

Appunti di Ricerca Operativa

A.A. 2024/2025

Tommaso Renzoni

Parte 1. Programmazione Lineare (PL)

Problema di produzione	1
Formati standard	2
Regione ammissibile e soluzione ottima	3
Problema di assegnamento	4
Problema di trasporto	6
Teoria della PL	7
Teorema di rappresentazione dei poliedri	9
Teorema fondamentale della PL	10
Calcolo dei vertici	12
Trasformazioni equivalenti	14
Teoria della dualità	15
Algoritmo del simplex	17
Algoritmo del simplex primale	18
Algoritmo del simplex duale	20
Problema duale ausiliario	21
Il caso degenere	22

Parte 2. Programmazione Lineare Intera (PLI)

Problema dello zaino	24
Valutazione superiore dello zaino intero	25
Valutazione superiore dello zaino booleano	26
Valutazioni inferiori e superiori dello zaino intero e booleano	26
Problemi particolari	27
Problema di trasporto e assegnamento interi	27
Problema del bin packing	29
Valutazione inferiore	29
Valutazione superiore	30
Algoritmo di riduzione del gap	30
Risultati teorici	30
Piani di taglio	31
Algoritmo di riduzione del gap	32
Problema del commesso viaggiatore (TSP)	37
TSP asimmetrico	37
TSP simmetrico	40
Algoritmo del Branch and Bound	42
Albero di enumerazione totale	42
Regole di taglio e algoritmo del branch and bound	43
Problema di copertura	46
Copertura di costo minimo	46
massima copertura	49

Parte 3. Programmazione Lineare su Reti

Flusso di costo minimo non capacitato	50
Come costruire un flusso di base	52
Reti non bilanciate	53
Calcolo dei potenziali e test di ottimalità	53
Simplex per flusso di costo minimo non capacitato	56
Flusso di costo minimo come classe di problemi	59
Flusso multiobiettivo	61
Cammini minimi	61
Simplex per cammini minimi	62
Algoritmo di Dijkstra	64
Flusso di costo minimo capacitato	66
Come costruire un flusso di base	67
Calcolo dei potenziali e test di ottimalità	70
Simplex per flusso di costo minimo capacitato	72
Flusso massimo	75
Algoritmo di Ford Fulkerson	77
Accoppiamento di cardinalità massima	81

Parte 4. Programmazione Non Lineare (PNL)

Un primo problema di PNL	83
Prerequisiti di Analisi II	84
Derivate parziali, gradiente, hessiana	84
Teoremi per minimi e massimi	84
Derivate direzionali e restrizioni	87
Classi di funzioni	87
Funzioni quadratiche	88
Funzioni convesse	89
Funzioni coercive	90
Domini della PNL	90
Domini compatti	91
Domini convessi	92
Domini regolari	92
Condizioni di ottimalità del primo ordine	94
Regola dei moltiplicatori di Lagrange (Analisi II)	94
Condizione necessaria ai minimi o massimi	95
Condizione sufficiente ai minimi o massimi	100
Analisi locale dei punti stazionari	101
Metodi per l'ottimizzazione libera	103
Esistenza di massimi e minimi	103
Successione iterativa e direzione di salita	103
Algoritmi iterativi per massimi	104
Metodi per l'ottimizzazione vincolata	109
Metodo di Frank-Wolfe	109
Metodo del gradiente proiettato	111

Programmazione Lineare (PL)

La programmazione lineare (PL) è quella branca della ricerca operativa che si occupa di studiare algoritmi di risoluzione per problemi di ottimizzazione lineari.

1. Problema di produzione

Esempio 1.1 una ditta produce LaminatoA, LaminatoB. Per la produzione, si deve passare dai reparti Materie Prime, Taglio, FinituraA, FinituraB. Il guadagno è di €8.400 e €11.200 rispettivamente. Ogni reparto impiega un certo numero di ore, vincolate secondo la seguente tabella.

Reparto	Vincolo	LaminatoA	LaminatoB
Materie Prime	120h	30h	20h
Taglio	80h	10h	20h
FinituraA	62h	20h	
FinituraB	105h		30h

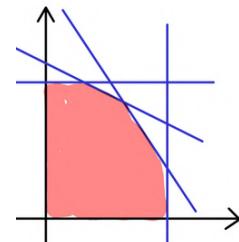
Abbiamo una **funzione obiettivo** da massimizzare, e dei **vincoli**. Una **regione ammissibile** è l'insieme delle **soluzioni ammissibili** i.e. le soluzioni che rispettano i vincoli. La **soluzione ottima** è, tra le funzioni ammissibili, quella che massimizza la funzione obiettivo.

Modello matematico:

$$\begin{cases} \max 8.4x_A + 11.2x_B, & f: \mathbb{R}^2 \rightarrow \mathbb{R} \\ 30x_A + 20x_B \leq 120 \\ 10x_A + 20x_B \leq 80 \\ 20x_A \leq 62 \\ 30x_B \leq 105 \end{cases}$$

Siano $A = \begin{bmatrix} 30 & 20 \\ 10 & 20 \\ 20 & 0 \\ 0 & 30 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}$, $b = \begin{bmatrix} 120 \\ 80 \\ 62 \\ 105 \\ 0 \\ 0 \end{bmatrix}$, $c = [8.4 \ 11.2] \therefore (p) = \begin{cases} \max c^T \cdot x \\ A \cdot x \leq b \end{cases}$

Abbiamo aggiunto il vincolo $(x_A, x_B) \geq 0 \Leftrightarrow -x_A \leq 0, -x_B \leq 0$



La regione ammissibile è un'intersezione di semipiani i.e. un poliedro in \mathbb{R}^2

Modello.

Problema di produzione:

Problema di ottimizzazione in cui si devono allocare risorse limitate (materie prime, tempo, forza lavoro...) per massimizzare il guadagno, o minimizzare i costi, rispettando una serie di vincoli (capacità produttiva, disponibilità di risorse, domanda di mercato...)

$$(p) \begin{cases} \max c^T x \\ Ax \leq b \\ x \geq 0 \end{cases} \text{ massimizza il guadagno} \quad (q) \begin{cases} \min c^T x \\ Ax \geq b \\ x \geq 0 \end{cases} \text{ minimizza il costo}$$

1.1. Formati standard

Un problema di programmazione lineare (PL) è un problema di min / max di una funzione lineare soggetta a vincoli lineari. Un **formato standard** è un formato che racchiude tutti i problemi.

$$(p) = \begin{cases} \min / \max c^T \cdot x \\ Ax \leq b \\ Bx \geq d \\ Cx = e \end{cases}$$

Definizione.

Definizione (formato primale standard):

Un problema di PL in **formato primale standard** è $(p) = \begin{cases} \max c^T \cdot x \\ Ax \leq b \end{cases}$

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

Siccome è un formato primale **standard**, vuol dire che può risolvere tutti i problemi di programmazione lineare. Per trasformare un problema di PL da un formato standard all'altro si devono usare alcuni "trucchi":

- $\geq \leftrightarrow \leq$ moltiplicando per -1
- $= \leftrightarrow \leq$ imponendo che sia \leq , \geq contemporaneamente
- $\max \leftrightarrow \min$ poiché $\max f = -\min -f$

In seguito ne vedremo anche altri due:

- $\leq \rightarrow =, \geq$
- $x \geq 0 \rightarrow x_i = x_{i+} - x_i^-, x_{i+} \geq x_i^+ \geq 0$

1.2. Regione ammissibile e soluzione ottima

La regione ammissibile di un problema di PL è un poliedro P , definibile in due modi:

1. **Poliedro algebrico:** $Ax \leq b$
2. **Poliedro geometrico:** intersezione di un numero finito di semispazi chiusi i.e. il cui bordo appartiene ai vincoli (\leq)

Osservo che $\nabla(c \cdot x) = c \neq 0$. Quindi, per il teorema di Fermat, i punti interni non possono essere di massimo. Per il teorema di Weierstrass, la soluzione, se esiste, deve trovarsi sul bordo.

Proposizione.

- $(p) = +\infty$, se il poliedro è illimitato. Vuol dire che ci siamo dimenticati dei vincoli
- $P = \emptyset$ i.e. $\max c \cdot x = -\infty \Rightarrow$ il poliedro è vuoto, ci sono troppi vincoli. In tal caso si può fare un ranking dei vincoli ed eliminare quello meno importante

Per trovare la soluzione ottima graficamente si usano le **linee di isoguadagno**, che hanno equazione $c \cdot x = k$

Il vettore c è il gradiente di $c \cdot x$, ovvero la direzione di massima crescenza della funzione obiettivo. Devo quindi trovare l'ultimo punto prima che la retta di isoguadagno esca dal poliedro.

Attenzione: non è vero in generale che la soluzione ottima sta sul vertice. Può accadere che c sia perpendicolare allo spigolo; in tal caso ci sono infinite soluzioni, poiché la linea di isoguadagno passa per tutto lo spigolo. Il **cono di competenza** indica il range di vettori c tali per cui un certo vertice è soluzione ottima.

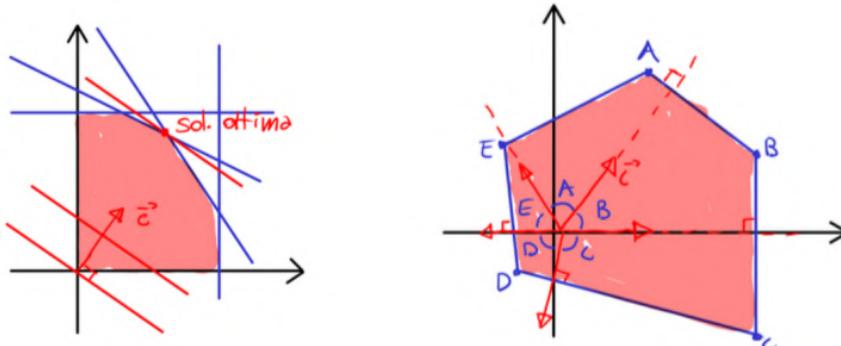


Figura 1: le linee di isoguadagno sono perpendicolari a \vec{c} .

Figura 2: coni di competenza. Se, per esempio, il vettore \vec{c} si trova nell'angolo β , lo spigolo B è la soluzione ottima.

Proposizione.

Se ci sono due soluzioni ottime, ci sono infinite soluzioni ottime.

Dimostrazione:

Siano z, w soluzioni ottime. Il segmento che le congiunge ha equazione $\lambda z + (1 - \lambda)w$, $\lambda \in [0, 1]$. La funzione obiettivo del segmento è il segmento stesso poiché $c^T(\lambda z + (1 - \lambda)w) = \lambda c^T z + (1 - \lambda)c^T w = \lambda v + (1 - \lambda)v = \vec{v}$. Quindi tutto il segmento è fatto di soluzioni ottime. \square

2. Problema di assegnamento

Esempio 2.1 n -persone, n -lavori, ogni persona ha un costo per ogni lavoro. Voglio fare un'associazione biunivoca che minimizza il costo. Gli assegnamenti diversi sono le permutazioni $n!$, non posso andare di forza bruta.

	Lavoro 1	Lavoro 2	Lavoro 3	Lavoro 4
Persona 1	20	18	16	14
Persona 2	22	16	19	15
Persona 3	21	17	15	23
Persona 4	19	18	14	24

Modello matematico:

$$x_{ij} = \begin{cases} 0, & \text{la persona non lavora} \\ 1, & \text{la persona lavora} \end{cases}, \quad X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{bmatrix}$$

Nei vincoli devo mettere che ogni lavoro deve essere svolto esattamente da una persona (primi 4), che ogni persona deve svolgere esattamente un lavoro (secondi 4), e infine che $x_{ij} \geq 0$

$$(p) = \begin{cases} \min c^T x = \min 20x_{11} + 18x_{12} + \dots + 24x_{44} \\ 1. x_{11} + x_{12} + x_{13} + x_{14} = 1 \quad 5. x_{11} + x_{21} + x_{31} + x_{41} = 1 \\ 2. x_{21} + x_{22} + x_{23} + x_{24} = 1 \quad 6. x_{12} + x_{22} + x_{32} + x_{42} = 1 \\ 3. x_{31} + x_{32} + x_{33} + x_{34} = 1 \quad 7. x_{13} + x_{23} + x_{33} + x_{43} = 1 \\ 4. x_{41} + x_{42} + x_{43} + x_{44} = 1 \quad 8. x_{14} + x_{24} + x_{34} + x_{44} = 1 \\ 9. x_{ij} \geq 0 \quad \forall i, j \in [0, 4] \end{cases}$$

x_{ij} potrebbe essere compreso in $]0, 1[$, per cui si ha un **assegnamento cooperativo**. Se si vogliono solo x_{ij} booleani, si ha un **assegnamento non cooperativo** (non è più PL)

Modello.

Problema di assegnamento:

Il problema di assegnamento è un problema di ottimizzazione combinatoria in cui si cerca di assegnare un insieme di risorse o lavoratori a un insieme di compiti o attività in modo da minimizzare i costi totali.

$$\begin{cases} \min c^T x \\ Ax = b \quad \text{dove } c, x \in \mathbb{R}^{n^2}, A \in \mathbb{R}^{2n \times n}, b \in \mathbb{R}^{2n} \\ x \geq 0 \end{cases}$$

Definizione.

Definizione (formato duale standard):

$$\text{Un problema di PL in formato duale standard è } (p) = \begin{cases} \min c^T \cdot x \\ Ax = b \\ x \geq 0 \end{cases}$$

dove $c, x \in \mathbb{R}^{n^2}, A \in \mathbb{R}^{2n \times n}, b \in \mathbb{R}^{2n}$

"Trucco" dello slack:

Alcuni programmi potrebbero non supportare solo i formati duali standard, per cui bisogna aggiungere il "trucco" n.4, che ci permette di trasformare $\leq \rightarrow =, \geq 0$

$$3x_1 + x_2 \leq 5 \equiv \begin{cases} 3x_1 + x_2 + Slack = 5 \\ Slack \geq 0 \end{cases}$$

Esempio 2.2 $\begin{cases} \max c^T x \\ Ax = b \end{cases}$ è in formato standard?

Se è riconducibile a un formato primale standard, allora è standard (risolve tutti i problemi di PL). Non è standard poiché lo slack dev'essere positivo.

Esempio 2.3 $\begin{cases} \min 3x_1 + x_2 \\ 5x_1 - x_2 \leq 7, \\ x \geq 0 \end{cases}$ trasformare nel formato sduale standard $\begin{cases} \min c^T \cdot x \\ Ax = b \\ x \geq 0 \end{cases}$

$\begin{cases} \min 3x_1 + x_2 + 0x_3 \\ 5x_1 - x_2 + x_3 = 7 \\ x \geq 0 \end{cases}$ in questo caso , e lo si riconosce nei problemi di PL perché ha coefficiente nullo nella funzione obiettivo e compare in un solo vincolo

3. Problema di trasporto

Esempio 3.1 Uno spedizioniere deve trasportare 11 tonnellate di prodotto divisibile. I suoi spedizionieri possono trasportare al massimo 5 tonnellate e 6 tonnellate, rispettivamente. Gli acquirenti hanno bisogno di almeno 3 tonnellate, 3 tonnellate, 2 tonnellate, rispettivamente. Si ipotizza che gli spedizionieri non possano comunicare tra loro, così come gli acquirenti.



$x = (x_{13}, x_{14}, x_{15}, x_{23}, x_{24}, x_{25})$. $\bar{x} = (3, 2, 0, 0, 1, 2)$ è una soluzione ammissibile.

$$\text{Modello matematico: } \begin{cases} \min c_{13}x_{13} + x_{14}c_{14} + x_{15}c_{15} + x_{23}c_{23} + x_{24}c_{24} + x_{25}c_{25} \\ \begin{aligned} &x_{13} + x_{14} + x_{15} \leq 5 \\ &x_{23} + x_{24} + x_{25} \leq 6 \end{aligned} \end{cases} \text{ vincoli di produzione}$$

$$\begin{cases} x_{13} + x_{23} \geq 3 \\ x_{14} + x_{24} \geq 3 \\ x_{15} + x_{25} \geq 2 \end{cases} \text{ vincoli di destinazione}$$

$$x_{ij} \geq 0$$

Nei vincoli di destinazione ci va \geq perché si ipotizza che un acquirente accetti prodotto in più da mettere in stoccaggio.

Modello.

Problema di trasporto:

Determinare il modo più efficiente di distribuire una certa quantità di merce da più punti di origine (magazzini, fabbriche) a più punti di destinazione (clienti, magazzini di distribuzione), minimizzando i costi di trasporto.

$$(p) \begin{cases} \min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \\ \sum_{i=1}^m x_{ij} \geq d_j \quad \forall j = 1, \dots, n \\ \sum_{j=1}^n x_{ij} \leq o_i \quad \forall i = 1, \dots, m \\ x_{ij} \geq 0 \end{cases}$$

4. Teoria della PL

Enunciamo tre teoremi che ci aiutano a trovare una soluzione ottima a problemi di PL. Prima, però, alcune definizioni.

Definizione.

Definizione (combinazione convessa):

Siano $x^1, \dots, x^k \in \mathbb{R}^n$, $y \in \mathbb{R}^n$

y si dice combinazione convessa di x^1, \dots, x^k se

$$\exists \lambda_1, \dots, \lambda_k \in [0, 1], \sum_{i=1}^k \lambda_i = 1 \text{ t. c. } y = \sum_{i=1}^k \lambda_i x^i$$

Ovvero se esistono k -coefficienti λ_i tra 0 e 1, a somma 1 tali che $y = \lambda_1 x^1 + \dots + \lambda_k x^k$

Questo concetto è molto simile alla combinazione lineare di $\{x^1, \dots, x^k\}$, solo che i coefficienti λ_i hanno delle proprietà aggiuntive. In particolare, sono in $[0, 1]$ e la somma fa 1.

Definizione.

Definizione (involucro convesso):

L'invilucro convesso di x^1, \dots, x^k ($\text{conv}\{x^1, \dots, x^k\}$) è l'insieme di tutte le possibili combinazioni convesse di x^1, \dots, x^k .

Esempio 4.1 in \mathbb{R}^2 :

- $k = 1 \implies y = \lambda_1 x^1 = 1x^1$, le combinazioni convesse di un punto sono il punto stesso
- $k = 2 \implies y = \lambda_1 x^1 + \lambda_2 x^2 = \lambda x^1 + (1 - \lambda)x^2$ dove $\lambda = \lambda_1 + \lambda_2$, $\lambda \in [0, 1]$
 $\text{conv}\{x^1, x^2\}$ è il segmento che congiunge x^1, x^2
- $k = 3 \implies \text{conv}\{x^1, x^2, x^3\}$ è il triangolo che ha come vertici i punti x^1, x^2, x^3 . Questo si vede ponendo a zero uno dei punti ($\lambda_i = 0$), ottenendo come conv il segmento che congiunge gli altri due.

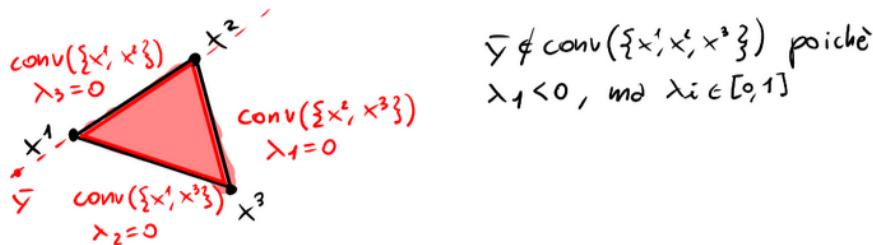


Figura 3: $\text{conv}\{x^1, x^2, x^3\}$

Ricordiamo che un insieme convesso K è un insieme dove il segmento che congiunge due punti qualsiasi "non esce" dall'insieme, ovvero $\lambda x^1 + (1 - \lambda)x^2 \in K \forall x^1, x^2, \lambda \in [0, 1]$.

Si può dimostrare che $\text{conv}\{x^1, \dots, x^k\}$ è il più piccolo insieme convesso che contiene x^1, \dots, x^k
 \Leftrightarrow un insieme convesso K coincide con il suo $\text{conv}(K)$.

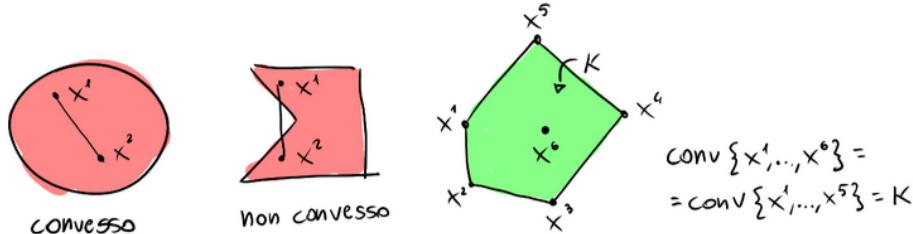


Figura 4: Insiemi convessi e non convessi, K convesso $= \text{conv}(K)$

Definizione.

Definizione (combinazione conica):

Siano $x^1, \dots, x^k \in \mathbb{R}^n$, $y \in \mathbb{R}^n$

y si dice combinazione conica di x^1, \dots, x^k se

$$\exists \lambda_1, \dots, \lambda_k \in \mathbb{R}^+ \text{ t.c. } y = \sum_{i=1}^k \lambda_i x^i$$

Ovvero se esistono k -coefficienti λ_i positivi tali che $y = \lambda_1 x^1 + \dots + \lambda_k x^k$

Definizione.

Definizione (cono):

Un cono K è un insieme tale che se contiene un punto di K , diverso dall'origine, allora contiene anche tutta la semiretta uscente dall'origine passante per quel punto.

Ovvero $\lambda x \in K \forall x \in K, \lambda \geq 0$.

I coni possono essere convessi o non. A noi interesseranno i coni convessi.

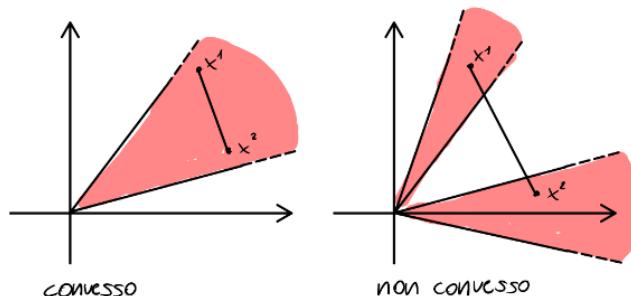


Figura 5: Cono convesso e cono non convesso

Definizione.

Definizione (involucro conico):

L'involucro conico di x^1, \dots, x^k (cono $\{x^1, \dots, x^k\}$) è l'insieme di tutte le possibili combinazioni coniche di x^1, \dots, x^k .

Similmente all'involucro convesso, si può dimostrare che cono $\{x^1, \dots, x^k\}$ è il più piccolo cono convesso che contiene x^1, \dots, x^k : un cono convesso K coincide con il suo cono(K).

Per esempio, per $k = 1$, cono(x^1) è una retta; per $k = 2$, cono(x^1, x^2) è un cono convesso come nella figura precedente, dove x^1, x^2 sono punti del bordo.

Esempio 4.2 $\text{cono}\{(1, 0), (0, 1), (1, 1)\} = \text{cono}\{(1, 0), (0, 1)\}$ è il primo quadrante poiché $(1, 1)$ è una combinazione conica di $(1, 0)$, $(0, 1)$ ovvero $(1, 1)$ è **conicamente dipendente** dagli altri due vettori.

4.1. Teorema di rappresentazione dei poliedri

O anche chiamato Teorema di Weil, il primo teorema del corso ci fornisce una **rappresentazione** di un poliedro, ovvero una condizione necessaria e sufficiente per caratterizzarlo.

TEOREMA.

Teorema (di rappresentazione dei poliedri):

Dato un poliedro $P = \{Ax \leq b\}$, allora

$$\exists V = \{v^1, \dots, v^k\} \subseteq P, \exists E = \{e^1, \dots, e^p\} \subset \mathbb{R}^n \text{ t.c. } P = \text{conv}(V) + \text{cono}(E)$$

La somma tra insiemi $P = \text{conv}(V) + \text{cono}(E)$ significa $\{a + b : a \in \text{conv}(V), b \in \text{cono}(E)\}$

Esempio 4.3

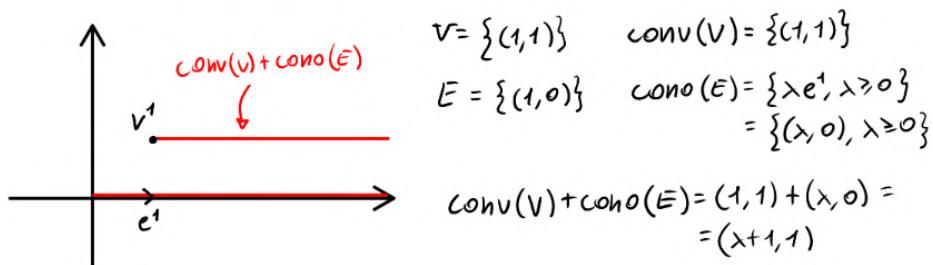
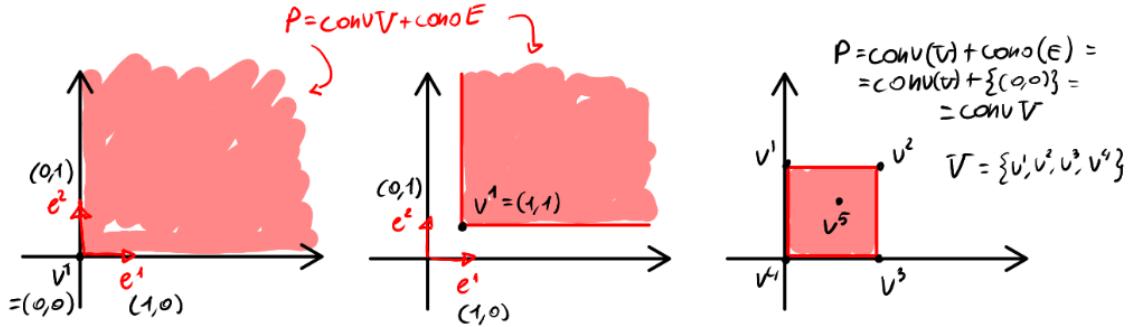


Figura 6: vettori λe^1 applicati a un punto

Conviene considerare V come un insieme di punti e E come un insieme di vettori: sommare $\text{conv}(V) + \text{cono}(E)$ significa **applicare i vettori di E sui punti di V** .

Figura 7: Altri esempi di $\text{conv}(V) + \text{cono}(E)$

È importante sapersi muovere tra le tre forme dei poliedri: **algebrica** ($Ax \leq b$), **geometrica** (intersezione di sottospazi) e **forma di Weil** ($\text{conv}(V) + \text{cono}(E)$)

4.2. Teorema fondamentale della PL

Il teorema fondamentale della PL, combinato al Teorema di Weil e ad altre considerazioni, ci permette di giungere a un importante risultato per la risoluzione di problemi di PL.

TEOREMA.

Teorema (fondamentale della PL):

Sia $(p) = \begin{cases} \max c^T \cdot x \\ Ax \leq b \end{cases}$ un problema di PL in forma primale standard, e sia V, E la rappresentazione del poliedro in forma di Weil. Vale una delle tre opzioni:

1. $P = \emptyset$, per cui ci sono troppi vincoli
2. $\max c^T x = +\infty$, ho dimenticato vincoli
3. $\exists r \in \{1, \dots, k\} : v^r \in V$ è soluzione ottima di (p)

Ovvero, se il poliedro non è vuoto e il problema è limitato, esiste un elemento di V che è soluzione ottima.

Dimostrazione:

Sia dato un problema di PL in formato primale standard, e sia $\bar{x} \in P$ una soluzione ammissibile.

$$\bar{x} = \text{conv}(V) + \text{cono}(E) = \sum_{i=1}^k \bar{\lambda}_i v^i + \sum_{i=1}^p \bar{\mu}_i e^i \quad \text{dove } \bar{\lambda}_i \in [0, 1], \ \sum \bar{\lambda}_i = 1, \ \bar{\mu}_i \geq 0$$

$$c^T \cdot \bar{x} = \sum_{i=1}^k \bar{\lambda}_i (v^i \cdot c^T) + \sum_{i=1}^p \bar{\mu}_i (e^i \cdot c)$$

Se un solo prodotto scalare $e^i \cdot c > 0$, allora $(p) = +\infty$.

Se i prodotti scalari $e^i \cdot c \leq 0 \ \forall i$, allora la soluzione di $\max c^T \cdot x = c \cdot v^r$ è il v^r più alto tra tutti i v^i . □

- Per verificare che $P \neq \emptyset$ basta trovare una soluzione ammissibile
- Per verificare che $(p) < +\infty$ devo fare i prodotti scalari $c^T e_j \forall j = 1 \dots p$. Se solo uno di questi prodotti scalari è positivo, allora $(p) = +\infty$. Il problema è che non ho un algoritmo per calcolare gli e_j . Quindi potrebbe tornare utile una verifica geometrica, disegnando il poliedro.

Definizione.

Definizione (vertice):

Sia P un poliedro. Un punto $\bar{x} \in P$ si dice vertice se non si può esprimere come combinazione convessa di due punti di P diversi da \bar{x} . (i.e. combinazione convessa propria)

Proposizione.

Proposizione 1: se il poliedro P è limitato, allora $E = \{\vec{0}\}$ quindi $\text{cono}(E) = \emptyset$ i.e. non ci sono direzioni di recessione (gli "insostituibili" di E). Inoltre, i poliedri limitati hanno vertici.

Proposizione 2: i vertici sono gli unici punti insostituibili nella combinazione convessa della rappresentazione di Weil, per cui, se il poliedro P ha vertici, allora $V = \{\text{vert}(P)\}$

Combinandole assieme: Se P è limitato, $P = \text{conv}(V) + \text{cone}(E) = \text{conv}\{\text{vert}(P)\}$

Attenzione: un poliedro può non avere vertici, ma posso comunque trovare la sua rappresentazione di Weil. I poliedri senza vertici in \mathbb{R}^2 sono i **semipiani**, le **striscie** e le **rette**.

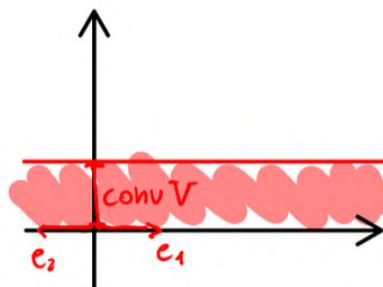


Figura 8: Poliedro senza vertici

Osservazione: i poliedri senza vertici sono quelli che contengono una retta. Se nei vincoli c'è $x \geq 0$, allora il poliedro ha almeno un vertice.

Combiniamo ora le proposizioni e il teorema fondamentale della PL per giungere a un importante risultato.

1. Per le proposizioni 1 e 2: se P è limitato allora $P = \text{conv}(\text{vert}(P))$
2. Per il teorema fondamentale della PL: se $P \neq \emptyset$ e P è limitato (quindi $(p) < +\infty$), allora esiste almeno un elemento di $V = \text{vert}(P)$ che è soluzione ottima di (p) . I poliedri nonvuoti di ricerca operativa, essendo limitati, hanno almeno un vertice che è soluzione ottima!

Proposizione.

Se il poliedro P è limitato, allora esiste almeno un vertice che è soluzione ottima.

4.3. Calcolo dei vertici

Ora che abbiamo dimostrato che possiamo cercare le soluzioni ottime nei vertici, calcoliamo i vertici del poliedro $Ax \leq b$ in formato primale standard.

Definizione.

Definizione (soluzione di base):

Sia $B := \{1, \dots, m\}$: $|B| = n$ un sottoinsieme degli indici di riga

Sia A_B la sottomatrice di A che ha le righe individuate dagli indici di B

Sia b_B il sottovettore di b individuato dagli indici di B

Se A_B è invertibile, ovvero $\det A_B \neq 0$, allora la soluzione del sistema $A_B \cdot x = b_B$ si chiama **soluzione di base**. Formula chiusa: $x = A_B^{-1}b_B$

Esempio 4.4 $n = 2, m = 4$

$$Ax \leq b \equiv \begin{cases} 1. x_1 + 2x_2 \leq 6 \\ 2. 3x_1 + x_2 \leq 7 \\ 3. -x_1 \leq 0 \\ 4. -x_2 \leq 0 \end{cases} \quad B := \{3, 4\}$$

Esempio 4.4 $n = 2, m = 4$

$$Ax \leq b \equiv \begin{cases} 1. x_1 + 2x_2 \leq 6 \\ 2. 3x_1 + x_2 \leq 7 \\ 3. -x_1 \leq 0 \\ 4. -x_2 \leq 0 \end{cases} \quad B := \{3, 4\}$$

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \Rightarrow A_B = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 6 \\ 7 \\ 0 \\ 0 \end{bmatrix} \Rightarrow b_B = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$A = \begin{bmatrix} A_B \\ A_N \end{bmatrix}, \quad b = \begin{bmatrix} b_B \\ b_N \end{bmatrix} \Rightarrow Ax \leq b \Leftrightarrow \begin{cases} A_B x \leq b_B \\ A_N x \leq b_N \end{cases}$$

$$\det(A_B) \neq 0$$

$$\therefore A_B \cdot \bar{x} = b_B \iff \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \iff \bar{x} = (0, 0) \text{ è la soluzione di base}$$

Se il determinante fosse zero, le rette sarebbero parallele o coincidenti, dunque non ci sarebbe un vertice. In pratica devo fare l'intersezione di n-rette.

Definizione.

Definizione (soluzione di base ammissibile):

Una soluzione di base si dice **ammissibile** se appartiene al poliedro.

Esempio 4.5 Nell'esempio precedente, $\bar{x} = (0, 0)$ è soluzione di base ammissibile perché non viola nessun vincolo del poliedro. Se prendo come base $B = \{1, 4\}$, si può verificare che $\bar{x} = (6, 0)$, che non è una base ammissibile.

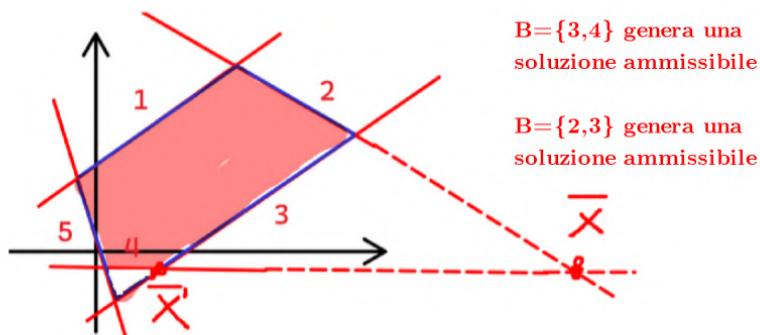


Figura 9: significato geometrico delle soluzioni di base ammissibili

TEOREMA.

Teorema (caratterizzazione dei vertici):

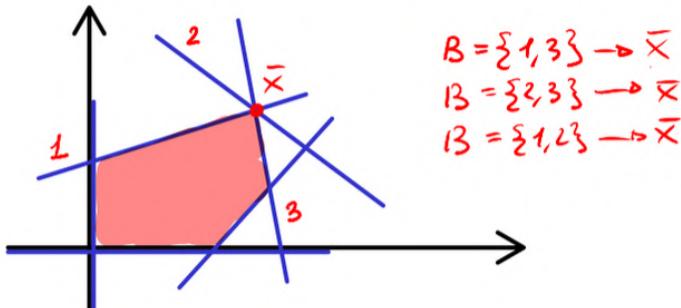
Un punto \bar{x} del poliedro è un vertice se e solo se è una soluzione di base ammissibile

Questo teorema è una definizione alternativa di vertice. In questo caso, l'altra definizione di vertice che abbiamo dato può essere usata come una proprietà.

Definizione.

Definizione (soluzione di base degenere):

La soluzione di base \bar{x} si dice degenere se più basi generano \bar{x} .



Degenere non vuol dire eliminabile (ma vale solo in R^2). Addirittura non esistono algoritmi per verificare se un vincolo è combinazione lineare di altri vincoli (per cui sarebbe eliminabile).

Un punto è soluzione di base se rispetta n vincoli con l'uguale. Un punto è soluzione di base degenere se ne rispetta almeno $n + 1$.

L'**algoritmo di enumerazione totale** esplora tutti i vertici (ovvero tutte le possibili soluzioni di base ammissibili) per trovare quello ottimale. Il problema con questo algoritmo è che bisogna calcolare le possibili basi, che sono le *combinazioni di n elementi presi da un basket di m* :

$$\binom{m}{n} = \frac{m!}{n!(m-n)!}$$

La complessità è fattoriale, dunque è parecchio inefficiente.

4.4. Trasformazioni equivalenti

Mantengono inalterati i punti di ottimo e il valore della funzione obiettivo, ma modificano la forma del problema. Sono utili per convertire problemi in forme standard.

Corollario.

Trasformazioni equivalenti:

- $\min \leftrightarrow \max \quad \min(f) = -\max(-f)$
- $\leqslant \leftrightarrow \geqslant \quad x \leqslant x_0 \Leftrightarrow -x \geqslant -x_0$
- $= \rightarrow \leqslant, \geqslant \quad x = x_0 \Rightarrow x \geqslant x_0, x \leqslant x_0$
- $\leqslant \rightarrow =, \text{ Slack} \quad x \leqslant x_0 \Rightarrow x = x_0 + S, S \geqslant 0$
- $x \geqslant 0 \rightarrow x_i^{++}, x_i^+ \geqslant 0 \quad x \geqslant 0 \Rightarrow x_i = x_i^{++} - x_i^+ \text{ con } x_i^{++} \geqslant x_i^+ \geqslant 0 \forall i$

A cosa serve l'ultima trasformazione equivalente? Ad esempio:

$$(p) \begin{cases} \min 3x_1 + x_2 \\ 5x_1 + x_2 \geq 7, \\ x_1 \geq 0 \end{cases} \quad \text{portare in formato standard } (q) \begin{cases} \min c^T x \\ Ax \geq b \\ x \geq 0 \end{cases}$$

Non posso semplicemente aggiungere la condizione $x_2 \geq 0$ perché il primo problema non ci dice niente sul suo segno: x_2 potrebbe essere negativo e i vincoli sarebbero comunque rispettati. Allora osservo che ogni numero in \mathbb{Z} si può scrivere come differenza di due numeri positivi.

$$\begin{cases} \min 3x_1 + x_2 \\ 5x_1 + x_2^{++} - x_2^+ \geq 7 \\ x_1, x_2^{++}, x_2^+ \geq 0 \end{cases} \equiv \begin{cases} \min 3x_1 + x_2^{++} - x_2^+ \\ 5x_1 + x_2^{++} - x_2^+ \geq 7 \\ x \geq 0 \end{cases}$$

Devo sostituire x_2 anche nella funzione obiettivo per ottenere un sistema in tre incognite. Osservo che, se $x' = (6, 3, 2)$ è una soluzione ammissibile di (q) , allora $x'' = (6, 3 - 2) = (6, 1)$ è una soluzione ammissibile di (p) .

4.5. Teoria della dualità

4.5.1. Vertici del poligono in formato duale standard

C'è un modo molto più rapido per calcolare i vertici di un poligono **in formato duale standard**. Sdoppiare le uguaglianze per portare il poliedro in formato primale standard significa avere tante righe che sono linearmente dipendenti.

Definizione.

Definizione (soluzione di base duale):

Si divide la variabile $x = (x_1, \dots, x_n)$ in blocco di base x_B e blocco non di base x_N $x_B \in \mathbb{R}^k$ dove k è il numero di equazioni.

Pongo $x_N = \vec{0}$, mi assicuro che il sistema di k equazioni in k incognite abbia soluzione (matrice invertibile) e risolvo per x_B . Ricostruisco $x = (x_B, x_N)$

Simile alla definizione per il formato primale standard, ma anziché le righe si prendono le variabili.

Esempio 4.6

$$(p) \begin{cases} 3x_1 + 2x_2 + 4x_3 + x_4 = 3 \\ 2x_1 + x_2 + 3x_3 + x_4 = 2 \\ x \geq 0 \end{cases}$$

$$B = \{1, 2\} \Rightarrow (p_B) \begin{cases} 3x_1 + 2x_2 = 3 \\ 2x_1 + x_2 = 2 \end{cases}, \det \begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix} \neq 0 \Rightarrow x_B = (1, 0) \Rightarrow x = (1, 0, 0, 0)$$

$$B = \{2, 4\} \Rightarrow (p_B) \begin{cases} 2x_2 + x_4 = 3 \\ x_2 + x_4 = 2 \end{cases}, \det \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \neq 0 \Rightarrow x_B = (1, 1) \Rightarrow x = (0, 1, 0, 1)$$

Le definizioni di soluzione di base ammissibile e degenere e il teorema di caratterizzazione dei vertici sono facilmente adattabili alle soluzioni di base duali.

