IL PROBLEMA DEL BROADCAST CON IL MINIMO DISPENDIO DI ENERGIA OVVERO IL MINIMO ALBERO RICOPRENTE

Prof. Tiziana Calamoneri

Corso di Algoritmi per le reti

A.A. 2012/13

Il Problema

Il Problema (1)

- Una rete senza fili *ad-hoc* consiste di un insieme S di stazioni radio (fisse) collegate da connessioni wireless.
- Supponiamo che le stazioni siano nel piano euclideo (hp solo parzialmente realistica).
- I nodi hanno antenne omnidirezionali: ogni trasmissione è ascoltata da tutto il vicinato (broadcast naturale)
- …

IL PROBLEMA (2)

Che vuol dire "sufficientemente vicine"…

- Due stazioni comunicano direttamente (single-hop) se sono sufficientemente vicine o attraverso nodi intermedi (multi-hop).
- Ad ogni stazione è assegnato un raggio di trasmissione: un *range assignment* $r : S \to R$ determina un grafo delle comunicazioni diretto $G=(S,E)$ dove l'arco $(i, j) \in E$ sse $\text{dist}(i, j) \leq r(i)$ (con $\text{dist}(i, j)=$ distanza euclidea tra i e j).
- In altre parole, $(i, j) \in E$ sse j è nel disco di raggio $r(i)$ centrato in i.

Il Problema (3)

- Per ragioni legate alla conservazione dell'energia, ogni stazione può dinamicamente modulare la sua potenza trasmissiva.
- In effetti, il raggio di trasmissione di una stazione dipende dalla potenza energetica che la stazione stessa ha a disposizione.

IL PROBLEMA (4)

- In particolare, la potenza P_s richiesta da una stazione s per trasmettere ad un'altra stazione t deve soddisfare:

$$\frac{P_s}{dist(s,t)^{\alpha}} \geq 1$$

dove $\alpha \geq 1$ è detto gradiente distance-power

Di solito $2 \leq \alpha \leq 4$ (dipendentemente dall'ambiente)

Nello spazio vuoto $\alpha = 2$

- Quindi la potenza per avere una comunicazione da s a t è proporzionale a $dist(s,t)^{\alpha}$

Il Problema (5)

- Le stazioni di una rete ad-hoc collaborano per garantire delle specifiche proprietà di connettività adattando il loro raggio trasmissivo.
- …

IL PROBLEMA (6)

- … A seconda della proprietà richiesta abbiamo diversi problemi. Alcuni esempi:
  - il grafo delle trasmissioni deve essere fortemente connesso. In tal caso il problema è NP-hard ed esiste un algoritmo 2-approssimante in 2-dim. [Kirousis, Kranakis, Krizanc, Pelc '01], ed esiste $r>1$ tale che il problema non è r-approssimante
  - il grafo delle trasmissioni ha diametro al più un fissato h. Non sono noti risultati di approssimazione non banali.

IL PROBLEMA (7)

Un'altra proprietà richiesta: dato un nodo sorgente s, il grafo delle connessioni deve contenere un albero ricoprente radicato in s.

- Un Broadcast Range Assignment (in breve semplicemente *Broadcast*) è un assegnamento dei raggi che permette al grafo di comunicazione G di contenere tale albero ricoprente

Il Problema (8)

- Un problema fondamentale nella progettazione delle reti ad-hoc è il problema del Broadcast con il minimo dispendio di energia (in breve *Min Broadcast*), che consiste nel trovare un broadcast di minima energia complessiva

IL PROBLEMA (9)

Th. *Min Broadcast* non è approssimabile entro un fattore costante.

Dim. Ricordiamo il problema *MinSetCover*:

data una collezione C di sottinsiemi di un insieme finito S, trovare un sottinsieme di C, C', di minima cardinalità tale che ogni elemento di S appartiene ad almeno un elemento di C'.

Esempio:

$S=\{1,2,3,4,5\}$ $C=\{\{1,2\}, \{1,2,3\}, \{3\}, \{3,4,5\}\}$

$C'=\{\{1,2,3\},\{3,4,5\}\}$

Il Problema (10)

Segue dim.

N.B. *MinSetCover* è non approssimabile entro $c \log n$ per qualche costante $c>0$, dove $n=|S|$.

Data un'istanza x di *MinSetCover* è possibile costruire un'istanza y di *MinBroadcast* tale che esiste una soluzione per x di cardinalità k sse esiste una soluzione per y di costo $k+1$.

Così, se *MinBroadcast* è approssimabile entro una cost. allora anche *MinSetCover* lo è. Assurdo.

IL PROBLEMA (11)

Segue dim. Riduzione:

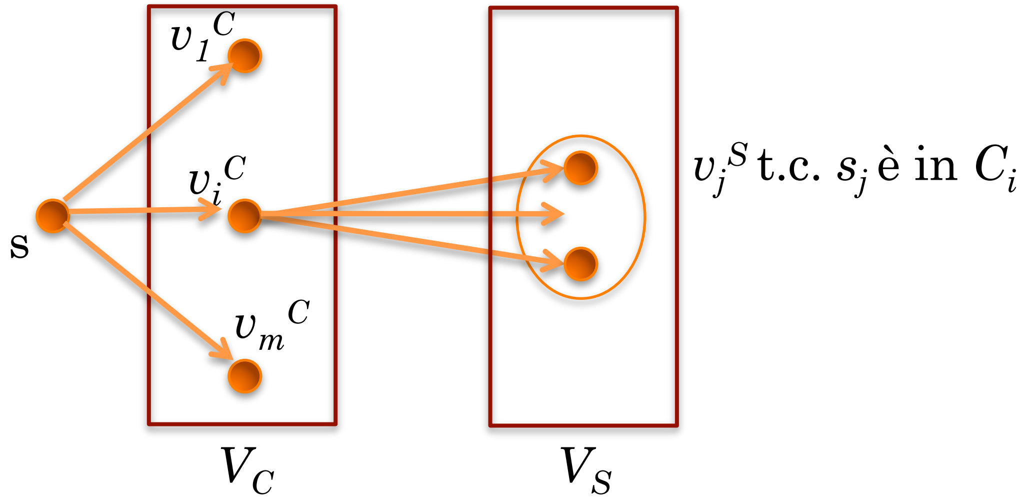
x=(S,C) istanza di *MinSetCover* dove:

