

Parte IV:

Matrici totalmente unimodulari

Formulazioni

Consideriamo il seguente problema di Knapsack 0-1

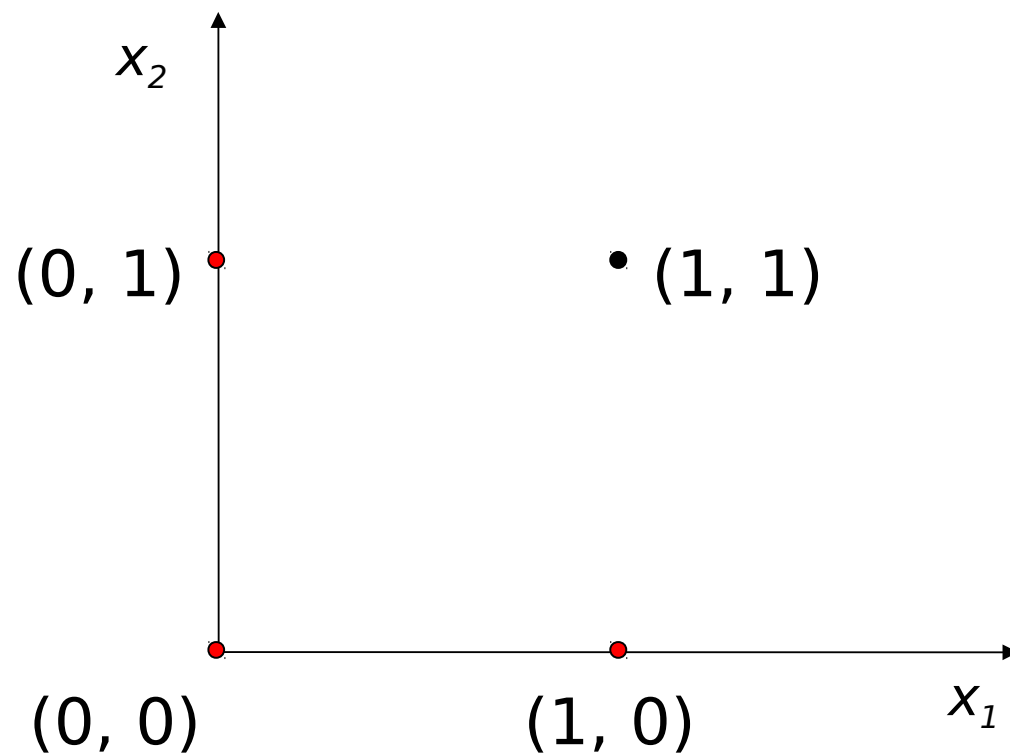
$$\begin{aligned} \max \quad & (5x_1 + 2x_2) \\ & 3x_1 + 4x_2 \leq 6 \\ & x \in \{0,1\}^2 \end{aligned}$$

Insiemi ammissibili

$$F = \{(0, 0), (0, 1), (1, 0)\}$$

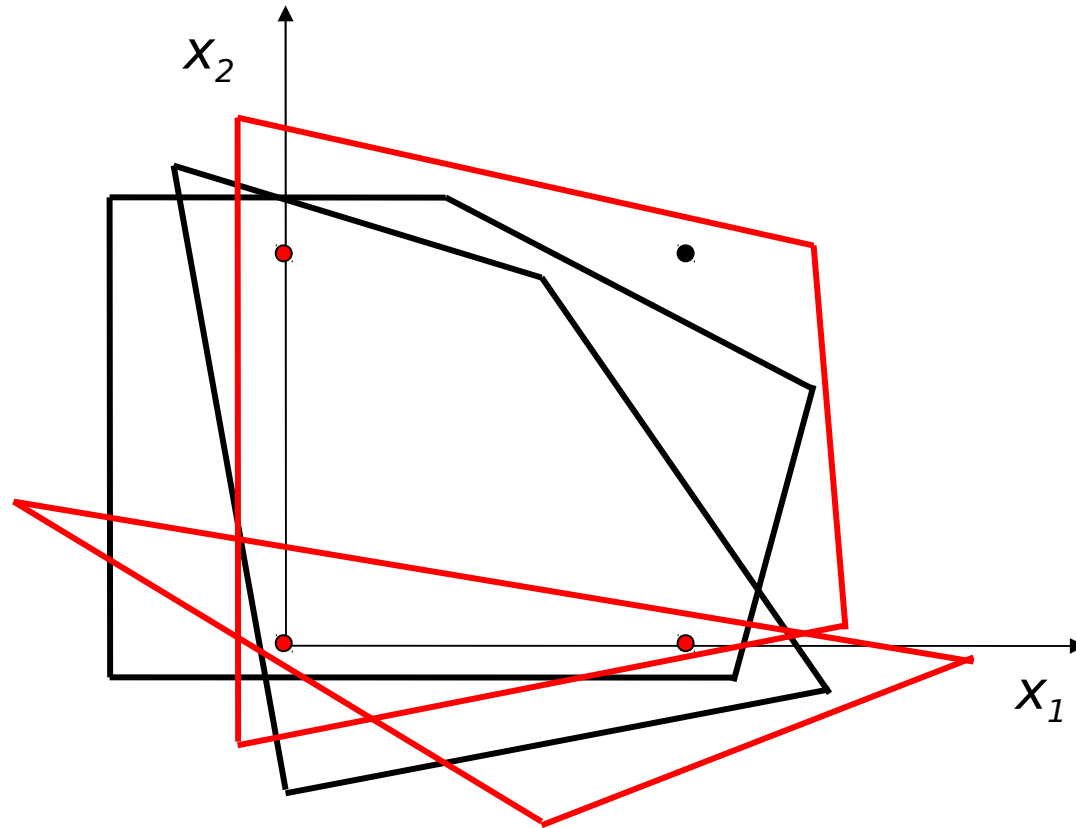
Rappresentiamo sul piano gli insiemi ammissibili.

Insiemi ammissibili



Formulazione

Un poliedro P è una formulazione di un problema di OC 0-1 se e solo se $P \cap \{0,1\}^n = F$



Il rilassamento lineare ...

