

Teorema: GREEDY LOAD BALANCING è un algoritmo 2-approssimato

Dim: Oss. 1  $L^* \geq \frac{1}{m} \sum_{j \in m} t_j$

Dim:  $\sum_{i \in m} L_i^* = \sum_{j \in m} t_j$

per pigeonholing, almeno per un  $i$   $L_i^* \geq \frac{1}{m} \sum_{j \in m} t_j$

$$\Rightarrow L^* = \max_{i \in m} L_i^* \geq \frac{1}{m} \sum_{j \in m} t_j$$

Oss. 2  $L^* \geq \max_i t_i$

Ovvio

Esempio GREEDY LOAD BALANCING.

$\leadsto L = L_{\hat{i}}$

sia  $t_{\hat{j}}$  l'ultimo task assegnato a  $\hat{i}$ .

$L_{\hat{i}} - t_{\hat{j}} \leq L_i' \leq L_i \quad \forall i \in m$

carico delle macchine nel momento dell'ass.

$$m(L_{\hat{i}} - t_j) \leq \sum_{i \in M} L_i = \sum_{j \in N} t_j$$

$$\Rightarrow L_{\hat{i}} - t_j \leq \frac{1}{m} \sum_{j \in N} t_j \stackrel{\text{oss. 1}}{\leq} L^*$$

$$L = L_{\hat{i}} = \underbrace{L_{\hat{i}} - t_j}_{\leq L^*} + \underbrace{t_j}_{\leq L^* \text{ oss. 2}} \leq 2L^*$$

$$\frac{L}{L^*} \leq 2$$

□

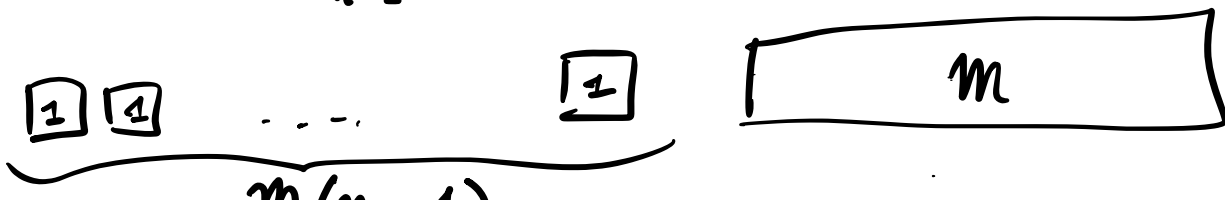
Teorema: Per ogni  $\varepsilon > 0$ , esiste un input su cui GREEDY LOAD BALANCING produce una soluzione

