Un giorno, le 6 tribù decidono di costruire un totem ciascuna. Ogni totem deve:
- Avere in effige l'immagine di una delle specie di tartaruga presente sul territorio di quella tribù.
- Riportare una tartaruga diversa dagli altri totem.

Dimostrare che ciò è sempre possibile comunque siano fatti i territori.

23/11/2020 VENTURA

E-MAIL: paolo.ventura@iasi.cnr.it

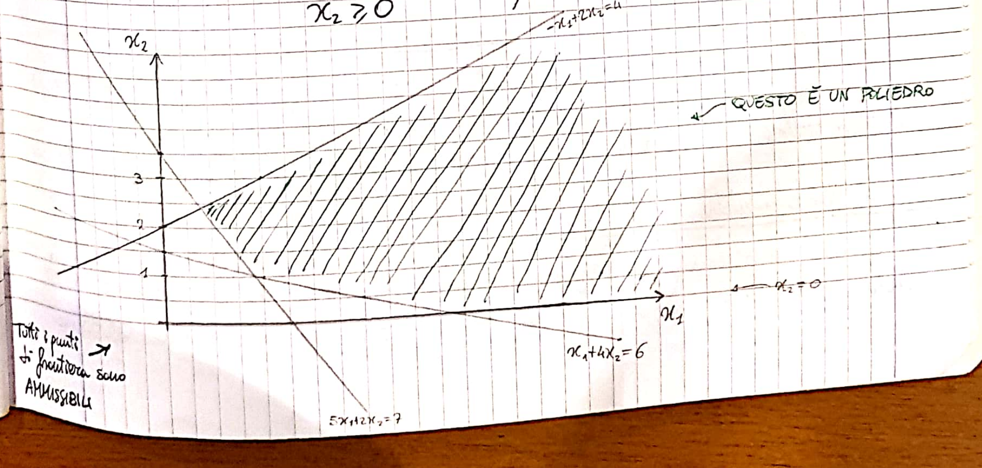
<u>Programmazione lineare:</u>

Un problema di PROGRAMMAZIONE LINEARE (PL) è un problema di ottimizzazione (in cui si vuole massimizzare o minimizzare una funzione obiettivo) in cui:
- La funzione obiettivo è lineare.
- La regione ammissibile è definita da un insieme <u>finito</u> di disuguaglianze <u>lineari</u> del tipo $\geq, \leq, =$.

ESEMPIO:

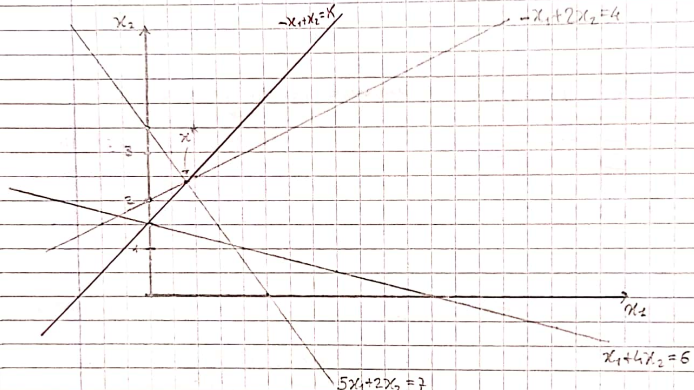
Vogliamo massimizzare la funzione obiettivo $-x_1+x_2$ con i seguenti vincoli lineari:

$$\left.\begin{array}{l} -x_1+2x_2 \leq 4 \\ 5x_1+2x_2 \geq 7 \\ x_1+4x_2 \geq 6 \\ x_2 \geq 0 \end{array}\right\} \text{REGIONE AMMISSIBILE}$$



Dobbiamo trovare il punto <u>ammissibile</u> che massimizza $-x_1+x_2$.
Scriviamo la retta $-x_1+x_2=K$

↳ Tutti i punti di questa retta avranno valore in funzione obiettivo pari a K.



Il nostro scopo è trovare la retta del tipo $-x_1+x_2=K$ tale che:
- Contiene almeno un punto ammissibile
- Massimizza il valore in funzione obiettivo (in modo tale che K sia maggiore possibile).

Nel nostro caso è facile intuire che la retta $-x_1+x_2=K$ che stiamo cercando contiene il punto di ottimo x^*, che è dato dall'intersezione delle seguenti due rette:

$$\begin{array}{rl} \oplus & -x_1+2x_2=4 \\ \ominus & \underline{5x_1+2x_2=7} \\ & -6x_1^* \quad =3 \end{array} \Rightarrow x_1^*=1/2\ ;\quad x_2^*=9/4$$

Il nostro valore K è dato da: $K=-x_1^*+x_2^*=-\frac{1}{2}+9/4=\frac{7}{4}$

<u>Teorema fondamentale della PL:</u>

Sia $\mathcal{P}$ un problema di Programmazione Lineare il cui insieme delle soluzioni ammissibili è definito dal poliedro P. Se $\mathcal{P}$ ammette soluzione ottima, allora esiste una soluzione ottima in un vertice del poliedro P.

P non ammette soluzione ottima se:

- P è vuoto.
- La funzione obiettivo è illimitata superiormente (caso P = problema di massimizzazione) / illimitata inferiormente (caso P = problema di minimizzazione) nel poliedro P.

↳ CIÒ SI VERIFICA QUANDO IL PUNTO DI OTTIMO BISOGNA CERCARLO ALL'INFINITO

Definizione:

Se fosse infinito, potremmo avere dei poliedri "non lineari" (e.g. cerchi)

Un POLIEDRO è l'insieme delle soluzioni ammissibili di un sistema finito di disuguaglianze lineari del tipo $\geq, \leq, =$.

↳ Se considerassimo disuguaglianze del tipo $>, <$, i punti di frontiera non sarebbero ammissibili e non avremmo un punto di ottimo (che, appunto, sarebbe un vertice)

Lemma:

Un poliedro P è un insieme convesso.

$Q \subseteq \mathbb{R}^n$ è convesso $\iff$ vale la seguente proprietà:

$x, y \in Q \Rightarrow \frac{1}{2}x + \frac{1}{2}y \in Q$ ← Il segmento che unisce x, y deve essere interamente nel poliedro

→ Se P è un poliedro definito in $\mathbb{R}^n$ $\Rightarrow$ $P := Ax \leq b$, dove → $A \in \mathbb{R}^{m \times n}$ è una matrice con m righe, n col.

→ $B \in \mathbb{R}^n$

$$P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$$

trasposto

- $a^t x \geq a_0 \iff (-a)^t x \leq -a_0$ ← Anche se si ha una relaz. del tipo $Ax \geq b$, si può invertire facilmente
- $a^t x = 0 \iff \begin{cases} a^t x \leq a_0 \\ -a^t x \leq -a_0 \end{cases}$

è un prodotto scalare

a È UN VETTORE COLONNA DI COEFF.
(a^t È UN VETTORE RIGA)
x È UN VETTORE COLONNA DI VARIABILI

→ Se P è un poliedro RAZIONALE definito in $\mathbb{R}^n$ $\Rightarrow$ $P = \{x \in \mathbb{R}^n \mid Ax \leq b,$ con $A \in \mathbb{Z}^{m \times n}, B \in \mathbb{Z}^n\}$

↳ I suoi vertici hanno tutti componenti razionali

24/11/2020

Il massimo di $c^t x$ è equivalente ~~al mi~~ all'opposto del minimo di $(-c)^t x$

Il minimo di $c^t x$ è equivalente all'opposto del massimo di $(-c)^t x$.

$\rightsquigarrow$ Se vogliamo massimizzare $c^t x$, dobbiamo prima trovare le $x \in \mathbb{R}^n \mid Ax \leq b$, $c \in \mathbb{Z}^n$, $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$

ESEMPIO:

Massimizzare la funzione obiettivo $-x_1 + x_2$ con i seguenti vincoli lineari:

$$\begin{cases} -x_1 + 2x_2 \leq 4 \\ 5x_1 + 2x_2 \geq 7 \quad \longleftrightarrow -5x_1 - 2x_2 \leq -7 \\ x_1 + 4x_2 \geq 6 \quad \longleftrightarrow -x_1 - 4x_2 \leq -6 \\ x_2 \geq 0 \quad \longleftrightarrow -x_2 \leq 0 \end{cases}$$

è importante avere tutti i segni di disuguaglianza concordi

ALLORA: $c = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$ $A = \begin{bmatrix} -1 & 2 \\ -5 & -2 \\ -1 & -4 \\ 0 & -1 \end{bmatrix}$ $b = \begin{bmatrix} 4 \\ -7 \\ -6 \\ 0 \end{bmatrix}$

Dimostriamo ora che qualunque poliedro P è convesso:

$P = \{x \in \mathbb{R}^n \mid Ax \leq b\} \overset{?}{\Longrightarrow} x, y \in P$ IMPLICA $\frac{1}{2}x + \frac{1}{2}y \in P$

Supponiamo quindi che $x, y \in P \Longrightarrow \begin{cases} Ax \leq b \\ Ay \leq b \end{cases}$

$A(\frac{1}{2}x + \frac{1}{2}y) \leq b \Rightarrow \frac{1}{2}\underbrace{Ax}_{\leq b \text{ per Hp}} + \frac{1}{2}\underbrace{Ay}_{\leq b \text{ per Hp}} \leq \frac{1}{2}b + \frac{1}{2}b = b \Rightarrow \frac{1}{2}x + \frac{1}{2}y \in P$

$\Rightarrow$ P è convesso. ■

I casi in cui c'è bisogno di imporre dei vincoli aggiuntivi alle variabili (come la loro interezza), che NON sono vincoli lineari.

Scansionato con CamScanner

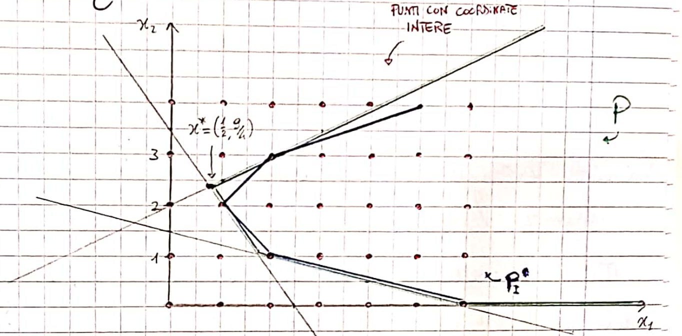
Problema di massimizzazione di $c^t x$: $Ax \leq b$, $x \in \mathbb{Z}^n$

↳ È UN PROBLEMA DI PROGRAMMAZIONE LINEARE INTERA (PLI)

ESEMPIO:

Massimizzare $-x_1 + x_2$ con i seguenti vincoli:

$$\begin{cases} -x_1 + 2x_2 \leq 4 \\ 5x_1 + 2x_2 \geq 7 \\ x_1 + 4x_2 \geq 6 \\ x_2 \geq 0 \\ x_1, x_2 \in \mathbb{Z} \end{cases}$$



Il nostro nuovo problema P_I non ammette più x^* come soluzione poiché $x^* \notin \mathbb{Z}^2$
⇝ I punti ammissibili sono punti isolati (non costituiscono un insieme convesso) ⇒ I PROBLEMI DI PROGRAMMAZIONE LINEARE SONO "FACILI", MENTRE QUELLI DI PROGRAMMAZIONE LINEARE INTERA SONO "DIFFICILI".

Tuttavia, è possibile ricondurre il nostro problema P_I a un problema di programmazione lineare tracciando il poliedro convesso P_I^*, che ha come vertici solo dei punti con coordinate intere interne al poliedro P

⇒ DA QUI SI PUÒ APPLICARE IL THM FONDAMENTALE DELLA PROGRAMM. LINEARE.

Consideriamo il seguente problema di programmazione lineare P_L:

~~MAX~~ MASSIMIZZARE $c^t x = z_L^*$, $P_L = \{x \in | Ax \leq b\}$

Consideriamo anche il seguente problema di programmazione lineare P_I:

MASSIMIZZARE $c^t x$, $X = \begin{cases} Ax \leq b \\ x \in \mathbb{Z}^n \end{cases}$

* CASO FACILE: il punto di ottimo nel problema di progr. lineare ha già coordinate intere

$P_I = \text{conv}(X) = \text{conv}(x \in P_L \mid x \in \mathbb{Z}^n)$

Il problema di massimizzare $c^t x$, $x \in X$ È EQUIVALENTE AL problema di massimizzare $c^t x$, $x \in P_I$

→ $\text{conv}(X)$ è l'insieme dei punti dello spazio che possono essere ottenuti come combinazione convessa dei punti di X.

Quindi teoricamente abbiamo un'equivalenza tra i problemi di programmaz. lineare e i problemi di programmaz. lineare intera. Tuttavia, nessuno ci dice com'è fatto il poliedro P_I che ha come vertici dei punti con coordinate intere interne a X → E CALCOLARLO SEMBRA DIFFICILE... *

Riconsideriamo i problemi $\mathcal{P}_L$, $\mathcal{P}_I$:

$$\mathcal{P}_L \begin{cases} \max \ c^t x = z_L^* \\ P_L \rightarrow Ax \le b \end{cases} \qquad \mathcal{P}_I \begin{cases} \max \ c^t x = z_I^* \ \text{(PUNTO DI OTTIMO)} \\ \underbrace{X}_{\text{INSIEME DEI PUNTI AMMISSIBILI}} \rightarrow \begin{cases} Ax \le b \\ x \in \mathbb{Z}^n \end{cases} \end{cases}$$

→ $\mathcal{P}_L$ è il RILASSAMENTO LINEARE di $\mathcal{P}_I$.

