (u.b.)

migliore soluzione corrente ammissibile intera

Il nodo h=5 è sondato per ottimalità **[solving]**

NOTA
La soluzione $\overline{x}^5$ **è migliore** della soluzione x^*.

Aggiornamento: $x^* \leftarrow \overline{x}^5$

Esempio

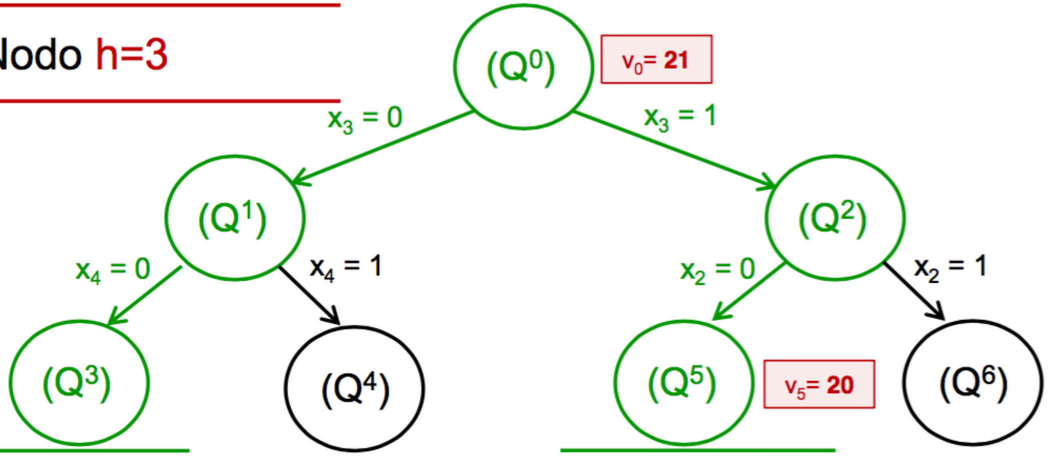
Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2

Iterazione 4: Nodo h=3

Iterazione 5: Nodo h=5



[solving]:

$\bar{x}^3 = (x_1=1, x_2=1, \mathbf{x_3=0}, \mathbf{x_4=0})$

$z(\bar{x}^3) = 19$

[solving]:

$\bar{x}^5 = (x_1=1, \mathbf{x_2=0}, \mathbf{x_3=1}, x_4=1)$

$z(\bar{x}^5) = 20$

$\boxed{z(x^*) = 20}$

Esempio

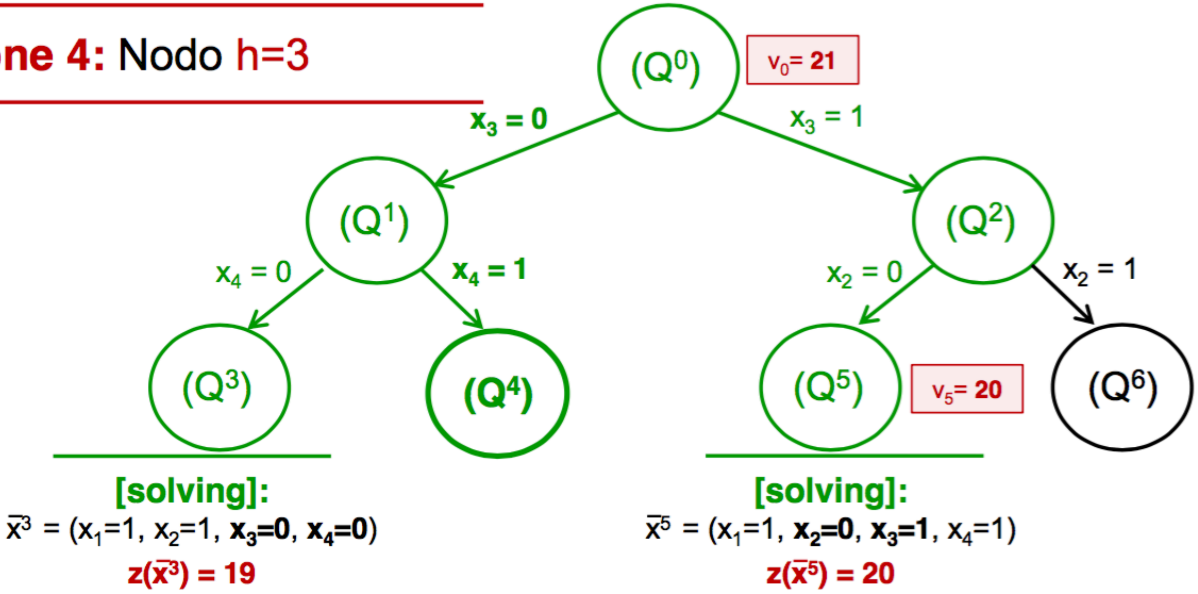
Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2

