Exam Sequencing and scheduling, 2P450,
January 21, 2011, 09:00-12:00

This test contains three questions on two pages, with 10 sub-items, for which a total of 50 points can be scored. Each good answer scores 5 points. The final grade is obtained by dividing the amount of points by five. Questions may be answered in English and/or in Dutch. Please explain your answers. You are NOT allowed to use the handouts distributed in class/or on the web.

Question 1. Consider the problem $1|r_j|L_{\max}$ of minimizing maximum lateness on a single machine under release dates. In particular consider the instance given by

jobs j	1	2	3	4	5	6	7
p_j	6	18	12	10	10	17	16
r_j	0	0	0	14	25	25	50
d_j	8	42	44	24	90	85	68

(a) Formulate and apply the *preemptive EDD rule*. Compute the optimal *preemptive* schedule, and list the results in a table with the following entries:

jobs j	1	2	3	4	5	6	7
C_j^{pEDD}							
d_j	8	42	44	24	90	85	68
L_j^{pEDD}							

(b) Prove optimality of your schedule by pointing out the correct *critical* job subset.

(c) Formulate and apply the *non-preemptive EDD rule*. List the results in a table with the following entries:

jobs j	1	2	3	4	5	6	7
C_j^{EDD}							
d_j	8	42	44	24	90	85	68
L_j^{EDD}							

(d) Formulate how the so-called α-schedule is obtained and compute it for this instance, for $\alpha = 0.5$. List the results in a table:

jobs j	1	2	3	4	5	6	7
α-point							
C_j^{α}							
d_j	8	42	44	24	90	85	68
L_j^{α}							

Is the result not contradicting the 'theorem' that $L_{\max}^{\alpha} \leq (\alpha + \frac{1}{\alpha}) L_{\max}^{\text{pEDD}}$?

Question 2. Consider the following flowshop problem: we have m machines, and n jobs that need to be processed. Job j consists of subtasks T_{ij}, for $i = 1, \ldots, m$, needing processing on machine M_i for a period of time p_{ij}. The flowshop characteristic entails that T_{ij} must be finished before $T_{(i+1)j}$ can be processed. A machine can handle one job at a time, and tasks cannot be preempted. The objective is to minimize the *maximum* job-completion time $\max_{j=1}^{n} C_{mj}$.

(a) Prove that when looking for an optimal schedule, we may restrict ourselves to schedules in which the job order on the first machine is equal to the job order on the second machine;

(b) Consider the following instance of $F2||C_{\max}$:

job j	1	2	3	4	5	6	7	8
$p_{1,j}$	6	6	5	4	4	3	2	1
$p_{2,j}$	5	7	5	1	3	2	3	3

Explain the rule by which such problem is solved to optimality and apply it on this instance.

(c) Consider the flow shop on four machines, with the regular objective of minimizing makespan. Is there always an optimal solution in which the job-orders on all machines are equal to the order on the first machine? If yes, give a proof; if not, give a counter-example.

Question 3. Consider a problem with m parallel machines.

(a) Minimize makespan. Prove that application of the so-called *List-schedule rule*, you will find a schedule that is guaranteed a makespan not larger than $2 - \frac{1}{m}$ times the optimal makespan.

(b) Minimize average completion time, assume that no two jobs have the same processing time. Prove that in an optimal schedule on each machine, jobs are scheduled in order of increasing processing time; and prove that the number of jobs on any two machines $M1$ and $M2$ say, do not differ by more than one. What is the practical rule to solve this problem?

(c) Minimize weighted sum of completion times, where the machines are *uniform*, with speeds $s_1, \ldots, s_m$. Formulate how to find the schedule with minimum weighted sum of completion times.