# Programmazione lineare intera

## Ricerca Operativa

Giovanni Righini

Università degli Studi di Milano

# Ottimizzazione nel discreto

Esistono diverse classi di problemi di ottimizzazione con variabili discrete:

- IP: integer programming
- BP: binary programming
- MIP: mixed-integer programming
- CO: combinatorial optimization

Considereremo solo modelli lineari: Integer Linear Programming (ILP).

Per ottimizzare nel discreto possiamo:

- selezionare “buone” *formulazioni lineari* e migliorarle fino a poterle risolvere un problema di PLI come problema di PL;
- *scomporre* il problema in sotto-problemi più piccoli e più facili;
- eseguire una *enumerazione implicita* delle soluzioni.

# Ottimalità

Dato un problema di ottimizzazione discreta $P$

$$z^* = \max\{z(x) : x \in X \subseteq \mathcal{Z}^n\}$$

l'ottimalità si dimostra calcolando un *upper bound* $\overline{z}$ e un *lower bound* $\underline{z}$, tali che

$$\underline{z} \leq z^* \leq \overline{z}.$$

- Se $P$ è un problema di minimizzazione, $\overline{z}$ è un bound priale e $\underline{z}$ è un bound duale.
- Se $P$ è un problema di massimizzazione, $\underline{z}$ è un bound primale e $\overline{z}$ è un bound duale.

La differenza $\overline{z} - \underline{z}$ è detta gap di ottimalità.
Quando $\overline{z} - \underline{z} = 0$ si ha la *garanzia di ottimalità*.

# Bounds primali

Un bound primale $\overline{z}$ è dato dal valore della funzione obiettivo $z(x)$ in una qualsiasi soluzione ammissibile $\overline{x} \in X$.

$$\overline{z} = z(\overline{x}), \quad \overline{x} \in X.$$

Bounds primali possono essere calcolati in vari modi:

- con algoritmi euristici o meta-euristici (ricerca locale, GRASP,...);
- con algoritmi di approssimazione con garanzia: in tal caso si ha anche un bound duale.

Per alcuni problemi di ottimizzazione discreta è difficile anche trovare una soluzione ammissibile (cioè calcolare un bound primale).

# Bounds duali

Un bound duale è dato dal valore della funzione obiettivo $z(x)$ in corrispondenza di una soluzione super-ottima $\overline{x}$. Quindi in generale $\overline{x}$ non è ammissibile.

Ci sono due tecniche principali per calcolare un bound duale per un problema $P$:

- risolvere all'ottimo un rilassamento $R$ di $P$;
- trovare una soluzione ammissibile al duale $D$ di $P$.

# Rilassamenti

Dato un problema

$$P = \min\{z_P(x) : x \in X(P)\}$$

un problema

$$R = \min\{z_R(x) : x \in X(R)\}$$

è un rilassamento di $P$ se valgono le seguenti due condizioni:

- $X(P) \subseteq X(R)$
- $z_R(x) \leq z_P(x) \;\; \forall x \in X(P)$.

[In caso di massimizzazione, le disequazioni vanno invertite.]

**Corollario:** $z_R^* \leq z_P^*$.

Ci sono molti tipi diversi di rilassamento.
Un rilassamento è tanto migliore quando più il suo valore ottimo $z_R^*$ è vicino a $z_P^*$.

# Rilassamento lineare continuo

Quando $P$ è un problema di ottimizzazione discreta

$$P)\ \min\{z(x) : x \in X, x \in \mathcal{Z}_+^n\},$$

il suo *rilassamento continuo* $C$ è ottenuto da $P$ trascurando le condizioni di integralità:

$$C)\ \min\{z(x) : x \in X, x \in \Re_+^n\}.$$

Quando $P$ è un problema di ottimizzazione *lineare* discreta

$$P)\ \min\{cx : Ax \leq b, x \in \mathcal{Z}_+^n\},$$

il suo rilassamento continuo $LP$

$$LP)\ \min\{cx : Ax \leq b, x \in \Re_+^n\}$$

è un problema di programmazione lineare (che sappiamo come risolvere molto efficacemente).

Se $x_{LP}^* \in \mathcal{Z}_+^n$, allora $x_P^* = x_{LP}^*$.

# Rilassamenti combinatori

Il rilassamento combinatorio $C$ di un problema di ottimizzazione combinatoria $P$ è ancora un problema di ottimizzazione combinatoria, ma tipicamente molto più facile da risolvere.

