Tutorial 4

Exercise

Suppose we have 3 identical machines (speed 1) and 6 tasks, with weights: $w_1 = w_2 = w_3 = 1$, $w_4 = w_5 = w_6 = \frac{1}{3}$.
Compute the PoA for this instance of the load balancing game, by considering only pure strategies for the players.

Reminder:

$$PoA = \max_{G \in \mathcal{G}(m)} \max_{P \in Nash(G)} \frac{cost(P)}{Cost(OPT)}$$

$cost(P)$: $\max_{j \in [m]}(\ell_j)$ = maximum latency over all machines, also called "makespan of P"

$Cost(OPT)$: minimum makespan over all assignments, OPT doesn't have to be a N.E allocation

$G \in \mathcal{G}(m)$: for $m \in \mathbb{N}$, let $\mathcal{G}(m)$ denote the set of all instances of load balancing games with m machines

$P \in Nash(G)$: set of all strategy profiles that are NE (or PNE, depending on the question) of instance G

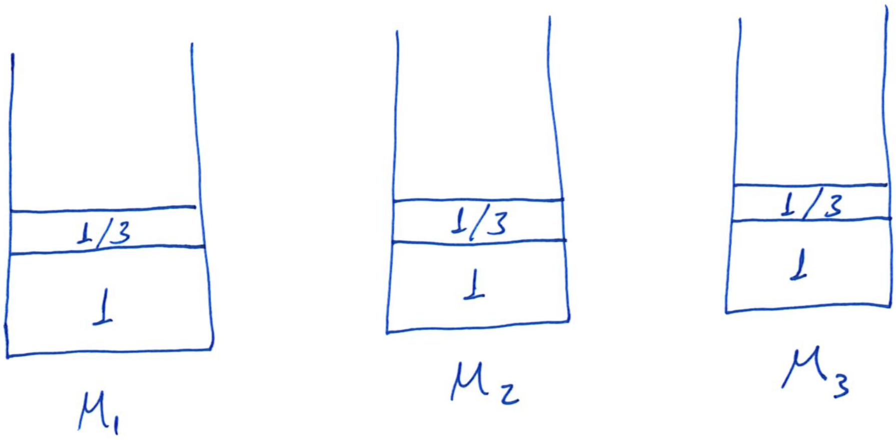
Remark: The players (tasks) try to minimize their personal cost (sum of weights of tasks that choose the machine of the player divided by the machine's speed) i.e. personal latency on their chosen machine (when they use mixed strategies it is personal expected cost (latency)).
Personal latency should not be confused with makespan.

* In the particular exercise we are only interested for the PoA of the specific instance given, thus, we can ignore the leftmost "max" of the PoA definition.

Solution

The optimal (we will call it "OPT") allocation (i.e. achieves minimum makespan) is:

OPT :



$$\text{cost}(OPT) = \max_{j\in[m]}(\ell_j) = \max\left\{\frac{4}{3}, \frac{4}{3}, \frac{4}{3}\right\} = \frac{4}{3}$$

Let us now find cost(P) for every allocation P that is a P.N.E.

Since the machines are identical, we will group all possible allocations of tasks in a smart way, according to the number of tasks on each machine, without caring about the identities of the machines (see below: $(M_1, M_2, M_3) = (0,6,0)$ is omitted because we already considered $(M_1, M_2, M_3) = (6,0,0)$.).

	# tasks on machines			
Group	M_1	M_2	M_3	
1	6	0	0	no P.N.E.
2	5	1	0	no P.N.E.
3	4	2	0	no P.N.E.
4	4	1	1	no P.N.E.
5	3	3	0	no P.N.E.
6	3	2	1	A_1 is the only P.N.E.
7	2	2	2	OPT is the only P.N.E.

