

Terzo compitino

Trasformare i vincoli di un problema in matrici

Possiamo vedere i vincoli di un problema come una matrice di coefficienti moltiplicata per il vettore di incognite, e \leq del vettore dei risultati dei vincoli dati dal problema. Infatti, anche se non tutti i vincoli sono \leq , possiamo ricondurci a questa forma attraverso equivalenze matematiche. Per esempio, nell'esempio della fonderia abbiamo questi vettori/matrici:

Esempio 3.2. Trasformazioni equivalenti

Il problema della Fonderia dell'Esempio 1.4

$$\begin{array}{rcllclclclcl}
 \min & 0.025x_1 & + & 0.030x_2 & + & 0.018x_3 & + & 10x_4 & & & \\
 & 4x_1 & + & x_2 & + & 0.6x_3 & & & & \geq & 3250 \\
 & 4x_1 & + & x_2 & + & 0.6x_3 & & & & \leq & 5500 \\
 & 0.45x_1 & + & 0.5x_2 & + & 0.4x_3 & + & 100x_4 & & \geq & 450 \\
 & x_1 & + & x_2 & + & x_3 & + & x_4 & = & 1000 \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & \geq & 0
 \end{array}$$

può essere portato in forma (3.1) applicando la trasformazione (3.2.i) alla funzione obiettivo, la trasformazione (3.2.ii) al quarto vincolo, la trasformazione (3.2.iii) al primo ed al terzo vincolo, ed introducendo vincoli espliciti per la non negatività delle variabili. Ciò porta alla formulazione

$$C = [0.025, 0.030, 0.018, 1] \rightarrow \text{VETTORE DEI COSTI DELLA F.O.}$$

$n=4$ $m=8 \rightarrow$ VINCOLI
 \downarrow var A

4	1	0.6	\emptyset
4	1	0.6	\emptyset
0.45	0.5	0.4	100
1	1	1	1
1	\emptyset	\emptyset	\emptyset
\emptyset	1	0	\emptyset
\emptyset	\emptyset	1	\emptyset
\emptyset	\emptyset	\emptyset	1

OP	6
\geq	3250
\leq	5500
\geq	450
$=$	1000
\geq	\emptyset
\geq	\emptyset
\geq	\emptyset
\geq	\emptyset

Prima forma normale

La prima forma normale è la forma che diamo ai nostri vincoli ovvero:

$$\max \quad \{cv : Ax \leq b\}$$

dove $c \in R^n$, $A \in R^{m \times n}$, $b \in R^m$

Trasformare tutti i vincoli in questa forma aiuta a risolvere meglio il problema in forma geometrica.

Per trasformare un vincolo di uguaglianza in un vincolo di disuguaglianza dobbiamo creare due vincoli di disuguaglianza che si completino.

Guardando l'esempio precedente,

$x_1 + x_2 + x_3 + x_4 = 1000$ possiamo scriverlo come

$$x_1 + x_2 + x_3 + x_4 \leq 1000 \text{ e } -x_1 - x_2 - x_3 - x_4 \leq -1000$$

Seconda forma normale

E' la forma che noi non usiamo ma che potremmo trovare nei libri di testo, ovvero:

$$\min \quad \{cv : Ax = b, x \geq 0\}$$

Per ottenerlo si aggiunge o sottrae una quantità s_1, s_2, \dots, s_m a ogni vincolo con una disequazione, e mettiamo come operando l'uguale.

Queste quantità saranno ≥ 0 , e si chiamano **variabili di scarto**, perché non ci importa molto del loro valore.

Trasformare una variabile non vincolata in segno in una vincolata

Nel nostro modus operandi le variabili non vincolate in segno possono tornarci scomode, perciò le vincoliamo formalmente.

Es:

X_m non è vincolata in segno.

Scriviamola come $X_m = X_m^+ - X_m^-$ con $X_m^+ \geq 0$ e $X_m^- \leq 0$ e poi nei vincoli del problema cambiamo X_m con la formula che abbiamo scritto. Così facendo aumentiamo anche il problema di un'incognita.

Gradiente

E' il vettore dato da una riga della matrice dei coefficienti.

Curve di livello

Le curve di livello sono tutti gli insiemi di punti che soddisfano l'equazione $f(x)=k$ per un valore di k deciso da noi. Le curve di livello sono ortogonali al gradiente che le crea.