Esempio 1:

- $P$: il TSP asimmetrico;
- $C$: il problema di matching bipartito di costo minimo.

Esempio 2:

- $P$: il TSP simmetrico;
- $C$: il problema dell’1-albero ricoprente di costo minimo.

# Rilassamento Lagrangeano

Il rilassamento Lagrangeano *LR* di un problema di ottimizzazione (lineare discreto) *P* si ottiene rimuovendo alcuni vincoli e aggiungendo all'obiettivo termini di penalità per la loro violazione.

$$P) \quad \min\{z(x) : Ax \leq b, x \in X \subseteq \mathcal{Z}_+^n\}$$

$$LR) \quad \min\{z_{LR}(x,\lambda) = z(x) + \lambda(Ax - b) : x \in X \subseteq \mathcal{Z}_+^n\}$$

con $\lambda \geq 0$.

Esso soddisfa entrambe le condizioni per essere un rilassamento:

- Vincoli: $\{x : Ax \leq b, x \in X\} \subseteq \{x : x \in X\}$
- Obiettivo:
  - $Ax - b \leq 0$ per tutte le soluzioni ammissibili per $P$;
  - $\lambda(Ax - b) \leq 0$ per tutte le soluzioni ammissibili per $P$;
  - $z_{LR}(x,\lambda) = z(x) + \lambda(Ax - b) \leq z(x)$ per tutte le soluzioni ammissibili per $P$.

# Rilassamento surrogato

Il rilassamento surrogato $S$ di un problema di ottimizzazione (lineare discreto) $P$ si ottiene sostituendo un insieme di vincoli con una loro combinazione convessa.

$$P) \quad \min\{z(x) : Ax \leq b, x \in X \subseteq \mathcal{Z}^n_+\}$$

$$S) \quad \min\{z(x) : \lambda^T Ax \leq \lambda^T b, x \in X \subseteq \mathcal{Z}^n_+\}$$

con $\lambda \geq 0$.

Esso soddisfa le due condizioni per essere un rilassamento:

- Vincoli: $Ax \leq b$ implica $\lambda^T Ax \leq \lambda^T b$ (ma non viceversa).
- Obiettivo: banale, perché non cambia.

# Rilassamenti e bounds

I rilassamenti lineare, Lagrangeano e surrogato possono fornire in generale bounds diversi.

In caso di minimizzazione valgono le seguenti relazioni:

$$z_{LP}^* \leq z_{LR}^* \leq z_S^* \leq z^*.$$

Con $z_{LR}^*$ e $z_S^*$ qui si indicano i migliori bounds ottenibili dal rilassamento Lagrangeano e surrogato, scegliendo cioè nel modo migliore i moltiplicatori $\lambda$.

# Dualità

La seconda tecnica per ottenere un bound duale consiste nel calcolare una soluzione ammissibile per il problema duale di $P$ o per il duale di un suo rilassamento.

Problema lineare duale:

$$P)z^* = \min\{cx : Ax \geq b, x \in \mathcal{Z}^n_+\}$$

$$D)w^* = \max\{yb : yA \leq c, y \in \Re^m_+\}$$

formano una coppia primale-duale debole.

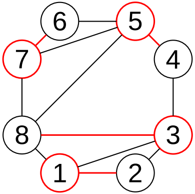
Problema duale combinatorio:
Il problema del massimo matching e il problem del *minimum vertex cover*

$$P)z^* = \max\{1x : Ax \leq 1, x \in \mathcal{B}^{|E|}_+\}$$

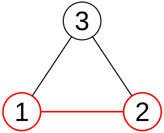
$$D)w^* = \min\{1y : yA \geq 1, y \in \mathcal{B}^{|V|}_+\}$$

dove $A$ è la matrice di incidenza di un grafo $G = (V, E)$, formano una coppia primale-duale debole.

# Esempio




$$z^* = 4 \quad w^* = 4$$




$$z^* = 1$$

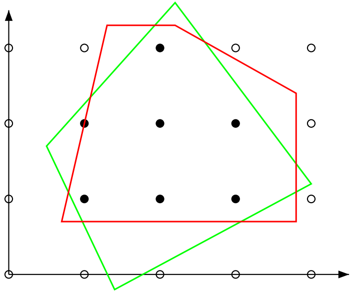
$$z^*_{LP} = z([\frac{1}{2}, \frac{1}{2}, \frac{1}{2}]) = \frac{3}{2}$$

$$w^*_{LP} = w([\frac{1}{2}, \frac{1}{2}, \frac{1}{2}]) = \frac{3}{2}$$

$$w^* = 2$$

# Formulazioni lineari

I problemi di ottimizzazione (lineare) discreti *non* hanno una formulazione unica.



