Thm. No on-line scheduling algorithm can achieve a competitive factor greater than 0.25.

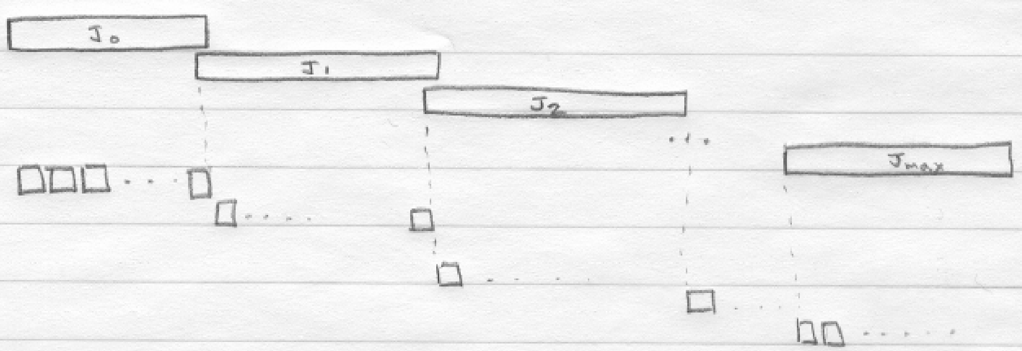
Proof: [Baruah, et. al, 1991]

An adversary creates two types of jobs:

(a) major jobs $J_0, J_1, \ldots, J_{max}$

(b) minor jobs $J_0^1, J_0^2 \ldots, J_1^1, J_1^2 \ldots, \ldots, J_{max}^1 \ldots$

Minor jobs each have run-time ε and are associated with major jobs; e.g. J_i^k is associated w/ J_i.



As long as the scheduler executes major job J_i, the adversary continues to create small associated jobs until $d_i - \varepsilon$.

Whenever the scheduler decides to execute a job associated with J_i, the adversary stops generating jobs associated with J_i — note this is the same as the EDF example — only one associated job of size ε is scheduled successfully.

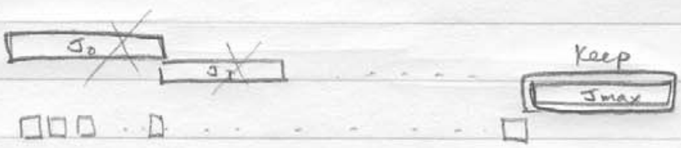
The adversary releases J_i just ε time units before the completion of $J_{i-1} = d_{i-1}$; that is,

$$r_i = r_{i-1} + e_{i-1} - \varepsilon = d_{i-1} - \varepsilon.$$

In this way, <u>only</u> J_i or J_{i-1} may be scheduled.

If the scheduler <u>discards J_{i-1} to execute J_i</u> then the adversary releases J_{i+1} up to J_{max}, in this case, the scheduler discards all jobs up to J_{max}

$$\Rightarrow \text{total value of schedule} = e_{max}.$$



An optimal, clairvoyant scheduler would schedule all associated jobs up to r_{max} and then J_{max}.

$$\Rightarrow \text{total value (optimal scheduler)} = \sum_{k=0}^{max} e_k - \varepsilon(max)$$

$$\rightarrow \sum_{k=0}^{max} e_k \quad \text{as } \varepsilon \rightarrow 0.$$

= to show.

$$\text{Thus, the competitive factor} \le \frac{e_{max}}{\sum_{k=0}^{max} e_k} \overset{T.S.}{\le} \frac{1}{4}$$

How to choose e_k?

Def. Set
$$\begin{cases} e_0 = 1 \\ e_k = \bar{c}\, e_{k-1} - \sum_{j=0}^{k-1} e_k \end{cases} \Rightarrow \sum_{j=0}^{k} e_k = \bar{c}\, e_{k-1}.$$

Set $\bar{c} = 4$.

Example, let

$\bar{c} = 4$:

i	e_i	$\sum_{k=0}^{i} e_i$	$\frac{e_i}{\sum_{k=0}^{i} e_i}$
0	1	1	1
1	$4(1) - 1 = 3$	4	$3/4$
2	$4(3) - 4 = 8$	12	$4/6$
3	$4(8) - 12 = 20$	32	$5/8$
4	$4(20) - 32 = 48$	80	$6/10$

On the other hand if the scheduler decides to schedule only J_i to completion for some $i < max$, then the scheduler's value is e_i; whereas an optimal scheduler obtains $\sum_{k=0}^{i+1} e_i$ by scheduling

all jobs associated with $J_0, \ldots J_i$ followed by job J_{i+1}.