Le curve di livello creano un Iperpiano

Curve di sottolivello

Sono come le curve di livello, ma soddisfano le equazioni $f(x) \leq k$, e quindi tagliano a metà la regione ammissibile. Il k sarà l'elemento i -esimo del vettore dei risultati.

Le curve di sottolivello creano un **semispazio**.

Il semispazio ortogonale al gradiente è sempre $n-1$ dimensioni.

Poliedro

E' un insieme di semispazi.

$$P. = \{x \in R^n : Ax \leq b\}$$

Faccia massimale

E' una faccia caratterizzata da un solo vincolo.

Faccetta

Una faccetta è fatta da più vincoli. Alcune sono ridondanti, infatti creano una figura più grande del reale poliedro.

$$P_s = \text{faccetta}.$$

$$P. = \{x \in R^n : Ax \leq b\},$$

$$I = \{1, 2, \dots, m\} \text{ indici dei vincoli,}$$

$$S \subseteq I,$$

$$P_s \subseteq P$$

$$P_s = \{x \in R^n : A_i x = b_i, i \in S, A_i x \leq b_i, i \in I/S\}$$

Vertici

I vertici sono faccette con m vincoli per un problema a m dimensioni.

- Non sempre presi m vincoli del problema corrisponde un vertice.
- Lo stesso vertice potrebbe essere identificato da vincoli diversi.
- Qualsiasi punto tranne il vertice può essere riscritto come inviluppo convesso di altri punti
- anche chiamato **punto estremo**

Inviluppo convesso

Dati due vertici, la regione ammissibile del problema è formata dal segmento che li unisce.

$$\text{conv}(X) = \{x = \sum_{i=1}^k \alpha_i x_i : \sum_{i=1}^k \alpha_i = 1, \alpha_i \geq 0, i = 1, \dots, k\}$$

$$X \subset R^n = \{x_1, x_2, \dots, x_k\} \rightarrow X \text{ è l'insieme di vertici.}$$

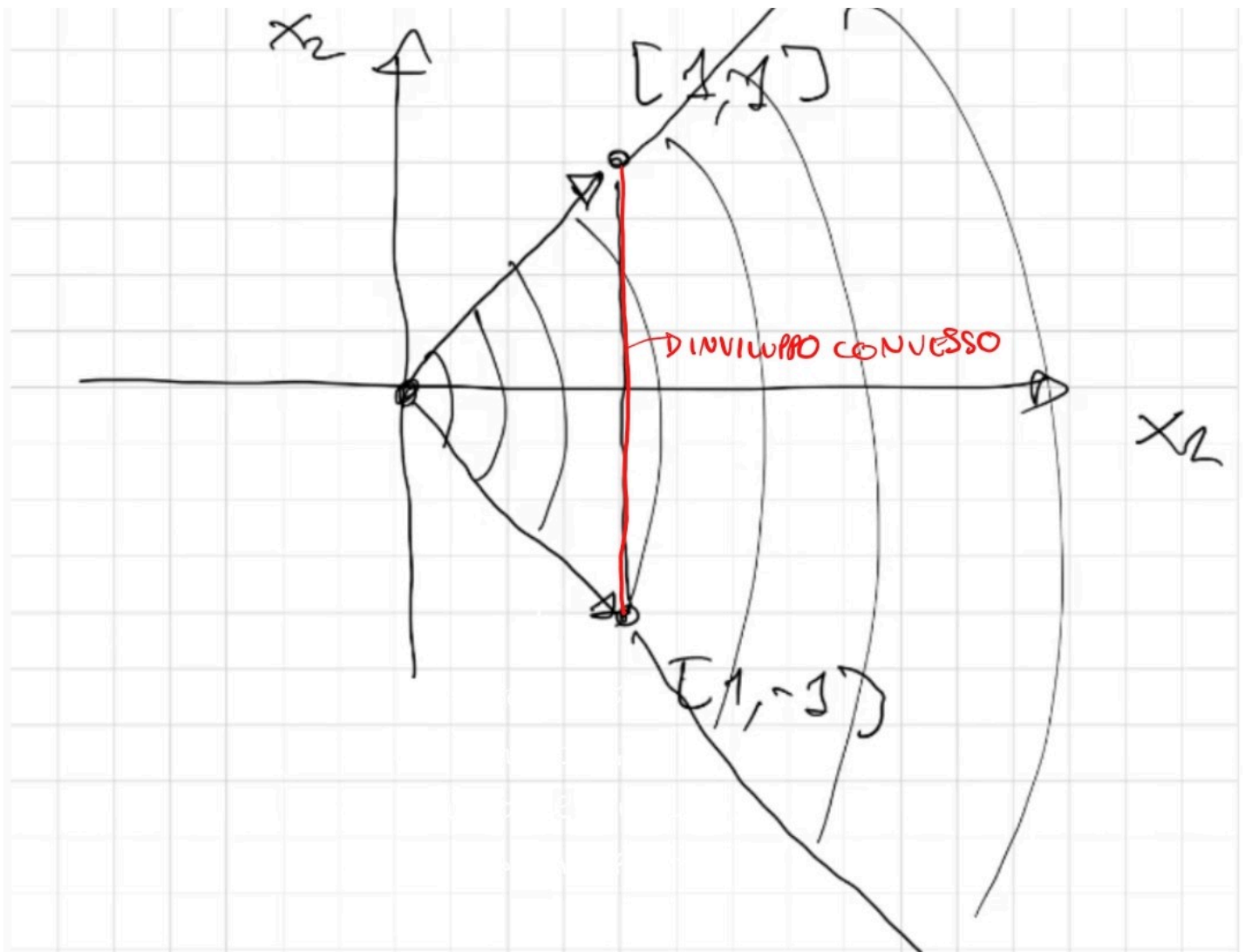
L'inviluppo convesso dell'insieme dei vertici si chiama **politopo**.