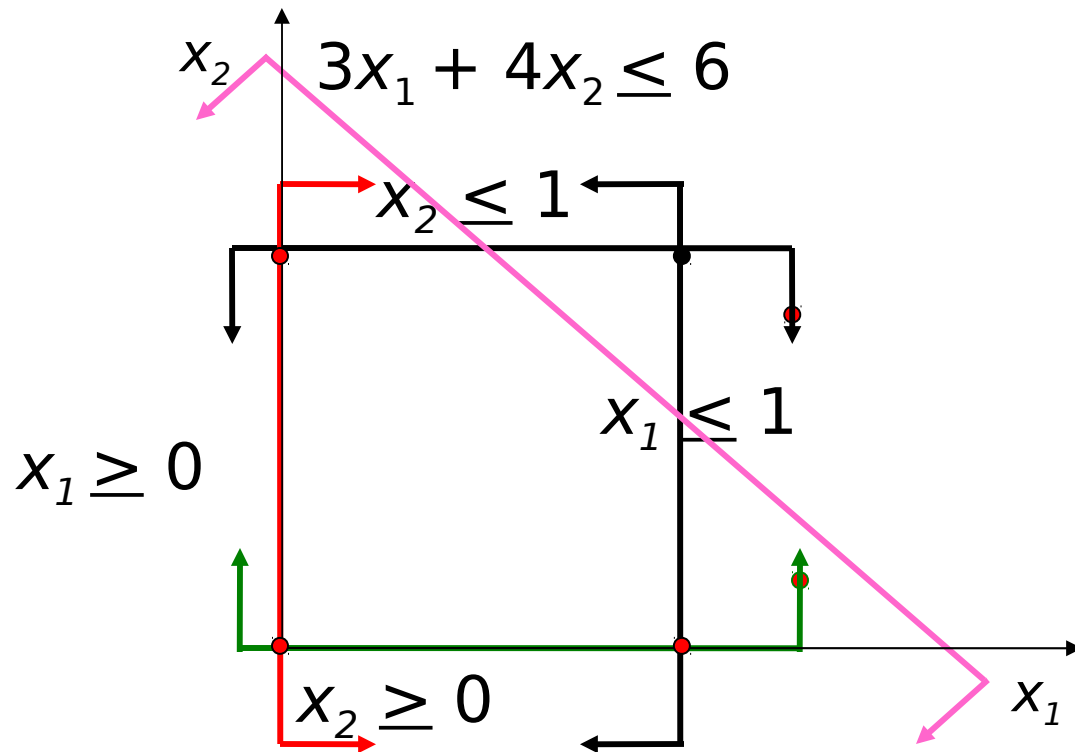
Il problema di knapsack 0-1

$$\begin{aligned} \max \quad & 5x_1 + 2x_2 \\ & 3x_1 + 4x_2 \leq 6 \\ & x \in \{0,1\}^2 \end{aligned}$$

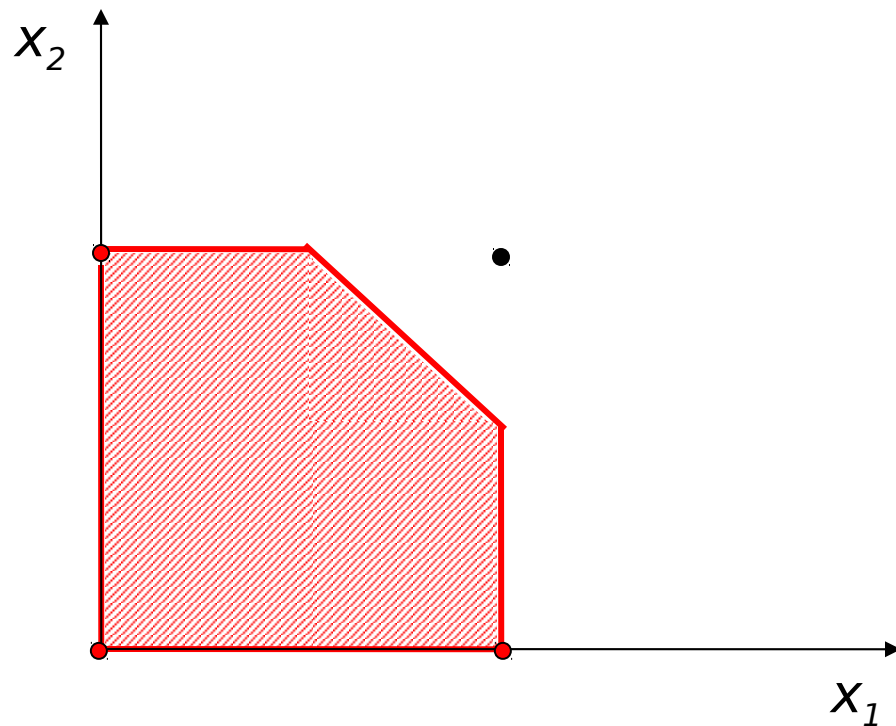
ha come rilassamento lineare

$$\begin{aligned} \max \quad & 5x_1 + 2x_2 \\ & 3x_1 + 4x_2 \leq 6 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \\ & x_1 \leq 1 \\ & x_2 \leq 1 \end{aligned}$$

... è un poliedro ...



... ovvero, è una formulazione



Gerarchia di formulazioni

Quando una formulazione è “migliore” di un'altra?

Definizione: Se un poliedro P_1 , formulazione di F , è contenuto in P_2 , formulazione di F , diciamo che P_1 è migliore di P_2 .