Osservazione: \bar{x} soddisfa il vincolo $A\bar{x} = b$ per costruzione. Quindi, per verificare se \bar{x} è una soluzione di base ammissibile, mi basta che $\bar{x} \geq 0$

4.5.2. Test di ottimalità

Dato un vertice \bar{x} di P , come faccio a sapere che sono arrivato all'ottimo (senza enumerare tutti i vertici)?

$$(p) \begin{cases} \max c^T x \\ Ax \leq b \end{cases}, \text{ il suo problema duale è } (d) = \begin{cases} \min y^T b \\ y^T A = c \\ y \geq 0 \end{cases}$$

dove $c, x \in \mathbb{R}^n$, $y, b \in \mathbb{R}^m$ sono vettori colonna. $A \in \mathbb{R}^{m \times n}$

Il problema primale ha n variabili e m vincoli, il duale ha m variabili e n vincoli.

Il problema duale è completamente diverso dall'originale, questa forma non ha a che vedere con le trasformazioni equivalenti.

Esempio 4.7

$$(p) \begin{cases} \max x_1 - x_2 \\ 3x_1 + 2x_2 \leq 8 \\ -x_1 + x_2 \leq 6 \\ 3x_1 - 2x_2 \leq 9 \end{cases}, A = \begin{bmatrix} 1 & -1 \\ 3 & 2 \\ -1 & 1 \\ 3 & -2 \end{bmatrix} \rightarrow (d) \begin{cases} \min 8y_1 + 6y_2 + 9y_3 \\ 3y_1 - y_2 + 3y_3 = 1 \\ 2y_1 + y_2 - 2y_3 = -1 \\ y \geq 0 \end{cases}$$

TEOREMA.

Teorema (scarti complementari):

Sia $\bar{x} \in P$, $\bar{y} \in D$ soluzioni ammissibili.

\bar{x} , \bar{y} sono soluzioni ottime (rispettivamente di (p) e (d)) $\iff \bar{y} \cdot (b - A\bar{x}) = 0$

Dove $\bar{y} \cdot (b - A\bar{x}) = 0$ si chiama **condizione degli scarti complementari**.

Osservazioni:

1. Due soluzioni di base \bar{x} , \bar{y} complementari (generate dalla stessa base) sono sempre in scarti complementari.
2. Essendo (d) in forma duale standard, se \bar{y} è una soluzione di base dual, allora se $\bar{y} \geq 0$, è anche ammissibile.

Corollario.

Test di ottimalità:

Oss. 1 \rightarrow Dato un vertice \bar{x} generato dalla base B , costruiamo la \bar{y} con la stessa base B .

Oss. 2 \rightarrow Se $\bar{y} \geq 0$, cioè $c(A_B^T)^{-1} \geq 0$, per il th. degli scarti complementari, \bar{x} è ottima.

Esempio 4.8

$$(p) \begin{cases} \max 3x_1 + 2x_2 \\ 5x_1 + 6x_2 \leq 9 \\ 3x_1 + 8x_2 \leq 7 \\ -x_1 \leq 0 \\ -x_2 \leq 0 \end{cases} \rightarrow (d) \begin{cases} \min 9y_1 + 7y_2 \\ 5y_1 + 3y_2 - y_3 = 3 \\ 6y_1 + 8y_2 - y_4 = 2 \end{cases}$$

So che $\bar{x} = (0, 0)$ è generato dalla base $B = \{3, 4\}$ ed è ammissibile, quindi è un vertice.
 \bar{x} è ottimo?

$$B = \{3, 4\}, \begin{cases} \min 9y_1 + 7y_2 \\ -y_3 = 3 \\ -y_4 = 2 \end{cases} \Rightarrow \bar{y} = (0, 0, -3, -2) < 0, \text{ quindi } \bar{x} \text{ non è l'ottimo.}$$

TEOREMA.**Teorema (dualità forte):**

Sia (p) un problema primale e (d) il suo duale. Siano $P, D \neq \emptyset$

Il valore ottimo del primale coincide con quello del problema duale i.e. $\max_{Ax \leq b} cx = \min_{yA=c, y \geq 0} yb$

Allora esiste il test di ottimalità anche per il problema duale: sia una soluzione di base ammissibile $\bar{y} \in D$, cioè un vertice del duale; costruiamo la soluzione di base (con la stessa base) del primale associato, \bar{x} . Se \bar{x} è ammissibile i.e. $b - A\bar{x} \geq 0$ siamo all'ottimo. **L'operazione di duale è involutoria:** il duale del duale è il primale, quindi i problemi di PL vanno sempre in coppia, e uno diventa il test di ottimalità dell'altro.

5. Algoritmo del simplex

L'algoritmo del simplex è stato ideato da George Dantzig nel 1946, mentre stava lavorando sui metodi di pianificazione dell'aeronautica militare degli Stati Uniti. L'idea fondamentale dell'algoritmo è partire da un vertice e spostarsi lungo gli spigoli di vertice in vertice, fino a raggiungere l'ottimo.

Correva l'anno 1984. Il matematico indiano Karmarkar ha scoperto un algoritmo polinomiale per risolvere problemi di PL. Incuriosito, il giovane Pappalardo parte per l'America per assistere alla presentazione. L'aula è affollata di matematici e alle 9:00 in punto, l'indiano entra, accompagnato dalle guardie del corpo. Sfodera i lucidi e li proietta a una velocità impressionante, circa tre secondi per diapositiva, con alcuni spettatori che scattano freneticamente foto. In circa 30 pagine, dimostra come il suo algoritmo, a differenza del metodo del simplex, non si muove da vertice a vertice, ma punta direttamente all'ottimo. Ciononostante, anche dopo 40 anni, l'algoritmo del simplex resta più efficace nella pratica, pur essendo esponenziale.

5.1. Algoritmo del simplex primale

Sia (p) $\begin{cases} \max cx \\ Ax \leq b \end{cases}$ un problema di PL in formato primale standard

5.1.1. Passo base

Supponiamo di avere un vertice $\bar{x} \in P$, $\bar{x} = A_B^{-1}b_B$

Costruiamo $\bar{y} = (cA_B^{-1}, 0)$ la soluzione di base duale complementare.

Se $\bar{y}_B \geq 0$ siamo all'ottimo. Altrimenti $\exists k \in B : \bar{y}_k < 0$

5.1.2. Passo induttivo

Idea: Se non siamo all'ottimo, effettuo un cambio di base. C'è un indice h uscente e un indice k entrante in base. In altre parole mi sposto lungo lo spigolo a un vertice adiacente. In generale, in \mathbb{R}^n , tra vertici adiacenti, ci sono $n - 1$ indici uguali.

Trovo l'indice uscente.

Voglio che lo spostamento produca un aumento della funzione obiettivo cx .

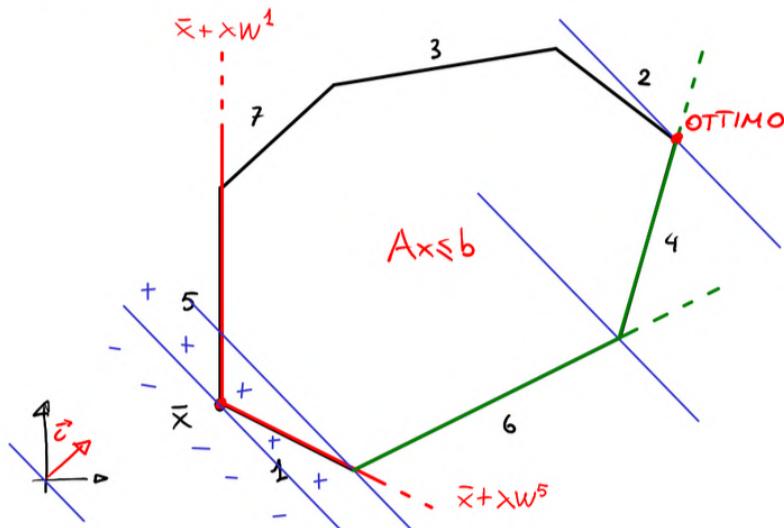


Figura 10: algoritmo del simplex con vertice iniziale \bar{x} e tre passi induttivi.

Proposizione.

Proposizione:

Sia $W := -A_B^{-1}$, siano W^i le colonne di W .

$\bar{x} + \lambda W^i, \lambda \geq 0$ è l'equazione della semiretta uscente da \bar{x} con direzione di W^i (attenzione, non mi sto muovendo lungo lo spigolo i , ma lungo quello adiacente).

Calcolo la funzione obiettivo lungo la semiretta:

$c(\bar{x} + \lambda W^i) = c\bar{x} + \lambda cW^i$ dove $c\bar{x}$ è il valore della funzione obiettivo corrente, $\lambda \geq 0$ sempre.

Quindi, per far crescere la funzione obiettivo mi serve che cW^i sia positivo.

Osservo che $cW^i = c(-A_B^{-1})^i = -c(A_B^{-1})^i = -\bar{y}_i$.

Questo è coerente con il fatto che se $\bar{y}_i \geq 0 \Rightarrow cW^i \leq 0$ ovvero siamo all'ottimo. Dunque se non trovo una $y_i \geq 0$ vuol dire che lo spostamento nella direzione dei vertici adiacenti produce sempre un aumento della funzione obiettivo. Ma quale devo scegliere?

Regola di anticiclo di Bland: Scelgo il primo indice per cui la duale è negativa.

$h = \min\{i \in B : y_i < 0\}$. Questo si fa per garantire che l'algoritmo non entri in un ciclo e quindi termini in un numero finito di passi.

Ad esempio: $\bar{y} = (0, 7, -4, 0, 0, -5) \Rightarrow h = 3$.

Trovo l'indice entrante.

$A(\bar{x} + \lambda W^h) \stackrel{?}{\leq} b$, mi chiedo per quali λ rimango dentro al poliedro?

Prendo gli indici i che si dividono in base e non base: $A_i(\bar{x} + \lambda W^h) \stackrel{?}{\leq} b_i, \{i \in B \vee i \in N\}$, per la proprietà distributiva: $A_i\bar{x} + \lambda A_i W^h \leq b_i$

Vediamo il caso per i vertici di base:

Poiché $i \in B$, allora $A_i\bar{x} = b_i$.

$b_i + \lambda A_i W^h \leq b_i \Rightarrow \lambda A_i W^h \leq 0$, poiché $\lambda \geq 0$, mi interessa il segno di $A_i W^h = A_i(-A_B^{-1})^h$.

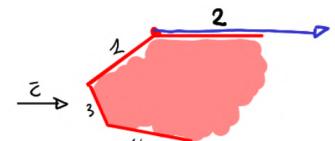
Se moltiplico una riga di A_B per una colonna di A_B^{-1} fa 0 oppure 1 perché fa l'identità. Siccome è scambiata di segno, fa 0 oppure -1 . In ogni caso vale sempre $A_i W^h \leq 0$. Non ci sono problemi per ogni indice di base: non è possibile uscire dal poliedro "per colpa" di un indice di base.

Vediamo il caso per i vertici non di base:

Se $A_i W^h \leq 0 \forall i \in N$ significa che posso procedere all'infinito, $(p) = +\infty$

$$\text{Se } \exists i : A_i W^h > 0, \text{ allora } A_i \bar{x} + \lambda A_i W^h \leq b_i \Leftrightarrow \lambda \leq \frac{b_i - A_i \bar{x}}{A_i W^h} =: r_i$$

Calcolo solo gli r_i dove $A_i W^h > 0$ poiché gli altri non hanno conseguenze sull'uscita del poliedro (calcolo al più $m - n$ rapporti). Per non uscire dal poliedro devo prendere.



$$\min \left\{ r_i = \frac{b_i - A_i \bar{x}}{A_i W^h} : i \in N, A_i W^h > 0 \right\}$$

Anche qui si applica la regola di anticiclo di Bland: prendo l'indice entrante k come il primo i nel caso in cui ci siano due rapporti r_i minimi uguali.

$\bar{\lambda} = \frac{b_k - A_k \bar{x}}{A_k W^h}$ è il massimo λ che posso prendere per rimanere dentro al poliedro.

Oss: $A_k \bar{x} + \frac{b_k - A_k \bar{x}}{A_k W^h} A_i W^h \Big|_{i=k} = b_k$ vuol dire che l'indice k è soddisfatto con l'uguale.

ALGORITMO.

Algoritmo del simplesso primale

1. Trova una base B che genera una soluzione di base primale ammissibile $\bar{x} := A_B^{-1} b_B$
2. Calcola la soluzione di base duale $\bar{y} := \begin{bmatrix} \bar{y}_B \\ \bar{y}_N \end{bmatrix}$ con $\bar{y}_B^T := c^T A_B^{-1}$, $\bar{y}_N := 0$
3. if $y_B \geq 0$ then STOP (x è ottima per (p) e y è ottima per (d)).
else calcola l'indice uscente $h := \min\{i \in N : \bar{y}_i < 0\}$
poni $W := -A_B^{-1}$ e indica W^h la h -esima colonna di W .
4. if $A_i W^h \leq 0 \forall i \in N$ then STOP ($(p) = +\infty$, $(d) = \emptyset$)
else pon $r_i := \frac{b_i - A_i \bar{x}}{A_i W^h}$ se $i \in N$, $A_i W^h > 0$ e calcola $\bar{\lambda} := \min\{r_i\}$
calcola l'indice entrante $k := \min\left\{i : \frac{b_i - A_i \bar{x}}{A_i W^h} = \bar{\lambda}\right\}$
aggiorna la base $B := B \setminus \{h\} \cup \{k\}$ e torna al passo 2.

5.2. Algoritmo del simplesso duale

Ricordiamo come i problemi di PL vanno sempre a coppie (primale/duale). Distinguo quattro casi:

- A. Soluzione di base ammissibile per (p) , $(d) \rightarrow$ siamo all'ottimo
- B. Soluzione di base ammissibile per (p) , non ammissibile per $(d) \rightarrow$ simplex primale
- C. Soluzione di base ammissibile per (d) , non ammissibile per $(p) \rightarrow$ simplex duale
- D. Soluzione di base non ammissibile per (p) , $(d) \rightarrow$ trovo un'altra soluzione di base.

Sia $\bar{y} \in D$ la soluzione di base ammissibile (vertice di D)

Sia $(p) \begin{cases} \max cx \\ Ax \leq b \end{cases}$ un problema di PL in formato primale standard

5.2.1. Passo base

Supponiamo di avere un vertice $\bar{y} \in D$, $\bar{y} = (\bar{y}_B, \bar{y}_N) = (c^T A_B^{-1}, 0) : c^T A_B \geq 0$

Se \bar{x} è ammissibile ovvero $b_i - A_i \bar{x} \geq 0$, allora siamo all'ottimo.

5.2.2. Passo induttivo

Indice entrante: Altrimenti, sia k il primo (anticiclo) indice $i \in N$ tale che $b_k - A_k \bar{x} < 0$.

Indice uscente: faccio i prodotti scalari $A_k W^i$ (k l'ho già trovato e devo trovare $i \in B$ uscente).

Prendo solo $A_k W^i < 0$ (se sono tutti ≥ 0 allora $(d) = -\infty$) e trovo il minimo di $r_i := \frac{-y_i}{A_k W^i}$.

h è il primo degli indici per cui viene raggiunto il minimo

ALGORITMO.

Algoritmo del simplex duale

1. Trova una base B che genera una soluzione di base primale ammissibile
 $\bar{y} := \begin{bmatrix} \bar{y}_B \\ \bar{y}_N \end{bmatrix}$ con $\bar{y}_B^T := c^T A_B^{-1}$, $\bar{y}_N := 0$
2. Calcola la soluzione di base duale $\bar{x} := A_B^{-1} b_B$
3. if $b_N - A_N \bar{x} \geq 0$ then STOP (\bar{y} è ottima per (d) e \bar{x} è ottima per (p)).
else calcola l'indice entrante $k := \min\{i \in N : b_i - A_i \bar{x} < 0\}$
poni $W := -A_B^{-1}$ e indica W^i la i -esima colonna di W .
4. if $A_k W^i \geq 0 \forall i \in B$ then STOP ($(d) = -\infty$, $(p) = \emptyset$)
else pon $r_i := \frac{y_i}{-A_k W^i}$ se $i \in B$, $A_k W^i < 0$ e calcola $\bar{\lambda} := \min\{r_i\}$
calcola l'indice uscente $h := \min\left\{i : \frac{y_i}{-A_k W^i} = \bar{\lambda}\right\}$
aggiorna la base $B := B \setminus \{h\} \cup \{k\}$ e torna al passo 2.

5.3. Problema duale ausiliario

Non sempre ho a disposizione un vertice per iniziare l'algoritmo del simplex. Vediamo un modo per trovare sicuramente un vertice da cui partire.

Definizione.

Definizione (duale ausiliario):

Sia $D = \begin{cases} Ax = b \\ x \geq 0 \end{cases}$ un poliedro in formato duale std, dove $A \in \mathbb{R}^{n \times m}$, $x \in \mathbb{R}^m$, $b \in \mathbb{R}^n$

$$\text{Il problema duale ausiliario è (da)} \begin{cases} \min \sum_{i=1}^m \epsilon_i \\ Ax + I\epsilon = b \\ x, \epsilon \geq 0 \end{cases}$$

$$\text{Ad esempio: } D = \begin{cases} 3x_1 + 5x_2 + 6x_3 + x_4 = 8 \\ 2x_1 + 3x_2 + 7x_3 + 3x_4 = 9 \\ x \geq 0 \end{cases} \rightarrow (\text{da}) \begin{cases} \min \epsilon_1 + \epsilon_2 \\ 3x_1 + 5x_2 + 6x_3 + x_4 + \epsilon_1 = 8 \\ 2x_1 + 3x_2 + 7x_3 + 3x_4 + \epsilon_2 = 9 \\ x, \epsilon \geq 0 \end{cases}$$

TEOREMA.

Teorema del duale ausiliario:

Sia $v(\text{da})$ il valore ottimo di (da).

- Se $v(\text{da}) > 0$, allora $D = \emptyset$
- Se $v(\text{da}) = 0$, allora $D \neq \emptyset$

Inoltre, se $D \neq \emptyset$, la soluzione ottima di (da) ci fornisce un vertice di D .

Mi accorgo che esiste una soluzione di base ammissibile banale, ovvero quella ottenuta settando a zero tutti gli x_i . Dall'esempio di prima, $(\bar{x}, \bar{\epsilon}) = (0, 0, 0, 0, 8, 9) \in \text{vert}(DA)$.

Il teorema ci dice anche che, se il poliedro non è vuoto, basta risolvere il problema duale ausiliario col metodo del simplex duale per trovare un vertice del problema di partenza. In questo modo, riusciamo a trovare sempre un vertice di ogni poliedro (a patto che esista).

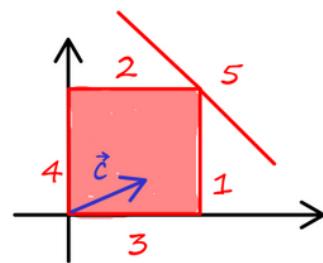
5.4. Il caso degenere

Grazie alla regola di anticiclo di Bland, nel caso degenere, se $\bar{\lambda} = \frac{b_k - A_k \bar{x}}{A_k W^h} = 0$ sempre, siamo

arrivati all'ottimo. Altrimenti, il denominatore prima o poi diventerà negativo e quindi mi sposto. Questo ci garantisce che l'algoritmo del simplex non cicla nei vertici degeneri. L'algoritmo del simplex cambia basi ad ogni passo, non i vertici.

Esempio 5.1

$$(p) \begin{cases} \max 2x_1 + x_2 \\ x_1 \leq 1 \\ x_2 \leq 1 \\ -x_1 \leq 0 \\ -x_2 \leq 0 \\ x_1 + x_2 \leq 2 \end{cases} \rightarrow (d) \begin{cases} \min y_1 + y_2 + 2y_5 \\ y_1 - y_3 + y_5 = 2 \\ y_2 - y_4 + y_5 \\ y \geq 0 \end{cases}$$



Prendo la base $B = \{1, 2\}$, $\bar{x} = (1, 1) \leftrightarrow \bar{y} = (2, 1, 0, 0, 0)$

Prendo la base $B = \{1, 5\}$, $\bar{x} = (1, 1) \leftrightarrow \bar{y} = (1, 1, 0, 0, 1)$

Prendo la base $B = \{2, 5\}$, $\bar{x} = (1, 1) \leftrightarrow \bar{y} = (0, -1, 0, 0, 2)$

Supponiamo di partire col simplesso primale dalla base $B'' = \{2, 5\}$. Il simplesso non si accorge che il punto è ottimo, ma solo riconoscere quando la base è ottima. Cambia base: esce $h = 2$, entra $k = 1$. Poi dichiara ottima la base $B' = \{1, 5\}$.

Fine teoria

Programmazione Lineare Intera (PLI)

La Programmazione Lineare Intera (PLI) è la branca della ricerca operativa che si occupa di risolvere problemi lineari in cui le variabili decisionali possono assumere solo valori interi o booleani.

Non esistono algoritmi che trovano direttamente l'ottimo di un problema di PLI non banale. Ogni problema di PLI inizia con la sua valutazione inferiore e superiore.

$$v_i \leq v_{PLI} \leq v_s : v_i, v_{PLI}, v_s \in \mathbb{Z}$$

Poi ci saranno degli algoritmi che ci permettono di ridurre l'intervallo $[v_i, v_s]$. Si arriva all'ottimo quando $v_i = v_s$ oppure ci possiamo fermare dopo un certo numero di passi quando siamo soddisfatti della stima ottenuta.

1. Problema dello zaino

In inglese "knapsack problem", anche conosciuto come "problema di caricamento". Ho uno zaino di capienza massima 30, ho degli oggetti con un peso e un valore:

Peso	$p_1 = 6$	$p_2 = 8$	$p_3 = 9$	$p_4 = 11$	$p_5 = 15$	$p_6 = 18$
Valore	$v_1 = 2$	$v_2 = 4$	$v_3 = 7$	$v_4 = 9$	$v_5 = 13$	$v_6 = 16$

$$\begin{cases} \max 2x_1 + 4x_2 + 7x_3 + 9x_4 + 13x_5 + 16x_6 \\ 6x_1 + 8x_2 + 9x_3 + 11x_4 + 15x_5 + 18x_6 \leq 30 \\ x_i \in \{0, 1\} \end{cases}$$

Modello.

Problema dello zaino:

$$\text{Booleano: } \begin{cases} \max c^T x \\ p^T x \leq P \\ x \in \{0, 1\}^n \end{cases} \quad \text{Intero: } \begin{cases} \max c^T x \\ p^T x \leq P \\ x \in \mathbb{Z}_n^+ \end{cases}$$

1.1. Valutazione superiore dello zaino intero

Con riferimento all'esempio precedente, costruisco il problema (p) ottenuto eliminando il vincolo di interezza e aggiungendo il vincolo $x_i \geq 0 \forall x_i$. (p) si chiama **rilassato continuo** del problema dello zaino intero.

$$(p) = \begin{cases} \max 2x_1 + 4x_2 + 7x_3 + 9x_4 + 13x_5 + 16x_6 \\ 6x_1 + 8x_2 + 9x_3 + 11x_4 + 15x_5 + 18x_6 \leq 30 \\ x_i \geq 0 \forall x_i \end{cases}$$

Costruisco il suo duale:

$$(d) = \begin{cases} \min 30y_1 + 0y_2 + \dots + 0y_7 \\ 6y_1 - y_2 = 2 \quad \text{Slack} \\ 8y_1 - y_3 = 4 \\ 9y_1 - y_4 = 7 \\ 11y_1 - y_5 = 9 \\ 15y_1 - y_6 = 13 \\ 18y_1 - y_7 = 16 \\ y_i \geq 0 \forall y_i \end{cases} \equiv \begin{cases} \min 30y_1 \\ 6y_1 \geq 2 \\ 8y_1 \geq 4 \\ 9y_1 \geq 7 \\ 11y_1 \geq 9 \\ 15y_1 \geq 13 \\ 18y_1 \geq 16 \\ y_i \geq 0 \forall y_i \end{cases}$$

Il problema è diventato banale e la soluzione ottima è $\bar{y} = \frac{8}{9}$. Il valore ottimo è $v(d) = 30 \cdot \frac{8}{9}$

Se trovo una \bar{x} tale che $v(p) = v(d)$ sono arrivato all'ottimo.

Provo a saturare lo zaino con x_6 :

$$(p) = \begin{cases} \max 2x_1 + 4x_2 + 7x_3 + 9x_4 + 13x_5 + 16x_6 \\ 18x_6 \leq 30 \\ x_i \geq 0 \end{cases} \implies \bar{x} = \left(0, 0, 0, 0, 0, \frac{30}{18}\right)$$

$$v(p) = \frac{30}{18} \cdot 16 = 30 \cdot \frac{8}{9} = v(d) \text{ quindi sono arrivato all'ottimo!}$$

Trovare la soluzione ottima del rilassato continuo del caricamento intero è triviale.

$v(p) = 26, \bar{6}$, ho trovato un limite superiore al problema di caricamento.

$v_{PLI} \leq 26$, ho troncato all'unità visto che il problema ammette solo soluzioni intere.

Per trovare una stima inferiore prendo $\bar{x} = \left(0, 0, 0, 0, 0, \left\lfloor \frac{30}{18} \right\rfloor \right) = (0, 0, 0, 0, 0, 1)$, che è sicuramente una soluzione ammissibile.

$$\therefore v_i = 16 \leq v_{PLI} \leq 26 = v_s$$

$$Err = \frac{v_s - v_i}{v_i} = \frac{10}{16} \implies \sim 60\% \text{ di errore (schifo).}$$

Come trovare la valutazione superiore?

Considero $r_i := \frac{v_i}{p_i}$ i rendimenti. La soluzione ottima del rilassato continuo \bar{x}_{RC} si trova saturando lo zaino con il bene di massimo rendimento.

La valutazione superiore è la parte intera del valore ottimo del rilassato continuo: $v_s = \lfloor c^T \bar{x}_{RC} \rfloor$

1.2. Valutazione superiore dello zaino booleano

Il rilassato continuo dello zaino booleano è $(p) = \begin{cases} \max v^T x \\ p^T x \leq P \end{cases}$ (elimino anche il vincolo $x \geq 0$)

Considero $r_i := \frac{v_i}{p_i}$ i rendimenti. La soluzione ottima del rilassato continuo \bar{x}_{RC} si trova caricando

per primo il bene di massimo rendimento, per secondo il secondo bene di massimo rendimento, e così via. Al primo bene che non ci sta, si satura lo zaino con una frazione di quel bene.

1.3. Valutazioni inferiori e superiori dello zaino intero e booleano

Il passo base di ogni problema di PLI è la valutazione dei limiti inferiori e superiori. Ricapitoliamo i quattro algoritmi che ci permettono di risolvere il problema del caricamento:

Sia $c = (10, 17, 22, 21)$, $p = (4, 5, 6, 2)$, $P = 7$, i rendimenti $r = (2.5, 3.4, 3.7, 10.5)$

Problema	Valutazione inferiore	Valutazione superiore
Zaino booleano: $\max_{38 \leq v_{PLI} \leq 39}$ $err = \frac{1}{38} = 2.6\%$	Algoritmo greedy: carico i beni di massimo rendimento finché non ci stanno più. $\bar{x}_A = (0, 1, 0, 1)$ $v_i = c^T \bar{x}_A = 38$	Rilassato continuo: carico i beni di massimo rendimento finché non ci stanno più; sature lo zaino con una frazione dell'elemento che non ci sta. $\bar{x}_{RC} = \left(0, 0, \frac{5}{6}, 1\right)$ $v_{RC} = c^T \bar{x}_{RC} = 39.\bar{3}$, $v_s = \lfloor v_{RC} \rfloor = \lfloor 39.\bar{3} \rfloor = 39$
Zaino intero: $\max_{63 \leq v_{PLI} \leq 73}$ $err = \frac{10}{63} = 16\%$	Algoritmo greedy: aggiungo sempre il bene di massimo rendimento finché non ci sta più. $\bar{x}_A = (0, 0, 0, 3)$ \bar{x}_A si arrotonda per difetto. $v_i = c^T \bar{x}_A = 63$	Rilassato continuo: sature lo zaino con il bene di massimo rendimento. $\bar{x}_{RC} = \left(0, 0, 0, \frac{7}{2}\right)$ $v_{RC} = c^T \bar{x}_{RC} = 73.5$, $v_s = \lfloor v_{RC} \rfloor = \lfloor 73.5 \rfloor = 73$

Un **algoritmo greedy** è facile, veloce, buono (statisticamente vicino al valore ottimo).

$$\text{Produzione intera: } \begin{cases} \max c^T x & \max \\ Ax \leq b & : v_i \leq v_{PLI} < v_s, \\ x \in \mathbb{Z}_n^+ & \end{cases} \quad \begin{cases} \min c^T x & \min \\ Ax \geq b & : v_i \leq v_{PLI} < v_s \\ x \in \mathbb{Z}_n^+ & \end{cases}$$

Attenzione: l'arrotondamento non è sempre l'algoritmo corretto per trovare una soluzione ammissibile. Non funziona per i problemi di produzione intera o booleana con più di un vincolo. Il problema dello zaino è un problema di produzione, ma il contrario non è vero in generale. Il metodo per trovare una soluzione ammissibile cambia a seconda del problema.

1.4. Problemi particolari

1.4.1. Problema dei costruttori

Su Clash of Clans gli aggiornamenti degli edifici avvengono in un certo numero di giorni. Sia $c = (6, 8, 9, 11, 13, 16, 18)$, il numero di giorni che impiega ciascun edificio. $\sum c_i = 81$

Ho k costruttori: come assegno i lavori per impiegare il minor tempo possibile? Questo può sembrare un problema di assegnamento, ma in realtà si tratta di uno zaino binario.

$$\begin{cases} \max 6x_1 + 8x_2 + 9x_3 + 11x_4 + 13x_5 + 16x_6 + 18x_7 \\ 6x_1 + 8x_2 + 9x_3 + 11x_4 + 13x_5 + 16x_6 + 18x_7 \leq \frac{81}{k} \text{ (per semplicità, } 81\%k = 0) \\ x \in \{0, 1\}^7 \end{cases}$$

1.4.2. Problema del multizaino

Devo tenere in considerazione, oltre al peso degli oggetti, anche il loro volume.

$$\begin{cases} \max c^T x \\ p^T x \leq P \\ v^T x \leq V \\ x \in \{0, 1\} \end{cases}$$

- La v_s si trova, come al solito, con il rilassato continuo.
- La v_i è molto difficile da trovare.

Questo problema è NP-completo e, addirittura, la variante booleana non ammette un algoritmo polinomiale che fornisce soluzioni approssimate.

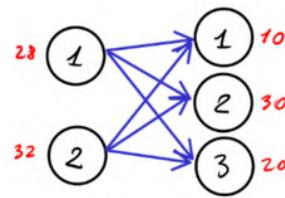
2. Problema di trasporto (e assegnamento intero)

Supponiamo che l'ottimo del rilassato continuo sia a componenti intere: $\bar{x}_{RC} \in \mathbb{Z}_n^+$.

Allora \bar{x}_{RC} è l'ottimo del problema di PLI poiché $c^T \bar{x}_{RC} \leq v_{PLI} \leq c^T \bar{x}_{RC}$.

Prendo un esempio di poliedro di un problema di trasporto:

$$\begin{cases} x_{11} + x_{12} + x_{13} = 28 \\ x_{21} + x_{22} + x_{23} = 32 \end{cases} \text{ (produzione)} \\ \begin{cases} x_{11} + x_{21} = 10 \\ x_{12} + x_{22} = 30 \\ x_{13} + x_{23} = 20 \end{cases} \text{ (destinazione)} \\ x_{ij} \geq 0$$



Sia $T := \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$ la matrice di trasporto.

$\text{rank}(T) = 4$, si può verificare vedendo che il determinante di tutti i minori 5×5 (sono 6) fa zero, e trovando un minore 4×4 che abbia determinante diverso da zero. Di conseguenza, un'equazione dei vincoli è eliminabile.

TEOREMA.

Teorema (unimodularità della matrice di trasporto):

Sia T una matrice di trasporto. Tutti i minori di T di rango massimo invertibili hanno determinante $+1$ o -1 .

Nell'esempio, tutti i minori 4×4 di T invertibili sono unimodulari.

Osservazione: le sottomatrici di rango massimo invertibili nella PL sono le matrici di base. Quindi tutte le matrici di base del trasporto sono unimodulari.

Allora la soluzione di base $x = A_B^{-1}b_B$ è a componenti intere (si può verificare risolvendo con il metodo di Cramer, per il quale devo dividere per il determinante $+1$ o -1). Se tutte le soluzioni di base del trasporto sono a componenti intere, allora tutti i vertici sono a componenti intere.

Proposizione.

L'ottimo del trasporto rilassato continuo è a componenti intere, quindi coincide con l'ottimo del trasporto intero.

L'assegnamento è un caso particolare di trasporto, quindi anch'esso gode della proprietà di interezza. Che l'assegnamento sia cooperativo o no, può essere risolto con linprog!

3. Problema del bin packing

Il problema del bin packing è un problema di ottimizzazione, in cui oggetti di diversi pesi devono essere inseriti in un numero finito di contenitori, ciascuno con una capacità, in modo da minimizzare il numero di contenitori utilizzati. In questo esempio ci sono solo contenitori con capacità 10.

Capacità: 10	$p_1 = 3$	$p_2 = 6$	$p_3 = 5$	$p_4 = 4$	$p_5 = 4$	$p_6 = 8$
--------------	-----------	-----------	-----------	-----------	-----------	-----------

$$x_{ij} := \begin{cases} 1, & \text{il contenitore } i\text{-esimo contiene l'oggetto } j\text{-esimo} \\ 0, & \text{altrimenti} \end{cases}$$

Si trova facilmente una soluzione ammissibile $v_s = 4$ prendendo i primi 2 elementi e mettendoli nel contenitore; gli ultimi 2 non ci stanno e quindi vanno in 2 contenitori diversi. Per questo so con certezza che il numero di contenitori è (al massimo) 4.

$$\left\{ \begin{array}{l} \min y_1 + y_2 + y_3 + y_4 \\ \begin{array}{l} x_{11} + x_{21} + x_{31} + x_{41} = 1 \\ x_{12} + x_{22} + x_{32} + x_{42} = 1 \\ x_{13} + x_{23} + x_{33} + x_{43} = 1 \\ \dots \\ x_{16} + x_{26} + x_{36} + x_{46} = 1 \end{array} \\ \begin{array}{l} 3x_{11} + 6x_{12} + 5x_{13} + 4x_{14} + 4x_{15} + 8x_{16} \leq 10y_1 \\ 3x_{21} + 6x_{22} + 5x_{23} + 4x_{24} + 4x_{25} + 8x_{26} \leq 10y_2 \\ \dots \\ 3x_{31} + 6x_{32} + 5x_{33} + 4x_{34} + 4x_{35} + 8x_{36} \leq 10y_3 \\ 3x_{41} + 6x_{42} + 5x_{43} + 4x_{44} + 4x_{45} + 8x_{46} \leq 10y_4 \end{array} \\ x \in \{0, 1\}, y \in \{0, 1\} \end{array} \right. \begin{array}{l} \text{vincoli sugli oggetti} \\ \text{vincoli sui contenitori} \end{array}$$

$$\text{Ho introdotto } y_i := \begin{cases} 1, & \text{il contenitore } i\text{-esimo viene usato} \\ 0, & \text{altrimenti} \end{cases}$$

$10y_1, \dots, 10y_4$ serve per evitare che l'ottimizzatore prenda come soluzione ammissibile $y = 0$.

3.1. Valutazione inferiore

La valutazione inferiore si trova con il rilassato continuo: $v_i = \left\lceil \frac{\sum_{j=1}^n p_j}{P} \right\rceil = 3$

3.2. Valutazione superiore

La valutazione superiore v_s si trova ordinando gli oggetti per peso decrescente: 8 6 5 4 4 3

Procedo con uno di questi tre algoritmi:

- **Next-Fit Decreasing:** Inserisci gli oggetti in un contenitore finché c'è spazio. Quando un oggetto non ci sta, chiudi il contenitore e aprine uno nuovo.
- **First-Fit Decreasing:** Per ogni oggetto, cerca il primo contenitore disponibile con abbastanza spazio. Se non ci sta, apri un nuovo contenitore.
- **Best-Fit Decreasing:** Inserisci ogni oggetto nel contenitore che ha lo spazio minimo sufficiente, riducendo al massimo lo spazio libero rimasto.

Next-Fit Decreasing:

Cont	Ogg	Peso	Sfrido
1	1	8	2
2	2	6	4
3	3, 4	5, 4	1
4	5, 6	4, 3	3

First-Fit Decreasing:

Cont	Ogg	Peso	Sfrido
1	1	8	2
2	2, 4	6, 4	0
3	3, 5	5, 4	1
4	6	3	7

Best-Fit Decreasing:

Cont	Ogg	Peso	Sfrido
1	1	8	2
2	2, 4	6, 4	0
3	3, 5	5, 4	1
4	6	3	7

4. Algoritmo di riduzione del gap

4.1. Risultati teorici

Assumiamo che A, b sono a componenti intere.

Sia (pli) $\begin{cases} \max c^T x \\ Ax \leq b \\ x \in \mathbb{Z}_n^+ \end{cases}$ un problema dello zaino. Sia (p) $\begin{cases} \max c^T x \\ Ax \leq b \end{cases}$ il suo rilassato continuo.

- Sia $S := \{x \in \mathbb{Z}_n^+ : Ax \leq b\}$ la regione ammissibile di (pli) . S è un insieme finito di punti. Se lo zaino fosse binario, ci sarebbero al massimo 2^n punti (vertici del cubo unitario).
- Sia $P = \{Ax \leq b, x \geq 0\}$ il poliedro del rilassato continuo.

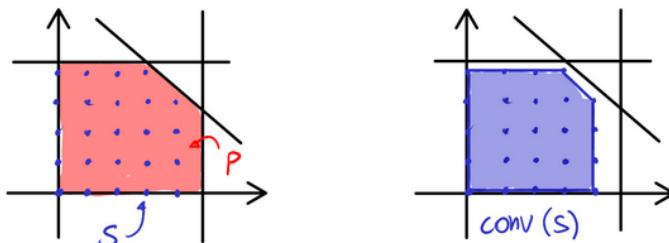


Figura 1: regione ammissibile di (pli) è contenuta in quella del suo rilassato continuo (p) .

So che il più piccolo insieme convesso che contiene S è $\text{conv}(S)$ (assumendo che sia limitato). Per il teorema di Weil, $\text{conv}(S)$ è un poliedro.

Mi accorgo di questa relazione: $S \subset \text{conv}(S) \subset P$

TEOREMA.

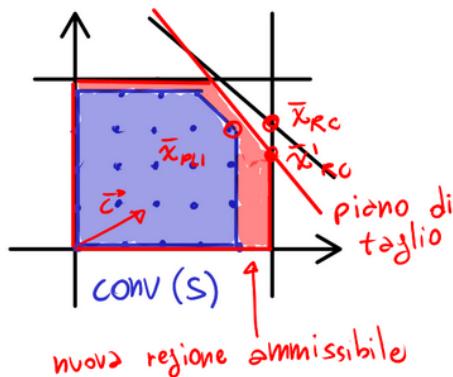
Teorema (astratto di equivalenza tra la PL e la PLI):

$$\text{Siano } A, b \text{ a componenti intere. Allora (pli)} \left\{ \begin{array}{l} \max c^T x \\ x \in S \end{array} \right. = (\text{pl}) \left\{ \begin{array}{l} \max c^T x \\ x \in \text{conv}(S) \end{array} \right.$$

La dimostrazione, che non vedremo, si basa sul fatto che $\text{conv}(S)$ ha i vertici a componenti intere. Come mai si chiama teorema astratto? Risolvere un problema di PL è facile e quindi questo teorema sarebbe estremamente utile, se non fosse per il fatto che non esiste un algoritmo per costruire $\text{conv}(S)$. Però ci dà l'idea fondamentale per l'algoritmo di riduzione del gap.

4.2. Piani di taglio

Idea: separo l'ottimo del rilassato continuo da quello di $\text{conv}(S)$ con dei **piani di taglio**.



Il piano di taglio è un vincolo che si aggiunge al rilassato continuo. La soluzione ottima del rilassato continuo trovata sulla nuova regione ammissibile è più vicino all'ottimo del problema di PLI.

Definizione.

Definizione (disuguaglianza valida):

Una disequazione $\gamma x \leq \gamma_0$ è detta disuguaglianza valida (D.V.).

Si dimostra facilmente che se $\gamma x \leq \gamma_0$ vale per S , allora vale anche per $\text{conv}(S)$.

Definizione.