Thus, the competitive factor is $\dfrac{e_i}{\sum_{k=0}^{i+1} e_k} = \dfrac{e_i}{e_{i+1} + \sum_{k=0}^{i} e_i}$

$$\overset{=}{\text{def.}} \frac{e_i}{\bar{c}\, e_i - \cancel{\sum_{k=0}^{i} e_k} + \cancel{\sum_{k=0}^{i} e_i}} = \frac{1}{\bar{c}} = \frac{1}{4}.$$

To find the tightest bound (smallest competitive factor or largest $\bar{c}$) – find the largest $\bar{c}$ s.t. the function defined by the recurrence relation

$$\begin{cases} e_0 = 1 & (1) \\ e_{i+1} = \bar{c}\, e_i - \sum_{j=0}^{i} e_j \quad \text{for } i \geq 0. & (2) \end{cases}$$

satisfied the property:

$$\left(\exists\, max : max \geq 0 : \frac{e_{max}}{\sum_{j=0}^{max} e_j} \leq \frac{1}{\bar{c}} \right) \qquad (3)$$

Note that e_i satisfies (3) iff

$$\left(\exists\, l : l \geq 0 : e_{l+1} \leq e_l \right)$$

$$\frac{e_{max}}{\sum_{j=0}^{max} e_j} \leq \frac{1}{\bar{c}} \equiv \frac{e_{max}}{\sum_{j=0}^{max-1} e_j + e_{max}} \leq \frac{1}{\bar{c}}$$

$$\equiv \frac{e_{max}}{\left(\sum_{j=0}^{max-1} e_j + \bar{c}\, e_{max-1} - \sum_{j=0}^{max-1} e_j \right)} \leq \frac{1}{\bar{c}}$$

$$\equiv \frac{e_{max}}{\bar{c}\, e_{max-1}} \leq \frac{1}{\bar{c}}$$

$$\equiv e_{max} \leq e_{max-1}. \qquad (4)$$

To solve the recurrence, note that

$$e_{i+2} = \bar{c}\; e_{i+1} - \sum_{j=0}^{i+1} e_j$$

$$- \quad e_{i+1} = \bar{c}\; e_i - \sum_{j=0}^{i} e_j$$

$$e_{i+2} - e_{i+1} = \bar{c}\,(e_{i+1} - e_i) - e_{i+1}$$

$$\Rightarrow \quad e_{i+2} = \bar{c}\,(e_{i+1} - e_i) \qquad (5)$$

$$\Rightarrow \quad e_{i+2} - \bar{c}\,e_{i+1} + \bar{c}\,e_i = 0$$

Characteristic equation:

$$x^2 - \bar{c}x + \bar{c} = 0$$

w/ roots

$$x = \frac{\bar{c} \pm \sqrt{\bar{c}^2 - 4(1)(\bar{c})}}{2}$$

$$= \bar{c} + \sqrt{\bar{c}^2 - 4\bar{c}}$$

When $\bar{c} = 4$, we have

$$e_i = d_1\, i \cdot 2^i + d_2 \cdot 2^i$$

$$e_0 = d_1\, 0 \cdot 2^0 + d_2 \cdot 2^0 = \underline{d_2 = 1}$$

$$e_1 = 4 \cdot e_0 - e_0 = 3$$

$$= d_1\, 1 \cdot 2^1 + 1 \cdot 2^1$$

$$= 2d_1 + 2 = 3$$

$$\Rightarrow d_1 = \tfrac{1}{2}.$$

$\Rightarrow \quad e_i = \frac{1}{2} \cdot i \cdot 2^i + 2^i$ which is strictly increasing. So we can't set $\bar{c} = 4$, but maybe close to it.

Second case : $(\bar{c} > 4)$

solving as above,

$$d_1 = \frac{1}{2} + \frac{\bar{c}-2}{2\sqrt{\bar{c}^2-4\bar{c}}} \quad \text{and} \quad d_2 = \frac{1}{2} - \frac{\bar{c}-2}{2\sqrt{\bar{c}^2-4\bar{c}}}$$

since $d_1 > 0$, e_k diverges ...

Third case : $(\bar{c} < 4)$

solving,

$$d_1 = \frac{1}{2} + \frac{\bar{c}-2}{2\sqrt{\bar{c}^2-4\bar{c}}} \quad \text{and} \quad d_2 = \frac{1}{2} - \frac{\bar{c}-2}{2\sqrt{\bar{c}^2-4\bar{c}}}$$

since $\bar{c}^2 - 4\bar{c} < 0$, we have complex conjugates and may be represented by

$$d_1 = s e^{j\theta} \quad \text{and} \quad d_2 = s e^{-j\theta}$$

where $s \in \mathbb{R}$, $j = \sqrt{-1}$, $-\pi/2 < \theta < 0$.

solving,

$$e_i = 2 s r^i \cos(\theta + i\omega), \quad 0 < \omega < \frac{\pi}{2}.$$

$\omega \neq 0 \Rightarrow \cos(\theta + i\omega)$ is negative for some $i \in \mathbb{N}$ which implies that e_i satisfies (4).