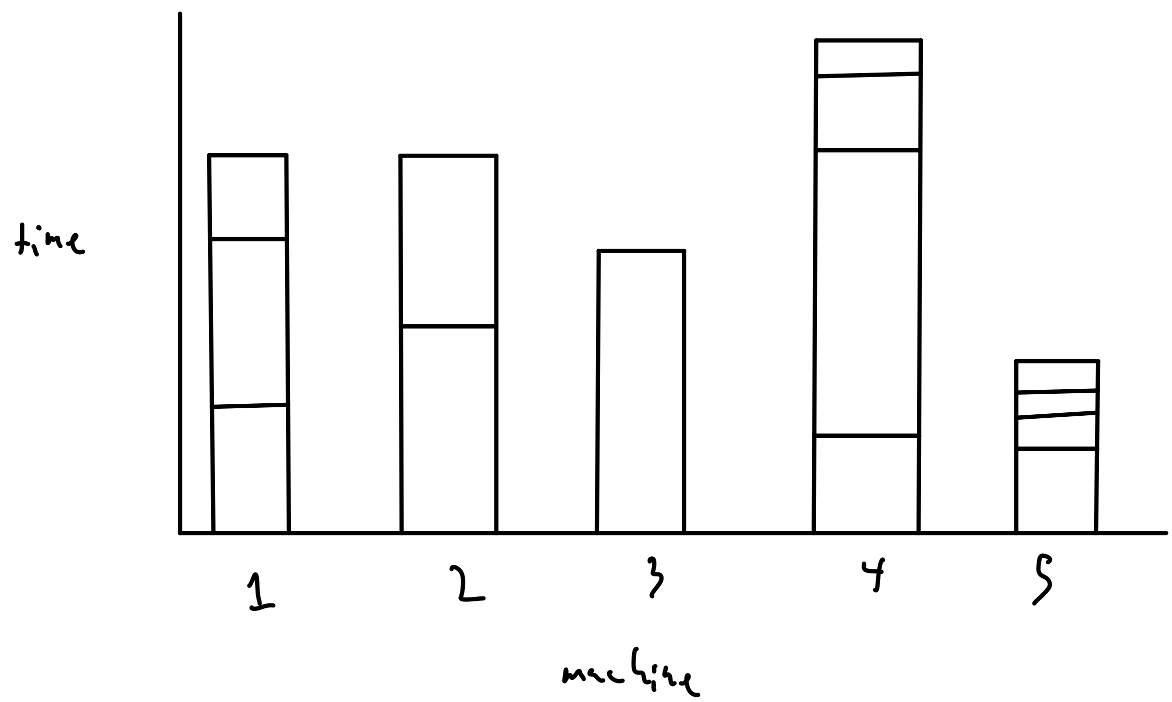
Min-Makespan on Identical Parallel Machines

Input: - Jobs J $(|J| = n)$

- Machines M $(|M| = k)$
- Processing times $p : J \to \mathbb{N}$

Feasible Solution: $\Phi : J \to M$

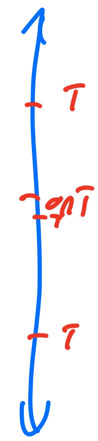
Objective: min makespan = $\min \max_{m \in M} \sum_{j \in J : \Phi(j) = m} p(j)$



Thm: Greedy is a 2-approximation.

want to do better!

Idea: what if we have a "guess" T for OPT?



<u>Def</u>: A $(1+\varepsilon)$-relaxed decision procedure is an algorithm which, given input and T:

1) If $T \geq OPT$, returns a solution with makespan $\leq (1+\varepsilon)T$
2) If $T < OPT$, either returns "false" or returns a solution with makespan $\leq (1+\varepsilon) \cdot OPT$

Let $P = \sum_{j \in J} p(j)$. Then $1 \leq OPT \leq P$

Idea: do binary search to find value of T s.t. procedure returns a solution on T, "false" on $T-1$

$\Rightarrow T \leq OPT$

$\Rightarrow$ solution has makespan $\leq (1+\varepsilon) \cdot OPT$

Takes $O(\log P)$ iterations: polytime!

So just want to design such a procedure, given guess T.

<u>Def</u>: $J_{small} = \{ j \in J : p(j) \leq \varepsilon T \}$

$J_{large} = \{ j \in J : p(j) > \varepsilon T \}$

Thm: Given schedule for J_{large} with makespan $\leq (1+\varepsilon)T$, we can find a schedule for J with makespan $\leq (1+\varepsilon)\max(T, OPT)$

Pf:

Start with schedule for J_{large}.

Greedy on J_{small}:

- in arbitrary order, add job to least-loaded machine

Consider machine m.

- Case 1: m has no small jobs.