$$2 - \varepsilon \leq \frac{L}{L^*} \leq 2$$

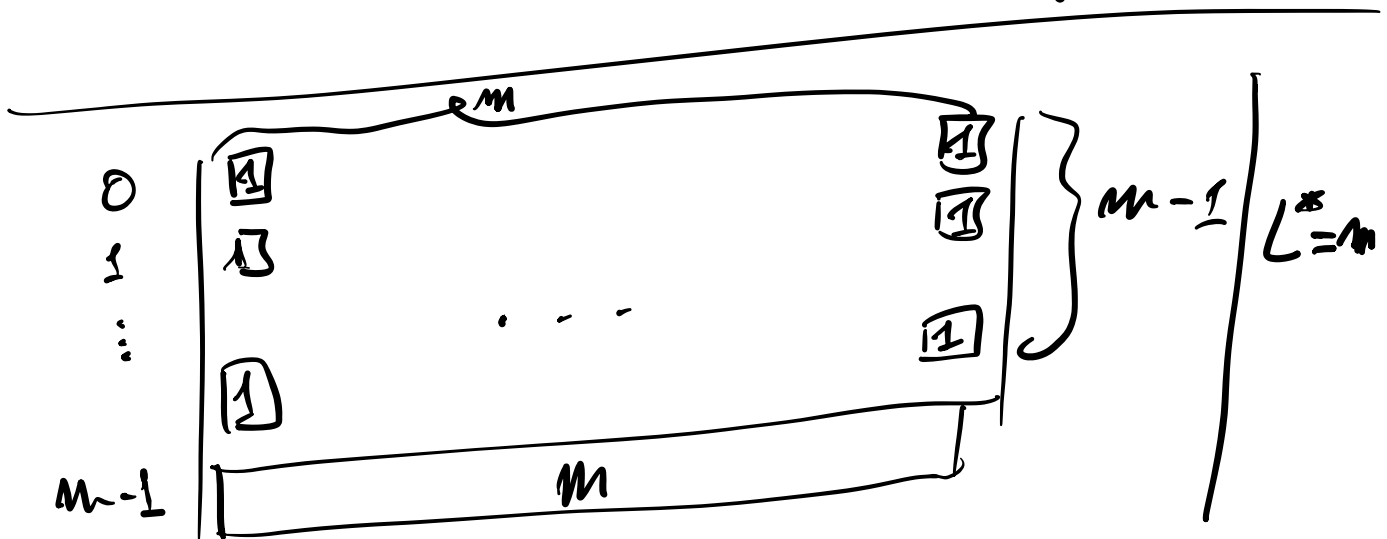
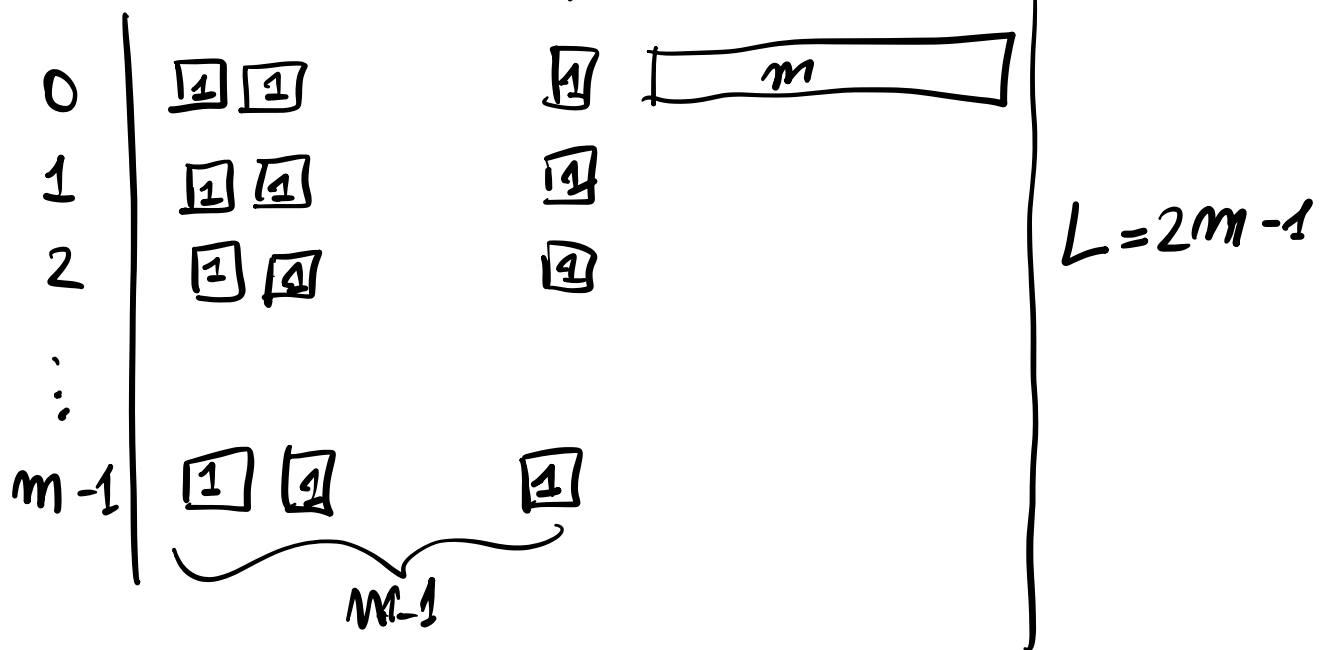
Dim: Sia  $m$  un intero  $m > \frac{1}{\varepsilon}$ .

Sia  $m = m(m-1) + 1$ .

Sia  $t_0, \dots, t_{m-2} = 1$   
 $t_{m-1} = m$



$$m(m-1)$$



$$\frac{L}{L^*} = \frac{2m-1}{m} = 2 - \frac{1}{m} \geq 2 - \varepsilon \quad \square$$

# ALGORITHMO

# SORTED GREEDY BALANCING

INPUT

- Ordinare i task in modo che  $t_0 \geq t_1 \geq \dots \geq t_{n-1}$
- Esegui GREEDY LOAD BALANCING

$O(n \log n + m \log m)$

Teorema: SORTED GREEDY BALANCING è una  $\frac{3}{2}$ -approssimazione.