Risoluzione della PL in base ai vertici

Se noi abbiamo un problema in cui invece dei vincoli ci vengono dati i vertici, la risoluzione è banale: basta infatti valutare tutti i vertici facendo il prodotto scalare tra loro e c, il vettore dei costi della funzione obiettivo, e prendere il risultato migliore, matematicamente parlando mettendo $\alpha_i = 1$.

Inviluppo conico

E' un altro modo per creare il poliedro a partire dai i vertici. Possiamo vedere l'inviluppo conico come due vettori che partono dall'origine e vanno verso due vertici. l'area tra i due vettori è regione ammissibile del cono.



$V = \{v_1, v_2, \dots, v_n\} \subset \mathbb{R}^N \rightarrow V =$ insieme di vertici visti come direzioni.

$$\text{cono}(V) = \{v = \sum_{i=1}^h \beta_i v_i : \beta_i \geq 0, i = 1, \dots, h\}.$$

Mentre l'inviluppo convesso crea una faccia del poliedro, l'inviluppo conico si protrae all'infinito, creando un **cono finitamente generato**.

Il problema è **superiormente illimitato** se troviamo un prodotto scalare tra $\beta_j v_j$ e $c > 0$.

Il cono è anche chiamato **cono di recessione**, perché, se io ho un poliedro che si estende all'infinito verso una direzione, continuando per essa e facendo zoom out alla fine i vertici si fonderanno insieme e rimarrà visibile solo la direzione infinita.

Teorema di decomposizione dei poliedri

$$P = \text{conv}(X) + \text{cono}(V).$$

Nel caso di un poliedro chiuso, lo sviluppo conico ha come solo vertice l'origine, e il problema ha soluzione finita.

Calcolo Facce e vertici in \mathbb{R}^n

$$\text{vertici} = 2^n$$

$$\text{facce} = 2 * n$$

Direzione ammissibile

d è una direzione ammissibile per $x \in P$ se $\exists \bar{\alpha} > 0 \in P : \forall d \in [0, \bar{\alpha}]$

$$\bar{\alpha} = \min\{(b_i - A_i x) / A_i d : A_i d > 0\}$$

d è ammissibile per x se e solo se $A_i d \leq 0 \quad \forall i : A_i x = b_i$ (ovvero $A_i d \leq 0$ per ogni vincolo attivo in $I(X)$)

Direzione di recessione

Sono direzioni che, partendo da un punto ammissibile x , consentono di continuare all'infinito, rimanendo sempre nella regione ammissibile del poliedro.

$$\forall x \in P, \forall \lambda \geq 0 \quad (x + \lambda d) \in P$$

l'insieme delle direzioni di recessione si chiama $\text{rec}(P)$.

$\text{Rec}(P)$ non è mai vuoto, o $\text{rec}(P) = \{0\}$ o contiene almeno un d diverso da 0.