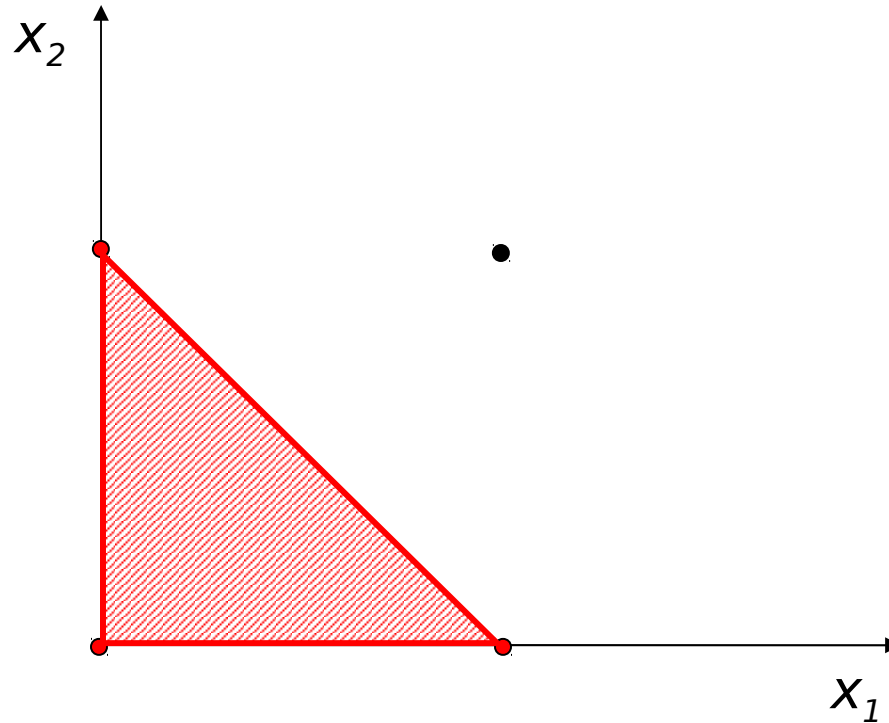
In generale

$$P_1 \subseteq P_2 \subseteq P_3 \dots$$

Esiste una formulazione “ideale”?

Formulazione ideale

“Geometricamente” la formulazione ideale coincide con il più piccolo poliedro contenente F .



Come si ottiene la formulazione ideale?

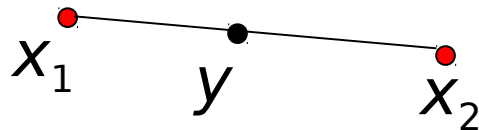
Proprietà

Osservazione: Ogni vertice del poliedro associato alla “formulazione ideale” è in corrispondenza biunivoca con un insieme ammissibile.

Definizione: Dati due vettori x_1 e x_2 di \mathbb{R}^n si definisce combinazione convessa il vettore

$$y = \lambda x_1 + (1 - \lambda) x_2 \quad \text{con } \lambda \in [0, 1]$$

Esempio:



Involucro convesso

Definizione: Dato un insieme $X \subseteq R^n$, l'*involucro convesso* di X , indicato con $\text{conv}(X)$, è definito nel modo seguente:

$$\text{conv}(X) = \{x: x = \sum_{i=1}^t \lambda_i x^i, \sum_{i=1}^t \lambda_i = 1, \lambda_i \geq 0 \text{ per } i=1, \dots, t \text{ su tutti i sottoinsiemi finiti } \{x^1, \dots, x^t\} \text{ di } X\}.$$

Pertanto, l'involucro convesso è l'insieme di tutte le possibili combinazioni convesse di un insieme di vettori X di R^n .

Osservazione: $\text{conv}(X)$ è un poliedro.

La formulazione ideale di F è $\text{conv}(F)$.

Calcolo di conv (F)

In linea di principio ...

Dati gli insiemi ammissibili

$$F = \{(0, 0), (0, 1), (1, 0)\}$$

$y \in \text{conv}(F)$ se e solo se si può esprimere come

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \lambda_1 \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \lambda_2 \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \lambda_3 \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$\lambda_1 + \lambda_2 + \lambda_3 = 1$$

$$\lambda_1, \lambda_2, \lambda_3 \geq 0$$

... a questo punto

Attenzione: questo sistema è nello spazio R^{n+m} , se m sono gli insiemi ammissibili.

Quindi per ottenere la formulazione ideale è necessario “proiettare” il sistema nello spazio R^n .

A questo scopo è possibile utilizzare l’algoritmo di Fourier-Motzkin che consente di “eliminare” le variabili λ .

Punto della situazione

Dato un problema di OC

1. Elenco tutti gli insiemi ammissibili.
2. Rappresento gli insiemi ammissibili come vettori a componenti in $\{0,1\}$.
3. Scrivo l'involucro convesso applicando la definizione.
4. Con l'algoritmo di Fourier-Motzkin elimino i coefficienti della combinazione convessa e ottengo una formulazione ideale nello spazio R^n .
5. Applico il metodo del simplesso e trovo la soluzione ottima.

È efficiente questo algoritmo?

Efficienza del calcolo di $\text{conv}(F)$

Problemi del precedente algoritmo

1. Gli insiemi ammissibili sono tipicamente in numero esponenziale.
2. L'algoritmo di Fourier-Motzkin non ha complessità polinomiale.

Però sappiamo che una formulazione ideale esiste sempre e quindi

1. Caso MOLTO fortunato: abbiamo una formulazione che è proprio la formulazione ideale.
2. Tentiamo di approssimare la formulazione ideale costruendo una gerarchia di formulazioni a partire da una formulazione iniziale.

Gerarchia di formulazioni

Abbiamo già visto che:

Se un poliedro P_1 , formulazione di F , è contenuto in P_2 , formulazione di F , diciamo che P_1 è migliore di P_2 .

In generale, una gerarchia di formulazioni è costituita da un insieme di poliedri $P_1 \subseteq P_2 \subseteq P_3 \dots$

Esempio

Consideriamo un problema di knapsack con il vincolo che l'oggetto k può essere scelto se e solo se nella bisaccia sono stati scelti gli oggetti i e j .

Formulazione 1.

$$\begin{aligned} \max \quad & c^T x \\ \text{s.t.} \quad & ax \leq b \\ & x_k \leq x_i \\ & x_k \leq x_j \\ & x \in \{0,1\}^n \end{aligned}$$

Formulazione 2.

$$\begin{aligned} \max \quad & c^T x \\ \text{s.t.} \quad & ax \leq b \\ & 2x_k \leq x_i + x_j \\ & x \in \{0,1\}^n \end{aligned}$$

Esempio (II)

Il rilassamento lineare della **Formulazione 1** è migliore di quello della **Formulazione 2**.

Infatti, il vincolo

$$2x_k \leq x_i + x_j$$

è implicato dai vincoli

$$x_k \leq x_i$$

$$x_k \leq x_j$$

Quindi, $P_1 \subseteq P_2$.

Formulazione ideale

La formulazione del problema di knapsack NON è una formulazione ideale.

Domanda: Esistono casi “fortunati” in cui la formulazione coincide con la formulazione ideale?

E' possibile “caratterizzare” le formulazioni ideali in modo da riconoscerle in tempo polinomiale?