Dim: CASO 1

Se  $n \leq m$ , troviamo la soluzione ottima.

CASO 2

$n > m$

$$L^* \geq 2t_m$$

$$t_0 \geq t_1 \geq \dots \geq t_{m-1} \geq t_m \geq \dots$$

$m+1$  task per pigeon holeing,  $\exists$  un task che è no due alle stesse macchine

$$L^* \geq L_i^* \geq 2t_m$$

Si  $\hat{i}$  la macchina con carico massimo.

- se ha avuto un solo assegnamento  $\Rightarrow$  OTTIMO

- se ha avuto  $\geq 2$  assegn.  
sia  $\hat{j}$  l'ultimo compito assegnato a  $\hat{i}$ .

$$\hat{j} \geq m$$

$$\Rightarrow t_{\hat{j}} \leq t_m \leq \frac{1}{2} L^*$$

$$L = L_{\hat{i}} = \underbrace{L_{\hat{i}} - t_{\hat{j}}}_{\leq L^*} + \underbrace{t_{\hat{j}}}_{\leq \frac{1}{2} L^*} \leq \frac{3}{2} L^*$$

$$\frac{L}{L^*} \leq \frac{3}{2} \quad \square$$

[Graham 1969] L'unica stretta mostra che è una  $\frac{4}{3}$ -approssimaz.

LOAD BALANCING  $\in$  PTAS [Hochbaum-Shmoys 1988]

se  $P \neq NP$ , LOAD BALANCING  $\notin$  FPTAS,

In questo il problema è fortemente NP-completo.

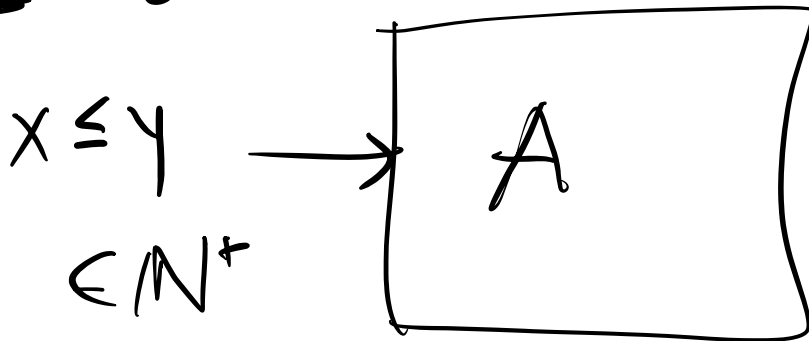
PSEUDO POLINOMIALITÀ E  
NP-COMPLETENZA FORTE

$$m = 2 \lg x + 2 \lg y = 2 \lg xy \Rightarrow \lg xy = \frac{m}{2}$$

$$xy = e^{\frac{m}{2}}$$

$$x \geq e^{\frac{m}{2}}$$

$$x \geq e^m$$



$$t_A = O(x + y) = O(e^m)$$

A è pseudo polinomiale

---

Problema di decisione  $\Pi$   
che è NP-completo

$x \in \mathbb{N} \rightarrow$  decidere se  $x \in \Pi$

è NP-completo

Se  $P \neq NP \Rightarrow$  NON AMMETTE  
SOLUZ. POLINOMIALI

Se  $\exists$  un algoritmo pseudopol.  
per  $\Pi$ , si dice che  
 $\Pi$  è debolmente NP-completo

PROBLEMA  
DEI CENTRI DELLA SELEZIONE  
(CENTER SELECTION)

SPAZIO METRICO

$\Omega$  insieme di punti  
 $d: \Omega \times \Omega \rightarrow \mathbb{R}$   
distanza