$P_L \supseteq X \implies \underbrace{z_L^* \geq z_I^*}_{\text{VALIDO PER LA MASSIMIZZAZIONE}}$

26/11/2020

Esercizio 1:

	PANE (1 Kg)	LATTE (1 l)	UOVA (1 Kg)	CARNE (1 Kg)	DOLCE (1 Kg)
▸ CALORIE (Kcal)	110	160	180	260	420
▸ PROTEINE (g)	4	8	13	14	4
▸ CALCIO (mg)	2	285	54	80	22

Fabbisogno minimo: 2000 Kcal + 50 g PROTEINE + 700 mg CALCIO

	① (1 Kg) PANE	② (1 l) LATTE	③ (1 Kg) UOVA	④ (1 Kg) CARNE	⑤ (1 Kg) DOLCE
▸ COSTO (€)	2	3	4	19	20
▸ MAX	4	8	3	2	2

Il nostro scopo è soddisfare il fabbisogno minimo senza superare il limite massimo di quantità e minimizzando il costo.

→ Dobbiamo scrivere il problema di programmazione lineare corrispondente.

x_i $(i=1,\dots,5)$ = quantità di prodotto i acquistata

Dobbiamo minimizzare la funzione obiettivo $\quad 2x_1+3x_2+4x_3+19x_4+20x_5$

~~Regole~~ Vincoli:

$$\begin{cases} x_1 \leq 4 \\ x_2 \leq 8 \\ x_3 \leq 3 \\ x_4 \leq 2 \\ x_5 \leq 2 \\ 110x_1+160x_2+180x_3+260x_4+420x_5 \geq 2000 \\ 4x_1+8x_2+13x_3+14x_4+4x_5 \geq 50 \\ 2x_1+285x_2+54x_3+80x_4+22x_5 \geq 700 \end{cases}$$

Non dimenticarti di questi vincoli!

$$\begin{cases} x_1 \geq 0 \\ x_2 \geq 0 \\ x_3 \geq 0 \\ x_4 \geq 0 \\ x_5 \geq 0 \end{cases}$$

A questo punto vogliamo scrivere questo problema nella seguente forma:

$\min \; c^t x \qquad Ax \geq b$

$$C = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 19 \\ 20 \end{bmatrix} \qquad A = \begin{bmatrix} -1 & & & & \\ & -1 & & & \\ & & -1 & & \\ & & & -1 & \\ & & & & -1 \\ 110 & 160 & 180 & 260 & 420 \\ 4 & 8 & 13 & 14 & 4 \\ 2 & 285 & 54 & 80 & 22 \\ 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix} \qquad b = \begin{bmatrix} -4 \\ -8 \\ -3 \\ -2 \\ -2 \\ 2000 \\ 50 \\ 700 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Esercizio 2:

PRODUZIONE LAMINATOIO
- BARRE (B) → capacità di produzione: 3 tonnellate/min
- TONDINI (T) → capacità di produzione: 2 tonnellate/min

DISPONIBILITÀ LAMINATOIO: 40 ore

QUANTITÀ MASSIMA DI PRODUZIONE
- 6000 tonnellate (B)
- 4000 tonnellate (T)

PROFITTO
- 25 € / tonn. (B)
- 30 € / tonn. (T)

Si vuole massimizzare il profitto rispettando i vincoli sopra indicati.

x_B = quantità di barre prodotta (in tonnellate)
x_T = quantità di tondini prodotta (in tonnellate)

Dobbiamo massimizzare la funzione obiettivo $25x_B + 30x_T$

Vincoli:

$$\begin{cases} x_B \leq 6000 \\ x_T \leq 4000 \\ \frac{1}{180}x_B + \frac{1}{120}x_T \leq 40 \quad \longleftrightarrow \quad 2x_B + 3x_T \leq 14400 \\ x_B \geq 0 \\ x_T \geq 0 \end{cases}$$

ALTRO POSSIBILE APPROCCIO:

y_B = minuti di lavorazione delle barre

y_T = minuti di lavorazione dei tondini

Dobbiamo massimizzare la funzione obiettivo $75y_B + 60y_T$

Vincoli:

$$\begin{cases} 3y_B \leq 6000 \\ 2y_T \leq 4000 \\ y_B + y_T \leq 2400 \\ y_B \geq 0 \\ y_T \geq 0 \end{cases}$$

$x_B = 3y_B$
$x_T = 2y_T$

30/11/2020

Esercizio 1:

- COMMESSI TIPO A: disponibili a lavorare esattamente 5 giorni consecutivi qualsiasi, guadagnando 50 € al giorno.
- COMMESSI TIPO B: disponibili a lavorare solo sabato e domenica per 70€ al giorno.
- COMMESSI TIPO C: disponibili a lavorare solo un giorno qualsiasi della settimana, tranne sabato e domenica, per 60€.

Numero di commessi da avere giorno per giorno:

	LUNEDÌ	MARTEDÌ	MERC.	GIOVEDÌ	VENERDÌ	SABATO	DOMEN.
COMMESSI	6	7	8	10	12	16	14

Formuliamo un problema di programmazione lineare intera che ci permetta di risolvere il problema di programmare i turni di lavoro in modo da coprire i turni e minimizzare il costo delle assunzioni.

VARIABILI:

- x_{A1} = # commessi di tipo A che lavorano dal giorno 1 al giorno 5

• x_{A_2} = # commessi di tipo ~~B~~ A che lavorano dal giorno 2 al giorno 6
• x_{A_3} = # commessi di tipo ~~C~~ A che lavorano dal giorno 3 al giorno 7
• x_B = # commessi di tipo B
• x_{C_i} = # commessi di tipo C che lavorano il giorno i, $i = 1, \dots, 5$

FUNZIONE OBIETTIVO (DA MINIMIZZARE):

$$250 \sum_{i=1}^{3} x_{A_i} + 140\, x_B + 60 \sum_{i=1}^{5} x_{C_i}$$

VINCOLI:

$$x_{A_1} + x_{C_1} \geq 6$$
$$x_{A_1} + x_{A_2} + x_{C_2} \geq 7$$
$$x_{A_1} + x_{A_2} + x_{A_3} + x_{C_3} \geq 8$$
$$x_{A_1} + x_{A_2} + x_{A_3} + x_{C_4} \geq 10$$
$$x_{A_1} + x_{A_2} + x_{A_3} + x_{C_5} \geq 12$$
$$x_{A_2} + x_{A_3} + x_B \geq 16$$
$$x_{A_3} + x_B \geq 14$$
$$x_{A_1} \geq 0$$
$$x_{A_2} \geq 0$$
$$x_{A_3} \geq 0$$
$$x_B \geq 0$$
$$x_{C_1} \geq 0$$
$$x_{C_2} \geq 0$$
$$x_{C_3} \geq 0$$
$$x_{C_4} \geq 0$$
$$x_{C_5} \geq 0$$
$$x_{A_1}, x_{A_2}, x_{A_3}, x_B, x_{C_1}, x_{C_2}, x_{C_3}, x_{C_4}, x_{C_5} \in \mathbb{Z}$$

Scansionato con CamScanner

$\min C^T x \qquad Ax \geq b \qquad x \in \mathbb{Z}^9$

$$C = \begin{bmatrix} 250 \\ 250 \\ 250 \\ 140 \\ 60 \\ 60 \\ 60 \\ 60 \\ 60 \end{bmatrix} \qquad A = \begin{bmatrix} 1 & & & & 1 & & & & \\ 1 & 1 & & & & 1 & & & \\ 1 & 1 & 1 & & & & 1 & & \\ 1 & 1 & 1 & & & & & 1 & \\ 1 & 1 & 1 & & & & & & 1 \\ & 1 & 1 & 1 & & & & & \\ & & 1 & 1 & & & & & \\ 1 & & & & & & & & \\ & 1 & & & & & & & \\ & & 1 & & & & & & \\ & & & 1 & & & & & \\ & & & & 1 & & & & \\ & & & & & 1 & & & \\ & & & & & & 1 & & \\ & & & & & & & 1 & \\ & & & & & & & & 1 \end{bmatrix}$$

$$b = \begin{bmatrix} 6 \\ 7 \\ 8 \\ 10 \\ 12 \\ 16 \\ 14 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Esercizio 2:

È necessario dotare un'isola con 5 paesi di alcuni presidi di guardia medica. Abbiamo quindi individuato 5 località A, B, C, D, E ognuna atta a ospitare un presidio. Le distanze in Km tra ciascun presidio e i 5 paesi P_1, P_2, P_3, P_4, P_5 sono riportate nella seguente matrice:

	P_1	P_2	P_3	P_4	P_5
A	11	33	10	18	25
B	20	8	17	12	30
C	23	21	9	34	12
D	7	24	26	14	22
E	28	10	21	32	11

Vogliamo attivare il numero minimo di presidi ma dobbiamo rispettare il seguente vincolo: per ogni paese, è necessario che sia attivo almeno un presidio nel raggio di 15 Km.

VARIABILI:

$$x_i = \begin{cases} 1 & \text{se apriamo un presidio in } i, \quad i \in \{A,B,C,D,E\} \\ 0 & \text{altrimenti} \end{cases}$$

FUNZIONE OBIETTIVO (DA MINIMIZZARE): $x_A + x_B + x_C + x_D + x_E$

VINCOLI:

$$\left.\begin{aligned} x_A + x_D &\geq 1 \\ x_B + x_E &\geq 1 \\ x_A + x_C &\geq 1 \\ x_B + x_D &\geq 1 \\ x_C + x_E &\geq 1 \end{aligned}\right\} \rightarrow \text{Vincoli per i paesi } P_1, \dots, P_5$$

$$\left.\begin{aligned} x_A &\geq 0 \\ x_B &\geq 0 \\ x_C &\geq 0 \\ x_D &\geq 0 \\ x_E &\geq 0 \\ x_A &\leq 1 \\ x_B &\leq 1 \\ x_C &\leq 1 \\ x_D &\leq 1 \\ x_E &\leq 1 \\ x_A, x_B, x_C&, x_D, x_E \in \mathbb{Z} \end{aligned}\right\}$$

Vincoli di binarietà: possono essere espresse con

$$x_A, x_B, x_C, x_D, x_E \in \{0,1\}$$

$$\min \ \mathbb{1}^t x \qquad x \in \{0,1\}^5 \qquad Ax \geq b$$

$$A = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

$b = \mathbb{1} \rightarrow$ vettore unitario

D,E} → È un problema di CARDINALITÀ MINIMA, dato che consiste nel prendere un insieme $\subseteq \{x_A, x_B, x_C, x_D, x_E\}$ più piccolo possibile rispettando i vincoli.

→ Il problema è detto anche di SET COVERING, perché si risolve prendendo il minor numero di colonne della matrice $A \in \{0,1\}^{m \times n}$ in modo tale che TUTTE le righe contengano almeno un valore pari a 1.

↳ Questo è l'unico modo per cui il prodotto Ax dà luogo a un vettore con tutte componenti non nulle ($\geq$ vettore unitario b)

Esercizio 3:

Si hanno m impianti che producono detersivo in polvere. Ciascun impianto i ha una capacità produttiva ω_i.

Si hanno inoltre n clienti. Ciascun cliente j ha una domanda b_j di detersivo.

PUÒ PROVENIRE ANCHE DA + IMPIANTI DIVERSI

$C \in \mathbb{Z}^{m \times n} \longrightarrow C_{ij}$ = costo unitario di trasporto dall'impianto i al cliente j.

Minimizzare i costi totali di trasporto.

VARIABILI:

x_{ij} = quantità di detersivo prodotta nell'impianto i e trasportata al cliente j.

FUNZIONE OBIETTIVO (DA MINIMIZZARE):

$$\sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} x_{ij}$$

VINCOLI:

$$\sum_{j=1}^{n} x_{ij} \leq \omega_i \qquad \forall i = 1, \dots, m \qquad \equiv \qquad \sum_{i=1}^{n} x_{ji} \leq \omega_j \qquad \forall j = 1, \dots, m$$

$$\sum_{i=1}^{m} x_{ij} \geq b_j \qquad \forall j = 1, \dots, n \qquad \equiv \qquad \sum_{j=1}^{m} x_{ji} \geq b_i \qquad \forall i = 1, \dots, n$$

$$x_{ij} \geq 0 \qquad \forall i = 1, \dots, m \qquad \forall j = 1, \dots, n$$

IL PRODOTTO È FRAZIONABILE, PER CUI NON SERVONO I VINCOLI DI INTEREZZA

Scansionato con CamScanner

01/12/2020

Riprendiamo lo stesso problema di ieri e aggiungiamo una nuova condizione: se si vuole tenere aperto l'impianto i, si aggiunge un costo q_i per l'apertura di questo impianto. Si vuole sempre minimizzare i costi soddisfando le domande dei clienti e senza sforare le capacità di produzione degli impianti.

Oltre a quelle che abbiamo già individuato, introduciamo anche delle variabili booleane y_i:

$$y_i \begin{cases} 1 & \text{se l'impianto } i \text{ viene aperto} \\ 0 & \text{se l'impianto } i \text{ viene chiuso} \end{cases}$$

FUNZIONE OBIETTIVO (DA MINIMIZZARE):

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} + \sum_{i=1}^{m} q_i y_i$$

VINCOLI:

$$\sum_{j=1}^{n} x_{ij} \leq w_i y_i \qquad \forall i = 1, \dots, m$$

$$\sum_{i=1}^{m} x_{ij} \geq b_j \qquad \forall j = 1, \dots, n$$

$$x_{ij} \geq 0 \qquad \forall i = 1, \dots, m \quad \forall j = 1, \dots, n$$

$$y_i \in \{0,1\} \qquad \forall i = 1, \dots, m$$

Complichiamo ulteriormente il problema: supponiamo ora di non essere obbligati a soddisfare le domande dei clienti; tuttavia, se una domanda b_j non viene soddisfatta, è necessario pagare una penale p_j.