Il caso “fortunato”

Consideriamo il seguente problema di PL in forma standard, in cui $A \in \mathfrak{R}^{m \times n}$ è una **matrice intera** e $b \in \mathfrak{R}^n$ è un **vettore intero**

$$\min c^T x$$

$$Ax = b$$

$$x \geq 0$$

con $\text{rg}(A) = m \leq n$.

Se il problema ammette soluzione ottima finita, allora il metodo del simplesso restituisce la soluzione ottima in corrispondenza di una soluzione di base ammissibile del tipo:

$$X = \begin{bmatrix} X_B \\ X_N \end{bmatrix} = \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix}$$

Il caso “fortunato” (II)

Osservazione: Se la base ottima B ha determinante $\det(B) = \pm 1$, allora il problema lineare ha una soluzione intera.

Infatti:

$B^{-1} = B^a / \det(B)$, dove B^a è la matrice aggiunta. Gli elementi di B^a sono tutti prodotti di termini di B . Pertanto B^a è una matrice intera e, poiché $\det(B) = \pm 1$, B^{-1} è ancora intera.

Pertanto la soluzione di base $x_B = B^{-1}b$ è intera per ogni intero b .

Matrici unimodulari

Definizione: Una matrice A intera $m \times n$ ($m \leq n$) si dice *unimodulare* se ogni sua sottomatrice B di dimensioni $m \times m$ ha $\det(B) = \{-1, 0, +1\}$.

Dall'osservazione precedente si ottiene il seguente risultato:

Teorema: Se A è una matrice unimodulare e b è un vettore intero, il poliedro $P = \{x: Ax = b, x \geq 0\}$ ha tutte le soluzioni di base intere.

Matrici totalmente unimodulari

Definizione: Una matrice A si dice *totalmente unimodulare* (TUM) se **ogni** sua sottomatrice quadrata ha determinante $\{-1, 0, +1\}$.

Proprietà delle matrici TUM

Una matrice A è TUM se e solo se

- la matrice trasposta A^T è TU.
- la matrice (A, I) è TU.

Matrici totalmente unimodulari

Consideriamo ora un problema di programmazione lineare intera

$$P = \min\{c^T x : Ax \geq b, x \in Z_+^n\}$$

(con A matrice intera e b vettore intero), e scriviamo il suo rilassamento lineare

$$\min c^T x$$

$$Ax \geq b \quad (P_{RL})$$

$$x \geq 0$$

Per portare questo problema in forma standard bisogna inserire le variabili di slack y :

$$\min c^T x$$

$$Ax - Iy = b \quad (P'_{RL})$$

$$x, y \geq 0$$

Matrici totalmente unimodulari

- Dalle proprietà delle matrici TUM abbiamo che se A è TUM, allora anche (A, I) è TUM.
- Inoltre ogni matrice TUM è, in particolare, una matrice unimodulare.
- Pertanto, poiché (A, I) è unimodulare, dal teorema precedente possiamo dedurre che P'_{RL} ha tutte le soluzioni di base intere.

Conclusione: Se A è una matrice intera TUM il poliedro P_{RL} ha tutti i vertici interi \Rightarrow Risolvendo il rilassamento lineare otteniamo l'ottimo per il problema intero.

Teorema (Hoffman-Kruskal [1956])

Teorema: Sia A una matrice intera. Il poliedro P definito da

$$P = \{x: Ax \geq b, x \geq 0\}$$

ha tutti i vertici interi *per ogni vettore intero b* se e solo se A è TUM.

Condizioni per la TU

Teorema (criterio di sufficienza): A è TUM se

- i) $a_{ij} \in \{-1, 0, 1\}$
- ii) Ogni colonna ha al più due coefficienti non nulli
- iii) Esiste una partizione (M_1, M_2) dell'insieme delle righe M tale che ogni colonna j contenente due coefficienti non nulli soddisfa

$$\sum_{i \in M_1} a_{ij} = \sum_{i \in M_2} a_{ij}$$

Osservazione:

- se la colonna j contiene due elementi $a_{ij} \neq 0$ e $a_{kj} \neq 0$ dello stesso segno allora $i \in M_1$ e $k \in M_2$.
- se la colonna j contiene due elementi $a_{ij} \neq 0$ e $a_{kj} \neq 0$ di segno opposto allora $i, k \in M_1$ oppure $i, k \in M_2$.

Dimostrazione

Supponiamo che le condizioni *i)*, *ii)* e *iii)* siano soddisfatte.

Dobbiamo dimostrare che ogni sottomatrice quadrata B di A ha $\det(B) \in \{-1, 0, 1\}$.

Procediamo per induzione:

Se B è una sottomatrice 1×1 , banalmente $\det(B) \in \{-1, 0, 1\}$.

Supponiamo ora che la tesi valga per ogni sottomatrice di A di dimensioni $(n - 1) \times (n - 1)$ e consideriamo una sottomatrice B di dimensioni $n \times n$.

- Se B contiene una colonna nulla, $\det(B) = 0$.

Dimostrazione

- Se B contiene una colonna con un unico elemento diverso da zero, allora $\det(B) = \pm \det(B')$, dove B' è di ordine $(n - 1) \times (n - 1)$. Pertanto, dall'ipotesi induttiva, $\det(B) \in \{-1, 0, 1\}$.
- Se ogni colonna di B contiene due elementi $\neq 0$, dall'ipotesi per la colonna j -esima si ottiene

$$\sum_{i \in M_1} a_{ij} = \sum_{i \in M_2} a_{ij}$$

Per come sono costruiti gli insiemi M_1 ed M_2 , elementi di segno opposto della colonna j -esima appartengono allo stesso insieme e quindi si elidono, mentre elementi uguali appartengono a insiemi diversi. Esiste quindi una combinazione lineare delle righe che fornisce il vettore nullo.

