Production and Operations Management Systems

Presented by,
K.Sudha
Associate Professor/CSE
MEC (Autnomous)

After viewing this presentation, you should be able to:

- Explain why production scheduling must be done by every organization whether it manufactures or provides services.
- Discuss the application of the loading function.
- Draw a Gantt chart and explain its information display.
- Describe the role of sequencing and how to apply sequencing rules for one facility and for more than one facility.

After viewing this presentation, you should be able to (continued)

- Classify scheduling problems according to various criteria that are used in practice.
- Explain the purpose of priority sequencing rules.
- Describe various priority rules for sequencing.
- Apply Johnson’s rule to the 2-machine flow shop problem.
- Analyze dynamic scheduling problems.

Loading, Sequencing and Scheduling

The production-schedules are developed by performing the following functions:

- *Loading*
- *Sequencing*
- *Scheduling*

Loading, Sequencing and Scheduling
(continued)

Loading: Which department is going to do what work?

Sequencing: What is the order in which the work will be done?

Scheduling: What are the start and finish times of each job?

Loading

Loading, also called shop loading assigns the work to various facilities like divisions, departments, work centers, load centers, stations, machines and people.

We will often use the term “machines” in this presentation when we refer to a facility.

Loading is done for both manufacturing and services.

Loading vs. Aggregate Planning

Aggregate planning is based on forecasts.

However, the loading function loads the real jobs and not the forecast.

If the aggregate scheduling job was done well, then the appropriate kinds and amounts of resources will be available for loading.

Loading Objectives

Each facility carries a backlog of work, which is its “load”—hardly a case of perfect just-in-time in which no waiting occurs.

The backlog is generally much larger than the work in process, which can be seen on the shop floor.

The backlog translates into an inventory investment which is idle and receiving no value-adding attention.

A major objective of loading is to spread the load so that waiting is minimized, flow is smooth and rapid, and congestion is avoided.

Sequencing

Sequencing models and methods follow the discussion of loading models and methods.

Sequencing establishes the order for doing the jobs at each facility.

Sequencing reflects job priorities according to the way that jobs are arranged in the queues.

Say that Jobs x, y, and z have been assigned to workstation 1 (through loading function).

Jobs x, y, and z are in a queue (waiting line). Sequencing rules determine which job should be first in line, which second, etc.

Sequencing (continued)

A good sequence provides less waiting time, decreased delivery delays, and better due date performance.

There are costs associated with waiting and delays.

There are many other costs associated with the various orderings of jobs, for example, set up cost and in-process inventory costs.

The objective function can be to minimize system's costs, or to minimize total system's time, or (if margin data are available) to maximize total system's profit.

We discuss-several objective functions later in the presentation.

Sequencing (continued)

Total savings from regularly sequencing the right way, the first time, can accumulate to substantial sums.

Re-sequencing can be significantly more costly. When there are many jobs and facilities, sequencing rules have considerable economic importance.

Sequencing also involves shop floor control, which consists of communicating the status of orders and the productivity of workstations.

Scheduling

A production schedule is the time table that specifies the times at which the jobs in a production department will be processed on various machines.

.

The schedule gives the starting and ending times of each job on the machines on which the job has to be processed.

Scheduling Example

Suppose there are three jobs in a production department that are to be processed on four categories (types) of machines. We designate the jobs as A, B, and C; and the machine types are designated as M1, M2, M3, and M4.

The three jobs consist of 4, 3, and 4 operations respectively; and there are four machines - one machine of each type. We designate them as M1, M2, M3, and M4 based on their categories.

The operations for job A are designated as A1, A2, A3, and A4. The operations of job B are designated as B1, B2, and B3. Similarly the four operations of job C are designated as C1, C2, C3, and C4.

Scheduling Example (continued)

Each job is characterized by its routing that specifies the information about the number of operations to be performed, the sequence of these operations, and the machines required for processing these operations.

The times required for processing these operations are also required for developing a production schedule.

Scheduling Example - Data

The table on right hand side (RHS) gives the data for this example.

The table gives the machine required for each operation of each job. For example, the first operation of job A, A1, is processed on machine M1; second operation, A2, is processed on machine M3 and so on.

The operations of all jobs have to follow their processing sequences. For example operation A3 of job A can not be processed before operation A2.

The processing time for each operation is also given in this table.