Definizione (piano di taglio):

Sia \bar{x}_{RC} la soluzione ottima del rilassato continuo. Una D.V. $\gamma x \leq \gamma_0$ tale che $\gamma \bar{x}_{RC} > \gamma_0$ è detta piano di taglio (P.T.).

$$\begin{cases} \max c^T x \\ Ax \leq b \rightarrow v_{p \text{ new}} < v_s \Rightarrow v_i \leq v_{PLI} \leq \lfloor v_{p \text{ new}} \rfloor \\ \gamma x \leq \gamma_0 \end{cases}$$

Procedo iterativamente: se $v_{p \text{ new}}$ è a componenti intere, vuol dire che sono arrivato all'ottimo.

4.3. Algoritmo riduzione del gap

Erano gli anni 60 e Ralph E. Gomory, uno dei massimi dirigenti di IBM a New York, stava per lasciare l'azienda. Tuttavia, IBM gli offriva la libertà di lavorare sui progetti che preferiva e uno stipendio di 3 milioni di dollari, convincendolo a restare. In seguito, inventò l'algoritmo di riduzione del gap. Il giovane Pappalardo, un giorno entrò nell'ufficio di un suo alunno per un incontro. Lo trovò lì, con i piedi comodamente appoggiati sulla scrivania: una tipica abitudine americana.

Per facilità, supponiamo che il problema di PLI sia nella forma

$$\begin{cases} \max c^T x \\ Ax = b \\ x \geq 0 \\ x \in \mathbb{Z}^n \end{cases}$$

Sia \bar{x}_{RC} la soluzione ottima del suo rilassato continuo. La soluzione ottima, essendo un vertice, è generata da una base. $A = [A_B \ A_N]$, $x = \begin{bmatrix} x_B \\ x_N \end{bmatrix}$

$$\begin{array}{c} \text{---}^m \\ | \quad | \\ n \quad m-n \\ \boxed{A_B} \quad \boxed{A_N} \end{array} \cdot \begin{array}{c} X \\ \text{---}^m \end{array} = \begin{array}{c} b \\ \text{---}^n \end{array}$$

$n \times m \quad m \times 1 \quad n \times 1$

Figura 2: La soluzione di base rispetta i vincoli $A\bar{x}_{RC} = b$

Definizione.

Definizione (parte frazionaria):

Sia $\alpha \in \mathbb{R}$. La parte frazionaria di α è $\{\alpha\} := \alpha - \lfloor \alpha \rfloor$.

Casi particolari: $\{\pi\} = \pi - 3$, $\{-\pi\} = -\pi + 4$

Sia $\tilde{A} := A_B^{-1} A_N$ e sia r l'indice di una componente di \bar{x}_{RC} non intera. Nota che esiste necessariamente una componente non intera poiché, altrimenti, saremmo all'ottimo del problema di PLI. Inoltre, $r \in B$ necessariamente.

TEOREMA.

Teorema (teorema di Gomory):

Se $(\bar{x}_{RC})_r \notin \mathbb{Z}$, allora $\sum_{j \in N} \{\tilde{a}_{rj}\} x_j \geq \{(\bar{x}_{RC})_r\}$ è un P.T.
dove \tilde{a}_{rj} è l'elemento di \tilde{A} situato alla r -esima riga e j -esima colonna.

Si osserva che \bar{x}_{RC} non verifica questa disequazione perché la parte a sinistra di \geq è sicuramente zero, mentre quella a destra è sicuramente maggiore di zero. Resta da dimostrare che il piano di taglio non taglia S e $\text{conv}S$, ma noi non lo faremo.

Esempio 4.1 fare un passo dell'algoritmo di riduzione del gap al seguente problema:

$$(p) \begin{cases} \max 8x_1 + 6x_2 \\ 12x_1 + 6x_2 \leq 55 \\ 6x_1 + 8x_2 \leq 65 \\ x \geq 0, \quad x \in \mathbb{Z}^n \end{cases}$$

$$\bar{x}_{RC} = \left(\frac{5}{6}, \frac{15}{2} \right),$$

$$v_i = c^T \lfloor \bar{x}_{RC} \rfloor = c^T (0, 7) = 42,$$

$$c^T \bar{x}_{RC} = \frac{40}{6} + \frac{90}{2} = 51, 6 \implies v_s = 51.$$

Quindi $42 \leq v_{PLI} \leq 51$.

Non mi serve più la funzione obiettivo perché ho trovato il vertice $\bar{x}_{RC} = \left(\frac{5}{6}, \frac{15}{2} \right)$.

Porto il poliedro in formato duale standard: $D := \begin{cases} 12x_1 + 6x_2 + x_3 = 55 \\ 6x_1 + 8x_2 + x_4 = 65 \\ x \geq 0 \end{cases}$

$$\bar{x}_{RC} = \left(\frac{5}{6}, \frac{15}{2}, 0, 0 \right), \quad B = \{1, 2\}, \quad A = \begin{bmatrix} 12 & 6 & 1 & 0 \\ 6 & 8 & 0 & 1 \end{bmatrix}, \quad \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array}$$

$$\tilde{A} := \begin{array}{cc} 1 & 12 \\ 2 & 2 \end{array} A_B^{-1} \begin{array}{cc} 1 & 34 \\ 2 & 2 \end{array} A_N = A_B^{-1} I = A_B^{-1} = \frac{1}{60} \begin{bmatrix} 8 & -6 \\ -6 & 12 \end{bmatrix} = \begin{array}{cc} 1 & 3 \\ 2 & 2 \end{array} \begin{bmatrix} 2/15 & -1/10 \\ -1/10 & 1/5 \end{bmatrix}$$

$r := 1$ (ma anche 2 va bene, non c'è nessuna regola anticiclo)

$$\text{Piano di taglio: } \left\{ \frac{2}{15} \right\} x_3 + \left\{ -\frac{1}{10} \right\} x_4 \geq \left\{ \frac{5}{6} \right\} \iff \frac{2}{15} x_3 + \frac{9}{10} x_4 \geq \frac{5}{6}$$

N.B. Per semplicità di notazione, mi riferisco alle righe di \tilde{A} usando i nomi degli indici di base (1,2), e alle colonne con gli indici non di base (3,4). Questo perché, anziché indicizzare A_N partendo da 1, uso gli indici reali corrispondenti alle variabili del problema.

Esempio 4.2 fare un passo dell'algoritmo di riduzione del gap al seguente problema:

$$(p) \begin{cases} \max x_1 + 19x_2 \\ 12x_1 + 6x_2 \leq 53 \\ 6x_1 + 8x_2 \leq 65 \\ x \geq 0, x \in \mathbb{Z}^n \end{cases}$$

$$\bar{x}_{RC} = \left(0, \frac{65}{8}\right). \text{ Il poliedro in formato duale è } D = \begin{cases} 12x_1 + 6x_2 + x_3 = 55 \\ 6x_1 + 8x_2 + x_4 = 65 \\ x \geq 0 \end{cases}$$

$$\Rightarrow \bar{x}_{RC} = \left(0, \frac{65}{8}, \frac{25}{4}, 0\right)$$

$$A = \begin{array}{c} \textcolor{red}{1} \ 2 \ 3 \ 4 \\ \hline \textcolor{red}{2} \ 12 \ 6 \ 1 \ 0 \\ \textcolor{red}{2} \ 6 \ 8 \ 0 \ 1 \end{array}, \quad A_B = \begin{array}{c} \textcolor{red}{1} \ 2 \ 3 \\ \hline \textcolor{red}{2} \ 6 \ 1 \\ \textcolor{red}{2} \ 8 \ 0 \end{array}, \quad A_N = \begin{array}{c} \textcolor{red}{1} \ 4 \\ \hline \textcolor{red}{2} \ 12 \ 0 \\ \textcolor{red}{2} \ 6 \ 1 \end{array}, \quad \tilde{A} := \begin{array}{c} \textcolor{orange}{1} \ \textcolor{orange}{4} \\ \hline \textcolor{yellow}{2} \ 3/4 \ 1/8 \\ \textcolor{yellow}{3} \ 15/2 \ -3/4 \end{array}$$

$$\text{Il piano di taglio per } r := 2 \text{ è } \frac{3}{4}x_1 + \frac{1}{8}x_4 \geq \frac{1}{8}$$

ALGORITMO.

Algoritmo di riduzione del gap

1. Calcolo l'ottimo del rilassato continuo \bar{x}_{RC}
2. Porto il poliedro in formato duale standard e (ri)calcolo \bar{x}_{RC}
3. Individuo la base che genera \bar{x}_{RC} , individuo A_B, A_N, r
4. Calcolo \tilde{A}
5. Scrivo il piano di taglio $\sum_{j \in N} \{\tilde{a}_{rj}\} x_j \geq \{(\bar{x}_{RC})_r\}$

Aggiungo il piano di taglio ai vincoli del problema e torno al punto 1.

Esempio 4.3 Fare un passo dell'algoritmo di riduzione del gap al seguente problema:

$$(p) \begin{cases} \max 30x_1 + 36x_2 + 27x_3 + 20x_4 + 24x_5 + 22x_6 \\ 13x_1 + 16x_2 + 14x_3 + 15x_4 + 17x_5 + 14x_6 \leq 57 \\ x \in \mathbb{Z}_n^+ \end{cases}$$

1. $\bar{x}_{RC} \in \mathbb{R}^6$, $\bar{x}_{RC} = \left(\frac{57}{3}, 0, 0, 0, 0, 0 \right) \Rightarrow v_s = c^T \bar{x}_{RC} = 131$, $v_i = c^T [\bar{x}_{RC}] = 120$
 $\therefore 120 \leq v_{PLI} \leq 131$
2. Il poliedro in formato duale, $D := \begin{cases} 13x_1 + 16x_2 + 14x_3 + 15x_4 + 17x_5 + 14x_6 + x_7 \leq 57 \\ x \geq 0 \end{cases}$
Ricalcolo $\bar{x}_{RC} \in \mathbb{R}^7$, $\bar{x}_{RC} = \left(\frac{57}{3}, 0, 0, 0, 0, 0, 0 \right)$,
3. $B = \{1\}$, $N = \{2, 3, 4, 5, 6, 7\}$, $A_B = [13]$, $A_N = [16 \ 14 \ 15 \ 17 \ 14 \ 1]$, $r := 1$
4. $\tilde{A} = A_B^{-1} A_N = \left[\begin{array}{cccccc} \frac{16}{13} & \frac{14}{13} & \frac{15}{13} & \frac{17}{13} & \frac{14}{13} & \frac{1}{13} \end{array} \right]$
5. $\frac{3}{13}x_2 + \frac{1}{13}x_3 + \frac{2}{13}x_4 + \frac{4}{13}x_5 + \frac{1}{13}x_6 + \frac{1}{13}x_7 \leq \frac{5}{13} \Leftrightarrow$
 $3x_2 + x_3 + 2x_4 + 4x_5 + x_6 + x_7 \leq 5$ è il piano di taglio.

Esempio 4.4 Fare un passo dell'algoritmo di riduzione del gap al seguente problema:

$$(p) \begin{cases} \max 30x_1 + 36x_2 + 27x_3 + 20x_4 + 24x_5 + 22x_6 \\ 13x_1 + 16x_2 + 14x_3 + 15x_4 + 17x_5 + 15x_6 \leq 57 \\ x \in \{0, 1\}^n \end{cases}$$

1. $\bar{x}_{RC} \in \mathbb{R}^6$, $\bar{x}_{RC} = \left(1, 1, 1, 0, 0, \frac{14}{15} \right)$
2. Attenzione: $D \neq \begin{cases} 13x_1 + 16x_2 + 14x_3 + 15x_4 + 17x_5 + 15x_6 + x_7 = 57 \\ x \geq 0 \end{cases}$

Quando passo al duale, devo aggiungere il vincolo $0 \leq x \leq 1$.

$$D = \begin{cases} 13x_1 + 16x_2 + 14x_3 + 15x_4 + 17x_5 + 15x_6 + x_7 = 57 \\ x_1 + x_8 = 1 \\ x_2 + x_9 = 1 \\ x_3 + x_{10} = 1 \\ x_4 + x_{11} = 1 \\ x_5 + x_{12} = 1 \\ x_6 + x_{13} = 1 // \text{non devo aggiungere anche } x_7 \text{ poiché è la variabile di scarto.} \\ x \geq 0 \end{cases}$$

$$\therefore \bar{x}_{RC} = \left(1, 1, 1, 0, 0, \frac{14}{15}, 0, 0, 0, 0, 1, 1, \frac{1}{15} \right)$$

$$3. B = \{1, 2, 3, 6, 11, 12, 13\}, \#B = 7; \quad N = \{4, 5, 7, 8, 9, 10\}, \#N = 6; \quad r := 6$$

$$A = \begin{array}{c|cccccccccccccc} & \textcolor{red}{1} & \textcolor{red}{2} & \textcolor{red}{3} & \textcolor{red}{4} & \textcolor{red}{5} & \textcolor{red}{6} & \textcolor{red}{7} & \textcolor{red}{8} & \textcolor{red}{9} & \textcolor{red}{10} & \textcolor{red}{11} & \textcolor{red}{12} & \textcolor{red}{13} \\ \hline \textcolor{red}{1} & 15 & 16 & 14 & 15 & 17 & 15 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \textcolor{red}{2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \textcolor{red}{3} & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \textcolor{red}{4} & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \textcolor{red}{5} & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ \textcolor{red}{6} & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \textcolor{red}{7} & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array}$$

$$A_B = \begin{array}{c|cccccc} & \textcolor{red}{1} & \textcolor{red}{2} & \textcolor{red}{3} & \textcolor{red}{6} & \textcolor{red}{11} & \textcolor{red}{12} & \textcolor{red}{13} \\ \hline \textcolor{red}{1} & 15 & 16 & 14 & 15 & 0 & 0 & 0 \\ \textcolor{red}{2} & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \textcolor{red}{3} & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \textcolor{red}{4} & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \textcolor{red}{5} & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \textcolor{red}{6} & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ \textcolor{red}{7} & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{array} \quad (7 \times 7), \quad A_N (7 \times 6)$$

$$4. \widetilde{A} (7 \times 6) = \begin{bmatrix} & \textcolor{red}{4} & \textcolor{red}{5} & \textcolor{red}{7} & \textcolor{red}{8} & \textcolor{red}{9} & \textcolor{red}{10} \\ \textcolor{red}{1} & A_B^{-1} & \left[\begin{array}{c|cccccc} & \textcolor{red}{4} & \textcolor{red}{5} & \textcolor{red}{7} & \textcolor{red}{8} & \textcolor{red}{9} & \textcolor{red}{10} \\ \hline \textcolor{red}{2} & & & & & & \\ \textcolor{red}{3} & & & & & & \\ \textcolor{red}{6} & & x & x & x & x & x \\ \textcolor{red}{11} & & x & x & x & x & x \\ \textcolor{red}{12} & & x & x & x & x & x \\ \textcolor{red}{13} & & x & x & x & x & x \end{array} \right] & \text{non lo continuo...} \\ \textcolor{red}{(7 \times 7)} & (7 \times 6) & \end{bmatrix}$$

Esempio 4.5 Fare un passo dell'algoritmo di riduzione del gap al seguente problema:

$$(p) \begin{cases} \min 5x_1 + 14x_2 \\ 16x_1 + 13x_2 \geq 62 \\ 6x_1 + 15x_2 \geq 52 \\ x \in \mathbb{Z}_n^+ \end{cases}$$

$$1. \bar{x}_{RC} = \left(\frac{26}{3}, 0 \right) \text{ risolto con linprog.}$$

$$2. D := \begin{cases} 16x_1 + 13x_2 - x_3 = 62 \\ 6x_1 + 15x_2 - x_4 = 52 * , \bar{x}_{RC} = \left(\frac{26}{3}, 0, x_3, 0 \right) \\ x \geq 0 \end{cases}$$

$$\text{Non mi interessa sapere } x_3 = \frac{26}{3} \cdot 16 - 62 \text{ perché tanto ho già l'indice } r := 1$$

3. $B = \{1, 3\}$, $A_B = \begin{bmatrix} 16 & -1 \\ 6 & 0 \end{bmatrix}$, $A_N = \begin{bmatrix} 13 & 0 \\ 15 & -1 \end{bmatrix}$, $r := 1$

4. $\tilde{A} = A_B^{-1} A_N = \begin{bmatrix} 15/6 & -1/6 \\ \dots & \dots \end{bmatrix}$, non mi interessa sapere la seconda riga in quanto $r = 1$

5. P.T. $\frac{3}{6}x_2 + \frac{5}{6}x_4 \geq \frac{2}{3} \Leftrightarrow 3x_2 + 5x_4 \geq 2$

Attenzione: $3x_2 + 5x_4 \geq 2 \Leftrightarrow 5x_1 + 13x_2 \geq 44$ è lo stesso piano di taglio!

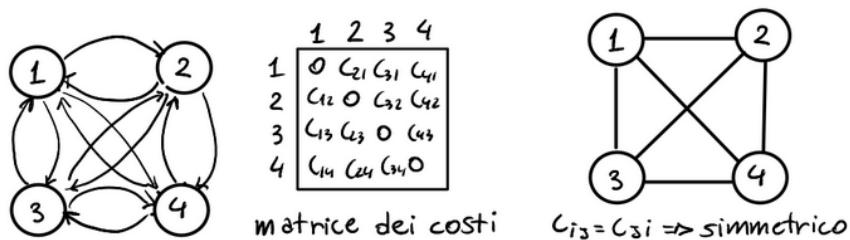
Si ottiene ricavando la variabile di scarto x_4 dalle x_1, x_2 :

* $x_4 = 6x_1 + 15x_2 - 52$, quindi $3x_2 + 5(6x_1 + 15x_2 - 52) \geq 4 \Rightarrow \dots \Rightarrow 5x_1 + 13x_2 \geq 44$

5. Problema del commesso viaggiatore

TSP: Travelling Salesman Problem. Dato un insieme di città, e note le distanze tra ciascuna coppia di esse, trovare il tragitto di minima percorrenza che un commesso viaggiatore deve seguire per visitare tutte le città una ed una sola volta e ritornare alla città di partenza.

Dato un grafo (N, A) completo, con c_{ij} costi sugli archi, trova un ciclo di costo minimo che passi su tutti i nodi una ed una sola volta (ciclo hamiltoniano).



5.1. TSP asimmetrico

Le soluzioni ammissibili sono tutte le permutazioni dei nodi, ovvero $n!$

Introduco $x_{ij} = \begin{cases} 1, & \text{il nodo } (i, j) \text{ viene utilizzato} \\ 0, & \text{altrimenti} \end{cases}$

La funzione da minimizzare è $c^T x$, dove $x \in \mathbb{Z}^{n^2}$ (è didatticamente più semplice se si lasciano gli archi $(i, j) : i = j$).

Ogni nodo deve avere un solo arco uscente:

$$\begin{cases} x_{11} + x_{12} + x_{13} + x_{14} = 1 \\ x_{21} + x_{22} + x_{23} + x_{24} = 1 \\ x_{31} + x_{32} + x_{33} + x_{34} = 1 \\ x_{41} + x_{42} + x_{43} + x_{44} = 1 \end{cases}$$

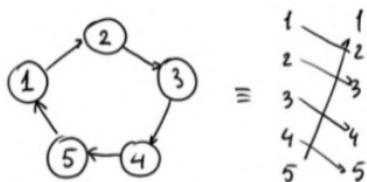
Ogni nodo deve avere un solo arco entrante:

$$\begin{cases} x_{11} + x_{21} + x_{31} + x_{41} = 1 \\ x_{12} + x_{22} + x_{32} + x_{42} = 1 \\ x_{13} + x_{23} + x_{33} + x_{43} = 1 \\ x_{14} + x_{24} + x_{34} + x_{44} = 1 \end{cases}$$

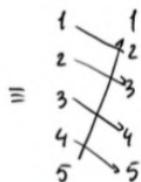
Con solo questi vincoli il problema del commesso viaggiatore pare un problema di assegnamento, che è banale da risolvere.

Proposizione.

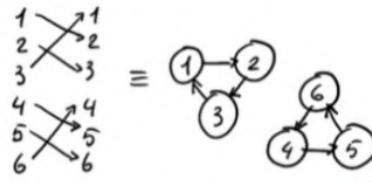
Tutti i cicli hamiltoniani sono assegnamenti, ma non tutti gli assegnamenti sono cicli hamiltoniani. Tutti gli assegnamenti sono unione di cicli hamiltoniani disgiunti.



Ciclo hamiltoniano.



Assegnamento.



Assegnamento.

Cicli hamiltoniani
disgiunti.

Figura 3: Proposizione di sopra

Il vincolo che bisogna aggiungere è il vincolo di **connessione**. Il problema del commesso viaggiatore è **difficile**, NP completo.

Nell'esempio della Figura 3: $x_{14} + x_{15} + x_{16} + x_{24} + x_{25} + x_{26} + x_{34} + x_{35} + x_{36} \geq 1$ dev'esserci almeno un collegamento tra i due cicli disgiunti.

In generale, questa condizione si esprime chiedendo che qualunque sottoinsieme dei nodi con qualunque complementare devono essere sempre connessi.

Sia N l'insieme dei nodi: $\sum_{\substack{i \in S \\ j \notin S}} x_{ij} \geq 1 \quad \forall S \subseteq N, S \neq \emptyset, S \neq N, 1 < |S| < n - 1$

Posso togliere i casi particolari $|S| = 0, |S| = 1, |S| = n - 1, |S| = |N|$, ovvero $2n + 2$ vincoli, ma la complessità è comunque esponenziale. Infatti, ci sono 2^n sottoinsiemi S di N .

\therefore In totale, si aggiungono $2^n - 2n - 2$ vincoli.

Modello.

TSP asimmetrico:

$$\begin{cases} \min c^T x \\ \sum_{i \in N \setminus \{j\}} x_{ij} = 1 \quad \forall j \in N \\ \sum_{j \in N \setminus \{i\}} x_{ij} = 1 \quad \forall i \in N \\ \sum_{i \in S, j \notin S} x_{ij} \geq 1 \quad \forall S \subseteq N, S \neq \emptyset, N \\ x_{ij} \in \{0, 1\} \end{cases}$$

Il problema è difficile perché non si riescono a trovare tutti i vincoli, ovvero trovare la soluzione ottima del rilassato continuo! Allora si tolgono i vincoli di connessione e si ottengono le stime inferiori e superiori in un altro modo.

5.1.1. Valutazione inferiore (assegnamento di costo minimo)

v_i si trova rilassando il problema di TSP in un problema di assegnamento di costo minimo. In generale, rilassare significa trovare un insieme sul quale è facile risolvere il problema che contiene la soluzione al problema di partenza.

5.1.2. Valutazione superiore (algoritmo delle toppe)

v_s si trova con l'algoritmo delle toppe. In alternativa, va bene un qualsiasi algoritmo greedy che fornisca una soluzione ammissibile.

Algoritmo.

Algoritmo delle toppe:

1. Calcolo l'assegnamento di costo minimo
2. Seleziono un arco (i, j) del primo ciclo e un arco (k, l) del secondo.
3. Li elimino e li sostituisco con (i, l) e (k, j) , rispettivamente.

L'algoritmo delle toppe è un algoritmo greedy che ci permette di ottenere un ciclo hamiltoniano.

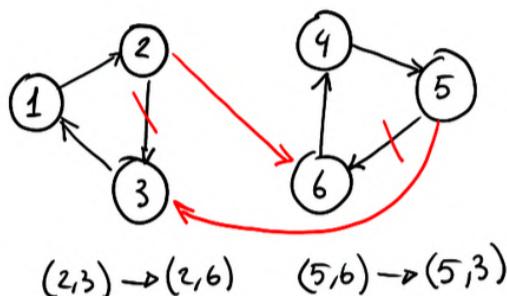


Figura 3: Algoritmo delle toppe, $(i, j) \rightarrow (i, l); (k, l) \rightarrow (k, j)$

Nella Figura 4: $v_{ass} - c_{23} - c_{56} + c_{26} + c_{53} = v_i \geq 0$ sicuramente, perché se fosse negativa il nuovo assegnamento sarebbe migliore, ma io ho già scelto l'assegnamento di costo minimo.

5.2. TSP simmetrico

Caso particolare di TSP. $c_{ij} = c_{ji}$ quindi si dimezzano le variabili $\frac{n^2 - n}{2} \in \mathbb{Z}$
 $x_{ij} : i < j$, nell'esempio di 5 nodi: $x = (x_{12}, x_{13}, x_{14}, x_{15}, x_{23}, x_{24}, x_{25}, x_{34}, x_{35}, x_{45})$

5.2.1. Valutazione superiore (algoritmo del nodo più vicino)

v_s si trova con l'algoritmo del nodo più vicino. Se parto dal nodo i , scelgo di collegarlo al nodo j tale che c_{ij} è il minimo.

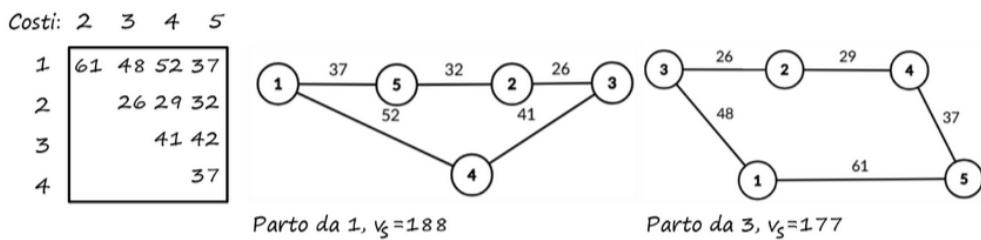


Figura 4: Partendo da nodi diverse non si ottiene la stessa soluzione, ma l'algoritmo mi garantisce che riesco sempre a trovare una valutazione superiore.

5.2.2. Valutazione inferiore (k -albero di costo minimo)

Nel caso simmetrico, non si può usare l'assegnamento di costo minimo.

k -albero: isolo il nodo k e costruisco un albero di copertura con i nodi rimanenti. Collego il nodo k all'albero di copertura con due archi.

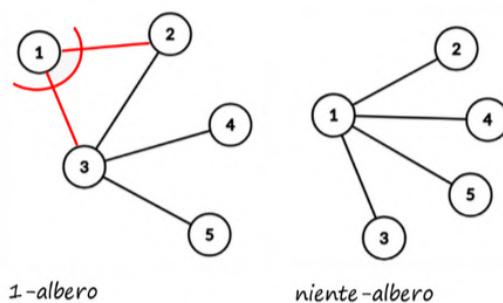


Figura 5: Un k -albero ha sempre n archi.

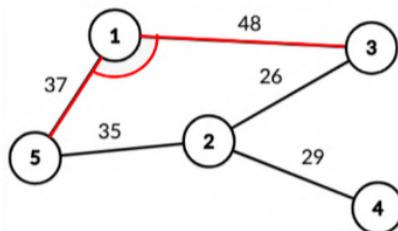
Per trovare il k -albero di costo minimo trovo il sottoalbero di copertura di costo minimo con l'algoritmo di Kruskal, poi aggiungo i due rami che collegano k meno costosi. Si può dimostrare che l'algoritmo di Kruskal è ottimo; siccome scelgo i rami di costo minimo, l'algoritmo appena descritto è ottimo.

Algoritmo.

Algoritmo di Kruskal:

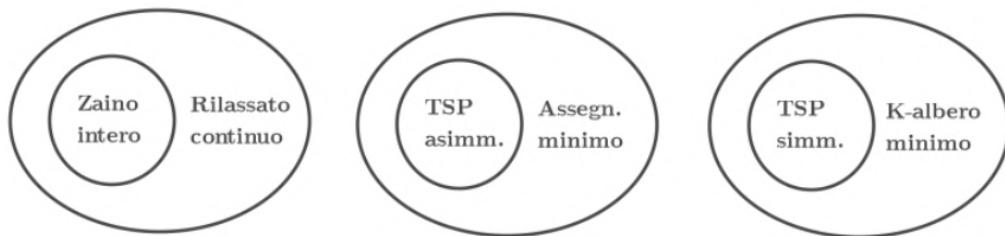
Per costruire l'albero di copertura di costo minimo (Minimum Spanning Tree)

1. Ordina tutti gli archi in ordine non decrescente in base al loro peso
2. Scegli l'arco più piccolo. Controlla se forma un ciclo con l'albero di copertura già formato. Se non si forma un ciclo, includi questo arco. Altrimenti, scartalo.
3. Ripeti il passo 2 finché non ci sono $(n - 1)$ archi nell'albero di copertura.



Ho trovato un rilassamento: ogni ciclo hamiltoniano è un k -albero. Se un k -albero è un ciclo hamiltoniano, allora sono all'ottimo! In generale, se l'ottimo di un rilassato è ammissibile, vuol dire che sono arrivato all'ottimo del problema di PLI.

Soluzioni ottime:



Modello.

TSP simmetrico:

$$\begin{cases} \min c^T \cdot x \\ \sum_{i < j} x_{ij} + \sum_{j < i} x_{ji} = 2 \quad \forall j & \leftarrow \text{vincoli di grado} \\ \sum_{\substack{i \in S, j \notin S, i < j}} x_{ij} + \sum_{\substack{i \notin S, j \in S, j < i}} \geq 1 \quad \forall j, S \subset N & \leftarrow \text{vincoli di connessione} \\ x_{ij} \in \{0, 1\} \end{cases}$$

Le variabili sono $|x| = \frac{n!}{|S|!(n - |S|)!}$

La matrice A_{eq} ha dimensione $n \times |x|$

La matrice A ha dimensione $(2^n - 2n - 2) \times |x|$

6. Algoritmo del Branch and Bound

Il Branch and Bound è un approccio che può essere applicato a una vasta gamma di problemi di PLI, inclusi problemi NP-hard come il TSP. Il TSP asimmetrico non può essere risolto con l'algoritmo di riduzione del gap perché non riesco a scrivere i vincoli.

Per semplicità considero solo problemi di zaino booleano e di TSP simmetrico.

6.1. Albero di enumerazione totale

Considero (p) $\begin{cases} \max x_1 + x_2 \\ 3x_1 + 6x_2 \leq 7 \\ x \in \{0, 1\} \end{cases}$ un problema di zaino intero.

Il problema (p) si può partizionare in due sottoproblemi (p_{11}) se $x_1 = 0$, (p_{12}) se $x_1 = 1$. I sottoproblemi (p_{11}), (p_{12}) possono essere partizionati allo stesso modo per $x_2 = 0$, $x_2 = 1$.

Posso disegnare un albero binario con quattro foglie, che rappresentano tutti i casi possibili. Sicuramente, la soluzione ottima si trova in una foglia.

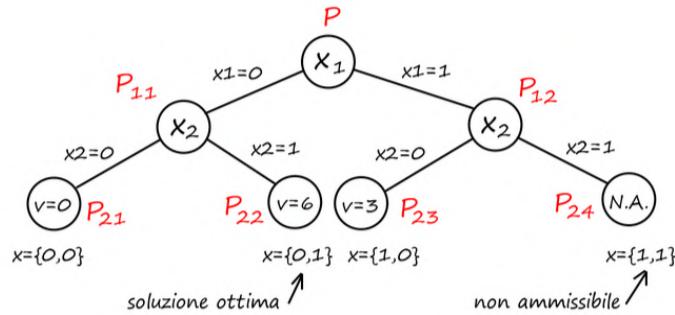


Figura 6: Albero di enumerazione totale di un problema di zaino binario a due variabili

In generale, se ho un problema dello zaino intero (con anche più di due variabili), il suo albero di enumerazione totale è:

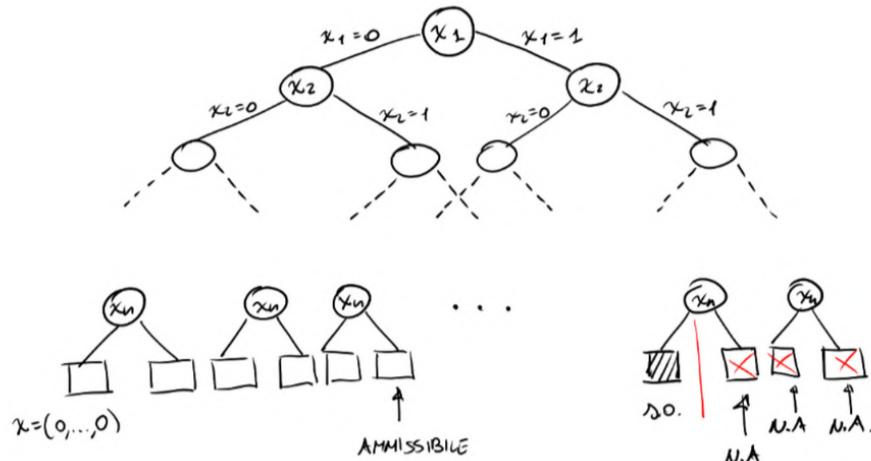


Figura 7: A partire da un certo punto, tutte le foglie sono inammissibili. Ci sono 2^n foglie.

Un metodo per risolvere un problema dello zaino (con anche più di due variabili) è il **metodo dell'enumerazione esplicita**: disegno l'albero di enumerazione totale e controllo le foglie per trovare la soluzione ottima. Questo metodo è esponenziale e quindi inefficiente.

6.2. Branch and Bound

L'algoritmo del Branch and Bound si basa sull'albero di enumerazione totale. Nonostante anch'esso sia esponenziale, è molto più efficiente perché si "taglano" i sottoalberi generati dai nodi oltre i quali sono sicuro che non ci sarà la soluzione ottima. L'algoritmo termina quando ho visitato, esplicitamente o implicitamente, tutte le foglie.

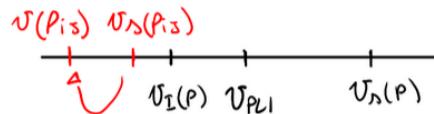
- **Branch:** partiziono la regione ammissibile per generare sottoproblemi (nel caso di zaino booleano è semplice, è un albero binario).
- **Bound:** stimo il valore dell'ottimo di ogni sottoproblema e utilizzo regole di taglio per depennare le sottoregioni che non contengono l'ottimo.

Corollario.

Regole di taglio per zaino booleano (max):

Sia P_{ij} il sottoproblema situato nell'albero a livello i e a posizione orizzontale j . Valgono le seguenti:

1. Se $P_{ij} = \emptyset$ allora devo chiudere il nodo P_{ij} perché tutto il sottoalbero è inammissibile.
2. Se $v_s(P_{ij}) \leq v_i(P)$ allora devo chiudere il nodo P_{ij} . Ogni passo dell'algoritmo richiede il calcolo di v_s dello zaino booleano con il metodo dei rendimenti



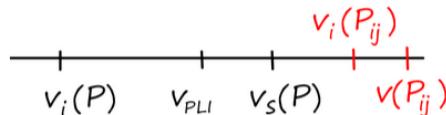
3. Se $v_s(P_{ij}) > v_i(P)$, con $v_s(P_{ij})$ ammissibile i.e. a componenti intere, allora devo aggiornare $v_i(P) = v_s(P_{ij})$ e chiudere il nodo P_{ij} (per la 2.)

Corollario.

Regole di taglio per TSP simmetrico (min):

Sia P_{ij} il sottoproblema situato nell'albero a livello i e a posizione orizzontale j . Valgono le seguenti:

1. Se $P_{ij} = \emptyset$ allora devo chiudere il nodo P_{ij} perché tutto il sottoalbero è inammissibile.
2. Se $v_i(P_{ij}) \geq v_s(P)$ allora devo chiudere il nodo P_{ij} . Ogni passo dell'algoritmo richiede il calcolo di v_i del TSP con il metodo dei K-alberi di costo minimo.



3. Se $v_i(P_{ij}) < v_s(P)$, con $v_i(P_{ij})$ ammissibile i.e. a componenti intere, allora devo aggiornare $v_s(P) = v_i(P_{ij})$ e chiudere il nodo P_{ij} (per la 2.)

Domanda: se ho un TSP simmetrico, ad ogni passo calcolo un nuovo ciclo hamiltoniano? No. Mi basta calcolare un k-albero. Infatti, per verificare se un k-albero è un ciclo hamiltoniano, basta verificare i vincoli di grado (quelli di connessione sono già verificati).

ALGORITMO.

Algoritmo del Branch and Bound

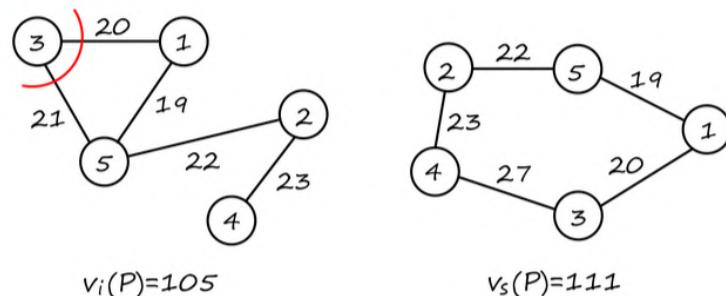
1. Calcola $v_i(P)$ se max, $v_s(P)$ se min.
2. Se tutti i nodi sono stati esplorati allora STOP
3. Seleziona il nodo (sottoproblema) da esplorare
4. Calcola $v_s(P_{ij})$ se max, $v_i(P_{ij})$ se min.
5. Controlla le regole di taglio.
6. Se il nodo non è chiuso, allora scendi e torna al passo 4.
7. Vai al passo 2.

Esempio 6.1 Sia data la matrice dei costi di un TSP simmetrico:

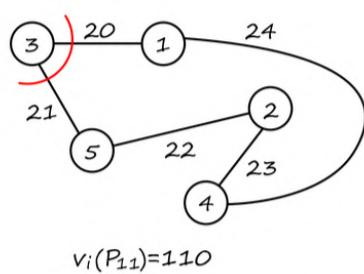
	2	3	4	5
1	26	20	24	19
2		34	23	22
3			27	21
4				32

➤ Fare un passo dell'algoritmo del Branch and Bound istanziando la variabile x_{15} .

Trovo innanzitutto le valutazioni inferiori e superiori. Faccio un k -albero di costo minimo e l'algoritmo del nodo più vicino.



- P_{11} : $x_{15} = 0$ i.e. non dev'esserci l'arco $(1, 5)$.

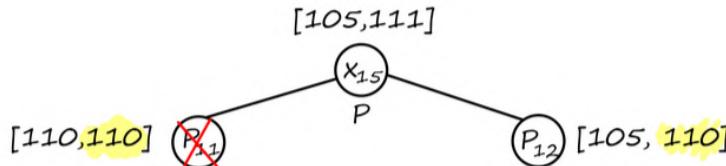


$110 = v_i(P_{11}) > v_s(P) = 111$. Siccome è anche un ciclo hamiltoniano, ovvero è ammissibile, devo aggiornare $v_s(P) = 110$ e chiudere il nodo P_{11}

- P_{12} : $x_{15} = 1$ i.e. dev'esserci l'arco $(1, 5)$.

Siccome c'era già quando avevo calcolato $v_i(P)$, allora

$105 = v_i(P) = v_i(P_{12}) < v_s(P) = 111$. Non è ammissibile (anche perché altrimenti prima sarei già arrivato all'ottimo), quindi non faccio niente.



➤ Fare un altro passo dell'algoritmo del Branch and Bound istanziando la variabile x_{35} .

- P_{23} : $x_{15} = 1, x_{35} = 0$ i.e. dev'esserci l'arco $(1, 5)$, non dev'esserci l'arco $(3, 5)$.

$v_i(P_{23}) = 111 \geq 110$, quindi chiudo il nodo P_{23} .

- P_{24} : $x_{15} = 1, x_{35} = 0$ i.e. devono esserci gli archi $(1, 5)$ e $(3, 5)$.

$v_i(P_{24}) = v_i(P_{12}) = 105$, ed è lo stesso caso di prima, in quanto l'arco $3, 5$ esiste già nella soluzione che genera $v_i(P_{12})$.

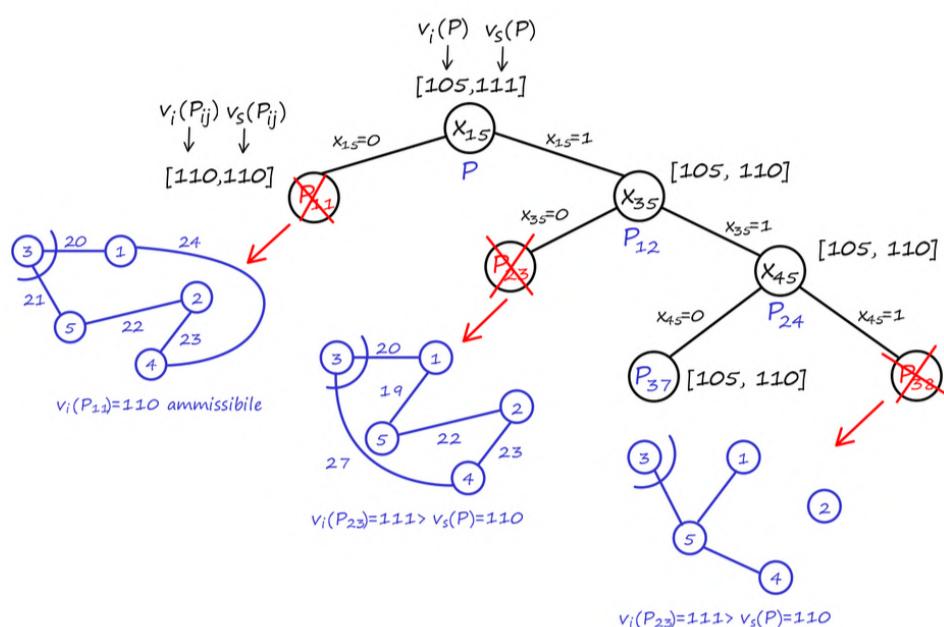
➤ Fare un altro passo dell'algoritmo del Branch and Bound istanziando la variabile x_{45} .