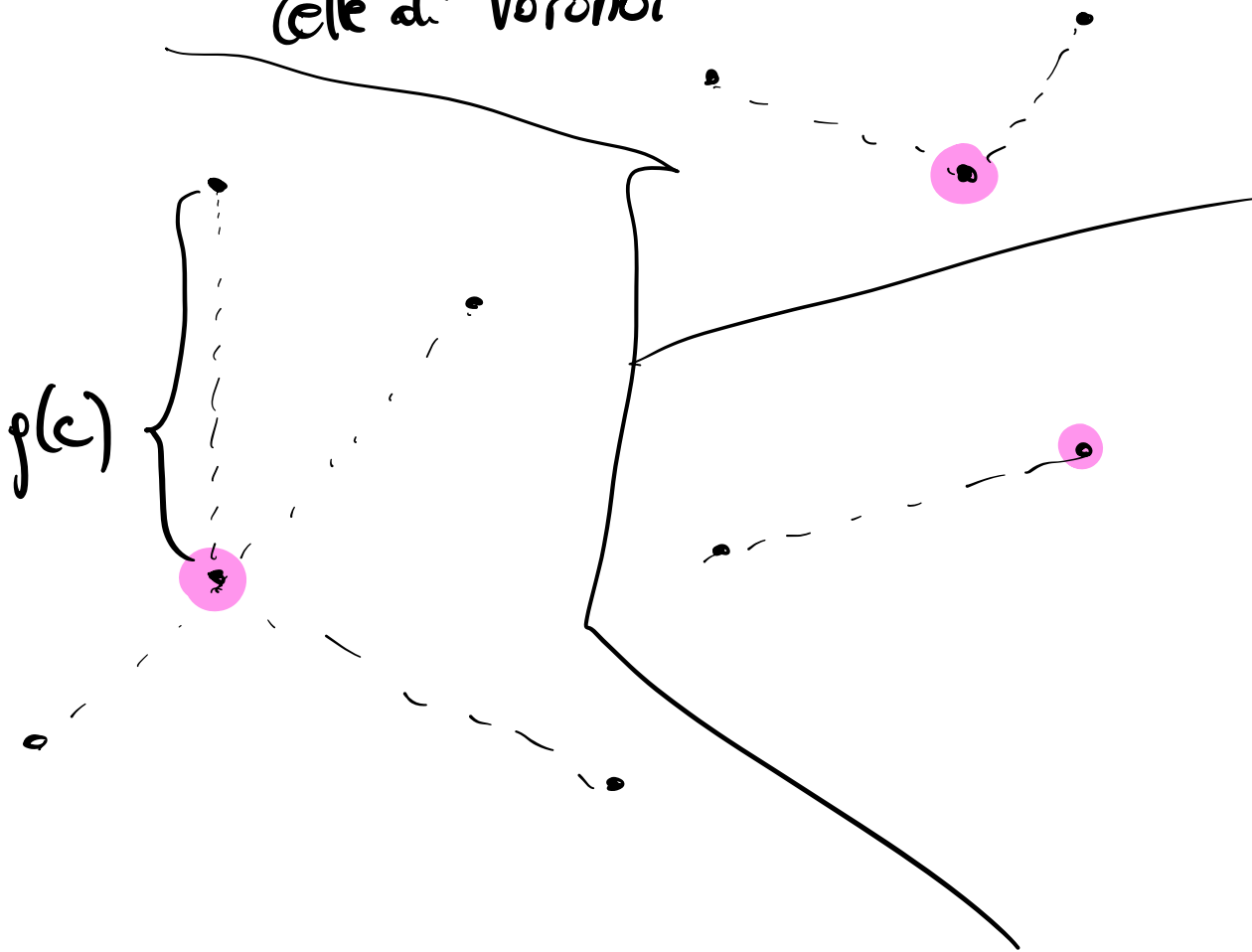
$\forall x, y, z \in \Omega$

- 1)  $d(x, y) \geq 0$  e  $d(x, y) = d(y, x)$
- 2)  $d(x, y) = 0$  sse  $x = y$
- 3)  $d(x, z) + d(z, y) \geq d(x, y)$   
[disug. triangolare]

$$(\mathbb{R}^n, d) \quad d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

$$(\mathbb{R}^n, d_{\max}) \quad d_{\max}(\underline{x}, \underline{y}) = \max_{i=1}^n |x_i - y_i|$$

celle di Voronoi



### CENTER SELECTION

INPUT:

$(\Omega, d)$  spazio metrico

$$S \subseteq \Omega$$

$$k \in \mathbb{N}^+$$

$$C \subseteq S \quad |C| \leq k$$

SOL. ANN.:

$$\forall x \in S \quad p(x, C) = \min_{c \in C} d(x, c)$$

$$p(C) = \max_{x \in S} p(x, C)$$

$f(C) = \min_{x \in S} f(x)$   
FUNZ. OBIETTIVO :  $f(C)$  SPERTURA  
 RAGGIO DI C

TIPO: MIN

$\frac{T_{coreus}}{L}$  : CENTER SELECTION  $\in$  NPO-capd.