$$S=\{s_1, s_2, \ldots, s_n\} \text{ e } C=\{C_1, C_2, \ldots, C_m\}.$$

Costruiamo $y=(G,w,s)$ di *MinBroadcast*.

Nodi di G: $\{s\}\ U\ \{V_C\}\ U\ \{V_S\}$

Archi di G:$\{(s, v_i^C), 1\leq i\leq m\} U \{(v_i^C, v_j^S)$ t.c. s_j in $C_i\}$



Il Problema (12)

Segue dim.

Infine, definiamo $w(e)=1$ per ogni arco e.

Sia C' una sol. per x.

Una sol. per y assegna 1 ad s e a tutti i nodi di V_C che stanno in C'.

Tale sol. contiene un albero ricoprente perché ogni elemento di S è contenuto in almeno un elemento di C'. Il costo di tale sol. è $|C'|+1$.

Il Problema (15)

Segue dim.

…

Viceversa, se r è una sol. per y, wlog $r(v)$ vale 0 o 1 se v è in V_C (altri valori non avrebbero senso) ed $r(v)=0$ se v è in V_S.

Si ottiene una sol. C' per x selezionando tutti i sottinsiemi C_i t.c. $r(v_i^C)=1$ e $|C'|=cost(r)-1$.

CVD

IL PROBLEMA (16)

Attenzione

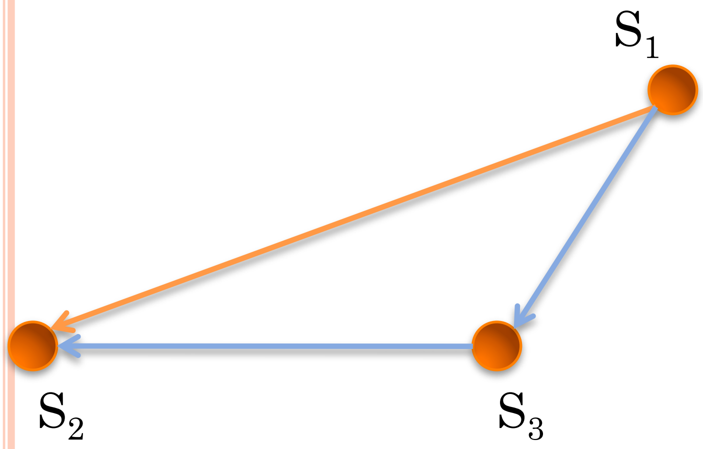
E' vero che abbiamo dimostrato che *Min Broadcast* non è approssimabile entro un fattore costante, ma abbiamo parlato del problema generale.

Ci sono casi (ad esempio quello bidimensionale euclideo), che ci interessano molto, che si comportano meglio!

Restringiamoci quindi a questo caso particolare...

Il Problema (17)

- La collaborazione per minimizzare l'energia complessiva è fondamentale:



  - S_1 deve comunicare con S_2
  - sia $\alpha=2$
  - costo di $S_1 \rightarrow S_2 = dist(S_1, S_2)^2$
  - costo di $S_1 \rightarrow S_3 \rightarrow S_2 =$ $dist(S_1, S_3)^2 + dist(S_3, S_2)^2$
  - quando l'angolo $S_1S_3S_2$ è ottuso: $dist(S_1, S_2)^2 >$ $dist(S_1, S_3)^2 + dist(S_3, S_2)^2$

IL PROBLEMA (18)

- Nel caso euclideo, un range assignment r può essere rappresentato dalla corrispondente famiglia $D = \{D_1, \ldots, D_l\}$ di dischi, e l'energia complessiva è definita:

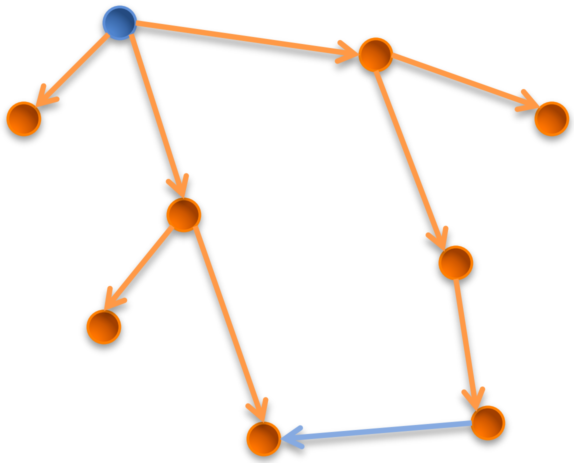
$$\mathrm{cost}(D) = \sum_{i=1}^{l} r_i^{\alpha}$$

dove r_i è il raggio di D_i.

