Programmazione Lineare Intera: Algoritmo Cutting Plane

Daniele Vigo

D.E.I. – Università di Bologna

daniele.vigo@unibo.it

rev. 1.0 – 2023

Algoritmi generali per PLI

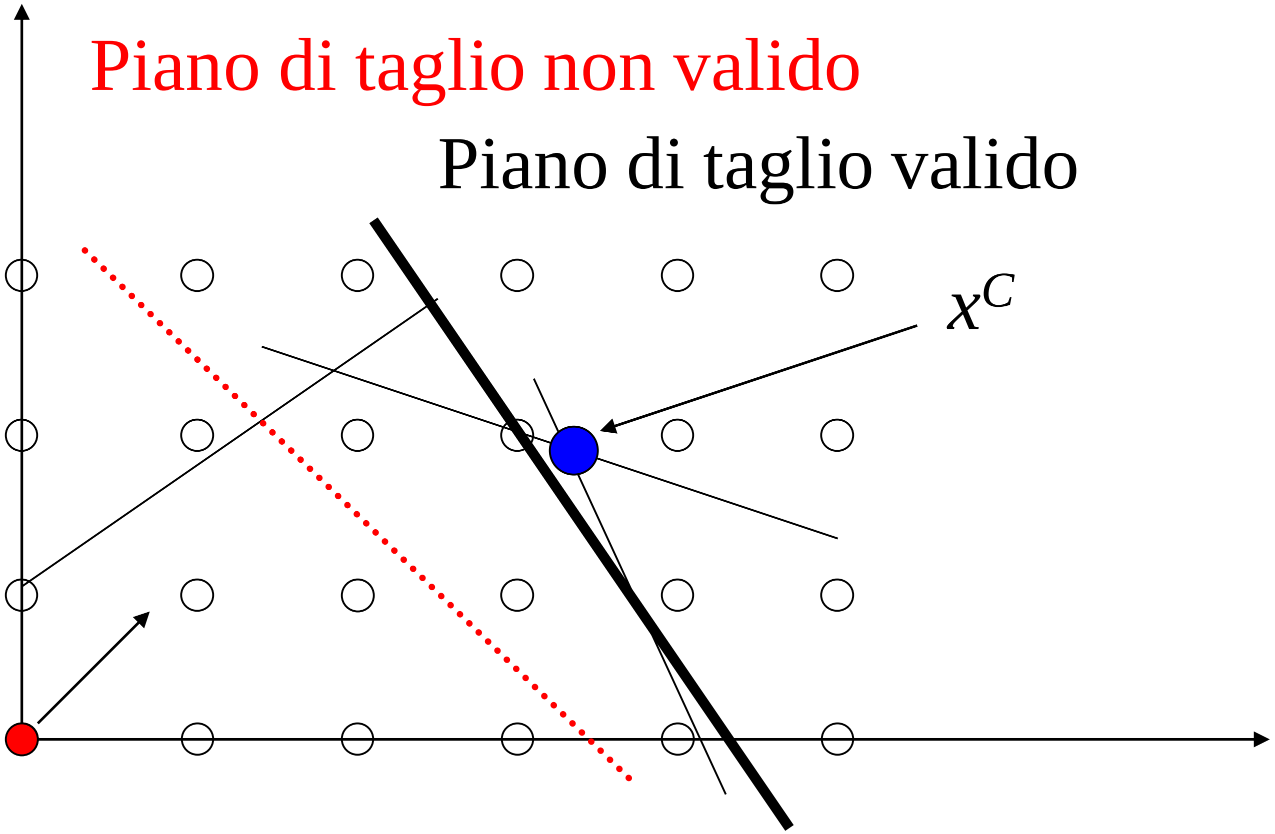
- Metodi esatti tradizionali (anni 60-oggi):
  - Metodo dei piani di taglio (cutting planes)
  - Branch-and-Bound
  - Programmazione Dinamica
- …
- Metodi esatti più avanzati (anni 90-oggi):
  - Branch-and-Bound + Cutting planes = Branch-and-Cut
  - Branch-and-Price/Column generation

Algoritmo Cutting Planes

- Sia P un problema di PLI
  - x^*: soluzione ottima (di valore z^*)
- $C(P)$ = rilassamento continuo di P
  - x^C: soluzione ottima di $C(P)$ (di valore z^C)
- Un iperpiano $\alpha x \geq \alpha_0$ si dice piano di taglio (cutting plane) se:

1. x^C non è più ammissibile ($\alpha x^C < \alpha_0$)
2. è valido per ogni soluzione intera del problema originale ($\alpha x \geq \alpha_0 \; \forall x$ ammissibile e intera)

Algoritmo Cutting Planes (2)



Algoritmo Cutting Planes (3)

Algoritmo *cutting planes* (*P*)

begin

1. risolvi il rilassamento $C(P)$ ottenendo x^C
2. if $C(P)$ è illimitato o impossibile then stop;
3. while x^C non è intero do
4. determina un cutting plane $\alpha x \geq \alpha_0$ ed aggiungilo ai vincoli di P
5. risolvi il rilassamento $C(P)$ ottenendo x^C
6. if $C(P)$ è impossibile then stop;
7. end while

end (si ha $x^* = x^C$)

Algoritmo Cutting Planes (4)

- La procedura determina la soluzione ottima di P:
  - $C(P)$ contiene tutte e sole le soluzioni intere di P (più altre non intere)
  - I tagli aggiunti non eliminano soluzioni intere
  - $C(P)$ e P hanno la stessa funzione obiettivo

- La procedura è molto efficace quando la soluzione ottima del problema $C(P)$ non è troppo diversa da quella del problema P

Algoritmo Cutting Planes (5)

SVANTAGGI :

- Il numero delle iterazioni del ciclo while non è polinomiale
- Il problema di PL da risolvere per $C(P)$ diventa ad ogni iterazione sempre più grande
  $\Rightarrow$lunghi tempi di calcolo

- Come si generano in modo automatico i piani di taglio ?

Tagli di Gomory (1958)

- $\forall\ y \in R^1$, $\lfloor y \rfloor = \max\{q \text{ intero: } q \le y\}$
- Sia Y il tableau finale di $C(P)$:
  - Elementi del tableau y_{ij}, $i = 0,\ldots,m$; $j = 0,\ldots,n$
  - β = colonne nella base ottima $\mathcal{B}$; $x_{\beta(0)} = -z$
- $\forall$ riga del tableau $i = 0,\ldots.,m$ si ha:

$$y_{i0} = x_{\beta(i)} + \sum_{A_j \notin B} y_{ij} x_j \qquad (\alpha)$$

$$x \ge 0 \Rightarrow \sum_{A_j \notin B} \lfloor y_{ij} \rfloor x_j \le \sum_{A_j \notin B} y_{ij} x_j \Rightarrow$$

Tagli di Gomory (2)

intera $\forall$ *x* intera

$$y_{i0} = x_{\beta(i)} + \sum_{A_j \notin B} y_{ij} x_j \qquad (\alpha)$$

$$\Rightarrow x_{\beta(i)} + \underbrace{\sum_{A_j \notin B} \lfloor y_{ij} \rfloor x_j}_{} \le y_{i0}$$

$$\Rightarrow x_{\beta(i)} + \sum_{A_j \notin B} \lfloor y_{ij} \rfloor x_j \le \lfloor y_{i0} \rfloor \qquad (\beta)$$

- (β) è il taglio di Gomory (violato da x^C)

Tagli di Gomory (3)

- $(\alpha) - (\beta)$:

$$\sum_{A_j \notin B} \left(y_{ij} - \lfloor y_{ij} \rfloor\right) x_j \geq \left(y_{i0} - \lfloor y_{i0} \rfloor\right)$$

  - $f_{ij} = y_{ij} - \lfloor y_{ij} \rfloor$ (*parte frazionaria* di y_{ij}); $0 \leq f_{ij} \leq 1$:

$$\sum_{A_j \notin B} f_{ij} x_j \geq f_{i0}$$

taglio di Gomory in forma frazionaria corrispondente alla riga *i* (generatrice del taglio)

Tagli di Gomory (4)

- Moltiplicando per -1 ed aggiungendo una variabile slack

$$-\sum_{A_j \notin B} f_{ij} x_j + x_s = -f_{i0}$$

- Se si aggiunge al tableau finale di *C*(*P*) il taglio

1. non si elimina alcun punto intero ammissibile
2. il nuovo tableau contiene una base che non è ammissibile per il problema
    - si deve riottimizzare il problema o utilizzare un algoritmo specializzato (simplesso duale)