# Formulazioni

Dal momento che non sono uniche, ha senso

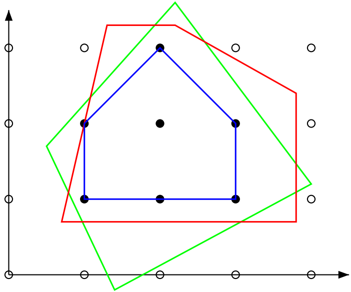
- *confrontare* formulazioni,
- *migliorare* formulazioni.

Una formulazione migliore si traduce in un algoritmo più efficiente.

La formulazione ideale di un problema di programmazione lineare discreta è quella che consente di risolverlo come se fosse un problema di programmazione lineare nel continuo.

# Formulazione ideale

La formulazione di un problema di programmazione lineare corresponde ad un *poliedro*.



I vincoli della formulazione ideale corrispondono al *guscio convesso* delle soluzioni intere.

# Guscio convesso (*convex hull*)

Dato un insieme discreto

$$X = \{x_1, \dots, x_t\} \text{ with } x_i \in \Re^n \ \forall i = i, \dots, t,$$

il suo *guscio convesso* è il poliedro

$$conv(X) = \{x \in \Re^n : x = \sum_{i=1}^{t} \lambda_i x_i, \sum_{i=1}^{t} \lambda_i = 1, \lambda_i \geq 0 \ \forall i = 1, \dots, t\}.$$

E' un *poliedro* i cui punti estremi sono elementi dell'insieme discreto $X$.

Data una formulazione $P$ e l'insieme discreto $X$ delle sue soluzioni ammissibili, vale la relazione

$$X \subseteq conv(X) \subseteq P.$$

# Polyhedral combinatorics

In generale

- non conosciamo la formulazione ideale dei problemi di ottimizzazione lineare intera;
- il numero dei vincoli del guscio convesso può crescere esponenzialmente con la dimensione dell'istanza.

Conosciamo la formulazione ideale solo per alcuni particolari problemi di ottimizzazione discreta: il problema del cammino minimo su grafo, il problema del matching bipartito di costo minimo, il problema dell'albero ricoprente di costo minimo,...

La disciplina che studia come selezionare e migliorare le formulazioni lineari dei problemi di PLI è la *polyhedral combinatorics*.

# Scelta della formulazione: esempio

In molti problemi di ottimizzazione discreta con vincoli di capacità (Bin Packing Problem, Facility Location Problem,...), ci sono vincoli di questa forma:

$$\sum_{i \in \mathcal{N}} x_{ij} \leq |\mathcal{N}| y_j \quad \forall j \in \mathcal{M}, \tag{1}$$

che esprime una condizione logica che lega le variabili $x$ e $y$:

$$\begin{cases} \exists (i,j) \in \mathcal{N} \times \mathcal{M} : x_{ij} > 0 & \Rightarrow y_j = 1 \\ \exists j \in \mathcal{M} : y_j = 0 & \Rightarrow x_{ij} = 0 \;\; \forall i \in \mathcal{N}. \end{cases}$$

La stessa condizione può essere espressa con

$$x_{ij} \leq y_j \quad \forall i \in \mathcal{N}, \forall j \in \mathcal{M}. \tag{2}$$

La formulazione (1) richiede $|\mathcal{M}|$ vincoli.
La formulazione (2) richeide $|\mathcal{M}||\mathcal{N}|$ vincoli.

# Scelta della formulazione: esempio

Sommando tra loro i vincoli (2) per ogni $i \in \mathcal{N}$ si ottiene

$$\sum_{i \in \mathcal{N}} x_{ij} \leq \sum_{i \in \mathcal{N}} y_j \quad \forall j \in \mathcal{M}$$

cioè proprio i vincoli (1): $\sum_{i \in \mathcal{N}} x_{ij} \leq |\mathcal{N}| y_j \quad \forall j \in \mathcal{M}$.

