Scheduling and Controlling Production Activities
by

Prof. G S Prakash

Dept of IEM - MSRIT, Bangalore-54

Scheduling and Controlling Production Activities

Introduction:

"Things must happen on time as scheduled". In the previous chapter we saw that the MRP system specifies what products are needed, in what quantities, and when they are required.

PAC then takes over to deliver the goods. They direct how, when and where the products should be made. Thus they bridge the gap between the material and capacity plans and the finished products.

Production Activity Controls (PAC) are the priority and capacity management techniques used to schedule and control production operations. Figure 1 depicts theses two major concerns of PAC systems.

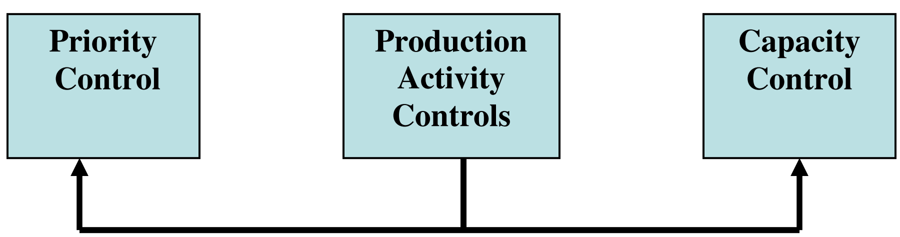


Figure – 1 Major Concerns of PAC

Priority control:

It ensures that production activities follow the priority plan (the material requirements plan) by controlling the orders to vendors and in-house production shops.

Capacity Control:

Capacity control helps by monitoring work centers to ensure that they are providing the amount of labor and equipment time that is necessary to do the scheduled work.

- Although the PAC concepts applies most directly to manufacturing facilities, many of the concepts apply equally well to service system

- However, the concepts of priority control (such as of medical care in hospitals), and capacity control (such as in the utilization of beds and operating rooms) still apply.

PAC Objectives and Data Requirements

System Objectives

- Know the current job status (what jobs are running and where they are located).
- Guide future job operations (determine what jobs should be next and in what work centers).
- Ensure the adequacy of material and capacities (make sure that correct quantities of materials are in the right place at the right time and that capacity and tooling are available to work on them).
- Maximize operational efficiency (maximize labor and machine utilization and minimize inventory, set-up, and other costs while meeting master schedule service objectives).
- Maintain operational control (monitor job status and lead times, measure progress, and signal corrective action when necessary).

System Characteristics

- Production activity control responsibility normally rests with the director of manufacturing or the production control manger.
- It is usually a staff function. A centralized production control department typically coordinates activities, sets schedules, and reviews results.
- Experience suggests some qualities which are desirable to make any PAC system run effectively

System Characteristics (Run Characteristic)

Realistic:

- Schedules should reflect what can realistically be accomplished. They can be "tough" goals that promote efficient use of firm's resources.
- But they should also be feasible, in terms of time standards, errors in lead times and stock levels.
- Otherwise, supervisors and workers will lose confidence in the schedules and circumvent the system.

Understandable:

- The system should become a reliable communication medium and support (not threaten) operating personnel.
- Simplicity of operation will enhance understanding and acceptance by both management and shop personnel.

Necessary:

- Accuracy, timeliness, and flexibility are key characteristics for generating confidence in the system.

- Systems that are accurate and timely become more indispensable over time.
- Flexibility will permit re planning and last-minute schedule modification, so planners can use actual order information rather than forecast data.

Priority and Capacity Control Activities

Priority and capacity controls receive their inputs from the MRP and CRP systems. They attempt to ensure that the production quantities specified in the material requirements plan are completed as scheduled within the hourly time estimates established by the capacity requirements plan.

Figure 2 depicts the major priority and capacity control functions. Priority control activities include order release, dispatching, and status control. Priority control system is concerned with the proper sequencing of jobs in work centers. It should be flexible enough to allow for the updating of priorities whenever quantities or due dates change.