Job	Operation Number	Machine Number	Processing Time (Days)
A	A1	M1	5
	A2	M3	3
	A3	M4	7
	A4	M2	4
B	B1	M2	2
	B2	M3	6
	B3	M4	8
C	C1	M1	4
	C2	M2	6
	C3	M3	8
	C4	M4	2

Scheduling Example – Objective Function

The objective is to schedule these jobs so as to minimize the time to complete all jobs. This time is called *make-span* or the *schedule time*. We will use the term *make-span* in this presentation.

Scheduling Example Solution – Gantt Chart

One of the schedules for this example is presented below in the form of a Gantt Chart.
The Gantt chart, for each machine, shows the start and finish times of all operations scheduled on that machine.

	Time (Days)																													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
M1	A1	A1	A1	A1	A1	C1	C1	C1	C1																					
M2	B1	B1								C2	C2	C2	C2	C2	C2	A4	A4	A4	A4											
M3						A2	A2	A2	B2	B2	B2	B2	B2	B2		C3	C3	C3	C3	C3	C3	C3	C3							
M4									A3	A3	A3	A3	A3	A3	A3	B3	B3	B3	B3	B3	B3	B3	B3	C4	C4					

Scheduling Example - Alternative schedules

Several alternative schedules can be generated for this example.
The schedules differ in the order in which the jobs are processed on the four machines. Three of these schedules are:

- The first schedule orders jobs as: A first, then B and then C (A-B-C).
- The second schedule orders jobs as: B first, then A, and then C (B-A-C).
- The third schedule orders jobs as: C first, then A, and then B (C-A-B).

The Gantt charts for these schedules are shown in next slide.

The values of make-span for these three schedules are 25, 27 and 30 days respectively. Schedule A-B-C is the best of these three schedules.

Scheduling Example - Alternative Schedules (continued)

Time (Days)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
M1	A1	A1	A1	A1	A1	C1	C1	C1	C1																					
M2	B1	B1								C2	C2	C2	C2	C2	C2	A4	A4	A4	A4											
M3						A2	A2	A2	B2	B2	B2	B2	B2	B2		C3	C3	C3	C3	C3	C3	C3	C3							
M4									A3	A3	A3	A3	A3	A3	A3	B3	B3	B3	B3	B3	B3	B3	B3	C4	C4					

Sequence A-B-C (Make-span = 25 days)

Time (Days)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
M1	A1	A1	A1	A1	A1	C1	C1	C1	C1																					
M2	B1	B1								C2	C2	C2	C2	C2	C2									A4	A4	A4	A4			
M3			B2	B2	B2	B2	B2	B2	A2	A2	A2					C3	C3	C3	C3	C3	C3	C3	C3							
M4									B3	B3	B3	B3	B3	B3	B3	B3	A3	A3	A3	A3	A3	A3	A3	C4	C4					

Sequence B-A-C (Make-span = 27 days)

Time (Days)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
M1	C1	C1	C1	C1	A1	A1	A1	A1	A1																					
M2	B1	B1			C2	C2	C2	C2	C2	C2										A4	A4	A4	A4							
M3			B2	B2	B2	B2	B2	B2		A2	A2	A2	C3	C3	C3	C3	C3	C3	C3	C3										
M4													A3	A3	A3	A3	A3	A3	A3		C4	C4	B3	B3	B3	B3	B3	B3	B3	B3

Sequence C-A-B (Make-span = 30 days)

Scheduling Example - Alternative Schedules (continued)

Is sequence A-B-C the global optimal? Can we find a better sequence than this? The scheduling techniques attempt to answer these questions.

It should be mentioned that there are different effectiveness measures of a schedule in different situations. Minimizing make-span is only one of them. We will study other effectiveness measures also.

Scheduling Example - Assumptions

Once a job is started on a machine, its processing can not be interrupted, that is, preemption is not allowed.

The machines are continuously available and will not break down during the planning horizon. This assumption is rather unrealistic but we make this assumption to avoid complexity in discussing scheduling concepts.

A machine is not kept idle if a job is available to be processed.

Also, each machine can process only one job at a time.

Classification of Scheduling Problems

The scheduling problems can be classified based on the following criteria:

- Sequence of machines
- Number of machines
- Processing times
- Job arrival time
- Objective functions

Sequence of Machines

The sequencing problems, based on the sequence of machines, are classified as:

- Flow Shops
- Job Shops

Flow Shop

In a flow-shop , processing of all jobs require machines in the same order.

