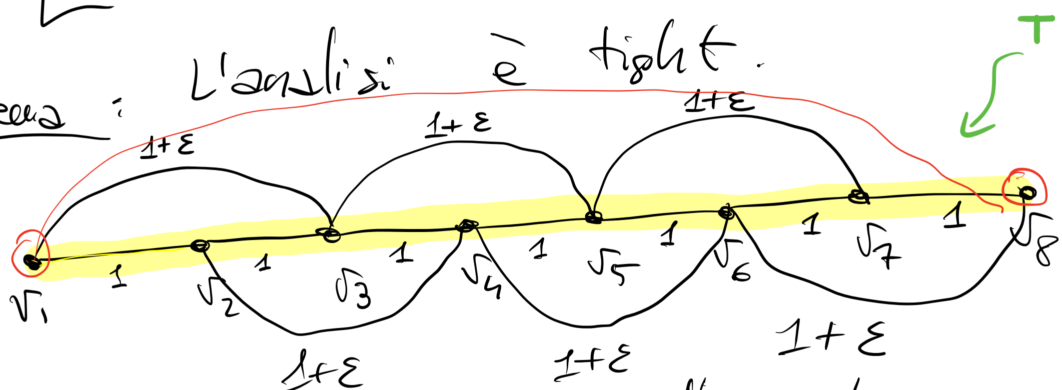
$$\delta(\pi_{HAM}) \le \delta(\pi_{EUL}) = \delta(T) + \delta(\mathcal{M})$$
$$\le \delta^* + \frac{1}{2}\delta^* = \frac{3}{2}\delta^*$$

Corollario: L'Alg. di Christofides è una $3/2$-approssr. per il TSP metrico.

Teorema: L'analisi è tight.

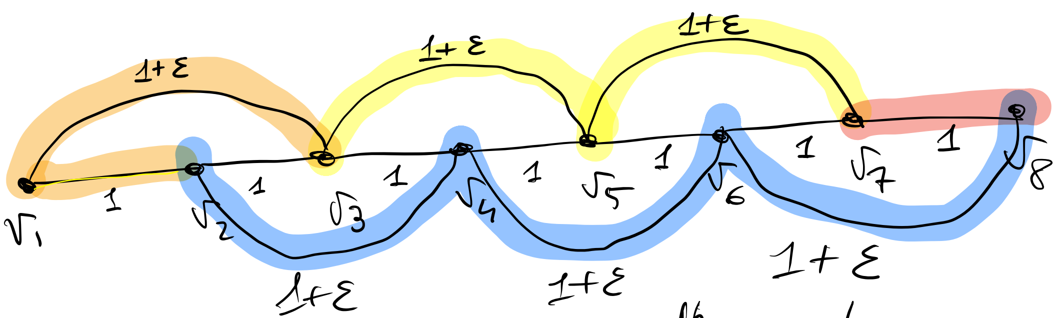


I lati che valgono hanno la lungh. del cammino minimo.

$$\delta(T) = (n-1)$$
$$\delta(\mathcal{M}) = (1+\varepsilon)\left(\frac{n}{2} - 1\right) + 1 =$$
$$= (1+\varepsilon)\frac{n}{2} - 1 - \varepsilon + 1 =$$
$$= (1+\varepsilon)\frac{n}{2} - \varepsilon$$

$$\delta(\pi_{CHRIST}) = n - 1 + (1+\varepsilon)\frac{n}{2} - \varepsilon =$$

$$= \frac{3}{2}n + \varepsilon\left(\frac{n}{2} - 1\right) - 1$$



hall no la

$$\delta^* = (1+\varepsilon)\left(\frac{n}{2} - 2\right) + 1 + (1+\varepsilon)\left(\frac{n}{2} - 1\right) + 1 + 1 + \varepsilon =$$

$$= (1+\varepsilon)(n - 3) + 3 + \varepsilon =$$

$$= n + \varepsilon(n-2) + 3$$

$$\frac{\delta(\pi_{CHRIST})}{\delta^*} = \frac{\frac{3}{2}n + \varepsilon\left(\frac{n}{2} - 1\right) - 1}{n + \varepsilon(n-2) + 3} \xrightarrow[n \to \infty]{} \frac{3}{2}$$

$\square$

INAPPROSSIMABILITÀ DEL TSP GENERALE

Decidere se un grafo ammette un ciclo hamiltoniano è NP-completo.

<u>Teorema</u>: Non esiste alcun $\alpha > 1$ t.c. TSP sia α-approssimabile

<u>Dim</u>: $G = (V, E)$ grafo non orientato
$\Rightarrow$ istanza di TSP (su una clique)
$G' = (V, \binom{V}{2})$

$$\delta_e = \begin{cases} 1 & \text{se } e \in E \\ \lceil \alpha n \rceil + 1 & \text{se } e \notin E \end{cases}$$

- Se c'è un circuito hamiltoniano in G, c'è anche in G', e quello ottimo è lungo n. $\Rightarrow$ L'algoritmo

α-approssimante ottiene un circuito di lungh. $\leq \alpha n$.

- Se in G non c'è un circuito hamiltoniano, qualunque circ. hamiltoniano su G' ha lungh. $\geq \lceil \alpha n \rceil + 1$.

$$\boxed{\alpha n} < \boxed{\lceil \alpha n \rceil + 1}$$

IMP.
$$\alpha n \geq \lceil \alpha n \rceil + 1$$
$$\alpha \geq \frac{\lceil \alpha n \rceil + 1}{n} \geq \frac{\alpha n + 1}{n} =$$
$$= \alpha + \frac{1}{n}$$

IMPOSSIBILE!