Direzione di linealità

d è una direzione di linearità se è percorribile da entrambe le parti come direzione di recessione, ovvero $d \in \text{rec}(P)$ e $-d \in \text{rec}(P)$.

d è una direzione di linearità di $P \Leftrightarrow A^*d=0 \Rightarrow A$ ha colonne linearmente dipendenti.

Risolvere P' in caso di direzione di linearità

Se A ha colonne L.D, posso escluderle iterativamente e risolvere un problema più piccolo del precedente:

$$(P) \quad \max\{cx : Ax \leq b\}$$

posso scriverlo come:

$$A = [A', A^n]$$

$$C = [C', C^n]$$

$$(P') \quad \max\{c'x' : A'x' + c_n x_n + a^n x^n \leq b\}$$

Risolvere P' significa che:

1. x' è superiormente illimitata $\Rightarrow (P)$ è superiormente illimitata o va creato un problema (P'') .
2. x' ottima finita \Rightarrow soluzione ammissibile per (P) .
 - a. se $cd \neq 0$, allora P è superiormente illimitato.
 - b. se $cd = 0$ allora P ammette soluzione ottima ed è proprio x' (significa che alla funzione obiettivo non serviva la colonna di A tolta).
3. (P') è vuoto $\Rightarrow (P)$ è vuoto.

Risolvere P' significa trovare una soluzione ammissibile per P , che di base sarebbe, e rimane, superiormente illimitata.

L'unico caso in cui P non è superiormente illimitata con una colonna L.D. è nel caso in cui nella funzione obiettivo non ci importa di quella colonna, e quindi $c^*d=0$.

Direzione di crescita

Una direzione di crescita è una direzione d tale per cui $c * d > 0$ (c = vettore di costi). Seguendola aumentiamo il valore della funzione obiettivo.

Se esiste una direzione di crescita e di recessione in P , il problema è **superiormente illimitato**.

Qual'è il max passo che posso fare da d per non violare i vincoli?

Es vincolo i -esimo:

$$\alpha \leq (b_i - a_i x) / A_i v_i \quad (\text{vedere lez 18})$$

Vincoli attivi

$$E' \text{ l'insieme } I(x) = \{i : A_i x = b_i\}$$

Da direzioni a coni

Esiste un metodo per trasformare un insieme di direzioni in cono in forma $\{x \in R^n : Ax \leq 0\}$ ovvero l'algoritmo di proiezione che il prof spiega in parte

Coni simpliciali

Sono dei coni che sono semplici da trovare i generatori per il cono in maniera finitamente generata. $conoS = \{d \in R^n : A_b d \leq 0\}$

A ha m vincoli

$$A_{I(x)} d \leq 0$$

Caratteristiche del cono:

1. L'insieme degli indici dei vincoli attivi ha cardinalità n, quindi la matrice $A_{i(x)}$ è quadrata. Quindi $|I(x)| = n$
2. La matrice deve essere invertibile, ovvero $\det(A_{i(x)}) \neq 0$
3. $I(x)$ forma una base $B \subset \{1, 2, \dots, m\}$ dove $\{1, 2, \dots, m\}$ è l'insieme degli indici

Corollario

1. Se l'insieme delle soluzioni ottime è non vuoto, allora almeno una delle soluzioni ottime è in corrispondenza di un vertice.
2. Se un punto all'interno di una faccia è soluzione ottima del problema, allora tutti punti all'interno della faccia sono soluzione ottima
3. x^* è soluzione ottima se e solo se non esiste una direzione di crescita
4. Se D ha base degenera = sol multiple nel primale e viceversa.

Soluzione primale

La soluzione primale di base è la soluzione $x = A_B^{-1} b_B$ dove x è un vertice