Iterazione 4: Nodo h=3

Iterazione 5: Nodo h=5



Dei due nodi “attivi” analizziamo ora il nodo **h=4**.

Esempio

Problema di knapsack binario

(Q^4)

$$\max\ 10x_1 + 9x_2 + 6x_3 + 4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 11$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^4})

$$\max\ 10x_1 + 9x_2 + 6x_3 + 4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq 11$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

$\mathbf{x_3 = 0}$
$\mathbf{x_4 = 1}$

Iterazione 6: Nodo h=4: problemi (Q^4) e (P_{Q^4}) $\quad v_4 = 20 \quad z(x^*) = 20$

Soluzione ottima di $(P_Q{}^4)$: $\bar{x}^4 = (x_1=1,\ x_2=5/7,\ x_3=0,\ x_4=1)$

$z(\bar{x}^4) = 20.428$ $\quad$ x_2 è frazionaria

$\Rightarrow \lfloor z(\bar{x}^4) \rfloor = 20$ ← valutazione (u.b.)

Si ha: $20 = \lfloor z(\bar{x}^h) \rfloor = z(x^*) = 20$

L'upper bound identificato non è utile perché si conosce già la soluzione x* che è ammissibile per (Q) e **ha già valore della f.o. pari a 20**.

⇒ Il nodo h=4 è sondato per bound **[bounding]**

Esempio

Iterazione 1: Nodo h=0

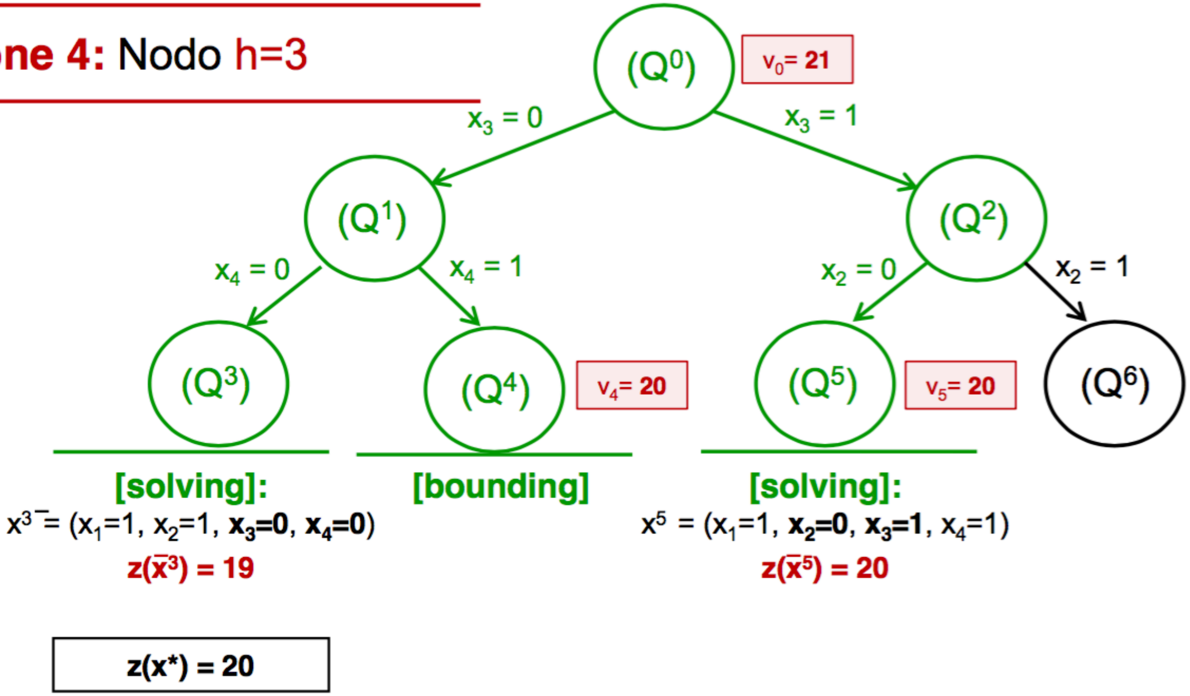
Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2