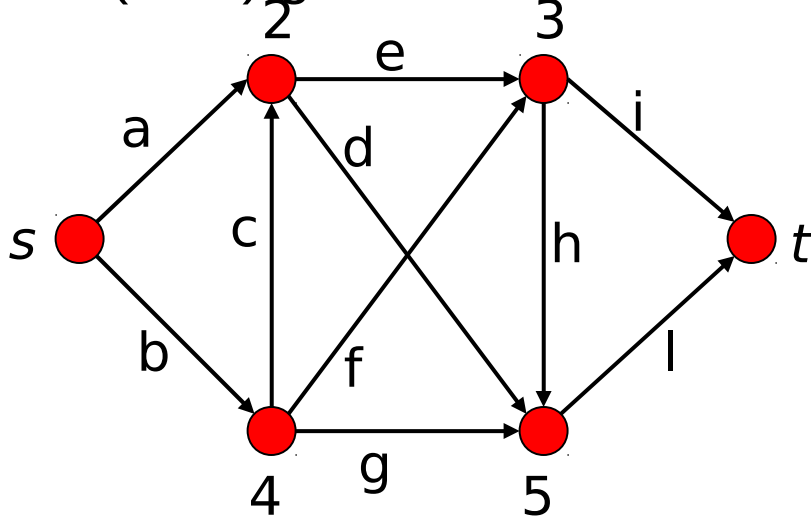
$$\sum_{i \in M_1} A_i - \sum_{i \in M_2} A_i = 0$$

Questo implica che $\det(B) = 0$.



Esempi di matrici TUM

$G(N, A)$ grafo diretto



M = Matrice di incidenza nodi-archi

$M_1 = M, M_2 = \emptyset$

	a	b	c	d	e	f	g	h	i	l
s	1	1	0	0	0	0	0	0	0	0
2	-1	0	-1	1	1	0	0	0	0	0
3	0	0	0	0	-1	-1	0	1	1	0
4	0	-1	1	0	0	1	1	0	0	0
5	0	0	0	-1	0	0	-1	-1	0	1
t	0	0	0	0	0	0	0	0	-1	-1

Teorema: La matrice di incidenza di un grafo diretto è TUM.

Problema di cammino minimo

Dati: $G(N, A)$ grafo diretto, due nodi (s, t) , vettore $c \in R_+^{|A|}$

$$\min \sum_{ij} c_{ij} x_{ij}$$

$$\sum_{k \in \delta^+(s)} x_{sk} - \sum_{k \in \delta^-(s)} x_{ks} = 1$$

$$\sum_{k \in \delta^+(i)} x_{ik} - \sum_{k \in \delta^-(i)} x_{ki} = 0 \quad \text{per } i \in V \setminus \{s, t\}$$

$$\sum_{k \in \delta^+(t)} x_{tk} - \sum_{k \in \delta^-(t)} x_{kt} = -1$$

$$0 \leq x_{ij} \leq 1, \text{ intera} \quad \text{per } (i,j) \in A$$

Problema di cammino minimo

$$z = \min \sum c_{ij} x_{ij}$$

st

$$Ax = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

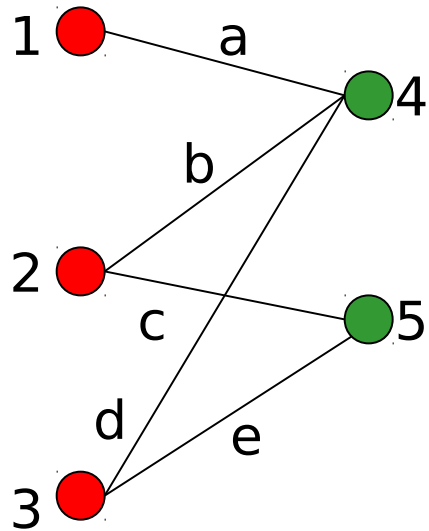
$$x \geq 0$$

$$x \leq 1$$

~~$$x \in \{0, 1\}$$~~

La stipula di interezza può essere rimossa in quanto A è TU

Matrice di incidenza di grafi bipartiti



	a	b	c	d	e
1	1	0	0	0	0
2	0	1	1	0	0
3	0	0	0	1	1
4	1	1	0	1	0
5	0	0	1	0	1

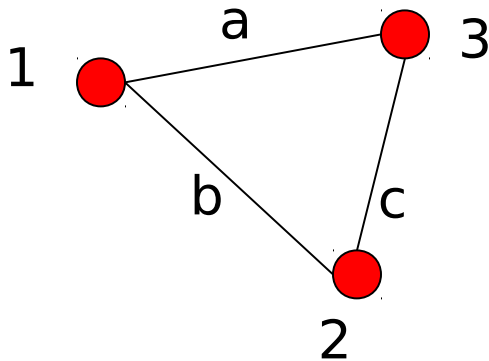
M_1

M_2

Teorema: Un grafo è bipartito se e solo se la sua matrice di incidenza è totalmente unimodulare.

Matrici di incidenza

Esempio di matrice di incidenza di un grafo che non è TUM.



	a	b	c
1	1	1	0
2	0	1	1
3	1	0	1

Matrici totalmente unimodulari

Primale	$\min c^T x$ $Ax \geq b$ $x \geq 0$	$\max y^T b$ $A^T y \leq c^T$ $y \geq 0$	Duale
---------	---	--	-------

Supponiamo che A sia TUM (quindi il primale è intero).

Poiché A è TUM se e solo se la matrice trasposta A^T è TUM, segue che anche il duale è intero.

Teorema: Se A è TUM allora sia il problema primale che il problema duale sono interi.

Algoritmo di Held & Karp
per il calcolo dell'1-albero di peso
minimo

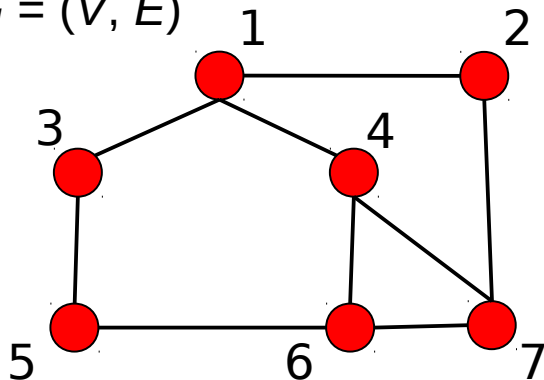
1-albero

Ricordiamo che:

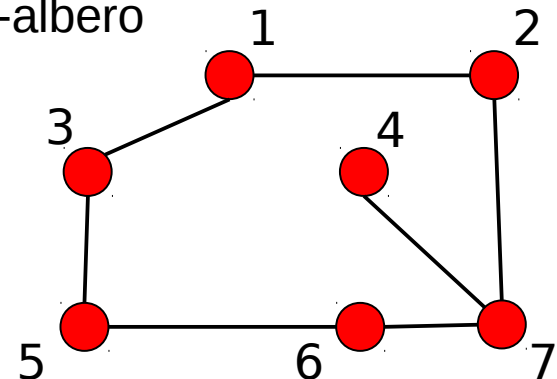
Definizione: Dato un grafo $G = (V, E)$, un *1-albero* è un sottografo di G che consiste di due archi adiacenti al nodo 1 più gli archi di un albero ricoprente i nodi $\{2, \dots, n\}$.