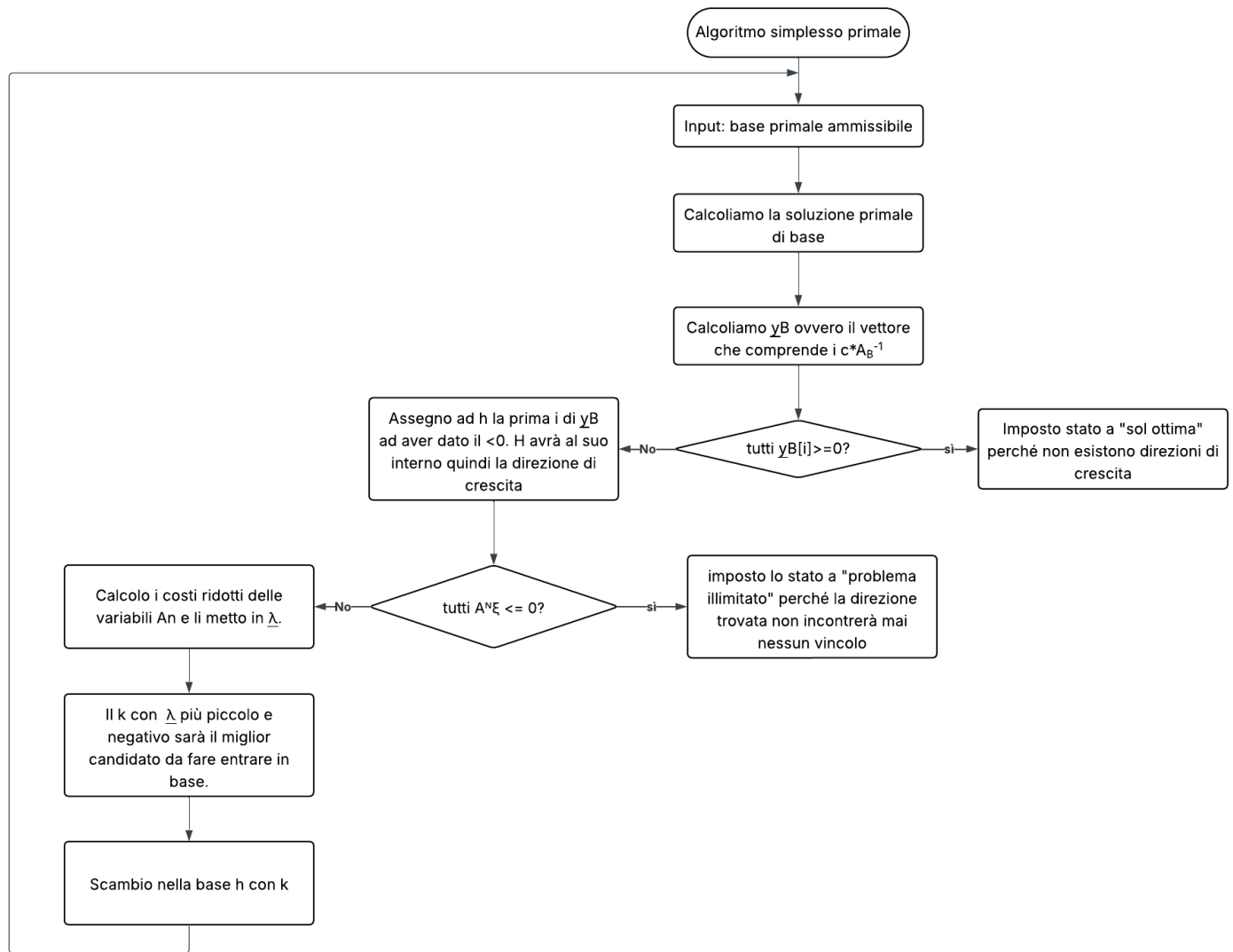
- x è ammissibile se i vincoli attivi creano un vertice
- se x non è ammissibile c'è almeno un vincolo non verificato $A_i x > b_i$
- la base deve essere non degenera, ovvero generata dal numero minimo di vincoli $I(x) = |n|$ in R^n .

Algoritmo simplesso primale

```

procedure (  $B, \bar{x}, \bar{y}, stato$  ) = Simplesso_Primale(  $A, b, c, B$  ) {
  for(  $stato = \text{" "}$  ; ) { // invariante: B primale ammissibile
     $\bar{x} = A_B^{-1} b_B$ ;  $\bar{y} = [\bar{y}_B, \bar{y}_N] = [c A_B^{-1}, 0]$ ;
    if(  $\bar{y}_B \geq 0$  ) then {  $stato = \text{"ottimo"}$ ; break; }
     $h = \min\{i \in B : \bar{y}_i < 0\}$ ;  $\xi = -A_B^{-1} u_{B(h)}$ ;
    if(  $A_N \xi \leq 0$  ) then {  $stato = \text{"P.illimitato"}$ ; break; }
     $\bar{\lambda} = \min\{\lambda_i = (b_i - A_i \bar{x}) / (A_i \xi), A_i \xi > 0, i \in N\}$ ;
     $k = \min\{i \in N : \lambda_i = \bar{\lambda}\}$ ;  $B = B \cup \{k\} \setminus \{h\}$ ;
  }
}
```