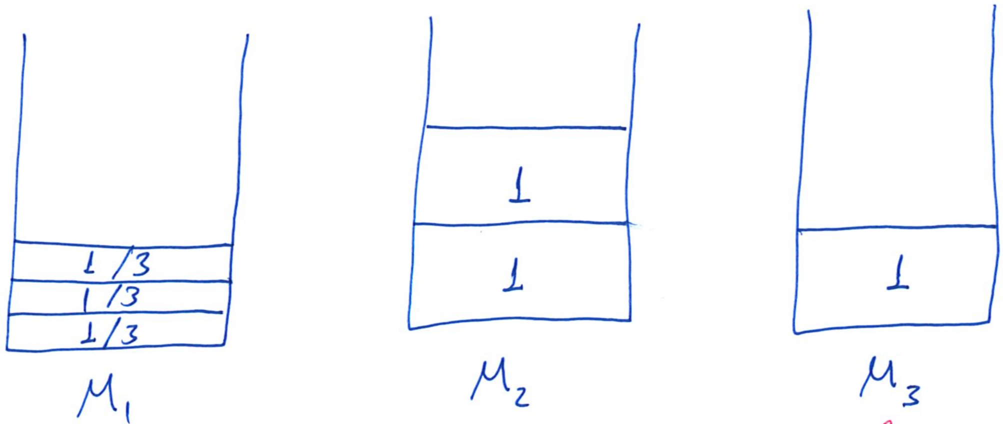
(see next page for explanation)

- The allocations in groups 1, 2, 3, 5 are not P.N.E. for the same reason: a machine with no task exists, and a task that is allocated with another one in an other machine would prefer to chose the empty machine and impove their personal cost.

- In every allocation of group 4, a player assigned to the heavy machine (with 4 tasks) has personal latency strictly greater than what she would have if she moved to one of the machines with 1 task (no matter the weight of the task).
  Check all allocations of this group as a homework.
  Therefore, non of them is a P.N.E.

- In the allocations of group 6, the only P.N.E. is the following allocation A_1 (Check as a homework the rest)

A_1:



Check here that no player (task) would prefer a unilateral deviation.

$$\text{cost}(A_1) = \max\{\tfrac{1}{3}+\tfrac{1}{3}+\tfrac{1}{3},\ 1+1,\ 1\} = 2$$

- The only P.N.E. allocation in group 7 is OPT.
  Check the rest as a homework.

We conclude that:

$$P.A = \max_{P\in\{A_1, OPT\}} \frac{\text{cost}(P)}{\text{cost}(OPT)} = \max\left\{\frac{\text{cost}(A_1)}{\text{cost}(OPT)}, \frac{\text{cost}(OPT)}{\text{cost}(OPT)}\right\} = \max\left\{\frac{2}{\frac{4}{3}}, 1\right\} = \boxed{\frac{3}{2}}$$

Exercise

Consider the following instance of the load balancing game:

- m identical machines $M_1, M_2, \ldots, M_m$, all of speed 1.
- n tasks $w_1, w_2, \ldots, w_n$.

Consider also the mixed strategy profile A, where each of the tasks is assigned to all machines equiprobably (i.e. with probability $\frac{1}{m}$).

Is the strategy profile A a N.E.?

Reminder: A strategy profile P is a N.E. iff $\forall i \in [n], \forall j \in [m]$:

$$p_i^j > 0 \Longrightarrow C_i^j \leq C_i^k \quad, \forall k \in [m]$$

expected cost of player i when she chooses machine j

Solution

Consider a player i with task w_i (arbitrary). Under strategy profile A, all of the actions $j \in [m]$ are in the support, since $p_i^j = \frac{1}{m} > 0 \; \forall j \in [m]$.

For an arbitrary machine j picked by player i, her cost will be:

$$C_i^j = w_i + \sum_{\substack{x=1 \\ x \neq i}}^{n} \frac{1}{m} w_x$$

and this is the same for any $j \in [m]$.

surely she will suffer her own load

expected load that she will find on j due to other players

Also, the above holds for every player $i \in [n]$, that is, for any player choosing a machine to surely assign her task yields the same latency for any machine. So, A is a N.E.