



UNIVERSITÀ DEGLI STUDI DI PERUGIA
Dipartimento di Matematica e Informatica

CORSO DI LAUREA TRIENNALE IN INFORMATICA

TESI DI LAUREA

Posizionamento di al massimo tre storage node in una griglia bidimensionale

Laureando

Andrea Madotto

Relatore

Maria Cristina Pinotti

Anno Accademico 2013/2014

Alla mia famiglia che mi è stata vicina, ai miei amici di sempre che non sono mancati mai. Ai miei amici dell'università e della piscina che mi hanno accompagnato in questo percorso. Un ringraziamento speciale a Federico Corò che ha contribuito alla realizzazione di questo lavoro.

Indice

1	Definizioni preliminari	1
2	Mediana di una griglia	4
2.1	Mediana di una griglia rettangolare	4
2.1.1	Casi particolari	7
2.2	Mediana di una griglia con scalino	11
2.3	Soluzione nota per il calcolo della mediana di una griglia rettangolare	18
2.4	Note e conclusioni	21
3	2-median problem in una griglia	22
3.1	Soluzione enumerativa	23
3.1.1	Algoritmo Enumrativo	23
3.1.2	Analisi dei costi	25
3.2	Soluzione proposta in letteratura	26
3.3	Soluzione con subottimalità	27
3.3.1	Tagli netti	28
3.3.2	Scalino singolo	33
4	3-median problem	35
4.1	Soluzione enumerativa	36
4.1.1	Analisi dell' algoritmo e dei costi	36
4.1.2	Soluzione Enumerativa	37
4.2	Soluzione euristica con formula chiusa	38
4.3	Soluzione euristica in tempo lineare	45

INDICE ii

4.4 Valutazione Sperimentale 49

Introduzione

Il facility location problem, noto anche come location analysis o k-center problem, è una branca della ricerca operativa e della geometria computazionale interessata al posizionamento ottimale di facility per minimizzare i costi di trasporto, considerando anche fattori come evitare l'immissione di materiali pericolosi nei pressi di abitazioni.

Un semplice problema di localizzazione delle facility è il problema Weber, in cui una singola facility deve essere posta con un unico criterio di ottimizzazione, il quale può essere la minimizzazione della somma ponderata delle distanze da un dato insieme di punti. Problemi più complessi considerati in questa disciplina includono il posizionamento di più facility, vincoli sulle posizioni degli impianti e criteri di ottimizzazione più complessi.

In una formulazione di base, il facility location problem considera un insieme di punti D , chiamati demand points, che devono essere serviti dalla facility. L'obiettivo è quello di posizionare un sottoinsieme di facility F , per ridurre al minimo la somma delle distanze da ogni demand points.

Il facility location problem su grafi generali è NP-arduo. Quindi sono stati sviluppati un certo numero di algoritmi di approssimazione per il facility location problem.

In particolare in questo lavoro ci siamo concentrati su grafi con alcuni vincoli che permettono di semplificare il problema. I grafi in questione hanno la proprietà principale di avere al più quattro archi. Questa tipologia di grafi viene chiamata griglia.

Dunque abbiamo risolto il facility location problem in una griglia calcolando la mediana, in quanto è il punto che minimizza il costo di trasmissione dalla posizione di una facility verso tutti i punti della griglia (o da tutti i

punti della griglia verso la posizione della facility). In pratica cerchiamo la posizione migliore dove localizzare una stampante, un sensore speciale in una griglia così che il costo di trasmissione da/verso quella posizione verso/da i nodi della griglia sia minimizzato. Inoltre, vedremo il posizionamento di più facility nella stessa griglia. In particolare analizzeremo il caso di avere due facility e il caso di averne tre.

Capitolo 1

Definizioni preliminari

Consideriamo un grafo finito connesso non orientato $G = (V, E)$, dove V e E sono rispettivamente vertici e archi. L'arco $e \in E$ denota una coppia non orientata di vertici (u, v) connessi. Assumiamo che la lunghezza di ogni arco sia sempre unitaria. Un cammino tra i vertici u e v è una sequenza di vertici tale che $u = u_1, \dots, u_n = v$ con $(u_{i-1}, u_i) \in E$, e $n - 1$ è la lunghezza del cammino.

La distanza fra due vertici è il numero di archi sul loro cammino. Denotiamo questa distanza con $d(u, v)$ dove $u, v \in V$.

Per ogni vertice $u \in V$, il *costo di trasmissione* da/verso u è definito come segue:

$$d(u) = \sum_{v \in V} d(u, v) \quad (1.1)$$

Fra tutti i costi di trasmissione particolarmente interessante è quello di valore minimo. Il vertice in corrispondenza del quale si ottiene il minimo costo di trasmissione è chiamato *mediana*:

$$med(G) = \operatorname{argmin}_{u \in V} \{ d(u) \} \quad (1.2)$$

Per un sottoinsieme di vertici $S \subseteq V$, la distanza tra $u \in V$ e S è data da:

$$d(u, S) = \min \{ d(u, v) \mid v \in S \} \quad (1.3)$$

Si noti che se $u \in S$ allora $d(u, S) = 0$; altrimenti la distanza $d(u, S)$ è la distanza fra u e il nodo più vicino di S . Definiamo tale nodo come il nodo associato ad u in S e lo denotiamo come $\sigma(u, S)$. Inoltre, il costo di trasmissione di S è definito come segue:

$$d(S) = \sum_{u \in V} d(u, S) \quad (1.4)$$

In parole, ad ogni nodo $u \in V$ associamo il nodo di S $\sigma(u, S)$ ad esso più vicino. Il costo di trasmissione di S è quindi la somma, su tutti i nodi di V , della distanza dal nodo a cui sono associati in S .

Dato un intero k con $1 \leq k \leq |V|$ ($|\cdot|$ denota la cardinalità di un insieme), allora denotiamo il costo di trasmissione rispetto all'insieme S di k -mediane con

$$OPT(S, k) = \min \{d(S) \mid S \subseteq V, |S| = k\} \quad (1.5)$$

Inoltre, denotiamo i vertici in corrispondenza del minimo con

$$med(G, k) = \operatorname{argmin}_{S \subseteq V; |S|=k} \{OPT(S, k)\} \quad (1.6)$$

Il grafo prodotto $G = (V, E) = G_1 \times G_2$, di due grafi $G_1 = (V_1, E_1)$ e $G_2 = (V_2, E_2)$, è formato da un insieme di vertici denotato con $V = V_1 \times V_2$ (prodotto cartesiano tra V_1 e V_2) e da un insieme di archi denotato con

$$E = \{((u_1, u_2), (v_1, v_2)) \mid u_1 = v_1 \in V_1 \ (u_2, v_2) \in E_2 \text{ o } u_2 = v_2 \in V_2 \ (u_1, v_1) \in E_1\}$$

In particolare, quando G_1 e G_2 sono path graphs possiamo denotarli con P_n e P_m , dove n e m sono il numero di vertici. Il loro prodotto cartesiano $P_n \times P_m$ è una griglia di dimensioni $n \times m$. Denotiamo una griglia con M e per comodità denotiamo i suoi vertici con coppie ordinate (x_i, y_i) , denotiamo i vertici di M con V_m .

Definizione 1. Data una griglia $M = (V_M, E_M)$ di dimensioni $n \times m$ definiamo la distanza Manhattan tra due vertici (x_i, y_i) e (x_j, y_j) appartenenti a V_m

$$d((x_i, y_i), (x_j, y_j)) = |x_i - x_j| + |y_i - y_j| \quad (1.7)$$

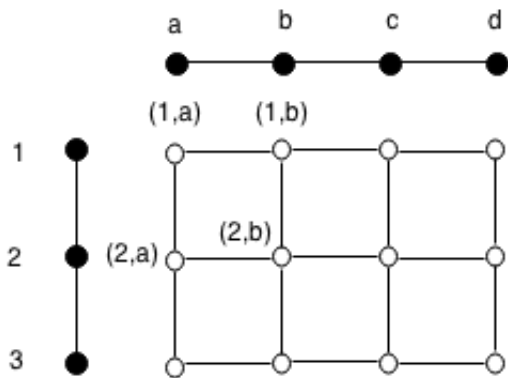


Figura 1.1: Esempio di una griglia M di dimensioni 3×4 ottenuta eseguendo il prodotto cartesiano fra i due paths P_3 e P_4

Capitolo 2

Mediana di una griglia

Si consideri una griglia $n \times m$ i cui vertici sono denotati dalle coppie (i, j) con $1 \leq i \leq n$ e $1 \leq j \leq m$

Definizione 2. Data una griglia M di dimensioni $n \times m$ definiamo mediana $med(M, 1)$ il nodo che minimizza il costo di trasmissione

$$med(M, 1) = \operatorname{argmin}(OPT_{S:|S|=1}(M, 1)) = \operatorname{argmin}(OPT_{v \in M}(M, v)) \quad (2.1)$$

e costo di trasmissione da/verso la mediana come

$$OPT(M, 1) = \min_{(x,y) \in V_M} \left\{ \sum_{i=0}^{n-1} \left(\sum_{j=0}^{m-1} (|x - i| + |y - j|) \right) \right\} \quad (2.2)$$

Per semplicità di notazione utilizzeremo $med(M)$ per indicare $med(M, 1)$ sopra definito.

2.1 Mediana di una griglia rettangolare

Teorema 1. Data una griglia M di dimensioni $n \times m$ con $m, n > 0$, $med(M)$ è il vertice $(x, y) \in V_m$ dove

$$x = \frac{n-1}{2} \quad y = \frac{m-1}{2} \quad (2.3)$$

Dimostrazione. Per determinare $med(M)$, consideriamo per un generico vertice $(x, y) \in V_M$ il costo di trasmissione

$$d(x, y) = \sum_{i=0}^{n-1} \left(\sum_{j=0}^{m-1} (|x - i| + |y - j|) \right) \quad (2.4)$$

Possiamo eliminare il modulo all'interno delle sommatorie riscrivendo l'espressione come segue:

$$d(x, y) = \underbrace{\sum_{i=0}^x \left(\sum_{j=0}^y (x - i + y - j) \right)}_{Q1} + \underbrace{\sum_{i=x+1}^{n-1} \left(\sum_{j=y+1}^{m-1} (i - x + j - y) \right)}_{Q2} + \underbrace{\sum_{i=0}^x \left(\sum_{j=y+1}^{m-1} (x - i + j - y) \right)}_{Q3} + \underbrace{\sum_{i=x+1}^{n-1} \left(\sum_{j=0}^y (i - x + y - j) \right)}_{Q4} \quad (2.5)$$

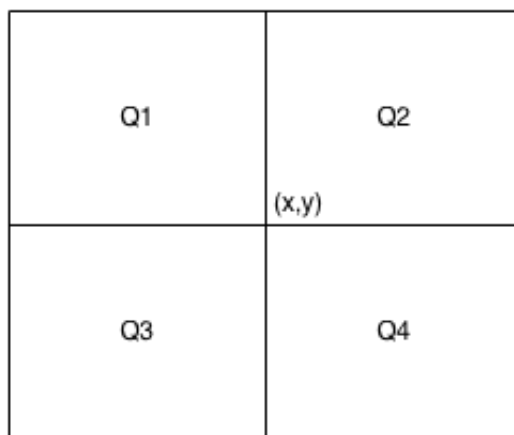


Figura 2.1: Divisione del calcolo delle equazioni

Avendo sommatorie ad indice finito siamo in grado di svolgerle e di sommarle. Pertanto $d(x, y) = d_{Q1}(x, y) + d_{Q2}(x, y) + d_{Q3}(x, y) + d_{Q4}(x, y)$ dove

$$d_{Q1}(x, y) = \sum_{i=0}^x \left(\sum_{j=0}^y (x - i + y - j) \right) = 1/2(x+1)(y+1)(x+y) \quad (2.6)$$

$$d_{Q2}(x, y) = \sum_{i=x+1}^{n-1} \left(\sum_{j=y+1}^{m-1} (i - x + j - y) \right) = 1/2(m-y-1)(n-x-1)(m+n-x-y) \quad (2.7)$$

$$d_{Q3}(x, y) = \sum_{i=0}^x \left(\sum_{j=y+1}^{m-1} (x - i + j - y) \right) = 1/2(x+1)(m-y-1)(m+x-y) \quad (2.8)$$

$$d_{Q4}(x, y) = \sum_{i=x+1}^{n-1} \left(\sum_{j=0}^y (i - x + y - j) \right) = 1/2(y+1)(n-x-1)(n-x+y) \quad (2.9)$$

Quindi avremo che

$$d(x, y) = 1/2(2ny^2 - 2mny + 2ny + 2mx^2 - 2mnx + 2mx + mn^2 + m^2n - 2mn) \quad (2.10)$$

Al fine di determinare il vertice (x, y) che minimizza il costo di trasmissione calcoliamo i punti stazionari passando alle derivate parziali

$$\begin{cases} 2mx - mn + m = 0 & f_x \\ 2ny - mn + n = 0 & f_y \end{cases}$$

quindi avremo i seguenti punti stazionari

$$\begin{cases} x = \frac{mn-m}{2m} \\ y = \frac{mn-n}{2n} \end{cases} \Rightarrow \begin{cases} x = \frac{n-1}{2} \\ y = \frac{m-1}{2} \end{cases}$$

Ora dimostriamo che sono punti di minimo. Quindi costruiamo la matrice dell' Hessiano ovvero la matrice delle derivate parziali seconde

$$H(x, y) = \begin{bmatrix} 2m & 0 \\ 0 & 2n \end{bmatrix}$$

Verifichiamo le condizioni per cui il suddetto punto è un minimo ovvero che la $f_{xx} > 0$ e che il determinante della matrice sia maggiore di zero. Quindi avremo che

$$f_{xx} > 0 \Rightarrow 2m > 0 \quad \det(H(x, y)) > 0 \Rightarrow 4mn > 0$$

Poiché entrambe le condizioni sono verificate in quanto $m, n > 0$, allora il punto stazionario è un punto di minimo per la funzione e quindi rappresenta la mediana di M . \square

2.1.1 Casi particolari

Dato che stiamo lavorando in una griglia, abbiamo un insieme finito di vertici V_M ad indice intero allora il punto di mediana deve appartenere ai vertici, quindi deve essere un intero. Il teorema precedente risolve il problema nel caso in cui sia m che n sono dispari, in quanto il risultato della divisione è intera.

Ora andiamo ad analizzare i restanti tre casi: il caso dove m, n sono entrambi pari, il caso dove m è pari e n è dispari ed infine il caso dove m è dispari e n è pari.