Introduciamo anche delle variabili booleane z_j:

$$z_j \begin{cases} 1 & \text{se la domanda } b_j \text{ viene soddisfatta} \\ 0 & \text{se la domanda } b_j \text{ non viene soddisfatta} \end{cases}$$

FUNZIONE OBIETTIVO (DA MINIMIZZARE):

$$\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}\, x_{ij} + \sum_{i=1}^{m} q_i\, y_i + \sum_{j=1}^{n} p_j (1-z_j)$$

VINCOLI:

$$\sum_{j=1}^{n} x_{ij} \le w_i\, y_i \qquad \forall i = 1, \dots, m$$

$$\sum_{i=1}^{m} x_{ij} \ge b_j\, z_j \qquad \forall j = 1, \dots, n$$

$$x_{ij} \ge 0 \qquad \forall i = 1, \dots, m \qquad \forall j = 1, \dots, n$$

$$y_i \in \{0,1\} \qquad \forall i = 1, \dots, m$$

$$z_j \in \{0,1\} \qquad \forall j = 1, \dots, n$$

<u>Esercizio per casa:</u>

→ ASTA CON n OGGETTI

→ m OFFERTE DA PARTE DEI CLIENTI; UN'OFFERTA È COMPOSTA DA UN INSIEME DI OGGETTI E DA UN PREZZO COMPLESSIVO

$$n \text{ oggetti} \left\{ \underbrace{\begin{bmatrix} & 1 & & \\ 1 & & & \\ & 1 & & \\ 1 & 1 & \cdots & \\ & & & \\ & 1 & & \\ & & & \end{bmatrix}}_{m \text{ offerte}} \right. = D$$

1 nelle righe 2, 4: la prima offerta è sugli oggetti 2 e 4

$$p_j = \overbrace{\begin{bmatrix} 50 & 100 & \cdots & \end{bmatrix}}^{m \text{ offerte}}$$

Prezzo della prima offerta

Il vostro obiettivo è scegliere quali offerte accettare e quali rifiutare massimizzando il profitto e stando attenti a <u>non</u> vendere lo stesso oggetto a + clienti.

03/12/2020

Svolgiamo l'esercizio per casa, ponendo $n=8$, $m=7$:

← OFFERTE

$S =$

	1	2	3	4	5	6	7
1	1			1			
2		1				1	
3	1				1		
4		1				1	
5				1		1	1
6		1		1			
7	1		1				1
8			1		1		
	30	40	20	30	15	25	35

← PREZZI

↑ OGGETTI

VARIABILI:

$$x_i = \begin{cases} 1 & \text{se accetto l'offerta } i \\ 0 & \text{altrimenti} \end{cases}$$

FUNZIONE OBIETTIVO (DA MASSIMIZZARE):

$$\sum_{i=1}^{m} p_i x_i$$

VINCOLI:

$$\sum_{j=1}^{m} S_{ij} x_j \leq 1 \qquad \forall i = 1, \dots, n$$

$$x_i \in \{0,1\} \qquad \forall i = 1, \dots, m$$

La seguente formulazione di funzione obiettivo e vincoli è equivalente:

$$\max p^t x$$
$$Sx \leq \mathbb{1} \qquad \text{con } S \in \{0,1\}^{n \times m}$$
$$x \in \{0,1\}^m$$

→ QUESTO TIPO DI PROBLEMA È DETTO SET PACKING
↳ dove vogliamo scegliere l'insieme di colonne che dà luogo al costo max e le cui righe presentano al massimo un 1

Vertex coloring di un grafo:

Dato un grafo $G=(V,E)$, un VERTEX COLORING di G è un assegnamento di colori ai vertici di G: $f: V \to \mathbb{Z}_+$ tale che $f(u) \neq f(v)$ $\forall \{u,v\} \in E$.

Il problema del vertex coloring di G consiste nel "colorare" i vertici di G utilizzando il minimo numero di colori.

Proviamo a scrivere questo problema in un problema di programmazione lineare.

VARIABILI:

$$y_{ui} = \begin{cases} 1 & \text{se il vertice } u \text{ viene colorato col colore } i \\ 0 & \text{altrimenti} \end{cases} \qquad \forall u \in V, \ \forall i = 1, \dots, n \ \leftarrow |V(G)|$$

FUNZIONE OBIETTIVO (DA MINIMIZZARE):

$$\sum_{i=1}^{n} x_i \qquad \text{dove } x_i = \begin{cases} 1 & \text{se il colore } i \text{ viene utilizzato} \\ 0 & \text{altrimenti} \end{cases}$$

VINCOLI:

$$y_{ui} + y_{vi} \leq 1 \qquad \forall \{u,v\} \in E, \ \forall i = 1, \dots, n$$

$$\sum_{i=1}^{n} y_{ui} = 1 \qquad \forall u \in V$$

$$y_{ui} \in \{0,1\} \qquad \forall u \in V \quad \forall i = 1, \dots, n$$

$$x_i \geq y_{ui} \qquad \forall u \in V \quad \forall i = 1, \dots, n$$

$$x_i \in \{0,1\} \qquad \forall i = 1, \dots, n$$

→ Questi vincoli NON sono equivalenti ma possono essere sostituiti con:

$$n x_i \geq \underbrace{\sum_{u \in V} y_{ui}}_{=0 \iff \text{nessun vertice è colorato col colore } i} \qquad \forall i = 1, \dots, n$$

Sudoku:

La risoluzione di un sudoku si può ricondurre al problema di vertex coloring.

Ogni casella corrisponde a un vertice; ∃ spigoli tra:
→ i vertici della stessa riga
→ i vertici della stessa colonna
→ i vertici della stessa sottomatrice 3x3

			3	2			4		9
2	8			4					8
	1		6			5			7
7		8					6		6
		3				8			5
	9					2		3	4
		7			6		5		3
				9			1	4	2
	4			7	2				1
1	2	3	4	5	6	7	8	9	← righe

↑ colonne

VARIABILI:

$$y_{ijk} = \begin{cases} 1 & \text{se l'elemento } (i,j) \text{ viene colorato col colore } k \\ 0 & \text{altrimenti} \end{cases}$$

$\forall i,j,k = 1,\dots,9$

→ Non c'è una particolare funzione obiettivo.

VINCOLI:

- $y_{812} = 1 \quad , \quad y_{328} = 1 \,, \quad y_{721} = 1 \quad , \dots$ ← Pre-assegnamenti
- $\sum_{k=1}^{9} y_{ijk} = 1 \qquad \forall i = 1,\dots,9 \qquad \forall j = 1,\dots,9$ ← ∀ cella dobbiamo assegnare un numero (un colore) che
- $\sum_{j=1}^{9} y_{ijk} = 1 \qquad \forall i = 1,\dots,9 \qquad \forall k = 1,\dots,9$ ← # volte in cui la cifra k compare nella riga i
- $\sum_{i=1}^{9} y_{ijk} = 1 \qquad \forall j = 1,\dots,9 \qquad \forall k = 1,\dots,9$ ← # volte in cui la cifra k compare nella colonna j
- $\sum_{i=1}^{3} \sum_{j=1}^{3} y_{ijk} = 1 \qquad \forall k = 1,\dots,9$
- $\sum_{i=1}^{3} \sum_{j=4}^{6} y_{ijk} = 1 \qquad \forall k = 1,\dots,9$

  $\dots$

  # volte in cui la cifra k compare in ciascuna sottomatrice 3x3

- $y_{ijk} \in \{0,1\} \qquad \forall i,j,k = 1,\dots,9$

10/12/2020

Riprendiamo il problema del matching:

Sia $G=(V,E)$ un grafo non orientato.

$M \subseteq E$ è un matching di G se, dati $\{s,t\}, \{u,v\} \in M$ con $\{s,t\} \neq \{u,v\}$

$\Longrightarrow \{s,t\} \cap \{u,v\} = \emptyset$ ← s, t, u, v tutti diversi tra loro

Sia $w: E \rightarrow \mathbb{R}^+$ una funzione che indica il peso di ciascuno spigolo.

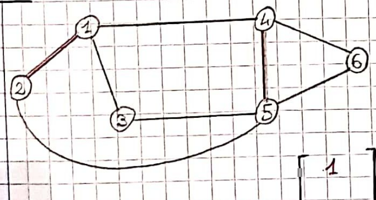
Il problema del matching di peso massimo possiamo esprimerlo come $\max w(M)$, dove M è un matching di G e $w(M) = \sum_{e \in M} w(e)$.

Introduciamo ora il VETTORE CARATTERISTICO, che è un vettore che indica un possibile sottoinsieme dell'insieme di partenza.

Sia x un vettore caratteristico e sia x_e un qualsiasi suo elemento.

$$x \in \{0,1\}^{|E|} \qquad x_e = \begin{cases} 1 & \text{se } e \in M \\ 0 & \text{se } e \notin M \end{cases}$$

ESEMPIO:



← Gli archi in rosso formano un possibile matching M

$$x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \rightarrow \begin{bmatrix} \{1,2\} \\ \{1,3\} \\ \{1,4\} \\ \{2,5\} \\ \{3,5\} \\ \{4,5\} \\ \{4,6\} \\ \{5,6\} \end{bmatrix}$$

FUNZIONE OBIETTIVO (DA MASSIMIZZARE):

$$\sum_{e \in M} \omega(e) = \sum_{e \in E} \omega_e x_e$$

VINCOLI:

Dobbiamo imporre che M è un matching di G:

$$|\{e \in M \cap \delta(u)\}| \leq 1 \quad \forall u \in V$$

← Il # di spigoli del matching che sono incidenti a u deve essere ≤ 1

Trasformiamolo in vincolo lineare

$$\sum_{e \in \delta(u)} x_e \leq 1 \quad \forall u \in V$$

$$x_e \in \{0,1\} \quad \forall e \in E$$

Vincolo esprimibile con una disuguaglianza del tipo $Hx \leq \mathbb{1}$, dove H è una matrice con $|V|$ righe e $|E|$ colonne

NELL'ESEMPIO PRECEDENTE:

È UNA MATRICE DI INCIDENZA

Questa è la matrice che impone in modo corretto la condizione $Hx \leq \mathbb{1}$

$$H = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} \begin{array}{c} \begin{array}{cccccccc} \{1,2\} & \{1,3\} & \{1,4\} & \{2,5\} & \{3,5\} & \{4,5\} & \{4,6\} & \{5,6\} \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \end{array}$$

Il problema appena descritto è di SET PACKING.

RICAPITOLANDO:

Dati $G = (V,E)$, $w \in \mathbb{R}^{|E|}$, il problema del matching di G di peso massimo equivale a:

$$\max \ w^t x$$
$$Hx \leq \mathbb{1} \qquad x \in \{0,1\}^{|E|}$$

dove H è la matrice di incidenza di G.

Formuliamo ora un altro problema.

Dati $G=(V,E)$, $c \in \mathbb{R}^{|V|}$:

Anche questo è un problema di SET PACKING

$$\max \; c^t y$$

$$H^t y \leq \mathbb{1} \qquad y \in \{0,1\}^{|V|}$$

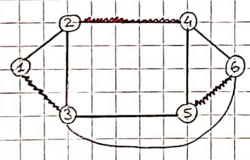
↳ È un problema in cui vogliamo selezionare un sottoinsieme S di vertici che siano a coppie non adiacenti (ovvero tali che, per ogni arco $\in E(G)$, l'arco è adiacente al più a un vertice di S).

⇝ $S \subseteq V$ per cui $H^t y \leq \mathbb{1}$ si chiama INSIEME STABILE di G.

⇒ Il problema appena descritto è noto come PROBLEMA DELL'INSIEME STABILE DI PESO MASSIMO.

14/12/2020

Dato $G=(V,E)$, $Q \subseteq E$ è un EDGE COVER di G se $\forall u \in V \; \exists \{u,v\} \in Q$



← ESEMPIO DI EDGE COVER

↳ $\delta(u) \cap Q \neq \emptyset \quad \forall u \in V$

→ Se gli archi sono caratterizzati da un peso, il problema dell'edge cover di peso minimo può essere formulato come:

VARIABILI:

$x \in \{0,1\}^{|E|}$ → Vettore che, per ogni componente x_e, indica se l'arco e appartiene o no all'edge cover

FUNZIONE OBIETTIVO (DA MINIMIZZARE):

$$\sum_{e \in E} w_e x_e$$

VINCOLI:

- $\sum_{t \in \delta(u)} x_t \geq 1 \quad \forall u \in V$ → Tutti i nodi devono essere adiacenti ad almeno un arco dell'edge cover
- $x_e \in \{0,1\} \quad \forall e \in E$

⇝ Equivalentemente si può scrivere $Hx \geq \mathbb{1}$, dove H è la matrice di incidenza di G.

↳ SI TRATTA DI UN PROBLEMA DI SET COVERING

Formuliamo ora un altro problema.

$$\min \sum_{v \in V} c_v y_v$$

$$H^t y \geq \mathbb{1} \qquad y_u \in \{0,1\} \quad \forall u \in V$$

↳ Questo è un problema in cui vogliamo trovare il $\overbrace{\text{VERTEX COVER}}^{\text{Insieme dei nodi che sono adiacenti a tutti gli archi del grafo}}$ di peso minimo di un grafo.

Problemi di scheduling:

Supponiamo di avere delle attività che devono essere svolte e che necessitano di risorse condivise (per cui non devono essere svolte tutte contemporaneamente).
Rappresentiamo le attività con dei nodi e le precedenze con degli archi orientati.
Un altro dato che ci offre il problema è la durata delle varie attività.