The following table gives an example of a flow-shop in which three jobs, A, B, and C are processed on four machines, M1, M2, M3, and M4.

The sequences of machines to process these jobs are same (M1-M3-M4-M2).

Example of a Flow Shop								
Job	Operation # 1	Operation # 2	Operation # 3	Operation # 4	Machine for Operation # 1	Machine for Operation # 2	Machine for Operation # 3	Machine for Operation # 4
A	A1	A2	A3	A4	M1	M3	M4	M2
B	B1	B2	B3	B4	M1	M3	M4	M2
C	C1	C2	C3	C4	M1	M3	M4	M2

Job Shop

In a job shop the sequence of machines will be mixed, that is, the jobs may require machines in different sequences.

Example of a Job Shop								
Job	Operation # 1	Operation # 2	Operation # 3	Operation # 4	Machine for Operation # 1	Machine for Operation # 2	Machine for Operation # 3	Machine for Operation # 4
A	A1	A2	A3	A4	M1	M3	M4	M2
B	B1	B2	B3		M2	M3	M4	
C	C1	C2	C3	C4	M1	M2	M3	M4

Number of Machines

Based on the number of machines, the scheduling problems are classified as:

- Single machine problems
- Two-machine problems
- Multiple (3 or more) machine problems

Processing Times

- *Deterministic:* If processing times of all jobs are known and constant the scheduling problem is called a deterministic problem.

- *Probabilistic:* The scheduling problem is called probabilistic (or stochastic) if the processing times are not fixed; i.e., the processing times must be represented by a probability distribution.

Job Arrival Times

Based on this criterion, scheduling problems are classified as static and dynamic problems.

- *Static:* In the case of static problems the number of jobs is fixed and will not change until the current set of jobs has been processed.

- *Dynamic:* In the case of dynamic problems, new jobs enter the system and become part of the current set of unprocessed jobs. The arrival rate of jobs is given in the case of dynamic problems.

Objective Functions

Scheduling researchers have studied a large variety of objective functions. In this presentation, we will focus on the following objectives.

- Minimize make-span
- Minimize average flow time (or job completion time)
- Average number of jobs in the system
- Minimize average tardiness
- Minimize maximum tardiness
- Minimize number of tardy jobs

Objective Functions (continued)

Minimizing make span is relevant for two or more machines.

In this presentation we will discuss the scheduling rule for static and deterministic flow shop problems consisting of two machines where the objective is to minimize make-span.

The other five objectives can be used for any number of machines, both deterministic and probabilistic processing times, and for static as well as dynamic problems.

However, we will study these objective functions for a single machine, deterministic and static problems.

The scheduling rule for job shops and for more than three machines are complex and beyond the scope of this presentation.

Example: 2-Machines Flow Shop

Consider a problem with five jobs (A, B, C, D, and E); and two machines designated as M1 and M2.

All five jobs consist of two operations each. The first operation of each job is processed on machine M1; and the second operation is processed on machine M2.

The next slide gives the machines required for each job; and the processing times for each operation of each job.

Data for a 2-Machine Flow Shop

Data for a 5-Job 2-Machine Flow Shop Problem

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operation # 2 (Days)
A	A1	A2	M1	M2	8	3
B	B1	B2	M1	M2	5	7
C	C1	C2	M1	M2	6	9
D	D1	D2	M1	M2	7	1
E	E1	E2	M1	M2	4	6

Scheduling Objective

The scheduling objective is to find an optimal sequence that gives the order in which the five jobs will be processed on the two machines.

Once we know a sequence, the time to complete all jobs can be determined. This time is called the schedule time, or the make-span. The optimal sequence is the one that minimizes make-span.

For example, A-B-C-D-E is a sequence order that tells us that A is the first job to be processed; B is the second job and so on. E is the last job to be processed. Is it optimal?

Another sequence could be B-C-A-E-D. Is it optimal?

What is the optimal sequence?

Number of Sequences

For this 5-job problem there are 120 (5!) different sequences. For a six-job problem, the number of sequences will be 720 (6!).

In general, for a "*n*" job problem there are *n!* (*n*-factorial) sequences.

Our goal is to find the best sequence that minimizes make-span.

Let us find the make-span for one of these sequences, say A-B-C-D-E. We will draw a Gantt chart to find make-span.

Gantt Chart
Sequence A-B-C-D-E

The Gantt chart for the sequence A-B-C-D-E is given below. We are assuming that the sequence is the same on both machines.