Caso m pari, n pari

Proposizione 1. *Data una griglia M di dimensioni $n \times m$ con m, n pari, la $Med(M)$ è data dalla combinazione dei vertici $(x_i, y_j) \in V_M$ dove*

$$x_1 = \left\lfloor \frac{n-1}{2} \right\rfloor \quad x_2 = \left\lceil \frac{n-1}{2} \right\rceil \tag{2.11}$$

$$y_1 = \left\lfloor \frac{m-1}{2} \right\rfloor \quad y_2 = \left\lceil \frac{m-1}{2} \right\rceil$$

quindi avremo le quattro soluzioni equivalenti nei punti

$$(x_1, y_1) \quad (x_1, y_2) \quad (x_2, y_1) \quad (x_2, y_2) \tag{2.12}$$

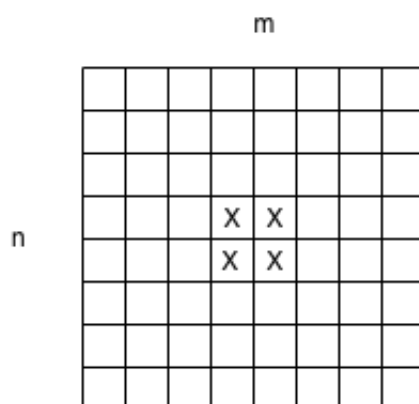


Figura 2.2: Esempio delle quattro posizioni equivalenti per una griglia di dimensioni 8×8

Dato che m e n sono pari la soluzione della divisione sarebbe venuta non intera e dunque un soluzione non valida; allora abbiamo utilizzato floor e ceil in modo da renderla intera. Notiamo che i quattro punti proposti sono equivalenti, in quanto non modificano la soluzione proposta dal Teorema 1.

Caso m pari, n dispari Ora andiamo ad analizzare il caso m pari ed n dispari. Quindi avremo la seguente proposizione

Proposizione 2. *Data una griglia M di dimensioni $n \times m$ con m pari e n dispari, la $Med(M)$ è data dalla combinazione dei vertici $(x_i, y_j) \in V_M$ dove*

$$x_1 = \frac{n-1}{2} \quad y_1 = \left\lfloor \frac{m-1}{2} \right\rfloor \quad y_2 = \left\lceil \frac{m-1}{2} \right\rceil \quad (2.13)$$

quindi avremo le due soluzioni equivalenti nei punti

$$(x_1, y_1) \quad (x_1, y_2) \quad (2.14)$$

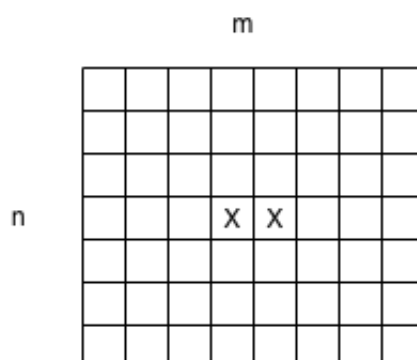


Figura 2.3: Esempio delle 2 posizioni equivalenti per una griglia di dimensioni 7×8

Come nel caso precedente abbiamo usato floor e ceil per rendere intero l'indice.

Caso m dispari, n pari Ora analizziamo l'ultimo caso ovvero quando abbiamo m dispari ed n pari. Quindi avremo

Proposizione 3. *Data una griglia M di dimensioni $n \times m$ con m dispari e n pari, la $Med(M)$ è data dalla combinazione dei vertici $(x_i, y_j) \in V_M$ dove*

$$x_1 = \left\lfloor \frac{n-1}{2} \right\rfloor \quad x_2 = \left\lceil \frac{n-1}{2} \right\rceil \quad y_1 = \frac{m-1}{2} \quad (2.15)$$

quindi avremo le due soluzioni equivalenti nei punti

$$(x_1, y_1) \quad (x_2, y_1) \quad (2.16)$$

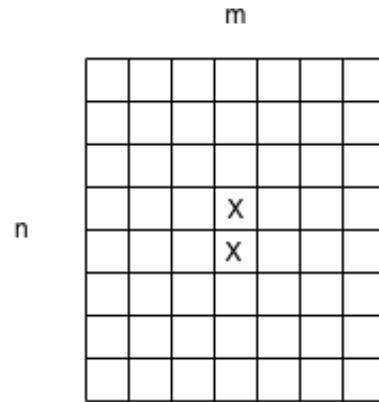


Figura 2.4: Esempio delle 2 posizioni equivalenti per una griglia di dimensioni 8×7

Notiamo che il problema di avere un soluzione ad indice intero è data dal fatto di avere in una delle dimensioni della griglia un numero pari. Infatti quando ciò accade la soluzione non è intera.

Nella prossima sezione analizziamo il problema della mediana in una griglia non rettangolare, ma con l'aggiunta di un'altra griglia a ridosso di uno dei lati. Anche questa soluzione ha un costo computazionale di $O(1)$, in quanto siamo in grado di sfruttare la definizione del problema.

2.2 Mediana di una griglia con scalino

Definizione 3. Data un griglia M di dimensioni $n \times m$, definiamo scalino S l'aggiunta di una griglia M' di dimensioni $t \times s$ alla griglia M . Denotiamo la griglia risultante con $M + S$.

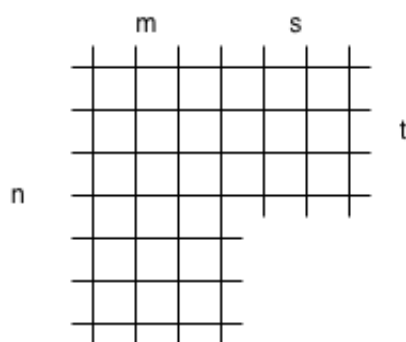


Figura 2.5: Esempio di uno scalino di altezza $s=2$ in posizione $t=4$

Proposizione 4. Sia M una griglia di dimensioni $n \times m$ e S uno scalino di dimensioni $t \times s$, allora

$$\text{med}(M + S) \subset \{C_A, C_B, C_C\}$$

dove C_A C_B C_C sono vertici appartenenti ad $M + S$ di indice pari a

$$\begin{aligned} x_A &= \frac{mn + m + st + s}{2s(m + s)} & y_A &= \frac{mn + n + st}{2n} \\ x_B &= \frac{mn + m + st + s}{2s(m + s)} & y_B &= \frac{-mn + 2mt + st + t}{2t} \\ x_C &= \frac{mn + m - st}{2m} & y_C &= \frac{mn + n + st}{2n} \end{aligned} \quad (2.17)$$

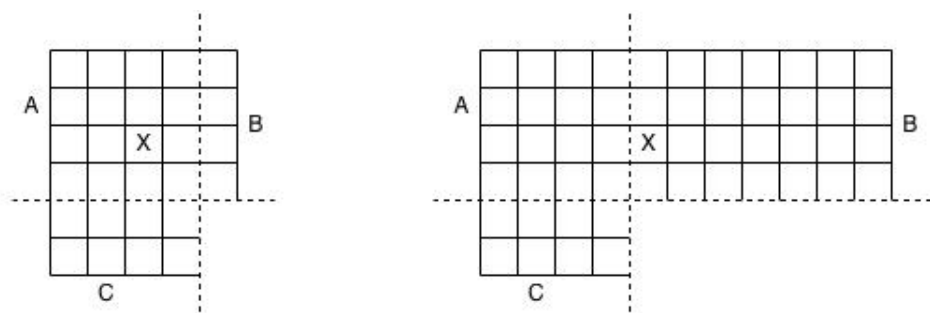


Figura 2.6: Confronto fra due griglie con scalini diversi. La griglia di sinistra ha parametri: $m = 4$, $n = 6$, $s = 1$ e $t = 4$. La mediana si trova in posizione $(2, 2)$ ed appartiene alla regione A . La griglia di destra ha parametri: $m = 4$, $n = 6$, $s = 7$ e $t = 4$. La mediana si trova in posizione $(3, 5)$ ed appartiene alla regione B . Notiamo come si muove la mediana all'aumentare dello scalino.

Dimostrazione. Sia $d(n, x)$ il costo di trasmissione per raggruppare i dati in posizione x in una colonna con n vertici. Allora il suo valore è dato da

$$d(n, x) = \sum_{j=0}^{x-1} j + \sum_{j=x+1}^{n-x} (j - x) = \frac{n^2}{2} - nx + \frac{n}{2} + x^2 - x$$

Ora dividiamo la nostra griglia $M + S$ in tre sottogriglie distinte che chiamiamo A, B, C .

La sottogriglia A ha dimensioni $(n - t) \times m$, la sottogriglia B ha dimensioni $t \times s$ e la sottogriglia C con le stesse dimensioni di A ma si trova sotto di essa, come illustrato in Figura 2.2.

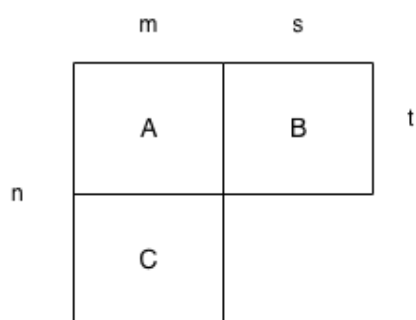


Figura 2.7: Divisione griglia

Data questa divisione, distinguiamo i seguenti tre casi che riguardano la posizione della mediana:

- la mediana è in A
- la mediana è in B
- la mediana è in C

Dunque andiamo a calcolare il costo di trasmissione di ognuno di questi tre casi in funzione delle coordinate dell'eventuale mediana.

Supponiamo che la mediana si trovi nella sottogriglia A , allora il costo di trasmissione è dato da

$$Costo(A) = m d(n, x_A) + s d(t, x_A) + t \sum_{w=m+1}^{m+s} (w - y_A) + n \sum_{w=y_A+1}^m (w - y_A) + n \sum_{w=1}^{y_A} (y_A - w) \quad (2.18)$$

dove: $d(n, x_A)$ è il costo per trasmettere i dati verso una colonna di $A \cup C$ a partire dalla posizione x verso tutti gli n nodi della colonna stessa, $d(t, x_A)$ è il costo per trasmettere i dati a tutti i t nodi di una colonna di B a partire dalla posizione x della colonna stessa, $\sum_{w=m+1}^{m+s} (w - y_A)$ è il costo per trasmettere dalla posizione y_A della riga x di A verso tutta la riga x di B , e $\sum_{w=y_A+1}^m (w - y_A) + \sum_{w=1}^{y_A} (y_A - w)$ è il costo per trasmettere dalla posizione y_A a tutta la riga di m posizioni in A . Dato che le sommatorie sono ad indice finito possiamo svolgerle nel seguente modo

$$\begin{aligned} Costo(A) = & m \left(\frac{n^2}{2-n} x_A + \frac{n}{2} + x_A^2 - x_A \right) + s \left(\frac{t^2}{2-t} x_A + \frac{t}{2} + x_A^2 - x_A \right) \\ & + \frac{1}{2} st (2m + s - 2y_A + 1) + \frac{1}{2} n (m - y_A) (m - y_A + 1) \\ & + \frac{1}{2} n (y_A - 1) y_A \end{aligned} \quad (2.19)$$

Ora abbiamo un'equazione che rappresenta il costo di trasmissione in funzione di x_A e y_A . Come nella sezione 2.1 andiamo a calcolare il minimo di questa funzione. Calcoliamo le derivate parziali rispetto ad x_A ed y_A e le poniamo

uguali a zero. Quindi avremo

$$\begin{cases} x_A = \frac{mn+m+st+s}{2(m+s)} \\ y_A = \frac{mn+n+st}{2n} \end{cases}$$

Questi rappresenteranno i punti stazionari della nostra funzione. Ora dimostriamo che sono punti di minimo. Quindi costruiamo la matrice dell'Hessiano ovvero la matrice delle derivate parziali seconde.

$$H(x, y) = \begin{bmatrix} 2(m+s) & 0 \\ 0 & 2n \end{bmatrix}$$

Verifichiamo le condizioni per cui il suddetto punto è un minimo ovvero che la $f_{xx} > 0$ e che il determinante della matrice sia maggiore di zero. Quindi avremo che

$$f_{xx} > 0 \Rightarrow 2(m+s) > 0 \quad \det(H(x, y)) > 0 \Rightarrow 2(m+s+n) > 0$$

Poiché entrambe le condizioni sono verificate in quanto $m, n, s > 0$ per ipotesi, allora il punto stazionario è un punto di minimo per la funzione. Dunque il vertice (x_A, y_A) rappresenta la mediana nel caso in cui appartenga alla sottogriglia A .

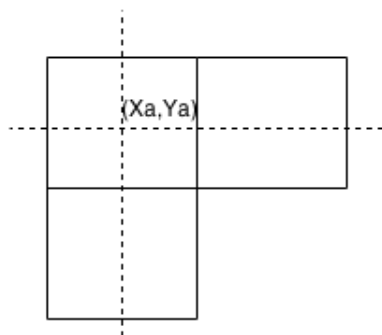


Figura 2.8: Posizione ipotetica della mediana nella sottogriglia A

Supponiamo che la mediana si trovi nella sottogriglia B , allora il costo di

trasmissione è dato da

$$\begin{aligned}
 Costo(B) = & m d(n, x_B) + s d(t, x_B) \\
 & + t \sum_{w=1}^{y_B} (y_B - w) + t \sum_{w=y_B+1}^{m+s} (w - y_B) \\
 & + (n - t) \sum_{w=1}^m (y_B - w)
 \end{aligned}$$

dove: $d(n, x_B)$ è il costo per trasmettere il dato dalla posizione x alla colonna di n vertici in AUC , $d(t, x_B)$ è il costo per trasmettere il dato dalla posizione x alla colonna di t vertici in B , $\sum_{w=1}^{y_B} (y_B - w) + \sum_{w=y_B+1}^{m+s} (w - y_B)$ è il costo per trasmettere i dati in una riga di $A \cup B$ a partire dalla posizione y_B , e $\sum_{w=1}^m (y_B - w)$ è il costo per trasmettere i dati agli m vertici più a sinistra di una riga di $A \cup B$ a partire dalla posizione y_B . Dato che le sommatorie sono ad indice finito possiamo svolgerle nel seguente modo

$$\begin{aligned}
 Costo(B) = & m \left(\frac{n^2}{2-n} x_B + \frac{n}{2} + x_B^2 - x_B \right) + s \left(\frac{t^2}{2-t} x_B + \frac{t}{2} + x_B^2 - x_B \right) \\
 & + \frac{1}{2} t (m + s - y_B) (m + s - y_B + 1) + \frac{1}{2} t (y_B - 1) y_B \\
 & - \frac{1}{2} (n - t) m (m - 2y_B + 1)
 \end{aligned} \tag{2.20}$$

Ora abbiamo un'equazione che rappresenta il costo di trasmissione in funzione di x_B e y_B . Come nella regione A andiamo a calcolare il minimo di questa funzione. Calcoliamo le derivate parziali rispetto ad x_B ed y_B e le poniamo uguali a zero. Quindi avremo

$$\begin{cases} x_B = \frac{mn+m+st+s}{2(m+s)} \\ y_B = \frac{-mn+2mt+st+t}{2t} \end{cases}$$

Come nel caso precedente dobbiamo passare all'Hessiano, e dimostrare la minimalità dei punti. Dato che i calcoli sono analoghi al precedente caso, saltiamo questo passaggio. Dunque il vertice (x_B, y_B) rappresenta la mediana

nel caso in cui appartenga alla sottogriglia B .