- P_{37} : $x_{15} = 1, x_{35} = 1, x_{45} = 0$ i.e. devono esserci $(1, 5)$ e $(3, 5)$, ma non $(4, 5)$.

$v_i(P_{37}) = v_i(P_{24}) = 105$, non cambia niente.

- P_{38} : $x_{15} = 1, x_{35} = 1, x_{45} = 1$ i.e. devono esserci $(1, 5), (3, 5)$ e $(4, 5)$.

Mi accorgo che il problema $P_{38} = \emptyset$ perché non esiste un k -albero con archi $(1, 5), (3, 5), (4, 5)$. Quindi chiudo il nodo P_{38} . **È un errore calcolare $v_i(P_{38})$.**



Esempio 6.2 Sia dato un problema di zaino booleano con:

$$\begin{aligned} v &= [7 \quad 21 \quad 17 \quad 6 \quad 22 \quad 19 \quad 8] \\ p &= [479 \quad 272 \quad 173 \quad 473 \quad 54 \quad 230 \quad 25], \quad c = 531 \end{aligned}$$

Trovo innanzitutto la valutazione superiore e inferiore:

$$r := (0.015, 0.08, 0.098, 0.013, 0.41, 0.083, 0.32)$$

$$x_{RC} = \left(0, 1, 1, 0, 1, \frac{57}{230}, 1\right) \Rightarrow v_s = \lfloor c \cdot x_{RC} \rfloor = 72$$

$$x_a = (0, 1, 1, 0, 1, 0, 1) \Rightarrow v_i = 68 \quad \therefore 68 \leq v_{PLI} \leq 72$$

➤ Fare passo dell'algoritmo del Branch and Bound.

Quale variabile mi conviene istanziare? La **variabile frazionaria del rilassato continuo**, x_6

$$P_{11}: \text{setto } x_6 = 0. \quad x_{RC} = \left(\frac{57}{479}, 1, 1, 0, 1, \cancel{0}, 1\right) \Rightarrow v_s(P_{11}) = \lfloor 68 + 7 \cdot (57 / 479) \rfloor = 68$$

$v_s(P_{11}) > v_i(P)$ ed è ammissibile, quindi setto $v_i(P) = v_s(P_{11})$ e chiudo il nodo P_{11}

$$P_{12}: \text{setto } x_6 = 1. \quad x_{RC} = \left(0, \frac{49}{222}, 1, 0, 1, \cancel{1}, 1\right) \Rightarrow v_s(P_{12}) = 70 > v_i(P).$$

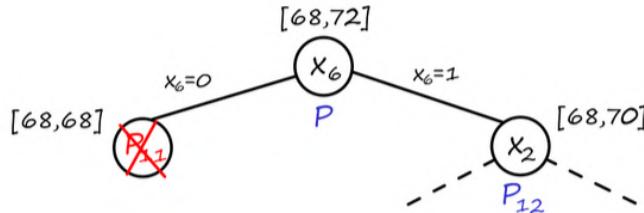


Figura 8: Se voglio fare un altro passo, devo istanziare la variabile frazionaria, ovvero x_2

7. Problema di copertura

7.1. Copertura di costo minimo

Dato un territorio suddiviso in 7 quartieri e 5 potenziali posizioni per un'ambulanza, l'obiettivo è posizionare il minor numero possibile di ambulanze, garantendo che tutti i quartieri siano coperti dal servizio di emergenza.

Quartieri	1	2	3	4	5	Depositi \mathbf{X}_i
1	1	1	0	0	1	
2	0	1	0	1	0	
3	1	0	0	0	0	
4	1	0	1	0	1	
5	1	0	0	1	1	
6	1	1	1	0	0	
7	1	0	1	1	0	
	c_1	c_2	c_3	c_4	c_5	Costi (opzionale)

A

$$i = 1 \dots 5, \quad x_i = \begin{cases} 0, & \text{non metto l'ambulanza nel deposito } i \\ 1, & \text{metto l'ambulanza nel deposito } i \end{cases}$$

$$\begin{cases} \min x_1 + x_2 + x_3 + x_4 + x_5 \\ (\text{oppure}) \min c_1 x_1 + c_2 x_2 + c_3 x_3 + c_4 x_4 + c_5 x_5 \quad \text{se le scelte hanno un costo.} \end{cases}$$

In ogni quartiere dev'esserci un'ambulanza che la possa servire.

$$\begin{cases} x_1 + x_2 + x_5 \geq 1 & \text{devo servire il quartiere 1} \\ x_2 + x_4 \geq 1 \\ x_1 \geq 1 \\ x_1 + x_3 + x_5 \geq 1 \\ x_1 + x_4 + x_5 \geq 1 \\ x_1 + x_2 + x_3 \\ x_1 + x_3 + x_4 \geq 1 \end{cases}$$

Modello.

Copertura di costo minimo:

$$(p) \begin{cases} \min c^T x \\ Ax \stackrel{\rightarrow}{\geq} 1 \\ x \in \{0, 1\}^n \end{cases}$$

7.1.1. Valutazione inferiore v_i

Si trova con il rilassato continuo

$$\begin{cases} \min c^T x \\ \sum_{i=1}^n a_{ij} x_j \geq 1 \\ 0 \leq x_j \leq 1 \end{cases}$$

7.1.2. Valutazione superiore v_s

Prima di tutto opero le regole di riduzione:

1. Una riga è tutta a zero \rightarrow problema vuoto
2. Una riga è tutta a 1 \rightarrow la tolgo
3. Una riga ha un solo 1 \rightarrow sono obbligato a prendere il sito corrispondente, cancello la colonna del sito e le righe dove la colonna ha $a_{ij} = 1$

	1	2	3	4	5
1	1	1	0	0	1
2	0	1	0	1	0
3	1	0	0	0	0
4	1	0	1	0	1
5	1	0	0	1	1
6	1	1	1	0	0
7	1	0	1	1	0

4. Regola della dominanza:

la riga 3 è "dominata" dalla riga 6.

Cancello la riga 5 perché devo lasciare in ballo le richieste di 4 perché sono più stringenti.

Def: Si dice che $a \in \mathbb{R}^m$ è dominato da $b \in \mathbb{R}^m$ se $a_i \leq b_i \forall i$

	1	2	3	4	5
1	1	1	0	0	1
2	0	1	0	1	0
3	1	0	0	0	0
4	1	0	1	0	1
5	1	0	0	1	1
6	1	1	1	0	0
7	1	0	1	1	0

"3" \leq "6"

La "3" è dominata dalla "6"

5. Regola dei costi:

se la colonna i è dominata dalla colonna j e $c_i > c_j$, elimino i .

	1	2	3	4	5
1	1	1	0	0	1
2	0	1	0	1	0
3	1	0	0	0	0
4	1	0	1	0	1
5	1	0	0	1	1
6	1	1	1	0	0
7	1	0	1	1	0

"3" \leq "1"

Elimino la colonna 3 se $c_3 > c_1$

$c=2 \quad c=10$

Una volta che ho ridotto il problema in uno più piccolo trovo una soluzione ammissibile.

Se tutte le scelte hanno costo 1, apro il servizio che soddisfa più clienti, lo pongo a 1 e cancello la sua colonna e tutte le righe da lui servite. Calcolo di nuovo il servizio che soddisfa più clienti e ripeto.

La soluzione ammissibile nel caso con i costi si trova con l'algoritmo di Chvatal:

- Costruisco il vettore dei costi unitari di copertura $u = \left(\frac{c_1}{6}, \frac{c_2}{3}, \frac{c_3}{3}, \frac{c_4}{3}, \frac{c_5}{3} \right)$
- Posiziono il servizio nel sito tale che r è il minimo. Elimino la colonna e le righe servite.

Supponiamo $\frac{c_2}{3} = \min(r)$. Allora elimino la colonna 2 e le righe 1, 2, 6.

	1	2	3	4	5
1	1	0	0	1	
2	0	0	1	0	
3	1	0	0	0	0
4	1	0	1	0	1
5	1	0	0	1	1
6	1	1	0	0	
7	1	0	1	1	0

- Ricalcolo i costi unitari di copertura $u = \left(\frac{c_1}{4}, \frac{c_3}{2}, \frac{c_4}{2}, \frac{c_5}{2} \right)$ e procedo iterativamente.

Algoritmo.

Algoritmo di Chvatal:

Per trovare una soluzione ammissibile ad un problema di copertura.

1. $I = \{1, \dots, m\}, J = \{1, \dots, n\}, x := 0$
2. Per ogni $j \in J$ calcola $u_j = \frac{c_j}{\sum_{i \in I} a_{ij}}$
3. Trova un indice $k \mid u_k = \min u_j \forall j \in J$
Poni $x_k := 1$, da J elimina k , da I elimina $\{i \mid a_{ik} = 1\}$
4. Se $I = \emptyset$ allora STOP (x è una copertura)
5. Se $J = \emptyset$ allora STOP (non esistono coperture)
- Altrimenti torna al passo 2.

7.2. Massima copertura

Dato un territorio suddiviso in 8 quartieri e 4 potenziali posizioni dove aprire un supermercato, l'obiettivo è posizionare due supermercati, massimizzando il numero di persone (per ogni quartiere i ci sono h_i persone) coperte.

$$z_i = \begin{cases} 1, & \text{il quartiere } i \text{ viene servito} \\ 0, & \text{il quartiere } i \text{ non viene servito} \end{cases}$$

Luoghi di apertura X_i :

Quartieri	1	2	3	4	A	2000	1000	800	3000	2500	1800	1700	480	h
1	1	1	0	0										
2	0	0	1	1										
3	1	0	1	0										
4	1	0	0	1										
5	0	1	0	1										
6	1	0	1	1										
7	0	1	0	1										
8	1	0	0	1										

$$\left\{ \begin{array}{l} \max h_1 z_1 + \dots + h_8 z_8 \\ x_1 + x_2 + x_3 + x_4 = 2 \quad (\text{apro 2 supermercati}) \\ x_1 + x_2 \geq z_1 \\ x_3 + x_4 \geq z_2 \\ \dots \\ x_1 + x_4 \geq z_8 \\ x \in \{0, 1\}^4 \\ z \in \{0, 1\}^8 \end{array} \right.$$

Modello.

Problema di copertura:

$$\text{Copertura di costo minimo: } \left\{ \begin{array}{l} \min c^T x \\ \vec{Ax} \geq \vec{1} \\ x \in \{0, 1\}^n \end{array} \right. \quad \text{Massima copertura: } \left\{ \begin{array}{l} \max h^T z \\ Ax \geq z_i \\ \sum x_i > p \\ x \in \{0, 1\}^n \\ z \in \{0, 1\}^m \end{array} \right.$$

Programmazione Lineare su Reti

La programmazione lineare su reti è la branca della programmazione lineare che si occupa di studiare algoritmi di risoluzione per problemi di ottimizzazione lineari che possono essere rappresentati mediante reti.

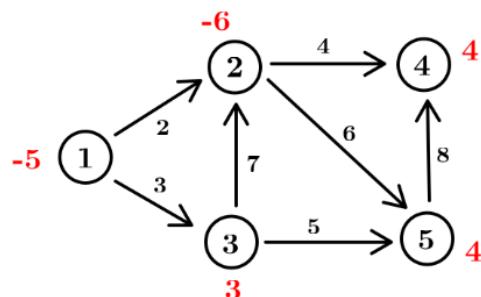
Una **grafo** è una coppia di insiemi $G = (N, A)$ dove $N \subset \mathbb{N}$ è l'insieme di nodi e $A \subset N \times N$ l'insieme di archi. Una **rete** è un grafo a cui vengono aggiunti attributi ai nodi o agli archi. Ad esempio, bilanci associati ai nodi, costi e capacità associati agli archi.

1. Flusso di costo minimo non capacitato

Si considerino n nodi e m archi. Sui nodi siano assegnati dei bilanci b_i con la convenzione che $b_i > 0$ significa che il nodo i è un pozzo (destinazione) e che $b_i < 0$, il nodo i è una sorgente (origine). Se $b_i = 0$ si dice che i è un nodo di transito.

Sia x_{ij} l'unità di flusso per l'arco (i, j) e c_{ij} il costo per unità di flusso dell'arco (i, j) . Si vuole determinare il flusso di costo minimo che rispetti i bilanci dei nodi e le capacità degli archi.

Prendiamo, come esempio, la seguente rete:



Funzione obiettivo: $\min 2x_{12} + 3x_{13} + 4x_{24} + 7x_{32} + 5x_{35} + 6x_{25} + 8x_{54}$

Equazioni di bilancio ai nodi:

$$\left\{ \begin{array}{l} \text{Nodo 1: } -x_{12} - x_{13} = -5 \\ \text{Nodo 2: } x_{12} + x_{32} - x_{24} - x_{25} = -6 \\ \text{Nodo 3: } x_{13} - x_{32} - x_{35} = 3 \\ \text{Nodo 4: } x_{24} + x_{45} = 4 \\ \text{Nodo 5: } x_{25} + x_{35} - x_{45} = 4 \end{array} \right.$$

Introduco la matrice E di dimensioni $|N| \times |A|$, detta **matrice di incidenza** del grafo. Alla riga $p \in N$ e alla colonna $(i, j) \in A$ si mette -1 se p sorgente, 1 se p destinazione, 0 se p è di transito (o se non incrocia l'arco).

$$E_{p,(i,j)} := \begin{cases} -1, & p = i \\ 1, & p = j \\ 0, & \text{altrimenti} \end{cases}$$

Nel nostro esempio: $E = \begin{bmatrix} 12 & 13 & 24 & 25 & 32 & 35 & 54 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & -1 & -1 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 & 0 & -1 & -1 & 0 \\ 4 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 5 & 0 & 0 & 0 & 1 & 0 & 1 & -1 \end{bmatrix}$

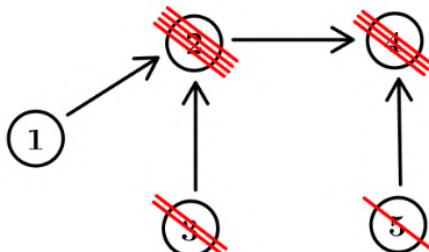
Osservo che $E \cdot \vec{x} = \vec{b}$ sono le equazioni di bilancio. Osservo anche che i termini delle equazioni di bilancio possono essere permutati per ottenere una differenza $\sum x_{pi} - \sum x_{iq}$ (vedi sotto).

Modello.

Flusso di costo minimo non capacitato:

Modello: (p)
$$\begin{cases} \min \sum_{(i,j) \in A} c_{ij} x_{ij} \\ \sum_{(p,i) \in A} x_{pi} - \sum_{(i,q) \in A} x_{iq} = b_i \\ x \geq 0 \end{cases} \equiv \begin{cases} \min c^T \cdot x \\ E \cdot x = b \\ x \geq 0 \end{cases}$$

Assumo che la rete sia connessa. Prendo un albero di copertura della rete: ci sono n nodi e E_T è sempre una matrice triangolare inferiore, quindi è invertibile. Siccome ho trovato un minore $(n-1) \times (n-1)$ di E , allora $rk(E) = (n-1)$. Questo significa che **posso cancellare un'equazione di bilancio perché è ridondante**.



$$E_T = \begin{bmatrix} 54 & 32 & 24 & 12 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & -1 & -1 \end{bmatrix}$$

TEOREMA.

Teorema (caratterizzazione delle basi):

Sia T un albero di copertura. La sottomatrice quadrata E_T , di dimensione $(n-1)$, ottenuta scegliendo le colonne relative agli archi di T è invertibile.

Vale anche il viceversa, ovvero tutte le matrici invertibili sono relative ad alberi di copertura.

La prima parte è stata dimostrata costruttivamente, la seconda si dimostra per assurdo.

Si può dimostrare anche che $\det E_T = \pm 1$, quindi, la soluzione di base è a componenti intere.

Allora, l'**ottimo del flusso di costo minimo non capacitato continuo coincide con l'ottimo del flusso di costo minimo non capacitato intero**, che può quindi essere risolto con linprog!

1.1. Costruire un flusso di base

D'ora in poi chiameremo le basi relative all'albero di copertura (T, L) , anziché (N, B) .

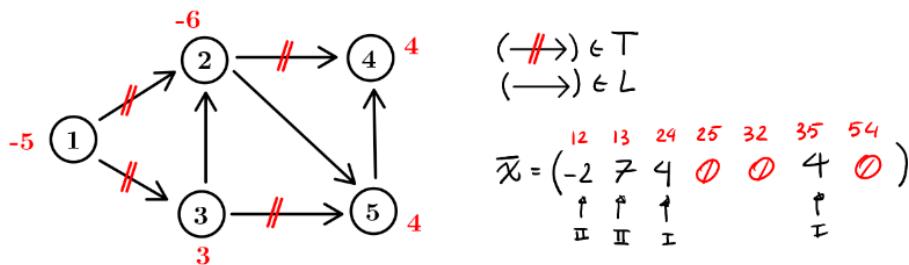
Voglio costruire $\bar{x} = (x_T, x_L) : E_T x_T = b$.

Algoritmo.

Costruzione di un flusso di base:

Per trovare un flusso di base si effettua una visita posticipata per foglie.

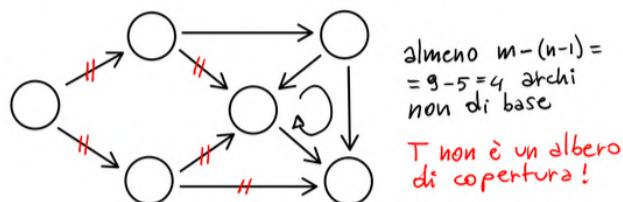
1. Pongo $x_{ij} = 0 \forall (i, j) \in L$
2. Parto dalla foglia j e attraverso l'unico arco insistente su essa per risalire al nodo precedente i . Pongo $x_{ij} = b$.
3. Elimino virtualmente l'arco (i, j) e la foglia j .
4. Rieseguo lo stesso passo di 2. tenendo conto dei beni che sono già usciti dalla foglia j (che prima era il nodo i)



Per verificare l'ammissibilità di una soluzione di base, basta controllare se $\bar{x}_i \geq 0 \forall i$. Ricordiamo che una soluzione di base ammissibile è un vertice del poliedro.

Domanda

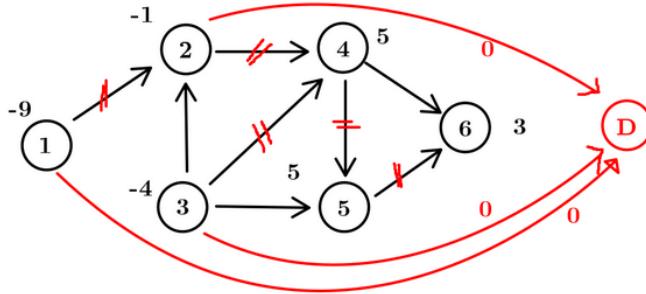
Come controllare se una soluzione \bar{x} è di base? Un flusso di base è generato da un albero di copertura, che ha esattamente $n - 1$ archi. Il resto degli archi, deve avere flusso zero, quindi è necessario che ci siano almeno $m - (n - 1)$ zeri nella soluzione di base. Se ce ne sono di più, la soluzione di base è degenere. Inoltre, gli indici posti a zero non devono formare un ciclo (altrimenti l'albero di copertura non è connesso).



1.2. Reti non bilanciate

Se $\sum_{i=1}^n b_i \neq 0$, si dice che la rete non è bilanciata.

- Se è sbilanciata in positivo, ovvero $\sum_{i=1}^n b_i > 0$, allora il poliedro è vuoto, $P = \emptyset$
- Se è sbilanciata in negativo, ovvero $\sum_{i=1}^n b_i < 0$, allora si ha lo stoccaggio: creo un nodo fittizio e ci faccio entrare le sorgenti a costo zero. Allora la rete è di nuovo bilanciata.



Per il teorema fondamentale della PL, $P = \emptyset$, oppure $P = -\infty$, oppure $\exists s. o.$

Siccome $c > 0$, allora $c^T x > 0$, quindi P non può mai andare a $-\infty$. In definitiva, l'ottimo esiste sempre se $P \neq \emptyset$, ovvero se la rete è ben progettata. Noi consideriamo solo le reti bilanciate perché non si perde generalità.

1.3. Calcolo dei potenziali e test di ottimalità

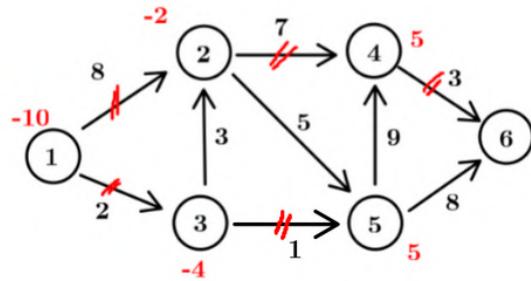
Faccio il primale associato a un problema di flusso di costo minimo non capacitato:

$$(d) \begin{cases} \min c^T x \\ Ex = b \\ x \geq 0 \end{cases} \Rightarrow (p) \begin{cases} \max b^T \pi \\ E^T \pi \leq c \end{cases} \quad \text{dove } E^T \text{ si chiama matrice dei potenziali.}$$

E è una matrice orizzontale, x è un vettore colonna, π è un vettore riga, quindi devo mettere la matrice trasposta E^T .

$$(d) \begin{cases} \min 8x_{12} + 2x_{13} + 7x_{24} + 5x_{25} + 3x_{32} + x_{35} + 3x_{46} + 9x_{54} + 8x_{56} \\ \begin{matrix} 12 & 13 & 24 & 25 & 32 & 35 & 46 & 54 & 56 \end{matrix} \\ \begin{matrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & -1 & -1 & 1 & 0 & 0 & 0 \\ 3 & 0 & 1 & 0 & 0 & -1 & -1 & 0 & 0 \\ 4 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 1 \\ 5 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \\ 6 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{matrix} \cdot \begin{bmatrix} x_{12} \\ x_{13} \\ x_{24} \\ x_{25} \\ x_{32} \\ x_{35} \\ x_{46} \\ x_{54} \\ x_{56} \end{bmatrix} = \begin{bmatrix} -10 \\ -2 \\ -4 \\ 5 \\ 5 \\ 6 \end{bmatrix} \\ x \geq 0 \end{cases}$$

$$(p) \left\{ \begin{array}{l} \max -9\pi_1 - 2\pi_2 - 5\pi_3 + 3\pi_4 + 5\pi_5 + 6\pi_6 \\ \text{T: } -\pi_1 + \pi_2 \leq 8 \\ \text{T: } -\pi_1 + \pi_3 \leq 2 \\ \text{T: } -\pi_2 + \pi_4 \leq 7 \\ \text{L: } -\pi_2 + \pi_5 \leq 5 \\ \text{L: } \pi_3 - \pi_4 \leq 3 \\ \text{T: } -\pi_3 + \pi_5 \leq 1 \\ \text{T: } -\pi_4 + \pi_6 \leq 3 \\ \text{L: } \pi_5 - \pi_4 \leq 9 \\ \text{L: } -\pi_5 + \pi_6 \leq 8 \end{array} \right.$$



12 13 24 25 32 35 46 54 56

$\bar{x} = (9, 1, 11, 0, 0, 5, 6, 0, 0)$ è una soluzione di base perché l'ho costruita con l'albero di copertura. Siccome $\bar{x}_i \geq 0$, allora è anche ammissibile. Dunque è un vertice del poliedro. Verifico l'ammissibilità anche nel primale e, se non è ammissibile, faccio il simplesso duale (scopriremo che il simplesso per il flusso di costo minimo non capacitato si semplifica drasticamente).

Osservazione 1: anche qui vale $rk(E^T) = (n-1)$, quindi posso porre a zero una variabile.

(Pappalardo consiglia di porre $\pi_1 = 0$ nei compiti perché sennò impazzisce a correggerli.)

Osservazione 2: $E^T \pi \leq c \iff -\pi_i + \pi_j \leq c_{ij} \forall (i, j) \in A$.

Chiamo $-\pi_i + \pi_j$ differenza di potenziale.

Definizione.

Definizione (potenziale di base):

Il potenziale associato alla soluzione di base è $-\pi_i + \pi_j = c_{ij} \forall (i, j) \in T$.

Devo trovare un vettore di potenziali tali che le differenze di potenziale agli archi sono uguali ai costi. Tale potenziale di base è ammissibile se sono soddisfatti i vincoli non di base

So che $\bar{\pi}_1 = 0 \implies \bar{\pi}_2 = 8, \bar{\pi}_3 = 2$. Procedo iterativamente considerando solo le diseguaglianze con indici di base per trovare anche gli altri potenziali. $\bar{\pi} = (0, 8, 2, 15, 3, 18)$. Le equazioni non di base sono la 4, 5, 7, 9. In questo caso il potenziale di base non è ammissibile perché viola la 9: $-x_5 + x_6 \leq 8$.

Sia $c_{ij}^\pi := c_{ij} + \pi_i - \pi_j$ il costo ridotto dell'arco (i, j) . Chiedere che $-\pi_i + \pi_j = 0$ equivale a chiedere che $c_{ij}^\pi = 0$.

TEOREMA.

Teorema (Teorema di Bellman):

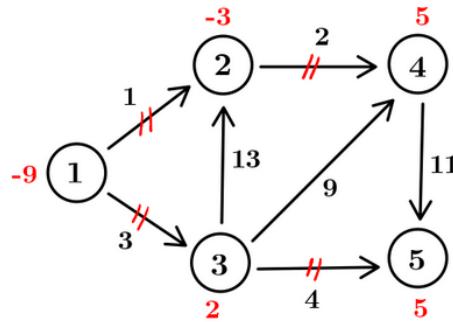
Sia dato T albero di copertura che genera un flusso di base ammissibile.

Se $c_{ij}^\pi \geq 0 \forall (i, j) \in L$ allora siamo all'ottimo.

$c_{ij}^\pi := c_{ij} + \pi_i - \pi_j$ si chiama **costo ridotto**, $c_{ij}^\pi \geq 0$ si chiama **condizione di Bellman**.

Le condizioni sui costi ridotti sono equivalenti all'ammissibilità della soluzione complementare $\bar{\pi}$

Esempio 1.1 Dato il seguente albero di copertura, costruisci la soluzione di base e stabilisci se è ammissibile, degenere, ottima. Poi costruisci il vettore dei potenziali e stabilisci se è ammissibile, degenere, ottimo.



12 13 24 32 34 35 45

$\bar{x} = (2, 7, 5, 0, 0, 5, 0)$ ammissibile perché $\bar{x}_i \geq 0$. Non è degenere perché non ci sono zeri oltre agli indici non di base. Non so ancora se è ottima.

$$E = \begin{bmatrix} 1 & 12 & 13 & 24 & 32 & 34 & 35 & 45 \\ 2 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 3 & 1 & 0 & -1 & 1 & 0 & 0 & 0 \\ 4 & 0 & 1 & 0 & -1 & -1 & -1 & 0 \\ 5 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \end{bmatrix} \Rightarrow E^T = \begin{bmatrix} 1 & 12 & -1 & 1 & 0 & 0 & 0 \\ 2 & 13 & -1 & 0 & 1 & 0 & 0 \\ 3 & 24 & 0 & -1 & 0 & 1 & 0 \\ 4 & 32 & 0 & 1 & -1 & 0 & 0 \\ 5 & 35 & 0 & 0 & -1 & 0 & 1 \\ 45 & 45 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

$$E^T \cdot \pi \leq c \Leftrightarrow \begin{cases} T - \pi_1 + \pi_2 \leq 1 \\ T - \pi_1 + \pi_3 \leq 3 \\ T - \pi_2 + \pi_4 \leq 2 \\ L - \pi_3 + \pi_2 \leq 13 \\ L - \pi_3 + \pi_4 \leq 9 \\ T - \pi_3 + \pi_5 \leq 4 \\ L - \pi_4 + \pi_5 \leq 11 \end{cases} \quad \begin{array}{l} \text{Pongo } \pi_1 := 0 \text{ e trovo:} \\ \bar{\pi} = (0, 1, 3, 3, 7) \\ \pi_1 \ \pi_2 \ \pi_3 \ \pi_4 \ \pi_5 \end{array}$$

Verifico le diseguaglianze degli indici non di base:

$-3 + 1 \leq 13, -3 + 3 \leq 9, -3 + 7 \leq 1 \Rightarrow \bar{\pi}$ è ammissibile, quindi sono arrivato all'ottimo!

Non è necessario costruire la matrice E^T per determinare le diseguaglianze; è sufficiente osservare il grafo. Consideriamo l'arco $(1, 2)$, che parte dal nodo 1 (negativo) e arriva al nodo 2 (positivo), con un costo $c_{12} = 1$. Quindi, la diseguaglianza associata è: $-\pi_1 + \pi_2 \leq 1$.

Inoltre, un modo semplice per costruire il vettore dei potenziali è leggere i costi cumulativi su ogni arco, quindi $\bar{\pi} = (0, 0+1, 0+3, 1+2, 3+4) = (0, 1, 3, 3, 7)$

12 13 24 35

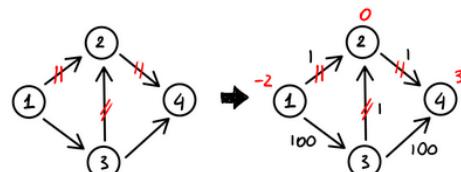
Domanda | dato un vettore dei potenziali, sapresti dire se è di base?
 Perché $\bar{\pi} := (0, 16, 3, 12, 7)$ sia di base, devono esserci almeno $n - 1$ costi ridotti nulli, in modo da poter costruire un albero di copertura.

$$c_{12}^\pi = c_{12} + \pi_1 - \pi_2 = 1 + 0 - 16 \neq 0$$

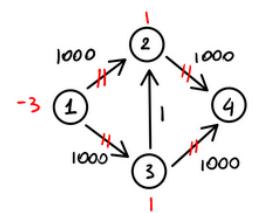
$$c_{13}^\pi = \dots, c_{24}^\pi = \dots, \dots, c_{45}^\pi = \dots$$

Domande da orale

➤ Dati una rete e un albero di copertura, metti i costi e i bilanci per ottenere il flusso ottimo.



➤ Disegna una rete dove il costo di ogni arco tranne uno è 1000. Il costo dell'arco rimanente è 1. Costruisci un albero di copertura che genera il flusso ottimo tale che l'arco a costo 1 non ne faccia parte.



1.4. Simplesso per flusso di costo minimo non capacitato

Il simplesso su reti si semplifica drasticamente rispetto a quello tradizionale perché non si invertono matrici, ma si fanno solo confronti.

Sia T un albero di copertura che genera un flusso di base (x_T, x_L) ammissibile, cioè $x_T \geq 0$.

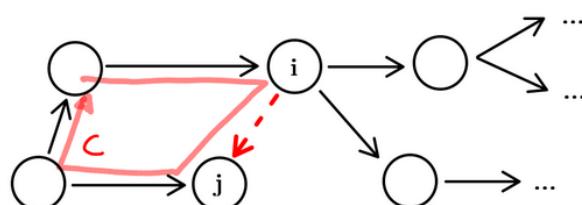
Se le condizioni di Bellman sono verificate i.e. $c_{ij}^\pi \geq 0 \forall (i, j) \in L$, allora il flusso è ottimo.

Altrimenti, significa che $\exists (i, j) \in L : c_{ij}^\pi < 0$. Si esegue quindi un passo del simplesso.

1.4.1. Arco entrante

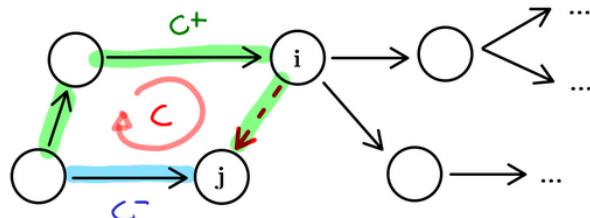
L'arco entrante è il primo (i, j) che viola la condizione di Bellman, ovvero tale che $c_{ij}^\pi < 0$.

L'arco entrante crea sicuramente un ciclo in T : lo chiamo \mathcal{C} .



1.4.2. Arco uscente

L'arco uscente deve per forza appartenere al ciclo perché lo voglio rompere (altrimenti non è più un albero di copertura). Divido $\mathcal{C} = \mathcal{C}^+ \cup \mathcal{C}^-$ dove \mathcal{C}^+ contiene gli archi concordi al verso di \mathcal{C} e \mathcal{C}^- contiene quelli discordi.



Sia $\theta \in \mathbb{N}$, propongo la seguente **regola di update** del flusso di base corrente \bar{x} :

$$x(\theta) = \begin{cases} \bar{x}_{ij} + \theta, & (i, j) \in \mathcal{C}^+ \\ \bar{x}_{ij} - \theta, & (i, j) \in \mathcal{C}^- \\ \bar{x}_{ij}, & (i, j) \notin \mathcal{C} \end{cases}$$

TEOREMA.

Teorema (correttezza del simplex per flussi):

Sia (i, j) l'arco che viola la condizione di Bellman, $c_{ij}^\pi := c_{ij} + \pi_i - \pi_j$ il costo ridotto.

Vale $c^T x(\theta) = c^T \bar{x} + \theta c_{ij}^\pi$

L'arco che viola Bellman è tale che $\theta c_{ij}^\pi < 0$. Quindi la funzione obiettivo decresce. $c^T x(\theta) \leq c^T \bar{x}$

Per quali θ è ammissibile $x(\theta)$?

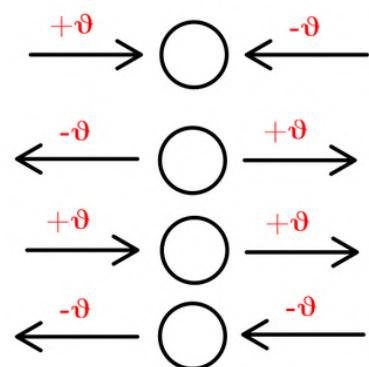
$x(\theta)$ è ammissibile se rispetta i bilanci, ovvero $E_T x_T + E_L \cancel{x_L} = b$ e se $x(\theta) \geq 0$.

$$= 0$$

1. Rispetta i bilanci?

1. Nodo non appartenente al ciclo: incidono archi che non appartengono al ciclo, non ho modificato il flusso, quindi i bilanci sono rispettati $\forall \theta$
2. Nodo appartenente al ciclo: ci sono quattro casi esaustivi.

- i. Entrambi gli archi sono **entranti**: da C^+ arriva $+\theta$, da C^- arriva $-\theta$, quindi il nodo rimane bilanciato
- ii. Entrambi gli archi sono **uscenti**: da C^+ parte $+\theta$, da C^- parte $-\theta$, quindi il nodo rimane bilanciato
- iii. Archi **concordi**: arriva $+\theta$, parte $+\theta$, quindi il nodo rimane bilanciato
- iv. Archi **discordi**: arriva $-\theta$, parte $-\theta$, quindi il nodo rimane bilanciato



I bilanci sono rispettati $\forall \theta$, quindi voglio θ più grande possibile per far decrescere la funzione obiettivo.

2. $x(\theta) \geq 0$?

1. Sui nodi non appartenenti al ciclo è rispettata, visto che non ho cambiato niente
2. Sugli archi di \mathcal{C}^+ certamente $\forall \theta$ poiché sommo una quantità positiva.
3. Sugli archi di \mathcal{C}^- devo sottrarre il massimo θ per far restare $x(\theta) \geq 0$:

$$\theta \leq \min_{\mathcal{C}^-} \{\bar{x}_{ij}\}. \text{ Allora mi conviene prendere } \theta = \min_{\mathcal{C}^-} \{\bar{x}_{ij}\}.$$

Miracolo: se prendo $\theta = \min_{\mathcal{C}^-} \{\bar{x}_{ij}\}$, uno degli archi di \mathcal{C}^- si azzera.

Per la regola di anticiclo di Bland, prendo il primo di questi archi: questo è l'arco uscente. Poi ricalcolo il nuovo potenziale, vedo se non è ammissibile e, in caso, eseguo un altro passo del simplex.

Domanda L'arco entrante può essere l'arco uscente? L'arco entrante è sempre in \mathcal{C}^+ , quindi non può essere anche l'arco uscente e il simplex non cicla mai. In realtà, non è nemmeno necessario calcolare \mathcal{C}^+ .

Domanda Quando il problema fa $-\infty$? Se tutti gli archi appartengono a \mathcal{C}^+ , allora $\mathcal{C}^- = \emptyset$, allora posso scegliere $\theta = +\infty$ e la funzione obiettivo va a $-\infty$. Questo è un segnale di qualche incongruenza nel modello.

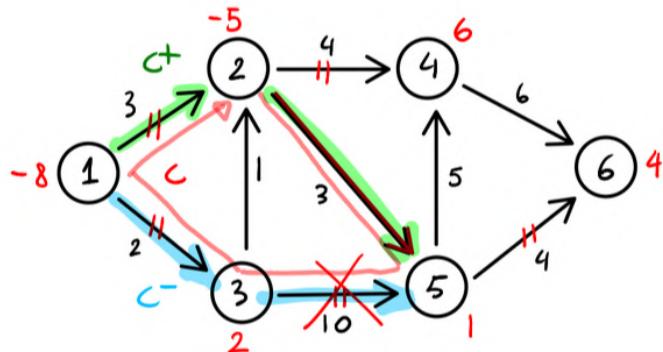
ALGORITMO.

Algoritmo del simplex per flussi non capacitati

1. Trovo una partizione degli archi (T, L) , con T un albero di copertura, che generi un flusso di base ammissibile
2. Calcolo il flusso di base $\bar{x} = (\bar{x}_T, \bar{x}_L)$, $\bar{x}_T = E_T^{-1}b$, $\bar{x}_L := 0$
Calcolo il potenziale di base $\bar{\pi}^T := c_T^T E_T^{-1}$. Indico con $c_{ij}^{\bar{\pi}} := c_{ij} + \bar{\pi}_i - \bar{\pi}_j$ il costo ridotto $\forall (i, j)$
3. if le condizioni di Bellman sono verificate i.e. $c_{ij}^{\bar{\pi}} \geq 0 \forall (i, j) \in L$ then STOP
(\bar{x} è un flusso ottimo e $\bar{\pi}$ è un potenziale ottimo).
else calcolo l'arco entrante $(p, q) := \min \{(i, j) \in L : c_{ij}^{\bar{\pi}} < 0\}$
dove il minimo è fatto rispetto all'ordine lessicografico sugli archi.
4. L'arco (p, q) forma un ciclo \mathcal{C} con gli archi di T . Indica con \mathcal{C}^+ gli archi del ciclo concordi con il verso di \mathcal{C} , e con \mathcal{C}^- quelli discordi.
Calcolo $\theta := \min \{\bar{x}_{ij} : (i, j) \in \mathcal{C}^-\}$.
if $\theta = +\infty$ then STOP (il flusso di costo minimo ha valore $-\infty$)
else trova l'arco uscente $(r, s) := \min \{(i, j) \in \mathcal{C}^- : \bar{x}_{ij} = \theta\}$
5. Aggiorno la partizione e torno al passo 2.

Esempio 1.2 esegui un passo del simplex sulla seguente rete relativa a un problema di flusso di costo minimo non capacitato partendo da $\bar{x} = (1, 7, 6, 0, 0, 5, 0, 0, 4)$

$$\begin{array}{ccccccccc} 12 & 13 & 24 & 25 & 32 & 35 & 46 & 54 & 56 \\ T & T & T & L & L & T & L & L & T \end{array}$$



Costruisco il potenziale di base $\bar{\pi} = (0, 3, 2, 7, 12, 16)$.

