Lema 1.

$$OPT \ge max\left\{\frac{1}{m}\sum_{1\le j\le n} t_j,\ max\left\{t_j| \le j \le n\right\}\right\}$$

$justificare:$

Primul termen reprezinta incarcatura medie pt o masina. Evident ca in orice configuratie, masina cu incarcatura maxima va lucra cel putin la fel de mult ca media tuturor masinilor

Al doilea termen reprezinta job – ul cel mai costisitor. Aceasta nu poate fi impartita pe mai multe masini.

Lema 2.

$$ALG \le 2 \cdot OPT$$

Fie k indicele masinii cu load maxim in urma executarii algoritmului.
Fie q ultimul job adaugat masinii k.
fie load'(M) - load-ul masiniii M dupa ce am asignat primele q-1 joburi dar nu si jobul q

$$load'(k) \le \frac{1}{m}\sum_{1\le i\le m} load'(i) = \frac{1}{m}\sum_{1\le j<q} t_j < \frac{1}{m}\sum_{1\le j\le n} t_j \le LB \le OPT$$

$$\begin{aligned}
ALG = L_k &= load'(k) + t_q \\
&\le LB + t_q \\
&\le LB + max\left\{t_j|1 \le j \le n\right\} \\
&\le LB + LB \\
&\le 2 \cdot OPT
\end{aligned}$$

Teorema: Algoritmul descris anterior este un algoritm 2-1/m aproximativ (slide 37)

$$load'(k) \le \frac{1}{m}\sum_{1\le i\le m} load'(i) = \frac{1}{m}\sum_{1\le j<q} t_j \le \frac{1}{m}\left(\sum_{1\le j\le n} t_j - t_q\right) \le LB - \frac{1}{m}t_q$$

t_{max} – *timpul maxim de executie al unui job din multimea de activitati*

$$\begin{aligned}
ALG = load'(k) + t_q &\le \frac{1}{m}\sum_{1\le j\le n} t_j - \frac{1}{m}t_q + t_q \le \frac{1}{m}\sum_{1\le j\le n} t_j - \frac{1}{m}t_{max} + t_{max} \\
&\le OPT - \frac{1}{m}OPT + OPT = \left(2 - \frac{1}{m}\right)OPT
\end{aligned}$$

Slide 39: este 2-1/m tight bound?
Da!
fie m masini si m(m-1) activitati de cost 1 si o activitate de cost m. Algoritmul nostru va produce un load maxim de 2m-1 (o activitate de cost m si m-1 activitati de cost 1)
Solotia optima: o masina are 1 activitate de cost m. Restul de m-1 masini au cate m activitati de cost 1. Per total fiecare masina are un load de exact m unitati de timp.

ALG=2m-1
OPT=m
ALG=**(2-1/m)***OPT

Lema 3 Slide 43
Fie o multime de n activitati cu timpul de procesare $t_1, t_2, ..., t_n$ astfel incat $t_1 \geq t_2 \geq \dots t_n$

Daca $n > m$, *atunci* $\boldsymbol{OPT \geq t_m + t_{m+1}}$
Din principiul cutiilor rezulta faptul ca daca avem minim m+1 activitati si m masini, atunci cel putin o masina va avea cel putin 2 activitati. Fie acele activitati cu timpii de executie ti si tj
OPT>=ti+tj>=t_m + t_{m+1}

Teorema 2 Slide 43
fie k indicele masinii cu loadul maxim in urma executarii algoritmului.

deci $ALG = load(k)$

fie q ultima activitate adaugata pe masina k

fie $load'(i)$ $-$ *load-ul masinii i fix inainte ca activitatea q sa fie asociata masinii k.*

adica load' semnifica load-ul masinilor dupa ce au fost distribuite primele $q-1$ *activitati*

$ALG = load(k) = load'(k) + t_q$

$$load'(k) + t_q \leq \frac{1}{m} \sum_{1 \leq j \leq n} t_j + t_q$$

ce se intampla daca $q \leq m$?

atunci q va fi pusa peste o masina goala, deci

$ALG = t_q \leq t_{max}$ (*activitatea cu timpul de lucru maxim*) $\leq OPT$

$ALG = OPT$ $-$ *deci daca q este in primele m activitati, atunci algoritmul nostru este exact*

daca $q > m$

$$laod'(k) + t_q \leq \frac{1}{m} \sum_{1 \leq i \leq n} t_i + t_q < \frac{1}{m} \sum_{1 \leq i \leq n} t_i + \frac{1}{2}\left(t_m + t_{m+1}\right) \leq OPT + \frac{1}{2} OPT = \frac{3}{2} OPT$$