Envy-free Scheduling with Identical Machines

Resources

- As far as the exam is concerned, it is enough studying the following slides.

- However, if interested, you can find results contained in these slides and much more at this paper:

V. Bilò, A. Fanelli, M. Flammini, G. Monaco, L. Moscardelli: The Price of Envy-Freeness in Machine Scheduling. *Theoretical Computer Science.* Vol. 613, pp. 65—78, 2016.

Introduction

- So far, all the scheduling settings we have considered did not envisage fair allocations in which no machine prefers (or envies) the set of the tasks assigned to another machine, i.e., for which her completion time would be strictly smaller.
- Consider a scenario in which a company, in order to fulfill several jobs, has to engage a set of employees that have to be all paid out the same wage.
- For making the workers as productive as they can, it is required that no envy is induced.
- We will consider only envies among the "engaged" employees, i.e., the employees receiving at least one job.

Introduction (2)

- We consider the Envy-free Scheduling Identical Machine MAKESPAN (minimization) problem.

- From now on we suppose that in any schedule ***S*** all the machines get at least one job, (we can always remove machines getting no job).

<u>k-Envy-free</u> Scheduling Identical Machine MAKESPAN (minimization) problem.

- INPUT: *m identical* machine ($h = 1, \ldots, m$), *n* jobs ($i = 1, \ldots, n$), $p_i > 0$, parameter k≥1.

- OUTPUT: *k-envy free schedule* $\boldsymbol{S}=(S_{1,\ldots,}S_m)$ *such that* $MC_j(S) \le k * MC_{j'}(S)$ *for any machines j, j'.*

- GOAL: Minimizing the Machine MAKESPAN $Max_{h=1,\ldots,m}\{MC_h(S)\}$

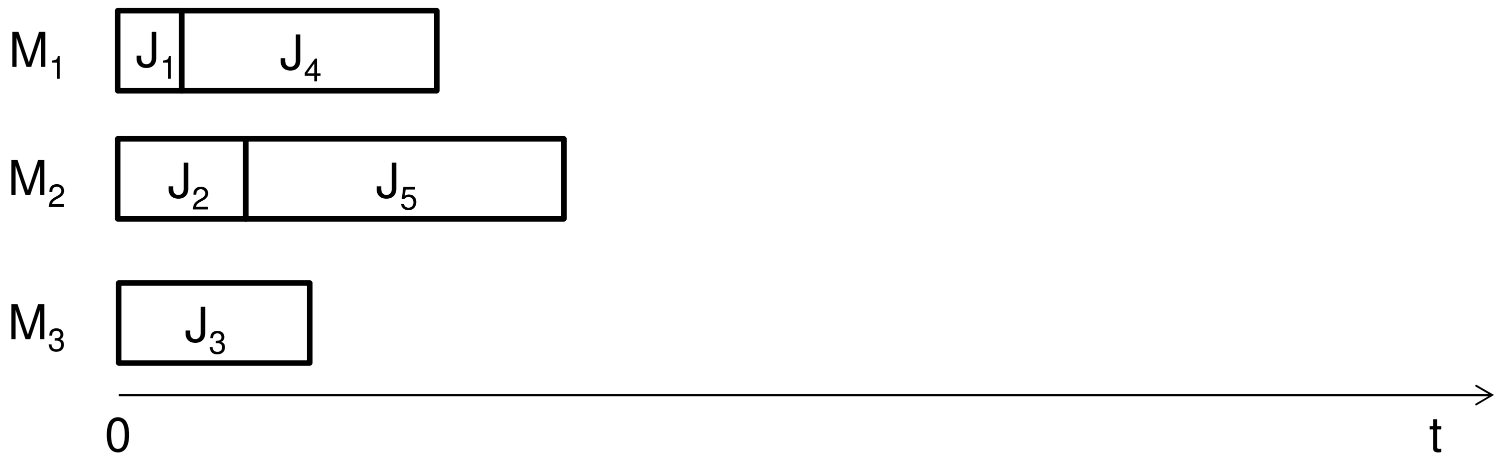
- If for some machine *j*, there exists a machine *j'* such that $MC_j(S) > k * MC_{j'}(S)$ we will say that machine *j* *envies* machine *j'* in the schedule *S*.

An example

- INPUT: 5 jobs, 3 machines.

$p_i=$

J_1	J_2	J_3	J_4	J_5
1	2	3	4	5



$MC_1(S)=1+4=5$; $MC_2(S)=2+5=7$; $MC_3(S)=3$

Is it a 1-envy free ? No: M_1 envies M_3; and M_2 envies M_1 and M_3

Is it a 2-envy free ? No: M_2 envies M_3; in fact $MC_2(S)=7>\underbrace{2}_{k}*\underbrace{3}_{MC_3(S)}=6$

Is it a 3-envy free ? Yes!

Then it is k-envy-free for any k≥3

An example (2)

- INPUT: 5 jobs, 3 machines.

p_i=

J_1	J_2	J_3	J_4	J_5
1	2	3	4	5

An 1-envy free schedule!

