Lesson 12.

Machine scheduling

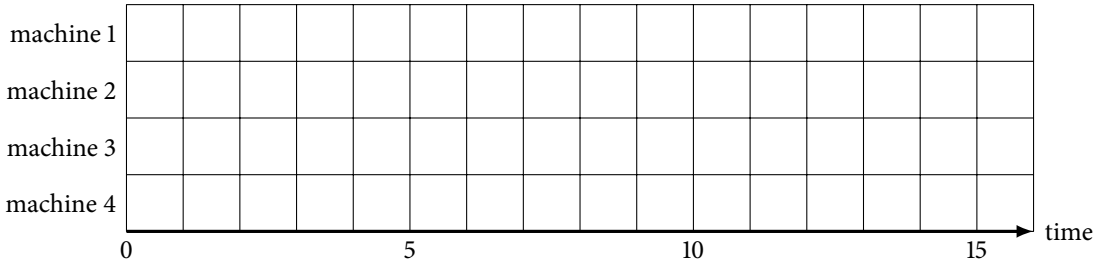
1 The problem

The Markov Micromanufacturing Company has 9 production jobs it needs to process in the next 24 hours. The company has 4 identical machines that run in parallel. Each of these 9 jobs must be run on one of these machines **nonpreemptively**: that is, once a job is started on a machine, it must stay on that machine until it is completed. The processing times of these jobs are given below:

job	1	2	3	4	5	6	7	8	9
processing time (hours)	7	7	6	6	5	5	4	4	4

The company wants to minimize the **makespan**, or the completion time of the last job to finish processing.

- Let m be the number of machines — in this case, $m = 4$
- Suppose we schedule the jobs using the **longest processing time first (LPT)** rule:
  - First, schedule the m longest jobs on the m machines
  - Whenever a machine becomes free, put the longest unprocessed job on that machine
- Idea: LPT puts shorter jobs towards the end of the schedule, where they can be used to balance the loads on each machine
- For our problem, this yields a schedule that looks like this:



- Therefore, the makespan for the LPT schedule is
- It turns out that the makespan of an LPT schedule is always at most $33.\overline{3}\%$ larger than the minimum makespan
- So... can we do better?
- Let's formulate this problem as a dynamic program

- Stages:
- States in stage t (nodes):
- Decisions, transitions, and rewards/costs at stage t (edges):
- Shortest/longest path?
- Minimum makespan $\longleftrightarrow$
- Assignments of jobs to machines $\longleftrightarrow$