Ora devo trovare il primo $c_{ij}^\pi < 0$. Osservo che ne devo calcolare al più quattro.

$$c_{25}^\pi = 3 + 3 - 12 < 0 \Rightarrow (2, 5) \text{ è l'arco entrante.}$$

$$\mathcal{C}^- = \{(1, 3), (3, 5)\}, \quad x_{13} = 7, \quad x_{35} = 5,$$

$$\theta = \min\{x_{ij} : (i, j) \in \mathcal{C}^-\} = \min\{x_{13}, x_{35}\} = \min\{3, 5\} = 5 \quad \therefore \text{l'arco uscente è } (3, 5)$$

La funzione obiettivo all'inizio valeva $c^T \bar{x} = 107$.

$$\begin{aligned} \text{Ora la funzione obiettivo vale } c^T \bar{x}(\theta) &= c^T \bar{x} + \theta c_{25} = 107 + 5 \cdot (c_{25} + \pi_2 - \pi_5) = \\ &= 107 + 5(3 + 3 - 12) = 107 - 30 = 77 \end{aligned}$$

Il nuovo flusso di base $\bar{x}_{\text{new}} = (6, 2, 6, 5, 0, 0, 0, 0, 4)$

$$\begin{array}{cccccccccc} \uparrow & \uparrow \\ +5 & -5 & / & +5 & / & -5 & / & / & / \\ T & T & T & T & L & L & L & L & T \end{array}$$

1.5. Flusso di costo minimo come classe di problemi

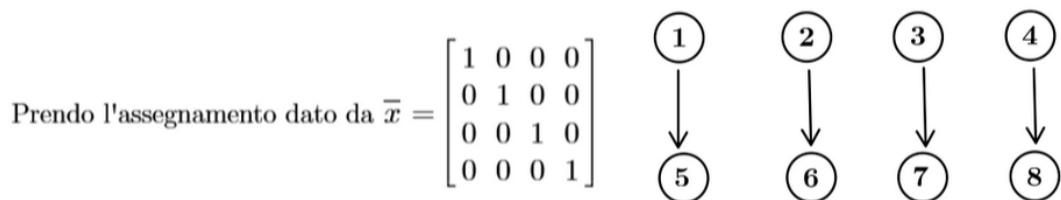
Abbiamo già visto esempi di problemi che possiedono la proprietà per cui i vertici del poliedro hanno componenti intere (quindi PL coincide con PLI). Tutti questi problemi rientrano nella classe dei problemi di flusso di costo minimo.

1.5.1. L'assegnamento è un flusso di costo minimo non capacitato

lavori				
	5	6	7	8
1	8	11	13	15
2	6	9	12	15
3	7	5	18	6
4	4	9	6	12
persone				

Sia data la matrice relativa ai costi di un problema di assegnamento:

Se non mi fossi dimenticato tutta la PL, mi dovrebbero ricordare che un vincolo del problema di assegnamento è $x_{15} + x_{16} + x_{17} + x_{18} = 1$ (equazioni sulle persone). Questa è proprio l'equazione di bilancio al nodo 1. Equivale infatti a $-x_{15} - x_{16} - x_{17} - x_{18} = -1$. $x_{15} + x_{25} + x_{35} + x_{45} = 1$ (equazioni sui lavori) è l'equazione di bilancio al nodo 5.



Mi accorgo che \bar{x} non solo è ammissibile, ma è anche di base (degenero).

Infatti $E = \begin{bmatrix} 8 & 11 & 13 & 15 \\ 6 & 9 & 12 & 15 \\ 7 & 5 & 18 & 6 \\ 4 & 9 & 6 & 12 \end{bmatrix}$ è l'albero di copertura.

Modifico il grafo dell'assegnamento aggiungendo i bilanci e altri archi gratis che mostrano bene l'albero di copertura.

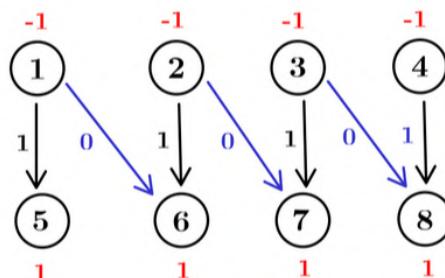
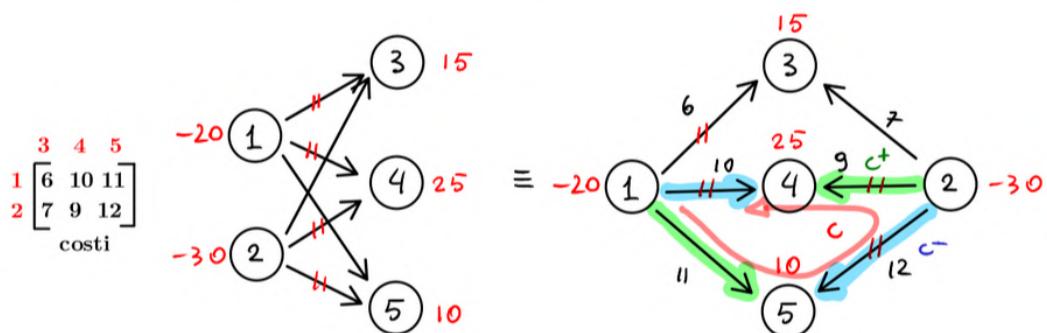


Figura 1: ci sono $(n^2 - n)$ archi a scelta blu, $(n - 1)$ quelli che devo mettere.

Tutte le combinazioni possibili per mettere gli archi blu sono $\binom{n^2 - n}{n - 1}$

1.5.2. Il trasporto è un flusso di costo minimo non capacitato

Sia data la matrice dei costi di un problema di trasporto, e siano date le capacità di spedizionieri e acquirenti come bilanci. Posso costruire un problema di flusso di costo minimo non capacitato.



1.6. Il flusso multiobiettivo è un problema difficile

Sia data una rete dove ogni arco ha un costo c_{ij} e una lunghezza l_{ij} . Non minimizzare costo e lunghezza contemporaneamente, ma posso trasformare un obiettivo in un vincolo.

Ad esempio, voglio che la lunghezza sia minima e mi accontento di un vincolo di budget:

$$\begin{cases} \min l^T x \\ Ex = b \\ c^T x \leq C \\ x \geq 0 \end{cases} \quad x \geq \{0, 1\}$$

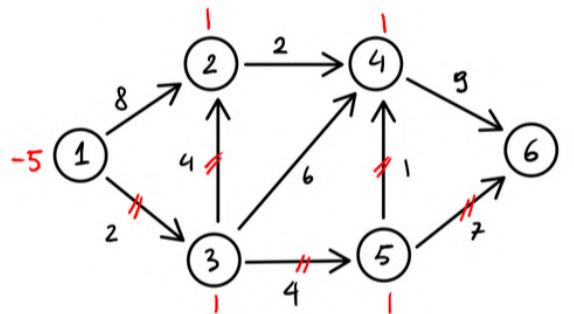
Serve anche il vincolo di interezza $x_{ij} = \{0, 1\}$. Ora non vale più il teorema dell'interesse dei vertici del poliedro, quindi è un problema di PLI.

2. Cammini minimi

Il TSP è il problema NP-hard più noto legato alle reti. Al contrario, il problema dei **cammini minimi** è il più conosciuto tra i problemi polinomiali semplici.

SPT (Shortest Path Tree): Qual'è il cammino minimo orientato dal nodo r al nodo q ? Questo è un problema di flusso minimo non capacitato, dove $b_r = -1$, $b_q = 1$, $b_i = 0 \forall i \neq r, i \neq q$. Senza perdere generalità, si può supporre che esista un cammino orientato da p a q , a meno di aggiungere un arco fittizio (r, q) di costo $+\infty$.

Addirittura, posso trovare tutti i cammini minimi che partono da un nodo, ponendo $b_r = -(n-1)$, $b_i = 1 \forall i \neq r$.



Modello.

Cammini minimi:

$$\begin{cases} \min c^T x \\ Ex = b : b_i = \begin{cases} -(n-1), & i = r \\ 1, & i \neq r \end{cases} \quad \text{è il problema di SPT di radice } r \\ x \geq 0 \end{cases}$$

Come mai si chiama Shortest Path Tree? Se io modellizzo il problema di cammini minimi come un problema di flusso di costo minimo su reti so per certo che la soluzione ottima è sempre legata a un albero di copertura.

Domanda Può esistere una soluzione ottima che non è un albero di copertura? Sì, perché il teorema fondamentale della PL non dice che tutte le soluzioni ottime sono vertici, ma che esiste almeno una soluzione ottima che è un vertice, se esiste (potrebbe non esserci nessun cammino orientato che connette un nodo).

In riferimento all'esempio precedente, trovo una soluzione di base operando una visita per foglie relativa a un albero di copertura T . $\bar{x} = (0, 5, 0, 1, 0, 3, 0, 1, 1)$

12 13 24 32 34 35 46 54 56

Chiedersi se \bar{x} è ottimo equivale a chiedersi se T è l'albero dei cammini minimi.

$$(i, j) \in T : \begin{cases} -\pi_1 + \pi_3 \leq 2, \pi_1 := 0 \Rightarrow \pi_3 = 2 \\ -\pi_3 + \pi_2 \leq 4 \Rightarrow \pi_2 = 6 \\ -\pi_3 + \pi_5 \leq 4 \Rightarrow \pi_5 = 6 \\ -\pi_5 + \pi_4 \leq 1 \Rightarrow \pi_4 = 7 \\ -\pi_5 + \pi_6 \leq 7 \Rightarrow \pi_6 = 13 \end{cases}$$

$$(i, j) \notin T : \{-\pi_1 + \pi_2 \leq 8, -\pi_2 + \pi_4 \leq 2, -\pi_3 + \pi_4 \leq 6, -\pi_4 + \pi_6 \leq 9\}$$

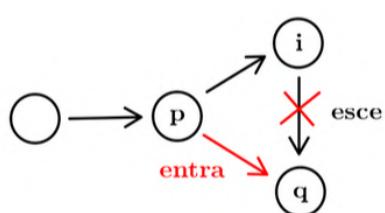
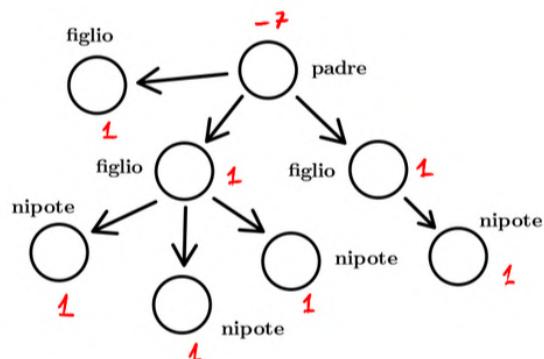
\uparrow \uparrow \uparrow \uparrow
 $6 \leq 8$ $1 \leq 2$ $5 \leq 6$ $6 \leq 9$

Dunque \bar{x} è ottimo e T è l'albero dei cammini minimi di radice (1)

2.1. Simplesso per cammini minimi

Il problema dei cammini minimi non ha mai soluzioni di base degeneri. Questo perché gli alberi di copertura T sono sempre orientati dai padri verso i figli (perché non possono esserci due archi entranti in un nodo): dunque non possono esserci unità di flusso uguali a zero.

Di conseguenza, non serve più la regola di anticiclo di Bland. Conviene calcolare tutti i costi ridotti e prendere come arco entrante quello che fa decrescere maggiormente la funzione obiettivo.



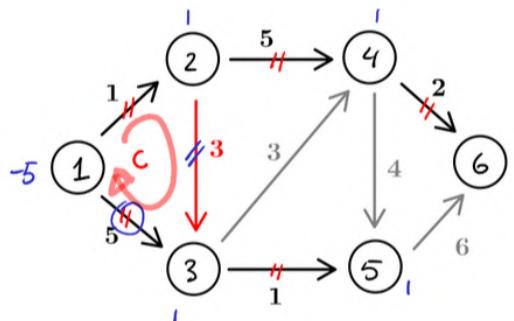
Se l'arco entrante è (p, q) , come arco uscente devo prendere per forza l'unico arco che entra in q (lo cerco per scorrimento), in quanto non possono esserci due archi entranti in un nodo.

ALGORITMO.

Algoritmo del simplex per cammini minimi

1. Trovo un albero T orientato di radice r
2. Calcolo il potenziale di base $\bar{\pi}^T := c_T^T E_T^{-1}$. Indico con $c_{ij}^{\bar{\pi}} := c_{ij} + \bar{\pi}_i - \bar{\pi}_j$ il costo ridotto $\forall (i, j)$
3. if le condizioni di Bellman sono verificate i.e. $c_{ij}^{\bar{\pi}} \geq 0 \forall (i, j) \notin T$ then STOP
(T è un albero dei cammini minimi).
else scelgo come arco entrante un arco (p, q) tale che $c_{pq}^{\bar{\pi}} < 0$ (possibilmente il più piccolo costo ridotto).
4. L'arco (p, q) forma un ciclo \mathcal{C} con gli archi di T .
if tutti gli archi di \mathcal{C} sono concordi con (p, q) then STOP (non esiste albero dei cammini minimi e \mathcal{C} è un ciclo di costo negativo).
else scelgo l'arco uscente come l'unico arco $(i, q) \in T$ che entra nel nodo q .
5. Aggiorno l'albero e torno al passo 2.

Esempio 2.1 Effettuare un passo del simplex sul seguente problema di cammini minimi, dato un albero di copertura.



$\bar{\pi} = (0, 1, 5, 6, 6, 8)$. Si vede a occhio che questo non è l'albero dei cammini minimi perché, per andare al nodo (3), è meglio passare prima da (2).

L'arco che viola Bellman è $(2, 3)$, infatti $c_{23}^{\bar{\pi}} = 3 + 1 - 5 = -1$.

L'arco uscente è per forza $(1, 3)$.

La funzione obiettivo diminuisce di $5 - 3 - 1 = 1$ per ogni cammino che includeva l'arco $(1, 3)$. Quindi è il flusso sull'arco $(1, 3)$, \bar{x}_{13} . In questo caso, 2.

2.2. Algoritmo di Dijkstra

Esiste un algoritmo ancora più semplice del simplex per risolvere il problema di cammini minimi. L'unica ipotesi aggiuntiva è che $c_{ij} \geq 0$.

Costruisco un vettore π di etichette nel seguente modo:

$$\pi_i := \begin{cases} 0, & i = r \\ +\infty, & i \neq r \end{cases}$$

dove π_i indica il costo per andare dal nodo r al nodo i .



Breve video intuitivo
sull'algoritmo di Dijkstra

Costruisco un vettore p di predecessori nel seguente modo:

$$p_i := \begin{cases} 0, & i = r \\ -1, & i \neq r \end{cases}$$

dove 0 significa per convenzione che non ha predecessori, -1 che non sono ancora stati calcolati.

Alla fine, in π ci saranno i costi minimi per raggiungere tutti i nodi. In p ci saranno i predecessori di ogni nodo dell'albero dei cammini minimi, che potrà essere dunque ricostruito.

ALGORITMO.

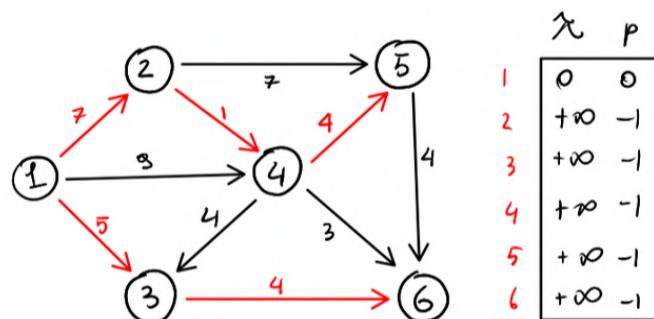
Algoritmo di Dijkstra

- Poni $\pi_i := \begin{cases} 0, & i = r \\ +\infty, & i \neq r \end{cases}$, $p_i := \begin{cases} 0, & i = r \\ -1, & i \neq r \end{cases}$,
 $Q := \{1, \dots n\}$ l'insieme dei nodi non ancora visitati.

- while** $Q \neq \emptyset$ **do**
 - estrai da Q il nodo i con etichetta π_i minima
 - per ogni nodo $j \in FS(i)$ ¹:
 - if $\pi_j > \pi_i + c_{ij}$ then aggiorna $p_j = i$, $\pi_j = \pi_i + c_{ij}$

1. $FS(i)$ è la stella uscente da i , ovvero l'insieme dei nodi raggiungibili da i con un arco diretto

Esempio 2.2 Trova l'albero dei cammini minimi della seguente rete con l'algoritmo di Dijkstra:



$$Q = \{1, 2, 3, 4, 5, 6\} \Rightarrow i = 1$$

$$FS(1) = \{2, 3, 4\}$$

- $\pi_2 > \pi_1 + c_{12} \Leftrightarrow +\infty > 0 + 7$. Sì, quindi $\pi_2 = 7$, $p_2 = 1$
- $\pi_3 > \pi_1 + c_{13} \Leftrightarrow +\infty > 0 + 5$. Sì, quindi $\pi_3 = 5$, $p_3 = 1$
- $\pi_4 > \pi_1 + c_{14} \Leftrightarrow +\infty > 0 + 9$. Sì, quindi $\pi_4 = 9$, $p_4 = 1$

$$Q = \{2, 3, 4, 5, 6\} \Rightarrow i = 3$$

$$FS(3) = \{6\}$$

- $\pi_6 > \pi_3 + c_{36} \Leftrightarrow +\infty > 5 + 4$. Sì, quindi $\pi_6 = 9$, $p_6 = 3$

$$Q = \{2, 4, 5, 6\} \Rightarrow i = 2$$

$$FS(2) = \{4, 5\}$$

- $\pi_4 > \pi_2 + c_{24} \Leftrightarrow 9 > 7 + 1$. Sì, quindi $\pi_4 = 8$, $p_4 = 2$
- $\pi_5 > \pi_2 + c_{25} \Leftrightarrow +\infty > 5 + 7$. Sì, quindi $\pi_5 = 14$, $p_5 = 2$

$$Q = \{4, 5, 6\} \Rightarrow i = 4$$

$$FS(4) = \{5, 6\}$$

- $\pi_5 > \pi_4 + c_{45} \Leftrightarrow 14 > 8 + 4$. Sì, quindi $\pi_5 = 12$, $p_5 = 4$
- $\pi_6 > \pi_4 + c_{46} \Leftrightarrow 9 > 8 + 3$. No, quindi lascio invariato

$$Q = \{5, \emptyset\} \Rightarrow i = 6$$

$F(S) = \emptyset$ quindi vado al passo successivo

$$Q = \{\emptyset\} \Rightarrow i = 5$$

$$FS(5) = \{6\}$$

$\pi_6 > \pi_5 + c_{56} \Leftrightarrow 9 > 12 + 4$. No, quindi lascio invariato

Fine dell'algoritmo.

Ricostruisco l'albero dei cammini minimi partendo dal nodo 6.

	π	p
1	0	0
2	+ 9 7	-1 1
3	+ 10 5	-1 1
4	+ 10 8	-1 12 2
5	+ 9 11 12	-1 4 4
6	+ 9 5	-1 3

Osservazione: $\pi_j > \pi_i + c_{ij} \Leftrightarrow c_{ij}^\pi < 0$, viene cambiato ponendo $c_{ij}^\pi = 0$

Anche questo algoritmo verifica le condizioni di Bellman, ma, a differenza del simplex, non mi obbliga a cercare nel grafo l'unico arco entrante, e soprattutto non mi obbliga a calcolare i potenziali poiché sto aggiornando solo quelli della stella uscente.

3. Flusso di costo minimo capacitato

Le reti capacitate includono un vettore $u \in \mathbb{R}^m$ detto **capacità superiore** degli archi. $x_{ij} \leq u_{ij}$

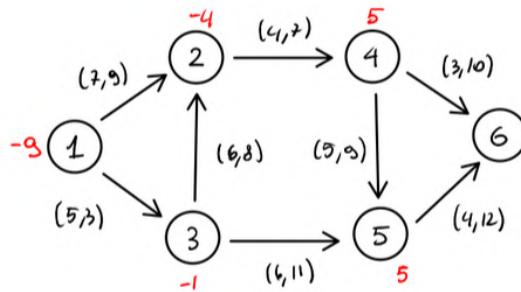
Modello.

Flusso di costo minimo capacitato:

$$\begin{cases} \min c^T x \\ Ex = b \\ 0 \leq x \leq u \end{cases} \equiv \begin{cases} \min c^T x \\ Ex = b \\ x + w = u \\ x, w \geq 0 \end{cases} \text{ in formato duale standard}$$

Dove $u, w \in \mathbb{R}^m$

$x_{ij} + w_{ij} = u_{ij}$, dove w_{ij} rappresenta la portata residua. Se $w_{ij} = 0$ significa che l'arco è saturo.



La matrice dei flussi è data da $\begin{cases} E_x = b \\ Ix + Iw = u \end{cases} \equiv \begin{matrix} E & 0 \\ I & I \end{matrix} \cdot \begin{matrix} x \\ w \end{matrix} = \begin{matrix} b \\ u \end{matrix}$.

La matrice dei flussi è una matrice triangolare a blocchi, ha rango $(m+n-1)$, quindi posso cancellare una riga e la matrice diventa di dimensioni $(m+n-1) \times (2m)$

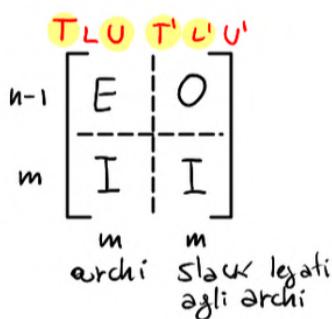
Consideriamo la tecnica della **tripartizione degli archi**:

Sia T, L, U una tripartizione degli archi con la condizione che in T ci sia un albero di copertura.

La tripartizione degli archi conduce ad una tripartizione T', L', U' delle variabili w , poiché sono una per ogni arco. Quindi $(x, w) = (x_T, x_L, x_U, w_{T'}, w_{L'}, w_{U'})$

Teorema (caratterizzazione delle basi II):

Una base del problema del flusso di costo minimo su reti capacitate è una tripartizione ed ogni tripartizione è una base scegliendo gli indici T, U, T', L' .



Quanti archi ci sono in T, U, T', L' ?

$$|T| = n - 1$$

$$|U| = |U'| \text{ perché } x_{ij} \in U \Leftrightarrow w_{ij} \in U'$$

$$\text{Quindi } |U \cup T' \cup L'| = |U' \cup T' \cup L'| = m$$

$$\text{Quindi } |T \cup U \cup T' \cup L'| = |T| + m = (m + n - 1)$$

Ho quindi selezionato una matrice quadrata di dimensione $(m + n - 1)$. Il teorema ci dice che tale matrice è invertibile (perché è una base) e che il suo rango è massimo.

3.1. Costruire un flusso di base

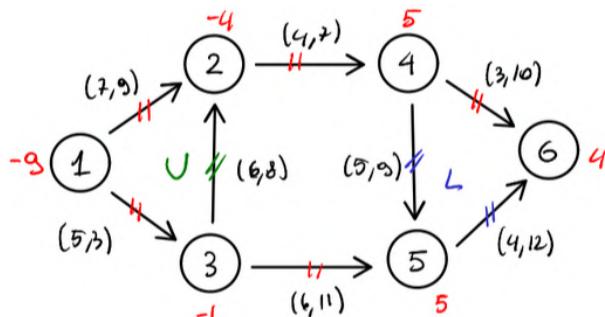
Questo teorema ci permette di calcolare il flusso di base:

$$(x, w) = (?, 0, u_U, ?, u_L, 0) \text{ dove } \begin{array}{l} T, U, T', L' \text{ base} \\ L, U' \text{ non base} \end{array}$$

- x_L e $w_{U'}$ sono nulli poiché non in base.
- Se $x_L = 0$, $w_{L'} = u_L - x_L \Rightarrow w_{L'} = u_L$
- $w_{U'} = 0$, $x_U + w_{U'} = u_U \Rightarrow x_U = u_U$

Quindi per calcolare un flusso di base mi basta calcolare gli archi dell'albero di copertura con la differenza che le altre variabili non sono poste tutte a zero. Effettuo sempre una visita posticipata per foglie.

Esempio 3.1 Costruisci il flusso di base data la tripartizione (T, L, U) . Stabilisci se è ammissibile e/o degenere.



$(x, w) = (x_T, x_L, x_U, w_{T'}, w_{L'}, w_{U'})$ implica la risoluzione di un sistema di $(m + n - 1) = 13$ equazioni in $2m = 16$ incognite. Siccome la matrice del poliedro dei flussi ha rango massimo, il sistema diventa 13×13 . In realtà, risolvere questo sistema è molto semplice.

$$(x, w) = \left(\begin{array}{ccccccccccccccccc} T & T & T & U & T & L & T & L & T & T & T & U & T & L & T & L \\ 12 & 13 & 24 & 32 & 35 & 45 & 46 & 56 & 12 & 13 & 24 & 32 & 35 & 45 & 46 & 56 \end{array} \mid \begin{array}{ccccccccc} 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 & 12 \end{array} \right) =$$

$$\begin{array}{ccccccccccccccccc} 12 & 13 & 24 & 32 & 35 & 45 & 46 & 56 & 12 & 13 & 24 & 32 & 35 & 45 & 46 & 56 \\ -3 & 12 & 9 & 8 & 5 & 0 & 4 & 0 & 12 & -9 & -2 & 0 & 6 & 9 & 6 & 12 \end{array} \quad (x, w) \text{ non è ammissibile}$$

$$\quad = \begin{array}{ccccccccccccccccc} -3 & 12 & 9 & 8 & 5 & 0 & 4 & 0 & 12 & -9 & -2 & 0 & 6 & 9 & 6 & 12 \end{array} \quad \text{e non è degenere.}$$

3.1.1. Costruire il flusso di base dalla tripartizione e viceversa

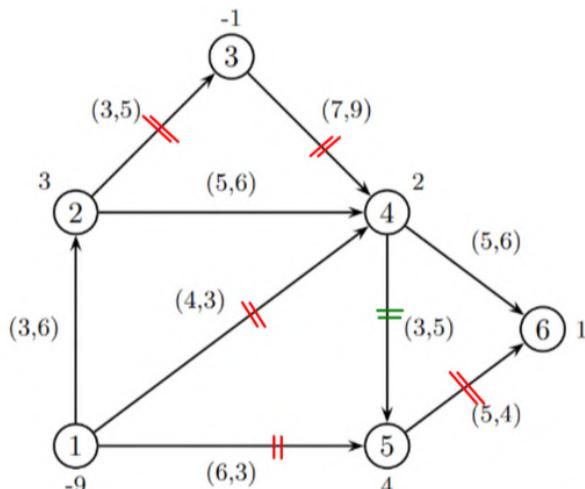
Algoritmo.

Costruzione del flusso di base data una tripartizione T, L, U :

Questo è un metodo (inventato da me) per calcolare un flusso di base:

1. Imposto $x_L := 0$, $x_U := u_U$
 2. Disegno solo gli archi dell'albero di copertura T
 3. Per ogni arco $(i, j) \in U$, aggiorno virtualmente $b_i = b_i + u_{ij}$, $b_j = b_j - u_{ij}$
 4. Partendo dalle foglie, calcolo i flussi secondo la seguente regola:
 - se la foglia i entra in j , allora $x_{ij} = -b_i$, aggiorno virtualmente $b_j = b_j - x_{ij}$
 - se la foglia i esce da j , allora $x_{ji} = b_i$, aggiorno virtualmente $b_j = b_j + x_{ji}$
- Procedo finché non ho visitato tutti i nodi della rete.

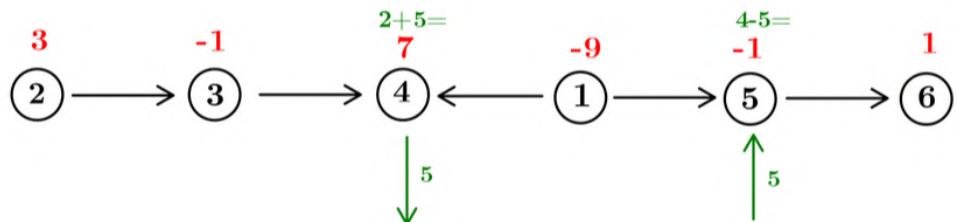
Esempio 3.2 Calcola il flusso di base della seguente rete, data la tripartizione T, L, U :



12 14 15 23 24 34 45 46 56

1. imposto i bilanci degli archi di L e di U : $\bar{x} = (0 \quad \quad \quad 0 \quad \quad \quad 0 \quad \quad \quad 5 \quad \quad \quad 0 \quad \quad \quad)$

2. Disegno solo l'albero di copertura T , 3. Aggiorno i bilanci dei nodi interessati da U



4. Calcolo i flussi:

- Il nodo 6 esce dal nodo 5, quindi $x_{56} = b_6 = 1$, aggiorno $b_5 = -1 + 1 = 0$
- Il nodo 5 esce dal nodo 1, quindi $x_{15} = b_5 = 0$, aggiorno $b_1 = -9 + 0 = -9$
- Il nodo 1 entra nel nodo 4, quindi $x_{14} = -b_1 = 9$, aggiorno $b_4 = 7 - 9 = -2$

- Il nodo 4 esce dal nodo 3, quindi $x_{34} = b_4 = -2$, aggiorno $b_3 = -1 - 2 = -3$
- Il nodo 3 esce dal nodo 2, quindi $x_{23} = b_3 = -3$. Ho visitato tutti i nodi.

$$\bar{x} = \begin{pmatrix} 12 & 14 & 15 & 23 & 24 & 34 & 45 & 46 & 56 \\ 0 & 9 & 0 & -3 & 0 & -2 & 5 & 0 & 1 \end{pmatrix}$$

Algoritmo.

Trovare la tripartizione T, L, U dato un flusso di base:

Per ogni componente di \bar{x}

- se $x_{ij} \neq 0, u_{ij}$, sicuramente l'arco (i, j) è in T
(N.B. non è detto che il flusso sia ammissibile)
- se $x_{ij} = 0$, l'arco (i, j) può stare in L o in T (per cui il flusso sarebbe degenero)
- se $x_{ij} = u_{ij}$, l'arco (i, j) può stare in U o in T (per cui il flusso sarebbe degenero)

Assumendo che la rete sia connessa, scelgo se allocare gli archi di (ii.) in L o T e gli archi di (iii.) in U o T in modo da avere alla fine esattamente $(n - 1)$ archi in T . Questo perchè un albero di copertura di una rete connessa ha esattamente $(n - 1)$ archi.

Esempio 3.3 Trovare la tripartizione T, L, U della rete precedente, dato il flusso di base:

$$\bar{x} = \begin{pmatrix} 12 & 14 & 15 & 23 & 24 & 34 & 45 & 46 & 56 \\ 0 & 9 & 0 & -3 & 0 & -2 & 5 & 0 & 1 \end{pmatrix} \text{ (che fantasia)}$$

$$\bar{x} = \begin{pmatrix} T/L, & T, & T/L, & T, & T/L, & T, & T/U, & T/L, & T \\ 12 & 14 & 15 & 23 & 24 & 34 & 45 & 46 & 56 \\ 0 & 9 & 0 & -3 & 0 & -2 & 5 & 0 & 1 \end{pmatrix}$$

Anche se $x_{14} = 9 > u_{14}$ (quindi \bar{x} non è ammissibile), posso comunque trovare la tripartizione.

Voglio esattamente $n - 1 = 5$ archi in T , quindi, per esempio, metto in T gli archi $(1, 5)$. Metto gli altri archi in L e U . In questo modo, ho ottenuto la tripartizione dell'esempio precedente.

Non è, però, l'unica tripartizione che genera questo flusso, che è infatti degenero, ovvero che è generato da più basi (tripartizioni).

3.1.2. Flusso di base ammissibile e degenero

Un flusso di base (x, w) è ammissibile se

- $x_u \geq 0$, questa condizione è sempre rispettata
- $x_T \geq 0$, come nel flusso non capacitato
- $w_T \geq 0$ significa che $u_T - x_T \geq 0 \Rightarrow x_T \leq u_T$

In definitiva, (x, w) è ammissibile se $0 \leq x_T \leq u_T$

Un flusso di base (x, w) è **degenero** se una tra x_T, x_U, w_T, w_L è zero:

- $x_U, w_L \neq 0$ sempre
- $x_T = 0$, come nel flusso non capacitato
- $w_T = 0$ significa che $u_T - x_T = 0 \Rightarrow x_T = u_T$

In definitiva, (x, w) è degenero se $x_T = 0 \vee x_T = u_T$ i.e. se un arco è vuoto o saturo.

3.2. Calcolo dei potenziali e test di ottimalità

Per trovare una tripartizione che genera un flusso di base ammissibile occorre il problema dei potenziali.

$$\begin{array}{l} \left\{ \begin{array}{l} \min c^T x + 0w \\ \left[\begin{array}{cc} E & 0 \\ I & I \end{array} \right] \cdot \left[\begin{array}{c} x \\ w \end{array} \right] = \left[\begin{array}{c} b \\ u \end{array} \right] \\ x, w \geq 0 \end{array} \right. \quad \rightarrow \quad \left\{ \begin{array}{l} \max \pi b + \mu u \\ \left[\begin{array}{cc} E & 0 \\ I & I \end{array} \right]^T \left[\begin{array}{c} \pi \\ \mu \end{array} \right] \leq \left[\begin{array}{c} c \\ 0 \end{array} \right] \end{array} \right. \equiv \left\{ \begin{array}{l} \max \pi b + \mu u \\ \left[\begin{array}{cc} E^T & I \\ 0 & I \end{array} \right] \cdot \left[\begin{array}{c} \pi \\ \mu \end{array} \right] \leq \left[\begin{array}{c} c \\ 0 \end{array} \right] \end{array} \right. \\ \parallel \\ \left\{ \begin{array}{l} \min c^T x + 0w \\ Ex = b \\ 0 \leq x \leq u \end{array} \right. \quad \rightarrow \quad \left\{ \begin{array}{l} \max \pi b + \mu u \\ E^T \pi + \mu \leq c \quad (\text{m disequazioni}) \\ \mu \leq 0 \quad (\text{m disequazioni}) \end{array} \right. \end{array}$$

Ci sono $2m$ disequazioni
in $(m+n-1)$ incognite.

$$\begin{matrix} m+n-1 \\ \textcolor{red}{T} \\ \textcolor{blue}{L} \\ \textcolor{red}{U} \\ \textcolor{blue}{T'} \\ \textcolor{red}{L'} \\ \textcolor{blue}{U'} \end{matrix} \left[\begin{array}{c|c} E^T & I \\ \hline O & I \end{array} \right] \cdot \begin{bmatrix} \pi \\ \mu \end{bmatrix} \leqslant \begin{bmatrix} c \\ o \end{bmatrix}$$

Per il Teorema di caratterizzazione delle basi,
 T, U, T', L' sono gli indici che devo risolvere con l'uguale.

$$\left\{ \begin{array}{ll} E_T^T \pi + \mu_{T'} \leq c_T & \textcolor{red}{T} \\ E_L^T \pi + \mu_{L'} \leq c_L & L \\ E_U^T \pi + \mu_{U'} \leq c_U & U \\ \mu_{T'} \leq 0 & \textcolor{red}{T'} \\ \mu_{L'} \leq 0 & \textcolor{red}{L'} \\ \mu_{U'} \leq 0 & U' \end{array} \right. \xrightarrow{\text{base}} \left\{ \begin{array}{ll} E_T^T \pi + \cancel{\mu_{T'}} = c_T \\ E_U^T \pi + \mu_{U'} = c_U \\ \mu_{T'} = 0 \\ \mu_{L'} = 0 \end{array} \right. \equiv \left\{ \begin{array}{ll} E_T^T \pi = c_T \\ \mu_{U'} = c_U - E_U^T \pi \end{array} \right.$$

$$\therefore (\pi, \mu) = \frac{\left(c_T E_T^{T^{-1}}, 0, 0, c_U - E_U^T \pi \right)}{\pi} \frac{}{\mu}$$

Quindi π è il potenziale del flusso su reti non capacitate, e μ si ricava da π . Per trovare la soluzione di base complementare mi basta calcolare π .

Come verificare che (π, μ) è ammissibile?

(π, μ) rispetta le diseguaglianze sugli indici di base per costruzione. Quindi devo verificarle per gli indici non di base, ovvero L, U .

$$\begin{aligned} & \left\{ \begin{array}{l} E_L^T \pi + \mu_{L'} \leq c_L \\ \mu_U = 0 \end{array} \right|_{\mu_{L'}=0} \iff \left\{ \begin{array}{l} E_L^T \pi \leq c_L \\ \mu_U \leq 0 \end{array} \right. \iff \left\{ \begin{array}{l} -\pi_i + \pi_j \leq c_{ij} \quad \forall (i,j) \in L \\ c_U - E_U^T \pi \leq 0 \implies c_U \leq E_U^T \pi \end{array} \right. \\ & \iff \left\{ \begin{array}{l} -\pi_i + \pi_j \leq c_{ij} \quad \forall (i,j) \in L \\ -\pi_i + \pi_j \geq c_{ij} \quad \forall (i,j) \in U \end{array} \right. \end{aligned}$$

TEOREMA.

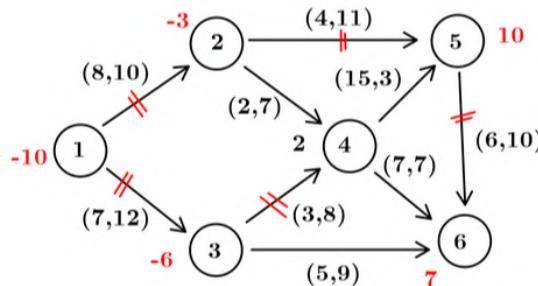
Teorema (Teorema di Bellman II):

Data T, L, U una tripartizione che genera un flusso di base ammissibile.

Se $\begin{cases} c_{ij}^\pi \geq 0 & \forall (i, j) \in L \\ c_{ij}^\pi \leq 0 & \forall (i, j) \in U \end{cases}$, allora siamo all'ottimo.

Un potenziale è **degenero** se uno degli archi di L ha costo ridotto nullo o se uno degli archi di U ha costo ridotto nullo. Sugli archi di T il costo ridotto è sempre nullo per costruzione.

Esempio 3.4 Data una tripartizione, ricavare flussi e potenziale



$$U = \{(4, 5)\}, \quad L = \{(2, 4), (3, 6), (4, 6)\}$$

$$\begin{array}{ccccccccc} T & T & L & T & T & L & U & L & T \\ 12 & 13 & 24 & 25 & 34 & 36 & 45 & 46 & 56 \\ \bar{x} = (11 & -1 & 0 & 14 & 5 & 0 & 3 & 0 & 7) \end{array}$$

Non mi serve calcolare \bar{w} . Il flusso di base è inammissibile, non degenero.

$$\begin{array}{ccccccccc} \pi_1 & \pi_2 & \pi_3 & \pi_4 & \pi_5 & \pi_6 \\ \bar{\pi} = (0 & 8 & 7 & 10 & 12 & 18), \text{ non mi serve calcolare } \bar{\mu}. \end{array}$$

Calcolo i costi ridotti sugli archi di L, U :

$$c_{24}^\pi = c_{24} + \pi_2 - \pi_4 = 2 + 8 - 10 = 0, \text{ va bene perch\'e } (2, 4) \in L \Rightarrow c_{24}\pi \geq 0$$

$$c_{36}^\pi = 5 + 7 - 18 = -6, \text{ non va bene perch\'e } (3, 6) \in L \Rightarrow c_{36}^\pi \geq 0$$

$$c_{45}^\pi = 15 + 10 - 12 = 13, \text{ non va bene perch\'e } (4, 5) \in U \Rightarrow c_{45}^\pi \leq 0$$

$$c_{46}^\pi = 7 + 10 - 18 = -1, \text{ non va bene perch\'e } (4, 6) \in L \Rightarrow c_{46}^\pi \geq 0$$

Dunque non siamo all'ottimo. Inoltre, il potenziale è degenero.

3.2.1. Tabella riassuntiva per flussi e potenziali di base

Riassumiamo nella tabella i possibili casi per i flussi e i potenziali di base:

Flusso di base	Potenziale di base
$\bar{x}_T = E_T^{-1}(b - E_U u_U)$, $\bar{x}_L = 0$, $\bar{x}_U = u_U$	$\bar{\pi}^T = c_T^T E_T^{-1}$
Amm. $0 \leq \bar{x}_{ij} \leq u_{ij} \forall (i, j) \in T$	$c_{ij}^\pi \geq 0 \forall (i, j) \in L$, $c_{ij}^\pi \leq 0 \forall (i, j) \in U$
N.Amm. $\exists (i, j) \in T : \bar{x}_{ij} < 0$ oppure $\exists (i, j) \in T : \bar{x}_{ij} > u_{ij}$	$\exists (i, j) \in L : c_{ij}^\pi < 0$ oppure $\exists (i, j) \in U : c_{ij}^\pi > 0$
Deg. $\exists (i, j) \in T : \bar{x}_{ij} = 0$ oppure $\exists (i, j) \in T : \bar{x}_{ij} = u_{ij}$	$\exists (i, j) \in L : c_{ij}^\pi = 0$ oppure $\exists (i, j) \in U : c_{ij}^\pi = 0$
N.Deg. $\bar{x}_{ij} \neq 0, \bar{x}_{ij} \neq u_{ij} \forall (i, j) \in T$	$c_{ij}^\pi \neq 0 \forall (i, j) \in L$, $c_{ij}^\pi \neq 0 \forall (i, j) \in U$,

3.3. Simplesso per flusso di costo minimo capacitato

Sia $\{T, L, U\}$ una tripartizione che genera un flusso di base (x_T, x_L, x_U) ammissibile, cioè $0 \leq x_T \leq u_T$.