Quindi ogni vincolo (1) è un vincolo *surrogato* di alcuni vincoli (2).
I vincoli (2) implicano i vincoli (1) ma non viceversa.
Ci sono soluzioni che soddisfano (1) ma violano (2):

$$\begin{cases} x_{ij} = 1 & \forall j \in \mathcal{M}, \forall i \in \mathcal{N} : i \in [k(j-1)+1, \ldots, kj] \\ y_j = 1/|\mathcal{M}| & \forall j \in \mathcal{M} \end{cases}$$

dove $k = |\mathcal{N}|/|\mathcal{M}|$.

I vincoli (2) generano una migliore formulazione rispetto ai vincoli (1).
Il poliedro con i vincoli (2) contiene il poliedro con i vincoli (1).

# Algoritmi “cutting planes”

Dato un problema di PLI

$$P^{(k)} = \max\{cx : Ax \leq b, x \in \mathcal{Z}^n_+\}$$

consideriamo il suo rilassamento continuo

$$L^{(k)} = \max\{cx : Ax \leq b, x \in \Re^n_+\}$$

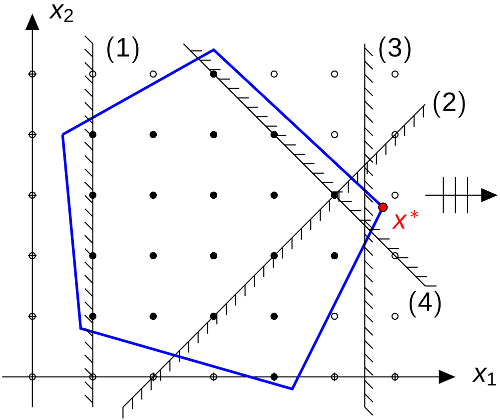
e la sua soluzione ottima $x^{*(k)}$. Quindi, generiamo un insieme di *disuguaglianze valide* $Qx \leq q$ tali che.

- $Qx \leq q \;\; \forall x \in \mathcal{Z}^n_+ : Ax \leq b$
- $Qx^{*(k)} > q$

e otteniamo così una formulazione più stretta

$$P^{(k+1)} = \max\{cx : Ax \leq b, Qx \leq q, x \in \mathcal{Z}^n_+\}.$$

# Disuguaglianze valide: esempio




La disequazione (1) è vaida ma inutile: non ”taglia” $x^*$.
La disequazione (2) non è valida: ”taglia” alcune soluzioni ammissibili intere.
La disequazione (3) è valida e utile.
La disequazione (4) è anche *facet defining*.

# Procedura di Chvátal-Gomory

Consideriamo un problema di PLI con insieme ammissibile

$$X = \{x \in \mathcal{Z}^n_+ : Ax \leq b\}$$

dove $A$ ha $m$ righe e $n$ colonne.

Scegliamo un vettore $u \in \Re^m_+$:

- $\sum_{j=1}^n u a_j x_j \leq ub$ è valida perché $ax \leq b$ e $u \geq 0$.
- $\sum_{j=1}^n \lfloor u a_j \rfloor x_j \leq ub$ è valida perché $x \geq 0$.
- $\sum_{j=1}^n \lfloor u a_j \rfloor x_j \leq \lfloor ub \rfloor$ è valida perché $x$ è intero.

Ogni disuguaglianza valida può essere generata con questa procedura in un numero finito di passi.

L'efficacia della procedura dipende dalla scelta di $u$.

# Algoritmi “cutting planes”

Gli algoritmi “cutting planes” iterativamente risolvono il rilassamento continuo $L$ di un problema discreto $P$ e rafforzano la sua formulazione generando ulteriori vincoli (*cutting planes*), in modo tale che la soluzione ottima del rilassamento continuo all’iterazione $k$ diventi inammissibile all’iterazione $k+1$.

- Pro:
  - se i piani di taglio sono generati in modo efficace, l’algoritmo può garantire di trovare la soluzione ottima discreta senza fare ricorso ad altre tecniche (ad es. enumerazione implicita);
  - una formulazione più stretta, anche se non ideale, può fornire *bounds duali* più efficaci in un algoritmo branch-and-bound.
- Contro:
  - è necessaria una procedura apposita per generare iterativamente disuguaglianze valide e utili: è chiamata *algoritmo di separazione*. Se il problema originale è difficile ($NP$-hard), anche il problema di separazione lo è.