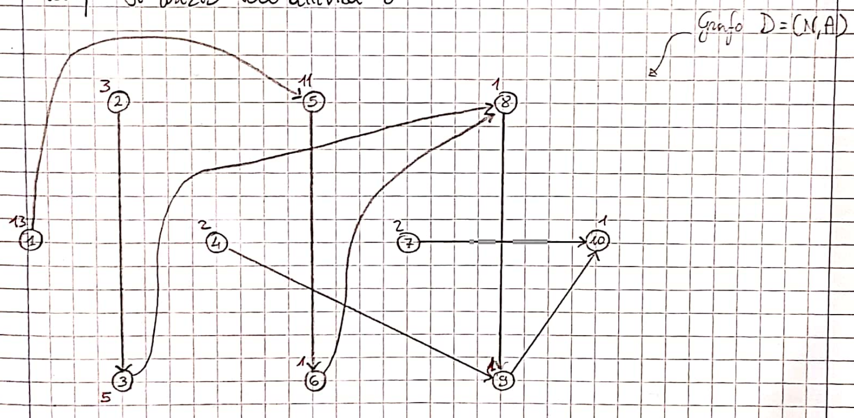
ESEMPIO (PREPARARE LA CARBONARA):

NUM.	ATTIVITÀ	DURATA (p)	PRECEDENZE
1	Bollire l'acqua	13	—
2	Preparare il soffritto	3	—
3	Cuocere il soffritto	5	2
4	Preparare le uova	2	—
5	Cuocere la pasta	11	1

6	Scolare la pasta	1	5
7	Grattugiare il pecorino	2	—
8	Aggiungere il soffritto	1	6, 3
9	Aggiungere le uova	1	4, 8
10	Aggiungere il pecorino	1	9, 7

<u>NB:</u> Una volta iniziata un'attività, <u>non</u> può essere interrotta per poi essere ripresa successivamente.

$t_i :=$ tempo di inizio dell'attività i



Scriviamo i vincoli lineari del problema:

$t_3 \geq t_2 + p_2$ $\qquad\qquad$ $t_5 \geq t_1 + p_1$ $\qquad$ ...

↳ IN GENERALE: $t_j \geq t_i + p_i \quad \forall (i,j) \in A$

$t_i \geq 0 \qquad \forall i \in N$

Il nostro obiettivo è minimizzare il tempo necessario a completare tutte le attività del progetto. (che indicheremo con T) $\Longrightarrow \min T$

Aggiungiamo così un altro vincolo: $\quad T \geq t_i + p_i \qquad \forall i \in N$

Tuttavia, non stiamo tenendo conto delle attività che condividono la stessa risorsa e che, quindi, non possono essere svolte contemporaneamente: se considerassimo solo le precedenze, sarebbe possibile svolgere contemporaneamente le attività 2, 4, cosa che non sarebbe possibile perché necessitano entrambe del piatto.

Introduciamo quindi un insieme R delle attività che necessitano della stessa risorsa:

$$R = \{2, 4, 6, 7, 8, 9, 10\}$$

$u, v \in R$ sono una COPPIA DISGIUNTIVA

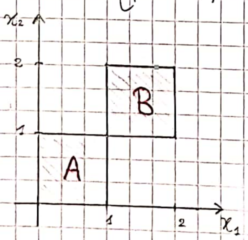
A valle di queste considerazioni, sembra che dovremo aggiungere vincoli di questo tipo:

$$\begin{cases} t_2 \geq t_4 + p_4 \\ \vee \\ t_4 \geq t_2 + p_2 \end{cases}$$

← VINCOLI IN "OR": NON VANNO BENE IN UN SISTEMA DI VINCOLI LINEARI

15/12/2020

Supponiamo di avere due regioni A, B fatte nel seguente modo:



← La regione $A \cup B$ NON è convessa, per cui non è esprimibile solo con vincoli lineari.

Tuttavia, le regioni A, B prese separatamente sono regioni ~~complesse~~ convesse:

$$A := \begin{cases} x_1 \leq 1 \\ x_1 \geq 0 \\ x_2 \geq 0 \\ x_2 \leq 1 \end{cases} \qquad B := \begin{cases} x_1 \leq 2 \\ x_1 \geq 1 \\ x_2 \geq 1 \\ x_2 \leq 2 \end{cases}$$

Per indicare che un punto x è ammissibile $\iff x \in A \vee x \in B$, introduciamo altre due variabili:

$$y_A = \begin{cases} 1 & \text{se } x \in A \\ 0 & \text{altrimenti} \end{cases} \qquad y_B = \begin{cases} 1 & \text{se } x \in B \\ 0 & \text{altrimenti} \end{cases}$$

Con la condizione che $y_A + y_B = 1$ indichiamo che il nostro punto ammissibile o si trova nella regione A o si trova nella regione B.

Scriviamo ora il seguente sistema di vincoli lineari:

$$\begin{cases} x_1 \leq 1 + 100(1-y_A) \\ x_1 \geq 0 - 100(1-y_A) \\ x_2 \leq 1 + 100(1-y_A) \\ x_2 \geq 0 - 100(1-y_A) \\ x_1 \leq 2 + 100(1-y_B) \\ x_1 \geq 1 - 100(1-y_B) \\ x_2 \leq 2 + 100(1-y_B) \\ x_2 \geq 1 - 100(1-y_B) \\ y_A + y_B = 1 \\ y_A, y_B \in \{0,1\} \end{cases}$$

→ se $x \in A$, questi termini scompaiono e ci ritroviamo le condizioni originali

→ se $x \in A$, queste condizioni diventano ridondanti

LA COSTANTE È DETTA BIG-M.

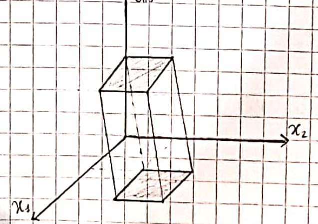
100 è una costante M arbitrariamente grande con lo scopo di rendere ridondanti ("disattivare") alcuni vincoli nel momento in cui non servono

Analogamente, al posto di introdurre due variabili y_A, y_B, avremmo potuto introdurre una sola:

$$y_{AB} = \begin{cases} 1 & \text{se } x \in A \\ 0 & \text{se } x \in B \end{cases}$$

In quest'ultimo caso possiamo introdurre una rappresentazione geometrica:

Tutto il parallelepipedo diventa una regione ammissibile se y_{AB} è una funzione continua che assume tutti i possibili valori reali tra 0 e 1.



Questo stesso ragionamento può essere applicato al problema di scheduling di ieri.

↳ E.G. PUÒ ESSERE INTRODOTTA UNA VARIABILE y_{24} T.C.

$$y_{24} = \begin{cases} 1 & \text{se l'attività 2 precede l'attività 4} \\ 0 & \text{se l'attività 4 precede l'attività 2} \end{cases}$$

↳ I VINCOLI CHE NE DERIVANO SONO:

$$\begin{cases} t_2 \geq t_4 + p_4 - M y_{24} \\ t_4 \geq t_2 + p_2 - M(1 - y_{24}) \\ y_{24} \in \{0, 1\} \end{cases}$$

→ POSSIAMO PRENDERE $M = \sum_{i=1}^{4} p_i$ PERCHÉ, NELLA PEGGIORE DELLE IPOTESI, ESEGUO LE ATTIVITÀ IN SEQUENZA (SENZA ALCUNA SOVRAPPOSIZIONE), PER CUI, $\forall i$ $\sum_{i=1}^{4} p_i \geq t_i + p_i$, IL CHE PORTA M A ESSERE UNA BIG-M

17/12/2020

Esercizio 1:

Supponiamo di avere dei vettori binari di dimensione 3: $\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$, $x_1, x_2, x_3 \in \{0,1\}$.

Scrivere dei vincoli lineari in modo tale che $x_3 = x_1$ AND x_2.

- $x_i \geq 0$, $x_i \leq 1$ $\forall i = 1, \dots, 3$
- $x_3 \leq x_1$
- $x_3 \leq x_2$
- $x_3 \geq x_1 + x_2 - 1$

Scrivere ora dei vincoli lineari in modo tale che $x_3 = x_1$ OR x_2

- $x_i \geq 0$, $x_i \leq 1$ $\forall i = 1, \dots, 3$
- $x_2 \leq x_3$
- $x_1 \leq x_3$
- $x_1 + x_2 \geq x_3$

Scrivere ora dei vincoli lineari in modo tale che $x_3 = x_1$ XOR x_2

- $x_i \geq 0$, $x_i \leq 1$ $\forall i = 1, \dots, 3$
- $x_3 \leq x_1 + x_2$
- $x_1 + x_2 + x_3 \leq 2$
- $x_3 \geq x_1 - x_2$
- $x_3 \geq x_2 - x_1$

21/12/2020

Esercizio 1:

Si hanno m tavole di lunghezza L. Dalle m tavole vogliamo tagliare n pezzi, con l_i = lunghezza del pezzo i-esimo.
Decidere da quale tavola tagliare ciascun pezzo i in maniera tale da minimizzare il numero complessivo di tavole utilizzate.

VARIABILI:

$y_j = \begin{cases} 1 & \text{se utilizzo la tavola } j \\ 0 & \text{altrimenti} \end{cases}$ || $x_{ij} = \begin{cases} 1 & \text{se il pezzo } i \text{ viene dalla tavola } j \\ 0 & \text{altrimenti} \end{cases}$

FUNZIONE OBIETTIVO:

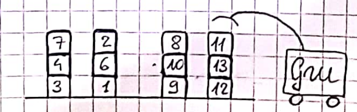
$$\min \sum_{j=1}^{m} y_j$$

VINCOLI:

- $y_j \in \{0,1\} \quad \forall j = 1,\dots,m$
- $x_{ij} \in \{0,1\} \quad \forall i = 1,\dots,n \quad \forall j = 1,\dots,m$
- $\left(\sum_{j=1}^{m} L y_j \geq \sum_{i=1}^{n} l_i \right)$ → Questo vincolo viene implicato dai due successivi, per cui può essere omesso.
- $\sum_{i=1}^{n} l_i x_{ij} \leq L y_j \quad \forall j = 1,\dots,m$
- $\sum_{j=1}^{m} x_{ij} = 1 \quad \forall i = 1,\dots,n$ ← Ogni pezzo va tagliato da una e una sola tavola

Esercizio 2:

Area di stoccaggio →



Container impilati a un'altezza nota

→ I numero indicano l'ordine di partenza dei container dall'area di stoccaggio

I container evidenziati in blu sono detti BLOCCANTI

Supponiamo di avere 3 pile di altezza 4 inizialmente vuote:

3
2
6
7
1
10
8
9
4
5

ORDINE CON CUI I CONTAINER DEVONO ESSERE INSERITI NELL'AREA DI STOCCAGGIO

Disporre i container nell'area di stoccaggio in modo tale da minimizzare il numero di container bloccanti.

VARIABILI:

$$x_{ij} = \begin{cases} 1 & \text{se il container } i \text{ è nella pila } j \\ 0 & \text{altrimenti} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{se il container } i \text{ è bloccante} \\ 0 & \text{altrimenti} \end{cases}$$

FUNZIONE OBIETTIVO:

$$\min \sum_{i=1}^{10} y_i$$

VINCOLI:

- $x_{ij} \in \{0,1\}$ $\quad \forall i = 1,\dots,10 \quad \forall j = 1,2,3$
- $y_i \in \{0,1\}$ $\quad \forall i = 1,\dots,10$
- $\sum_{i=1}^{10} x_{ij} \leq 4$ $\quad \forall j = 1,2,3$
- $\sum_{j=1}^{3} x_{ij} = 1$ $\quad \forall i = 1,\dots,10$
- $y_6 \geq x_{6j} + x_{3j} - 1$ $\quad \forall j = 1,2,3$ ← Se i container 3, 6 finiscono sulla stessa pila, allora 6 è un container bloccante
- $y_6 \geq x_{6j} + x_{2j} - 1$ $\quad \forall j = 1,2,3$

E COSÌ PER I CONTAINER 7, 10, 8, 9, 4, 5

In maniera compatta: $y_i \geq x_{ij} + x_{kj} - 1$

$\forall j = 1,2,3$

$\forall i,k : i > k \wedge \underbrace{\phi(i) > \phi(k)}_{\text{i si trova sotto k nella pila iniziale da 10}}$

22/12/2020

Esercizio 1:

Liquidi con i seguenti volumi: $V_1 = 180$, $V_2 = 100$, $V_3 = 120$, $V_4 = 200$, $V_5 = 90$

I 5 liquidi devono essere caricati in 4 navi di capacità definita dal vettore $C = [200, 80, 150, 120]$.

In ciascuna nave non possono essere caricati + di 3 liquidi diversi.

RICAVO: 3€/unità di volume per i primi tre liquidi

5€/unità di volume per gli ultimi due liquidi.

ALTRE CONDIZIONI:
- ∀ liquido devono essere caricate almeno 60 unità di volume.
- I liquidi 1, 3 non possono andare su due navi consecutive.

~~Svolgimento:~~ Massimizzare il ricavo complessivo per il liquido trasportato sulle navi.