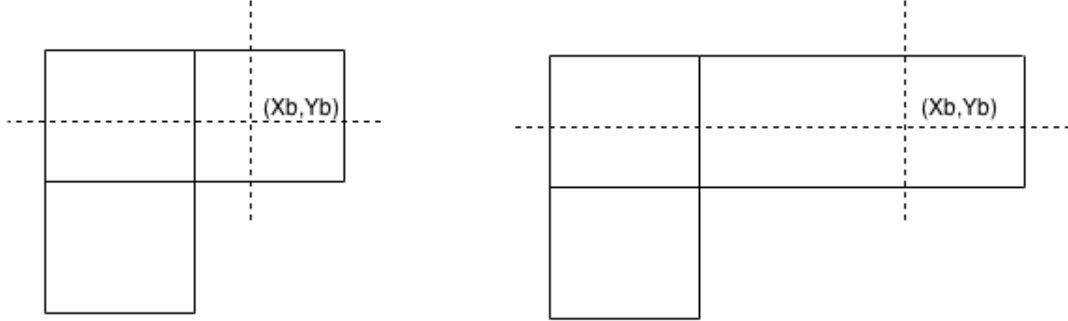


Figura 2.9: Posizione dell'eventuale mediana nella regione B , inoltre possiamo vedere come si muove la mediana all'aumentare di s .

Infine supponiamo che la mediana si trovi nella sottogriglia C , allora il costo di trasmissione è dato da

$$Costo(C) = m d(n, x_C) + n d(m, y_C) + t \left(\frac{s(s+1)}{2} \right) + s \sum_{w=1}^t (x_C - w) + st(m - y_C)$$

dove: $d(n, x_C)$ è il costo per trasmettere dalla posizione x_C a tutta la colonne di $A \cup C$, $d(m, y_C)$ è il costo per trasmettere dalla posizione y_C alla riga di m vertici, $st(m - y_C)$ è il costo per trasmettere da y_C alla colonna m , $t \left(\frac{s(s+1)}{2} \right)$ è il costo per trasmettere a tutti i vertici in una riga di B a partire dalla colonna m , e $s \sum_{w=1}^t (x - w)$ è il costo di trasmettere da x_C ai topmost t i vertici della colonna y_C . Dato che le sommatorie sono ad indice finito possiamo svolgerle nel seguente modo

$$\begin{aligned} Costo(C) = & m \left(\frac{n^2}{2-n} x_C + \frac{n}{2} + x_C^2 - x_C \right) + n \left(\frac{m^2}{2-m} y_C + \frac{m}{2} + y_C^2 - y_C \right) \\ & + t \left(\frac{s(s+1)}{2} \right) - \frac{1}{2} ts (t - 2x + 1) \\ & + st (m - y_C) \end{aligned} \tag{2.21}$$

Ora abbiamo un'equazione che rappresenta il costo di trasmissione in funzio-

ne di x_C e y_C . Come nelle regioni precedenti andiamo a calcolare il minimo di questa funzione. Calcoliamo le derivate parziali rispetto ad x_C ed y_C e le poniamo uguali a zero. Quindi avremo

$$\begin{cases} x_C = \frac{mn+m-st}{2m} \\ y_C = \frac{mn+n+st}{2n} \end{cases}$$

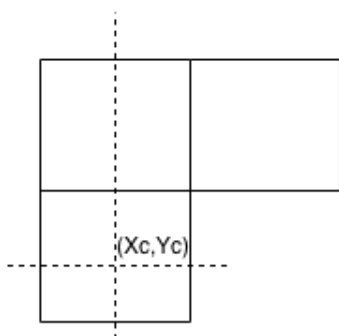


Figura 2.10: Posizione ipotetica della mediana nella sottogriglia C

Dunque la mediana $med(M + S, 1)$ è rappresentata da uno dei tre vertici cercati. Infatti, abbiamo calcolato per ciascuno dei tre vertici (x_A, y_A) , (x_B, y_B) e (x_C, y_C) il costo pagato per trasmettere una distinta copia del dato a partire da uno di tali vertici a tutti gli altri vertici della griglia. Ovviamente il vertice che ha il minimo costo è la mediana. \square

In conclusione, il costo computazionale per il calcolo della mediana proposto in questa sezione è $O(1)$.

Nella prossima sezione presenteremo una soluzione presentata in letteratura per il calcolo della mediana in una griglia rettangolare completa. Questa soluzione sfrutta la moltiplicazione tra due grafi e in particolare la moltiplicazione tra due path graph.

2.3 Soluzione nota per il calcolo della mediana di una griglia rettangolare

[1, 2] La seguente equazione riguarda la distanza tra i vertici dei grafi G_1 e G_2 e il loro grafo prodotto $G_1 \times G_2$. Quindi scriviamo la seguente proposizione che ci servirà in seguito

Proposizione 5. *Dati due grafi G_1, G_2 con $(V_1, E_1) \in G_1$ e $(V_2, E_2) \in G_2$. Allora se $u_1, u_2 \in V_1$ e $v_1, v_2 \in V_2$ avremo che*

$$d((u_1, u_2) , (v_1, v_2)) = d(u_1, v_1) + d(u_2, v_2) \quad (2.22)$$

dove la distanza $d(u_1, u_2)$ è riferita a G_1 e la distanza $d(v_1, v_2)$ è riferita a G_2 .

Dimostrazione. Sia $P : (u_1, u_2) = (a_0, b_0), (a_1, b_1), \dots, (a_s, b_s) = (v_1, v_2)$ il più breve cammino tra i vertici (u_1, u_2) e (v_1, v_2) . Allora, per ogni indice i , una delle seguenti equazioni è fissata per definizione:

$$(a_{i-1}, a_i) \in E_1 \quad e \quad b_{i-1} = b_i \quad (2.23)$$

$$(b_{i-1}, b_i) \in E_2 \quad e \quad a_{i-1} = a_i \quad (2.24)$$

Sia I_1 un insieme di indici i' che soddisfa l'Equazione 2.23 e I_2 un insieme di indici i' che soddisfa l'Equazione 2.24. Allora, $|I_1| + |I_2| = s$ fissato. D'altronde, $\{(a_{i-1}, a_i)\}_{i \in I_1}$ è il più breve percorso tra $u_1 = a_0$ e $v_1 = a_s$, perché se esiste un cammino minimo $a_0, a'_1, \dots, a'_k, a_s$ tra u_1 e v_1 allora abbiamo un cammino $(a_0, b_0), (a'_1, b_0), \dots, (a'_k, b_0), (a'_s, b_0), \{(a_s, b_i)\}_{i \in I_1}$ tra (u_1, u_2) e (v_1, v_2) il quale è più breve di s , e così contraddice la definizione del cammino P . D'altra parte anche $\{(b_{i-1}, b_i)\}_{i \in I_2}$ è il più breve cammino tra $u_2 = b_0$ e $v_2 = b_s$. Dunque abbiamo che

$$d((u_1, u_2) , (v_1, v_2)) = s = |I_1| + |I_2| = d(u_1, v_1) + d(u_2, v_2)$$

□

Data questa equazione possiamo ottenere la seguente proposizione riguardante il costo di trasmissione

Proposizione 6. *Dati due grafi G_1, G_2 con $(V_1, E_1) \in G_1$ e $(V_2, E_2) \in G_2$ allora*

$$d[(u_1, u_2)] = |V_2|d(u_1) + |V_1|d(u_2) \quad (2.25)$$

Dimostrazione. Dalla definizione di $d(u)$ abbiamo che

$$d[(u_1, u_2)] = \sum \left\{ d[(u_1, u_2) , (v_1, v_2)] ; (v_1, v_2) \in V_1 \times V_2 \right\} \quad (2.26)$$

grazie all'Equazione 2.22 possiamo scrivere

$$d[(u_1, u_2)] = \sum \left\{ d(u_1, v_1) + d(u_2, v_2) ; (v_1, v_2) \in V_1 \times V_2 \right\} \quad (2.27)$$

ora possiamo spezzare questo insieme in due parti dato che vale la proprietà distributiva

$$\begin{aligned} d[(u_1, u_2)] &= \sum \left\{ d(u_1, v_1) ; (v_1, v_2) \in V_1 \times V_2 \right\} \\ &+ \sum \left\{ d(u_2, v_2) ; (v_1, v_2) \in V_1 \times V_2 \right\} \end{aligned} \quad (2.28)$$

dalla definizione di $d(u)$ possiamo scrivere

$$d[(u_1, u_2)] = |V_2|d(u_1) + |V_1|d(u_2) \quad (2.29)$$

in quanto tutti i vertici $V_2 \in G_2$ pagano un costo di trasmissione $d(u_1)$ per allinearsi al generico vertice $u_1 \in V_1$ e viceversa tutti i vertici $V_1 \in G_1$ pagano un costo di trasmissione $d(u_2)$ per allinearsi al generico vertice $u_2 \in V_2$. Da ciò è vero l'asserto. \square

Teorema 2. *Dati due grafi G_1, G_2 con $(V_1, E_1) \in G_1$ e $(V_2, E_2) \in G_2$ allora*

$$Med(G_1 \times G_2) = Med(G_1) \times Med(G_2) \quad (2.30)$$

Dimostrazione. Preso un' arbitrario $u'_1 \in Med(G_1)$ e $u'_2 \in Med(G_2)$ allora per definizione abbiamo che

$$d(u'_i) = \min \{d(u); u \in V_i\} \quad (i = 1, 2) \quad (2.31)$$

Dalla Proposizione 6 possiamo scrivere

$$d[(u'_1, u'_2)] = |V_2|d(u'_1) + |V_1|d(u'_2) \quad (2.32)$$

D'altro canto, sempre per la Proposizione 6, per ogni vertice $(u_1, u_2) \in V_1 \times V_2$ possiamo scrivere

$$d[(u_1, u_2)] = |V_2|d(u_1) + |V_1|d(u_2) \quad (2.33)$$

Allora mettendo a confronto le due equazioni precedenti abbiamo che

$$d[(u'_1, u'_2)] \leq d[(u_1, u_2)] \quad \forall (u_1, u_2) \in V_1 \times V_2$$

Dunque, abbiamo che $(u_1, u_2) \in G_1 \times G_2$, cioè $Med(G_1) \times Med(G_2) \subseteq Med(G_1 \times G_2)$. Invece, per ogni $(u'_1, u'_2) \in Med(G_1 \times G_2)$ e per ogni $u_1 \in V_1$ e $u_2 \in V_2$, abbiamo che

$$d(u'_1) \leq d(u_1) \quad d(u'_2) \leq d(u_2) \quad (2.34)$$

Quindi, se noi assumiamo per assurdo che $d(u'_1) > d(u_1)$, allora avremo che

$$d[(u_1, u'_2)] = |V_2|d(u_1) + |V_1|d(u'_2) < |V_2|d(u'_1) + |V_1|d(u'_2) = d[(u'_1, u'_2)]$$

questo porta all'assurdo in quanto $(u'_1, u'_2) \in Med(G_1 \times G_2)$.

Quindi, $u'_1 \in Med(G_1)$ e $u'_2 \in Med(G_2)$. Dunque abbiamo che $Med(G_1 \times G_2) \subseteq Med(G_1) \times Med(G_2)$, allora è vero l'asserto. \square

Quest'ultimo teorema ci permette di trovare la soluzione del problema calcolando separatamente le mediane di due path graph P_n e P_m .

Quindi una volta trovate le mediane, è sufficiente moltiplicare i due path

così da ottenere una griglia di dimensioni $n \times m$. La soluzione è data dal vertice $(med(P_n, 1), med(P_m, 1))$, in quanto esso è il vertice corrispondente alla moltiplicazione $P_n \times P_m$ dove $med(P_n, 1) \in P_n$ e $med(P_m, 1) \in P_m$.

Possiamo notare che la soluzione ha un costo computazionale di $O(1)$, in quanto la mediana di un path graph P_n si trova nel vertice di indice $n/2$ e la moltiplicazione di due path graph svolge un numero di operazioni costanti. Quindi la soluzione proposta precedentemente e quella proposta in letteratura hanno lo stesso costo computazionale.

2.4 Note e conclusioni

La soluzione proposta nella sezione precedente può essere estesa al problema delle 2-mediane come vedremo nel capitolo successivo. Infatti dobbiamo trovare 2 mediane su un path graph P_n e 1-mediana in un altro path graph P_m . I vertici ottenuti facendo il prodotto cartesiano delle soluzioni parziali dei path danno la soluzione al problema 2-median.

In questo capitolo abbiamo proposto una soluzione originale per il problema di determinare la mediana in una griglia di dimensioni $n \times m$, trovando il vertice mediana, sia per griglie pari sia per griglie dispari o combinazioni di esse, ed una formula chiusa che ci indica il costo di trasmissione della griglia. La soluzione calcola sia la mediana della griglia che il corrispondente costo di trasmissione in tempo $O(1)$. Inoltre abbiamo studiato il caso di matrice a 'scalino'. Anche questo caso può essere risolto in tempo costante.

Capitolo 3

2-median problem in una griglia

In questo capitolo affronteremo il problema di localizzare 2 mediane in una griglia di dimensione $n \times m$, ossia cerchiamo l'insieme S , con $|S| = 2$, tale da minimizzare $d(S)$.

Definizione 4. Data una griglia M di dimensioni $n \times m$ definiremo la 2 – *mediana* di M come

$$med(M, 2) = \operatorname{argmin} \{ d(S) \mid |S| = 2 ; S \subseteq V_M \} \quad (3.1)$$

e il costo di trasmissione

$$OPT(M, 2) = \min \{ d(S) \mid |S| = 2 ; S \subseteq V_M \} \quad (3.2)$$

Nella prima parte del capitolo diamo una soluzione algoritmica che trova il $med(M, 2)$ e $OPT(M, 2)$ con un costo computazionale di $O((mn)^3)$.

In seguito introduciamo una soluzione del 2 – *median problem* proposta in letteratura. Questa soluzione sfrutta le proprietà della moltiplicazione fra due grafi per risolvere il problema in modo semplice.

Infine analizziamo una soluzione che sfrutta la subottimalità del problema.

3.1 Soluzione enumerativa

In questa sezione affrontiamo il 2-median problem utilizzando un algoritmo che sfrutta la definizione di $OPT(M, 2)$.

Dunque cerchiamo tra tutti i possibili costi di trasmissione di un insieme di vertici $S \subset V_M$, con $|S| = 2$, quello di valore minimo.

In formule andremo a cercare il minimo nell'insieme

$$\{ d(S) \mid |S| = 2 ; S \subseteq V_M \}$$

dove $d(S)$ rappresenta il costo di trasmissione fissato l'insieme S .

Il nostro algoritmo deve provare tutte le possibili mediane. Dato che stiamo trattando un 2-median problem, consideriamo candidati ad essere soluzione per il problema tutti i sottoinsiemi della griglia di due vertici.

Allora fissata una coppia di vertici andiamo a valutare il suo valore $d(S)$. Ricordiamo che la definizione di quest'ultimo è

$$d(S) = \sum_{u \in V} d(u, S)$$

Quindi per ogni soluzione fissata calcoliamo il rispettivo costo di trasmissione. Così la coppia con il valore minimo rappresenta $med(M, 2)$ e il rispettivo costo di trasmissione rappresenta $OPT(M, 2)$.

3.1.1 Algoritmo Enumrativo

In questa parte vediamo lo pseudo codice che realizza l'idea descritta precedentemente. In input abbiamo una griglia di dimensioni $n \times m$ invece in output abbiamo sia $med(M, 2)$ sia $OPT(M, 2)$.