# Algoritmi “cutting planes”: pseudo-codice

**Begin**
$t$:=0; $P^{(0)}$:=$P$; [$P$ è il rilassamento continuo]
**repeat**
  $z^{*(t)}$:=$\max\{cx : x \in P^{(t)}\}$
  $x^{*(t)}$:=argmax$\{cx : x \in P^{(t)}\}$
  **if** $x^{*(t)} \notin \mathcal{Z}^n$ **then**
    Genera una disuguaglianza valida $\pi x \leq \pi_0 : \pi x^{*(t)} > \pi_0$
    $P^{(t+1)}$:=$P^{(t)} \cap \{x : \pi x \leq \pi_0\}$
    $t := t + 1$
  **end if**
**until** $(x^{*(t)} \in \mathcal{Z}^n) \vee$ (no inequalities found)
**End**

# Algoritmi “cutting planes”

Dopo ogni iterazione $z^{*(t)}$ è un bound duale valido.

Può capitare che non venga trovata nessuna disuguaglianza valida se l’algoritmo di separazione è ristretto a cercarla all’interno di specifici sottinsiemi di disuguaglianze con una struttura particolare, che non bastano per descrivere completamente il guscio convesso del problema di PLI.

# Tagli di Gomory

Data una soluzione frazionaria $x^*$ del rilassamento continuo di un problema di PLI, si utilizza la procedura di Chvátal-Gomory sul vincolo associato ad una variabile frazionaria: si ottiene così una disuguaglianza valida violata da $x^*$.

Dato un problema di PLI

$$P) \;\; \max\{cx : ax = b, x \geq 0, x \in \mathcal{Z}^n\}$$

ed il suo rilassamento continuo

$$LP) \;\; \max\{cx : ax = b, x \geq 0\}$$

siano $x^*$ e $z^*$ la soluzione ottima di $LP$ e il suo valore.

$$z^* = \overline{a}_{00} + \sum_{j \in N^*} \overline{a}_{0j} x_j^*$$

$$\begin{cases} x^*_{B^*i} + \sum_{j \in N^*} \overline{a}_{ij} x_j^* = \overline{a}_{i0} & \forall i = 1, \ldots, m \\ x^* \geq 0 \end{cases} \tag{3}$$

dove $B^*$ e $N^*$ sono gli indici delle variabili in base e fuori base in $x^*$.

# Tagli di Gomory

Se $x^*$ non è intero, esiste almeno un vincolo $\hat{\imath}$ tale che $\overline{a}_{\hat{\imath}0}$ non è intero.

Eseguendo la procedura di Chvátal-Gomory su di esso si ottiene:

$$x_{B*\hat{\imath}} + \sum_{j \in N*} \lfloor \overline{a}_{\hat{\imath}j} \rfloor x_j \leq \lfloor \overline{a}_{\hat{\imath}0} \rfloor .$$

Sottraendo questa disuguaglianza dal vincolo di uguaglianza

$$x^*_{B*\hat{\imath}} + \sum_{j \in N*} \overline{a}_{\hat{\imath}j} x^*_j = \overline{a}_{\hat{\imath}0}$$

si ottiene il taglio di Gomory:

$$\sum_{j \in N*} f_{\hat{\imath}j} x_j \geq f_{\hat{\imath}0}$$

dove $f_{\hat{\imath}j} = \overline{a}_{\hat{\imath}j} - \lfloor \overline{a}_{\hat{\imath}j} \rfloor$ e $f_{\hat{\imath}0} = \overline{a}_{\hat{\imath}0} - \lfloor \overline{a}_{\hat{\imath}0} \rfloor$.

Anche la variabile di slack/surplus associata a questa disuguaglianza è intera.