VARIABILI:

x_{ij} = qtà di liquido i su nave j

$$y_{ij} = \begin{cases} 1 & \text{se liquido } i \text{ su nave } j \\ 0 & \text{altrimenti} \end{cases}$$

FUNZIONE OBIETTIVO:

$$\max \sum_{j=1}^{4} \left(3 \sum_{i=1}^{3} x_{ij} + 5 \sum_{i=4}^{5} x_{ij} \right) \longleftrightarrow \max \sum_{j=1}^{4} \sum_{i=1}^{5} w_i x_{ij}, \quad w = \begin{bmatrix} 3 \\ 3 \\ 3 \\ 5 \\ 5 \end{bmatrix}$$

VINCOLI:

$x_{ij} \geq 0 \quad \forall i = 1, \dots, 5 \quad \forall j = 1, \dots, 4$

$\sum_{j=1}^{4} x_{ij} \geq 60 \quad \forall i = 1, \dots, 5$

$\sum_{i=1}^{5} x_{ij} \leq C_j \quad \forall j = 1, \dots, 4$

$\sum_{i=1}^{5} y_{ij} \leq 3 \quad \forall j = 1, \dots, 4$

$\sum_{j=1}^{4} x_{ij} \leq V_i \quad \forall i = 1, \dots, 5$

$y_{ij} + y_{i(j+1)} \leq 1 \quad \forall i = 1, 3 \quad \forall j = 1, 2, 3$

$y_{ij} \in \{0, 1\} \quad \forall i = 1, \dots, 5 \quad \forall j = 1, \dots, 4$

$V_i y_{ij} \geq x_{ij} \quad \forall i = 1, \dots, 5 \quad \forall j = 1, \dots, 4$

Esercizio 2:

La distanza di Hamming tra due stringhe binarie di lunghezza n è pari al numero di bit in cui le due stringhe differiscono.

Siano $U_1 = [1, 0, 1, 0, 0, 0, 1]$

$U_2 = [0, 0, 0, 1, 0, 0, 0]$

$U_3 = [1, 1, 0, 1, 1, 0, 1]$ tre stringhe binarie di lunghezza 7.

Trovare la stringa $x = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]$ che minimizzi la sua distanza di Hamming massima dalle tre stringhe U_1, U_2, U_3.

VARIABILI:

x_i = valore dell'i-esimo bit della stringa x

h → variabile che ci aiuta a trovare il massimo tra le 3 distanze di Hamming

DEFINIAMO DELLE FUNZIONI PER AIUTARCI A DEFINIRE LA FUNZIONE OBIETTIVO:

Distanza di Hamming tra x, U_1: $H(x, u_1) = \sum_{i=1}^{7} diff(x_i, U_{1i}) =$

$= (1-x_1) + x_2 + (1-x_3) + x_4 + x_5 + x_6 + (1-x_7)$

$H(x, U_2) = x_1 + x_2 + x_3 + (1-x_4) + x_5 + x_6 + x_7$

$H(x, U_3) = (1-x_1) + (1+x_2) + x_3 + (1-x_4) + (1-x_5) + x_6 + (1-x_7)$

FUNZIONE OBIETTIVO: $\min h$

VINCOLI:

$x_i \in \{0, 1\} \qquad \forall i = 1, \dots, 7$

$h \geq 1 - x_1 + x_2 + 1 - x_3 + x_4 + x_5 + x_6 + 1 - x_7$

$h \geq x_1 + x_2 + x_3 + 1 - x_4 + x_5 + x_6 + x_7$

$h \geq 1 - x_1 + 1 - x_2 + x_3 + 1 - x_4 + 1 - x_5 + x_6 + 1 - x_7$

Esercizio 3:

Tre clienti con budget di 100 000 € ciascuno.

Occorre decidere come ripartire il budget in 2 fondi di investimento diversi in modo tale che i 3 clienti non possano avere soldi investiti nello stesso fondo.

Si hanno 10 fondi, con tassi di remunerazione definiti dal vettore t:

$t = [2,3,3,5,2,6,3,1,2,5]$ (%) e consentono budget min e max di investimento definiti rispettivam. dai vettori $l = [10,5,20,15,30,25,35,7,20,10]$ K € e $u = [50,55,60,45,30,75,65,70,40,30]$ K €.

Definire una ripartizione del budget dei 3 clienti tra i vari fondi di investimento in modo da massimizzare la remunerazione ~~massima~~ minima tra i 3 clienti.

VARIABILI:

x_{ij} = qtà investita dal cliente i nel fondo j

$$y_{ij} = \begin{cases} 1 & \text{se i investe in j} \\ 0 & \text{altrimenti} \end{cases}$$

R → variabile che ci aiuta a trovare il minimo tra le 3 remuneraz. totali dei clienti

FUNZIONE OBIETTIVO:

VINCOLI:

$y_{ij} \in \{0,1\}$ $\quad \forall i = 1,2,3 \quad \forall j = 1,\dots,10$

$\sum_{j=1}^{10} x_{ij} = 100$ $\quad \forall i = 1,2,3$ $\qquad \sum_{j=1}^{10} y_{ij} = 2$ $\quad \forall i = 1,2,3$

$x_{ij} \geq l_j y_{ij}$ $\quad \forall i = 1,2,3 \quad \forall j = 1,\dots,10$

$x_{ij} \leq u_j y_{ij}$ $\quad \forall i = 1,2,3 \quad \forall j = 1,\dots,10$

$\sum_{i=1}^{3} y_{ij} \leq 1$ $\quad \forall j = 1,\dots,10$

$x_{ij} \geq 0$ $\quad \forall i = 1,2,3$

$R \leq \sum_{j=1}^{10} t_j x_{ij}$ $\quad \forall i = 1,2,3$

07/01/2021

SIMPLESSO

Linear Programming di V. Chvatal:

Supponiamo di avere il seguente problema di programmazione lineare:

$\max \quad 5x_1 + 4x_2 + 3x_3$ ← funzione obiettivo che denomineremo $\mathcal{Z}$

$$\begin{cases} 2x_1 + 3x_2 + x_3 \leq 5 \\ 4x_1 + x_2 + 2x_3 \leq 11 \\ 3x_1 + 4x_2 + 2x_3 \leq 8 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

Riscriviamo i vincoli lineari nel seguente modo: → ABBIAMO AGGIUNTO LE VARIABILI x_4, x_5, x_6, CHE SONO DETTE VARIABILI DI SLACK

$$\begin{cases} x_4 = x_5 - 2x_1 - 3x_2 - x_3 \\ x_5 = 11 - 4x_1 - x_2 - 2x_3 \\ x_6 = 8 - 3x_1 - 4x_2 - 2x_3 \\ x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{cases}$$

← Mantenendo la funzione obiettivo $\max \; 5x_1 + 4x_2 + 3x_3$ otteniamo un problema equivalente a quello originale.

↳ Così è facile dedurre che una delle soluzioni ammissibili è: $\bar{x}_4 = 5$, $\bar{x}_5 = 11$, $\bar{x}_6 = 8 \Rightarrow \bar{x}_1 = \bar{x}_2 = \bar{x}_3 = 0$

↳ In questo modo la funzione obiettivo vale $\bar{\mathcal{Z}} = 0$ (il minimo sindacale) (*)

Adesso consideriamo la variabile che nella funzione obiettivo ha coefficiente + alto (x_1), portiamo a zero le altre due variabili originali (x_2, x_3) e cerchiamo di massimizzare x_1 rispettando i vincoli:

$$\begin{cases} x_4 = 5 - 2x_1 \geq 0 & \Rightarrow x_1 \leq 5/2 \\ x_5 = 11 - 4x_1 \geq 0 & \Rightarrow x_1 \leq 11/4 \\ x_6 = 8 - 3x_1 \geq 0 & \Rightarrow x_1 \leq 8/3 \end{cases}$$

La condizione + stringente è $x_1 \leq 5/2 \Rightarrow$ prendiamo la seguente soluzione:

$\tilde{x}_1 = 5/2$, $\tilde{x}_2 = 0$, $\tilde{x}_3 = 0$, $\tilde{x}_4 = 0$, $\tilde{x}_5 = 1$, $\tilde{x}_6 = 1/2 \Rightarrow \tilde{\mathcal{Z}} = 25/2$

$n = \#$ variabili originali

$m = \#$ vincoli lineari

$n' = n+m$

→ Abbiamo preso $\underbrace{m \text{ variabili espresse}}$ in funzione delle altre $\underbrace{n'-m \text{ variabili}}$.

x_4, x_5, x_6 ← VARIABILI DI BASE $\qquad$ x_1, x_2, x_3 ← VARIABILI NON IN BASE

$$\left.\begin{array}{ll} x_i = 0 & \forall x_i \text{ NON in base} \\ \Rightarrow x_i = b_i & \forall x_i \text{ in base} \end{array}\right\} \text{SOLUZIONE DI BASE } (*)$$

$b_i \geq 0 \quad \forall\, i = 1, \dots, m \quad \Rightarrow$ La soluzione di base è ammissibile

A questo punto, applicando il procedimento di prima (con cui si è ottenuto $\tilde{x}_1 = 5/2$), una variabile NON in base (x_1, appunto) entra in base al posto di un'altra (nel nostro caso x_4).

Reiteriamo dunque con x_1, x_5, x_6 IN BASE e x_4, x_2, x_3 NON IN BASE:

$$\begin{cases} x_1 = \frac{5}{2} - \frac{3}{2}x_2 - \frac{1}{2}x_3 - \frac{1}{2}x_4 \\ x_5 = 1 + 5x_2 + 2x_4 \\ x_6 = \frac{1}{2} + \frac{1}{2}x_2 - \frac{1}{2}x_3 + \frac{3}{2}x_4 \\ x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{cases}$$

} EQUAZ. OTTENUTE SOSTITUENDO x_1 A QUELLE ORIGINALI

$$z = \frac{25}{2} - \frac{7}{2}x_2 + \frac{1}{2}x_3 - \frac{5}{2}x_4$$

Se poniamo $\bar{x}_2 = \bar{x}_3 = \bar{x}_4 = 0 \Rightarrow \bar{x}_1 = 5/2, \quad \bar{x}_5 = 1, \quad \bar{x}_6 = 1/2, \quad Z = 25/2$ } SOLUZ. DI BASE

Ora, per aumentare la funzione obiettivo, possiamo solo aumentare x_3: è lei la variabile candidata a entrare in base:

$$\begin{cases} x_1 = 5/2 - 1/2\, x_3 \geq 0 & \Rightarrow x_3 \leq 5 \\ x_5 = 1 \geq 0 & \text{SEMPRE} \\ x_6 = 1/2 - 1/2\, x_3 \geq 0 & \Rightarrow x_3 \leq 1 \end{cases}$$

La condizione + stringente è $x_3 \leq 1 \Rightarrow$ prendiamo la seguente soluzione:

$\tilde{x}_2 = 0, \ \tilde{x}_3 = 1, \ \tilde{x}_4 = 0 \Rightarrow \tilde{x}_1 = 2, \ \tilde{x}_5 = 1, \ \tilde{x}_6 = 0$

Reiteriamo con x_1, x_3, x_5 IN BASE e x_2, x_4, x_6 NON IN BASE:

$$\begin{cases} x_1 = 2 - 2x_2 - 2x_4 + x_6 \\ x_5 = 1 + 5x_2 + 2x_4 \\ x_3 = 1 + x_2 + 3x_4 - 2x_6 \\ x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \end{cases}$$

$$z = \frac{25}{2} - \frac{7}{2}x_2 + \frac{1}{2}(1 + x_2 + 3x_4 - 2x_6) - \frac{5}{2}x_4 = 13 - 3x_2 - x_4 - x_6$$

Se poniamo $\bar{x}_2 = \bar{x}_4 = \bar{x}_6 = 0 \Rightarrow \bar{x}_1 = 2, \bar{x}_5 = 1, \bar{x}_3 = 1$ $\quad \bar{z} = 13$

→ Abbiamo trovato la soluzione OTTIMA, poichè, a questo punto, per qualunque variabile che aumentiamo nella nuova funzione obiettivo, quest'ultima decresce.

↳ Questo meccanismo che abbiamo adottato per calcolare la soluzione ottima del problema di programmaz. lineare dato è detto METODO DEL SIMPLESSO.

Riconsideriamo il medesimo problema nella sua forma originale:

$$\max \; 5x_1 + 4x_2 + 3x_3 = z \longrightarrow -z + 5x_1 + 4x_2 + 3x_3 = 0$$

$$\begin{cases} +2x_1 + 3x_2 + x_3 + x_4 = 5 \\ +4x_1 + x_2 + 2x_3 + x_5 = 11 \\ +3x_1 + 4x_2 + 2x_3 + x_6 = 8 \\ x_1, x_2, \dots, x_6 \geq 0 \end{cases}$$

Costruiamo il seguente tableau: (matrice)

	z	x_1	x_2	x_3	x_4	x_5	x_6	
x_4		+2	3	1	1	0	0	5
x_5		+4	1	2	0	1	0	11
x_6		+3	4	2	0	0	1	8
	-1	5	4	3	0	0	0	0

SI PUÒ OMETTERE, VISTO CHE È SEMPRE −1

Qui leggeremo sempre $-z$

Abbiamo: x_4, x_5, x_6 IN BASE e x_1, x_2, x_3 NON IN BASE.

x_1 ha il coefficiente in funzione obiettivo maggiore $\Rightarrow$ è candidata a entrare in base.

Ora, nella matrice consideriamo solo le righe con il coefficiente di x_1 positivo (queste righe corrispondono alle equazioni che limitano superiormente x_1):

→ Se non ce ne sono, la funzione obiettivo non è limitata superiormente e il problema non ha una soluzione ottima.

→ Se ce ne sono, tra tali righe, prendiamo quella con rapporto ultima colonna / coeff. di x_1 MINORE. Questa sarà la riga corrispondente alla variabile che uscirà dalla base (nel nostro caso x_4).