Esempio

$G = (V, E)$



1-albero



1-albero

Osservazione: Ogni ciclo hamiltoniano è un 1-albero.

Pertanto, il problema

Dati: grafo $G = (V, E)$, pesi sugli archi c_e per ogni arco $e \in E$.

Domanda: trovare un 1-albero di peso minimo.

è un rilassamento del problema del TSP.

Valore ottimo dell' 1-albero = lower bound per il valore ottimo del TSP

In particolare, se

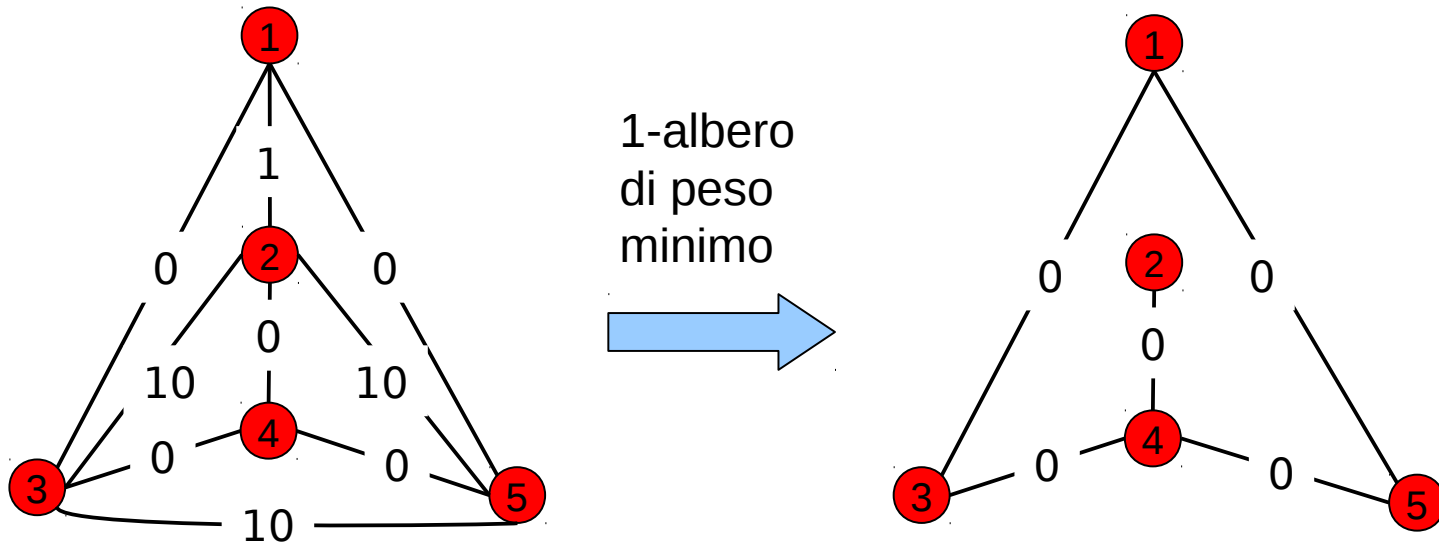
$$A = \min\{c_e + c_f : e, f \in \delta(1), e \neq f\}, e$$

$$B = \text{costo del minimo albero ricoprente su } G \setminus \{1\}$$

allora $A+B$ è un lower bound per il valore ottimo del TSP su G .

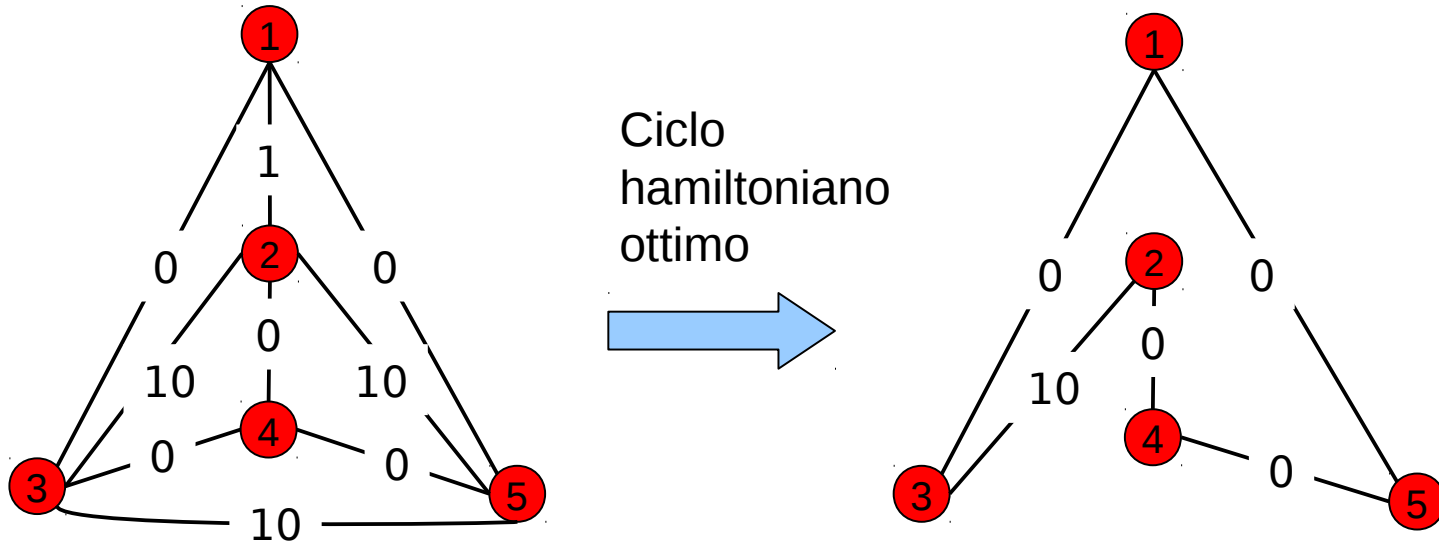
Esempio

Consideriamo il seguente grafo:

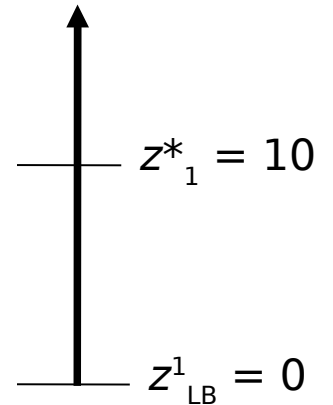


Il valore del lower bound è $z_1^{\text{LB}} = 0$. Pertanto, il ciclo hamiltoniano ottimo ha valore $z_1^* \geq 0$.