IL PROBLEMA (19)

- Si consideri il grafo completo pesato $G^{(\alpha)}$ in cui il peso di un arco $e=(u,v)$ è $dist(u,v)^{\alpha}$.
- Il problema del broadcast è in stretta relazione con il minimo albero ricoprente, poiché ne ha alcune importanti proprietà: …

IL PROBLEMA (20)

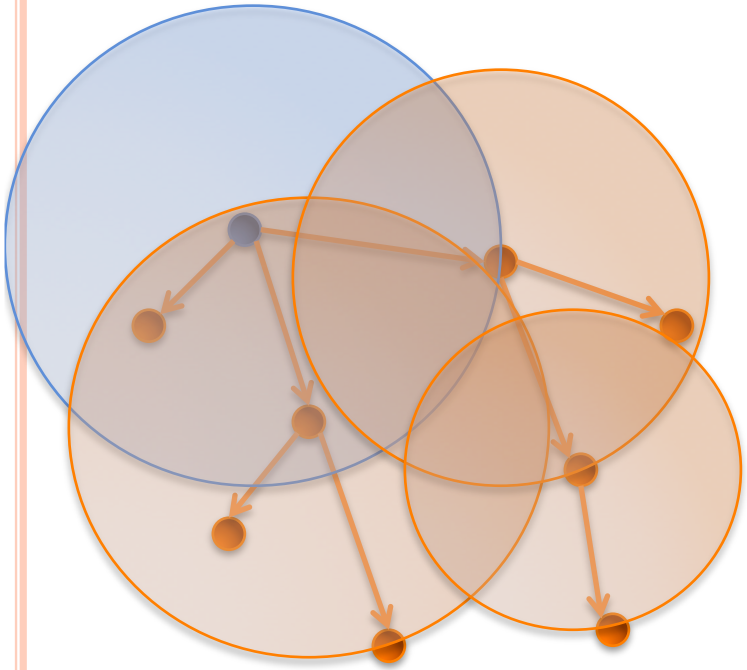


Insieme delle connessioni usate per informare da *s*:

- Non potrà generare un ciclo, poiché il nodo su cui il ciclo si richiude è già stato informato: albero
- Un criterio per minimizzare l'energia: le connessioni lunghe usano maggiore energia delle corte

IL PROBLEMA (21)

- Tuttavia, il problema del Minimum Broadcast non è equivalente al Min Spanning Tree:



- L'energia spesa da ogni nodo u è pari a

$$\max_{(u,v)\in T}\{dist(u,v)\}^{\alpha}$$

(cioè non tutti gli archi contribuiscono)

- Le foglie spendono energia nulla

Il Problema (22)

- Il problema del Minimum Broadcast nella sua versione generale è NP-hard e non è approssimabile in meno di $(1-\varepsilon)\Delta$ dove Δ è il max grado di T ed ε è una costante arbitraria
- Per la versione geometrica (la nostra) non si sa nulla!

Il Problema (23)

- Un algoritmo di approssimazione è basato sul calcolo del MST (minimo albero ricoprente):
  - calcola il MST del grafo completo indotto da *S*,
  - assegna una direzione agli archi dalla sorgente *s* alle foglie
  - assegna ad ogni nodo *i* un raggio pari alla lunghezza dell'arco più lungo che esce da i
- Facile da implementare, quindi analisi del rapporto di approssimazione soggetta a molto lavoro negli ultimi anni.
  - Il primo rapporto costante (circa 40) [Clementi+al.'01]
  - miglior rapporto attuale (stretto) 6 [Ambüehl '05]

Il Problema del Minimo Albero Ricoprente (ripasso)

Minimo Albero Ricoprente (1)

Tre algoritmi classici:

- Algoritmo di Kruskal [‘56]
- Algoritmo di Prim [‘57]
- Algoritmo di Boruvka [’26]

Minimo Albero Ricoprente (2)

- Tutti e tre gli algoritmi greedy sono basati sullo stesso algoritmo generico:
  - Dato un insieme A che contiene alcuni archi del MST di G, e è un arco sicuro per A se $A \cup e$ contiene ancora solo archi del MST.
  - A=insieme vuoto

    While A non è un MST

      trova un arco e sicuro per A parte "difficile"

      $A = A \cup e$

Minimo Albero Ricoprente (3)

- A=insieme vuoto
  while A non è un MST
  trova un arco e sicuro per A
  $A=A \cup e$

In ogni momento:

- A è aciclico
- il grafo $G_A=(V, A)$ è una foresta in cui ogni componente è un nodo o un albero
- ogni arco sicuro collega componenti separate di G_A
- Il ciclo while viene eseguito $n-1$ volte

Algoritmo di Kruskal (1)

- A=insieme vuoto
  While A non è un MST
    trova un arco e sicuro per A
    $A=A \cup e$

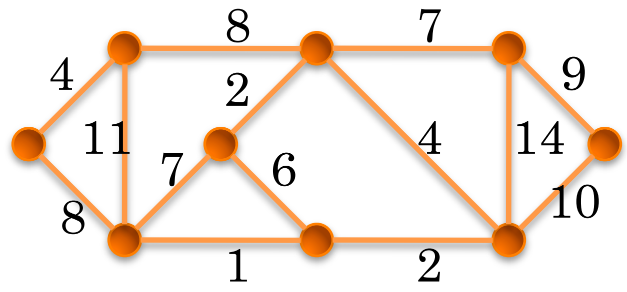
tra quelli che collegano due componenti diverse di G_A, quello di peso minimo