The value of make-span (time to complete all jobs) is 36 days.

Our objective is to identify the sequence that minimizes the value of make-span.

Gantt Chart for Sequence A-B-C-D-E																																				
Time (Days)																																				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
M1	A1	A1	A1	A1	A1	A1	A1	A1	B1	B1	B1	B1	B1	C1	C1	C1	C1	C1	C1	D1	D1	D1	D1	D1	D1	D1	E1	E1	E1	E1						
M2									A2	A2	A2			B2	B2	B2	B2	B2	B2	B2	C2	C2	C2	C2	C2	C2	C2	C2	C2	D2	E2	E2	E2	E2	E2	E2

Identifying the best sequence

There may be multiple optimal sequences.
We will study Johnson's rule that identifies one of these optimal sequence.

There are five sequence positions 1 through 5.
Johnson's rule assigns each job to one of these positions in an optimal manner.

Position 1	Position 2	Position 3	Position 4	Position 5

Johnson's Rule to Minimize Make-span

We use the following four step process to find the optimal sequence.

Step 1: Find the minimum processing time considering times on both machines.

Step 2: Identify the corresponding job and the corresponding machine for the minimum time identified at Step 1.

Johnson's Rule (continued)

Step 3: Scheduling Rule

(a) If the machine identified in Step 2 is machine M1 then the job identified in Step 2 will be scheduled in the first available schedule position.

(b) If the machine identified in Step 2 is machine M2 then the job identified in Step 2 will be scheduled in the last available schedule position.

Step 4: Remove the job from consideration whose position has been fixed in Step 3; and go to Step 1.

Continue this process until all jobs have been scheduled.

Johnson's Rule (continued)

Johnson's rule makes the following assumptions:

- The same optimal sequence is used on both machines.
- Preemption is not allowed, that is, once a job is started it is not interrupted.

Iteration 1

Step 1: The minimum time is 1.

Step 2: The job is D and the machine is M2.

Step 3: Since the machine identified at Step 2 is machine M2, job D will be assigned to the last available sequence position which is position 5; and the resulting partial sequence is given below.

Position 1	Position 2	Position 3	Position 4	D

Step 4: Delete job D from consideration.

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operation # 2 (Days)
A	A1	A2	M1	M2	8	3
B	B1	B2	M1	M2	5	7
C	C1	C2	M1	M2	6	9
D	D1	D2	M1	M2	7	1
E	E1	E2	M1	M2	4	6

Iteration 2

Step 1: The next minimum time is 3.

Step 2: The job is A and the machine is M2.

Step 3: The job A will be assigned to the last available schedule position, which is position 4. After assigning job A to position 4, the partial sequence is given below.

Position 1	Position 2	Position 3	A	D

Step 4: Delete job A from consideration.

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operation # 2 (Days)	
A	A1	A2	M1	M2	8	3	
B	B1	B2	M1	M2	5	7	
C	C1	C2	M1	M2	6	9	
~~D~~	~~D1~~	~~D2~~	~~M1~~	~~M2~~	~~7~~	~~1~~	Scheduled
E	E1	E2	M1	M2	4	6	

Iteration 3

Step 1: The minimum time is 4.
Step 2: The job is E and the machine is M1.
Step 3: The job E will be assigned to the first available schedule position, which is position 1. The partial sequence after assigning job E to position 1 is given below.

E	Position 2	Position 3	A	D

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operation # 2 (Days)	
~~A~~	~~A1~~	~~A2~~	~~M1~~	~~M2~~	~~8~~	~~3~~	Scheduled
B	B1	B2	M1	M2	5	7	
C	C1	C2	M1	M2	6	9	
~~D~~	~~D1~~	~~D2~~	~~M1~~	~~M2~~	~~7~~	~~1~~	Scheduled
E	E1	E2	M1	M2	4	6	

Step 4: Delete job E from consideration

Iteration 4

Step 1: The minimum time is 5.
Step 2: The job is B and the machine is M1.
Step 3: The job B will be assigned to the first available schedule position, which is position 2. The partial sequence after assigning job B to position 2 is given below.