Possiamo notare, che per il calcolo delle distanze da i due vertici candidati come mediane, abbiamo utilizzato la metrica Manhattan definita precedentemente. Dunque questo algoritmo scorre tutte le possibili mediane con i due cicli for più esterni e ne calcola il costo di trasmissione con il ciclo for più interno. Naturalmente, come ci suggerisce la definizione, fissate le due mediane ogni vertice sceglie il cammino più breve per arrivare in uno dei due vertici.

Algorithm 1: Soluzione $OPT(M, 2)$ e $med(M, 2)$

Input: Griglia M di dimensioni $n \times m$ **Output:** $med(M, 2)$, $OPT(M, 2)$ **begin**

/* inizializzazione variabili */

 $MIN \leftarrow MAX_INT$ $posizione_min \leftarrow \{(0, 0), (0, 1)\}$

/* cicli per le posizioni delle possibili mediane */

for $(i, j) \in V_M$ **do** **for** $(z, t) \in V_M$ **do**

/* inizializzazione variabile temporanea */

 $costo_tx \leftarrow 0$

/* ciclo per il calcolo del costo di trasmissione */

for $(h, k) \in V_M$ **do**

/* calcolo delle distanze */

 $Distanza_ij \leftarrow |h - i| + |k - j|$ $Distanza_zt \leftarrow |h - z| + |k - t|$

/* controllo per scelta del cammino di costo minimo */

if $Distanza_ij < Distanza_zt$ **then** $costo_tx \leftarrow costo_tx + Distanza_ij$ $\sigma(h, k) = (i, j)$ **else** $costo_tx \leftarrow costo_tx + Distanza_zt$ $\sigma(h, k) = (z, t)$

/* controllo per scelta del costo minimo tra tutte le possibili mediane */

if $costo_tx < MIN$ **then** $MIN \leftarrow costo_tx$ $posizione_min \leftarrow \{(x, y), (z, t)\}$ $OPT(M, 2) \leftarrow MIN$ $med(M, 2) \leftarrow posizione_min$

3.1.2 Analisi dei costi

Data una griglia di dimensioni $n \times m$ l'algoritmo proposto lavora in $O((mn)^3)$, vediamo in dettaglio l'analisi del costo computazionale.

Il ciclo for esterno scorre tutti vertici della griglia, quindi effettua mn passi; stesso numero di passi è effettuato dal ciclo for subito sotto, dato che svolge la stessa operazione. Detto ciò la coppia di cicli for più esterni effettuano $(mn)^2$ passi, in quanto per ogni vertice fissato dal ciclo esterno, il ciclo interno scorre di nuovo tutti i vertici.

Ora per ogni coppia di vertici, dei due cicli for esterni, abbiamo il costo computazionale per calcolare $d(S)$. Allora troviamo un ciclo for che scorre tutti i vertici e per ognuno di essi calcola il minimo cammino per arrivare in una delle due mediane. Il costo di questa operazione è classificabile in $O(1)$ in quanto per calcolare la distanza si effettuano tre operazioni elementari e due moduli, e per calcolare il minimo tra due valori è necessario solamente un confronto. Invece per scorrere tutti i vertici sono necessari, come prima detto, mn passi; dato che questa operazione va fatta per tutte le coppie possibili di mediane (i due cicli for esterni), allora il numero totale di passi è dell'ordine $(mn)^3$.

Le restanti operazioni di assegnamento e confronto sono state considerate $O(1)$ in quanto quando si parla di costo asintotico sono trascurabili. Detto ciò il costo computazionale di questo algoritmo è di $O((mn)^3)$.

Possiamo notare che questo approccio al problema può essere utilizzato anche per un numero di mediane pari a tre, in quanto è sufficiente aggiungere un ciclo che prova tutte le possibili posizioni dell'altro vertice candidato come mediana. Naturalmente l'aggiunta di questo ciclo incide sul costo computazionale, facendolo crescere di un'ordine. Discuteremo di questo algoritmo nel capitolo successivo.

Inoltre questa soluzione vale anche per un generico grafo G , dove come distanza si può utilizzare il risultato di una BFS. In particolare vale in path graph come vedremo nella sezione successiva.

3.2 Soluzione proposta in letteratura

In questa sezione estendiamo la soluzione proposta in letteratura vista nel capitolo precedente. Quindi risolviamo il 2-median problem con la moltiplicazione di due path graph.

Differentemente dal caso precedente in uno dei due path dobbiamo calcolare la due mediana. Per fare questo, utilizziamo l'algoritmo enumerativo analizzato nella sezione precedente, con la differenza nel costo computazionale in quanto in input non abbiamo una griglia $n \times m$ ma un path P_m . Possiamo considerare quest'ultimo come una griglia di dimensioni $m \times 1$. Quindi il costo computazionale risulta essere $O(m^3)$, in quanto per provare tutte le possibili mediane non dobbiamo scorrere mn vertici ma solo m , ovvero i vertici del path.

Dunque la soluzione proposta svolge le seguenti operazioni: calcola la uno mediana in un path graph P_n come nel capitolo precedente, calcola la due mediana in un path graph P_m con il metodo appena descritto, ed infine moltiplica i due path graph per ottenere i punti di mediana. Il Teorema 2 ci assicura che la soluzione data è corretta.

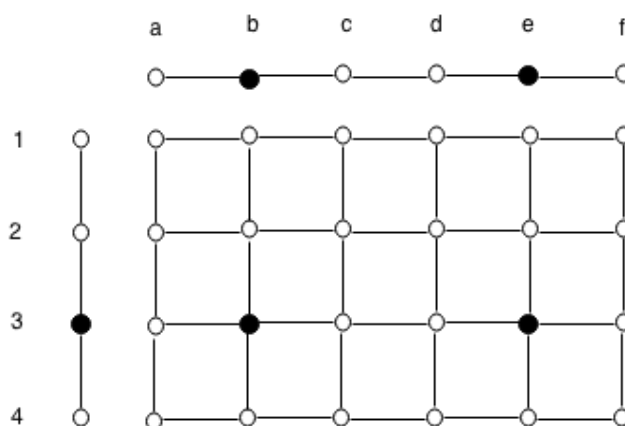


Figura 3.1: Esempio di soluzione con moltiplicazione di due path graph P_4, P_6 . La soluzione è data dai vertici $(3, b)$ e $(3, e)$.

Il costo computazionale dominante in questa soluzione risulta essere quello di calcolare la due mediana in un path graph. I restanti costi, ovvero la uno mediana nell'altro path e la moltiplicazione fra i due path vengono svolti

in un numero costante di passi indipendenti dalla dimensione della griglia. Detto questo risulta chiaro che il costo computazionale di questa soluzione è $O(n^3)$, dove n è la minima estensione fra i due path graph in considerazione. Come nel caso della singola mediana di un griglia, possiamo notare che la soluzione non è unica. Infatti anche nell'esempio precedente abbiamo almeno un'altra soluzione ottima.

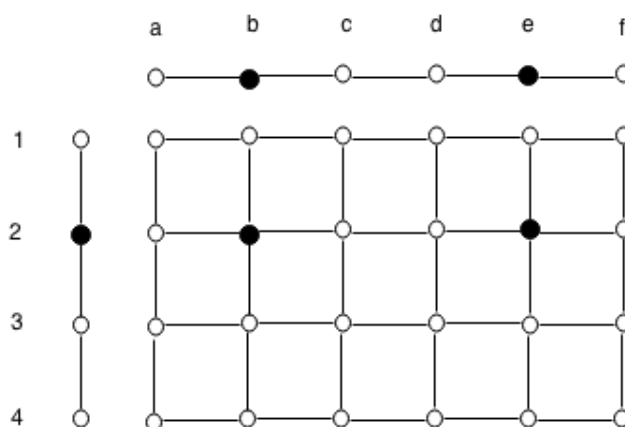


Figura 3.2: Esempio di soluzione equivalente con moltiplicazione di due path graph P_4, P_6 . La soluzione equivalente è data dai vertici $(2, b)$ e $(2, e)$.

3.3 Soluzione con subottimalità

In questo paragrafo analizziamo una soluzione del 2-median problem che sfrutta la subottimalità. L'idea è quella di provare tutte le possibili divisioni in regioni distinte non sovrapposte e di trovare quella di costo minore possibile. Questa divisione in due regioni rappresenta la soluzione del problema, in quanto vale la subottimalità. Ma entriamo nel dettaglio introducendo alcune definizioni e il seguente Lemma.

Lemma 1. *Data una griglia M di dimensioni $n \times m$, allora se partizioniamo la griglia M in due qualsiasi sottogriglie M' e M'' avremo che*

$$OPT(M, 2) \leq OPT(M', 1) + OPT(M'', 1) \quad (3.3)$$

Ossia per ogni sottogriglia cerchiamo la soluzione ottimale, e fra tutte le possibili coppie di sottogriglie tratteniamo quella di costo minimo. In particolare andiamo ad analizzare tutte le sottogriglie tali che

$$OPT(M, 2) = \min_{\substack{M', M'' \subseteq M \\ M' \cup M'' = M \\ M' \cap M'' = \emptyset}} \{OPT(M', 1) + OPT(M'', 1)\} \quad (3.4)$$

Ossia cerchiamo le due sottogriglie M' M'' tali che la somma dei rispettivi costi di trasmissione sia minima. In formule avremo

$$OPT(M, 2) = \min_{|S|=2} \{d(S)\} = d(M_1) + d(M_2) \quad (3.5)$$

dove $M_1 = med(M', 1)$ e $M_2 = med(M'', 1)$.

Nel nostro lavoro abbiamo analizzato due possibili regioni: quelle generate da tagli netti e quelle generate da scalini di altezza unitaria. Iniziamo a considerare le sottogriglie generate da tagli netti

3.3.1 Tagli netti

Definizione 5. Data una griglia M di dimensioni $n \times m$ definiremo taglio i -esimo la divisione in due sottogriglie di dimensione $i \times m$ e $(n - i) \times m$ rispettivamente.

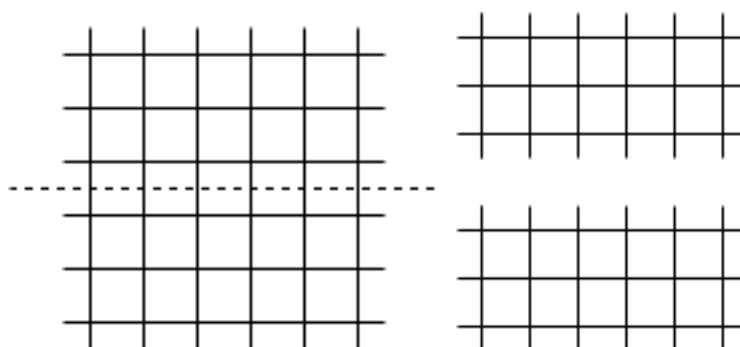


Figura 3.3: Esempio di taglio di una griglia con $i=3$

Definizione 6. Data una griglia M di dimensioni $n \times m$, con m pari, il costo x_i della riga a distanza i dalla riga della mediana è così definito:

$$x_i = i + 2 \sum_{j=0}^k (i + j) \quad (3.6)$$

con $k = m/2$.

Si noti che il costo x_{i-1} della riga a distanza $i-1$ dalla riga della mediana è minore del costo x_i della riga a distanza i dalla riga della mediana.

Proposizione 7. Sia M una griglia di dimensioni $n \times m$ e sia $S \in V_m$ un sottoinsieme di vertici con $|S| = 2$. Se n è pari, il taglio di indice $n/2$ genera due sottogriglie A e B tali che $d(\text{Med}(A)) = d(\text{Med}(B))$. Se n è dispari, il taglio di indice $n/2$ genera due sottogriglie A e B i cui costi differiscono di una sola riga.

Dimostrazione. Sia M una griglia di dimensioni $n \times m$ e supponiamo che n sia pari allora, si fa' un taglio a $n/2$ così da avere due sottogriglie entrambe di dimensioni $n/2 \times m$. Grazie al Teorema 1 sappiamo calcolare la mediana di una griglia di dimensioni qualsiasi, quindi anche delle due appena ricavate. Risulta chiaro che il costo della mediana dipenderà solo dalla dimensione della griglia, quindi con questa divisione entrambe le sottogriglie hanno lo stesso costo.

Per il caso che n sia dispari avremo due tagli equivalenti, ovvero il taglio a $n/2$ e quello a $n/2 + 1$, in quanto non considerando la riga centrale si ritorna al caso n pari. Risulta indifferente assegnare la riga alla sottogriglia A o alla sottogriglia B in quanto essa si trova equidistante alle due mediane quindi avrà lo stesso costo se assegnata ad A o a B . \square

Proposizione 8. Sia M una griglia di dimensioni $n \times m$ allora il taglio con costo minimo è dato dal taglio della Proposizione 7

Dimostrazione. Supponiamo che $n/2$ sia pari, quindi il costo della griglia con taglio a $n/2$ può essere visto come la somma di ogni riga di ognuna delle due

sottogriglie. Allora avremo

$$C = 2 \left(x_0 + 2 \sum_{i=1}^{\lfloor \frac{n+2}{4} \rfloor} x_i - x_{\frac{n+2}{4}} \right) \quad (3.7)$$

Ora consideriamo un taglio con j dispari e $j < n/2$, allora il costo della griglia sarà dato dalla somma del costo della sottogriglia A di dimensioni $j \times m$ e dal costo della sottogriglia B . Si noti che essendo n pari, e j dispari, allora $n-j$ è dispari. Quindi avremo rispettivamente

$$C(A) = x_0 + 2 \sum_{i=1}^{\frac{j-1}{2}} x_i \quad C(B) = x_0 + 2 \sum_{i=1}^{\frac{n-j-1}{2}} x_i \quad (3.8)$$

Quindi chiameremo C' la somma di $C(A)$ e di $C(B)$. Ora dimostriamo che $C < C'$ quindi avremo che

$$\begin{aligned} 2 \left(x_0 + 2 \sum_{i=1}^{\frac{n+2}{4}} x_i - x_{\frac{n+2}{4}} \right) &< x_0 + 2 \sum_{i=1}^{\frac{j-1}{2}} x_i + x_0 + 2 \sum_{i=1}^{\frac{n-j-1}{2}} x_i \\ 2 \sum_{i=1}^{\frac{n+2}{4}} x_i + 2 \sum_{i=1}^{\frac{n+2}{4}} x_i - 2 x_{\frac{n+2}{4}} &< 2 \sum_{i=1}^{\frac{j-1}{2}} x_i + 2 \sum_{i=1}^{\frac{n-j-1}{2}} x_i \\ 2 \sum_{i=\frac{j-1}{2}+1}^{\frac{n+2}{4}} x_i - 2 x_{\frac{n+2}{4}} &< 2 \sum_{i=\frac{n+2}{4}+1}^{\frac{n-j-1}{2}} x_i \end{aligned}$$

Intanto avremo che $2 x_{\frac{n+2}{4}}$ è sempre maggiore di zero per ipotesi iniziale che $n > 0$. Quindi non ci resta da verificare

$$2 \sum_{i=\frac{j-1}{2}+1}^{\frac{n+2}{4}} x_i < 2 \sum_{i=\frac{n+2}{4}+1}^{\frac{n-j-1}{2}} x_i$$

Notiamo che la sommatoria di destra (A) e la sommatoria di sinistra (B)

faranno lo stesso numero di passi in quanto

$$\begin{aligned}\#(A) &= \frac{n+2}{4} - \left(\frac{j-2}{2} + 1\right) = \frac{n-2j}{4} \\ \#(B) &= \frac{n-j-1}{2} - \left(\frac{n+2}{4} + 1\right) = \frac{n-2j}{4}\end{aligned}$$

quindi ora è facile dimostrare che la disequazione è sempre verificata in quanto il più grande indice della sommatoria di sinistra $\binom{n+2}{4}$ è minore del più piccolo indice della sommatoria di destra $\left(\frac{n+2}{4} + 1\right)$, e per definizione si ha che $x_{i-1} < x_i \forall i = 1, \dots, n$.