Iterazione 4: Nodo h=3

Iterazione 5: Nodo h=5

Iterazione 6: Nodo h=4



Esempio

Iterazione 1: Nodo h=0

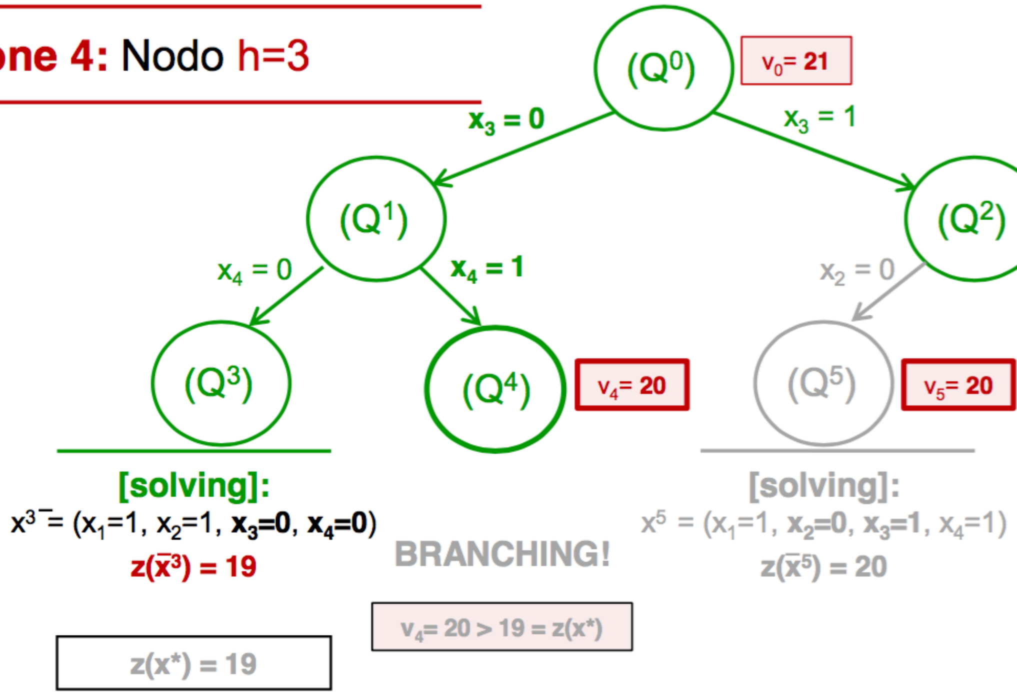
Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2

Iterazione 4: Nodo h=3

Iterazione 5: Nodo h=5

Iterazione 6: Nodo h=4



Se avessimo visitato il nodo h=4 prima del nodo h=5, per il nodo h=4 avremmo avuto un **branching** invece che un **bounding**.