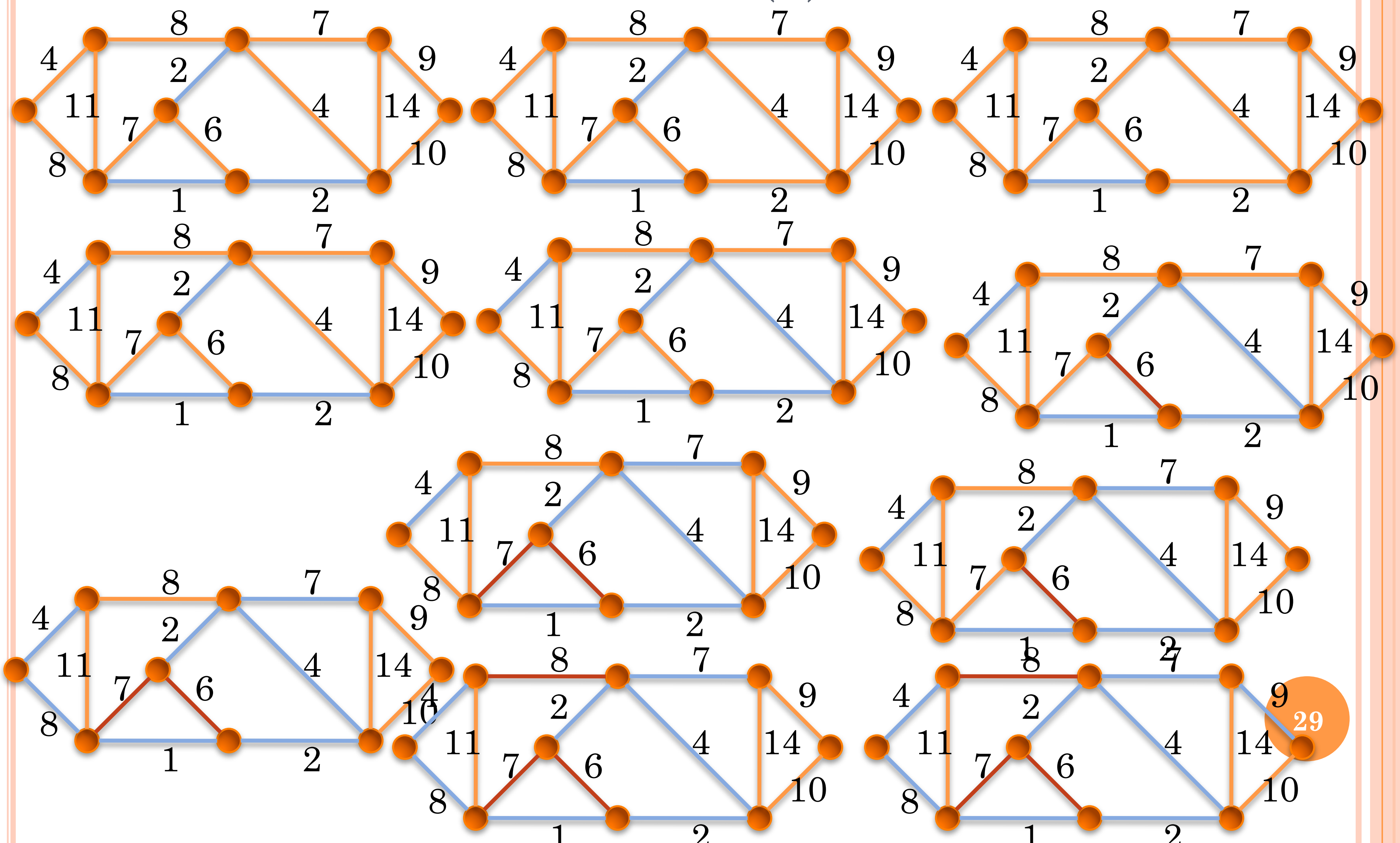
Implementazione tramite:

- struttura Union-Find
- Insieme degli archi di G ordinati per peso crescente
- Complessità: $O(m \log n)$

[Johnson '75, Cheriton & Tarjan '76]

Algoritmo di Kruskal (2)





ALGORITMO DI PRIM (1)

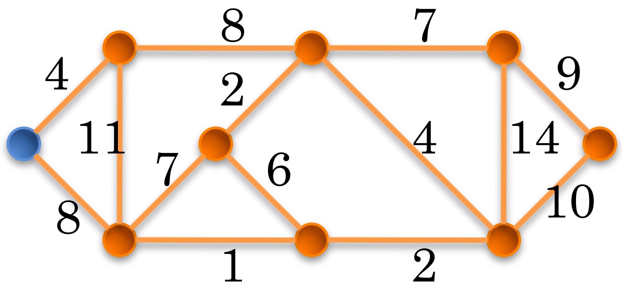
- A=insieme vuoto
  While A non è un MST
  trova un arco e sicuro per A
  $A=A \cup e$