  $\Rightarrow$ load $\leq (1+\varepsilon)T$

- Case 2: m has ≥ 1 small job

  Let j be last small job assigned to m

  $\Rightarrow p(j) \leq \varepsilon \cdot T$,

  load on m just before $\leq \frac{P}{k} \leq OPT$

  $\Rightarrow$ total load $\leq OPT + \varepsilon T \leq (1+\varepsilon)\max(T, OPT)$

So just need to schedule J_{large} with makespan $\le (1+\varepsilon)T$

Let $b = \lceil \frac{1}{\varepsilon} \rceil$, so $\frac{1}{b} \le \varepsilon$

Def: Let $p'(j) = \left\lfloor \frac{p(j)\, b^2}{T} \right\rfloor \cdot \frac{T}{b^2}$ ("rounded instance")

p' just p rounded down to multiple of $\frac{T}{b^2}$

$\Rightarrow p'(j) \le p(j) \le p'(j) + \frac{T}{b^2}$

$p'(j) = k \frac{T}{b^2}$ for some $k \in \{b, b+1, \dots, b^2\}$ (since $j \in J_{large}$)

For schedule Φ, let $m(\Phi)$ be makespan in original

$m_r(\Phi)$ makespan in rounded.

Lemma 1: Let Φ schedule. Then $m_r(\Phi) \le m(\Phi)$

Pf:

$p'(j) \le p(j) \quad \forall j \in J$

Lemma 2: Let Φ schedule with $m_r(\Phi) \le T$. Then

$m(\Phi) \le (1+\varepsilon)T$

Pf:

Let m some machine

since $p'(j) \geq \frac{T}{b}$, $\quad |\Phi^{-1}(m)| \leq b$

$$\Rightarrow \sum_{j \in \Phi^{-1}(m)} p(j) \leq \sum_{j \in \Phi^{-1}(m)} \left(p'(j) + \frac{T}{b^2} \right)$$

$$= \underbrace{\sum_{j \in \Phi^{-1}(m)} p'(j)}_{\leq T} + \underbrace{\sum_{j \in \Phi^{-1}(m)} \frac{T}{b^2}}_{\leq b \cdot \frac{T}{b^2}}$$

$$\leq T + \frac{T}{b} \leq T + \varepsilon T = (1+\varepsilon) T$$

Suppose we had an algorithm to find some Φ with $m_r(\Phi) \leq T$ *if* one exists

If $T \geq OPT$:

$\exists \, \Phi$ with $m(\Phi) \leq T \Rightarrow m_r(\Phi) \leq T$ (Lemma 1)

$\Rightarrow \exists$ schedule w/ makespan $\leq T$ in rounded

$\Rightarrow$ alg will find some Φ' with $m_r(\Phi') \leq T$

$\Rightarrow m(\Phi') \leq (1+\varepsilon) T$ (Lemma 2)

If $T < OPT$:

If alg returns Φ, have

$m(\Phi) \leq (1+\varepsilon)\, T$ (Lemma 2)

So just need to find Φ with $m_r(\Phi) \leq T$ if one exists!

Def: A configuration is a tuple $(a_b, a_{b+1}, \dots, a_{b^2})$ such that:

1) Each $a_i \in \{0, 1, \dots, b\}$
2) $\sum_{i=b}^{b^2} a_i \cdot i \cdot \frac{T}{b^2} \leq T$

Idea: Look at jobs assigned to some machine, let a_i be # with $p'(j) = i \cdot \frac{T}{b^2}$

$\Rightarrow$ get a configuration

Let $\mathcal{C}(T)$ be set of all configurations

$$|\mathcal{C}(T)| \le (b+1)^{b^2-b} \approx (b)^{b^2}$$

Dynamic Programming: given (rounded) jobs, how many machines are necessary?

More formally:

$f(n_b, n_{b+1}, \ldots, n_{b^2}) = \min m$ s.t. can schedule n_i jobs of length $i \cdot \frac{T}{b^2}$ $\forall i$ with makespan $\le T$

$$f(\vec{0}) = 0$$

$$f(n_b, \ldots, n_{b^2}) =$$

$$1 + \min_{\vec{a} \in \mathcal{C}(T)} f(n_b - a_b, n_{b+1} - a_{b+1}, \ldots, n_{b^2} - a_{b^2})$$

Time/table entry: $|\mathcal{C}(T)| \approx b^{b^2}$

table entries: $\le n^{b^2}$

$\Rightarrow$ total time $\leq n^{O(b^2)} = n^{O((1/\varepsilon)^2)}$

Once table filled out, check entry corresponding to J_{large} (rounded)

If $> m$: no Φ with $m_r(\Phi) \leq T$

$\Rightarrow$ no Φ with $m(\Phi) \leq T$

$\Rightarrow$ return false

If $\leq m$: Return Φ with $m_r(\Phi) \leq T$