Sequencing :

Sequencing

Processing Time, t | Due Date, d

Lateness (+ve or −ve), L | Tardiness (measure of Positive Lateness), T = Max(0,L)

Flow Time, F_i = Span between, task i Available for Processing & time at which Job i is completed

Completion Time, C_i = Span between beginning of 1st Job and when Job, i is Finished

If all Jobs are available at t=0, then $C_i = F_i$

Makespan = Span of Time when we start working on the 1st Job on the 1st Machine till we finish working on the Last Job on the Last Machine

Makespan for n tasks in Sequence s, $M_s = \sum_{i=1}^{n} t_i$

Mean Flow Time is Sequence s, $\overline{F_s} = \frac{1}{n}\sum_{i=1}^{n} F_{i,s}$

$L_{i,s} = C_{i,s} - d_i \mid T_{i,s} = Max\{0, L_{i,s}\} = Max\{0, C_{i,s} - d_i\}$

Mean Lateness in Sequence s, $\overline{L_s} = \frac{1}{n}\sum_{i=1}^{n} L_{i,s}$ | Mean Tardiness in Sequence s, $\overline{T_s} = \frac{1}{n}\sum_{i=1}^{n} T_{i,s}$

No. of Tardy Jobs, $N_t = \sum_{i=1}^{n} \delta_i$, where $\delta_i = 1$ *if* $T_i > 0, \delta_i = 0$ *otherwise*

$T_{Max} = \max\{0, L_{Max}\} \mid L_{Max} = \max\{L_{i,s}\} \forall i \text{ in } n$

Sequencing *n* jobs on one machine

Example

All tasks are available at t=0

Task i	Proc. time t_i	Due date d_i	Flow time F_i	Lateness L_i
1	5	15	5	-10
2	8	10	13	3
3	6	15	19	4
4	3	25	22	-3
5	10	20	32	12
6	14	40	46	6
7	7	45	53	8
8	3	50	56	6

$\overline{F_s} = 30.75$ $\overline{L_s} = 3.25$

Sequencing *n* jobs on one machine

Use Shortest Processing time (SPT) Rule

Task i	t_i	d_i	F_i	L_i
4	3	25	3	-22
8	3	50	6	-44
1	5	15	11	-4
3	6	15	17	2
7	7	45	24	-21
2	8	10	32	22
5	10	20	42	22
6	14	40	56	16

$\overline{F_s} = 23.875$ $\overline{L_s} = -3.625$

Sequencing *n* jobs on one machine

Minimize Maximum Lateness

Use EDD (earliest due date) Rule

Task i	t_i	d_i	F_i	L_i
2	8	10	8	-2
1	5	15	13	-2
3	6	15	19	4
5	10	20	29	9
4	3	25	32	7
6	14	40	46	6
7	7	45	53	8
8	3	50	56	6

Sequencing n Jobs in One Machine

Hodgson's Algorithm (Minimize the number of Tardy Jobs)

Example

Task	1	2	3	4	5	6	7	8
t_i	5	8	6	3	10	14	7	3
d_i	15	10	15	25	20	40	45	50

Arrange the tasks in the order of Earliest Due Date rule

Task	2	1	3	5	4	6	7	8
t_i	8	5	6	10	3	14	7	3
C_i	8	13	19	29	32	46	53	56
d_i	10	15	15	20	25	40	45	50
L_i	-2	-2	4	9	7	6	8	6

Set aside Task 2 (since it is the 1st task with positive Lateness from the Left)

Task	1	3	5	4	6	7	8
t_i	5	6	10	3	14	7	3
C_i	5	11	21	24	38	45	48
d_i	15	15	20	25	40	45	50
L_i	-10	-14	1	-1	-2	0	-2

Set aside Task 5 (since it is the 1st task with positive Lateness from the Left)

Task	1	3	4	6	7	8	2	5
t_i	5	6	3	14	7	3	8	10
C_i	5	11	14	28	35	38	46	56
d_i	15	15	25	40	45	50	10	20
L_i	-10	-4	-11	-12	-10	-12	36	36

Only 2 Tardy Jobs

Sequencing n Jobs in Two Machine

Johnson's Algorithm

- Create a list of processing times of all jobs on machine 1 (*M1*) and machine 2 (*M2*).
- Identify the shortest processing time in this list. Break ties arbitrarily.
- If the shortest processing time is on *M1*, then assign the corresponding job to the next available position starting at the beginning of the sequence. Go to step 4. If it is on *M2*, then assign the corresponding job to the next available position starting from the end of the sequence. Go to step 4. .
- Remove the assigned job from the list. Repeat steps 2 and 3 until all jobs are assigned.

Sequencing n Jobs in Three Machine

Convert this into a Two Machine Problem

$Machine\ 1' = Machine\ 1 + Machine\ 2$

$Machine\ 2' = Machine\ 2 + Machine\ 3$

Condition for Optimality

The solution to the three machine problem will be optimal using the above method if

Either $Min\ T_{i_1} \geq Max\ t_{i_2}$ or $Min\ T_{i_3} \geq Max\ t_{i_2}$ is satisfied