- \bar{y}_B è tutto il vettore che comprende i $c * A_B^{-1}$
- questo algoritmo funziona solo con problemi non vuoti
- base primale significa una base di un cono simpliciale
- ξ è la colonna del vincolo associata alla variabile non di base che vuoi far entrare in base
- A_N = le colonne di A corrispondenti alle variabili non di base
- $\bar{\lambda}_j$ = costo ridotto della variabile j per tutte le variabili non in base.



Se la base è degenere, l'algoritmo potrebbe fare un **passo degenerare**, ovvero ottenere lo stesso vertice cambiando il cono simpliciale che creano le direzioni in base.

Tipi di base

tipo di base	\bar{x}	\bar{y}
ammissibile	$A_N \bar{x} \leq b_N$	$\bar{y}_b \geq 0$
non ammissibile	$\exists i \in N : A_i \bar{x} > b_i$	$\exists i \in B : \bar{y}_i < 0$
degenere	$\exists i \in N : A_i \bar{x} = b_i$	$\exists i \in B : \bar{y}_i = 0$
non degenere	$\exists i \in N : A_i \bar{x} \neq b_i$	$\exists i \in B : \bar{y}_i \neq 0$

Correttezza algoritmo

L'algoritmo è corretto, e può non terminare in caso di basi non degeneri, ma la probabilità di non terminazione è davvero molto bassa anche se potremmo evitarla con la **regola di anticiclo di Bland**

Regola di anticiclo di Bland

nel caso in cui esistano più indici h candidati ad uscire dalla base corrente B e/o più indici k candidati ad entrare in base, l'algoritmo seleziona sempre l'indice minimo, cioè $h = \min\{i \in B : \bar{y}_i < 0\}$ e $k = \min\{i \in N : \lambda_i = \bar{\lambda}\}$

Generare una base ammissibile

Nonostante l'algoritmo del simplesso primale richieda una base ammissibile, per generarla usiamo l'algoritmo stesso applicato a un problema ausiliario opportunamente costruito.

Simpleso duale

$$(D) \quad \min\{y * b : y * A = c, y \geq 0\}$$

```

procedure (  $B, \bar{x}, \bar{y}, stato$  ) = Simpleso_Duale(  $A, b, c, B$  ) {
  for(  $stato = \text{" "}$  ; ; ) { // invariante:  $B$  duale ammissibile
     $\bar{x} = A_B^{-1}b_B$ ;  $\bar{y} = [\bar{y}_B, \bar{y}_N] = [cA_B^{-1}, 0]$ ;
    if(  $A_N\bar{x} \leq b_N$  ) then {  $stato = \text{"ottimo"}$ ; break; }
     $k = \min\{i \in N : A_i\bar{x} > b_i\}$ ;  $\eta_B = A_kA_B^{-1}$ ;
    if(  $\eta_B \leq 0$  ) then {  $stato = \text{"P.vuoto"}$ ; break; }
     $\bar{\theta} = \min\{\theta_i = \bar{y}_i/\eta_i : \eta_i > 0, i \in B\}$ ;
     $h = \min\{i \in B : \theta_i = \bar{\theta}\}$ ;  $B = B \cup \{k\} \setminus \{h\}$ ;
  }
}

```

Caratteristiche

- Il vettore η_b è il gradiente del vettore $A_i x$ del vincolo violato k e moltiplicato a sinistra per Ab^{-1}
- Se $\eta_b > 0$ il primale è vuoto e ci si ferma
- k = vincolo fuori base del definizione
- h = vincolo in base per definizione
- cerchiamo a ogni iterazione di inserire i vincoli violati per avvicinarci sempre di più alla soluzione del primale (ricordiamo che D crea il limite superiore della soluzione ottima di P).
- $\bar{\theta}$ è il più corto dei passi che azzerano le variabili nel vettore \bar{y}_i (=il complementare di y)

Legame tra (P) e (D)

(P) (D)	Sol. ottima	Vuoto	Illimitato
Sol. ottima	✓	✗	✗
Vuoto	✗	✓	✓
Illimitato	✗	✓	✗

Teorema debole della dualità

Se abbiamo un \bar{x} ammissibile per (P) e un \bar{y} ammissibile per (D), allora $Z(D) \geq Z(P)$, dove $Z(D) = y * b$

Teorema forte della dualità

$Z(D) = Z(P)$ Con valore ottimo finito.