Esempio



Il valore della soluzione ottima del TSP è $z_1^* = 10$.

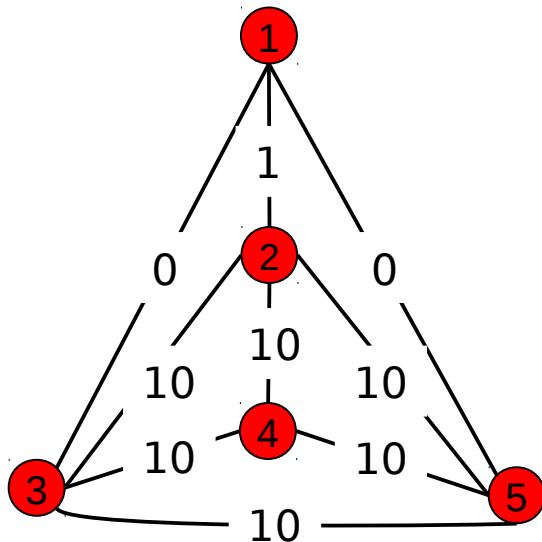


Esempio

Osservazione: La soluzione associata all'1-albero di peso minimo può utilizzare tutti e tre gli archi incidenti sul nodo 4 e aventi peso nullo.

D'altra parte, il ciclo hamiltoniano ottimo deve necessariamente utilizzare un arco avente peso diverso da 0.

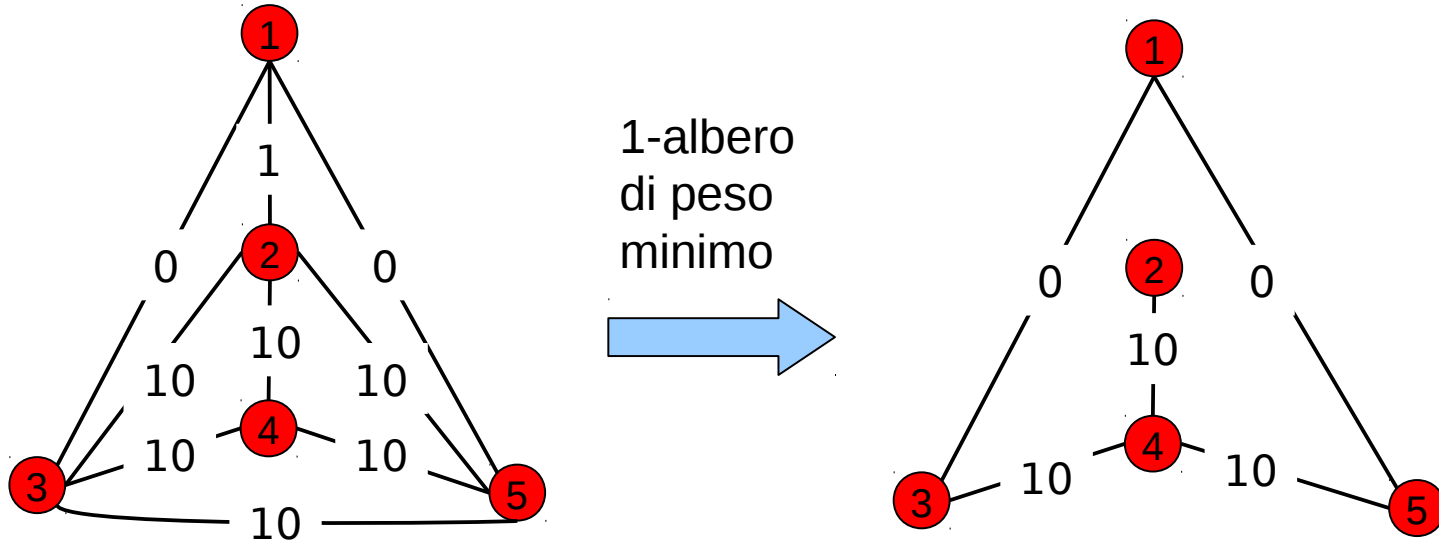
Cosa succede se incrementiamo di un valore pari a 10 il peso degli archi incidenti sul nodo 4?



Ogni ciclo hamiltoniano utilizza esattamente due archi incidenti sul nodo 4. Pertanto il costo di ogni ciclo hamiltoniano è incrementato di un valore pari a 20, ossia $z_2^* = z_1^* + 20$.

Esempio

Calcoliamo l'1-albero di peso minimo sul grafo con i costi degli archi variati.



In questo caso $z_2^{\text{LB}} = 30$ e pertanto il ciclo hamiltoniano ottimo ha valore $z_2^* \geq 30$.

