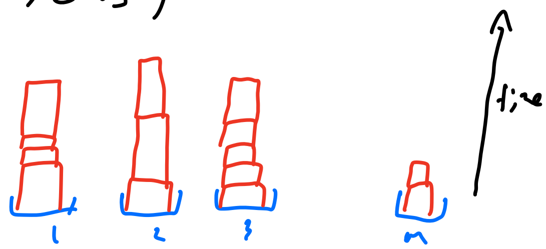
Load Balancing Game:

- n players/jobs, job i has weight w_i
- m machines/strategies: $S_i = [m]$ $\forall i \in [n]$, $S = [m]^n$
  ($A \in S$ assignment $A : [n] \to [m]$)
- load $\ell_j(A) = \sum_{i : A(i) = j} w_i$



- $c_i(A) = \ell_{A(i)}(A)$
- $\text{cost}(A) = \max_{j \in [m]} \ell_j(A)$ (makespan)

<u>Thm</u>: Load Balancing game has a pure Nash equilibrium

$A \to (\ell_1(A), \ell_2(A), \dots, \ell_m(A))$

$\longrightarrow$ sort non-increasing $\to (\lambda_1(A), \lambda_2(A), \dots, \lambda_m(A)) = \lambda(A)$

$(\mathrm{cost}(A) = \lambda_1(A))$

Let $<$ be <u>lexicographic</u> ordering of vectors:

$(1,2,10) < (1,2,15) < (1,3,1) < (3,2,1) \dots$

For $A, B \in S$, $A < B$ if $\lambda(A) < \lambda(B)$

Note: if $\lambda(A) \neq \lambda(B)$, then either $A < B$ or $B < A$

Let $A \in S$ be min according to $<$

<u>Claim</u>: A is a pure Nash

Say i has incentive to deviate, to get A'

$$\lambda(A) = (\lambda_1(A), \dots, \lambda_j(A), \dots, \lambda_k(A), \dots, \lambda_m(A))$$

machine i is assigned by A

deviate to here

Note: without loss of generality, $\lambda_j(A) > \lambda_{j+1}(A)$

$$\lambda(A') = (\lambda_1(A'), \dots, \lambda_j(A'), \dots, \lambda_k(A'), \dots, \lambda_m(A))$$

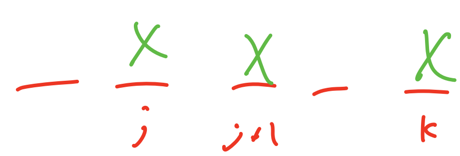
Since they don't change:

$$\lambda_1(A') = \lambda_1(A), \quad \lambda_2(A') = \lambda_2(A), \dots, \lambda_{j-1}(A') = \lambda_{j-1}(A)$$

Only changes at machine j and k.

Since i has incentive to deviate: $\ell_j(A) > \ell_k(A')$

$$\Rightarrow \lambda_j(A') < \lambda_j(A)$$



(if same machine, goes down

if k, still less

if $j+1$, know $\lambda_{j+1}(A) < \lambda_j(A)$)

<u>Cor</u>: Price of Stability $= 1$

<u>Pf</u>: A pure Nash from before.
B arbitrary assignment

$\Rightarrow \lambda(A) \preceq \lambda(B) \quad \forall B$

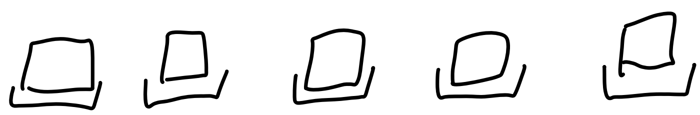
$\Rightarrow \lambda_1(A) \leq \lambda_1(B)$

$\Rightarrow \mathrm{cost}(A) = \lambda_1(A) \leq \lambda_1(B) = \mathrm{cost}(B)$ ✓

Pure Nash vs. Mixed Nash

Ex: Set $n=m$, $w_i = 1$ $\forall i \in [n]$

Pure Nash:



each job gets own machine $\Rightarrow$ cost $= 1$

Mixed Nash:

Each job chooses machine uniformly at random

$$E[c_i(A)] = 1 + \sum_{k \neq i} \Pr[A(k) = A(i)] = 1 + \frac{n-1}{n}$$

$$E[c_i(A_{-i}, j)] = 1 + \sum_{k \neq i} \Pr[A(k) = j] = 1 + \frac{n-1}{n}$$

$\Rightarrow$ Nash

$E[\mathrm{cost}(A)] > 1$ even though $\max_j E[\ell_j(A)] = 1$!

Ex: $m = n = 2$

$\Rightarrow$ 

$\frac{1}{4}$ $\qquad$ $\frac{1}{4}$ $\qquad$ $\frac{1}{4}$ $\qquad$ $\frac{1}{4}$

$\Rightarrow E[\mathrm{cost}] = \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 2 = \frac{3}{2}$

So pure PoA $= 1$, PoA $= \frac{3}{2}$!

