Lema 1 (slide 33)

$$OPT \geq \max\left(\frac{1}{m}\sum_{1\leq j\leq n} t_j, \max\{t_j | 1 \leq j \leq n\}\right)$$

Justificare:

Primul termen reprezintă timpul mediu de lucru al tuturor mașinilor pentru a efectua întreaga mulțime de task-uri. Mașina care lucrează ce mai mult va lucra mereu cat cel puțin media timpului de lucru necesar.

Al doilea termen reprezintă timpul de lucru a celei mai costisitoare activități. Una dintre mașini va trebui sa prelucreze acea activitate, deci mașina cu încărcătura cea mai mare va lucra minim timpul de lucru necesar pentru cea mai costisitoare activitate

Lema 2 (slide 33-34)

Algoritmul descris anterior (slide 19) este 2-aproximativ;

ALG≤2*OPT

Justificare

Fie

- k - indicele masinii cu load-ul maxim in urma rularii algoritmului
- q – ultimul job asignat masinii k
- load'(M) – load-ul masinii M dupa asignarea primelor q-1 job-uri, dar nu si job-ul q

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

$$load'(k) \leq \frac{1}{m}\sum_{1\leq i\leq m} load'(i) = \frac{1}{m}\sum_{1\leq j<q} t_j < \frac{1}{m}\sum_{1\leq j\leq n} t_j \leq OPT\ (conf. Lema\ 1)$$

$$t_q \leq \max\{t_j | 1 \leq j \leq n\} \leq OPT\ (conf. Lema\ 1)$$

$$\boldsymbol{ALG} = load'(k) + t_q \leq OPT + OPT = \mathbf{2} * \boldsymbol{OPT}$$

Teorema 1 (slide 37): Algoritmul Greedy descris anterior (slide 19) este un algoritm 2-1/m aproximativ

Justificare:

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

$$ALG = load'(k) + t_q \leq \frac{1}{m}\left(\sum_{1\leq j\leq n} t_j\right) - \frac{1}{m} * t_q + t_q = \frac{1}{m}\left(\sum_{1\leq j\leq n} t_j\right) + t_q\left(1 - \frac{1}{m}\right)$$
$$\leq OPT + OPT\left(1 - \frac{1}{m}\right) = \left(2 - \frac{1}{m}\right)OPT$$
$$ALG \leq \left(2 - \frac{1}{m}\right)OPT$$

Slide 39: este 2-1/m tight bound?
Da!
fie m mașini si m(m-1) activități 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 activități de cost 1)
Soluția optima: o mașina are 1 activitate de cost m. Restul de m-1 mașini au cate m activități de cost 1. Per total fiecare mașina are un load de exact m unități de timp.
ALG=2m-1
OPT=m
ALG=**(2-1/m)***OPT
yp

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* $OPT \geq t_{\mathrm{m}} + t_{\mathrm{m+1}}$

Justificare
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
$OPT \geq t_i + t_j \geq t_m + t_{m+1}$

Teorema 2 Slide 43
Algoritmul “Ordered Scheduling Algorithm” (slide 42) este 3/2-aproximativ

Justificare:
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 $q > m$

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