Ora, lo stesso ragionamento deve essere ripetuto per ogni $j \leq n/2$ e j pari. Si noti che essendo n pari, allora anche $n - j$ è pari. Dopo tutti i conti, si verifica ancora che il taglio di costo minimo è quello in corrispondenza di $n/2$.

Il ragionamento dovrebbe essere ripetuto per n dispari. \square

La Proposizione 8 trova il taglio netto ottimale a metà della griglia. Naturalmente i tagli superiori ed inferiori hanno un costo superiore, quindi partendo dal primo taglio possibile (in posizione 1) fino ad arrivare all'ultimo taglio possibile (in posizione $n - 1$) la somma dei costi di trasmissione delle due sottogriglie seguirà l'andamento di una parabola. Quest'ultima è rivolta verso l'alto e ha come punto di minimo l'indice del taglio ottimo dunque quello a $n/2$. Da notare che la parabola ha due punti di minimo nel caso in cui n è dispari in quanto esisto due tagli equivalenti. Inoltre, la parabola ha tre punti di minimo quando n è divisibile per 4. Ad esempio con $n = 8$ i possibili tagli netti che separano la griglia hanno indice $i = 1, \dots, 7$. Quindi i tagli con lo stesso valore sono $i = 3, 4, 5$.

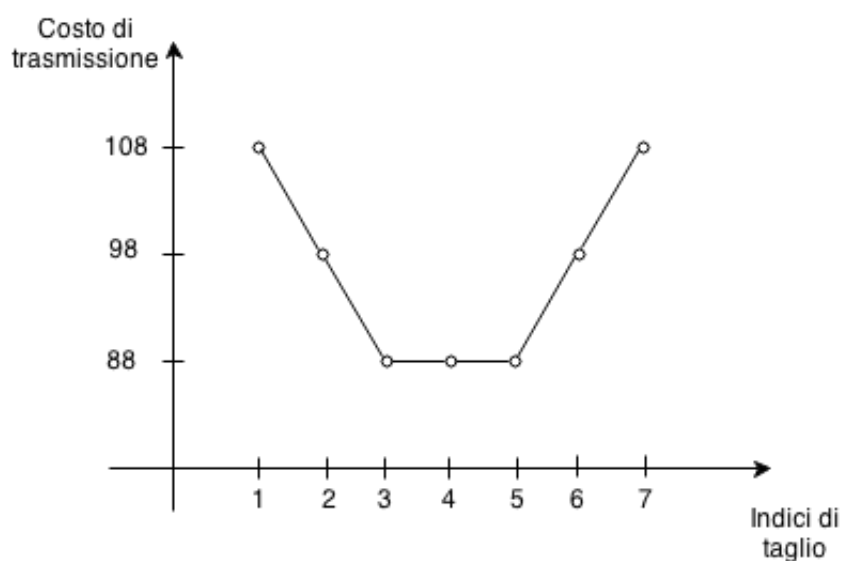


Figura 3.4: Esempio di parabola per una griglia di dimensione 8×5 , nell'ascisse troviamo l'indice dei vari tagli netti e nelle ordinate il rispettivo costo di trasmissione

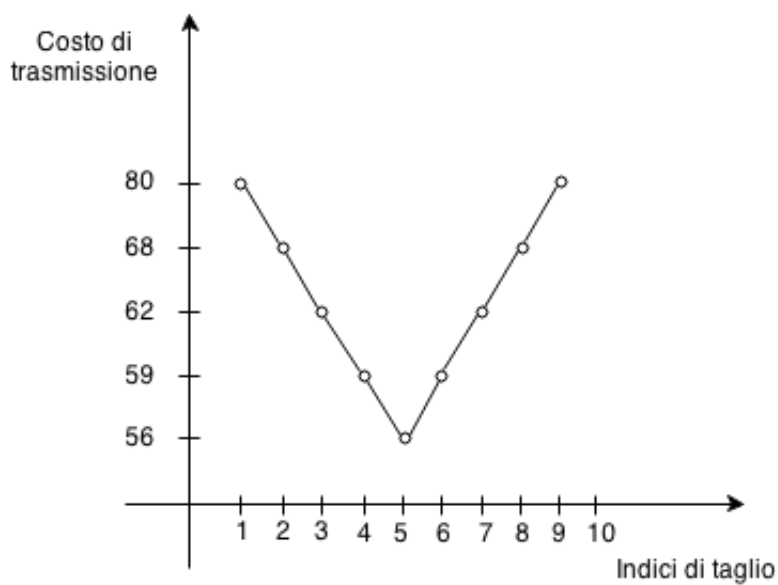


Figura 3.5: Esempio di parabola per una griglia di dimensione 10×3 , nell'ascisse troviamo l'indice dei vari tagli netti e nelle ordinate il rispettivo costo di trasmissione. Qui il punto di minimo è unico

3.3.2 Scalino singolo

Nota bene. Anche se sappiamo qual è il taglio di costo minimo, siamo ancora lontani da trovare l'ottimalità in quanto dovremmo considerare tutte le possibili partizioni dei vertici fra due gruppi M' e M'' , che sono 2^{nm} . Tuttavia è facile rendersi conto che una distribuzione dei vertici M' e M'' , ad esempio, a macchia di leopardo, sicuramente non è ottima in quando ogni vertice deve scegliere di associarsi a M' o M'' a seconda della sua distanza da $med(M')$ e $med(M'')$. Questo riduce moltissimo le sottogriglie da controllare.

Conoscendo già la soluzione per il 2-median problem non continueremo per questa strada, ma semplicemente osserviamo nella prossima sezione che esistono anche tagli non netti con costo uguale. In altre parole, questo conferma che l'assegnamento dei nodi della griglia alle mediane non è unico.

Quindi in questo paragrafo ci occupiamo di sottogriglie con uno scalino ma, questa volta ci limitiamo allo scalino di altezza unitaria.

Definizione 7. Data una griglia M di dimensioni $n \times m$ definiremo un i -scalino di un t -esimo taglio come una suddivisione ulteriore in verticale del taglio. Quindi suddivideremo la riga di indice $t + 1$ in due parti, con taglio verticale in posizione i con $0 \leq i \leq m$, e ne assegneremo una parte alla sottogriglia superiore e la restante parte alla sottogriglia inferiore.

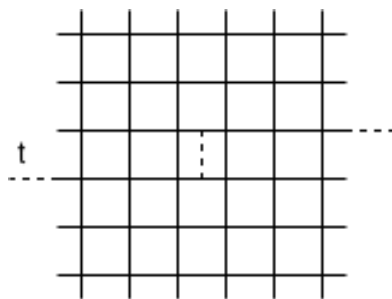
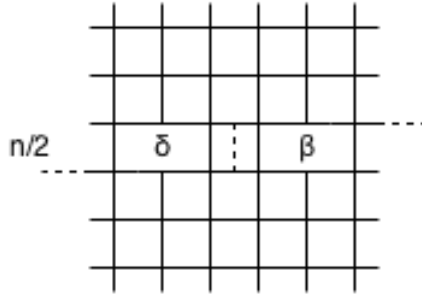


Figura 3.6: Esempio di uno scalino con t -taglio

Proposizione 9. Sia M una griglia di dimensioni $n \times m$, con n dispari e con un $n/2$ -taglio allora il costo della griglia non cambia se si inserisce un i -scalino nella riga $n/2+1$.

Dimostrazione. Data A la griglia con solo un $n/2$ -taglio e B la griglia con l'aggiunta della scalino in posizione i , entrambe di dimensione $n \times m$ con n dispari, allora avremo che la regione superiore e la regione inferiore al taglio avrà egual costo (Proposizione 8). Quindi rimane da confrontare la riga in posizione $n/2 + 1$ che per semplicità indicheremo con ρ . Denoteremo la parte sinistra dello scalino con δ e la parte destra con β .



Grazie alla Definizione 5 possiamo scrivere il costo della riga ρ . Quindi avremo che

$$x_\rho = \rho + 2 \sum_{j=0}^k (\rho + j) \quad (3.9)$$

Con $k = m/2$. Ora notiamo che $\rho = \delta + \beta$ in quanto possiamo scrivere che

$$\delta = \rho + \sum_{j=0}^t (\rho + j) \quad \beta = \sum_{j=0}^z (\rho + j) \quad (3.10)$$

privilegiando δ senza perdita di generalità. Allora risulta chiaro che sommando $\delta + \beta$ avremo esattamente ρ in quanto $t + z = k$. Quindi è equivalente inserire o no uno scalino in un $n/2$ -taglio. \square

Nota 1. Le considerazioni fatte riguardano tagli orizzontali della griglia, naturalmente le esse valgono anche per tagli verticali. Quindi esisterà almeno un'altra soluzione ottima che fa un taglio verticale ad $m/2$ e posiziona i vertici nelle regioni risultanti. Un'altra cosa da notare è che si fanno spesso considerazioni sulla metà sinistra o in basso a sinistra della griglia ma per simmetria tutte le assunzioni fatte valgono anche dagli altri lati.

Capitolo 4

3-median problem

In questo capitolo affrontiamo il problema del 3-median problem con due metodologie distinte, la prima che calcola la soluzione con un algoritmo e la seconda approssima la soluzione con un'euristica. Si noti che non possiamo risolvere il problema vedendo la griglia come prodotto cartesiano di path graphs perchè la tecnica usata nel capitolo 2 funziona solo per un numero di mediane pari, che puo' essere bilanciato fra i due paths. Ad esempio tale tecnica vale per un k -median, dove k è una potenza di 2.

Il problema 3-median sulla griglia M è indicato formalmente con $OPT(M, 3)$ oppure semplicemente con OPT se il contesto non è ambiguo.

Definizione 8. Data una griglia M di dimensioni $n \times m$ definiremo il costo del 3-median di M come

$$OPT(M, 3) = \min \{ d(S) \mid |S| = 3; S \subseteq V_m \} \quad (4.1)$$

e come 3-mediana

$$med(M, 3) = \operatorname{argmin} \{ d(S) \mid |S| = 3; S \subseteq V_M \} \quad (4.2)$$

Cominciamo col riprendere la soluzione enumerativa che trova esattamente i valori delle mediane e quindi anche il miglior costo di trasmissione.

4.1 Soluzione enumerativa

In questa sezione affrontiamo il 3-median problem utilizzando un algoritmo che sfrutta la definizione di $OPT(M, 3)$. Questo algoritmo lavora allo stesso modo di quello presentato nel capitolo precedente. A differenza di questo, si devono aggiungere tutte le combinazioni derivanti dall'inserimento di un nuovo nodo di mediana.

Quindi l'algoritmo sfrutta la definizione del problema, prova tutte le possibili combinazioni di mediane e prende quella con costo di trasmissione minore.

4.1.1 Analisi dell' algoritmo e dei costi

In questo algoritmo a differenza del precedente troviamo un ciclo for in più, in quanto stiamo trattando un 3-median problem. L'aggiunta di questo ciclo va' ad intaccare nel costo complessivo, infatti con l'aggiunta di questo abbiamo un costo pari a $O((mn)^4)$. Tutto ciò è dovuto dal fatto, come spiegato dettagliatamente nel capitolo precedente, che si devono provare tutti i possibili insiemi di tre mediane. Per fare questo si devono scorrere tutte le possibili combinazioni fissando volta alla volta i vertici candidati.

La parte del calcolo del costo di trasmissione per ogni 3 mediana candidata (il ciclo for più interno) è svolto come nell'algoritmo precedente.

Naturalmente la scelta della destinazione di ogni vertice è dettata dalla minima distanza. La cosa che cambia dall'algoritmo precedente è il fatto che si confrontano tre distanze invece che due, ma questo non incide nel costo computazionale.

Le restanti operazioni di assegnamento e confronto sono state considerate $O(1)$ come in precedenza. Detto ciò il costo computazionale di questo algoritmo è di $O((mn)^4)$.

Nella sezione successiva è riportato uno pseudo codice dell'algoritmo proposto.

4.1.2 Soluzione Enumerativa

Algorithm 2: Soluzione $OPT(M, 3)$ e $med(M, 3)$

Input: Griglia M di dimensioni $n \times m$ **Output:** $med(M, 3)$, $OPT(M, 3)$ **begin**

/* inizializzazione variabili */

 $MIN \leftarrow MAX_INT$ $posizione_min \leftarrow \{(0, 0), (0, 1), (1, 0)\}$

/* cicli per le posizioni delle possibili mediane */

for $(i, j) \in V_M$ **do** **for** $(z, t) \in V_M$ **do** **for** $(w, r) \in V_M$ **do**

/* inizializzazione variabile temporanea */

 $costo_tx \leftarrow 0$

/* ciclo per il calcolo del costo di trasmissione */

for $(h, k) \in V_M$ **do**

/* calcolo delle distanze */

 $Distanza_ij \leftarrow |h - i| + |k - j|$ $Distanza_zt \leftarrow |h - z| + |k - t|$ $Distanza_wr \leftarrow |h - w| + |k - r|$

/* controllo per scelta cammino minimo */

 $min_distanza \leftarrow$ $\min\{Distanza_wr, Distanza_zt, Distanza_ij\}$ **if** $Distanza_ij = min_distanza$ **then** $costo_tx \leftarrow costo_tx + Distanza_ij$ $\sigma(h, k) = (i, j)$ **if** $Distanza_zt = min_distanza$ **then** $costo_tx \leftarrow costo_tx + Distanza_zt$ $\sigma(h, k) = (z, t)$ **if** $Distanza_wr = min_distanza$ **then** $costo_tx \leftarrow costo_tx + Distanza_wr$ $\sigma(h, k) = (w, r)$

/* controllo per scelta del costo minimo tra tutte le possibili mediane */

if $costo_tx < MIN$ **then** $MIN \leftarrow costo_tx$ $posizione_min \leftarrow \{(x, y), (z, t), (w, r)\}$ $OPT(M, 3) \leftarrow MIN$ $med(M, 3) \leftarrow posizione_min$

4.2 Soluzione euristica con formula chiusa

In questa sezione proponiamo una soluzione approssimata del 3-median problem. Questo tipo di soluzione è dettata dal fatto che l'algoritmo che calcola OPT ha un costo computazionale che, come abbiamo provato sperimentalmente, per valori di m e n grandi è troppo alto. Quindi per griglie abbastanza grandi risulta molto pesante la sua esecuzione.