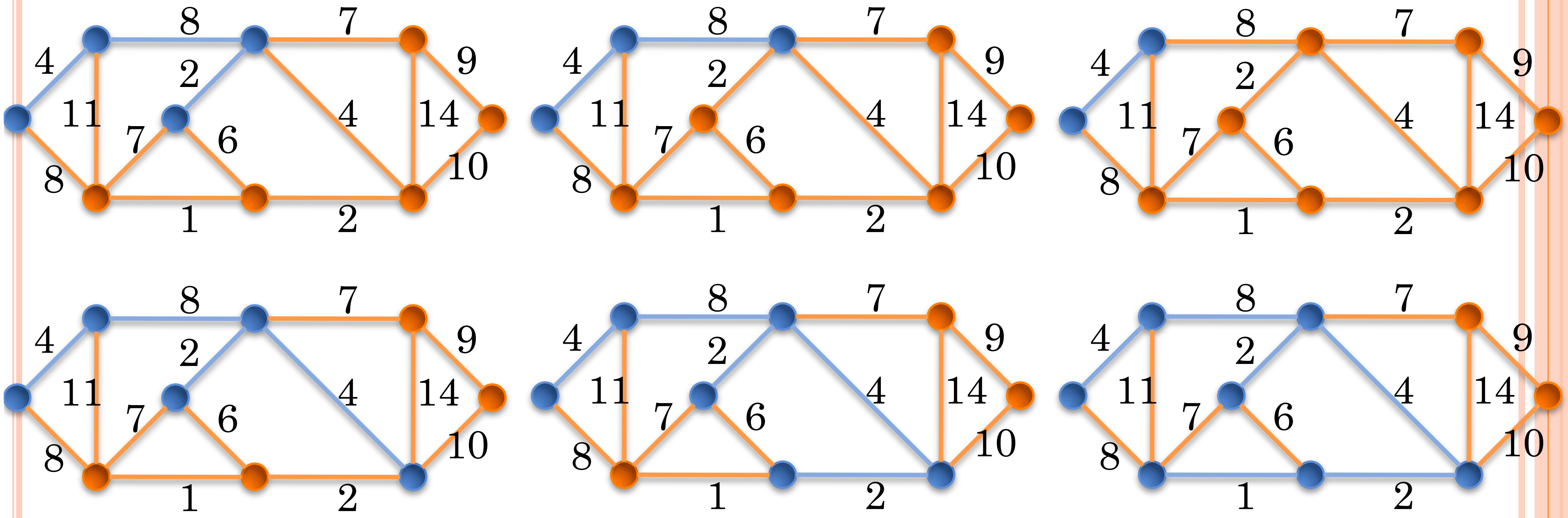
tra quelli che collegano la componente principale con un nodo isolato, quello di peso minimo

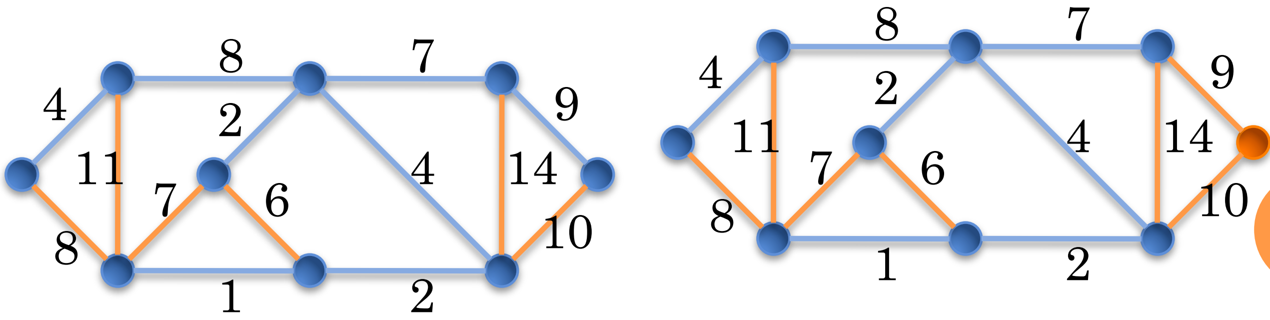
Implementazione tramite:

- Nodi in una coda con min-priorità basata sul campo $key(v)$=min peso di un arco che collega v ad un nodo della comp. principale; ∞ se non esiste
- Se coda=heap Complessità: $O(m \log n)$
- Se coda= heap di Fibonacci Complessità: $O(m+n \log n)$ [Ahuja, Magnanti & Orlin '93]

ALGORITMO DI PRIM (2)







ALGORITMO DI BORUVKA (1)

tra quelli che collegano C_i ad un'altra componente, quello di peso minimo

Ipotesi: tutti i pesi distinti

- A=insieme vuoto

  While A non è un MST

  per ogni componente C_i di A

  trova un arco e_i sicuro per C_i

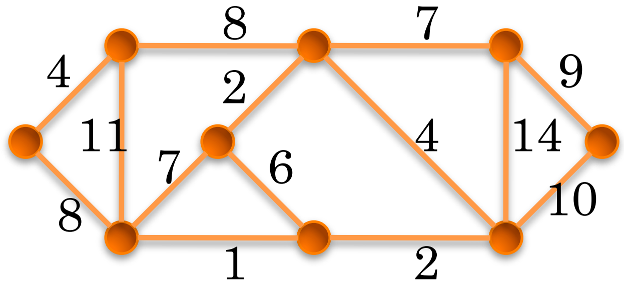
  $A=A \cup \{e_i$, per ogni $i\}$

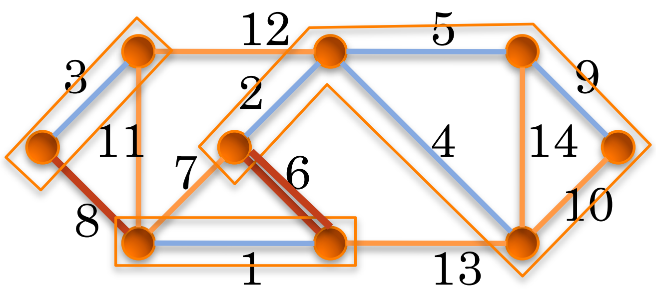
Trucco: trattare tanti (pari al log del # di componenti) archi durante la stessa iterazione

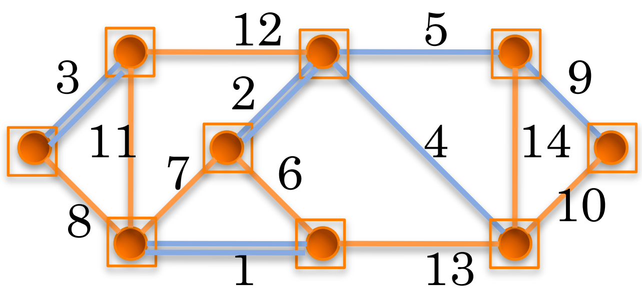
Garanzia di non introdurre cicli grazie all'ipotesi!