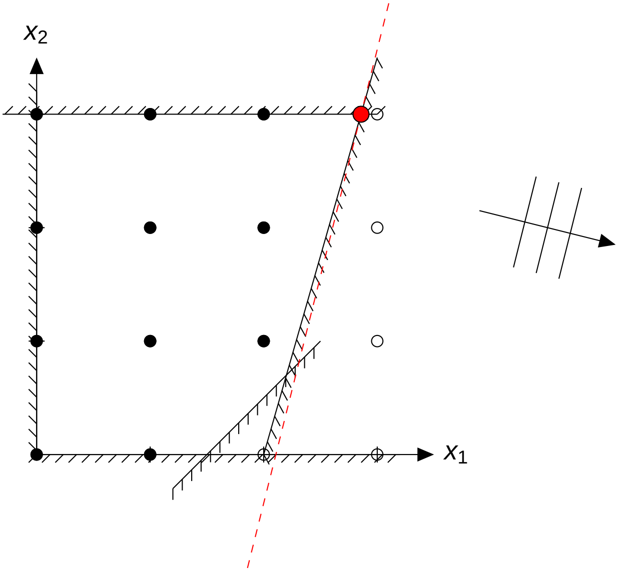
# Un esempio

$$
\begin{aligned}
\text{maximize } z = 4x_1 - x_2 \\
7x_1 - 2x_2 &\leq 14 \\
x_2 &\leq 3 \\
2x_1 - 2x_2 &\leq 3 \\
x &\geq 0 \text{ (integer)}
\end{aligned}
$$

Risolvendo il rilassamento continuo, si ottiene $B^* = \{1, 2, 5\}$, $N^* = \{3, 4\}$:

$$
\begin{aligned}
z = \frac{59}{7} \quad & - \frac{4}{7}x_3 - \frac{1}{7}x_4 \\
x_1 \quad & + \frac{1}{7}x_3 + \frac{2}{7}x_4 && = \frac{20}{7} \\
x_2 \quad & \phantom{+ \frac{1}{7}x_3} + x_4 && = 3 \\
& - \frac{2}{7}x_3 + \frac{10}{7}x_4 + x_5 && = \frac{23}{7} \\
x \geq 0
\end{aligned}
$$

# Un esempio



# Un esempio

$$
\begin{aligned}
z = \frac{59}{7} \quad & -\frac{4}{7}x_3 - \frac{1}{7}x_4 \\
x_1 \quad & +\frac{1}{7}x_3 + \frac{2}{7}x_4 = \frac{20}{7} \\
x_2 \quad & +x_4 = 3 \\
& -\frac{2}{7}x_3 + \frac{10}{7}x_4 + x_5 = \frac{23}{7} \\
x \geq 0 &
\end{aligned}
$$

Dal primo vincolo si genera un taglio di Gomory:

$$x_1^* = \frac{20}{7} \Rightarrow \frac{1}{7}x_3 + \frac{2}{7}x_4 \geq \frac{6}{7}.$$