Definizione 9. Data una griglia M di dimensioni $n \times m$ definiamo un rombo $r(d, P)$ di centro P e diametro d come il rombo in M con diagonali di lunghezza $2d$ e baricentro, ossia intersezione delle diagonali, nel vertice P .

Tale rombo consiste di un numero di vertici pari a

$$S(d) = 1 + \sum_{i=1}^d 4i = 1 + 2d(d+1) \tag{4.3}$$

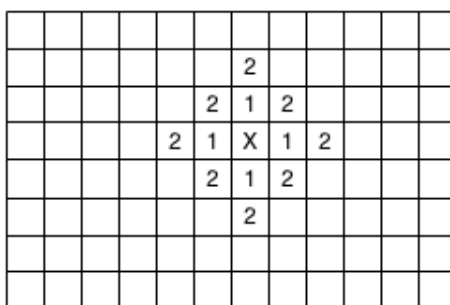


Figura 4.1: Esempio di rombo con $d = 2$ e con $S(d)=13$

Proposizione 10. Data una griglia M di dimensioni $n \times m$ il rombo $r(d, P)$ ha costo di trasmissione

$$C(d) = \sum_{i=1}^d 4i^2 = \frac{4d(d+1)(2d+1)}{6} \tag{4.4}$$

Dimostrazione. Dimostriamo l'asserto per induzione.

Caso base: supponiamo di avere $d = 1$ allora il costo di trasmissione è 4 in

quanto la cella unitaria è la somma del costo di quattro vertici adiacenti al vertice P .

Passo induttivo: supponiamo l'asserto vero per d e dimostriamo l'asserto con $d + 1$. Avendo come raggio del rombo $d + 1$ avremo la seguente somma

$$C(d + 1) = \sum_{i=1}^{d+1} 4i^2$$

dalle proprietà delle sommatorie possiamo spezzare la nostra somma in due parti

$$C(d + 1) = \sum_{i=1}^d 4i^2 + \sum_{i=d+1}^{d+1} 4i^2$$

per ipotesi induttiva possiamo scrivere $C(d + 1) = C(d) + 4(d + 1)^2$, dunque per ogni d $C(d)$ implica $C(d + 1)$. \square

Si noti che il rombo $r(d, P)$ ha il minimo costo di trasmissione fra tutte le sottogriglie di dimensione $S(d)$ perchè raggiunge il massimo numero di vertici $4i$ per ogni distanza, con $0 \leq i \leq d$.

Definizione 10. Data una griglia M di dimensioni $n \times m$ definiamo una tassellazione di k -rombi con centri P_1, P_2, \dots, P_k la sottogriglia M' ricoperta dall'unione di rombi

$$\bigcup_{i=0}^k r(d, P_i) \tag{4.5}$$

e la sottogriglia M ha dimensioni

$$\sum_{i=1}^k S(d)$$

Dato che ci stiamo occupando di un 3-median problem utilizzeremo questa tassellazione, con solo 3 rombi, per cercare un limite inferiore di $OPT(M, 3)$. Quindi ci poniamo il problema di trovare, data una matrice M di dimensione

$n \times m$, un \tilde{d} tale che

$$3S(\tilde{d} + 1) > mn$$

$$3S(\tilde{d}) \leq mn$$

Ciò significa trovare tre rombi di diagonale d più grandi possibile senza che si sovrappongano, in modo da coprire più superficie possibile della griglia. Dato che la tassellazione non tiene conto della posizione reale dei rombi sulla griglia ma solo del numero dei vertici che occupa, dobbiamo fare una particolare considerazione: in questo caso la griglia viene vista come un toroide. Precisamente, i vertici in riga 1 sono adiacenti ai vertici in riga n , e i vertici in colonna 1 sono adiacenti a quelli in colonna m . Chiameremo toroidale una griglia con tali proprietà di vicinanza.

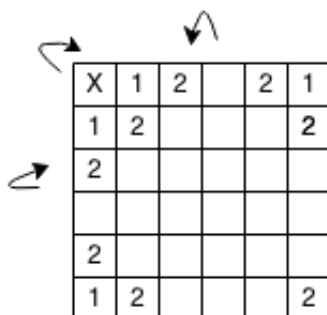


Figura 4.2: Esempio di rombo con $d = 2$ e $P = (0, 0)$, in una griglia toroidale di dimensioni $m = 6$ $n = 6$

Assumendo la griglia toroidale il costo di trasmissione di un rombo è sempre minore o uguale al costo di trasmissione di un rombo di pari dimensioni in una griglia convenzionale (ossia non toroidale). Infatti una griglia toroidale di dimensione $kS(d)$ può essere sempre ricoperta interamente da k rombi, mentre per una griglia convenzionale sono richiesti un numero di rombi maggiore perchè ai bordi i rombi sono incompleti.

Proposizione 11. *Sia M una griglia toroidale di dimensioni $n \times m$ sufficientemente grandi, allora un ricoprimento per la griglia è dato da una tassellazione di 3 rombi con parametro d di ognuno compreso tra*

$$-\frac{3}{2} + \frac{\sqrt{2mn-3}}{2\sqrt{3}} < d \leq -\frac{1}{2} + \frac{\sqrt{4mn-9}}{2\sqrt{3}} \quad (4.6)$$

con $d > 0$.

Dimostrazione. Per ipotesi la griglia ha dimensioni $n \times m$ quindi dovremo cercare un ricoprimento che utilizza 3 rombi tali che il loro parametro d rispetti contemporaneamente le seguenti condizioni

$$\begin{cases} 3S(d+1) > mn & (a) \\ 3S(d) \leq mn & (b) \end{cases}$$

Quindi avremo che la disequazione (a) sarà

$$3 + 6(d+1)(d+2) > mn$$

e quindi sarà verificata nell'intervallo

$$d > -\frac{3}{2} + \frac{\sqrt{2mn-3}}{2\sqrt{3}}$$

in quanto la parte negativa non la considereremo dato che $d > 0$ per ipotesi. Invece la disequazione (b) sarà

$$3 + 3d(d+1) \leq mn$$

e quindi sarà verificata nell'intervallo

$$d \leq -\frac{1}{2} + \frac{\sqrt{4mn-9}}{2\sqrt{3}}$$

in quanto la soluzione negativa non la considereremo dato che $d > 0$ per ipotesi. Entrambe le disequazioni hanno come campo di esistenza $mn > 3/2$ ma questo è sempre verificato in quanto il nostro problema ha senso per

almeno $m, n > 1$.

Dato che

$$-\frac{1}{2} + \frac{\sqrt{4mn - 9}}{2\sqrt{3}} > -\frac{3}{2} + \frac{\sqrt{2mn - 3}}{2\sqrt{3}}$$

è sempre verificato per $mn \geq 9/4$ quindi è sempre vera grazie alle ipotesi, avremo che :

$$-\frac{3}{2} + \frac{\sqrt{2mn - 3}}{2\sqrt{3}} < d \leq -\frac{1}{2} + \frac{\sqrt{4mn - 9}}{2\sqrt{3}}$$

Da ciò vero l'asserto. □

Corollario 1. *Data una griglia toroidale M di dimensioni $n \times m$, definiremo limite inferiore al costo del 3-median in una griglia convenzionale il costo $LSB = 3C(\tilde{d})$ della tassellazione di tre rombi con parametro*

$$\tilde{d} = \frac{\sqrt{2mn - 3}}{2\sqrt{3}}$$

	1			2	
1	X	1	2	Y	2
	1			2	
		3			
	3	Z	3		
		3			

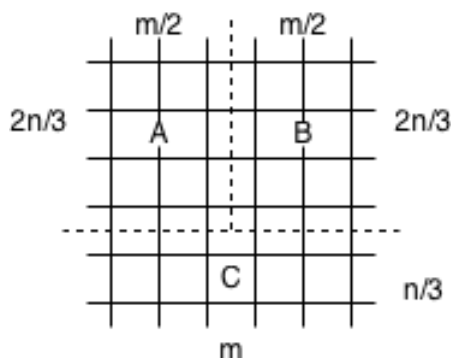
Figura 4.3: Esempio tassellazione con $d = 1$ in una griglia toroidale di dimensioni $m = 6$ $n = 7$

La figura mostra graficamente la considerazione precedente, quindi il costo di questo limite inferiore è minore di $OPT(M, 3)$.

Ora introduciamo la nostra soluzione che approssima il costo della soluzione ottima, l' OPT .

Definizione 11. *Data una griglia M di dimensioni $n \times m$ definiremo SEMP come la soluzione che divide in tre regioni la griglia M . Denoteremo con A la sottogriglia di dimensioni $\frac{2n}{3} \times \frac{m}{2}$, con B l'altra sottogriglia di dimensioni*

$\frac{2n}{3} \times \frac{m}{2}$ e con C la sottogriglia di dimensioni $\frac{n}{3} \times m$. Le divisioni sono mostrate in figura:



Definizione 12. Data una griglia M di dimensioni $n \times m$ le posizioni dei vertici di SEMP sono:

$$\begin{aligned} x_1 &= \frac{2n-3}{6} & y_1 &= \frac{m-2}{4} \\ x_2 &= \frac{2n-3}{6} & y_2 &= \frac{3m-2}{4} \\ x_3 &= \frac{5n-3}{6} & y_3 &= \frac{m-1}{2} \end{aligned} \quad (4.7)$$

ovvero i punti di mediana delle tre sottogriglie.

Inoltre il costo di SEMP è dato dalla seguente equazione

$$C(SEMP) = 1/36(mn - 3)(6m + 5n) \quad (4.8)$$

dove $C(SEMP)$ è stato ottenuto sommando il costo di trasmissione nelle tre griglie A, B, C. Si ricorda che il costo ciascuna griglia è ottenuto sostituendo le coordinate delle mediane in Equazione 4.7 nell'Equazione 2.10.

Proposizione 12. Sia M una griglia di dimensioni $n \times m$ allora il costo di SEMP è maggiore o uguale al costo della soluzione ottima.

Dimostrazione. La dimostrazione è immediata in quanto SEMP divide la griglia in regioni e l'assegnazione dei vertici non è dettata dalla minima distanza ma solo dalla divisione effettuata. Quindi il costo di SEMP considerato

è maggiore del costo di prendere i punti della soluzione e calcolarne il minimo. Allora $C(SEMP)$ non è il costo ottimo per i punti della soluzione dunque non sarà il costo ottimo neanche per $C(OPT)$. \square

Ora andremo a dimostrare che il rapporto tra il costo di SEMP e LSB è costante, in modo da mostrare che il loro comportamento non cambia al crescere della griglia.

Teorema 3. *Sia M una griglia di dimensioni $n \times m$, con m, n sufficientemente grandi, allora il rapporto tra $C(SEMP)$ e LSB è costante ed il suo valore è*

$$K = \frac{11}{3} \cdot \sqrt{2} \cong 5,1$$

Dimostrazione. Dato che stiamo ragionando in ordine di grandezza possiamo considerare una griglia con $n = m$, quindi quadrata, e sostituire nelle formule il $\max\{n, m\}$. Passiamo al limite

$$\lim_{n \rightarrow \infty} \frac{C(SEMP)}{LSB}$$

Ricordando che $LSB = 3 C(d) = 4 d(d+1)(2d+1)/3$ e d è lo stesso del Corollario 1, possiamo svolgere alcune minorazioni dato che d dipende da n . Allora avremo che

$$LSB = \frac{4d^3 + d^2 + 4d + 4}{3} > \frac{4d^3}{3}$$

Dato che $d = \sqrt{2mn - 3}/(2\sqrt{3})$ possiamo sostituirlo all'interno della formula

$$LSB > \frac{4}{3} \left(\frac{1}{2} \cdot \sqrt{\frac{2n^2 - 3}{3}} \right)^3 > \frac{4}{3} \left(\frac{1}{2} \cdot \sqrt{\frac{2n^2}{4}} \right)^3 = \frac{n^3}{12\sqrt{2}}$$

per $n > \sqrt{6}$ e $n = \min\{n, m\}$. Dato che n è grande per ipotesi la minorazione è corretta.

D'altro canto $C(SEMP) = 1/36(mn - 3)(6m + 5n)$, quindi possiamo fare la sostituzione e una maggiorazione, ed avremo

$$C(SEMP) = \frac{11n^3 - 33n}{36} < \frac{11n^3}{36}$$

Dunque ora possiamo risolvere il limite e dimostrare che è una costante

$$\lim_{n \rightarrow \infty} \frac{\frac{11n^3}{36}}{\frac{n^3}{12\sqrt{2}}} = \frac{11}{3} \cdot \sqrt{2}$$

□

Grazie a questa dimostrazione abbiamo limitato il rapporto tra il costo dell'euristica e il costo della soluzione ottima con una costante, in quanto vale la seguente disequazione.

$$\frac{C(SEMP)}{OPT} \leq \frac{C(SEMP)}{LSB} \leq K \quad (4.9)$$

la soluzione proposta è una approssimazione della soluzione ottima, ossia $C(SEMP) \leq KOPT$.

4.3 Soluzione euristica in tempo lineare

Un'altra soluzione proposta è quella data da un algoritmo che approssima la soluzione e lavora in $O(nm)$. L'idea di questo algoritmo è quella di posizionare inizialmente i vertici in posizioni note, poi di definire tre regioni distinte date da questi vertici ed infine di risolvere il problema della mediana all'interno di queste regioni. Le posizioni iniziali sono date dai vertici di indice

$$\begin{aligned} x_1 &= 0 & y_1 &= 0 \\ x_2 &= 0 & y_2 &= m \\ x_3 &= n & y_3 &= \frac{m}{2} \end{aligned}$$

Ora riporteremo lo pseudo codice dell'algoritmo.

L'algoritmo *PuntiMed* calcola i punti di mediana delle regioni che si vengono a creare. Questo algoritmo avrà un costo di $O(k + z)$ dove k e z sono rispettivamente le dimensioni delle righe e delle colonne della regione.