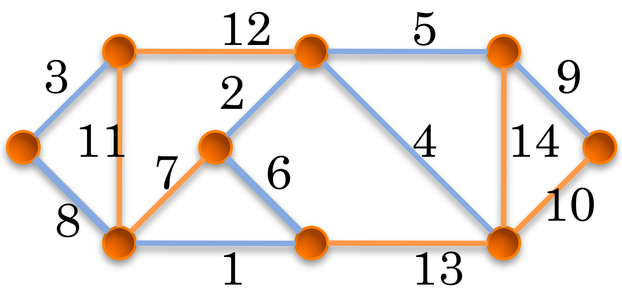
Complessità: $O(m \log n)$

Algoritmo di Boruvka (2)









ALTRI ALGORITMI (1)

- Un algoritmo con complessità lineare $O(n+m)$ [Friedman & Willard '94], ma suppone che gli archi siano già ordinati per peso. Non si usa in pratica perché la notazione asintotica nasconde una costate moltiplicativa molto alta
- Un algoritmo con complessità lineare $O(n)$ per grafi planari [Matsui '95] -> TESINA

ALTRI ALGORITMI (2)

- E' anche stato studiato il problema di trovare un nuovo minimo albero ricoprente se ne conosciamo già uno e cambiamo il peso di un arco [Frederickson '85, Eppstein '94]. Una modifica si può fare in tempo medio $O(\log n)$
- Si può verificare in tempo lineare $O(n+m)$ se un dato albero ricoprente è minimo.

Ancora Broadcast con Minimo Dispendio di Energia

EURISTICHE (1)

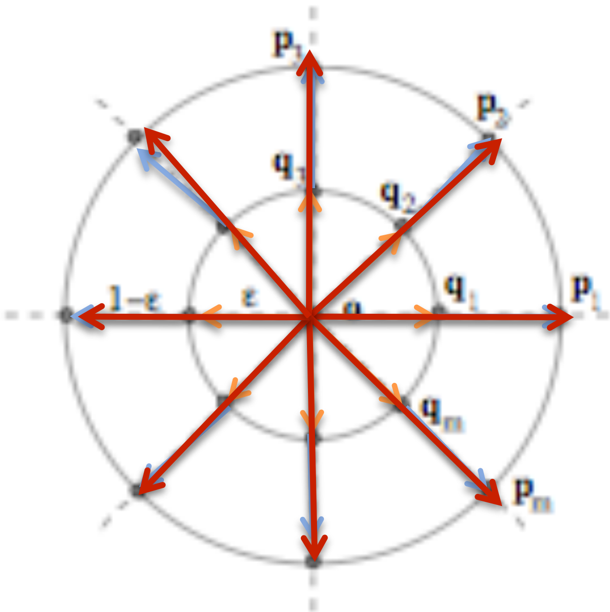
Tre euristiche proposte in [Wieselthier, Nguyen, Ephremides, 00] tutte basate sulla tecnica greedy:

- Euristica MST (min spanning tree): applica l'algoritmo di Prim per ottenere un min albero ricoprente e poi orienta l'albero dalla radice verso le foglie
- Euristica SPT (spanning path tree): applica l'algoritmo di Dijkstra per ottenere un albero dei cammini minimi dalla sorgente e poi orienta l'albero dalla radice verso le foglie
- Euristica BAIP (Broadcast Average Incremental Power): versione basata sui nodi dell'algoritmo di Dijkstra (si aggiungono nuovi nodi all'albero sulla base del minor costo medio)

EURISTICHE (2)

Il greedy non sempre funziona bene:

- Euristica SPT: applica l'algoritmo di Dijkstra per ottenere un albero dei cammini minimi dalla sorgnete e poi orienta l'albero dalla radice



(sia α=2)

- SPT trova un albero con dispendio di energia:

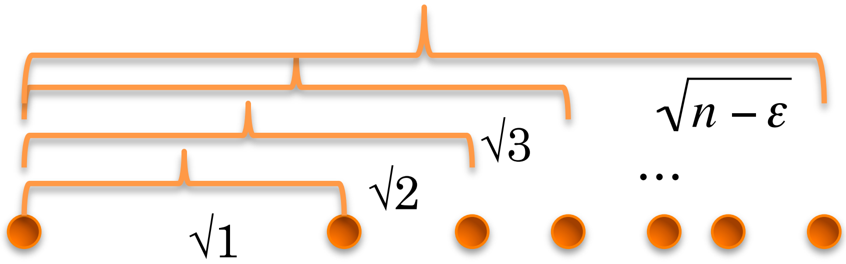
  $\varepsilon^2+n/2(1-\varepsilon)^2$

- Se la radice trasmette con raggio 1 il dispendio è 1
- Se $\varepsilon \rightarrow 0$ SPT sbaglia di n/2

EURISTICHE (3)

- Euristica BAIP (Broadcast Average Incremental Power): si aggiungono nuovi nodi all’albero sulla base del minor costo medio=aumento di energia/# nodi aggiunti
- ideato per ovviare al problema precedente