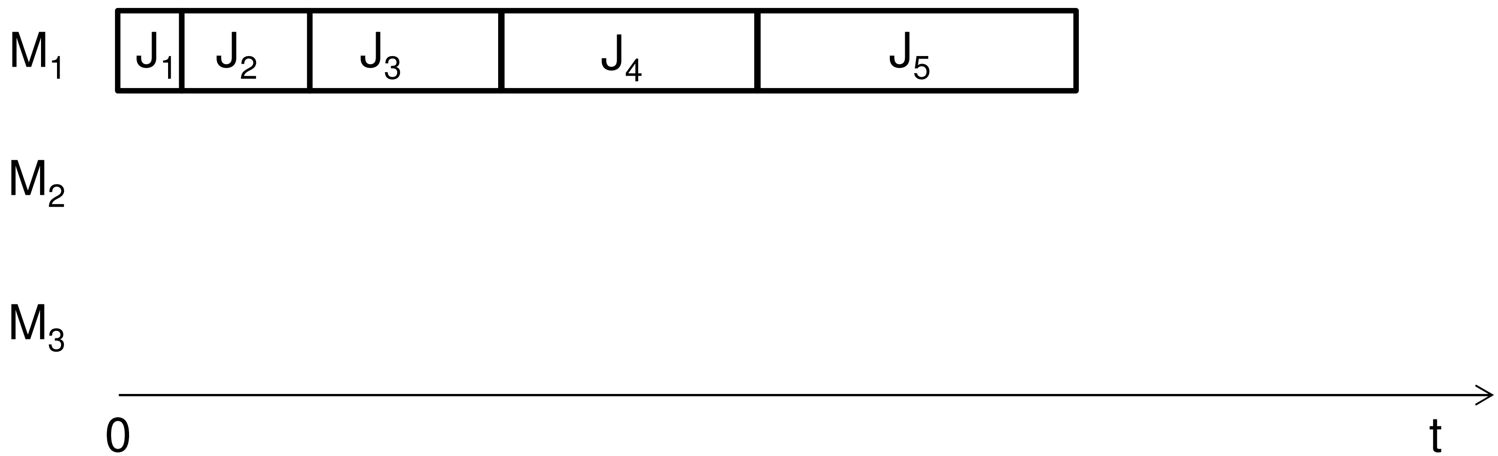
Notice that it is also an optimal solution.

An example (2)

- INPUT: 5 jobs, 3 machines.

$p_i =$

J_1	J_2	J_3	J_4	J_5
1	2	3	4	5



Another possible 1-envy free schedule!

Notice that a 1-envy-free solution (schedule) can be always obtained by assigning all jobs to one machine.
That is, k-envy-free solutions always exist! for any k≥1

Our goal

- We are interested in bounding the *Price of k-envy-freeness* that is the ratio between the MAKESPAN of the best k-envy-free schedule and the MAKESPAN of an optimal scheduling (non necessarily k-envy-free).

- The *Price of k-envy-freeness* is important because measures how much we lose by requiring that schedulings have to be k-Envy-free.

Price of k-envy-freeness

- Let k-*Envy-OPT* be the optimal solution for the problem with k-envy-free constraint and let *OPT* be an optimal solution to the problem without k-envy-free constraint.
- Moreover let C_{max}(k-*Envy-OPT)* (respectively C_{max}(*OPT)*) be the Machine MAKESPAN of the optimal k-*Envy-free* schedule (respectively of the optimal schedule *OPT* without the k-envy-free constraint).

- That is

$$C_{\max}(\text{k-Envy-OPT}) = Max_{h=1,\dots,m}\{MC_h(\text{k-Envy-OPT})\}$$

$$C_{\max}(OPT) = Max_{h=1,\dots,m}\{MC_h(OPT)\}$$

- Clearly we always have that $C_{\max}(k-Envy-OPT) \geq C_{\max}(OPT)$
- <u>Our Objective</u>: *we want to bound (in the worst case, i.e., for any possible instance of the problem) the following ratio for any value k≥1:*

$$\frac{C_{\max}(k-Envy-OPT)}{C_{\max}(OPT)}$$

<u>k-Envy-free</u> Scheduling Identical Machine MAKESPAN (minimization) problem.

- INPUT: *m identical* machine (*h = 1, . . .,m*), *n* jobs (*i = 1, . . . , n*), $p_i > 0$, parameter k≥1.

- OUTPUT: *k-envy free schedule **S**=($S_{1,\ldots,}S_m$) such that $MC_i(S) \le k*MC_j(S)$ for any machines i,j.*

- GOAL*:* Minimizing the Machine MAKESPAN $Max_{h=1,\ldots,m}\{MC_h(S)\}$

- <u>Our Objective</u>: *we want to bound the following ratio for any value k≥1 :*

$$\frac{C_{\max}(k-Envy-OPT)}{C_{\max}(OPT)}$$