Se le condizioni di Bellman sono verificate i.e. $\begin{cases} c_{ij}^\pi \geq 0 \forall (i, j) \in L \\ c_{ij}^\pi \leq 0 \forall (i, j) \in U \end{cases}$, allora il flusso è ottimo.

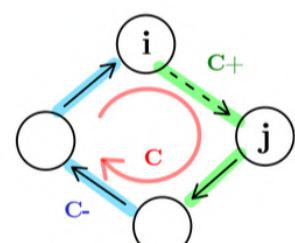
Altrimenti, significa che un indice di L o di U viola le condizioni di Bellman. Si esegue quindi un passo del simplesso.

3.3.1. L'arco che viola Bellman è in L

$\exists (i, j) \in L : c_{ij}^\pi < 0$. L'arco entrante è il primo di questi archi.

Si crea un ciclo \mathcal{C} , lo partiziona in $\mathcal{C}^+, \mathcal{C}^-$.

Opero il cambiamento di flusso $x(\theta) = \begin{cases} \bar{x}_{ij} + \theta & \forall (i, j) \in \mathcal{C}^+ \\ \bar{x}_{ij} - \theta & \forall (i, j) \in \mathcal{C}^- \\ \bar{x}_{ij} & \forall (i, j) \notin \mathcal{C} \end{cases}$



Il Teorema di correttezza del simplesso per flussi dice che

$c^T x(\theta) = c^T \bar{x} + \theta c_{ij}^\pi$. Siccome l'arco che viola Bellman è in L , allora $\theta c_{ij}^\pi < 0$. Quindi $c^T x(\theta) \leq c^T \bar{x}$.

Similmente al simplesso per reti non capacitate, i bilanci sono rispettati $\forall \theta$. Quindi l'unica condizione è che $0 \leq x(\theta) \leq u$.

i. $0 \leq x(\theta)$ può essere violata dagli archi di \mathcal{C}^- , quindi $\theta \leq \min_{\mathcal{C}^-} \{\bar{x}_{ij}\}$.

ii. $x(\theta) \leq u$ può essere violata dagli archi di \mathcal{C}^+ , quindi $\theta \leq \min_{\mathcal{C}^+} \{\mu_{ij} - \bar{x}_{ij}\}$

Definisco $\theta^- := \min_{\mathcal{C}^-} \{\bar{x}_{ij}\}$, $\theta^+ := \min_{\mathcal{C}^+} \{\mu_{ij} - \bar{x}_{ij}\}$.

Siccome θ non deve violare entrambe le condizioni, devo prendere $\theta := \min\{\theta^+, \theta^-\}$

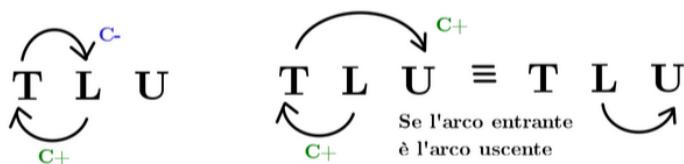
- Se $\theta = \theta^-$, allora l'arco uscente è un arco di \mathcal{C}^- che si è svuotato e quindi va in L .

- Se $\theta = \theta^+$, allora l'arco uscente è un arco di \mathcal{C}^+ che si è saturato e quindi va in U .

N.B. Se $\theta^+ = \theta^-$, prendo come θ quello relativo al primo arco in ordine lessicografico.

In entrambi i casi (se si rispetta la regola di anticiclo di Bland) l'algoritmo non cicla mai.

- Se l'arco uscente è in \mathcal{C}^- , siccome l'arco entrante è in \mathcal{C}^+ per costruzione, non può essere lo stesso.
- Se l'arco uscente è in \mathcal{C}^+ , potrebbe essere lo stesso arco entrante, ma, siccome è entrato da L ed esce in U , allora la tripartizione è effettivamente cambiata.



Osservazione: se l'arco uscente è uguale a quello entrante, T resta invariato e un passo del simplesso equivale a trasferire un arco da L a U .

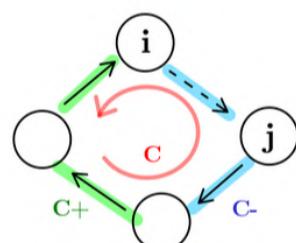
3.3.2. L'arco che viola Bellman è in U

$\exists(i, j) \in U : c_{ij}^\pi > 0$. L'arco entrante è il primo di questi archi.

Si crea un ciclo \mathcal{C} , con la differenza che questa volta è in senso opposto, lo partiziono in $\mathcal{C}^+, \mathcal{C}^-$.

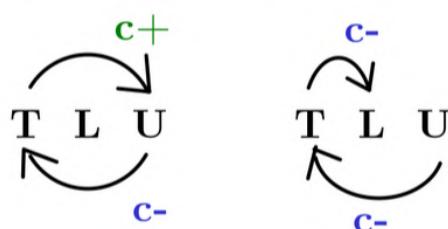
$$\text{Opero il cambiamento di flusso } x(\theta) = \begin{cases} \bar{x}_{ij} + \theta & \forall (i, j) \in \mathcal{C}^+ \\ \bar{x}_{ij} - \theta & \forall (i, j) \in \mathcal{C}^- \\ \bar{x}_{ij} & \forall (i, j) \notin \mathcal{C} \end{cases}$$

Questa volta $c^T x(\theta) = c^T \bar{x} - \theta c_{ij}^\pi$. Siccome l'arco che viola Bellman è in U , allora $\theta c_{ij}^\pi > 0$. Vale sempre $c^T x(\theta) \leq c^T \bar{x}$.



Come prima, $\theta = \min\{\theta^+, \theta^-\}$

- Se l'arco uscente è in \mathcal{C}^+ , siccome l'arco entrante è in \mathcal{C}^- per costruzione, non può essere lo stesso.
- Se l'arco uscente è in \mathcal{C}^- , potrebbe essere lo stesso arco entrante, ma, siccome è entrato da U ed esce in L , allora la tripartizione è effettivamente cambiata.



ALGORITMO.

Algoritmo del simplesso per flussi

1. Trovo una partizione degli archi (T, L, U) , con T un albero di copertura, che generi un flusso di base ammissibile
2. Calcolo il flusso di base \bar{x} : $\bar{x}_T = E_T^{-1}(b - E_U u_U)$, $\bar{x}_L := 0$, $\bar{x}_U := u_U$
 Calcolo il potenziale di base $\bar{\pi}^T := c_T^T E_T^{-1}$. Indico con $c_{ij}^{\bar{\pi}} := c_{ij} + \bar{\pi}_i - \bar{\pi}_j$ il costo ridotto $\forall(i, j)$
3. **if** le condizioni di Bellman sono verificate i.e.
 $c_{ij}^{\bar{\pi}} \geq 0 \forall(i, j) \in L$
 $c_{ij}^{\bar{\pi}} \leq 0 \forall(i, j) \in U$ **then STOP** ($\bar{x}, \bar{\pi}$ sono ottimi).
else calcolo l'arco entrante

$$(p, q) := \min \left\{ \begin{array}{l} \{(i, j) \in L : c_{ij}^{\bar{\pi}} < 0\} \cup \{(i, j) \in U : c_{ij}^{\bar{\pi}} > 0\} \end{array} \right\}$$
 dove il minimo è fatto rispetto all'ordine lessicografico sugli archi.
4. L'arco (p, q) forma un ciclo \mathcal{C} con gli archi di T . Fissa su \mathcal{C} un verso concorde con (p, q) se $(p, q) \in L$, oppure un verso discordo con (p, q) se $(p, q) \in U$.
 Indica con \mathcal{C}^+ gli archi del ciclo concordi con il verso di \mathcal{C} , e con \mathcal{C}^- quelli discordi.
 Calcolo $\theta^+ := \min_{e^+} \{u_{ij} - \bar{x}_{ij}\}$, $\theta^- := \min_{e^-} \{\bar{x}_{ij}\}$, $\theta := \min \{\theta^+, \theta^-\}$
if $\theta = +\infty$ **then STOP** (il flusso di costo minimo ha valore $-\infty$)
else trova l'arco uscente

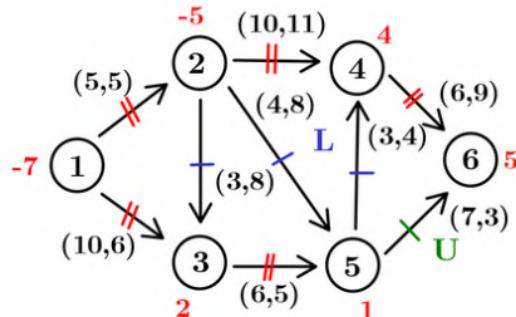
$$(r, s) := \min \left\{ \begin{array}{l} \{(i, j) \in \mathcal{C}^- : \bar{x}_{ij} = \theta\} \cup \{(i, j) \in \mathcal{C}^+ : u_{ij} - \bar{x}_{ij} = \theta\} \end{array} \right\}$$
5. Aggiorno la partizione e torno al passo 2.

Osservazione: nell'algoritmo, la condizione **if** $\theta = +\infty$ **then STOP** si verifica solo se $U = \emptyset$. Questo succede nelle reti non capacitate perché $\mathcal{C}^- = \emptyset \implies \theta = +\infty$, quindi $c^T x = -\infty$.

Nelle reti capacitate, $\mathcal{C}^- = \emptyset \implies \mathcal{C}^+ \neq \emptyset$, quindi $\theta \neq +\infty$ sempre.
 $\mathcal{C}^+ = \emptyset \implies \mathcal{C}^- \neq \emptyset$, quindi $\theta \neq +\infty$ sempre.

Infatti, il poliedro delle reti capacitate è limitato, quindi, per Weierstrass, esiste soluzione ottima limitata.

Esempio 3.5 Eseguire un passo del simplex data la partizione T, L, U di un problema di flusso capacitato:



$$\begin{array}{ccccccccc} T & T & L & T & L & T & T & L & U \\ 12 & 13 & 23 & 24 & 25 & 35 & 46 & 54 & 56 \end{array}$$

Calcolo il flusso di base: $\bar{x} = (1 \ 6 \ 0 \ 6 \ 0 \ 4 \ 2 \ 0 \ 3)$

\bar{x} è degenero perché $(1, 3)$ è saturo (ma non importa per fare un passo del simplex).

$$c^T x = 182$$

$$\pi_1 \ \pi_2 \ \pi_3 \ \pi_4 \ \pi_5 \ \pi_6$$

Calcolo i potenziali: $\bar{\pi} = (0 \ 5 \ 10 \ 15 \ 16 \ 21)$

$$c_{23}^\pi = c_{23} + \pi_2 - \pi_3 = 3 + 5 - 10 = -2 < 0 \Rightarrow (2, 3) \text{ è l'arco entrante}$$

$$\mathcal{C}^+ = \{(1, 2), (2, 3)\}, \quad \mathcal{C}^- = \{(1, 3)\}$$

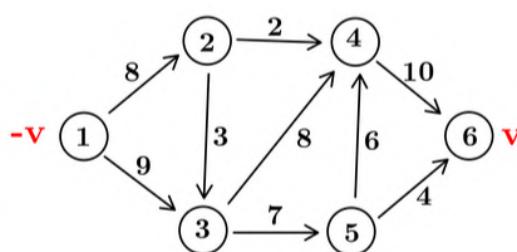
$$\theta^- := \min\{\bar{x}_{ij} : (i, j) \in \mathcal{C}^-\} = 6,$$

$$\theta^+ := \min\{\mu_j - \bar{x}_{ij} : (i, j) \in \mathcal{C}^+\} = \min\{5 - 1, 8 - 0\} = 4$$

$$\theta := \min\{\theta^+, \theta^-\} = 4 \therefore \text{l'arco } (1, 2) \text{ è l'arco uscente.}$$

4. Flusso massimo

Si considerino n nodi e m archi. Su ciascun arco (i, j) è definita una capacità u_{ij} . Non sono specificati costi sugli archi. Voglio massimizzare il flusso dal nodo s al nodo t .

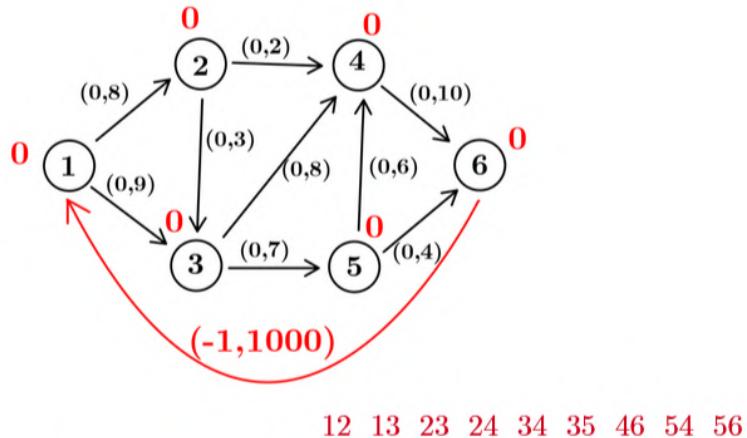


Il poliedro non è mai vuoto poiché $\vec{x} = \vec{0}$ è sempre una soluzione ammissibile. Il flusso massimo può fare zero; in tale caso significa che non esiste un cammino orientato dal nodo s al nodo t .

Pongo a $-v$ il bilancio del nodo s , pongo a v il bilancio del nodo t . Il flusso deve essere coerente, ovvero i bilanci di tutti i nodi intermedi devono essere zero. Questo non è un flusso di costo minimo poiché i bilanci sono variabili.

$$\begin{cases} \max v \\ Ex = b : b = \begin{cases} -v, & i = s \\ 0, & i \neq s, t \\ v, & i = t \end{cases} \\ 0 \leq x \leq u \\ v \geq 0 \end{cases}$$

Il flusso massimo si può trasformare in un flusso di costo minimo se aggiungo un arco fittizio da t a s con costo -1 e capacità arbitrariamente grande, tipo 10000.



In riferimento all'esempio precedente, se prendo $\bar{x} = (2 \ 4 \ 0 \ 2 \ 0 \ 4 \ 2 \ 0 \ 4)$ soluzione di base ammissibile, aggiungendo l'arco $(6, 1)$ fittizio ottengo

$$(\bar{x}, \bar{v}) = (12 \ 13 \ 23 \ 24 \ 34 \ 35 \ 46 \ 54 \ 56 \ 61)$$

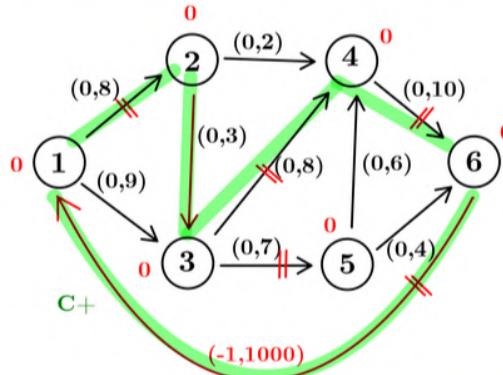
Linprog vuole attirare il massimo flusso verso il nodo 6 perché così guadagna -1 ogni volta che manda un'unità di flusso in 1.

Modello.

Flusso massimo:

$$\begin{cases} \max v \\ Ex = b : b = \begin{cases} -v, & i = s \\ 0, & i \neq s, t \\ v, & i = t \end{cases} \\ 0 \leq x \leq u \\ v \geq 0 \end{cases} \equiv \begin{cases} \min -v \\ \left[\begin{array}{c} 1 \\ 0 \\ \dots \\ 0 \\ -1 \end{array} \right] \cdot \begin{bmatrix} x \\ v \end{bmatrix} = 0 \quad (\text{flusso di costo minimo equiv.}) \\ 0 \leq x \leq u \\ v \geq 0 \end{cases}$$

Esempio 4.1 Eseguire un passo del simplex sulla rete del seguente problema di flusso massimo:



12 13 23 24 34 35 46 54 56 61

Parto dalla soluzione di base ammissibile $(x, v) = (2 \ 9 \ 0 \ 2 \ 5 \ 4 \ 7 \ 0 \ 4 \ 11)$
generata dalla tripartizione $T = \{(1, 2), (3, 4), (3, 5), (4, 6), (6, 1)\}$, $U = \{(1, 3), (2, 4), (5, 6)\}$

Il vettore dei potenziali $\pi = (0, 0, 1, 1, 1, 1)$

$$c_{13}^\pi = -1 \checkmark,$$

$c_{23}^\pi = -1$ viola bellman $\Rightarrow (2, 3)$ arco entrante

$\mathcal{C}^+ = \{(1, 2), (2, 3), (3, 4), (4, 6), (6, 1)\}$, $\mathcal{C}^- = \emptyset \Rightarrow \theta^- := +\infty$ per convenzione.
 $\theta^+ = \min\{6, 3, 3, 3, +\infty\} = 3 \Rightarrow \theta = 3$ relativo all'arco $(2, 3)$.

$(2, 3)$ è sia l'arco entrante che l'arco uscente. Non è un problema perché $(2, 3)$ entra da L , ma esce in U .

12 13 23 24 34 35 46 54 56 61

Il nuovo flusso $(x, v) = (5 \ 9 \ 3 \ 2 \ 8 \ 4 \ 10 \ 0 \ 4 \ 14)$
+3 +3 +3 +3 +3

4.1. Algoritmo di Ford Fulkerson

Un algoritmo più efficiente del simplex per calcolare il flusso massimo è l'algoritmo di Ford-Fulkerson. Noi vedremo la variante Edmonds-Karp (FFEK).

Erano gli anni '90, e alla Technical University of Amsterdam Edmonds stava per tenere un convegno di ricerca operativa. Pappalardo, che non aveva mai visitato Amsterdam, rimase affascinato dalla città e dalla grande fontana al centro della piazza. Durante l'evento, Edmonds fece un ingresso trionfale: arrivò a cavallo, attraversò il portone principale, smontò e legò il cavallo alla fontana centrale. Erano altri tempi, e certe eccentricità erano quasi la norma.

Definizione.

Definizione: Un taglio della rete è una partizione N_S, N_T dei nodi tali che $s \in N_S, t \in N_T$

Definizione: Gli archi diretti del taglio sono $A^+ := \{(i, j) \in A : i \in N_S, j \in N_T\}$

Definizione: La portata di un taglio è $u(N_S, N_T) := \sum_{(i,j) \in A^+} u_{ij}$

Nell'esempio, $N_S = \{1\}, N_T = \{2, 3, 4, 5\}, A^+ = \{(1, 2), (1, 3)\}, u(N_S, N_T) = 17$

TEOREMA.

Teorema (max-flow / min-cut)

Il flusso massimo è minore o uguale delle portate di ogni taglio: $\bar{v} \leq u(N_S, N_T) \forall u$

Il flusso massimo è uguale al taglio di portata minima: $\bar{v} = \min\{u(N_S, N_T)\}$

La dimostrazione (che non vedremo) si basa sul fatto che il problema di taglio di portata minima è il duale del problema di flusso massimo.

Possiamo quindi procedere per enumerazione totale, ma il numero di tagli possibili è 2^n , quindi è inefficiente. Useremo l'algoritmo di FFEK, che si basa su questo teorema e dunque è corretto.

Definizione.

Definizione (grafo residuo e capacità residue):

dato un flusso x ammissibile, il grafo residuo $G(x) = (N, A(x))$ è un grafo con gli stessi nodi del grafo G mentre per ogni arco (i, j) esiste un arco (j, i) .

Le capacità residue r_{ij} sono definite $r_{ij} := u_{ij} - x_{ij} + x_{ji}$

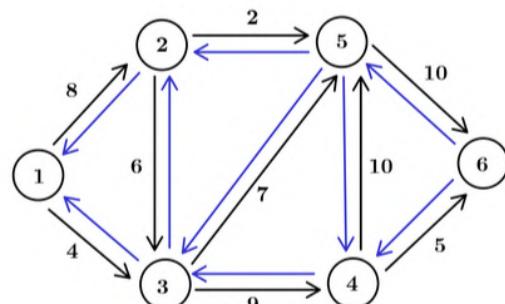
Ad esempio, se $\bar{x} = \vec{0}$, allora

$$r_{12} = u_{12} - \bar{x}_{12} + \bar{x}_{21} = 8 - 0 + 0 = 8$$

$$r_{21} = u_{21} - \bar{x}_{21} + \bar{x}_{12} = 0 - 0 + 0 = 0$$

Se $\bar{x} = 2 \Rightarrow r_{12} = 6, r_{21} = 2$

Le capacità residue rappresentano quanto flusso posso ancora mandare.



Definizione.

Definizione (cammino aumentante):

Un cammino aumentante (C_{aum}) sul grafo residuo è un cammino orientato da s a t formato dagli archi (i, j) tali per cui $r_{ij} > 0$.

La sua portata $\delta := \min\{r_{ij} : (i, j) \in C_{aum}\}$

Nell'esempio, se $\bar{x} = \vec{0} \Rightarrow 1, 2, 5, 6$ è un cammino aumentante con $\delta = 2$.

L'algoritmo di Ford-Fulkerson cerca ad ogni iterazione un cammino aumentante. Se tale cammino esiste, allora il flusso viene aggiornato spedendo il massimo flusso possibile sul cammino aumentante, uguale alla portata. Se non esistono cammini aumentanti, allora il flusso corrente è di valore massimo e l'algoritmo trova anche un taglio di capacità minima. Allora si capisce che gli archi fittizzi inseriti nel grafo residuo servono ad **annullare le decisioni sbagliate**.

Inoltre, l'algoritmo trova anche un taglio di capacità minima visto che min cut è il duale del max flow.

ALGORITMO.

Algoritmo del simplex per flussi non capacitati

1. Inizializza il flusso $\bar{x} = \vec{0}$, che è sicuramente ammissibile.
2. Costruisci il grafo residuo $G(x)$.
3. If esiste un cammino aumentante C_{aum} then
 calcola $\delta = \min\{r_{ij} : (i, j) \in C_{aum}\}$
 spedisci δ unità di flusso lungo C_{aum} cioè $x_{ij} = \begin{cases} x_{ij} + \delta & (i, j) \in C_{aum} \\ x_{ij} - \delta & (j, i) \in C_{aum} \end{cases}$
 torna al passo 2.
 else STOP, x è un flusso massimo ed un taglio di capacità minima è (N_S, N_T)
 dove $N_S = \{i \in N : \exists C_{aum} \text{ da } s \text{ a } i \in G(x)\}$, $N_T = N \setminus N_S$

4.1.1. Come trovare un cammino aumentante

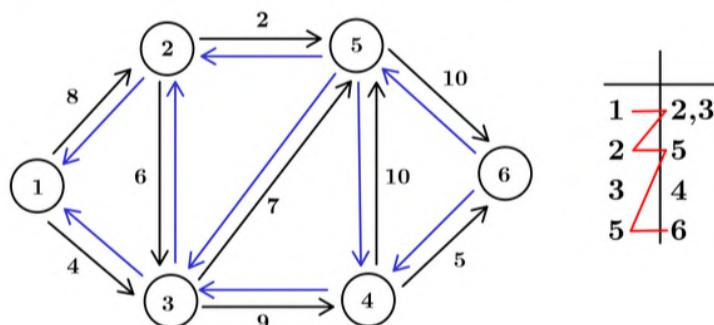
Se non ci sono regole precise per scegliere i cammini aumentanti, la convergenza dell'algoritmo può essere molto lenta. C'è una variante dell'algoritmo, dovuta ad Edmonds-Karp in cui si sceglie il cammino aumentante col minimo numero di archi.

Algoritmo.

Algoritmo di Edmonds-Karp:

1. Disegno una tabella 1x2. Sulla colonna di sinistra scrivo il nodo s . Sulla colonna di destra scrivo in ordine lessicografico i nodi di $FS(s)$ tali che $r_{ij} > 0$.
2. Si prende primo nodo (FIFO) della colonna di destra, diciamo i e si scrive sotto, nella colonna di sinistra. Nella colonna di destra si scrivono i nodi di $FS(i)$ tali che $r_{ij} > 0$ non ancora visitati.
3. Itero il passo 2. finché nella colonna di destra non appare il nodo t . Il cammino aumentante si trova vedendo qual'è il nodo nella colonna di sinistra che ha permesso di raggiungere t . Si cerca la riga in cui questo nodo appare sulla colonna di destra e si procede iterativamente guardando quale fosse il nodo da cui è stato raggiunto finché non si arriva a s .

Esempio 4.2 trova il flusso massimo sulla seguente rete utilizzando l'algoritmo FFEK:



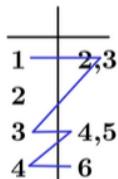
$12 \ 13 \ 23 \ 25 \ 34 \ 35 \ 45 \ 46 \ 56$

Sia $\bar{x} = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$ ammissibile

$12 \ 13 \ 23 \ 25 \ 34 \ 35 \ 45 \ 46 \ 56$

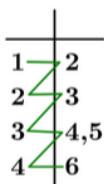
$C_{aum} = 1, 2, 5, 6 \Rightarrow \delta = 2, \bar{x} = (2 \ 0 \ 0 \ 2 \ 0 \ 0 \ 0 \ 0 \ 2), v = 2$

Aggiorno le capacità residue: $r_{12} = 6, r_{21} = 2, r_{25} = 0, r_{52} = 2, r_{56} = 8, r_{65} = 2$



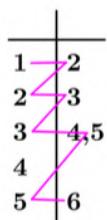
$12 \ 13 \ 23 \ 25 \ 34 \ 35 \ 45 \ 46 \ 56$
 $1, 3, 4, 6 \Rightarrow \delta = 4, \bar{x} = (2 \ 4 \ 0 \ 2 \ 4 \ 0 \ 0 \ 4 \ 2), v = 6$

Aggiorno le capacità residue.



$12 \ 13 \ 23 \ 25 \ 34 \ 35 \ 45 \ 46 \ 56$
 $1, 2, 3, 4, 6 \Rightarrow \delta = 1, \bar{x} = (3 \ 4 \ 1 \ 2 \ 5 \ 0 \ 0 \ 5 \ 2), v = 7$

Aggiorno le capacità residue.



$1, 2, 3, 5, 6 \Rightarrow \delta = 5, \bar{x} = (8 \ 4 \ 6 \ 2 \ 5 \ 5 \ 0 \ 5 \ 7), v = 12$
Aggiorno le capacità residue.

$N_S = \{1\}$ quindi il taglio di capacità minima vale 12.

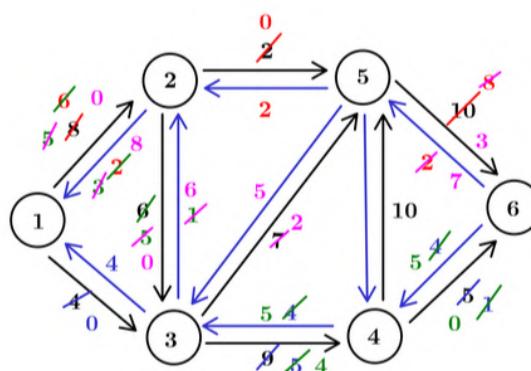


Figura 2: Ad ogni passo, se $(i, j) \in C_{aum}$, allora $r_{ij} = r_{ij} - \delta, r_{ji} = r_{ji} + \delta$

In questo caso non c'è stato bisogno di utilizzare gli archi fittizi, ma non è detto in generale.

Per migliorare l'efficienza dell'algoritmo, si può iniziare senza l'ausilio degli archi fittizi: si procede normalmente finché non si trovano più un cammino aumentante. Allora introduco gli archi fittizi e proseguo.

Per vedere se siamo arrivati all'ottimo senza aggiungere gli archi fittizi si possono analizzare i piani di taglio del grafo: qualora ce ne sia uno con capacità residue degli archi diretti del taglio tutte nulle allora siamo all'ottimo (è il caso dell'esempio).

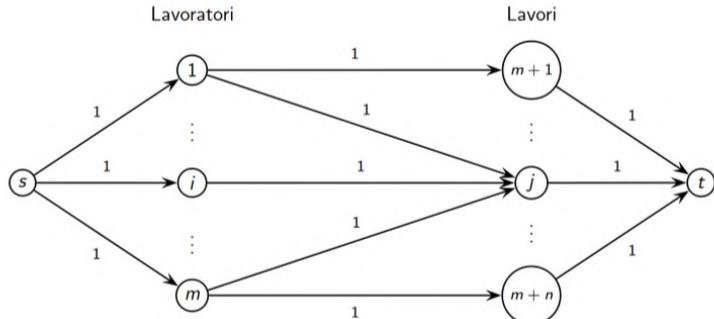
4.2. Accoppiamento di cardinalità massima

Supponiamo di avere m lavoratori e n lavori da svolgere. Ogni lavoratore sa svolgere alcuni lavori, e ad ogni lavoratore si può assegnare al massimo un lavoro.

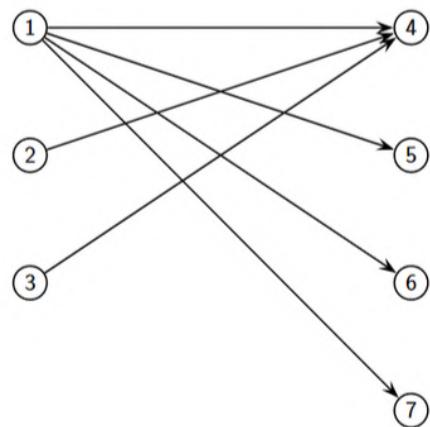
Voglio assegnare i lavori in modo da eseguire il numero massimo di lavori.

Questo è un problema di flusso massimo: i nodi $1, \dots, m$ rappresentano i lavoratori, i nodi $m+1, \dots, m+n$ rappresentano i lavori, due nodi fittizi s, t .

Gli archi collegano ogni lavoratore a tutti i lavori che sa svolgere, il nodo s a tutti i lavoratori e tutti i lavori al nodo t .



Ad esempio, se ho 3 lavoratori e 4 lavori collegati nel seguente modo:



Il problema può essere formulato come:

$$\begin{cases} \max x_{14} + x_{15} + x_{16} + x_{17} + x_{24} + x_{34} \\ x_{14} + x_{15} + x_{16} + x_{17} \leq 1 \\ x_{24} \leq 1 \\ x_{34} \leq 1 \\ x_{14} + x_{24} + x_{34} \leq 1 \\ x_{15} \leq 1 \\ x_{16} \leq 1 \\ x_{17} \leq 1 \\ x_{ij} \in \{0, 1\} \end{cases}$$

Fine teoria

Programmazione Non Lineare (PNL)

La programmazione non lineare è una branca della ricerca operativa che si occupa di risolvere problemi in cui la funzione obiettivo e/o i vincoli sono non lineari.

$$\begin{cases} \min f(x), & f: \mathbb{R}^n \rightarrow \mathbb{R} \\ x \in D, & D \subseteq \mathbb{R}^n \end{cases}$$

Per studiare la PNL, riprenderemo alcuni concetti di Analisi 2 e vedremo dei metodi per determinare estremi liberi di funzioni. Successivamente, estenderemo il discorso al caso vincolato.

1. Un primo problema di PNL

In una nuova linea di produzione sono integrati sei robot, ciascuno dei quali può ruotare di 360° lungo l'asse verticale. I robot non devono interferire nei loro raggi d'azione. Si vuole minimizzare la lunghezza di cavo in fibra ottica utilizzato per collegare i robot.

Robot	1	2	3	4	5	6
Raggio d'azione (cm)	120	800	100	70	45	120

Abbiamo 12 variabili decisionali in totale, che sono $(x_1, y_1), \dots, (x_6, y_6)$.

La funzione obiettivo è la distanza $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} + \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} + \dots$

Siccome i robot non devono collidere, allora i vincoli sono $d_{ij} \geq r_i + r_j$.

Se li scrivo in termini di variabili decisionali: $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \geq r_i + r_j$.

Aggiungo, senza perdere generalità, anche i vincoli $0 \leq x \leq M$, $0 \leq y \leq M$.

Voglio che la funzione sia di classe C^1 (per far valere le ipotesi del th. di Fermat), quindi devo togliere le radici. Per i vincoli è sufficiente elevare tutto alla seconda, mentre per la funzione obiettivo posso togliere la radice perché è una funzione monotona crescente, e quindi minimizzare $\sqrt{f(x)}$ equivale a minimizzare $f(x)$.

$$\begin{cases} \min(x_1 - x_2) + (y_1 - y_2) + (x_1 - x_3) + (y_1 - y_3) + \dots \\ (x_i - x_j) + (y_i - y_j) \geq (r_i + r_j)^2 \\ 0 \leq x \leq M \\ 0 \leq y \leq M \end{cases}$$

2. Prerequisiti di Analisi II

2.1. Derivate parziali, gradiente, Hessiana

$$\frac{\partial f(\bar{x})}{\partial x_k} := \lim_{t \rightarrow 0} \frac{f(\bar{x} + te^k) - f(\bar{x})}{t} = \lim_{t \rightarrow 0} \frac{f(\bar{x}_1 + \dots + \bar{x}_k t + \dots + \bar{x}_n) - f(\bar{x}_1, \dots, \bar{x}_n)}{t}$$

La maggior parte dei limiti a più variabili non è risolvibile, ma le derivate parziali sono sempre calcolabili, poiché stiamo facendo il limite in una sola variabile t . Infatti, la derivata parziale rispetto a una componente x_k è la derivata della funzione che si ottiene restringendosi sull'asse x_k .

Il **gradiente** è una notazione con cui raccogliere tutte le derivate parziali:

$$\nabla f(\bar{x}) := \left(\frac{\partial f(\bar{x})}{\partial x_1}, \dots, \frac{\partial f(\bar{x})}{\partial x_n} \right)$$

Sotto opportune ipotesi (gradiente definito in un intorno e la funzione è differenziabile) posso calcolare le **derivate seconde**:

$$\frac{\partial f_{x_j}}{\partial x_k} =: \frac{\partial^2 f}{\partial x_j \partial x_k} \quad \text{e in particolare} \quad \frac{\partial f_{x_k}}{\partial x_k} =: \frac{\partial^2 f}{\partial x_k^2}$$

La **matrice Hessiana** $Hf(x)$ è una matrice $n \times n$ che ha per componenti tutte le derivate seconde. D'ora in poi, vedremo il caso particolare di $n = 2$.

$$Hf(\bar{x}) := \left(\frac{\partial^2 f}{\partial x_i \partial x_j} \right)_{i,j=1}^n \quad \text{e nel caso } n = 2, Hf(\bar{x}) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix}$$

2.2. Teoremi per minimi e massimi

2.2.1. Strumenti per la matrice Hessiana.

In questa sezione, esponiamo alcuni risultati fondamentali di Analisi 2 e Algebra Lineare che saranno utili per formulare condizioni necessarie e sufficienti per identificare i punti di minimo e massimo.

Teorema.

Teorema (teorema di Schwartz):

La matrice Hessiana è simmetrica, ovvero $\frac{\partial^2 f}{\partial x_1 \partial x_2} = \frac{\partial^2 f}{\partial x_2 \partial x_1}$ nel caso $n = 2$.

Teorema.

Teorema (teorema spettrale):

Una matrice simmetrica ha autovalori reali.

Ricordando che un **autovalore** di una matrice A è un $\lambda \in \mathbb{C}$ tale che $\det(A - \lambda I) = 0$.
 $\sigma(A)$ è lo **spettro** di una matrice A ovvero l'insieme di tutti i suoi autovalori.

Definizione.

Definizione (matrice definita/semidefinita positiva/negativa):

- Una matrice è definita positiva se $\langle A \cdot x, x \rangle > 0 \forall x \in \mathbb{R}^n$,
- Una matrice è semidefinita positiva se $\langle A \cdot x, x \rangle \geq 0 \forall x \in \mathbb{R}^n$,
- Una matrice è definita negativa se $\langle A \cdot x, x \rangle < 0 \forall x \in \mathbb{R}^n$,
- Una matrice è semidefinita negativa se $\langle A \cdot x, x \rangle \leq 0 \forall x \in \mathbb{R}^n$,

Siccome una matrice simmetrica ha autovalori reali:

- Una matrice simmetrica è definita positiva se $\lambda_i > 0 \forall \lambda_i$
- Una matrice simmetrica è semidefinita positiva se $\lambda_i \geq 0 \forall \lambda_i$
- Una matrice simmetrica è definita negativa se $\lambda_i < 0 \forall \lambda_i$
- Una matrice simmetrica è semidefinita negativa se $\lambda_i \leq 0 \forall \lambda_i$

2.2.2. Teoremi per i minimi e massimi

Possiamo ora concentrarci sui teoremi che permettono di individuare e classificare i punti stazionari di una funzione come minimi o massimi locali.

TEOREMA.

Teorema I (teorema di Fermat):

Sia $f: \mathbb{R}^n \rightarrow \mathbb{R}$ una funzione che ammette minimo (o massimo) locale in \bar{x} .

Allora $\nabla f(\bar{x}) = 0$.

In realtà, questo teorema vale sotto l'ipotesi aggiuntiva che f sia derivabile in tutte le direzioni, ma le funzioni che studieremo lo sono sempre. I punti $\bar{x}: \nabla f(\bar{x}) = 0$ si chiamano **punti stazionari**.

Questa è una **condizione necessaria** ai minimi (o massimi). Vuol dire che un punto di minimo (o massimo) ha necessariamente gradiente nullo: i punti di minimo (o massimo) locale sono un sottoinsieme dei punti stazionari.

TEOREMA.

Teorema II (condizioni sulla hessiana):

Sia \bar{x} un punto stazionario:

- i. **Condizione sufficiente:** Se $Hf(\bar{x})$ è definita positiva (negativa) allora \bar{x} è un punto di minimo (massimo) locale
- ii. **Condizione necessaria:** Se \bar{x} è un punto di min (max) locale allora $Hf(\bar{x})$ è semidefinita positiva (negativa).

Siccome $Hf(\bar{x})$ è simmetrica (teo Schwartz), allora i suoi autovalori sono reali (teo spettrale) e il teorema ora diventa:

- i. Se $\lambda_i > 0 \forall \lambda_i \in \sigma(Hf(\bar{x}))$, allora \bar{x} è un minimo locale
- ii. Se \bar{x} è un minimo locale, allora $\lambda_i \geq 0 \forall \lambda_i \in \sigma(Hf(\bar{x}))$

La ii. è una condizione necessaria ai minimi, mentre la i. è una **condizione sufficiente**.

Esempio 2.1

- Se nel punto (x_1, x_2) la funzione f ha gradiente nullo e Hf ha $\lambda_1 = 2, \lambda_2 = 0$, allora (x_1, x_2) è un ottimo candidato ad essere il minimo perché verifica due condizioni necessarie.
- Se nel punto (x_1, x_2) la funzione f ha gradiente nullo e Hf ha $\lambda_1 = -5, \lambda_2 = -6$, allora (x_1, x_2) è un punto di massimo locale poiché verifica la condizione sufficiente.

N.B. Se $Hf(\bar{x})$ è **indefinita**, ovvero ha autovalori sia positivi sia negativi, allora \bar{x} è un punto di sella e non può essere né un minimo né un massimo locale. Se $Hf(\bar{x})$ ha tutti autovalori nulli, allora il comportamento della funzione deve essere analizzato con altri metodi.

Ricapitoliamo tutti i casi possibili per i segni degli autovalori
(con un piccolo abuso di notazione):

- $Hf(\bar{x}) \geq 0 \implies \bar{x}$ candidato minimo locale, sicuramente non è massimo locale
- $Hf(\bar{x}) > 0 \implies \bar{x}$ è un minimo locale
- $Hf(\bar{x}) \leq 0 \implies \bar{x}$ candidato massimo locale, sicuramente non è minimo locale
- $Hf(\bar{x}) < 0 \implies \bar{x}$ è un massimo locale
- $Hf(\bar{x}) \geq 0$ (indefinita) $\implies \bar{x}$ è un punto di sella
- $Hf(\bar{x}) = 0 \implies$ boh!

2.2.3. Perché non esiste il simplex per la PNL

Una caratteristica fondamentale della PL è che l'ottimo si può cercare sul vertice del poligono. Il simplex si muove tra i vertici del poligono, che corrispondono a una soluzione di base ammissibile. Ogni vertice ha un numero finito di spigoli, e il simplex sfrutta proprio la struttura discreta dei problemi lineari per raggiungere sempre l'ottimo. Il simplex si ferma quando è verificata la condizione sufficiente i.e. quando la duale complementare è ammissibile.

Come mai un approccio simile non funziona per la PNL?

Nella PNL ci sono infinite direzioni di spostamento poiché non ci sono spigoli discreti in generale. Anche se ci fossero, la soluzione ottima potrebbe trovarsi in un punto interno del poliedro, oppure nel mezzo del vertice.