Algorithm 3: PuntiMed

Data: SottoGriglia S con i vertici V_S **Result:** Indici x, y che rappresentano la mediana**begin**

/* inizializzazione strutture dati */

 $y \leftarrow 0$ **for** $C_i \in \text{column}(S)$ **do** $T_1 \leftarrow T_1 + \text{Card}(C_i)$ $T_2 \leftarrow \text{Card}(C_0)$

/* ricerca della posizione per le colonne */

while $T_2 > T_1$ **do** $y \leftarrow y + 1$ $T_2 \leftarrow T_2 + \text{Card}(C_y)$ $T_1 \leftarrow T_1 - \text{Card}(C_y)$ $x \leftarrow 0$ **for** $R_i \in \text{row}(S)$ **do** $T_1 \leftarrow T_1 + \text{Card}(R_i)$ $T_2 \leftarrow \text{Card}(R_0)$

/* ricerca della posizione per le righe */

while $T_2 > T_1$ **do** $x \leftarrow x + 1$ $T_2 \leftarrow T_2 + \text{Card}(R_x)$ $T_1 \leftarrow T_1 - \text{Card}(R_x)$

Algorithm 4: Soluzione SOL

Data: Griglia M di dimensioni $n \times m$ **Result:** Costo e indici dei vertici della soluzione**begin**

/* inizializzazione */

 $(x_1, y_1) \leftarrow (0, 0)$ $(x_2, y_2) \leftarrow (0, m)$ $(x_3, y_3) \leftarrow (n, m/2)$

/* divisioni in regioni */

for $(i, j) \in V_M$ **do** $K \leftarrow \{ \tilde{k} \in K \mid K = 1, 2, 3 \mid \min\{d((i, j), (x_{\tilde{k}}, y_{\tilde{k}}))\} \}$ **if** $K = 1$ **then** \lfloor $regioneA \leftarrow (i, j)$ **else if** $K = 2$ **then** \lfloor $regioneB \leftarrow (i, j)$ **else** \lfloor $regioneC \leftarrow (i, j)$

/* calcolo punti della soluzione */

 $(x_1, y_1) \leftarrow PuntMed(regioneA)$ $(x_2, y_2) \leftarrow PuntMed(regioneB)$ $(x_3, y_3) \leftarrow PuntMed(regioneC)$

/* calcolo costo soluzione */

for $(i, j) \in V_M$ **do** \lfloor $Route \leftarrow \min\{d((i, j), (x_k, y_k)) \mid k = 1, 2, 3\}$ \lfloor $SOL \leftarrow SOL + Route$

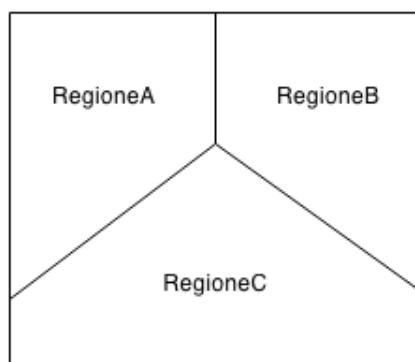


Figura 4.4: Regioni che si creano

L'algoritmo SOL esamina due volte tutti i vertici e richiama tre volte *PuntiMed* nelle regioni in Figura 4.3. Il costo all'interno dei for è $O(1)$ in quanto calcola il minimo tra tre numeri e fa' altri tre confronti ed assegnazioni. Quindi il costo totale sarà

$$SOL = (3n + 3m) + 2mn$$

dove il primo termine è il costo delle tre chiamate di *PuntiMed*. Quindi concludiamo che $SOL \in O(mn)$. Naturalmente SOL ha un costo maggiore uguale di OPT in quanto l'algoritmo proposto calcola il minimo delle distanze dai vertici proposti, ma non è detto che i punti scelti siano quelli ottimi. D'altro canto SOL ha un costo minore di SEMP in quanto lavora con le minime distanze diversamente da quest'ultima. Dunque questa soluzione proposta approssima in modo migliore la soluzione ottima ed inoltre potremmo scrivere che:

$$\frac{C(SOL)}{OPT} \leq \frac{C(SEMP)}{OPT} \leq \frac{C(SEMP)}{LSB} \leq K \quad (4.10)$$

4.4 Valutazione Sperimentale

In questa sezione mostriamo alcuni grafici che rappresentano i costi analizzati in precedenza. Per facilitare la lettura del grafico abbiamo imposto che $n = m$, cioè abbiamo considerato griglie quadrate. In quanto senza questa precisazione avremmo avuto dei grafici tridimensionali, sicuramente meno adatti all'obiettivo.

In particolare vedremo il rapporto $\frac{SEMP}{LSB}$ e $\frac{SOL}{LSB}$ calcolati sperimentalmente e analiticamente, l'andamento delle soluzioni proposte in confronto della soluzione ottima (solo per input piccoli) ed ancora le soluzioni proposte con dimensioni molto grandi per evidenziare il comportamento asintotico.

Quando si parla di input grandi si intendono delle griglie di dimensioni $n \times m$ dell'ordine di 10^5 . Non abbiamo potuto rappresentare la soluzione ottima ($OPT(M, 3)$) con input di queste dimensioni dato il suo costo computazionale. Per rendere l'idea, un calcolatore con un processore Intel(R) Core(TM) i5-3317U CPU 3.60GHz di clock e con 3.8 GB di RAM, il tempo necessario per svolgere l'algoritmo con griglie di dimensioni 30×30 è di circa 2 ore, e per griglie di dimensioni 35×35 il tempo necessario per svolgere l'algoritmo è di circa 10 ore. Anche questo giustifica l'approccio al problema scelto, ovvero una soluzione approssimata, in quanto per m, n molto grandi il tempo di attesa è inaccettabile.

Il primo grafico mostra come il rapporto calcolato sperimentalmente ($\frac{SEMP}{LSB}$) tende a diventare costante e di poco superiore all'unità per griglie di dimensioni maggiori 10. Ciò significa che l'approssimazione fatta è buona, infatti quando la griglia cresce la soluzione approssimata non si distacca dalla soluzione ottima, in quanto viene imprigionata tra $SEMP$ e LSB .

Nelle ascisse viene riportata la dimensione della griglia e nelle ordinate il valore del rapporto quindi l'errore commesso. Vediamo come inizialmente il rapporto è molto alto in quanto le griglie in gioco sono molto piccole, ma con il crescere delle dimensioni si appiattisce molto rapidamente.

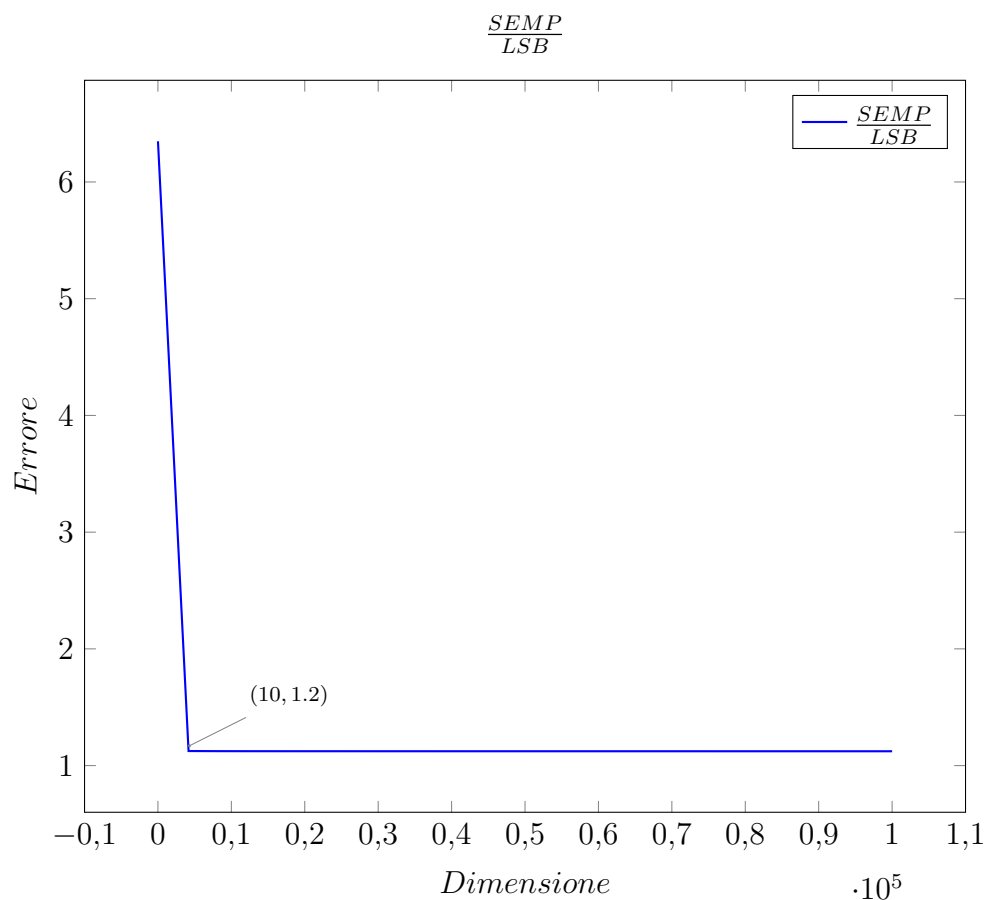


Figura 4.5: Grafico del rapporto $\frac{SEMP}{LSB}$ per griglie di dimensioni molto grandi

Per capire meglio il comportamento del rapporto diminuiamo la grandezza della griglia e aggiungiamo anche $\frac{SOL}{LSB}$ per capire meglio come le due soluzioni differiscono. Possiamo notare che il rapporto $\frac{SOL}{LSB}$ è sempre minore di $\frac{SEMP}{LSB}$ in quanto come prima detto SOL è un' approssimazione migliore della soluzione ottima. In questo grafico dovrebbe essere riportata anche la costante K . Essendo però, tale costante 5 e molto maggiore del reale rapporto, non la riportiamo per non appiattire il grafico.

Si noti inoltre che l'andamento zig-zagato è dovuto al fatto che LSB è una funzione a gradini.

Come dimensioni della griglia riportate sull'asse delle x abbiamo scelto un intervallo compreso tra 12 e 50 così da accentuare la convergenza. Il valore

sperimentale del rapporto è di circa 1,2 il che ci indica che si avvicina molto ad uno dunque è un altro segnale che l'approssimazione K è corretta e l'euristica è molto buona.

Si noti infine che il costo della soluzione SOL è sempre inferiore a $SEMP$.

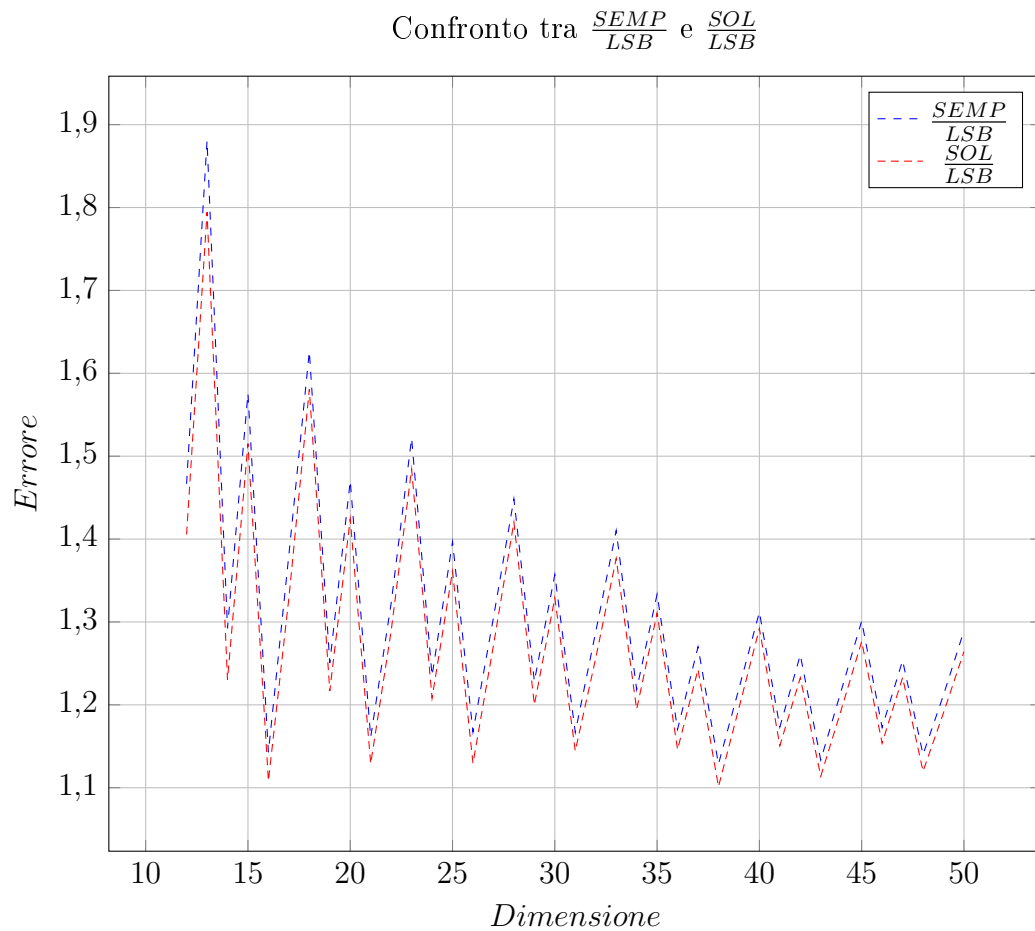


Figura 4.6: Grafico che riporta sia $\frac{SEMP}{LSB}$, $\frac{SOL}{LSB}$.

Il secondo grafico riporta la percentuale di miglioramento che si ha tra *SEMP* e *SOL*. Quindi consideriamo griglie di dimensioni comprese tra 12 e 36, per facilitare la lettura, e per ognuna di esse calcoliamo il rapporto percentuale tra le due soluzioni. Nelle ascisse, come nei grafici precedenti, viene riportata la dimensione della griglia invece nelle ordinate la percentuale di miglioramento.

A fronte di un costo computazionale $O(mn)$, dunque, in media si ha un miglioramento del 2,8% della soluzione.

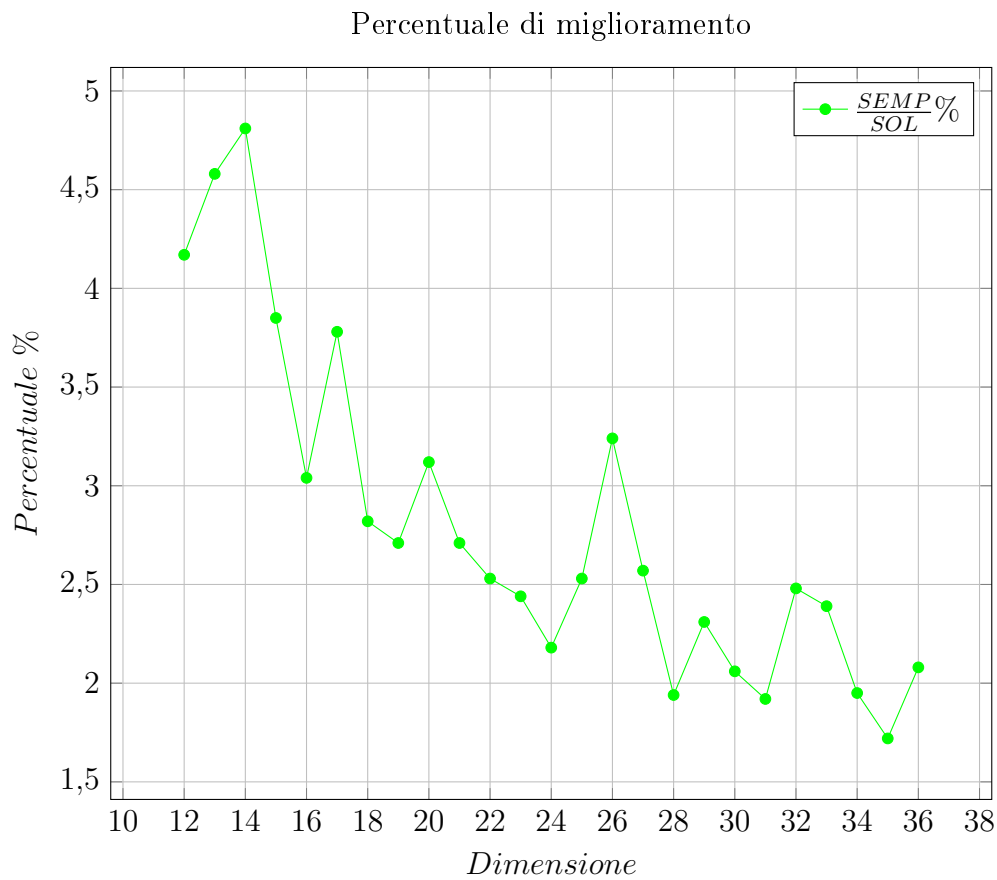


Figura 4.7: Grafico che riporta il miglioramento percentuale che si ha se si utilizza *SOL* rispetto a *SEMP*.