Condizione degli scarti complementari

$$\sum_{i=1}^m y_i(b_i - A_i x^*) = 0 \text{ dove } y_i^* \geq 0 \text{ e } (b_i - A_i x^*) \geq 0.$$

Solitamente noi vediamo la formula in questa forma (due equazioni):

- $y_i^* \geq 0 \Rightarrow A_i x^* = b_i$
- $A_i x^* \leq b_i \Rightarrow y_i^* = 0$

Non possono essere entrambe verificate, o una, o l'altra, o nessuna, o entrambi verificati ma **uguali a zero** (Condizione *Snake eyes*).

Snake eyes

Gli scarti complementari del problema primale e duale sono entrambi a zero, quindi esiste una soluzione ottima, ma potrebbe non essere l'unica (problema di **basi degeneri**).

Caratteristiche problema

- La soluzione ottima del problema duale è un certificato di ottimalità per la soluzione ottima del problema primale.

Correttezza algoritmo

Possiamo dire che il problema del simplesso duale è corretto perché rispetta la condizione degli scarti complementari

Direzione di decrescita

La direzione di decrescita d è definita come:

$-\eta_i$ se $i \in B$

1 se $i = k$

0 altrimenti

Domande parziale

$x = [1,2]$ è ottima?

- verifico se è una soluzione ammissibile di P vedendo se viola o meno i vincoli
- se non li viola segno $I(x)$
- mi creo D con A_B
- prendo A_B e risolvo l'equazione con i termini noti b ($A_B \geq b$)
- creo il vettore di scarti, se non ci sono scarti negativi allora la soluzione del duale è ammissibile e per il teorema forte della dualità è ottima.

Quali sono tutte le soluzioni ottime del primale a partire da D?

se noi conosciamo D ammissibile, allora dobbiamo trovare il vettore Y degli scarti

2) una volta trovato, le y con scarto ≥ 0 , non devono averlo nel primale (controllo i vincoli per uguaglianza di $I(x)$)

3) ricordiamo che le soluzioni ottime di P sono molteplici \Leftrightarrow la base è degenere e quindi snake eyes.

Quale vincolo di D è in uscita?

- è in uscita il vincolo con $y < 0$
- se c è parametrico, sostituiamo e scriviamo in funzione di una variabile, e darà il limite della funzione ammissibile (ovvero che le y rimangano positive)

Trovare X^* in caso di D degenere

- il vettore di scarti ha uno zero nei vincoli attivi, significa che quella variabile è inutile per il creare delle soluzioni ottime, e può avere qualsiasi valore
- trovo le soluzioni ottime del primale quindi prendendo il vincolo attivo che non ha lo scarto in D e lo scrivo in funzione degli altri attivi di P

Risolvere PLI

Per risolvere la maggior parte dei problemi PLI, usiamo tanti problemi PL a catena, tutti connessi da una soluzione ammissibile che viene violata dal lancio successivo dell'algoritmo che aggiunge sempre 1 vincolo in più alla volta.

Branch & Bound

```

procedure  $z = BB(P, \varepsilon)$  {
   $Q = \{(P)\}$ ;  $z = -\infty$ ;
  do {  $(P') = Next(Q)$ ;  $Q = Q \setminus \{(P')\}$ ;
     $(\bar{z}, \underline{z}) = rilassamento(P')$ ;
    if ( $\underline{z} > z$ ) then  $z = \underline{z}$ ;
    if ( $\bar{z} \leq z + \varepsilon$ ) then continue;
     $\underline{z} = euristica(P')$ ;
    if ( $\underline{z} > z$ ) then  $z = \underline{z}$ ;
    if ( $\bar{z} \leq z + \varepsilon$ ) then continue;
     $Q = Q \cup branch(P')$ ;
  } while ( $Q \neq \emptyset$ );
}

```

Si tratta di fare l'**albero di decisioni** di tutte le scelte possibili delle variabili. Questo algoritmo diventa facilmente esponenziale, ed è un algoritmo generale che deve avere implementate le funzioni rilassamento(), euristica() e branch().