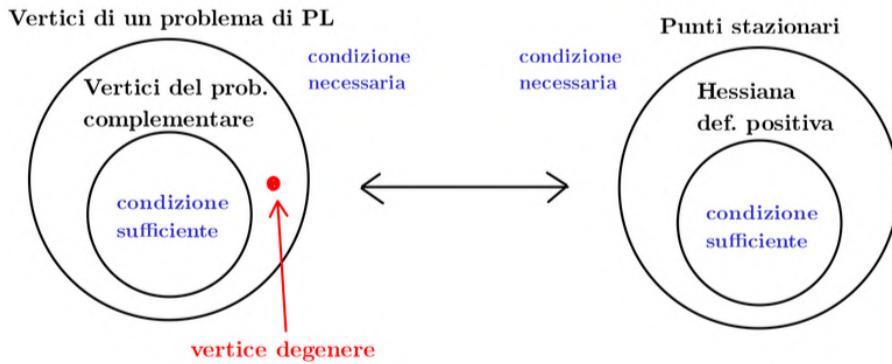


Figura 1: Confronto tra condizioni necessarie / sufficienti nella PL e PNL.

Un altro problema è che nella PL la condizione sufficiente è globale, mentre nella PNL la condizione sufficiente è locale.

2.3. Derivate direzionali e restrizioni

$$\frac{\partial f}{\partial d}(\bar{x}) := \lim_{t \rightarrow 0} \frac{f(\bar{x} + td) - f(\bar{x})}{t}$$

è la derivata di f lungo la direzione di d .

Data una funzione $f: \mathbb{R}^n \rightarrow \mathbb{R}$, si definisce restrizione della funzione una funzione $\phi: \mathbb{R} \rightarrow \mathbb{R}$ ottenuta restringendo il dominio della funzione. Nel nostro caso ci interessa la restrizione a una semiretta, dove scriviamo l'equazione parametrica della semiretta e la sostituiamo nella funzione f .
 $\phi(t) := f(\bar{x} + td)$

Ad esempio $f(x) = x_1^2 - x_2^2$, $\bar{x} = (3, 1)$, $\bar{d} = (1, 1)$

$$\begin{aligned} \frac{\partial f}{\partial d}(\bar{x}) &:= \lim_{t \rightarrow 0} \frac{f((3, 1) + t(1, 1)) - f(3, 1)}{t} = \lim_{t \rightarrow 0} \frac{f((3, 1) + (t, t)) - f(3, 1)}{t} = \\ &= \lim_{t \rightarrow 0} \frac{f(3+t, 1+t) - f(3, 1)}{t} = \lim_{t \rightarrow 0} \frac{(3+t)^2 - (1+t)^2 - 8}{t} \text{ dove } \phi := (3+t)^2 - (1+t)^2 \end{aligned}$$

è la restrizione della funzione f sulla semiretta $\bar{x} + t\bar{d}$

3. Classi di funzioni

Ci sono categorie di funzioni che condividono specifiche proprietà matematiche utili nella risoluzione dei problemi di ottimizzazione.

3.1. Funzioni quadratiche

Definizione.

Definizione (funzione quadratica):

Una funzione quadratica è una funzione polinomiale di secondo grado:

$$f(x) = \frac{1}{2}x^T Qx + c^T x, \text{ o in modo equivalente } f(x) = \langle x, Qx \rangle + \langle c^T, x \rangle$$

dove Q è una matrice $n \times n$, c^T è un vettore di \mathbb{R}^n

Ogni matrice Q è associata ad una funzione quadratica:

$$\text{ad esempio } Q = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}, Qx = \begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_1 + 2x_2 \\ 2x_1 + 3x_2 \end{bmatrix},$$

$$x^T Qx = (x_1, x_2) \begin{bmatrix} x_1 + 2x_2 \\ 2x_1 + 3x_2 \end{bmatrix} = x_1^2 + 2x_1x_2 + 2x_1x_2 + 3x_2^2 = x_1^2 + 3x_1^2 + 4x_1x_2$$

Al contrario, una funzione quadratica è associata ad infinite matrici Q :

ad esempio, $f(x) = 3x_1^2 + 5x_1x_2 - x_2^2$ è associata a

$$Q = \begin{bmatrix} 3 & 1 \\ 4 & -1 \end{bmatrix}, \text{ ma anche } Q = \begin{bmatrix} 3 & 2 \\ 3 & -1 \end{bmatrix} \text{ etc...}$$

Come mai la funzione quadratica è definita moltiplicata per $\frac{1}{2}$?

Osservo che $\nabla \left(\frac{1}{2}x^T Qx + c^T x \right) = Qx + c$, $\nabla(Qx + c) = Q$

- Quindi, se la funzione è $f(x) = \frac{1}{2}x^T Qx + c^T x$, allora $Hf(x) = Q$,

ovvero Q è una matrice simmetrica che rappresenta l'Hessiana della funzione! Allora c'è una corrispondenza biunivoca tra la funzione quadratica $f(x)$ e la matrice Q .

- Invece, se la funzione è $f(x) = x^T Qx + c^T x$, allora $Hf(x) = 2Q$.

L'altra forma è semplicemente più comoda perché mi permette di trovare subito l'Hessiana.

Esempio 3.1

1. Trovare la Hessiana della funzione quadratica $3x_1^2 + 5x_2^2$ nella forma $f(x) = \frac{1}{2}x^T Qx$

$$f(x) = 3x_1^2 + 5x_2^2 = \frac{1}{2}6x_1 + 10x_2^2 \implies Q = \begin{bmatrix} 6 & 0 \\ 0 & 10 \end{bmatrix} = Hf(x)$$

2. Trovare la Hessiana della funzione quadratica $6x_1^2 + 3x_1x_2 + 5x_2^2$ nella forma $f(x) = x^T Qx$

$$f(x) = 6x_1^2 + 3x_1x_2 + 5x_2^2 \implies Q = \begin{bmatrix} 6 & 3/2 \\ 3/2 & 5 \end{bmatrix} \implies Hf(x) = 2Q = \begin{bmatrix} 12 & 3 \\ 3 & 10 \end{bmatrix}$$

Le funzioni quadratiche sono comode perché la matrice Hessiana è costante.

3.1.1. La funzione quadprog

È disponibile su Matlab la funzione `quadprog`, utile per risolvere problemi di programmazione quadratica.

$$\text{quadprog}(Q, c, A, b, A_{eq}, b_{eq}, LB, UB) \text{ risolve il problema} \begin{cases} \min \frac{1}{2} x^T Q x + c^T x \\ Ax \leq b \\ A_{eq} = b_{eq} \\ LB \leq x \leq UB \end{cases}$$

3.2. Funzioni convesse

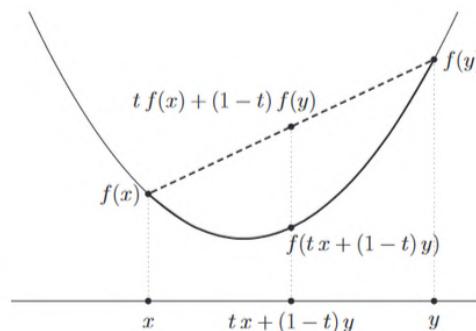
Definizione.

Definizione (funzione convessa):

Sia $f: \mathbb{R}^n \rightarrow \mathbb{R}$.

f si dice **convessa** se $f(tx + (1-t)y) \leq t f(x) + (1-t)f(y)$
 $\forall t \in [0, 1] \ \forall x, y \in \mathbb{R}^n$

Presi due punti del dominio, il segmento di estremi $(x, f(x))$ e $(y, f(y))$ giace al di sopra del grafico della funzione f ristretta al segmento $[x, y]$.



Questa definizione è impraticabile perché ci sono tre quantificatori universali (\forall). Le funzioni convesse si possono caratterizzare in base alle proprietà delle loro derivate parziali seconde.

Teorema.

Teorema (caratterizzazione delle funzioni convesse):

Sia $f: \mathbb{R}^n \rightarrow \mathbb{R}$, $f \in C^2$.

Allora f è convessa se e solo se $Hf(x)$ è semidefinita positiva $\forall x$

Nel caso di una funzione quadratica, la matrice hessiana è costante e coincide con Q , quindi f è convessa se e solo se gli autovalori di Q sono non negativi. Similmente, f è concava sse $Hf(x) \leq 0$.

TEOREMA.

Teorema (punti stazionari di funzioni convesse):

Sia $f: \mathbb{R}^n \rightarrow \mathbb{R}$ convessa. Se $\nabla f(\bar{x}) = 0$ allora \bar{x} è un punto di minimo globale.

Questo teorema mi dice che basta trovare un punto che annulla il gradiente per trovare il minimo globale. Similmente, se f è concava e $\nabla f(\bar{x}) = 0$, allora \bar{x} è un punto di massimo globale.

Pappalardo fece tappa a New York, in un dipartimento interamente dedicato alla ricerca operativa, un paradiso per gli appassionati del settore. All'ingresso vi era un enorme cartello con Snoopy che citava la scritta "Happiness is thinking linear". Ma Pappalardo, più legato alla programmazione non lineare, immaginava un'altra scritta: "Happiness is thinking convex". Solo in America, pensò, potevano permettersi un cartello con Snoopy così grande.

3.3. Funzioni coercive

Definizione.

Definizione (funzione coerciva):

Una funzione coerciva è una funzione $f: \mathbb{R}^n \rightarrow \mathbb{R}$ continua tale che $\lim_{\|x\| \rightarrow +\infty} f(x) = +\infty$

Una funzione anticoerciva è una funzione $f: \mathbb{R}^n \rightarrow \mathbb{R}$ continua tale che $\lim_{\|x\| \rightarrow +\infty} f(x) = -\infty$

In altre parole, una funzione coerciva è una funzione continua che tende a $+\infty$ in tutte le direzioni.

TEOREMA.

Teorema:

Una funzione coerciva ammette minimo globale.

Una funzione anticoerciva ammette massimo globale.

4. Domini della PNL

Ricapitoliamo i quattro teoremi importanti visti finora per l'ottimizzazione libera. Sia $f: \mathbb{R}^n \rightarrow \mathbb{R}$:

- **Teorema di Fermat:** \bar{x} punto di minimo $\implies \nabla f(\bar{x}) = 0$
- **Condizione necessaria sulla hessiana:** \bar{x} punto di minimo $\implies Hf(\bar{x}) \geq 0$
- **Condizione sufficiente sulla hessiana:** $Hf(\bar{x}) > 0 \implies \bar{x}$ punto di minimo
- **Punti stazionari di funzioni convesse:** f convessa, $\nabla f(\bar{x}) = 0 \implies \bar{x}$ è punto di minimo globale.

In generale, i problemi di PNL sono vincolati da un dominio $D \subseteq \mathbb{R}^n$, $\begin{cases} \min f(x) \\ x \in D \end{cases}$

$$D := \left\{ x \in \mathbb{R}^n : g_1(x) \leq 0, \dots, g_m(x) \leq 0, h_1(x) = 0, \dots, h_p(x) = 0 \right\}$$

$$= \left\{ x \in \mathbb{R}^n : g(x) \leq 0, h(x) = 0 \right\} \text{ dove } g: \mathbb{R}^n \rightarrow \mathbb{R}^m, h: \mathbb{R}^n \rightarrow \mathbb{R}^p$$

I domini sono quindi intersezioni di m vincoli con \leq e p vincoli di uguaglianza.

Esempio 4.1 $D := \left\{ x_1^2 + x_2^2 \leq 4 \right\}$ è un cerchio di raggio 2 centrato nell'origine.

$$\Rightarrow g_1(x_1, x_2) = x_1^2 + x_2^2 - 4.$$

D'ora in poi i problemi saranno della forma $\begin{cases} \min / \max f(x) \\ g(x) \leq 0, g: \mathbb{R}^n \rightarrow \mathbb{R}^m \\ h(x) = 0, h: \mathbb{R}^n \rightarrow \mathbb{R}^p \end{cases}$

Finora abbiamo visto i domini come poliedri: $D = \{Ax \leq b\}$.

In questi casi abbiamo che $p = 0$, $g(x) = Ax - b$, dove $g_i(x) = a_i^T x - b_i \forall i = 1, \dots, m$.

Infatti, ci sono tanti vincoli $g_i(x) \leq 0$ quante righe della matrice A .

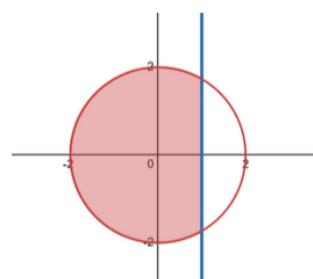
4.1. Domini compatti

Noi assumiamo che tutte le funzioni f, g, h siano di classe C^2 . Questo vuol dire che i domini D con i quali lavoreremo sono **chiusi**, i.e. non ci sono vincoli con minore stretto. Alla fine del modello si aggiunge il vincolo $-M \leq x \leq M$, in modo che D sia anche **limitato** (a meno chè non ci siamo dimenticati dei vincoli).

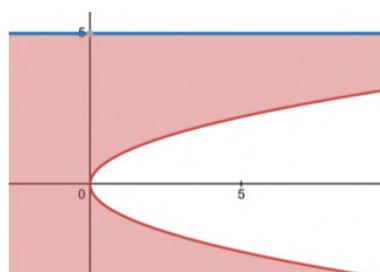
Siccome D è chiuso e limitato, allora, per il Teorema di Weierstrass, esiste il minimo.

Esempio 4.2 Disegna il dominio dato $g_1(x) = x_1^2 + x_2^2 - 4$, $g_2(x) = x_1 - 1$

Come si capisce se la zona da considerare è quella interna o esterna ad una curva? Si disegna la curva $g_i(x) = 0$, poi si prova a vedere se un punto interno rispetta $g_i(x) \leq 0$. Se sì, la zona da considerare è quella interna, se no è quella esterna.

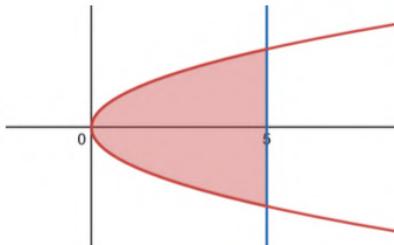


Esempio 4.3 Disegna il dominio dato $g_1(x) = x_1 - x_2^2$, $g_2(x) = x_2 - 5$



D è chiuso ma non è limitato

Esempio 4.4 Disegna il dominio dato $g_1(x) = -x_1 + x_2^2$, $g_2(x) = x_1 - 5$



Questa volta i punti da considerare sono quelli interni alla parabola.

4.2. Domini convessi

Definizione.

Definizione (dominio convesso):

D è convesso sse $tx + (1-t)y \in D \quad \forall x, y \in D, t \in [0, 1]$.

A noi ci piacciono molto i domini convessi perché tutti gli algoritmi che vedremo si spostano lungo una direzione. Nei domini convessi, quando trovo una direzione su cui spostarmi sono sicuro che tutto il segmento è ammmissible. Questo non è detto se D non è convesso.



Figura 2: In un poliedro non convesso il segmento potrebbe tornare ad essere ammmissible

Proposizione.

Proposizione (condizione sufficiente):

Se tutte le g_i sono convesse e tutte le h_i sono lineari, allora D è convesso.

Nell'ultimo esempio, in cui $g_1(x) = -x_1 + x_2^2$, $g_2(x) = x_1 - 5$:

$g_2(x)$ è convessa poiché le funzioni lineari sono convesse

$$g_1(x) \text{ è convessa poiché } \nabla g_1(x) = (-1, 2x_2) \implies Hg_1(x) = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix} \geq 0$$

4.3. Domini regolari

Intuitivamente, un dominio regolare è una regione ben definita, senza buchi, angoli troppo stretti o contorni irregolari. Nel piano, è una porzione di superficie delimitata da curve regolari. Vediamo alcune classi di domini regolari a cui fare riferimento.

TEOREMA.

Teorema (classi di domini regolari):

Sia $D \subseteq \mathbb{R}^n$ un dominio, ci sono tre classi di domini che sono regolari:

1. Poliedri: se g_i e h_j sono tutte funzioni lineari, allora D è regolare

2. Condizione di Slater:

Se g_i sono tutte convesse, h_i sono tutte lineari e $\exists \bar{x} : g_i(\bar{x}) < 0 \forall i$,

allora D è regolare. In pratica, i domini convessi con un punto interno sono regolari.

3. Condizione di Mangasarian:

Se $\bar{x} \in D$ ed i vettori $\begin{cases} \nabla g_i(\bar{x}), & i \in A(\bar{x}) \\ \nabla h_i(\bar{x}), & i = 1, \dots, p \end{cases}$ sono linearmente indipendenti $\forall \bar{x}$,

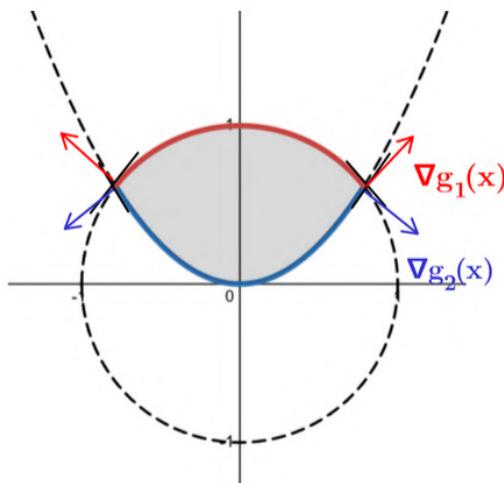
allora D è regolare. In pratica, se i gradienti dei vincoli attivi ($i \in A(\bar{x})$) e i gradienti dei vincoli di uguaglianza sono linearmente indipendenti, allora il dominio è regolare.

(I vincoli attivi sono definiti $A(\bar{x}) : g_i(\bar{x}) = 0$).

Esempio 4.5 Determina se il seguente dominio è regolare secondo le tre classi:

$$g_1(x) = x_1^2 + x_2^2 - 1, \quad g_2(x) = x_1^2 - x_2$$

$D = \{x_1^2 + x_2^2 \leq 1, \quad x_1^2 - x_2 \leq 0\}$ è l'intersezione tra un cerchio e una parabola.



- D non è un poliedro
- D rispetta la condizione di Slater, poiché g_1 e g_2 sono funzioni convesse e si vede ad occhio che esiste un punto interno al dominio.
- D rispetta la condizione di Mangasarian poiché:
 - I punti rossi sono regolari poiché $\nabla g_1(x) = (2x_1, 2x_2) \neq 0$ è linearmente indipendente da sè stesso.
 - I punti blu sono regolari poiché $\nabla g_2(x) = (2x_1, -1) \neq 0$ è linearmente indipendente da sè stesso.

– Insomma, i punti che bisogna controllare per davvero sono le intersezioni:

$$\begin{cases} x_1^2 + x_2^2 = 1 \\ x_2 = x_1^2 \end{cases} \Rightarrow x_2 = \frac{\sqrt{5}-1}{2}, \quad x_1 = \sqrt{\frac{\sqrt{5}-1}{2}}$$

$$\begin{bmatrix} \nabla g_1(x) \\ \nabla g_2(x) \end{bmatrix} = \begin{bmatrix} 2x_1 & 2x_2 \\ 2x_1 & -1 \end{bmatrix} = \begin{bmatrix} 2\sqrt{\frac{\sqrt{5}-1}{2}} & 2 \\ 2\sqrt{\frac{\sqrt{5}-1}{2}} & -1 \end{bmatrix},$$

$\det \neq 0$ quindi i gradienti sono linearmente indipendenti

Dunque D è un dominio regolare.

5. Condizioni di ottimalità del primo ordine

5.1. Regola dei moltiplicatori di Lagrange (Analisi 2)

Questa parte può essere saltata.

Ad Analisi 2 abbiamo visto l'ottimizzazione di funzioni soggette a un solo vincolo di uguaglianza.

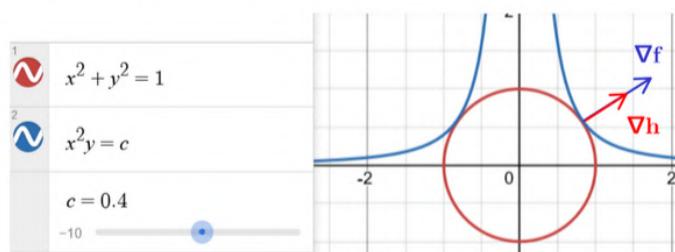
Secondo me fornisce una prima idea utile per comprendere il concetto alla base del teorema LKKT.

Ripropongo quindi quanto discusso ad Analisi 2.

$$\max_{x \in D \subseteq \mathbb{R}^n} f(x), \text{ dove } f(x) = x_1^2 x_2, D = \{x_1^2 + x_2^2 = 1\}$$

Intanto $D = \{h(x) = 0\}$, dove la funzione implicita $h(x) = x_1^2 + x_2^2 - 1$.

Le linee di livello di $f(x)$ sono $f(x) = c$. Voglio c più grosso possibile, il che succede quando la curva $f(x) = c$ è tangente a $h(x) = 0$.



Il gradiente è perpendicolare alle linee di livello di $f(x)$. L'idea chiave è che nel punto di tangenza di $f(x) = c$ e $h(x) = 0$ i gradienti puntano nella stessa direzione, ovvero $\nabla f(x) = \lambda \nabla h(x)$, dove λ si chiama moltiplicatore di Lagrange.

Per trovare la soluzione ottima basta scrivere il sistema $\begin{cases} \nabla f(x) = \lambda \nabla h(x) \\ h(x) = 0 \end{cases}$

In realtà, potrebbe anche essere che $\nabla h(x) = 0$.

Teorema.**Teorema (moltiplicatori di Lagrange):**

Sia \bar{x} un punto di massimo del problema $\begin{cases} \max f(x) \\ h(x) = 0 \end{cases}$

Allora vale una delle due alternative:

- $\nabla h(x) = 0$ (caso irregolare)
- $\nabla f(x) = \lambda \nabla h(x)$ (caso regolare)

5.1.1. Funzione Lagrangiana

Posso riscrivere queste due alternative definendo la funzione lagrangiana:

$$\mathcal{L}(x) := f(x) + \lambda h(x) \implies \nabla \mathcal{L}(x) = \nabla f(x) + \lambda \nabla h(x)$$

Quindi la condizione $\nabla \mathcal{L}(x) = 0$ equivale a $\nabla f(x) = -\lambda \nabla h(x)$.

Il caso regolare si ha quando $\nabla \mathcal{L}(x) = 0$, mentre il caso irregolare si ha quando $\nabla h(x) = 0$.

Posso scrivere la lagrangiana anche nella forma Fritz John:

$$\mathcal{L}(x) := \lambda_1 f(x) + \lambda_2 h(x) \text{ dove } \lambda_1 \text{ e } \lambda_2 \text{ non si annullano contemporaneamente.}$$

Allora $\nabla \mathcal{L}(x) = 0$ equivale a $\lambda_1 \nabla f(x) + \lambda_2 \nabla h(x) = 0$.

Il caso irregolare si ha quando $\lambda_1 = 0 \implies \lambda_2 \nabla h(x) = 0 \Big|_{\lambda_2 \neq 0} \implies \nabla h(x) = 0$

Il caso regolare si ha quando $\lambda_0 \neq 0 \implies \nabla f(x) + \frac{\lambda_2}{\lambda_1} \nabla h(x) = 0$, dove $\lambda = \frac{\lambda_2}{\lambda_1}$

Teorema.**Teorema (moltiplicatori di Lagrange in forma Fritz-John):**

Sia \bar{x} un punto di massimo del problema $\begin{cases} \max f(x) \\ h(x) = 0 \end{cases}$

Allora $\exists \lambda_1, \lambda_2 : \nabla \mathcal{L}(x) = 0$ dove $\mathcal{L}(x) := \lambda_1 f(x) + \lambda_2 h(x)$ lagrangiana in forma F. John.

5.2. Condizione necessaria ai minimi o massimi

Sia il problema $\begin{cases} \min f(x) \\ g(x) \leq 0 \\ h(x) = 0 \end{cases}$

Si definisce Lagrangiana del problema la funzione $\mathcal{L}(x, \lambda, \mu) = f(x) + \sum_{i=1}^m \lambda_i g_i(x) + \sum_{j=1}^p \mu_j h_j(x)$

dove $x \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^m$, $\mu \in \mathbb{R}^p$

TEOREMA.

Teorema 1 (Lagrange-Karush-Kuhn-Tucker):

Sia $D \subseteq \mathbb{R}^n$ regolare. Siano $f, g, h \in C^1$

- i. Se $\bar{x} \in D$ è un punto di minimo locale, allora $\exists \bar{\lambda} \in \mathbb{R}^m, \bar{\lambda} \geq 0, \exists \bar{\mu} \in \mathbb{R}^p$
tale che $(\bar{x}, \bar{\lambda}, \bar{\mu})$ soddisfa il sistema LKKT.
- ii. Se $\bar{x} \in D$ è un punto di massimo locale, allora $\exists \bar{\lambda} \in \mathbb{R}^m, \bar{\lambda} \leq 0, \exists \bar{\mu} \in \mathbb{R}^p$
tale che $(\bar{x}, \bar{\lambda}, \bar{\mu})$ soddisfa il sistema LKKT.

Il sistema LKKT:

$$\begin{cases} \nabla \mathcal{L}(\bar{x}, \bar{\lambda}, \bar{\mu}) = \nabla f(\bar{x}) + \sum_{i=1}^m \bar{\lambda}_i \nabla g_i(\bar{x}) + \sum_{j=1}^p \bar{\mu}_j \nabla h_j(\bar{x}) = 0 & n \text{ equazioni} \\ \langle \bar{\lambda}, g(\bar{x}) \rangle = 0 & m \text{ equazioni} \\ h(\bar{x}) = 0 & p \text{ equazioni} \end{cases}$$

$$\langle \bar{\lambda}, g(\bar{x}) \rangle = 0 \iff \bar{\lambda}_i g_i(\bar{x}) = 0 \quad \forall i = 1, \dots, m$$

Quindi devo risolvere un sistema non lineare di $(n + m + p)$ equazioni in $(n + m + p)$ incognite.
Il minimo è una soluzione del sistema LKKT.

Questo teorema fornisce solo condizioni necessarie per i minimi locali, ma non sufficienti. Se il sistema LKKT non ha soluzioni, significa che non esiste minimo, né massimo. Se il dominio D è compatto, allora, per Weierstrass, il sistema LKKT ha almeno due soluzioni.

Osservazione:

I punti interni al dominio, dove non sono attivi vincoli di uguaglianza o disegualanza, sono localmente liberi. In questi casi, le condizioni LKKT si riducono a richiedere che il gradiente della funzione obiettivo sia nullo.

Sul bordo, uno o più vincoli diventano attivi. Qui le condizioni LKKT includono anche la necessità di considerare come il gradiente della funzione obiettivo interagisce con i gradienti dei vincoli attivi. Il teorema è particolarmente rilevante per gestire i punti sul bordo, poiché il gradiente non si annulla in generale.

Come si classificano le soluzioni di LKKT?

- Se $\bar{\lambda}$ sono discordi allora \bar{x} è un punto di sella,
- Se $\bar{\lambda}$ sono concordi positivi, allora \bar{x} è un punto di minimo locale o di sella,
- Se $\bar{\lambda}$ sono concordi negativi, allora \bar{x} è un punto di massimo locale o di sella.

Esempio 5.1 Cerchiamo il minimo della funzione $f(x) = x_1^2 + x_2^2 - 4x_1 - 2x_2$ sulla regione $D = \{x_1^2 + 4x_2^2 - 4x_1 - 8x_2 + 4 \leq 0, x_1 - 2 \leq 0\}$

I punti stazionari sono dati: $\bar{x}^1 = (0, 1)$, $\bar{x}^2 = (2, 0)$, $\bar{x}^3 = (2, 2)$, $\bar{x}^4 = (2, 1)$
 $\nabla f(x) = (2x_1 - 4, 2x_2 - 2)$, $\nabla g_1(x) = (2x_1 - 4, 8x_2 - 8)$, $\nabla g_2(x) = (1, 0)$

$$\begin{cases} \nabla f(x) + \lambda_1 \nabla g_1(x) + \lambda_2 \nabla g_2(x) = 0 \\ \lambda_1 g_1(x) = 0 \\ \lambda_2 g_2(x) = 0 \end{cases} \Leftrightarrow \begin{cases} 2x_1 - 4 + \lambda_1(2x_1 - 4) + \lambda_2 = 0 \\ 2x_2 - 2 + \lambda_1(8x_2 - 8) = 0 \\ \lambda_1(x_1^2 + 4x_2^2 - 4x_1 - 8x_2 + 4) = 0 \\ \lambda_2(x_1 - 2) = 0 \end{cases}$$

$$\bar{x}^1 = (0, 1) \Rightarrow \bar{\lambda}^1 = (-1, 0)$$

- Trovo λ_2 : $\lambda_2(0 - 2) = 0 \Rightarrow \lambda_2 = 0$
- Trovo λ_1 :
 - $\triangleright 2 - 2 + \lambda_1(8 - 8) = 0 \Rightarrow 0 = 0$ non mi da nessuna informazione
 - $\triangleright \lambda_1(0 + 4 - 0 - 8 + 4) = 0 \Rightarrow 0 = 0$ non mi da nessuna informazione
 - $\triangleright -4 + \lambda_1(-4) = 0 \Rightarrow \lambda_1 = -1$ bingo!

$$\bar{x}^2 = (2, 0) \Rightarrow \bar{\lambda}^2 = \left(-\frac{1}{4}, 0 \right)$$

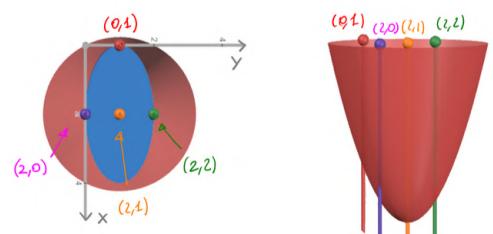
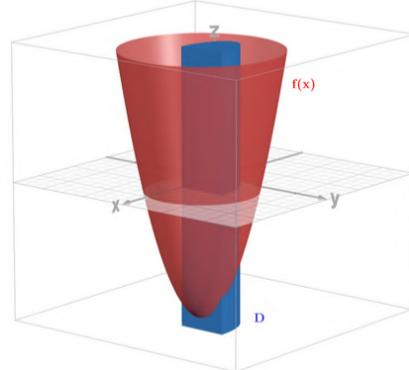
- Trovo λ_1 : $-2 + \lambda_1(-8) = 0 \Rightarrow \lambda_1 = -\frac{1}{4}$
- Trovo λ_2 : $4 - 4 - \frac{1}{4}(4 - 4) + \lambda_2 = 0 \Rightarrow \lambda_2 = 0$

$$\bar{x}^3 = (2, 2) \Rightarrow \bar{\lambda}^3 = \left(-\frac{1}{4}, 0 \right)$$

- Trovo λ_1 : $4 - 2 + \lambda_1(16 - 8) = 0 \Rightarrow \lambda_1 = -\frac{1}{4}$
- Trovo λ_2 : $4 - 4 - \frac{1}{4}(4 - 4) + \lambda_2 = 0 \Rightarrow \lambda_2 = 0$

$$\bar{x}^4 = (2, 1) \Rightarrow \bar{\lambda}^4 = (0, 0)$$

- Trovo λ_1 : $\lambda_1(4 + 4 - 8 - 8 + 4) = 0 \Rightarrow \lambda_1 = 0$
- Trovo λ_2 : $4 - 4 + \lambda_2 = 0 \Rightarrow \lambda_2 = 0$



Siccome D è limitato (intersezione di un ellisse e di un semipiano), allora esiste massimo e minimo globale.

\bar{x}	$\bar{\lambda}$	$f(\bar{x})$	Punto
$(0, 1)$	$(-1, 0)$	-1	max globale (per Weierstrass)
$(2, 0)$	$\left(-\frac{1}{4}, 0 \right)$	-4	max locale o sella
$(2, 2)$	$\left(-\frac{1}{4}, 0 \right)$	-4	max locale o sella
$(2, 1)$	$(0, 0)$	-5	min globale (per Weierstrass)

Esempio 5.2 Trova il minimo di $f(x) = x_1^2 - 2x_2^2 + 8x_2$ sull'insieme $D = \{1 - x_1^2 - x_2^2 \leq 0, x_1^2 - x_2 - 2 \leq 0\}$.

La funzione f è un paraboloide rovesciato, quindi è anticoerciva. Dunque ammette massimo globale. Non faccio i calcoli, vado direttamente alla classificazione dei punti stazionari:

\bar{x}	$\bar{\lambda}$	$f(\bar{x})$	Punto
(0, -2)	(0, 16)	-24	minimo locale o sella
(0, -1)	(-6, 0)	-10	massimo locale o sella
(0, 1)	(2, 0)	6	minimo locale o sella
(0, 2)	(0, 0)	8	massimo globale (perché anticoerciva)
$\left(\frac{\sqrt{15}}{2}, \frac{7}{4}\right)$	(0, 1)	< 8	minimo locale o sella
$\left(-\frac{\sqrt{15}}{2}, \frac{7}{4}\right)$	(0, 1)	< 8	minimo locale o sella

ALGORITMO.

Algoritmo (come classificare i punti stazionari):

Mi chiedo: D è regolare?

- ↳ Se no, questa parte è coperta alla fine degli appunti
- ↳ Se si, mi chiedo: D è limitato?
 - ↳ Se sì, esiste minimo globale e massimo globale
 - ↳ Se no, mi chiedo: f è coerciva o anticoerciva?
 - ↳ Se f è coerciva, esiste minimo globale,
 - ↳ Se f è anticoerciva, esiste massimo globale,
 - ↳ Altrimenti, non ho nessuna informazione.

Procedo calcolando $\bar{\lambda}$, $\bar{\mu}$ e classificando i punti stazionari con l'analisi locale (vedere dopo).

Esempio 5.3 Trova il minimo di $f(x) = x_1 + x_2$, dove $g_1(x) = x_1^2 + x_2^2 + 2x_1x_2 - 36$, $g_2(x) = 1 - x_1x_2$.

Siccome il dominio è limitato, allora f ammette minimo e massimo globale.

\bar{x}	$\bar{\lambda}$	$f(\bar{x})$	Punto
(−1, −1)	(0, −1)	−2	massimo locale o sella
(1, 1)	(0, 1)	2	minimo locale o sella
$(3 + \sqrt{8}, 3 - \sqrt{8})$	$\left(-\frac{1}{12}, 0\right)$	6	massimo globale
$(3 - \sqrt{8}, 3 - \sqrt{8})$	$\left(-\frac{1}{12}, 0\right)$	6	massimo globale
$(-3 - \sqrt{8}, -3 - \sqrt{8})$	$\left(\frac{1}{12}, 0\right)$	−6	minimo globale
$(-3 - \sqrt{8}, -3 + \sqrt{8})$	$\left(\frac{1}{12}, 0\right)$	−6	minimo globale

Per stabilire se $(-1, -1)$ e $(1, 1)$ sono delle selle bisogna fare l'analisi locale (che affronteremo in seguito). Possiamo anche risolvere il problema geometricamente usando le linee di isocosto.

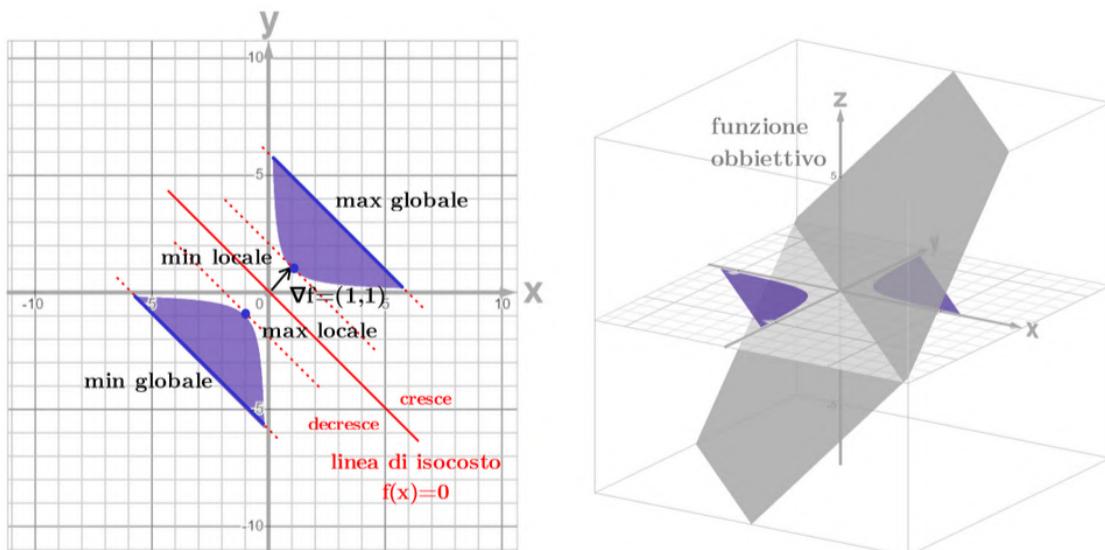
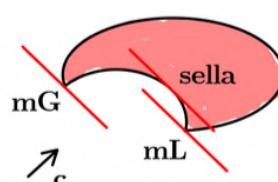


Figura 3: Siccome tutti i punti dell'intorno di $(1, 1)$ sono sopra la linea di isocosto, allora $(1, 1)$ è un punto di minimo locale. Siccome tutti i punti dell'intorno di $(-1, -1)$ sono sotto la linea di isocosto, allora $(-1, -1)$ è un punto di massimo locale.

Se nell'intorno ci fossero dei punti che stanno sopra la linea di isocosto, e dei punti che stanno sotto, allora sarebbe una sella.



5.3. Condizione sufficiente ai minimi o massimi

TEOREMA.

Teorema 2 (condizione sufficiente):

Sia $D \subseteq \mathbb{R}^n$ regolare, siano $f, g_i, h_j \in C^1$

- i. Siano f, g_i convesse e h_j lineari $\forall i = 1, \dots, m \quad \forall j = 1, \dots, p$

Se \bar{x} soddisfa LKKT con $(\bar{\lambda}, \bar{\mu}) \in \mathbb{R}^{m+p}$ e $\bar{\lambda} \geq 0$ allora \bar{x} è minimo globale,

- ii. Siano f concava, g_i convesse e h_j lineari $\forall i = 1, \dots, m \quad \forall j = 1, \dots, p$

Se \bar{x} soddisfa LKKT con $(\bar{\lambda}, \bar{\mu}) \in \mathbb{R}^{m+p}$ e $\bar{\lambda} \leq 0$ allora \bar{x} è massimo globale.

Si definiscono **problemi convessi** quelli che hanno f convessa, D convesso,

Si definiscono **problemi concavi** quelli che hanno f concava, D convesso.

Osservazioni:

Siano soddisfatte le ipotesi del Teorema 2 nel caso i.

- Se $\bar{\lambda} \geq 0$ allora non esiste minimo globale i.e. D è illimitato. Questo perchè, per il Teorema 1, se esistesse minimo locale, si avrebbe $\bar{\lambda} \geq 0$,
- Se ci sono due punti \bar{x} tali che $\bar{\lambda} \geq 0$, allora il problema ha due minimi globali i.e. i punti assumono lo stesso valore nella funzione obiettivo.

TEOREMA.

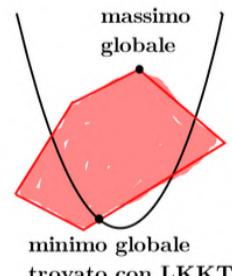
Teorema 3 (condizione sufficiente per i poliedri):

Sia $D \subseteq \mathbb{R}^n$ un **poliedro limitato**.

- i. Se f è convessa, allora esiste un vertice che è massimo globale.
- ii. Se f è concava, allora esiste un vertice che è minimo globale.

Questo teorema è un caso particolare del Teorema 2.

Si dimostra facilmente osservando che una funzione convessa ha massimo sul bordo del poliedro poiché un punto stazionario interno è un minimo globale (vedere il teorema sui punti stazionari di funzioni convesse).



Ottimizzazione libera vs vincolata:

- Ottimizzazione libera:

- Se f è convessa, è facile trovare minimo, in quanto ogni punto stazionario è minimo globale.
- Se f è concava, è facile trovare massimo, in quanto ogni punto stazionario è massimo globale.

- Ottimizzazione vincolata (quando D è un poliedro):
 - Se f è convessa, è facile trovare massimo globale, in quanto si trova necessariamente su un vertice di D
 - Se f è concava, è facile trovare minimo globale, in quanto si trova necessariamente su un vertice di D

Osservazione filosofica (man vs machine): gli algoritmi come **quadprog** eccellono nel trovare minimi in problemi convessi, ma non sono altrettanto performanti per i massimi. L'uomo, invece, è più abile nel trovare massimi in problemi convessi (poiché può sfruttare il Teorema 3), ma risulta meno efficace nel gestire i minimi senza strumenti computazionali.

5.4. Analisi locale dei punti stazionari

Serve a classificare i punti stazionari, determinando se siano minimi locali, massimi locali o punti di sella.

Teorema.