E	B	Position 3	A	D

Step 4: Delete job B from consideration

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operation # 2 (Days)	
~~A~~	~~A1~~	~~A2~~	~~M1~~	~~M2~~	~~8~~	~~3~~	Scheduled
B	B1	B2	M1	M2	5	7	
C	C1	C2	M1	M2	6	9	
~~D~~	~~D1~~	~~D2~~	~~M1~~	~~M2~~	~~7~~	~~1~~	Scheduled
~~E~~	~~E1~~	~~E2~~	~~M1~~	~~M2~~	4	~~6~~	Scheduled

Iteration 5

The only unscheduled job at this stage is C and; it will be assigned to the remaining unassigned position 3.

The final sequence is given below. The value of make-span for this sequence will be determined by drawing the Gantt chart.

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operation # 2 (Days)	
~~A~~	~~A1~~	~~A2~~	~~M1~~	~~M2~~	~~8~~	~~3~~	Scheduled
~~B~~	~~B1~~	~~B2~~	~~M1~~	~~M2~~	~~5~~	~~7~~	Scheduled
C	C1	C2	M1	M2	6	9	
~~D~~	~~D1~~	~~D2~~	~~M1~~	~~M2~~	~~7~~	~~1~~	Scheduled
~~E~~	~~E1~~	~~E2~~	~~M1~~	~~M2~~	4	~~6~~	Scheduled

E	B	C	A	D

Finding Make-span

The sequence E-B-C-A-D identified by Johnson's rule guarantees the minimum value of make-span.

However, Johnson's rule does not give the value of make-span. It only identifies the best sequence.

The value of make-span is obtained either by drawing the Gantt chart or a computerized algorithm can be developed.

The Gantt chart for this optimal sequence is given in the next slide.

Sequence E-B-C-A-D

The Gantt chart for the sequence E-B-C-A-D is given below. The value of make-span is 31 days.

The optimal (minimum) value of make-span for this problem is therefore, 31 days.

Gantt Chart for Sequence E-B-C-A-D																																				
Time (Days)																																				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
M1	E1	E1	E1	E1	B1	B1	B1	B1	B1	C1	C1	C1	C1	C1	C1	A1	A1	A1	A1	A1	A1	A1	A1	D1	D1	D1	D1	D1	D1	D1						
M2					E2	E2	E2	E2	E2	E2	B2	B2	B2	B2	B2	B2	B2	C2	C2	C2	C2	C2	C2	C2	C2	C2	A2	A2	A2		D2					

Multiple Sequences

It may be noted that multiple optimal sequences are possible for a given problem. It means that several sequences can have the same minimum value of make-span.

For example, for the problem studied in previous slides, the sequence E-C-B-A-D also gives a make-span of 31 days. TRY IT.

However, Johnson's rule identifies only one of these sequences.

Breaking Ties

It might happen at Step 1, that there are more than one minimum times. In such a situation, which job should be picked up for assignment.

We will discuss three different cases of these ties.

Case 1: Minimum time is on both machines but for different jobs.

Case2: Minimum time is on the same machine but for different jobs.

Case 3: Minimum time is on both machines but for the same job.

Ties: Case 1 - Data

Consider the problem given below. The minimum time is 1 but occurs at two places – Job A at M1 and Job D at M2.

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operatio n # 2 (Days)
A	A1	A2	M1	M2	1	3
B	B1	B2	M1	M2	5	7
C	C1	C2	M1	M2	6	9
D	D1	D2	M1	M2	7	1
E	E1	E2	M1	M2	4	6

Ties: Case 1 - Solution

The ties are broken at random.

A may be selected before D or D may be selected before A.

In either case, the resulting partial sequence after both A and D are scheduled, is given below.

The scheduling algorithm continues until all jobs are scheduled. The final sequence is also shown below.

Partial scequence after scheduling A and D				
A	Position 2	Position 3	Position 4	D
Final sequence				
A	E	B	C	D

Ties: Case 2 - Data

Consider the problem given below. The minimum time is 1 but occurs at two places – Job B at M2 and Job D at M2.

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operatio n # 2 (Days)
A	A1	A2	M1	M2	8	3
B	B1	B2	M1	M2	5	1
C	C1	C2	M1	M2	6	9
D	D1	D2	M1	M2	7	1
E	E1	E2	M1	M2	4	6

Ties: Case 2 – Solution

The ties are broken at random.

B may be selected before D or D may be selected before B.

In this case, two different partial sequences will result based on which job is selected first. These partial sequences are shown below.

The scheduling algorithm continues until all jobs are scheduled.

The two final sequences are also shown below.