Esempio

Poiché

$$z_1^* = z_2^* - 20 \geq 30 - 20 = 10$$

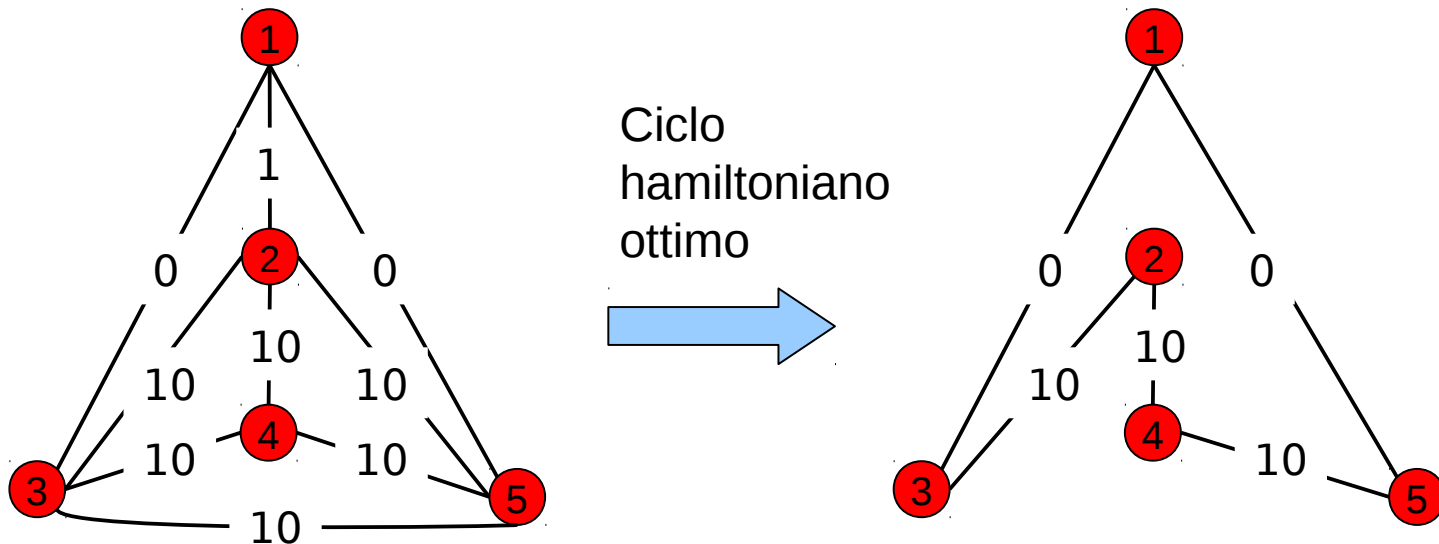
possiamo concludere che il ciclo hamiltoniano sul grafo originario (ossia senza i costi alterati su alcuni archi) deve avere un costo almeno pari a 10.

Tramite una semplice trasformazione abbiamo migliorato il lower bound sul valore ottimo del problema iniziale.

Osservazione: Sebbene la trasformazione non alteri il TSP, essa fondamentalmente altera il calcolo del minimo albero ricoprente e, quindi, dell'1-albero.

Esempio

Osserviamo che il ciclo hamiltoniano ottimo è rimasto invariato ed il suo costo è ora pari a $z_2^* = z_1^* + 20 = 30 = z^{LB}_2$.



Bound di Held e Karp

Dati

- $G = (V, E)$, c_{ij} = peso dell'arco $(i, j) \in E$
- $v_1 \in V$
- y_v = numero reale associato a ciascun nodo $v \in V$
- $w_{ij} = c_{ij} - y_i - y_j$ = peso alterato sull'arco $(i, j) \in E$

sia $w(T)$ il costo dell'1-albero di peso minimo calcolato rispetto ai pesi w_{ij} .

Allora

$$2 \sum_{v \in V} y_v + w(T)$$

è un lower bound per il valore ottimo del TSP sul grafo originario.

Bound di Held e Karp

La trasformazione effettuata sui costi degli archi del grafo consiste

1. nell'associare al nodo $v = 4$ il valore $y_v = -10$
2. nell'alterare i costi degli archi incidenti sul nodo $v = 4$ di una quantità pari a -10 :

$$w_{24} = w_{34} = w_{54} = 0 - (-10) - 0 = 10$$

Il bound ottenuto con questa trasformazione è noto come *bound di Held e Karp* (1970).

Bound di Held e Karp

Per determinare i valori da assegnare ai singoli nodi del grafo, procediamo nel modo seguente:

1. Sia T l'1-albero calcolato ad una generica iterazione rispetto ai pesi correnti w_{ij} .
2. Per ogni nodo v , sia $d_T(v)$ il numero di archi dell'1-albero incidenti su v .
3. Se $d_T(v) \geq 2$, allora il valore y_v deve essere decrementato.
4. Se $d_T(v) = 1$, allora il valore y_v deve essere aumentato.

Bound di Held e Karp

Per aggiornare i valori associati a ciascun nodo alla k -esima iterazione, consideriamo:

- z_{UB} = upper bound sul valore della soluzione ottima del TSP
- z_{LB}^{H} = lower bound di Held e Karp corrente
- $\alpha^{(k)}$ = numero reale nell'intervallo $(0, 2]$

Definiamo la *lunghezza del passo*:

$$t^{(k)} = \alpha^{(k)}(z_{\text{UB}} - z_{\text{LB}}^{\text{H}}) / \sum_{v \in V} (2 - d_T(v))^2$$

e aggiorniamo i valori y_v come segue:

$$y_v = y_v + t^{(k)}(2 - d_T(v))$$

Con questa scelta della lunghezza del passo, la procedura di Held e Karp converge al bound ottimo in tempo polinomiale.

Procedura di Held e Karp

Input

- Grafo $G = (V, E)$ con costi c_e per ogni $e \in E$ e $v_1 \in V$.
- Upper bound z_{UB} .
- Numero reale positivo ITERATIONFACTOR
- Intero positivo MAXCHANGES

Inizializzazione

- $y_v = 0$ per ogni $v \in V$.
- $z_{LB}^* = -\infty$.
- TSMALL = 0.001.
- $\alpha = 2$.
- $\beta = 0.5$.
- NUMITERATIONS = ITERATIONFACTOR \times $|V|$

Procedura di Held e Karp

Algoritmo

For $i = 1$ to MAXCHANGES

For $k = 1$ to NUMITERATIONS

$T = 1$ -albero ottimo rispetto ai costi $w_{ij} = c_{ij} - y_i - y_j$

$z_{LB} =$ lower bound corrispondente all'1-albero T

If $z_{LB} > z_{LB}^*$, then $z_{LB}^* = z_{LB}$.

If T è un ciclo hamiltoniano, then STOP.

$$t^{(k)} = \alpha^{(k)}(z_{UB} - z_{LB}^H) / \sum_{v \in V} (2 - d_T(v))^2.$$

If $t^{(k)} < \text{TSMALL}$, then STOP.

Sostituisci $y_v = y_v + t^{(k)}(2 - d_T(v))$ per ogni $v \in V$.

Sostituisci $\alpha = \beta\alpha$.