$\square$

PTAS PER 2-LOAD BALANCING (A.K.A. MINIMUM PARTITION)

$t_1, t_2, \dots, t_n$

$\boxed{\text{PTAS}}^{(1+\varepsilon)\text{-appr.}}$

INPUT: $t_1, t_2, \dots, t_n \in \mathbb{N}^+$
$\varepsilon > 0$

- If $\varepsilon \geq 1$, dai tutto alla prima macchina
- Ordina i t_i in ordine decrescente
- $k \leftarrow \left\lceil \frac{1}{\varepsilon} - 1 \right\rceil$
- Cerca esaustivamente l'assegnamento ottimo dei task $t_1, t_2, \dots, t_k$
- Assegna i task $t_{k+1}, \dots, t_n$ in modo greedy.

Teorema: L'algoritmo fornisce in tempo polinomiale una $(1+\varepsilon)$-appr.

$$\boxed{L = \frac{t_1 + t_2 \dots + t_n}{2}}$$

Dim:

$\boxed{\text{CASO } 1}$ $\varepsilon \geq 1$

$L \leq L^*$

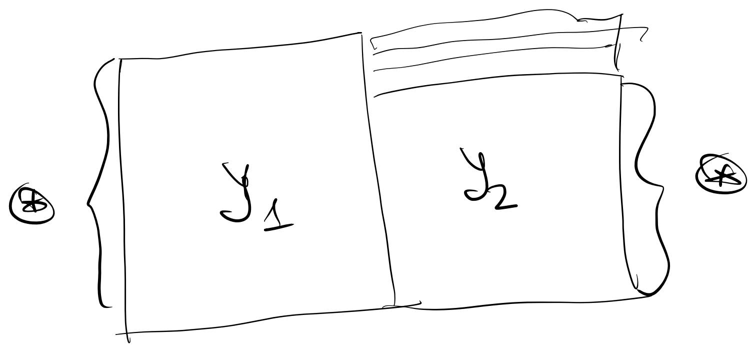
Nostra sol. è $2L$

$$\frac{2L}{L^*} \leq \frac{2L}{L} = 2 \leq 1 + \varepsilon$$

$<$ $<$

$\Rightarrow (\cancel{1}+\varepsilon)$ - appross.

$\boxed{\text{CASO 2}}$ $\varepsilon < 1$



$\circledast$ $t_1, t_2, \ldots, t_k$

Alla fine $L_1 \geq L_2$ (s.p.g.)

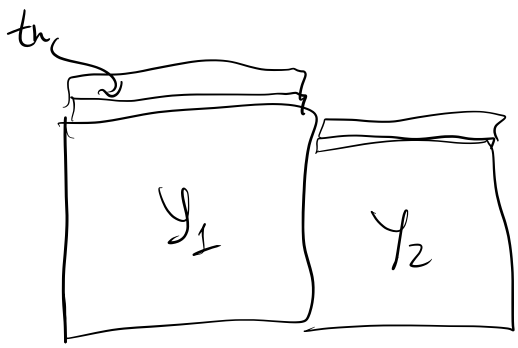
e che t_h sia l'ultimo task

assegnato alla macchina 1.

$\boxed{\text{CASE 2A: } h \leq k}$

Ho trovato la sol. ottima.

$\boxed{\text{CASE 2B: } h > k}$



$$L_1 - \underbrace{t_h}_{} \le \underbrace{L_2'}_{\text{carico della macchina 2 quando viene ass. } t_h} \le L_2$$

$$L_1 - t_h \le L_2$$

sommo L_1 a entrambi i membri

$$L_1 + L_1 - t_h \le L_1 + L_2 = 2L$$

$$L_1 - \frac{t_h}{2} \le L$$

$$\boxed{L_1 \le L + \frac{t_h}{2}} \quad (*)$$

$$2L = t_1 + \dots + t_m \ge t_h(k+1) \quad (**)$$

$$\underbrace{t_1 + \dots + t_k}_{\ge t_h k} + \underbrace{t_{k+1} + \dots + t_h + \dots + t_m}_{\ge t_h}$$

$L^* \ge L$

$$\frac{\max(L_1, L_2)}{L^*} = \frac{L_1}{L^*} \overset{(*)}{\le} \frac{L + \frac{t_h}{2}}{L^*} \le \frac{L + \frac{t_h}{2}}{L} =$$

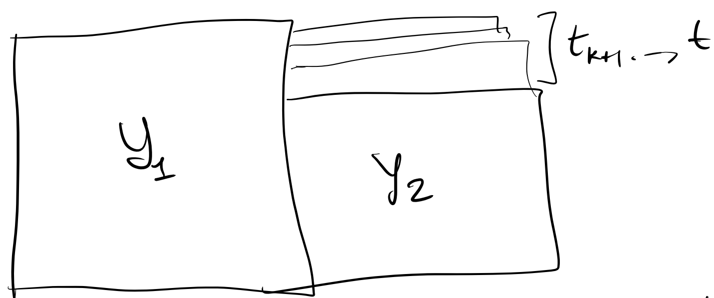
$$= 1 + \frac{t_h}{2L} \overset{(**)}{\le} 1 + \frac{t_h}{t_h(k+1)} =$$

$$= 1 + \frac{1}{k+1} \leq 1 + \frac{1}{\frac{1}{\varepsilon} - 1 + 1} =$$

$$= \varepsilon + 1 \quad . \qquad \square$$

$\boxed{\text{CASE 2: } \overset{\text{aggiunto}}{} \quad h \leq k}$

Ho trovato la sol. ottima.



$t_1, \dots, t_k$

Dopo aver eseguito t_i la sol. è ottima. $i = k$ vero

$t_i \to t_{i+1}$

$$2^k = 2^{\frac{1}{\varepsilon}-1}$$