Partial Sequence if B is selected first and then D				
Position 1	Position 2	Position 3	D	B
Final sequence if B is selected first and then D				
E	C	A	D	B
Partial Sequence D is selected first and then B				
Position 1	Position 2	Position 3	B	D
Final sequence if D is selected first and then B				
E	C	A	B	D

Ties: Case 3 - Data

Consider the problem given below. The minimum time is 2 but occurs at two places – Job C at M1 and also Job C at M2.

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operatio n # 2 (Days)
A	A1	A2	M1	M2	8	3
B	B1	B2	M1	M2	5	7
C	C1	C2	M1	M2	2	2
D	D1	D2	M1	M2	7	9
E	E1	E2	M1	M2	4	6

Ties: Case 3 - Solution

The ties are broken at random.

Job C at M1 may be selected before Job C at M2 or vice versa.

In this case, two different partial sequences will result based on which combination is selected first. These partial sequences are shown below.

The scheduling algorithm continues until all jobs are scheduled.
The two final sequences are also shown below.

Note: C can be the first job or the last job in the optimal sequence

Partial Sequence if C on M1 is selected first				
C	Position 2	Position 3	Position 4	Position 5
Final sequence if C on M1 is selected first				
C	E	B	D	A
Partial Sequence if C on M2 is selected first				
Position 1	Position 2	Position 3	Position 4	C
Final sequence if C on M2 is selected first				
E	B	D	A	C

Comments About Ties

In general all three cases of ties may exist in a given problem.

The examples that we considered show ties in the first iteration.

However, the ties may occur during any iteration.

The general rule is to break the ties at random. However, we will break the ties in the alpha order, that is A before B etc. For case 3, the ties will be broken by the rule machine M1 before machine M2.

All resulting sequences, irrespective of the tie breaking rule, will give the same minimum value of the make-span.

Example with Multiple Ties

Consider the problem given below.

Johnson's rule will give four different sequences for this problem because of various ties.

This problem is included on students' website.

TRY IT.

Job	Operation # 1	Operation # 2	Machine for Operation # 1	Machine for Operation # 2	Time for Operation # 1 (Days)	Time for Operatio n # 2 (Days)
A	A1	A2	M1	M2	2	3
B	B1	B2	M1	M2	5	7
C	C1	C2	M1	M2	2	2
D	D1	D2	M1	M2	8	9
E	E1	E2	M1	M2	4	2

Single Machine Scheduling

There is one machine on which several jobs have to be processed.

The order in which these jobs will be processed needs to be specified. This schedule will not be changed until all jobs have been processed. This is the "static" version of the problem.

In the "dynamic" version, the schedule can be altered. The dynamic version is studied later in this presentation.

Scheduling Rules

There are several rules that can be used to find the order of processing. We will study the following three rules.

- First Come First Served (FCFS)
- Shortest Processing Time (SPT). This is also called as Shortest Operation Time.
- Earliest (shortest) Due Date (EDD)

Objective Functions

There are several objective functions that can be minimized in a single machine problem. We will study the following objective functions.

- ❖Minimize average completion (flow) time.
- ❖Minimize average number of jobs in the system.
- ❖Minimize average tardiness.
- ❖Minimize maximum tardiness
- ❖Minimize number of tardy jobs.

Note: A job is tardy if it is not completed by its due date.

Example

Consider the example given on the RHS.

There are five jobs A, B, C, D, and E. A is the first job that arrived in the production department. B,C, D, and E followed A in this order.

The processing times and due dates of all jobs are also given.

The order in which these jobs have to be processed needs to be specified.

	Days	
Job	Time	Due Date
A	17	45
B	12	35
C	22	27
D	18	54
E	26	47

First Come First Served (FCFS)

The table on RHS gives answers by using the FCFS rule. The order of processing is A, B, C, D, and E.

A is the first job to be processed and will be completed at time 17. Its due date is 45. So job A is not late (tardy); tardiness is zero.

Job B starts after job A, and is completed at time 29 (17+12). This is also not tardy.

In this way the completion time and tardiness of all jobs are completed.

First Come First Served				
Job	Time	Due Date	Completion Time	Tardiness
A	17	45	17	0
B	12	35	29	0
C	22	27	51	24
D	18	54	69	15
E	26	47	95	48

Tardiness = Completion Time – Due Date

Make it zero if you get a negative value.

FCFS: Calculation of Objective Functions

Average Completion Time: Add completion times of all jobs and divide by the number of jobs (261/5). It is 52.5.