Q: Is pure PoA always 1? Always small?

How big can PoA be?

How big can $\frac{\text{PoA}}{\text{pure PoA}}$ be?

A: pure PoA always small, PoA can get pretty big!

Thm: Pure price of anarchy $\le 2 - \frac{2}{m+1}$

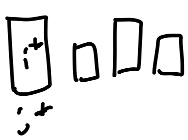
Pf: Let A pure Nash

j^* machine with highest load under A

$(\mathrm{cost}(A) = \ell_{j^*}(A))$

i^* job with smallest weight assigned to j^*

Sps i^* only job assigned to j^*:



$$OPT \ge w_{i^*} = \ell_{j^*}(A) = \mathrm{cost}(A)$$

So WLOG, ≥ 2 jobs assigned to j^*

$$\Rightarrow w_{i^*} \le \frac{1}{2} \cdot \ell_{j^*}(A) = \frac{1}{2} \cdot \mathrm{cost}(A)$$

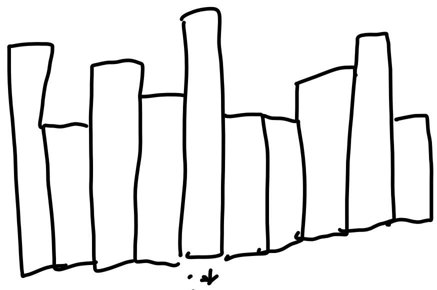
Since Nash, $\forall j \in [m]$:

$$\ell_j(A) \geq \ell_{j^*}(A) - w_{i^*}$$

or else i^* would switch to j

$$\geq \mathrm{cost}(A) - \tfrac{1}{2} \cdot \mathrm{cost}(A) = \tfrac{1}{2} \cdot \mathrm{cost}(A)$$

$\ell_{j^*}(A) = \mathrm{cost}(A)$, $w_{i^*} \leq \frac{1}{2} \cdot \ell_{j^*}(A)$



$$\mathrm{OPT} \geq \frac{\sum_{i=1}^{n} w_i}{m}$$ (max load $\geq$ average load)

$$= \frac{\sum_{j=1}^{m} \ell_j(A)}{m}$$ (total load = total load)

$$= \frac{\ell_{j^*}(A) + \sum_{j \neq j^*} \ell_j(A)}{m}$$

$$\geq \frac{\mathrm{cost}(A) + \sum_{j \neq j^*} \frac{1}{2} \mathrm{cost}(A)}{m}$$ (previous)

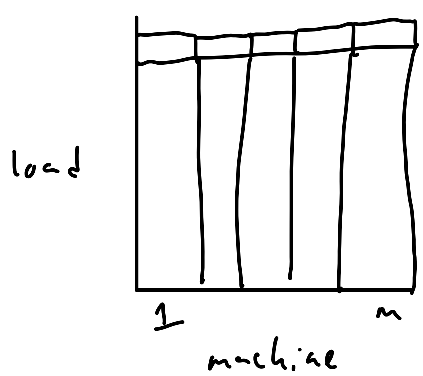
$$= \frac{\left(1+\frac{m-1}{2}\right)\text{cost}(A)}{m}$$

$$= \left(\frac{m+1}{2m}\right)\cdot \text{cost}(A)$$

$$\Rightarrow \frac{\text{cost}(A)}{\text{OPT}} \le \frac{2m}{m+1} = 2 - \frac{2}{m+1} \quad \checkmark$$

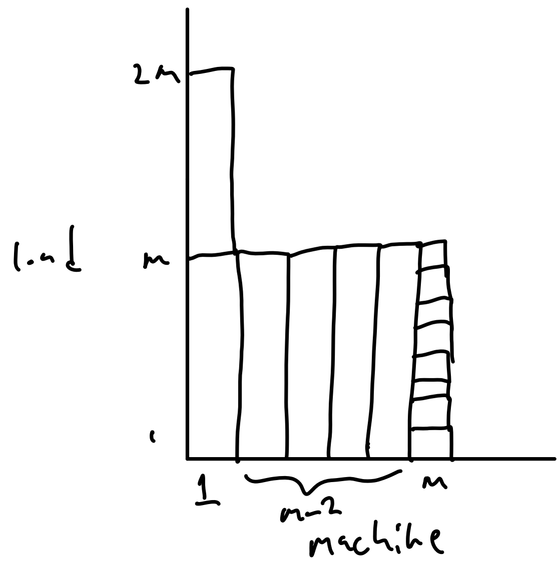
Analysis tight: $n = 2m$, m jobs weight m
m jobs weight 1

OPT:



Each machine has 1 big, 1 small job
$\Rightarrow \text{OPT} = m+1$

Pure Nash:



$\Rightarrow$ Pure price of anarchy $= \frac{2m}{m+1} = 2 - \frac{2}{m+1}$

Mixed Nash:

Look at previous example: $n = m$, $w_i = 1$ $\forall i \in [n]$

Thm: Price of Anarchy $\geq \Omega\left(\frac{\ln m}{\ln \ln m}\right)$

Pf (sketch): OPT $= 1$

Nash σ: All players choose machine u.a.r.