La sua variabile ausiliaria è

$$s_1 = -\frac{6}{7} + \frac{1}{7}x_3 + \frac{2}{7}x_4.$$

# Un esempio

Dai vincoli

$$x_1 + \frac{1}{7}x_3 + \frac{2}{7}x_4 = \frac{20}{7}$$
$$x_2 + x_4 = 3$$

si ottiene

$$x_3 = -7x_1 + 2x_2 + 14$$
$$x_4 = -x_2 + 3$$

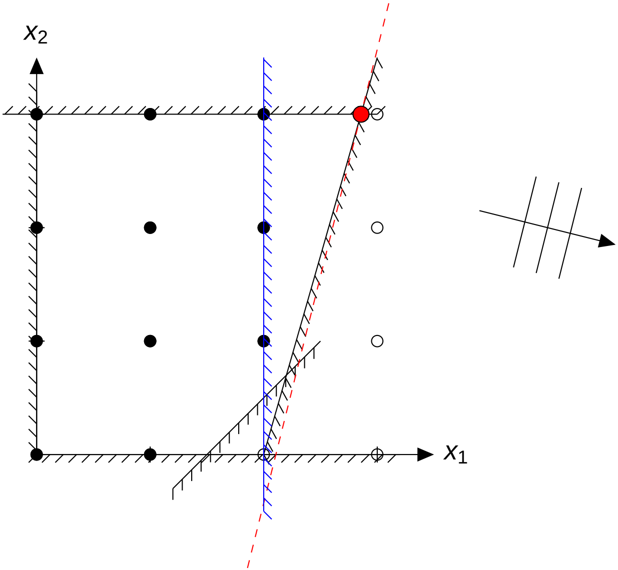
e l'equazione del taglio di Gomory

$$\frac{1}{7}x_3 + \frac{2}{7}x_4 \geq \frac{6}{7}$$

si può riscrivere come

$$x_1 \leq 2.$$

# Un esempio

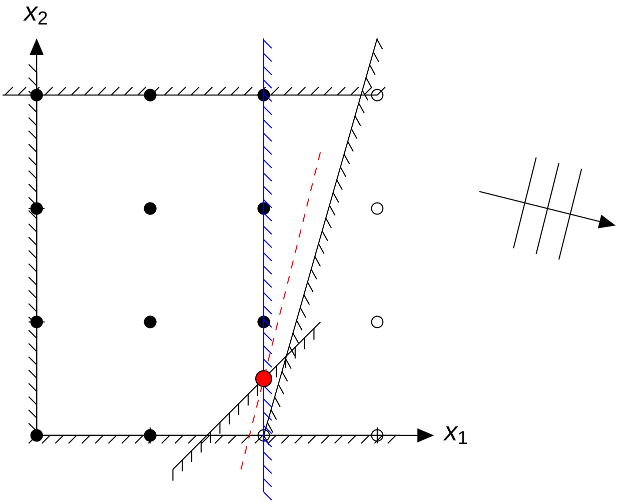


# Un esempio

Ri-ottimizzando si ottiene:

$$
\begin{aligned}
z = \frac{15}{2} \qquad\qquad & -\frac{1}{2}x_5 - 3s_1 \\
x_1 \qquad\qquad & \qquad\quad + s_1 = 2 \\
x_2 \qquad & -\frac{1}{2}x_5 + \; s_1 = \frac{1}{2} \\
x_3 \quad & -x_5 - 5s_1 = 1 \\
x_4 & + \frac{1}{2}x_5 - s_1 = \frac{5}{2} \\
x, s \geq 0 &
\end{aligned}
$$

# Un esempio



# Un esempio

$$\begin{aligned}
z = \frac{15}{2} \qquad\qquad & -\frac{1}{2}x_5 - 3s_1 \\
x_1 \qquad\qquad & \qquad\quad +s_1 = 2 \\
x_2 \qquad & -\frac{1}{2}x_5 + \ s_1 = \frac{1}{2} \\
x_3 \qquad & -x_5 - 5s_1 = 1 \\
x_4 & +\frac{1}{2}x_5 - s_1 = \frac{5}{2} \\
x, s \geq 0 &
\end{aligned}$$

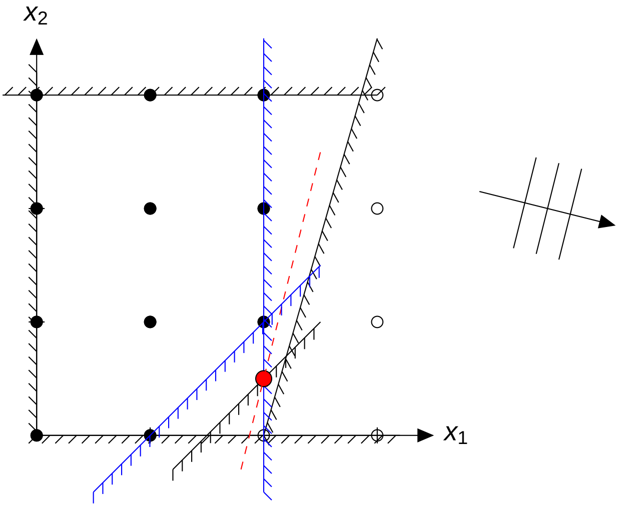
Dal secondo vincolo si può generare un taglio di Gomory:

$$x_2^* = \frac{1}{2} \quad \Rightarrow \quad \frac{1}{2}x_5 \geq \frac{1}{2} \quad \Rightarrow \quad x_1 - x_2 \leq 1.$$

La sua variabile ausiliaria è

$$s_2 = -\frac{1}{2} + \frac{1}{2}x_5.$$

# Un esempio



# Un esempio

Ri-ottimizzando ancora, si ottiene:

$$
\begin{aligned}
z = 7 \qquad\qquad\qquad\qquad & -3s_1 - s_2 \\
x_1 \qquad\qquad\qquad & +s_1 \qquad\quad = 2 \\
x_2 \qquad\qquad & + s_1 - s_2 = 1 \\
x_3 \qquad & -5s_1 - 2s_2 = 2 \\
x_4 & -s_1 + s_2 = 2 \\
x_5 & \qquad -2s_2 = 1 \\
x, s \geq 0 &
\end{aligned}
$$

Ora la soluzione ottima del rilassamento continuo è intera e quindi è anche la soluzione ottima discreta.

# Un esempio