Average Number of Jobs in the System: This is obtained by dividing the total of completion times of all jobs by the completion time of the last job (261/95). It is 2.75.

Average Tardiness : This is obtained by adding the tardiness of all jobs and dividing it by the number of jobs (87/5). It is 17.4.

Maximum Tardiness: This is the maximum of all tardiness values, It is 48.

Number of Tardy Jobs: Count the number of jobs that are tardy. Three jobs (C, D, and E) are tardy for this problem. It is 3.

First Come First Served				
Job	Time	Due Date	Completion Time	Tardiness
A	17	45	17	0
B	12	35	29	0
C	22	27	51	24
D	18	54	69	15
E	26	47	95	48
		Total	261	87
Average Completion Time				52.2
Average Number of Jobs in System				2.75
Average Tardiness				17.4
Maximum Tardiness				48
Number of Tardy Jobs				3

Shortest Processing Time (SPT)

The jobs are processed in the increasing order of their processing times.

The job with the minimum processing time (B) is processed first. B is followed by A, D, C, and E.

The calculations of the objective functions follow the same procedure as for the FCFS rule.

Shortest Processing Time				
Job	Time	Due Date	Completion Time	Tardiness
B	12	35	12	0
A	17	45	29	0
D	18	54	47	0
C	22	27	69	42
E	26	47	95	48
		Total	252	90
Average Completion Time				50.4
Average Number of Jobs in System				2.65
Average Tardiness				18
Maximum Tardiness				48
Number of Tardy Jobs				2

Earliest Due Date (EDD)

The jobs are processed in the increasing order of their due dates.

The job with the minimum due date (C) is processed first; and is followed by B, A, E, and D.

The calculations of the objective functions follow the same procedure as for the FCFS rule

Earliest Due Date				
Job	Time	Due Date	Completion Time	Tardiness
C	22	27	22	0
B	12	35	34	0
A	17	45	51	6
E	26	47	77	30
D	18	54	95	41
		Total	279	77
Average Completion Time				55.8
Average Number of Jobs in System				2.94
Average Tardiness				15.4
Maximum Tardiness				41
Number of Tardy Jobs				3

Dynamic Scheduling Problems

A scheduling problem is classified as a dynamic problem if the number of jobs is not fixed.

The examples include new production orders, customers in a bank, shoppers in a store, and cars at a gas station etc.

The new jobs (production orders, customers, cars etc.) keep on coming into the system; and the schedule needs to integrate the new arrivals when an updated schedule is prepared.

A new schedule is prepared every time a job is completed.

Example – Dynamic Scheduling Problem

Consider a single machine problem for which the data are given on RHS.

There are five jobs A, B, C, D, and E that are waiting to be processed.

Suppose Shortest Processing Time (SPT) rule is being used.

SPT sequence is given on RHS.

B is the first job to be processed.

When B is being processed, suppose two new jobs F and G arrive.

F arrives on the 5th day and G arrives on the 10th day. Also assume that the processing time of job F is 8 days and that of G is 20 days.

Data for Dynamic Problem

Job	Time (Days)
A	17
B	12
C	22
D	18
E	26

SPT Sequence

Job	Time (Days)
B	12
A	17
D	18
C	22
E	26

Example – Dynamic Scheduling Problem
(continued)

After job B has been processed, there are six jobs (A, C, D, E, F, and G) that are waiting to be processed.

Since the scheduling rule is SPT, the job with the minimum processing time from among the jobs that are waiting to be processed will be scheduled next. Table 3 gives the schedule at this time.

The next job is F. This will be completed at time 20 (12+8) where 12 is the completion time of job B. If more jobs arrive in the system while F is being processed, they will be integrated with the current jobs and a new schedule will be developed.

These are called dynamic problems since the schedule is continuously updated. Dynamic situations are faced in multiple machines problems also.

SPT Sequence - New	
Job	Time (Days)
F	8
A	17
D	18
G	20
C	22
E	26

Objective Functions – Dynamic Scheduling Problems

Objective functions for dynamic problems are defined in the same way as for single machine static problems.

- ❖ Minimize average completion (flow) time.
- ❖ Minimize average number of jobs in the system.
- ❖ Minimize number of late (tardy) jobs.
- ❖ Minimize average lateness (tardiness).

The values of completion time and tardiness of each job are recorded; and the values of the objective functions can be calculated at any time based on the number of jobs completed at that time.

Thank you.