"Balls-in-bins": throw n balls randomly into m bins

Expected max occupancy $= \mathbb{E}_{A \sim \sigma}\left[\max_{j \in [m]} l_j(A)\right]$

$= \Theta\left(\frac{\ln m}{\ln\left(1 + \frac{m}{n} \ln m\right)}\right)$

For us $m=n$: $\Theta\left(\frac{\ln m}{\ln \ln m}\right)$ ✓

So: pure Nash always $\leq 2 \cdot OPT$

mixed Nash can be $\geq \Omega\left(\frac{\ln m}{\ln \ln m}\right) \cdot OPT$

<u>Thm</u>: Price of Anarchy $\leq O\left(\frac{\ln m}{\ln \ln m}\right)$

<u>Pf sketch</u>:

Let σ be mixed Nash

want to show: $\mathop{E}_{A \sim \sigma}\left[\max_{j \in [m]} \ell_j(A)\right] \leq O\left(\frac{\ln m}{\ln \ln m}\right) \cdot OPT$

Easier: bound $\max_{j \in [m]} \mathop{E}_{A \sim \sigma}\left[\ell_j(A)\right]$

<u>Lemma</u>: $\forall j \in [m]$, $\mathop{E}_{A \sim \sigma}\left[\ell_j(A)\right] \leq \left(2 - \frac{2}{n+1}\right) \cdot OPT$

<u>Pf sketch</u>: just like pure Nash!

- Let j^* machine with max expected load
- Let i^* job of min weight with nonzero probability of choosing j^*.

Now show concentration

Simple Chernoff bound:

Thm: If $X = X_1 + X_2 + \cdots + X_n$ where each X_i is an independent random variable in $[0,1]$, then

$$\Pr[X > t] \le 2^{-t} \quad \forall\, t > 2e\, E[X]$$

Fix machine $j \in [m]$

$$X_i = \begin{cases} \frac{w_i}{OPT} & \text{if} \quad A(i) = j \\ 0 & \text{if} \quad A(i) \neq j \end{cases}$$

$$X = \sum_{i=1}^{n} X_i = \sum_{i : A(i) = j} \frac{w_i}{OPT} = \frac{\ell_j(A)}{OPT}$$

$$\Rightarrow E[X] = \sum_{i=1}^{n} E[X_i] = \sum_{i=1}^{n} \frac{w_i}{OPT} \cdot \Pr[A(i) = j]$$

$$= \frac{1}{OPT} \cdot \sum_{i=1}^{n} w_i \cdot \Pr[A(i) = j]$$

$$= \frac{1}{OPT} \cdot E[\ell_j(A)]$$

$$\leq \frac{1}{OPT} \cdot 2 \cdot OPT = 2$$

Set $t = y \log m$, apply Chernoff:

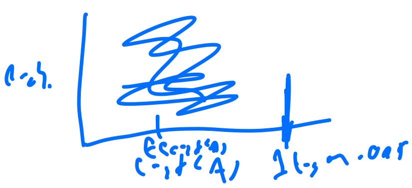
$$\Pr[X > t] = \Pr[X > y \log m] \leq 2^{-y \log m} = \frac{1}{m^y}$$

$$X = \frac{\ell_j(A)}{OPT} \Rightarrow \Pr[\ell_j(A) > y \log m \cdot OPT] \leq \frac{1}{m^y}$$

Union bound over all machines:

$$\Pr[cost(A) > y \cdot \log m \cdot OPT] \leq \sum_{j=1}^{m} \Pr[\ell_j(A) > y \log m \cdot OPT]$$

$$\leq m \cdot \frac{1}{m^y} = \frac{1}{m^{y-1}}$$



$$\Rightarrow E[cost(A)] \leq OPT \cdot \log m \cdot \Pr[cost(A) \leq OPT \cdot \log m]$$

$$+ \int_{OPT \log m}^{\infty} x \cdot \Pr[cost(A) = x]\, dx$$

$$\leq OPT \cdot \log m + \int_{1}^{\infty} (y \log m) \cdot OPT \cdot \frac{1}{m^{y-1}}\, dy$$

$\leq O(OPT \cdot \log m)$

To get $O\left(\frac{\ln m}{\ln\ln m}\right)$: slightly stronger version of Chernoff.