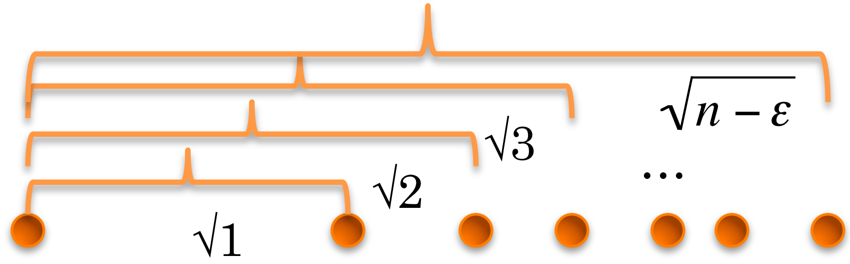
EURISTICHE (4)



Il greedy non sempre funziona bene (sia α=2):

- la min potenza per la sorgente per raggiugere il nodo k è $\sqrt{k^2}=k$ e la media dell'energia aggiunta è quindi $k/k=1$
- la min potenza per la sorgente per raggiungere tutti i nodi è $(\sqrt{n-\varepsilon})^2=n-\varepsilon$ e quindi la media è $(n-\varepsilon)/n=1-\varepsilon/n$…

EURISTICHE (5)



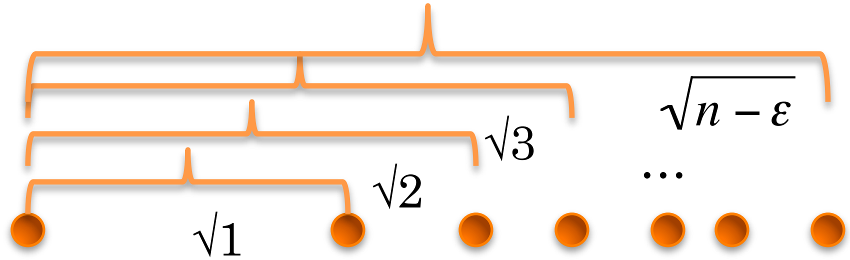
- BAIP sceglie la sol. in cui la sorgente trasmette con raggio $\sqrt{n}$- ε
- La soluzione costituita dal cammino ha aggiunta media di energia peggiore, ma il consumo energetico è:

$$\sum_{i=1}^{n-1}(\sqrt{i}-\sqrt{i-1})^2+(\sqrt{n-\varepsilon}-\sqrt{n-1})^2<\sum_{i=1}^{n}(\sqrt{i}-\sqrt{i-1})^2=$$

$$\sum_{i=1}^{n}(\sqrt{i}-\sqrt{i-1})^2\frac{(\sqrt{i}+\sqrt{i-1})^2}{(\sqrt{i}+\sqrt{i-1})^2}=\sum_{i=1}^{n}\frac{((\sqrt{i}-\sqrt{i-1})(\sqrt{i}+\sqrt{i-1}))^2}{(\sqrt{i}+\sqrt{i-1})^2}=$$

$$=\sum_{i=1}^{n}\frac{(i-(i-1))^2}{(\sqrt{i}+\sqrt{i-1})^2}=\sum_{i=1}^{n}\frac{1}{(\sqrt{i}+\sqrt{i-1})^2}=1+\sum_{i=2}^{n}\frac{1}{(\sqrt{i}+\sqrt{i-1})^2}\leq$$

EURISTICHE (6)



(segue calcolo dell'errore che può fare BAIP)

$$\leq 1+\sum_{i=2}^{n}\frac{1}{i+(i-1)+2\sqrt{i}\sqrt{i-1}} \leq 1+\sum_{i=2}^{n}\frac{1}{2i-1+2(i-1)} \leq$$

$$\leq 1+\sum_{i=2}^{n}\frac{1}{2i-1+2(i-1)} = 1+\sum_{i=2}^{n}\frac{1}{4i-3} \leq 1+\sum_{i=2}^{n}\frac{1}{4(i-1)} \leq$$

ponendo i=j+1:

$$\leq 1+\sum_{j=1}^{n-1}\frac{1}{4j} \leq 1+\frac{1}{4}\sum_{j=1}^{n-1}\frac{1}{j} \leq 1+\frac{1}{4}(\ln(n-1)+1) = \frac{\ln(n-1)+5}{4}$$

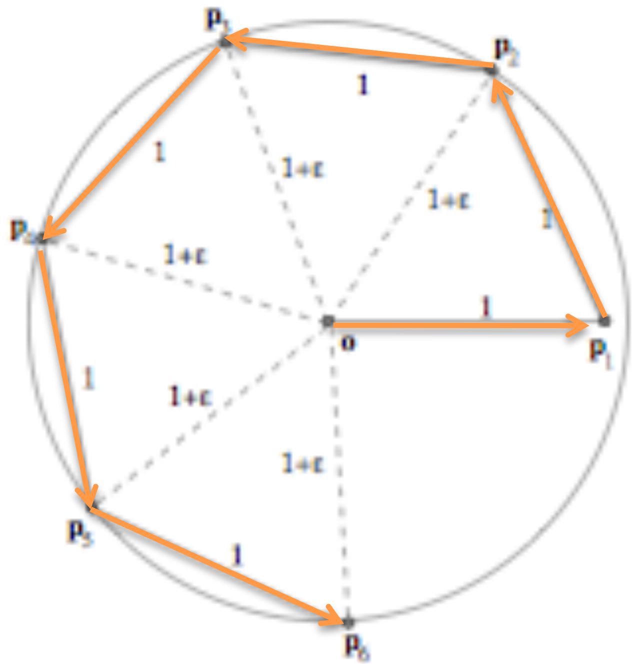
quindi la frazione con cui può sbagliare BAIP è:

$$\frac{n-\varepsilon}{\frac{\ln(n-1)+5}{4}} \rightarrow (\varepsilon \rightarrow 0)\frac{4n}{\ln(n-1)+5} = \frac{4n}{\ln n}+o(1)$$