Esempio

Iterazione 1: Nodo h=0

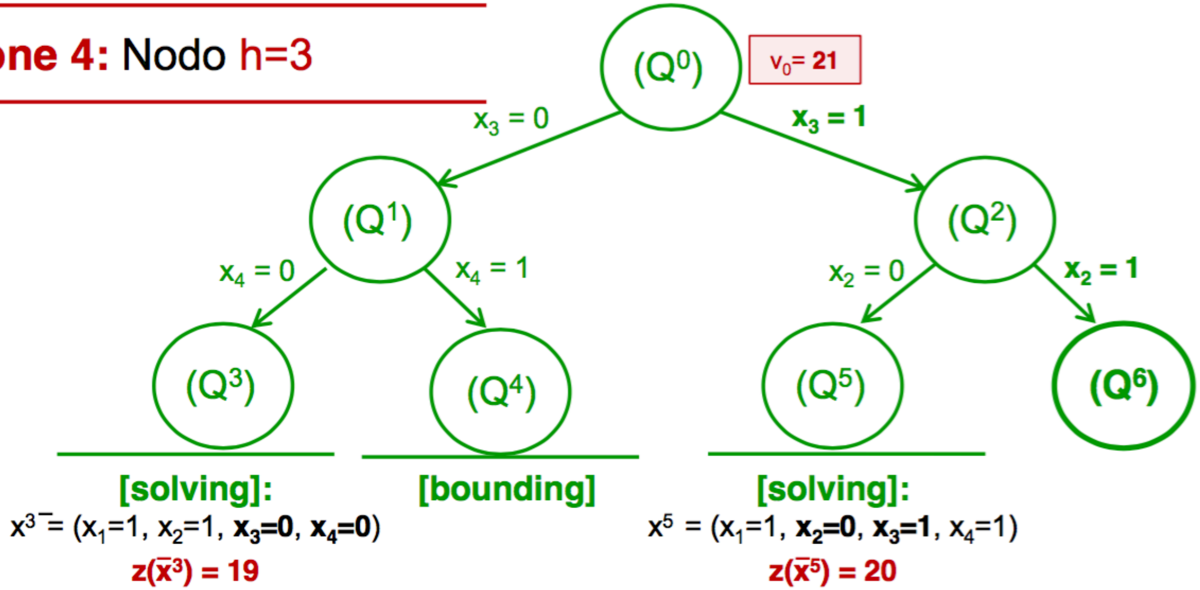
Iterazione 2: Nodo h=1

Iterazione 3: Nodo h=2

Iterazione 4: Nodo h=3

Iterazione 5: Nodo h=4

Iterazione 6: Nodo h=4



- Rimane da analizzare il solo nodo attivo **h=6**.

Esempio

Problema di knapsack binario

(Q^0)

$$\max 10x_1 + \mathbf{9} + \mathbf{6} + 4x_4$$

$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq \mathbf{3}$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

Rilassamento continuo

(P_{Q^0})

$$\max 10x_1 + \mathbf{9} + \mathbf{6} + 4x_4$$

$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq \mathbf{3}$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

$\mathbf{x_3 = 1}$

$\mathbf{x_2 = 1}$

Iterazione 7: Nodo $h=6$: problemi (Q^6) e (P_{Q^6})

Esempio

Problema di knapsack binario | **Rilassamento continuo**

(Q^0)

$$\max 10x_1 + \mathbf{15} + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq \mathbf{3}$$
$$x_1, x_2, x_3, x_4 \in \{0,1\}$$

(P_{Q^0})

$$\max 10x_1 + \mathbf{15} + 4x_4$$
$$6x_1 + 7x_2 + 5x_3 + 4x_4 \leq \mathbf{3}$$
$$x_1, x_2, x_3, x_4 \in [0,1]$$