# ALGORITHM

## CENTER SELECTION PLUS

INPUT  
D  
Center Selection

INPUT :  $S \subseteq \mathbb{R}^d, k \in \mathbb{N}^+, \pi \in \mathbb{R}^+$

$C \leftarrow \emptyset$

while

$S \neq \emptyset$

take any  $\bar{s} \in S$

-  $C \leftarrow C \cup \{\bar{s}\}$

- remove from  $S$  all

$x$  s.t.  $d(x, \bar{s}) \leq 2\pi$

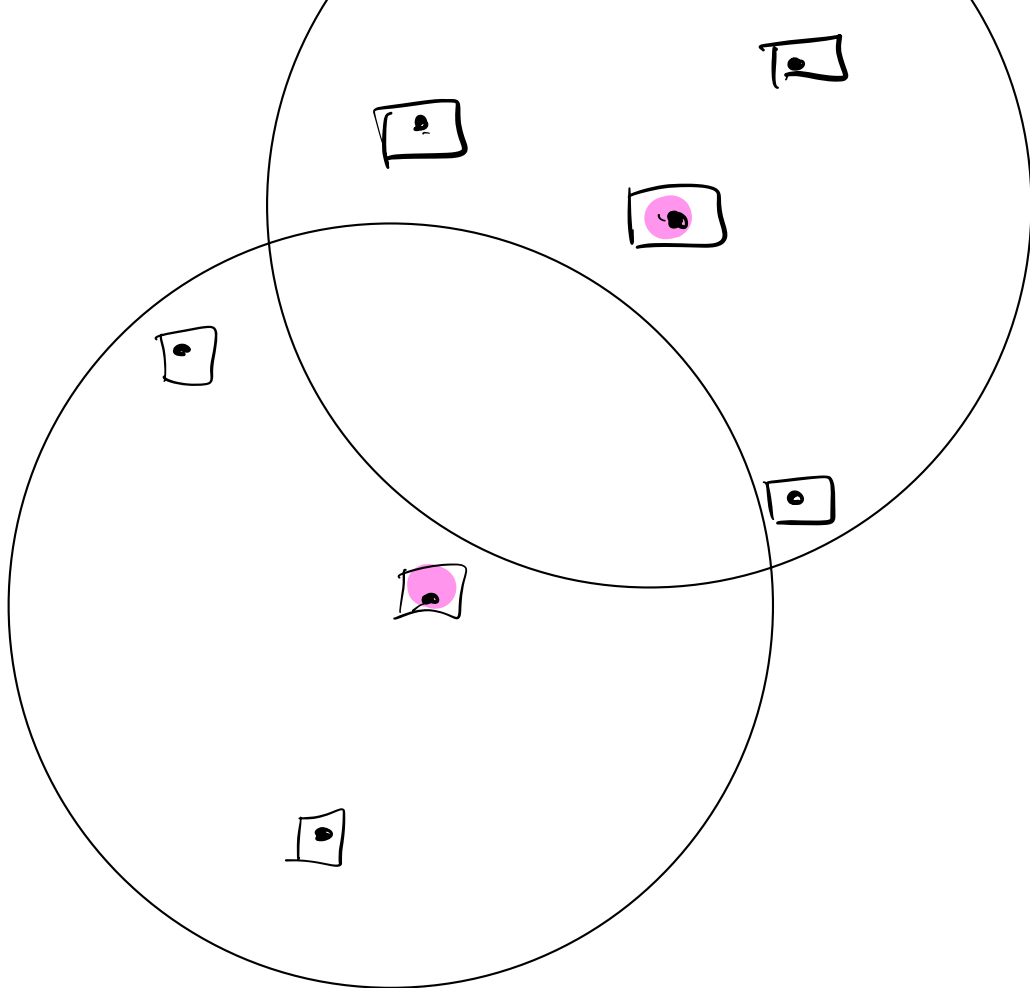
if

$|C| > k$

output "Impossible"

else

output  $C$



$\pi$