Otteniamo dunque la seguente matrice:

	x_1	x_2	x_3	x_4	x_5	x_6	
x_1	1	$3/2$	$1/2$	$1/2$	0	0	$5/2$
x_5	0	-5	0	-2	1	0	1
x_6	0	$-1/2$	$1/2$	$-3/2$	0	1	$1/2$
	0	$-7/2$	$1/2$	$-5/2$	0	0	$-25/2$

Abbiamo: x_1, x_5, x_6 IN BASE e x_2, x_3, x_4 NON IN BASE.

x_3 è la variabile candidata a entrare in base e, facendo le stesse considerazioni di prima, prenderà il posto di x_6.

Otteniamo quindi:

	x_1	x_2	x_3	x_4	x_5	x_6	
x_1	1	2	0	2	0	-1	2
x_5	0	-5	0	-2	1	0	1
x_3	0	-1	1	-3	0	2	1
	0	-3	0	-1	0	-1	-13

NON CI SONO COEFFICIENTI POSITIVI NELLA RIGA RELATIVA AI COEFFICIENTI DELLA FUNZ. OBIETTIVO →

⇒ Abbiamo trovato la soluzione ottima del problema di programmazione lineare.

11/01/2021

Consideriamo il seguente problema di programmazione lineare:

$$\max\ x_1 - x_2 + x_3$$

$$\begin{cases} -2x_1 + x_2 - 2x_3 \geq -4 \\ 2x_1 - 3x_2 + x_3 \leq -5 \\ -x_1 + x_2 - 2x_3 \leq -1 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

Riscriviamo i vincoli lineari nel seguente modo:

$$\begin{cases} -2x_1 + x_2 - 2x_3 - x_4 = -4 \quad \longleftrightarrow 2x_1 - x_2 + 2x_3 + x_4 = 4 \\ 2x_1 - 3x_2 + x_3 + x_5 = -5 \\ -x_1 + x_2 - 2x_3 + x_6 = -1 \\ x_1, \dots, x_6 \geq 0 \end{cases}$$

In questo esempio non è così banale trovare una soluzione ammissibile di base. Dobbiamo dunque introdurre una fase preliminare che ha lo scopo di trovare tale soluzione ammissibile:

$$\begin{cases} 2x_1 - x_2 + 2x_3 + x_4 = 4 \\ -2x_1 + 3x_2 - x_3 - x_5 + x_7 = 5 \\ x_1 - x_2 + 2x_3 - x_6 + x_8 = 1 \\ x_1, \dots, x_8 \geq 0 \end{cases} \Longleftarrow$$

Abbiamo invertito segno alle equazioni con termine noto negativo e ci abbiamo sommato un'altra nuova variabile (risp. x_7, x_8)

Se $\exists$ soluz. ammissibile con $x_7, x_8 = 0$, il problema precedente ammette le stesse soluz. ammissibili (e, quindi, è ammissibile)

ALTRIM. NON SAREBBE PROPRIO AMMISSIBILE

Vogliamo minimizzare $x_7 + x_8$

** STIAMO MINIMIZZANDO

(*) x_7 è la candidata a uscire dalla base

con questa riga sto esprimendo che la funz. obiettivo è $x_7 + x_8$

	2	-1	2	1	0	0	0	0	4	$\leftarrow x_4$
	-2	3	-1	0	-1	0	1	0	5	$\leftarrow x_7$
	1	-1	2	0	0	-1	0	1	1	$\leftarrow x_8$
	0	0	0	0	0	0	1	1	0	
costi ridotti $\rightarrow$	1	-2	-1	0	1	1	0	0	-6	

Li vogliamo nulli

$5+1=6$ [illegible]

Li abbiamo ottenuti sottraendo 2° e 3° riga dalla riga relativa alla funz. obiettivo (l'ultima)

$\hookrightarrow$ Costo ridotto minore** in corrispondenza della colonna x_2 $\rightarrow$ è x_2 la candidata a entrare in base

$\frac{4}{3}$	0	$\frac{5}{3}$	1	$-\frac{1}{3}$	0	$\frac{1}{3}$	0	$\frac{17}{3}$	$\leftarrow x_4$
$-\frac{2}{3}$	1	$-\frac{1}{3}$	0	$-\frac{1}{3}$	0	$\frac{1}{3}$	0	$\frac{5}{3}$	$\leftarrow x_2$
$\frac{1}{3}$	0	$\frac{5}{3}$	0	$-\frac{1}{3}$	-1	$\frac{1}{3}$	1	$\frac{8}{3}$	$\leftarrow x_8$
$-\frac{1}{3}$	0	$-\frac{5}{3}$	0	$\frac{1}{3}$	1	$\frac{2}{3}$	0	$-\frac{8}{3}$	

Dobbiamo reiterare finché non otteniamo un valore nullo della funz. obiettivo.

x_3 è candidata a entrare in base al posto di x_8.

1	0	0	1	0	1	0	-1	3	$\leftarrow x_4$
$-\frac{3}{5}$	1	0	0	$-\frac{2}{3}$	$-\frac{1}{5}$	$\frac{2}{5}$	$\frac{1}{5}$	$\frac{11}{5}$	$\leftarrow x_2$
$\frac{1}{5}$	0	1	0	$-\frac{1}{5}$	$-\frac{3}{5}$	$\frac{1}{5}$	$\frac{3}{5}$	$\frac{8}{5}$	$\leftarrow x_3$
0	0	0	0	0	0	1	1	0	

$\rightarrow$ Abbiamo praticamente trovato la sol. ammissibile di base che cercavamo $\rightarrow$ Possiamo eliminare le variabili x_7, x_8 e tornare così al problema originale

1	0	0	1	0	1	3	$\leftarrow x_4$
$-\frac{3}{5}$	1	0	0	$-\frac{2}{5}$	$-\frac{1}{5}$	$\frac{11}{5}$	$\leftarrow x_2$
$\frac{1}{5}$	0	1	0	$-\frac{1}{5}$	$-\frac{3}{5}$	$\frac{8}{5}$	$\leftarrow x_3$
1	-1	1	0	0	0	0	
$\frac{1}{5}$	0	0	0	$-\frac{1}{5}$	$\frac{2}{5}$	$\frac{3}{5}$	

Costo ridotto maggiore (ora stiamo massimizzando: siamo tornati al problema originale) in corrispondenza della colonna x_6
$\rightarrow$ è x_6 la candidata a entrare in base

1	0	0	1	0	1	3	$\leftarrow x_6$
$-\frac{2}{5}$	1	0	$\frac{1}{5}$	$-\frac{2}{5}$	0	$\frac{14}{5}$	$\leftarrow x_2$
$\frac{4}{5}$	0	1	$\frac{3}{5}$	$-\frac{1}{5}$	0	$\frac{17}{5}$	$\leftarrow x_3$
$-\frac{1}{5}$	0	0	$-\frac{2}{5}$	$-\frac{1}{5}$	0	$-\frac{3}{5}$	

$x_1 = x_4 = x_5 = 0$

$\rightarrow$ Abbiamo trovato la soluzione ottimale, poiché i coefficienti delle variabili nella funzione obiettivo $(-\frac{1}{5}, -\frac{2}{5}, -\frac{1}{5})$ sono tutti non positivi $\rightarrow$ Il valore massimo che può assumere la funzione obiettivo è $\frac{3}{5}$

Ora, nel caso in cui avessimo + variabili candidate a uscire dalla base, **non** possiamo sceglierne una arbitrariamente, altrimenti rischieremmo di generare dei cicli inutili all'interno dell'algoritmo del simplesso.
Si deve applicare quindi la cosiddetta REGOLA DI BLAND.

Regola di Bland:
In caso di due o più variabili candidate a entrare in base, o in caso di due o più variabili candidate a uscire dalla base, scegli **sempre** quella di indice + basso (o, alternativamente, scegli **sempre** quella di indice + alto).

L'algoritmo del simplesso è un'ulteriore prova del Teorema fondamentale della programmazione lineare: infatti, tutte le soluzioni di base ammissibili corrispondono ai vertici del poliedro che descrive graficamente il problema.

$P = \{x \subseteq \mathbb{R} \mid Ax \leq b,\ x \geq 0\}$ $\qquad A \in \mathbb{Z}^{m \times n}$

Con l'aggiunta delle variabili di slack, abbiamo che $(A, I)\,\bar{x} = b$

- m variabili "in base"
- n variabili "fuori base"

$\}$ $\bar{x}$ = soluzione di base

$$\bar{x}_i = 0 \begin{cases} \text{se } i \in \{1, \dots, n\} \rightarrow x_i = 0 \\ \text{se } i \in \{n+1, \dots, n+m\} \rightarrow A_i^t x = b \end{cases}$$

x_i fuori base

⇝ La soluzione $\bar{x}$ si trova nell'intersezione di n iperspazi di dimensione n e, quindi, è un punto (un vertice).

12/01/2021
Consideriamo il seguente problema di programmazione lineare:

$$\max\ Z = 4x_1 + x_2 + 5x_3 + 3x_4$$

$$\begin{cases} ① \quad x_1 - x_2 - x_3 + 3x_4 \leq 1 \\ ② \quad 5x_1 + x_2 + 3x_3 + 8x_4 \leq 55 \\ ③ \quad -x_1 + 2x_2 + 3x_3 - 5x_4 \leq 3 \\ \quad\ \ x_1, x_2, x_3, x_4 \geq 0 \end{cases}$$

Sia Z^* = valore max della funzione obiettivo.
Prendiamo una soluzione ammissibile del problema:
$(x_1 = 1,\ x_2 = 1,\ x_3 = 0,\ x_4 = 1/3)$ → Abbiamo trovato un lower-bound della soluzione ottima: $Z^* \geq 6$
Ma possiamo trovare anche un upper-bound per la soluzione del nostro problema?

Facciamo la somma delle disuguaglianze ①, ②, ③:

$$5x_1 + 2x_2 + 5x_3 + 6x_4 \leq 59$$

Poiché $x_1, x_2, x_3, x_4 \geq 0$, $\quad 5x_1 + 2x_2 + 5x_3 + 6x_4 \geq \overbrace{4x_1 + x_2 + 5x_3 + 3x_4}^{\text{funzione obiettivo}}$

⇒ Sicuramente 59 è un upper-bound per la nostra funzione obiettivo

⇒ $Z^* \leq 59$

Sommiamo ora le disuguaglianze ②, ③:

$4x_1 + 3x_2 + 6x_3 + 3x_4 \leq 58$; $\quad 4x_1 + 3x_2 + 6x_3 + 3x_4 \geq \overbrace{4x_1 + x_2 + 5x_3 + 3x_4}^{\text{funzione obiettivo}}$

⇒ Abbiamo trovato un upper-bound + stringente per la nostra funzione obiettivo

⇒ $Z^* \leq 58$

In entrambi i casi, abbiamo trovato una combinazione lineare tra i vincoli del problema che risultasse maggiore o uguale della funzione obiettivo $\forall x_1, x_2, x_3, x_4$.
In generale, se y_1, y_2, y_3 sono i coefficienti dei nostri vincoli, possiamo scrivere la seguente disuguaglianza ⊛.

$$y_1(x_1 - x_2 - x_3 + 3x_4) + y_2(5x_1 + x_2 + 3x_3 + 8x_4) + y_3(-x_1 + 2x_2 + 3x_3 - 5x_4) \leq$$
$$\leq 1 \cdot y_1 + 55y_2 + 3y_3$$

Se $y_1, y_2, y_3 \geq 0 \Rightarrow$ (*) è rispettata da ogni x ammissibile.

Se
$$\left.\begin{array}{l}(y_1 + 5y_2 - y_3) \geq 4 \\ (-y_1 + y_2 + 2y_3) \geq 1 \\ (-y_1 + 3y_2 + 3y_3) \geq 5 \\ (3y_1 + 8y_2 - 5y_3) \geq 3\end{array}\right] \rightarrow y_1 + 55y_2 + 3y_3 \geq z^* \geq z$$

I coefficienti delle variabili nella funz. obiettivo

Generalizziamo:

$$\max \ c^T x \qquad A \in \mathbb{Z}^{m \times n}$$
$$Ax \leq b \qquad b \in \mathbb{Z}^m$$
$$x \geq 0 \qquad c \in \mathbb{Z}^n$$

Vogliamo praticamente trovare un vettore y a m elementi tale che

$$\begin{cases} A^t y \geq c \\ y \geq 0 \end{cases} \leftarrow \text{Se } y \text{ soddisfa questi 2 vincoli} \Rightarrow b^t y \geq z^* \geq z$$

Chiaramente, un upper-bound è tanto migliore quanto è più vicino a z^*.
In pratica, vogliamo $\min b^t y$.

$\rightarrow \max c^T x$
$$\boxed{\begin{array}{l} Ax \leq b \\ x \geq 0 \end{array}} \rightarrow \text{Poliedro } P \qquad \Rightarrow \text{PROBLEMA PRIMALE } P$$

$\rightarrow \min b^t y$
$$\boxed{\begin{array}{l} A^t y \geq c \\ y \geq 0 \end{array}} \rightarrow \text{Poliedro } D \qquad \Rightarrow \text{PROBLEMA DUALE } D \quad \rightarrow D(P)$$

Sappiamo che

$$\begin{array}{c} c^t x \\ Ax \leq b \\ x \geq 0 \end{array} \leqslant \begin{array}{c} \max c^t x \\ Ax \leq b \\ x \geq 0 \end{array} \leqslant \begin{array}{c} \min b^t y \\ A^t y \geq c \\ y \geq 0 \end{array} \leqslant \begin{array}{c} b^t y \\ A^t y \geq c \\ y \geq 0 \end{array}$$

poliedri

Se $x \in P$, $y \in D$ tali che $c^T x = b^T y \Rightarrow x = x^*$, $y = y^*$

Questi due poliedri si trovano in spazi diversi, rispettivamente in $\mathbb{R}^n$, in $\mathbb{R}^m$

Si tratta delle soluz. ottime dei 2 problemi

Ma esistono casi in cui la soluzione ottima del primale sia strettam. minore della soluzione ottima duale?