$\mathbf{x_3 = 1}$

$\mathbf{x_2 = 1}$

Iterazione 7: Nodo h=6: problemi (Q^6) e (P_{Q^6}) $\boxed{v_6 = \mathbf{20}}$ $\boxed{z(x^*) = \mathbf{20}}$

Soluzione ottima di (P_{Q^6}): $\bar{x}^6 = (\mathbf{x_1=1/2}, \mathbf{x_2=1}, \mathbf{x_3=1}, x_4=0)$

x_1 è frazionaria

$z(\bar{x}^6) = \mathbf{20}$

valutazione (u.b.)

Si ha: $\mathbf{20} = \lfloor z(\bar{x}^h) \rfloor = z(x^*) = \mathbf{20}$

L'upper bound identificato non è utile perchè si conosce già la soluzione x* che è ammissibile per (Q) e **ha già valore della f.o. pari a 20**.

Il nodo h=6 è sondato per bound **[bounding]**

Esempio

Iterazione 1: Nodo h=0

Iterazione 2: Nodo h=1

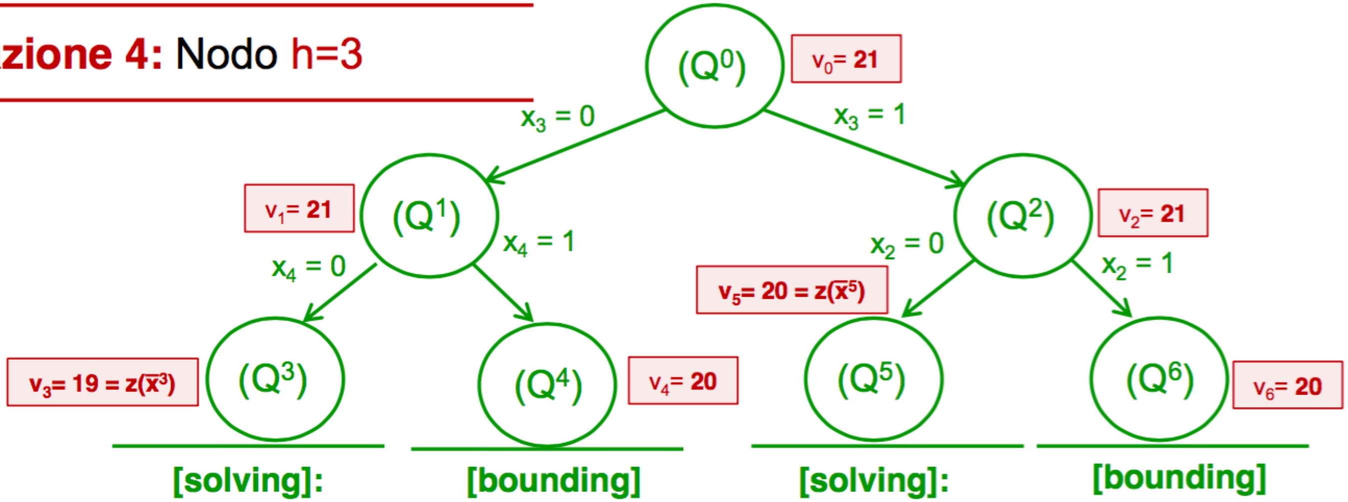
Iterazione 3: Nodo h=2

Iterazione 4: Nodo h=3

Iterazione 5: Nodo h=5

Iterazione 6: Nodo h=4

Iterazione 7: Nodo h=6



$\bar{x}^3 = (x_1{=}1,\ x_2{=}1,\ \mathbf{x_3{=}0},\ \mathbf{x_4{=}0})$

$z(\bar{x}^3) = 19$

$\bar{x}^5 = (x_1{=}1,\ \mathbf{x_2{=}0},\ \mathbf{x_3{=}1},\ x_4{=}1)$