EURISTICHE (7)

Il greedy non sempre funziona bene:

- Euristica MST: applica l'algoritmo di Prim per ottenere un min albero ricoprente e poi orienta l'albero dalla radice verso le foglie



- MST trova un albero con dispendio di energia 6
- Se la radice trasmette con raggio $1+\varepsilon$ il dispendio è $(1+\varepsilon)^{\alpha}$
- Il rapporto è 6, se ε tende a 0

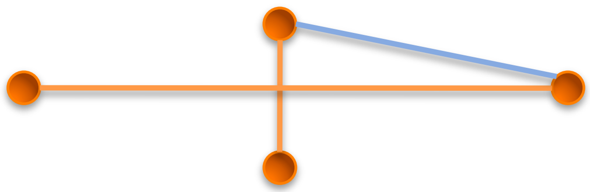
Euristiche (8)

- E' stato dimostrato [Wan, Calinescu, Li, Frieder '02] che MST ha un rapporto di approssimazione costante e tale rapporto, che è almeno 6, non è più di 12.
- Il limite sup. di 12 si dimostra sfruttando alcune proprietà del MST geometrico:
  - …

EURISTICHE (9)

- nessuna coppia di archi di un MST si incrocia

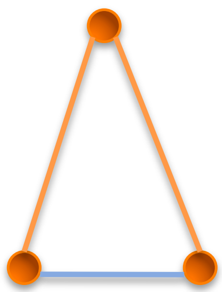
L'arco azzurro è necessariamente più corto di uno dei due archi che si incrociano



EURISTICHE (10)

(segue proprietà del MST geometrico)

- l'angolo tra ogni coppia di archi incidenti di un MST è almeno π/3



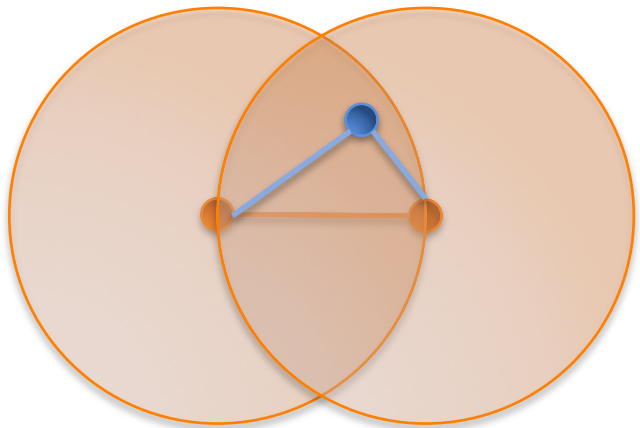
L'arco azzurro è necessariamente più corto di uno dei due archi arancioni

EURISTICHE (11)

(segue proprietà del MST geometrico)

- la luna costituita da ogni arco non contiene nodi

La luna è il luogo dei punti con dist. $<dist(P_1,P_2)$ sia da P_1 che da P_2, quindi un punto interno formerebbe un ciclo

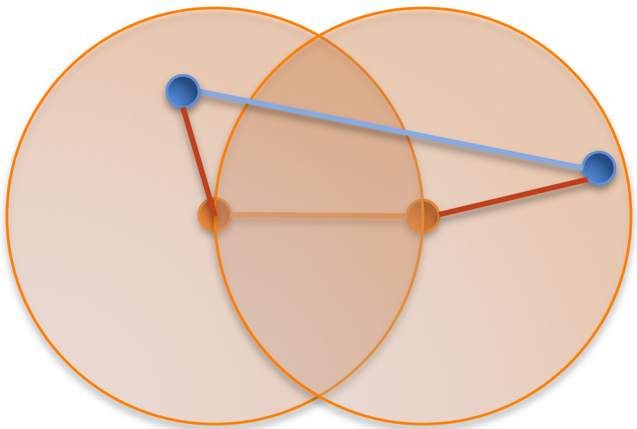


EURISTICHE (12)

(segue proprietà del MST geometrico)

- per ogni arco P_1P_2, ogni altro arco ha o entrambi gli estremi fuori del disco aperto $D(P_1, dist(P_1, P_2))$ oppure entrambi fuori del disco aperto $D(P_2, dist(P_1, P_2))$

Prima dell'arco azzurro sono stati inseriti gli archi rossi perché più brevi, e quindi si formerebbe un ciclo



EURISTICHE (13)

- La dim. in [Wan, Calinescu, Li, Frieder '02] contiene un piccolo errore, correggendo il quale il rapporto di approssimazione diventa 12,15 [Klasing, Navarra, Papadopoulos, Perennes '04]
- Indipendentemente, rapporto di approssimazione 20 [Clementi, Crescenzi, Penna, Rossi, Vocca '01]
- Rapporto migliorato a 7,6 [Flammini, Klasing, Navarra, Perennes '04]
- Rapporto migliorato a 6,33 [Navarra '05]
- Risultato ottimo 6 [Ambüehl '05]

EURISTICHE (14)

- Per istanze realistiche, le sperimentazioni suggeriscono che il corretto rapporto di approssimazione non sia 6 ma 4 [Flammini, Navarra, Perennes '06] -> TESINA