<u>Teorema della dualità forte (TDF):</u>

La seguente relazione è sempre vera:

$$\begin{array}{c} \max c^t x \\ Ax \leq b \\ x \geq 0 \end{array} = \begin{array}{c} \min b^t y \\ A^t y \geq c \\ y \geq 0 \end{array}$$

<u>Dimostrazione:</u>

Sia x^* la soluzione ottima del problema primale P. Prendiamo il tableau di x^*:

Coef. delle variabili della funz. obiettivo nella soluz. ottima $\rightarrow$ $\bar{c}_1 \quad \bar{c}_2 \quad \dots \quad \bar{c}_{n+m} \mid -z_0$

$\rightarrow$ Funzione obiettivo: $z = \bar{c}^t \bar{x} + z_0$

$$z^* = \sum_{i=1}^{n+m} \bar{c}_i \bar{x}_i + z_0^*$$

$\leftarrow$ Questa relaz. vale per ogni iterazione dell'algoritmo del simplesso

$\rightarrow$ Valore della soluzione ottima: $z^* = \sum_{i=1}^{n} c_i x_i^*$

Definiamo il vettore della soluzione ottima del problema duale D nel seguente modo:

$$y_i^* = -\bar{c}_{n+i} \quad \forall i = 1, \dots, m \quad \Rightarrow y_i^* \geq 0 \quad \forall i = 1, \dots, m$$

$$\sum_{i=1}^{n} c_i x_i^* = z_0^* + \sum_{i=1}^{n+m} \bar{c}_i \bar{x}_i = z_0^* + \sum_{i=1}^{n} \bar{c}_i \bar{x}_i + \sum_{i=n+1}^{n+m} \bar{c}_i \bar{x}_i =$$

$$= z_0^* + \sum_{i=1}^{n} \bar{c}_i \bar{x}_i + \sum_{i=1}^{m} (-y_i^*)(\bar{x}_{i+n}) = z_0^* + \sum_{i=1}^{n} \bar{c}_i \bar{x}_i - \sum_{i=1}^{m} y_i^* \left(b_i - \sum_{j=1}^{n} A_{ij} x_j \right) =$$

$$= z_0^* - \sum_{i=1}^{m} b_i y_i^* + \sum_{i=1}^{n} \bar{c}_i x_i + \sum_{i=1}^{m} y_i^* \sum_{j=1}^{n} A_j x_j =$$

$$= \left(z_0^* - \sum_{i=1}^{m} b_i y_i^* \right) + \sum_{i=1}^{n} \left(\bar{c}_i + \sum_{j=1}^{m} y^* A_{ji} \right) x_i = \sum_{i=1}^{n} c_i x_i$$

$$\Rightarrow \begin{cases} z_0^* - \sum_{i=1}^{m} b_i y_i^* = 0 \\ c_i = \bar{c}_i + \sum_{j=1}^{m} A_{ji} y_j^* \quad \forall i = 1, \dots, n \end{cases}$$

Nella soluzione ottima del tableau: $\bar{c}_i \leq 0 \; \forall i \Rightarrow c_i - \sum_{j=1}^{m} A_{ji} y_j^* \leq 0 \quad \forall i = 1, \dots, n$

$\Rightarrow \sum_{j=1}^{m} A_{ji} y_j^* \geq c_i \quad \forall i = 1, \dots, n \quad \Leftrightarrow \quad A^t y^* \geq c$

$\rightsquigarrow b^T y^* = z^* = c^T x^* \Rightarrow \exists$ sempre la soluzione ottima di $\mathcal{D}$ di pari valore della soluzione ottima di $\mathcal{P}$. ■

13/01/2021

$\mathcal{P}$ \ $\mathcal{D}(\mathcal{P})$	$\exists$ soluzione ottima y^*	$\mathcal{D}$ è illimitato inferiorm.	$\mathcal{D} = \emptyset$
$\exists$ soluzione ottima x^*	✓	✗	✗
$\mathcal{P}$ è illimitato superiorm.	✗	✗	✓
$P = \emptyset$	✗	✓	✓

✓ = POSSIBILE
✗ = IMPOSSIBILE

Le variabili x_j di $\mathcal{P}$ corrispondono ai vincoli $(A^t y \geq c)$ di $\mathcal{D}(\mathcal{P})$, così come i vincoli $(Ax \leq b)$ di $\mathcal{P}$ corrispondono alle variabili y_i di $\mathcal{D}(\mathcal{P})$.

$$A_{i.}x \leq b_i \Rightarrow y_i \geq 0 \qquad\qquad x_j \geq 0 \Rightarrow (A_{.j})^t y \geq c_j$$

$$A_{i.}x \geq b_i \Rightarrow y_i \leq 0 \qquad\qquad x_j \leq 0 \Rightarrow (A_{.j})^t y \leq c_j$$

$$\underbrace{A_{i.}}x = b_i \Rightarrow y_i \lesseqgtr 0 \qquad\qquad x_j \lesseqgtr 0 \Rightarrow (\underbrace{A_{.j}})^t y = c_j$$

$A_{i.}$ = i-esima riga della matrice A

$A_{.j}$ = j-esima colonna della matrice A

Teorema:

$$\mathcal{D}(\mathcal{D}(\mathcal{P})) = \mathcal{P}$$

Tra $\mathcal{P}$ e $\mathcal{D}$ valgono le cosiddette relazioni di COMPLEMENTARY SLACKNESS:

$$x_j^* \left[(A_{.j})^t y^* - c_j \right] = 0 \qquad \forall j = 1, \dots, n$$

$$y_i^* \left[A_{i.} x^* - b_i \right] = 0 \qquad \forall i = 1, \dots, m$$

Infatti: $$c^t x^* = \sum_{j=1}^{n} c_j x_j^* \leq \sum_{j=1}^{n} \underbrace{\left(\sum_{i=1}^{m} A_{ij} y_i^* \right)}_{\geq c_j} \overset{\geq 0}{x_j^*} = \sum_{i=1}^{m} \underbrace{\left(\sum_{j=1}^{n} A_{ij} x_j^* \right)}_{\leq b_i} y_i^* \leq$$

$$\leq \sum_{i=1}^{m} b_i y_i^* = b^t y^*$$

↳ Per il Thm della dualità forte, $c^t = b^t$ ⇒ la catena di disuguaglianze che abbiamo appena ottenuto può essere scritta in realtà come una catena di uguaglianze.

① $$\begin{cases} x_j^* = 0 \\ c_j = \sum_{i=1}^{m} A_{ij} y_i^* \end{cases} \Rightarrow x_j^* \left(\sum_{i=1}^{m} A_{ij} y_i^* - c_j \right) = 0 \qquad \forall j = 1, \dots, n$$

② $$\begin{cases} y_i^* = 0 \\ b_i = \sum_{j=1}^{n} A_{ij} x_j^* \end{cases} \Rightarrow y_i^* \left(\sum_{j=1}^{n} A_{ij} x^* - b_i \right) = 0 \qquad \forall i = 1, \dots, m$$

Riconsideriamo ora il seguente problema:

$$\max \ 5x_1 + 4x_2 + 3x_3$$

$$\begin{cases} 2x_1 + 3x_2 + x_3 \leq 5 \\ 4x_1 + x_2 + 2x_3 \leq 11 \\ 3x_1 + 4x_2 + 2x_3 \leq 8 \\ x_1, x_2, x_3 \geq 0 \end{cases} \longrightarrow \text{Abbiamo visto che la soluzione ottima è data dal vettore } x^* = \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix}$$

Scriviamo ora il corrispondente problema duale:

$$\min \ 5y_1 + 11y_2 + 8y_3$$

$$\begin{cases} 2y_1 + 4y_2 + 3y_3 \geq 5 \\ 3y_1 + y_2 + 4y_3 \geq 4 \\ y_1 + 2y_2 + 2y_3 \geq 3 \\ y_1, y_2, y_3 \geq 0 \end{cases}$$

Per i vincoli di complementary slackness:

$$y_2^* = 0 \quad \leftarrow \text{Poichè } 4x_1^* + x_2^* + 2x_3^* < 11$$

$$2y_1^* + 4y_2^* + 3y_3^* = 5$$

$$y_1^* + 2y_2^* + 2y_3^* = 3 \quad \Longrightarrow \begin{cases} 6 - 4y_3^* + 3y_3^* = 5 \\ y_1^* = 3 - 2y_3^* \end{cases} \Longrightarrow \begin{cases} y_1^* = 1 \\ y_3^* = 1 \end{cases}$$

$$3y_1^* + y_2^* + 4y_3^* \geq 4$$

$$y_1^*, y_2^*, y_3^* \geq 0$$

$$\Downarrow$$

$$y^* = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

<u>NOTAZIONE:</u>

Consideriamo il tableau iniziale e il tableau finale (relativo alla soluzione ottima) di un problema $\mathcal{P}$:

$$\underbrace{\begin{array}{c:c|c} A & I & b \\ \hline c & 0 & 0 \end{array}}_{\text{MATRICE } \mathcal{H} \in \mathbb{R}^{m \times (m+n)}} \xrightarrow[\text{ottengo}]{\text{Se moltiplico per una matrice quadrata } Q} \underbrace{\begin{array}{c:c|c} A' & \underset{Q \in \mathbb{R}^{m\times m}}{Q} & b' \\ \hline \multicolumn{2}{c|}{\bar{c}} & -z_0^* \end{array}}_{\text{MATRICE } \mathcal{H}' \in \mathbb{R}^{m \times (m+n)}}$$

Nel tableau finale, le colonne con indice in $\mathcal{B}$ definiscono la matrice d'identità

Sia $\mathcal{B} \subseteq \{1, \ldots, m+n\}$ l'insieme ordinato degli indici delle variabili in base nella soluzione ottima $\Rightarrow |\mathcal{B}| = m$.

Sia B la sottomatrice di A definita dalle colonne con indice in $\mathcal{B}$
$\Rightarrow B = \mathbb{R}^{m \times m}$.

$$QB = I \implies Q = B^{-1}$$

VARIABILI NELLA SOLUZIONE OTTIMA : $\bar{x}^* \begin{cases} \bar{x}_B^* = b' = B^{-1} b & \leftarrow \text{IN BASE} \\ \bar{x}_N = 0 & \leftarrow \text{NON IN BASE} \end{cases}$

⇝ Vedere esempio pratico nella videolezione (video (5)).

14/01/2021

Definizione:

Una matrice $M \in \mathbb{Z}^{m \times n}$ è totalmente unimodulare (TUM) $\iff$ i determinanti di tutte le sue sottomatrici quadrate (↳ incluse quelle 1×1) hanno valore $+1, -1, 0$.

↳ ciò implica che tutte le componenti di M possono assumere solo i valori $1, -1, 0$.

$$M \in \{-1, 0, 1\}^{m \times n}$$

Esempio:

$$M = \begin{bmatrix} -1 & 0 & 1 & -1 \\ 1 & 1 & 0 & 1 \\ -1 & -1 & 1 & -1 \\ 0 & 1 & -1 & 1 \end{bmatrix}$$

M è totalmente unimodulare?

→ È possibile calcolare i determinanti di tutte le sottomatrici quadrate di M. Ma ~~quant~~ qual è il costo temporale di questa operazione?

- \# di sottomatrici di ordine $K \leq \min(m, n) = \sum_{K=1}^{\min\{m,n\}} \binom{m}{K}\binom{n}{K}$

Nel nostro esempio: $\binom{4}{1}\binom{4}{1} + \binom{4}{2}\binom{4}{2} + \binom{4}{3}\binom{4}{3} + \binom{4}{4}\binom{4}{4} = 73$

↳ Si tratta di un algoritmo inefficiente

Teorema:

$M \in \{0, -1, 1\}^{m \times n}$ è TUM $\Longleftrightarrow$ Sono TUM:

- M^t
- La matrice ottenuta da M aggiungendo/cancellando righe/colonne di I.
- La matrice ottenuta da M moltiplicando una riga/colonna per -1.
- La matrice ottenuta da M moltiplicando copie di righe/colonne.
- La matrice ottenuta da M permutando righe/colonne.

Vediamo perché ci interessa lo studio delle matrici unimodulari.