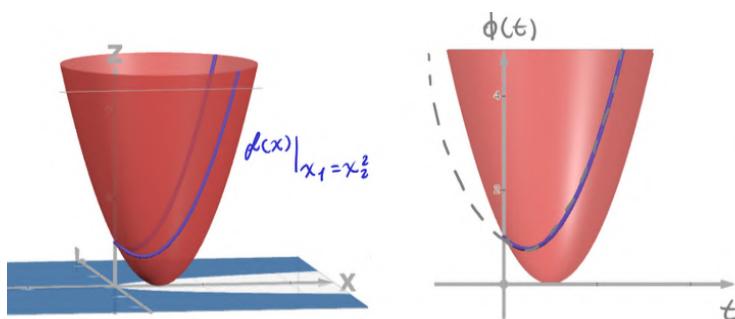
Teorema:

Se \bar{x} è un punto di minimo locale, allora \bar{x} è un punto di minimo locale su qualunque restrizione. Vale anche il viceversa.

Esempio 5.4 trova il minimo di $f(x) = (x_1 - 1)^2 + x_2^2$ dove $g_1(x) = x_1 - x_2^2$.
 $f(x)$ è una parabola con vertice in $(1, 0)$, quindi è coerciva. Dunque esiste minimo globale

\bar{x}	$\bar{\lambda}$	$f(\bar{x})$	Punto
$(0, 0)$	2	1	minimo locale o <u>sella</u>
$\left(\frac{1}{2}, \frac{\sqrt{2}}{2}\right)$	1	$\frac{3}{4}$	minimo globale perché coerciva
$\left(\frac{1}{2}, -\frac{\sqrt{2}}{2}\right)$	1	$\frac{3}{4}$	minimo globale perché coerciva

Per classificare il punto $(0, 0)$ mi restringo alla parabola r : $x_1 = x_2^2$



$$f(x) = (x_1 - 1)^2 + x_2^2 \Big|_{x_1=x_2^2} = (x_1 - 1)^2 + x_1 = x_1^2 - x_1 + 1 \stackrel{t:=x_1}{\Rightarrow} \phi(t) := t^2 - t + 1$$

$\phi(t)$ è una funzione di una variabile. $\bar{x} = (0, 0) \Rightarrow \bar{t} = 0$

Siccome $t := x_1$, allora $t = x_2^2 \geq 0$ sempre. Quindi devo studiare $\phi(t)$ in un intorno destro di $\bar{t} = 0$

$\bar{t} = 0$ non è un minimo locale, quindi $\bar{x} = (0, 0)$ è un punto di sella.

Esempio 5.5 trova il minimo di $f(x) = x_1 + x_2$ dove $g_1 = x_1^2 + x_2^2 - 2$, $h_1 = x_2^2 - x_1$

D è regolare per Mangasarian: si vede ad occhio che i gradienti dei vincoli attivi non sono linearmente dipendenti.

D è limitato, quindi esistono minimo e massimo globali.

I punti stazionari sono dati: $\left(\frac{1}{4}, -\frac{1}{2}\right)$, $(1, 1)$, $(1, -1)$.

\bar{x}	$(\bar{\lambda}, \bar{\mu})$	$f(\bar{x})$	Punto
$\left(\frac{1}{4}, -\frac{1}{2}\right)$	$(0, 1)$	$-\frac{1}{4}$	minimo globale
$(1, 1)$	$\left(-\frac{1}{2}, 0\right)$	2	massimo globale
$(1, -1)$	$\left(-\frac{1}{6}, \frac{2}{3}\right)$	0	<u>massimo locale</u> o sella. Attenzione: $\mu = 2/3 > 0$ non vuol dire che è sella.

Siccome c'è $h_1 = x_2^2 - x_1$, l'unica restrizione possibile è $r: x_2^2 = x_1$

$$f(x) = x_1 + x_2 \Big|_{x_2^2=x_1} = x_2^2 + x_2 \Rightarrow \phi(t) = t^2 + t$$

$$(x_1, x_2) = (1, -1) \Rightarrow \bar{t} = -1$$

Essendo $t := x_2$, allora $t^2 = x_1 \Rightarrow t = \pm \sqrt{x_1}$, questo non basta per capire in quale intorno studiare la funzione.

Dal vincolo $x_1 + x_2^2 \leq 2$:

$$x_1 + t^2 \leq 2 \Rightarrow t^2 \leq 2 - x_1 \Rightarrow -\sqrt{2-x_1} \leq t \leq \sqrt{2-x_1} \Big|_{x_1=1} \Rightarrow -1 \leq t \leq 1,$$

quindi devo studiare un intorno destro di $\bar{t} = -1$

La parabola $\phi(t) = t^2 + t$ è decrescente in un intorno destro di \bar{t} , quindi \bar{t} è massimo locale. Concludo che \bar{x} è massimo locale poiché ho ispezionato tutte le restrizioni.

6. Metodi per l'ottimizzazione libera

Siamo nel caso dell'ottimizzazione libera: $f: \mathbb{R}^n \rightarrow \mathbb{R}$, $\begin{cases} \max / \min f(x) \\ x \in \mathbb{R}^n \end{cases}$

6.1. Esistenza di massimi e minimi

Se il dominio non è compatto non è detto che esistano massimi o minimi. E' bene sapere che ci sono dei casi in cui esistono:

6.1.1. funzioni (anti)coercive

- Se una funzione è coerciva, allora esiste minimo
- Se una funzione è anticoerciva, allora esiste massimo

6.1.2. livelli inferiori e superiori

Proposizione.

Proposizione (livello inferiore e superiore):

$$\text{lev}_{\leq k} f(x) := \{x \in \mathbb{R}^n : f(x) \leq k\}, \quad \text{lev}_{\geq k} f(x) := \{x \in \mathbb{R}^n : f(x) \geq k\},$$

- Se $\text{lev}_{\leq k} f(x)$ compatti $\forall k$ allora esiste minimo,
- Se $\text{lev}_{\geq k} f(x)$ compatti $\forall k$ allora esiste massimo.

Geometricamente, $\text{lev}_{\leq k} f(x)$ sono i punti x per i quali la funzione non supera un certo k .

6.2. Successione iterativa e direzione di salita

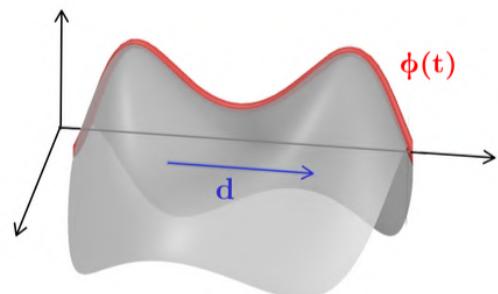
Tutti gli algoritmi che vedremo sono algoritmi iterativi che partono da un punto x^k e si spostano lungo una direzione d^k .

Successione iterativa:

$$\{x^k\} : x^{k+1} = x^k + t_k d^k, \quad t_k \geq 0$$

Dove $x^k + td^k$ è l'equazione parametrica della semiretta uscente da x^k con direzione d^k

$\phi(t) := f(x^k + td^k)$ è la restrizione di f lungo questa semiretta, ovvero il profilo di f tagliata lungo la direzione d^k



Ad esempio, nella PL la direzione d^k è W^h . La direzione migliora sempre la funzione obiettivo, e ci si ferma quando usciamo dal poliedro. Nell'ottimizzazione non lineare, invece, non è garantito che una direzione porti sempre a un miglioramento globale della funzione obiettivo. Ad esempio, osservando la figura sopra, si nota che la funzione f può salire e scendere lungo la direzione d .

Definizione.**Definizione (direzione di salita):** d^k è una direzione di salita se $\exists \bar{t} : f(x^k + td^k) > f(x^k) \forall t \in (0, \bar{t}]$ Ovvero, d^k è una direzione di salita se all'inizio $\phi(t)$ cresce.Noto che $\phi(t)$ è una funzione a una variabile. Dall'analisi 1, so che $\phi(t)$ cresce se $\phi'(t) > 0$.Chain rule: $\phi(t) = f(x^k + td^k) \Rightarrow \phi'(t) = \nabla f(x^k + td^k) \cdot \frac{d}{dt}td^k = \nabla f(x^k + td^k) \cdot d^k$.Voglio che $\phi(t)$ parta crescendo, quindi prendo $\phi'(0) = \nabla f(x^k) \cdot d^k > 0$ **Corollario.****Corollario:** $d^k : \nabla f(x^k) \cdot d^k > 0 \Rightarrow d^k$ è una direzione di salita.In particolare, posso prendere $d^k := \nabla f(x^k) \Rightarrow \nabla f(x^k) \cdot \nabla f(x^k) = |\nabla f(x^k)|^2$

- Se $\nabla f(x^k) \neq 0$, allora $|\nabla f(x^k)|^2 > 0$ e quindi ho trovato una direzione di salita,
- Se $\nabla f(x^k) = 0$, ho trovato un punto stazionario.

Quindi $d^k := \nabla f(x^k)$ è una direzione di salita.**6.3. Algoritmi iterativi per massimi**Supponiamo di star cercando $\max f(x)$. Mi piacerebbe molto che gli algoritmi arrivino all'ottimo in un numero finito di passi, come succedeva per la PL e PLI. Purtroppo però, questo non succede quasi mai.

- Gli algoritmi arrivano ad un punto stazionario e non all'ottimo.

Consolazione: se f è convessa (concava), allora il punto stazionario è min (max) globale

- Gli algoritmi non terminano mai.

Consolazione: utilizzo dei criteri di STOP.

- Mi fermo dopo un certo numero di passi: $x^{1000} \rightarrow \text{STOP}$,
- Mi fermo quando $|\nabla f(x^{1000})|$ è molto piccolo perché sono vicino ad un punto staz.,
- Mi fermo quando il gap $f(x^{1000}) - f(x^{999})$ è molto piccolo,
- Mi fermo dopo un certo tempo.

TEOREMA.**Teorema (di convergenza):**Sia $f: D \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$ una funzione che ammette massimo.La successione $\{x^k\} \subset \mathbb{R}^n$ ha punti di accumulazione, ognuno dei quali è stazionario.

I punti di accumulazione sono i punti tali che ogni intorno contiene infiniti punti. Un caso particolare è $\{x^k\}$ convergente, per cui c'è un solo punto di accumulazione.

6.3.1. Metodo del gradiente a passo fissato

Sia x_0 un punto di partenza arbitrario:

$$\begin{cases} d^k = \nabla f(x^k) \\ t^k = \eta \in \mathbb{R} \end{cases} \implies x^{k+1} = x^k + \eta \nabla f(x^k) \text{ dove } \eta \in \left(0, \frac{2}{L}\right)$$

Per il teorema di convergenza, esiste un punto di accumulazione $\bar{x} : \nabla f(\bar{x}) = 0 \forall x_0$

Commento	<p>Questo è il metodo più semplice, il più usato nel machine learning.</p> <p>Il difetto è che η non è arbitrario, ma dev'essere scelto nel range $\eta \in \left(0, \frac{2}{L}\right)$ dove L (costante di Lipschitz) è tale che $\frac{\partial^2 f}{\partial x_1 \partial x_2} \leq L$. La costante di Lipschitz è una stima superiore delle derivate seconde; il come mai lo sanno solo i matematici.</p> <p>L'ingegnere prova un η piccolo e, se la successione non ha punti di accumulazione, ne sceglie uno ancora più piccolo.</p>
-----------------	--

6.3.2. Metodo del gradiente a passo ideale

Sia x_0 un punto di partenza arbitrario:

$$\begin{cases} d^k = \nabla f(x^k) \\ t^k \in \arg \max_{t \geq 0} f(x^k + td^k) \end{cases} \implies x^{k+1} = x^k + t_k \nabla f(x^k)$$

Per il teorema di convergenza, esiste un punto di accumulazione $\bar{x} : \nabla f(\bar{x}) = 0 \forall x_0$

Commento	<p>Il metodo più classico: utilizzo il gradiente come direzione di salita e cerco il passo ottimale. Lo svantaggio è che devo trovare un massimo di una funzione ad una sola variabile. Vantaggioso se, per esempio, f è una forma quadratica, per cui $\phi(t) := f(x^k + td^k)$ è una parabola ed è facile trovare il massimo.</p>
-----------------	--

6.3.3. Metodo di Newton

Sia x_0 un punto di partenza "vicino" al punto di massimo:

$$\begin{cases} d^k = -[Hf(x^k)]^{-1} \nabla f(x^k) \\ t^k = 1 \end{cases} \implies x^{k+1} = x^k - [Hf(x^k)]^{-1} \nabla f(x^k)$$

Dove dove $Hf(x^k)$ è una matrice **definita negativa**. Per il teorema di convergenza, esiste un punto di accumulazione $\bar{x} : \nabla f(\bar{x}) = 0$, se x_0 è un punto "vicino" al punto di massimo.

Commento Il vantaggio è che converge in poche iterazioni. Lo svantaggio è che devo calcolare una matrice inversa ad ogni passo, e la convergenza dipende dal passo iniziale.
L'ingegnere prova un x_0 . Se vede che non converge, ne prova un altro.

Osservazione:

$$d^k := -[Hf(x^k)]^{-1} \nabla f(x^k) \text{ è di salita?}$$

Siccome so che $\nabla f(x^k)$ è di salita, allora perché anche d^k sia di salita deve succedere che

$$\langle -[Hf(x^k)]^{-1} \nabla f(x^k), \nabla f(x^k) \rangle > 0$$

Ricordiamo che per definizione, una matrice A è definita positiva se $\langle Ay, y \rangle > 0$, quindi mi basta vedere se $-[Hf(x^k)]^{-1}$ è definita positiva. Siccome $Hf(x^k)$ è definita negativa per ipotesi, allora $-[Hf(x^k)]^{-1}$ è definita positiva. Dunque d^k è di salita.

6.3.4. Metodo di Armijo-Goldstein-Wolfe

$$\begin{aligned} \text{Condizioni di Armijo-Goldstein-Wolfe: si cerca } t_k \text{ tale che} \\ \left\{ \begin{array}{l} \phi(t_k) \geq \phi(0) + \alpha \phi'(0)t_k \\ \phi'(t_k) \leq \beta \phi'(0) \\ 0 < \alpha < \beta < 1 \end{array} \right. \end{aligned}$$

La prima condizione impone l'aumento della funzione obiettivo, la seconda condizione garantisce che l'aumento sia consistente. E' garantito che esiste un intervallo (A, B) tale che le condizioni di AGW sono rispettate.

Per ricercare t_k che rispetti le condizioni di AGW, si può usare una tecnica di tipo backtracking: si parte con un valore $t_k = \bar{t}_k > 0$ e finché la seconda condizione di AGW non è soddisfatta, si moltiplica per un fattore $\gamma \in (0, 1)$. Questa procedura garantisce che il punto trovato soddisfi la condizione di salita per ϕ , ma assicura che il passo non sia troppo grande.

Esempio 6.1 Fai un passo del metodo del gradiente a passo ideale data la funzione $f(x) = x_1^2 + x_2^2 - x_1 x_2 + 3x_1$ dato $x^k = (2, 1)$

$$\nabla f(x) = (2x_1, -x_2 + 3, 2x_2 - x_1)$$

$$d^k = \nabla f(x^k) = (6, 0)$$

$$r : (2, 1) - t(6, 0) = (2 + 6t, 1)$$

$$\phi(t) := (2 - 6t)^2 + 1 - (2 - 6t) + 3(2 - 6t) = 36t^2 - 36t + 9$$

$$t_k \in \arg \min_{t \geq 0} \phi(t), \quad \phi'(t) = 72t - 36, \quad \phi'(\bar{t}) = 0 \iff \bar{t} = \frac{1}{2}$$

$$\text{Quindi } t_k = \frac{1}{2} \quad x^{k+1} = (2, 1) - \frac{1}{2}(6, 0) = (-1, 1)$$

ALGORITMO.

Algoritmi per **massimi** e **minimi** liberi

Metodo del gradiente a passo costante:

1. Scelgo $x_0 \in \mathbb{R}^n$, scelgo $\eta \in \left(0, \frac{2}{L}\right)$ e pongo $k := 0$
2. **while** $\nabla f(x^k) \neq 0$ **do**
 - se cerco il massimo: $x^{k+1} := x^k + \eta \nabla f(x^k)$, incremento k .
 - se cerco il minimo: $x^{k+1} := x^k - \eta \nabla f(x^k)$, incremento k .

Metodo del gradiente a passo ideale:

1. Scelgo $x_0 \in \mathbb{R}^n$ e pongo $k := 0$
2. **while** $\nabla f(x^k) \neq 0$ **do**
 - se cerco massimo calcolo una sol. t_k del problema $\max_{t \geq 0} f(x_k + t \nabla f(x^k))$
 $x^{k+1} := x^k + t_k \nabla f(x^k)$, incremento k .
 - se cerco minimo calcolo una sol. t_k del problema $\min_{t \geq 0} f(x_k - t \nabla f(x^k))$
 $x^{k+1} := x^k - t_k \nabla f(x^k)$, incremento k .

Metodo di Newton:

1. Scelgo un punto $x_0 \in \mathbb{R}^n$ "vicino" al punto di massimo e pongo $k := 0$
2. **while** $\nabla f(x^k) \neq 0$ **do**
 - se cerco massimo $d^k := -[Hf(x^k)]^{-1} \nabla f(x^k)$ dove $Hf(x^k) < 0$
 $x^{k+1} := x^k + d^k$, incremento k .
 - se cerco minimo $d^k := -[Hf(x^k)]^{-1} \nabla f(x^k)$ dove $Hf(x^k) > 0$
 $x^{k+1} := x^k + d^k$, incremento k .

Metodo di Armijo-Goldstein-Wolfe:

Per trovare t_k tale che $\begin{cases} \phi(t_k) \geq \phi(0) + \alpha \phi'(0)t_k \\ \phi'(t_k) \leq \beta \phi'(0) \\ 0 < \alpha < \beta < 1 \end{cases}$ per i massimi

Per trovare t_k tale che $\begin{cases} \phi(t_k) \leq \phi(0) + \alpha \phi'(0)t_k \\ \phi'(t_k) \geq \beta \phi'(0) \\ 0 < \alpha < \beta < 1 \end{cases}$ per i minimi

1. Inizio con $t_k > 0$
2. Ridimensiono $t_k \leftarrow \gamma t_k$ con $\gamma \in (0, 1)$ finché non soddisfo le condizioni di AGW

- Aggiorno per il massimo $x^{k+1} := x^k + t_k \nabla f(x^k)$, incremento k
- Aggiorno per il minimo $x^{k+1} := x^k - t_k \nabla f(x^k)$, incremento k

Esempio 6.2 Fai un passo dei quattro metodi data la funzione
 $f(x) = -2x_1^2 - 5x_2^2 + x_1x_2 - x_1$ dato $x^k = (0, 0)$

$$\nabla f(x) = (-4x_1 + x_2 - 1, -10x_2 + x_1)$$

Gradiente a passo costante:

$$\eta < 2 / L = 2 / 10 = 1 / 5$$

$$\nabla f(0, 0) = (-1, 0) \implies x^{k+1} = (0, 0) + \frac{1}{5}(-1, 0) = \left(-\frac{1}{5}, 0\right)$$

$$f(x^k) = 0, \quad f(x^{k+1}) = 3 / 25$$

Gradiente a passo ideale:

$$r : (0, 0) + t(-1, 0) = (-t, 0)$$

$$\phi(t) := -2(-t)^2 + t = -2t^2 + t, \quad \phi(t) = 0 \iff t = \frac{1}{4} \implies t_k = \frac{1}{4}$$

$$x^{k+1} = (0, 0) + \frac{1}{4}(-1, 0) = \left(-\frac{1}{4}, 0\right), \quad f(x^k) = 0, \quad f(x^{k+1}) = 1 / 8$$

Metodo di Newton:

$$-Hf^{-1} = -\begin{bmatrix} -10/39 & -1/39 \\ -1/39 & -4/39 \end{bmatrix}$$

$$x^{k+1} = (0, 0) - Hf^{-1} \cdot \begin{bmatrix} -1 \\ 0 \end{bmatrix} = (0, 0) - \left(\frac{10}{39}, \frac{1}{39}\right) \implies f(x^{k+1}) = \frac{5}{39}$$

Metodo di Armijo Goldstein Wolfe:

$$\nabla f(x) = (-4x_1 + x_2 - 1, -10x_2 + x_1), \quad \nabla f(0, 0) = (-1, 0)$$

$$d^k = \nabla f(x^k) = (-1, 0)$$

$$\begin{aligned} \phi(t) &:= f(x^k + td^k) = f((0, 0) + t(-1, 0)) = f(-t, 0) \\ &= -2(-t)^2 - 5(0)^2 + (-t)(0) - (-t) = -2t^2 + t \end{aligned}$$

$$\phi'(t) = -4t + 1$$

1° condizione di AGW:

$$\phi(t_k) \geq \phi(0) + \alpha\phi'(0)t_k, \text{ prendo } \alpha := \frac{1}{4}$$

$$\text{Siccome } \phi(0) = 0, \quad \phi'(0) = 1, \text{ allora } \phi(t_k) \geq 0 + \frac{1}{4}t_k \iff -2t_k^2 + \frac{3}{4}t_k \geq 0$$

$$\text{ovvero } -8t_k^2 + 3t_k \geq 0, \text{ quindi } t_k \in \left[0, \frac{3}{8}\right]$$

2° condizione di AGW:

$$\phi'(t_k) \leq \beta\phi'(0), \text{ prendo } \beta := \frac{1}{2}$$

$$-4t_k + 1 \leq \frac{1}{2} \iff -4t_k - \frac{1}{2} \leq 0 \iff -8t_k + 1 \leq 0 \implies t_k \in \left[\frac{1}{8}, +\infty \right)$$

Dunque il famoso intervallo $(A, B) = \left(\frac{1}{8}, \frac{3}{8} \right)$, procedo con un valore iniziale t_k e, se non

verifica le condizioni, procedo con il backtracking. Per fortuna, $t_k := \frac{1}{4}$ va bene.

$$x^{k+1} = (0, 0) + \frac{1}{4}(-1, 0) = \left(-\frac{1}{4}, 0 \right), \quad f(x^{k+1}) = \frac{1}{8}$$

7. Metodi per l'ottimizzazione vincolata

Vale ancora il teorema di convergenza: se $f: D \rightarrow \mathbb{R}$ ammette massimo, allora i punti di accumulazione di $\{x^k\}$ sono punti stazionari. Tuttavia, nell'ottimizzazione vincolata, i punti stazionari non si limitano a quelli che annullano il gradiente della funzione obiettivo (come nell'ottimizzazione libera), ma sono quelli che risolvono LKKT.

Adesso x^{k+1} non solo dev'essere migliore di x^k , ma deve essere anche ammissibile. Per semplicità, assumiamo che il dominio sia un poliedro limitato.

Vedremo due algoritmi che convergono a un punto stazionario del problema $\begin{cases} \min / \max f(x) \\ x \in P \text{ limitato} \end{cases}$

7.1. Metodo di Frank-Wolfe

Sia il problema $P \begin{cases} \max f(x) \\ x \in P \end{cases}$

Il problema linearizzato è definito $PL(x^k) \begin{cases} \max_{x \in P} \nabla f(x^k)^T \cdot x \end{cases}$

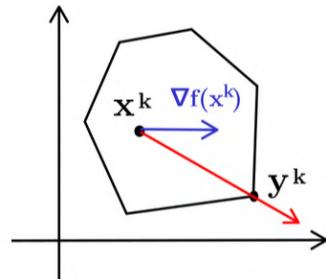
Il problema linearizzato è definito $PL(x^k) \begin{cases} \max_{x \in P} \nabla f(x^k)^T \cdot x \end{cases}$

L'idea fondamentale dietro al metodo di Frank-Wolfe è che il problema lineare è molto più facile da risolvere. Consideriamo infatti lo sviluppo di Taylor del primo ordine di $f(x)$ attorno al punto x^k : $f(x) \approx f(x^k) + \nabla f(x^k)^T (x - x^k) = f(x^k) + \nabla f(x^k)^T x + \nabla f(x^k)^T x^k$. Se voglio massimizzare $f(x)$, posso togliere le costanti dallo sviluppo di Taylor, ovvero devo massimizzare $\nabla f(x^k)^T x$. Questo termine, rappresenta un prodotto scalare lineare tra il gradiente calcolato in x^k e il vettore x .

Indichiamo con y^k il vertice ottimo del problema $PL(x^k)$.

La parametrizzazione del segmento che congiunge x^k e y^k è $(1-t)x^k + ty^k$, $t \in [0, 1]$.

Questa è uguale a $x^k - tx^k + ty^k = x^k + t(y^k - x^k)$,
dove $d^k := y^k - x^k$ è la direzione che va da x^k a y^k .



Il passo t_k non deve uscire dal poliedro, quindi $t^k \in \arg \max_{t \in [0,1]} f(x^k + t(y^k - x^k))$.

In alternativa, posso trovare t_k con il metodo di Armijo-Goldstein-Wolfe, o con un passo costante, a patto che x^{k+1} sia ammissibile.

Si pone $x^{k+1} := x^k + t^k d^k = x^k + t^k (y^k - x^k)$. Per il teorema di convergenza, l'algoritmo di Frank-Wolfe converge ad un punto stazionario.

ALGORITMO.

Metodo di Frank-Wolfe per massimi

1. Scelgo $x_0 \in P$ e pongo $k := 0$
2. Calcolo una soluzione y^k del problema linearizzato $\begin{cases} \max \nabla f(x^k)^T x \\ x \in P \end{cases}$
3. if $\nabla f(x^k)^T (y^k - x^k) = 0$ then STOP. $d^k = 0$ e x^k è un punto stazionario.
else calcolo una soluzione t^k del problema $\max_{t \in [0,1]} f(x^k + t(y^k - x^k))$
2. Pongo $x^{k+1} := x^k + t^k (y^k - x^k)$, incremento k e torno al passo 2.

Metodo di Frank-Wolfe per minimi

1. Scelgo $x_0 \in P$ e pongo $k := 0$
2. Calcolo una soluzione y^k del problema linearizzato $\begin{cases} \min \nabla f(x^k)^T x \\ x \in P \end{cases}$
3. if $\nabla f(x^k)^T (y^k - x^k) = 0$ then STOP. $d^k = 0$ e x^k è un punto stazionario.
else calcolo una soluzione t^k del problema $\min_{t \in [0,1]} f(x^k + t(y^k - x^k))$
2. Pongo $x^{k+1} := x^k + t^k (y^k - x^k)$, incremento k e torno al passo 2.

Esempio 7.1 Fare un passo dell'algoritmo FW data la seguente funzione e il seguente poliedro:

$$\max f(x) = 2x_1^2 + 4x_2^2 + 4x_1 - x_2, \quad V = \{(0,1), (4,1), (4,3), (-1,5)\}, \text{ dato } x^k = \left(4, \frac{5}{3}\right)$$

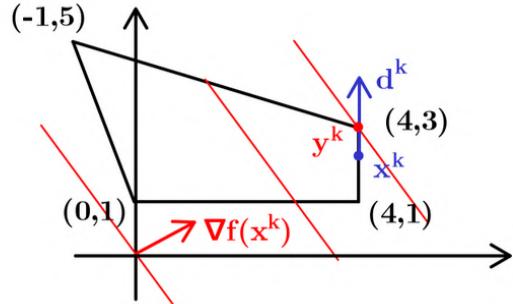
$$\nabla f(x) = (4x_1 + 4, 8x_2 - 1), \quad \nabla f(x^k) = \left(20, \frac{37}{3}\right) \approx (2, 1.2) \text{ per vedere la direzione.}$$

Il problema linearizzato

$$PL(x^k) \begin{cases} \max \nabla f(x^k) \cdot x \\ x \in P \end{cases}$$

si risolve geometricamente con le linee di isoguadagno. $y^k = (4, 3)$.

$$d^k = y^k - x^k = (4, 3) - \left(4, \frac{5}{3}\right) = \left(0, \frac{4}{3}\right)$$



$$t_k \in \arg \max_{t \in [0,1]} f(x^k + td^k) = \arg \max_{t \in [0,1]} \phi(t)$$

$$f\left(\left(4, \frac{5}{3}\right) + t\left(0, \frac{4}{3}\right)\right) = f\left(4, \frac{5}{3} + \frac{4}{3}t\right) = 2 \cdot 16 + 4\left(\frac{5}{3} + \frac{4}{3}t\right)^2 + 4 \cdot 4\left(\frac{5}{3} + \frac{4}{3}t\right) = \phi(t)$$

Osservo che se $\phi(t) = at^2 + bt + c$, posso eliminare la costante c perché sto cercando il punto di massimo, e non il valore massimo.

$$\phi(t) \equiv \frac{160}{9}t + \frac{64}{9}t^2 - \frac{4}{3}t = \frac{64}{9}t^2 - \frac{148}{9}t$$

$\phi(t)$ è monotona crescente, quindi $\max_{t \in [0,1]} \phi(t) \iff t = 1$

$$\text{Quindi } x^{k+1} = x^k + t_k d^k = \left(4, \frac{5}{3}\right) + \left(0, \frac{4}{3}\right) = (4, 3).$$

➤ Il punto $(4, 3)$ è di massimo globale?

Richiamo il Teorema 3, che dice che funzioni convesse su poliedri limitati hanno massimo globale in uno dei vertici. Provo tutti i vertici: $f(4, 3) = 81$, $f(-1, 5) = 93$, $f(0, 1) = 3$, $f(5, 1) = 51$. Dunque $(-1, 5)$ è minimo globale.

➤ Fai un altro passo dell'algoritmo FW

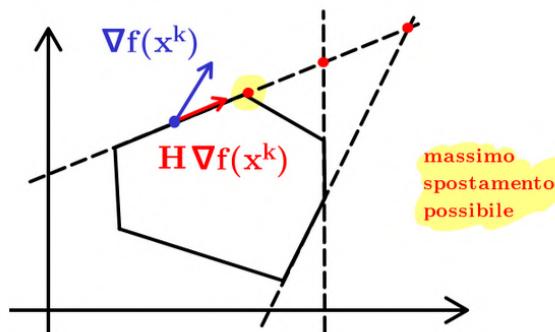
$$\nabla f(4, 3) = (20, 23), \quad y^k = (3, 4), \quad d^k = y^k - x^k = (0, 0)$$

Siccome $d^k = \vec{0}$, l'algoritmo termina con un punto stazionario i.e. $(4, 3)$ risolve LKKT.

7.2. Metodo del gradiente proiettato

Come abbiamo visto per l'ottimizzazione libera, se il punto x^k è interno alla regione ammissibile, si può scegliere la direzione $d^k = \nabla f(x^k)$ (metodo del gradiente).

Se invece x^k è sulla frontiera, si sceglie d^k uguale alla proiezione ortogonale di $\nabla f(x^k)$ sul sottospazio vettoriale definito dai vincoli attivi nel punto x^k .



Si osserva che d^k è una direzione di salita perché l'angolo compreso tra $\nabla f(x^k)$ e la sua proiezione è sempre minore di $\pi / 2$, quindi $\nabla f(x^k) \cdot d^k > 0$

Lemma.

Lemma (matrice della proiezione ortogonale):

Sia $\mathcal{A}(x^k) := \{i \in I : A_i x^k = b_i\}$ il sottospazio degli indici attivi

Sia M la sottomatrice di A formata dagli indici attivi (righe).

La proiezione ortogonale di un vettore y su M è $H \cdot y$

dove $H := \begin{cases} Id, & \bar{y} \text{ non ha vincoli attivi} \\ Id - M^T (MM^T)^{-1} M, & \text{altrimenti} \end{cases}$

Quindi la direzione è $d^k := H \nabla f(x^k)$.

Per rimanere dentro al poliedro, calcolo il massimo spostamento possibile

$$\hat{t}_k = \max_t \begin{cases} t \\ A(x^k + td^k) \leq b \end{cases}, \text{ devo risolvere disequazioni lineari in } t \text{ (semplice).}$$

\hat{t}_k non è il passo, ma il massimo spostamento possibile, ovvero per quando posso scorrere lungo lo spigolo prima di uscire dal poliedro. Allora lo spostamento è $t^k \in \arg \max_{t \in [0, \hat{t}_k]} f(x^k + td^k)$.

Cosa succede quando $d^k = 0$?

Se $d^k = 0$, sia $\lambda := -(MM^T)^{-1} M \nabla f(x^k)$.

- Se $\lambda \leq 0$ allora x^k risolve LKKT e sono arrivato a un punto stazionario,
- Altrimenti, sia $j : \lambda_j = \max_{j=1,\dots,k} \lambda_j > 0$, devo eliminare la riga j da M e ricalcolare H .

ALGORITMO.

Metodo del gradiente proiettato per **massimi**

1. Sia $\mathcal{A} = \{i : A_i x^k = b_i\}$ e M la sottomatrice di A avente per righe i vettori A_i con $i \in \mathcal{A}$
2. Calcolo $H := Id - M^T (MM^T)^{-1} M$ e la direzione $d^k := H \nabla f(x^k)$
3. **if** $d^k \neq 0$
 - then** calcolo una soluzione \hat{t}_k del problema $\begin{cases} \max t \\ A(x^k + td^k) \leq b \end{cases}$
 - calcolo una soluzione t_k del problema $\max_{t \in [0, \hat{t}_k]} f(x^k + td^k)$
 - aggiorno $x^{k+1} := x^k + t_k d^k$, incremento k e torno al passo 1.
- else** $d^k = 0$ calcolo $\lambda := -(MM^T)^{-1} M \nabla f(x^k)$
- if** $\lambda \leq 0$ **then** STOP (x^k è stazionario).
- else** calcolo $\lambda_j := \max_{i \in \mathcal{A}} \lambda_i$, elimino da M la riga A_j e torno al passo 2.

Metodo del gradiente proiettato per **minimi**

1. Sia $\mathcal{A} = \{i : A_i x^k = b_i\}$ e M la sottomatrice di A avente per righe i vettori A_i con $i \in \mathcal{A}$
2. Calcolo $H := Id - M^T (MM^T)^{-1} M$ e la direzione $d^k := H(-\nabla f(x^k))$
3. **if** $d^k \neq 0$
 - then** calcolo una soluzione \hat{t}_k del problema $\begin{cases} \max t \\ A(x^k + td^k) \leq b \end{cases}$
 - calcolo una soluzione t_k del problema $\min_{t \in [0, \hat{t}_k]} f(x^k + td^k)$
 - aggiorno $x^{k+1} := x^k + t_k d^k$, incremento k e torno al passo 1.
- else** $d^k = 0$ calcolo $\lambda := -(MM^T)^{-1} M \nabla f(x^k)$
- if** $\lambda \geq 0$ **then** STOP (x^k è stazionario).
- else** calcolo $\lambda_j := \min_{i \in \mathcal{A}} \lambda_i$, elimino da M la riga A_j e torno al passo 2.

Esempio 7.2 Fare un passo del gradiente proiettato dato il punto $x^k = \left(-\frac{10}{3}, -\frac{2}{3} \right)$ la funzione $f(x) = 2x_1^2 + 4x_2^2 - 2x_1 - 9x_2$, dati i vertici $V = \{(-3, 0), (4, 1), (5, -3), (-4, -2)\}$

Determino il poliedro in forma algebrica:

Retta passante per 2 punti: $\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1}$

$$A = (-3, 0), B = (4, 1), C = (5, -3), D = (-4, -2)$$

- Retta passante per \overline{AB} : $\frac{x+3}{4+3} = \frac{y}{1} \Rightarrow \frac{x+3}{7} = y \Rightarrow x - 7y = -3$

provando il punto $(0, 0)$, scopro che $0 \geq -3$, quindi la disequazione è $-x_1 + 7x_2 \leq 3$

- Retta passante per \overline{BC} : $\frac{x-4}{5-4} = \frac{y-1}{-3-1} \Rightarrow 4x - 16 = 1 - y \Rightarrow 4x + y = 17$

provando il punto $(0, 0)$, scopro che $0 \leq 17$, quindi $4x_1 + x_2 \leq 17$

- Retta passante per \overline{CD} : $\frac{x-5}{-4-5} = \frac{y+3}{-2+3} \Rightarrow x - 5 = -9y - 27 \Rightarrow x + 9y = -22$

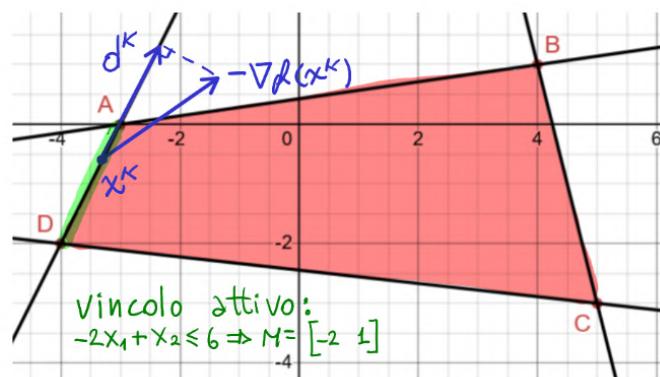
provando il punto $(0, 0)$, scopro che $0 \geq -22$, quindi $-x_1 - 9x_2 \leq 22$

- Retta passante per \overline{DA} : $\frac{x+4}{-3+4} = \frac{y+2}{2} \Rightarrow 2x + 8 = y + 2 \Rightarrow 2x - y = -6$

provando il punto $(0, 0)$, scopro che $0 \geq -6$, quindi $-2x_1 + x_2 \leq 6$

Determino il gradiente:

$$\nabla f(x) = (4x_1 - 2, 8x_2 - 9), \quad \nabla f(x^k) = \left(-\frac{46}{3}, -\frac{43}{3} \right)$$



Determino le matrici M e H :

$$M = (2, 1)$$

$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} -2 \\ 1 \end{bmatrix} \left((-2, 1) \begin{bmatrix} -2 \\ 1 \end{bmatrix} \right)^{-1} (-2, 1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \frac{1}{5} \begin{bmatrix} -2 \\ 1 \end{bmatrix} (-2, 1) = \\ = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \frac{1}{5} \begin{bmatrix} 4 & -2 \\ -2 & 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{5} & \frac{2}{5} \\ \frac{2}{5} & \frac{4}{5} \end{bmatrix}$$

Determino la direzione:

$$d^k := H \begin{bmatrix} \frac{46}{3} \\ \frac{43}{3} \\ \frac{3}{3} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{46}{3} \\ \frac{43}{3} \\ \frac{3}{3} \end{bmatrix} = \begin{bmatrix} \frac{132}{15} \\ \frac{15}{264} \\ \frac{15}{15} \end{bmatrix} \equiv \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$$

La direzione è la stessa, cambia il modulo. Quindi cambia anche t_k , ma il prodotto $t_k d^k$ rimane sempre costante.

Determino il passo massimo \hat{t}_k :

$$x^k + td^k = \left(-\frac{10}{3}, -\frac{2}{3} \right) + t(1, 2) = \left(-\frac{10}{3} + t, -\frac{2}{3} + 2t \right)$$

Il vincolo per il quale la direzione esce dal poliedro è $-x_1 + 7x_2 \leq 3$ (retta passante per \overline{AB})

$$\begin{aligned} \text{Quindi } & \left(-\frac{10}{3} + t \right) + 7\left(-\frac{2}{3} + 2t \right) \leq 3 \Rightarrow \frac{10}{3} - t - \frac{14}{3} + 14t \leq 3 \Rightarrow -\frac{4}{3} + 13t \leq 3 \\ \Rightarrow & 13t \leq \frac{4+9}{3} \Rightarrow t \leq \frac{13}{3 \cdot 13} \Rightarrow t \leq \frac{1}{3} \Rightarrow \hat{t}_k = \frac{1}{3} \end{aligned}$$

Determino il passo t_k :

$$\begin{aligned} \phi(t) = 2\left(-\frac{10}{3} + t \right)^2 + 4\left(-\frac{2}{3} + 2t \right)^2 = & -2\left(-\frac{10}{3} + t \right) - 9\left(-\frac{2}{3} + 2t \right) = 2t^2 + 16t^2 - \frac{40}{3}t + \\ & + \frac{32}{3}t - 2t - 18t = 18t^2 - 44t + \dots \end{aligned}$$

Il termine di grado zero non mi interessa perché tanto ci faccio la derivata.

Anzi, siccome ϕ è monotona decrescente in $\left[0, \frac{1}{3} \right]$, allora $t_k = \frac{1}{3}$

Determino il vertice successivo:

$$x^{k+1} = \left(-\frac{10}{3} + \frac{1}{3}, -\frac{2}{3} + \frac{2}{3} \right) = (-3, 0)$$

Alla fine dell'ultima lezione, il giorno 11 dicembre 2024, gli studenti ormai ridotti allo stremo, decidono di tributare il loro omaggio all'epopea della ricerca operativa. 92 minuti di applausi! Sullo sfondo, un'aula semi-deserta e il prof. Pappalardo che, ormai commosso, si inchina. Leggenda vuole che qualcuno stia ancora applaudendo...