Nel terzo grafico vediamo il confronto tra i costi delle varie soluzioni: *OPT*, *SOL*, *SEMP* ed *LSB*. Abbiamo inserito tutte le soluzioni per mostrare i vari andamenti al crescere della griglia. Come possiamo notare dal grafico, la qualità delle soluzioni trovate dagli algoritmi *OPT*, *SOL*, e *SEMP* ha un andamento simile, invece *LSB* si trova nettamente al disotto della soluzione *OPT*, come ci aspettavamo. La dimensione della griglia è limitata dal fatto che per calcolare la soluzione *OPT* sono necessari tempi computazionali molto grandi. Quindi non abbiamo potuto riportare la soluzione ottima con dimensioni maggiori di 36, comunque come possiamo notare le nostre soluzioni si avvicinano molto al costo ottimale. Nelle ascisse viene riportata la dimensione della griglia e nelle ordinate il valore della soluzione ovvero il costo di trasmissione in funzione della dimensione della griglia.

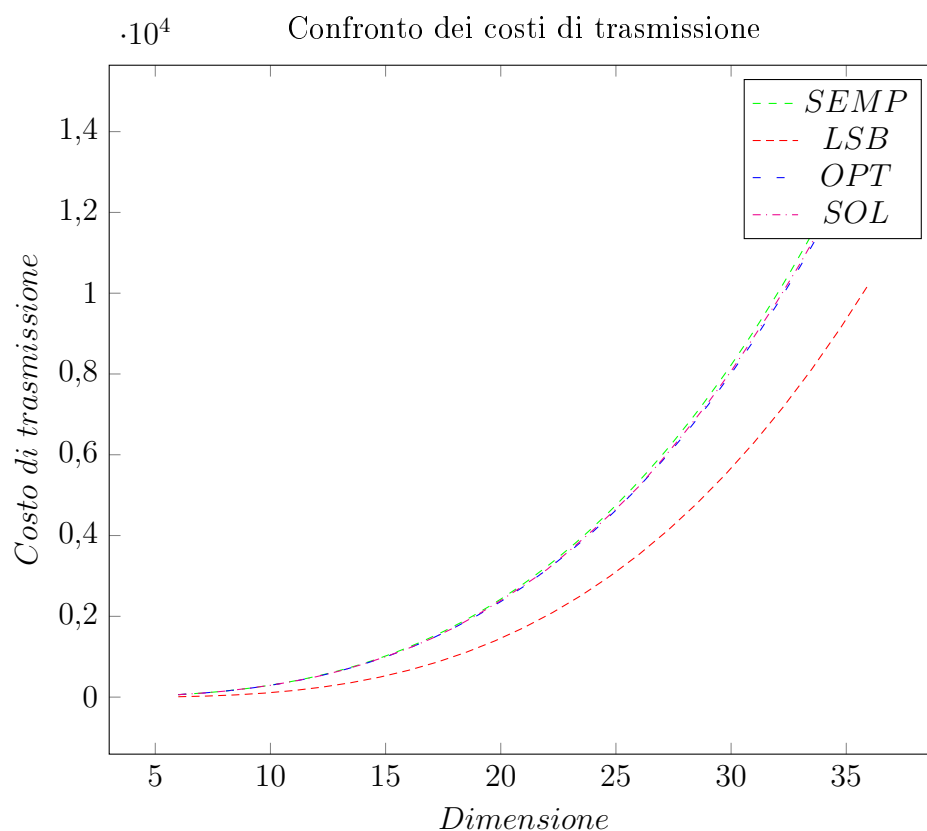


Figura 4.8: Grafico che mostra l'andamento delle varie soluzioni approssimate, in più è riportata anche la soluzione ottima

Dato che le soluzioni proposte sono molto vicine facciamo uno zoom su una parte del grafico, così possiamo vedere più in dettaglio il tutto. Notiamo che la soluzione SOL è sempre minore della soluzione $SEMP$ e che LSB è sempre minore di OPT . Questo risultato rispecchia la Disequazione 4.10, da ciò abbiamo una conferma in più del dato analitico calcolato precedentemente. Il grafico riportato, considera dimensioni della griglia nell'intervallo $[26, 30]$ e i loro rispettivi costi di trasmissioni compresi tra $[5000, 7000]$. Negli assi sono riportati il costo di trasmissione e la dimensione della griglia come nel grafico precedente.

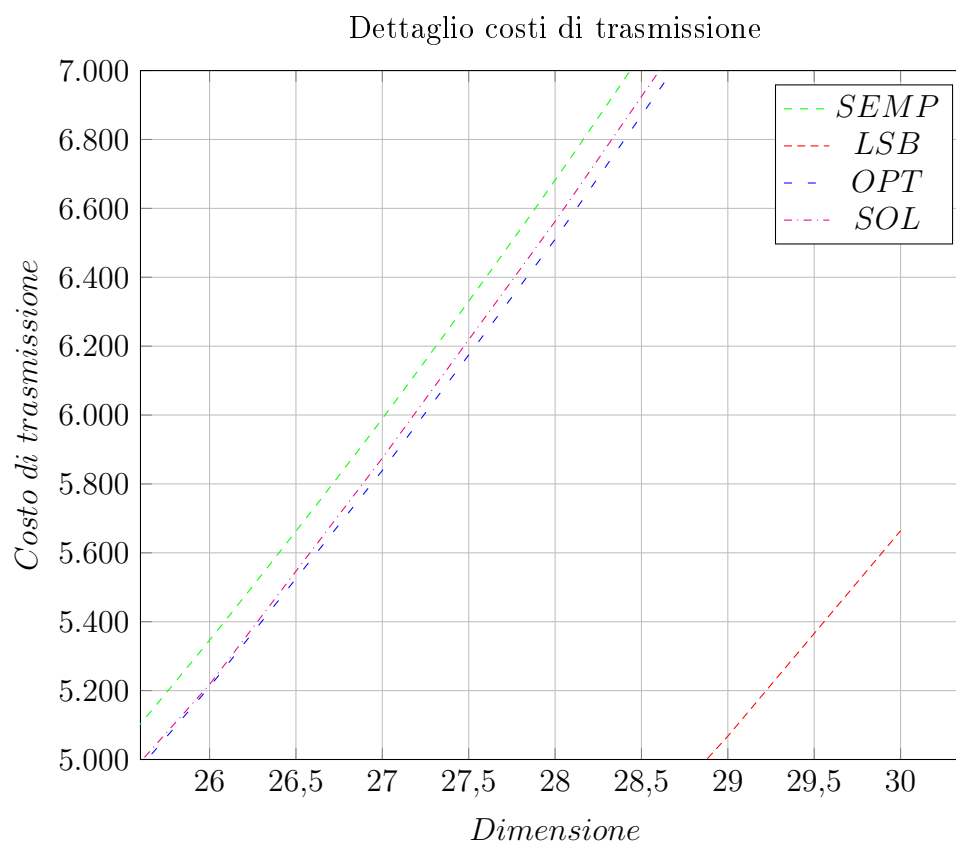


Figura 4.9: Grafico che mostra più dettagliatamente quanto le varie soluzioni approssimate si avvicinano alla soluzione ottima. Inoltre vediamo in dettaglio come OPT si trova tra LSB e le altre euristiche

Nel quarto grafico possiamo vedere il comportamento di *SEMP* e *LSB* con griglie molto grandi; abbiamo riportato solo queste due soluzioni in quanto anche *SOL* per input molto grandi richiede tempi computazionali elevati. Le due soluzioni hanno lo stesso comportamento asintotico come avevamo dimostrato analiticamente. Inoltre possiamo notare che, come nel grafico del rapporto, le due soluzioni rimangono molto vicine ed enfatizzano il concetto di imprigionamento della soluzione *OPT*. L'ascissa e l'ordinata hanno lo stesso significato del precedente grafico e la dimensione della griglia arriva a $1 \cdot 10^5$.

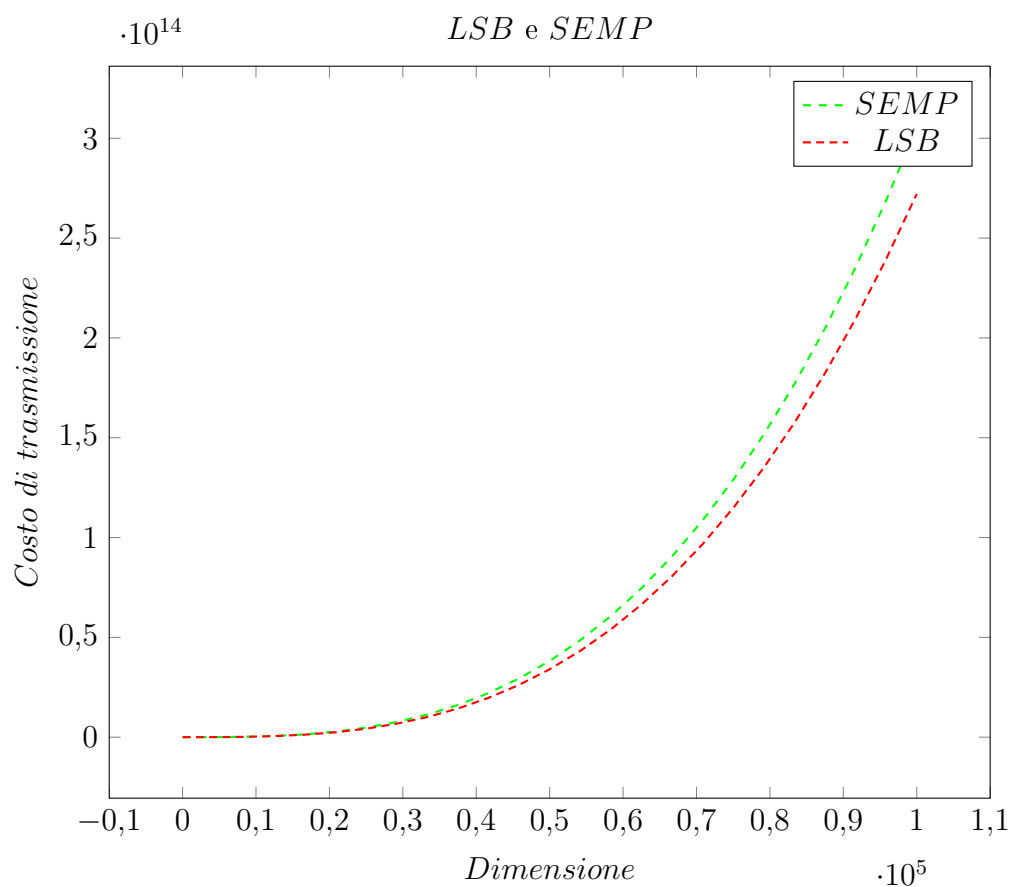


Figura 4.10: Grafico che mostra l'andamento di *SEMP* e *LSB* per griglie molto grandi.

In questa parte analizzeremo il costo delle varie soluzioni proposte. Sia il costo di *SEMP* che il costo di *LSB* sono $O(1)$ in quanto abbiamo una formula chiusa che ci calcola la soluzione. Invece come sopra analizzato il costo della soluzione *SOL* è $O(mn)$. Per concludere, il costo per calcolare la soluzione *OPT* è molto alto circa $(mn)^4$ se consideriamo griglie quadrate. La tabella sottostante riassume i costi analizzati.

Soluzione	Costo
<i>LSB</i>	$O(1)$
<i>SOL</i>	$O(mn)$
<i>SEMP</i>	$O(1)$
<i>OPT</i>	$O((mn)^4)$

Conclusioni

In questo lavoro abbiamo analizzato: la mediana di una griglia, il 2-median problem e il 3-median problem.

Nel primo capitolo abbiamo trovato la soluzione ottima per la mediana di una griglia rettangolare. Abbiamo notato che la mediana non è unica quando le dimensioni m , n della griglia non sono entrambe pari. Sempre nel primo capitolo abbiamo trovato la soluzione ottima per la mediana di una griglia rettangolare irregolare. Tutte le soluzioni viste in questo capitolo lavorano in tempo costante.

Nel secondo capitolo abbiamo trovato la soluzione ottima del 2-median problem in una griglia rettangolare tramite un algoritmo enumerativo che lavora in $O((mn)^3)$. Inoltre abbiamo sfruttato la soluzione proposta in letteratura per risolvere il 2-median problem con un costo computazionale pari a $O(n^3)$. Infine abbiamo analizzato una possibile soluzione che sfrutta la subottimalità del problema, quindi prova tutte le possibili sottogriglie, ne trova la mediana e sceglie la suddivisione ottimale.

Nel terzo capitolo abbiamo trovato la soluzione ottima del 3-median problem in una griglia rettangolare tramite un algoritmo enumerativo che lavora in $O((mn)^4)$. Dato che questa soluzione è troppo dispendiosa allora abbiamo trovato due euristiche che approssimano la soluzione. La prima euristica, *SEMP*, lavora in tempo costante, in quanto fissa tre regioni distinte e per ogni regione calcola la mediana. La seconda euristica, *SOL*, invece inizialmente partiziona i nodi della griglia rispetto a tre posizioni predefinite. Tale partizione individua tre regioni. In ciascuna regione, la mediana è poi raffinata e il costo della soluzione rispetto alle nuove mediane è restituito. Il costo computazionale di *SOL* è $O(mn)$.

Per garantire la qualità delle soluzioni trovate, si è valutato il rapporto di approssimazione fra il costo dell'euristica e il costo OPT della soluzione ottima. La soluzione OPT è sostituita per matrici con dimensioni molto grandi da un lower bound LSB , calcolato sfruttando la tecnica della tassellazione della griglia in rombi con costi di trasmissione minima. Il calcolo di LSB richiede costo computazionale $O(1)$.

E' stato provato quindi che entrambe le soluzioni euristiche garantiscono un costo di trasmissione non più alto di $K = 5.1$ volte il costo di trasmissione della soluzione ottima. Infine abbiamo confrontato sperimentalmente le varie soluzioni proposte con i dati sperimentali.

In conclusione il lavoro svolto ha riportato risultati originali in tutti e tre i problemi trattati.

Bibliografia

- [1] M.J. Golin and Y. Zhang, Two-median Problem on Manhattan Meshes, *Networks*, Vol.49, 3, 2007, pp. 226–233
- [2] T. Yamaguchi and I. Kaji, Some Location Problems on Grid Graphs, *Electronics and Communications in Japan*, Vol. 70, 10, 1987, pp. 31–40
- [3] F.C.M. Lau, P.K.W. Cheng AND S.S.H. TSE Aan algorithm for the two median problem on two-dimensional meshes, *The computer journal*, Vol. 44, No. 2,2001, pp. 101–108