Consideriamo il poliedro relativo a un problema di programmazione lineare:

$$P = \{x \in \mathbb{R}^n \mid Ax \leq b\}, \quad A \in \mathbb{Z}^{m \times n}, \quad b \in \mathbb{Z}^m$$

$$(A, I) = b$$

COMPONENTI IN BASE: $x_B = B^{-1} b$

COMPONENTI NON IN BASE: $x_N = 0$

$$B \in \mathbb{Z}^{m \times m} \text{ sottomatrice di } (A, I) \in \mathbb{Z}^{m \times (m+n)}$$

$$(B^{-1})_{ij} = (\pm 1) \frac{\det(\tilde{B}_{ij})^{\in \mathbb{Z}}}{\det(B)}$$

$\leadsto$ Se A è TUM $\xrightarrow[B \text{ invertibile}]{}$ $\det(B) \in \{-1, 1\} \Rightarrow (B^{-1})_{ij} \in \mathbb{Z} \Rightarrow B^{-1} \in \mathbb{Z}^{m \times m}$

$\Rightarrow x_B = B^{-1} b \in \mathbb{Z}^m, \; x_N = 0 \Rightarrow x^* \in \mathbb{Z}^n$

Teorema 1:

$M \in \{-1, 0, 1\}^{m \times n}$ è TUM $\Longleftrightarrow$ $\forall$ sottoinsieme R delle righe di M $\exists$ una bipartizione R^+, R^- t.c.

$$\sum_{i \in R^+} M_{ij} - \sum_{i \in R^-} M_{ij} \in \{0, 1, -1\} \text{ per ogni colonna } j \text{ di } M$$

Riprendiamo la matrice dell'esempio precedente:

$$M = \begin{bmatrix} -1 & 0 & 1 & -1 \\ 1 & 1 & 0 & 1 \\ -1 & -1 & 1 & -1 \\ 0 & 1 & -1 & 1 \end{bmatrix}$$

I casi da studiare sono:

$R=\{1,2,3,4\}$ $\quad R=\{1,2,3\}$ $\quad R=\{1,2,4\}$ $\quad R=\{1,3,4\}$

$R=\{2,3,4\}$ $\quad R=\{1,2\}$ $\quad R=\{1,3\}$ $\quad R=\{1,4\}$

$R=\{2,3\}$ $\quad R=\{2,4\}$ $\quad R=\{3,4\}$ $\quad R=\{1\}$

$R=\{2\}$ $\quad R=\{3\}$ $\quad R=\{4\}$

1) $R=\{1,2,3,4\} \longrightarrow \begin{bmatrix} -1 & 0 & 1 & -1 \\ 1 & 1 & 0 & 1 \\ -1 & -1 & 1 & -1 \\ 0 & 1 & -1 & 1 \end{bmatrix}$ $\begin{array}{l} \leftarrow \text{in } R^+ \\ \leftarrow \text{in } R^- \\ \leftarrow \text{in } R^- \\ \leftarrow \text{in } R^+ \end{array}$ ✓

2) $R=\{1,2,3\} \longrightarrow \begin{bmatrix} -1 & 0 & 1 & -1 \\ 1 & 1 & 0 & 1 \\ -1 & -1 & 1 & -1 \end{bmatrix}$ $\begin{array}{l} \leftarrow \text{in } R^+ \\ \leftarrow \text{in } R^- \\ \leftarrow \text{in } R^- \end{array}$ ✓

3) $R=\{1,2,4\} \longrightarrow \begin{bmatrix} -1 & 0 & 1 & -1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & -1 & 1 \end{bmatrix}$ $\leftarrow \nexists$ bipartizione lecita ✗

$\Longrightarrow$ M <u>NON</u> è TUM.

<u>NB</u>: Per i sottoinsiemi R composti da un'unica riga la condizione

$$\sum_{i \in R^+} M_{ij} - \sum_{i \in R^-} M_{ij} \in \{0,1,-1\} \quad \forall j \quad \text{e per qualche bipartiz. } R^+, R^-.$$

CONSIGLIO: Per matrici con tante righe e poche colonne conviene applicare l'algoritmo alla matrice trasposta.

<u>Teorema 1 bis:</u>

La matrice $M \in \{0,1,-1\}^{m \times n}$ che ha al più due elementi <u>diversi</u> da 0 per ogni colonna è TUM $\Longleftrightarrow$ la proprietà descritta nel Teorema 1 è verificata per $R = \{1, \dots, m\}$ ← TUTTE LE RIGHE

CONSIGLIO: Per matrici con al più due elementi diversi da 0 per ogni riga conviene applicare l'algoritmo alla matrice trasposta.

⟶ Le matrici di incidenza sono un valido esempio di matrici con al più due elementi diversi da 0 per ogni colonna.

IN PARTICOLARE: Se M è la matrice di incidenza di un grafo diretto $\Rightarrow M$ è TUM.

Esempio (problema di max flusso):

VINCOLI:

$Mx = b$ ⟵ vincoli di bilancio ai nodi

$Ix \leq c$ ⟵ vincoli di capacità

$Ix \geq 0$

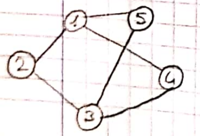
$$\Rightarrow \begin{cases} Mx \leq b \\ -Mx \leq -b \\ Ix \leq c \\ -Ix \leq 0 \end{cases}$$

è TUM

Poliedro P: $$\begin{pmatrix} M \\ I \\ -I \\ -M \end{pmatrix} x \leq b'$$

Supponiamo ora che M sia la matrice di incidenza di un grafo $G = (V,E)$ non orientato.

$$M \in \{0,1\}^{m \times n} \quad \text{con } m = |V|,\ n = |E|$$



$$M = \begin{array}{cccccc|c} 1 & 1 & 1 & 0 & 0 & 0 & ① \\ 1 & 0 & 0 & 1 & 0 & 0 & ② \\ 0 & 0 & 0 & 1 & 1 & 1 & ③ \\ 0 & 1 & 0 & 0 & 1 & 0 & ④ \\ 0 & 0 & 1 & 0 & 0 & 1 & ⑤ \\ \{1,2\} & \{1,4\} & \{1,5\} & \{2,3\} & \{3,4\} & \{3,5\} & \end{array}$$

M è TUM $\Longleftrightarrow$ G è bipartito.

Inoltre, se G è bipartito, vale la seguente uguaglianza:

$$\begin{array}{l} \max \mathbb{1}^t x \\ Hx \leq \mathbb{1} \\ x \in \{0,1\}^{|E|} \end{array} = \begin{array}{l} \max \mathbb{1}^t x \\ Hx \leq \mathbb{1} \\ x \geq 0 \end{array}$$

E anche la seguente uguaglianza:

$$\begin{array}{l} \min \mathbb{1}^t y \\ H^t y \geq \mathbb{1} \\ y \geq 0 \end{array} = \begin{array}{l} \min \mathbb{1}^t y \\ H^t y \geq 1 \\ y \in \{0,1\}^{|V|} \end{array}$$

Anche questi 2 problemi hanno lo stesso valore, dato che sono DUALI

18/01/2021

Esercizio 1:

$$\max \; 2x_1 + 4x_2 + 3x_3 + x_4$$

$$\begin{cases} 3x_1 + x_2 + x_3 + x_4 \leq 12 \\ x_1 - 3x_2 + 2x_3 + 3x_4 \leq 7 \\ 2x_1 + x_2 + 3x_3 - x_4 = 10 \\ x_1, x_3, x_4 \geq 0 \end{cases}$$

Prendiamo le due soluzioni: $\tilde{x} = \begin{bmatrix} 1 \\ 2 \\ 2 \\ 0 \end{bmatrix}$, $\bar{x} = \begin{bmatrix} 0 \\ 52/5 \\ 0 \\ 2/5 \end{bmatrix}$

Utilizzando le relazioni di complementary slackness, cosa si può dire delle due soluzioni?

Iniziamo con lo scrivere il problema duale (e col notare che sicuramente $\tilde{x}, \bar{x}$ sono ammissibili):

Scansionato con CamScanner

$$\min \quad 12y_1 + 7y_2 + 10y_3$$

$$\begin{cases} 3y_1 + 2y_2 + 2y_3 \geq 2 \\ y_1 - 3y_2 + y_3 = 4 \\ y_1 + 2y_2 + 3y_3 \geq 3 \\ 4y_1 + 3y_1 - y_3 \geq 1 \\ y_1, y_2 \geq 0 \end{cases}$$

$\boxed{\tilde{x}}$

$3y_1^* + 2y_2^* + 2y_3^* = 2$ $\longrightarrow$ NON SODDISFATTA $\Rightarrow y_1^* = 0$

$y_1^* - 3y_2^* + 3y_3^* = 4$ $\longrightarrow$ NON SODDISFATTA $\Rightarrow y_2^* = 0$

$y_1^* + 2y_1^* + 3y_3^* = 3$

$4y_1^* + 3y_2^* - y_3^* \geq 1$

$y_1^*, y_2^* \geq 0$

$\Downarrow$

Dalla prima equazione:

$2y_3^* = 2 \Rightarrow y_3^* = 1$

$\Rightarrow$ NON È COMPATIBILE CON LE ALTRE CONDIZIONI $\Rightarrow$ $\tilde{x}$ NON È OTTIMA

$\boxed{\bar{x}}$

$3y_1^* + 2y_2^* + 2y_3^* \geq 2$

$y_1^* - 3y_2^* + y_3^* = 4$ $\longrightarrow$ NON SODDISFATTA $\Rightarrow y_2^* = 0$

$y_1^* + 2y_2^* + 3y_3^* \geq 3$

$4y_1^* + 3y_2^* - y_3^* = 1$

$y_1^*, y_2^* \geq 0$

$\Downarrow$

Dalle altre equazioni:

$$\begin{matrix} y_1^* + y_3^* = 4 \\ 4y_1 - y_3^* = 1 \end{matrix} \Rightarrow \begin{cases} y_1^* = 1 \\ y_3^* = 3 \end{cases}$$

$\Rightarrow y^* = (1, 0, 3) \rightsquigarrow$ Soluzione ammissibile

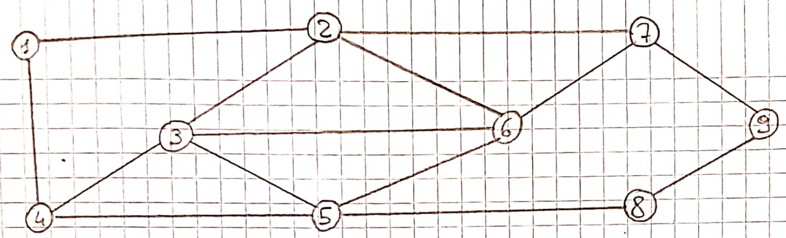
$\Downarrow$

$\bar{x}$ È OTTIMA

Esercizio 2:

Consideriamo una pinacoteca con la seguente pianta:

- VERTICI = salette
- SPIGOLI = corridoi



Scrivere un problema PLI per decidere in quali salette mettere un guardiano in modo tale che tutti i corridoi siano controllati e minimizzando il numero di guardiani assunti (un guardiano in una saletta può controllare i corridoi incidenti a quella saletta).

FUNZIONE OBIETTIVO:

$$\min \sum_{i=1}^{9} x_i$$

VARIABILI:

$$x_i = \begin{cases} 1 & \text{se assegno un guardiano alla sala } i \\ 0 & \text{altrimenti} \end{cases}$$

VINCOLI:

$$x_i + x_j \geq 1 \qquad \forall \{i,j\} \in E$$
$$x_i \in \{0,1\} \qquad \forall i = 1, \dots, 9$$

→ Se prendiamo i nodi 1, 2, 3, 5, 6, 9 otteniamo una soluzione ammissibile
⟹ 6 è un upper-bound.

Se volessimo trovare un lower-bound del problema, potremmo considerare il problema duale di questo (= del vertex cover), ovvero il problema della massima cardinalità del matching del grafo. In particolare, per quello che abbiamo detto la volta scorsa:

MAX CARDINALITÀ MATCHING ≤ MIN CARDINALITÀ VERTEX COVER

→ ∃ ~~cardinalità~~ matching di cardinalità 4 nel grafo dell'esercizio
⟹ 4 è un lower-bound del problema.

Esercizio 3:

$$M = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & -1 \\ 1 & 0 & 1 & 1 & 1 & -1 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix} \quad \text{è TUM?}$$

$$M' = \begin{bmatrix} 1 & 1 & 0 & 0 & -1 \\ 1 & 0 & 1 & 1 & -1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

$\rightarrow R = \{1,2,3,4\}$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & -1 \\ 1 & 0 & 1 & 1 & -1 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{bmatrix} \begin{matrix} + \\ + \\ - \\ - \end{matrix} \rightsquigarrow \checkmark$$

$\rightarrow R = \{1,2,3\}$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & -1 \\ 1 & 0 & 1 & 1 & -1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix} \times \quad \nexists \text{ bipartizione lecita } R^+, R^-$$

$\Downarrow$

M NON è TUM $\Leftarrow$ M' NON è TUM

Esercizio 4:

$$\max \; 3x_1 + 4x_2 + \alpha x_3$$

$$\begin{cases} x_1 + x_2 + x_3 \leq 2 \\ 2x_1 + 3x_2 + x_3 \leq 3 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

Per quali valori di α la soluzione $x^* = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$ NON è ottima?

Possiamo utilizzare le relazioni di complementary slackness; scriviamo il duale del problema dato:

$\hookrightarrow$ PAG. SUCCESSIVA

$\min \; 2y_1 + 3y_2$

$$\begin{cases} y_1 + 2y_2 \geq 3 \\ y_1 + 3y_2 \geq 4 \\ y_1 + y_2 \geq \alpha \\ y_1, y_2 \geq 0 \end{cases}$$

$x_1^* \neq 0 \Rightarrow y_1^* + 2y_2^* = 3$

$x_2^* = 0 \Rightarrow y_1^* + 3y_2^* \geq 4$

$x_3^* \neq 0 \Rightarrow y_1^* + y_2^* = \alpha$

RISOLVO 1° E 3° EQUAZ.

$$\begin{cases} y_1^* = 2\alpha - 3 \Rightarrow 2\alpha - 3 \Rightarrow \alpha \geq 3/2 \\ y_2^* = 3 - \alpha \Rightarrow 3 - \alpha \geq 0 \Rightarrow \alpha \leq 3 \end{cases}$$

SOSTITUISCO NELLA SECONDA EQUAZ.

$2\alpha - 3 + 9 - 3\alpha \geq 4 \Rightarrow \alpha \leq 2$

In definitiva: $\alpha \geq 3/2 \wedge \alpha \leq 3 \wedge \alpha \leq 2$

$\Rightarrow$ Il primale ha x^* come soluzione ottima per $3/2 \leq \alpha \leq 2$

$\Rightarrow x^*$ <u>NON</u> è ottima per $\alpha > 2$, $\alpha < 3/2$.