$z(\bar{x}^5) = 20$

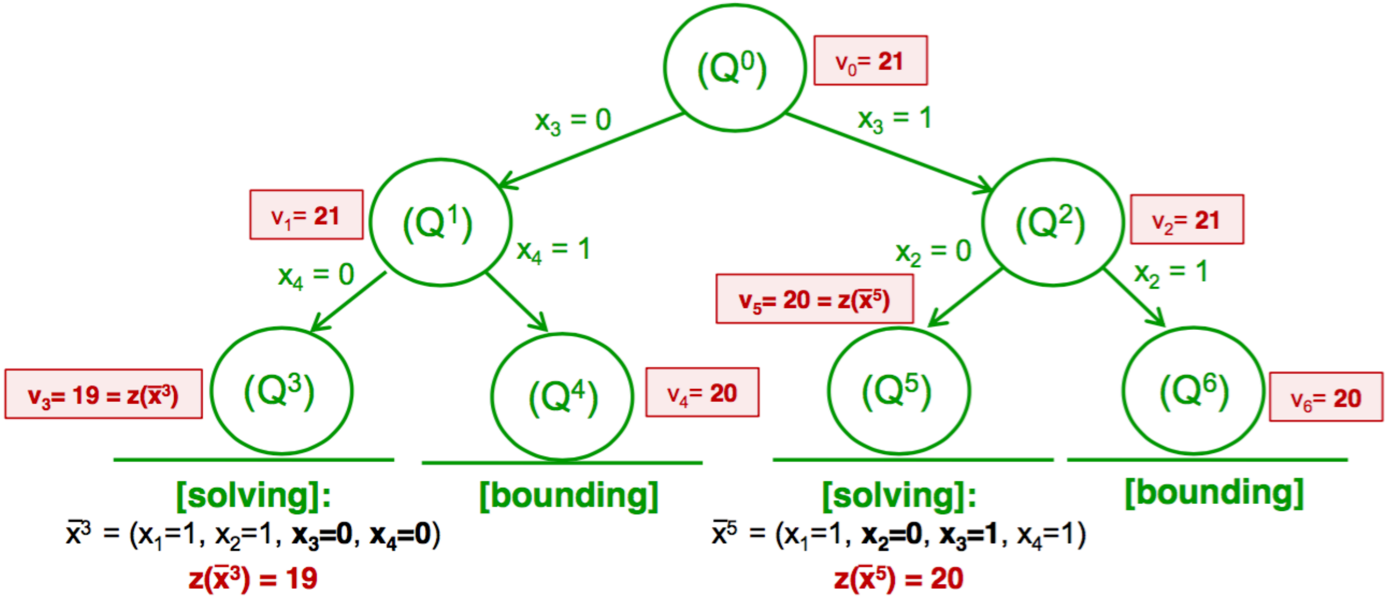
Non ci sono più nodi attivi da valutare

Il B&B termina e la soluzione ottima è $x^* = (x_1{=}1,\ x_2{=}0,\ x_3{=}1,\ x_4{=}1)$ con **$z(x^*) = 20$**.

Esempio

NOTA

Con la tecnica di enumerazione implicita abbiamo valutato (enumerato) in maniera diretta **4 soluzioni parziali**, invece delle **$2^n = 2^4 = 16$ soluzioni complete** (foglie dell'albero) che avremmo dovuto valutare se avessimo adottato la tecnica di enumerzione totale.



Non ci sono più nodi attivi da valutare ➡ Il B&B termina e la soluzione ottima è $x^* = (x_1=1, x_2=0, x_3=1, x_4=1)$ con $\mathbf{z(x^*) = 20}$.

Esempio

NOTA

Con la tecnica di enumerazione implicita abbiamo valutato (enumerato) in maniera diretta **4 soluzioni parziali**, invece delle **$2^n = 2^4 = 16$ soluzioni complete** (foglie dell'albero) che avremmo dovuto valutare se avessimo adottato la tecnica di enumerzione totale.