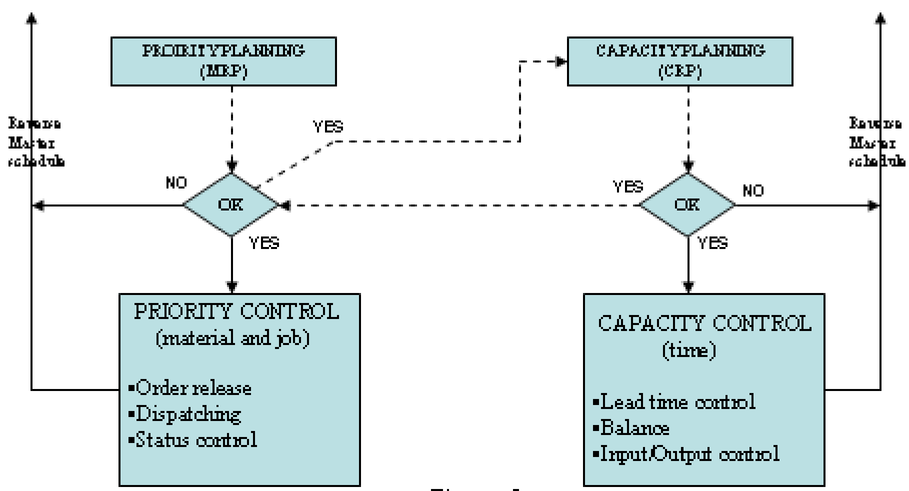


Figure - 2

For capacity control, the major concerns are lead-time control, balance of workload, and input/output control. The load on the shop (in unit hours) determines how long it takes to produce an item (that is, the lead time).

DATA REQUIREMENTS

A well-organized database is essential to a workable PAC system. Figure 3 identifies some key requirements in terms of planning and control functions

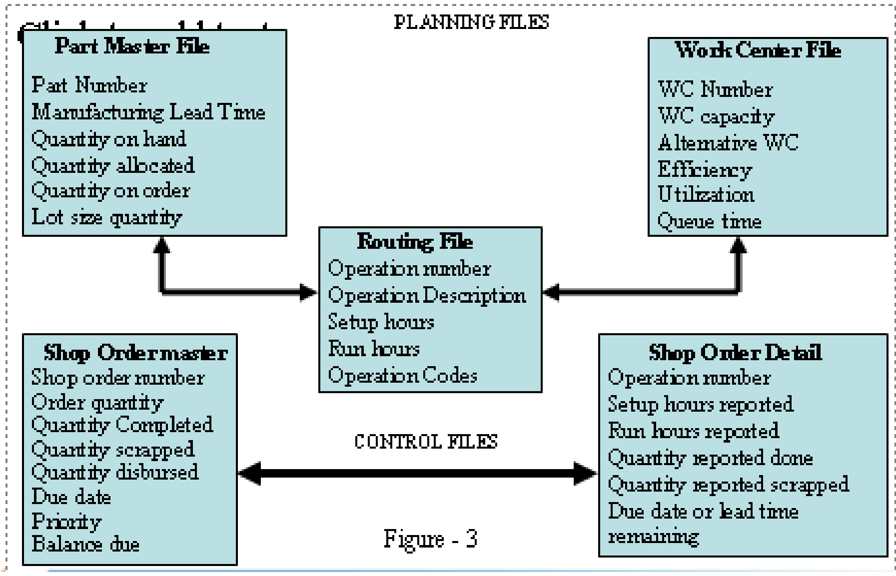


Figure - 3

The ***part master file*** contains all relevant component-part data, such as the description, standard cost and inventory status. This file contains one record for each part number. Some companies have over 100,000 part numbers, so considerable computer storage space may be necessary

The ***work center file***, load, and utilization history. For example, capacity data would include the number of shifts worked per week and the number of machine and labor hours per shift. The efficiency is expressed as the ratio of standard performance to actual performance (in number of hours). Utilization is the ratio of hours worked to hours available for work. The work center file contains one complete record for each work center.

The ***routing file*** stores the data related to operations required to fabricate or assemble an item. It contains a set of records for each manufactured part, with one record for each operation. The operation codes specify the number of machines an operator may run simultaneously, whether an alternative operation is being used, if an outside suppler does an operations, etc.

The ***shop order master file*** contains summary data on each active shop order. Along with the shop order detail file, it is especially useful for production activity controls. In addition to cost information and quantity data, this file contains the priority value assigned to the order. There is one record in this file for each active shop order.

The ***shop order details file*** stores all the scheduling, progress, and priority data relevant to each shop order. The due date or lead-time-remaining information is especially useful in recalculating priorities to keep them valid.A satisfactory functioning PAC system is, of course, heavily dependent upon priority and capacity inputs, such as from an MRP system. The MRP systems sets order release dates and lot sizes for products (and components) needed to satisfy the master schedule. As planning and control files are updated with shop floor information, the MRP system should revise due dates (or signal revisions to the master schedule) as required.

Scheduling strategy and Guidelines

As the production quantities embodied in the aggregate plan and master production schedule are translated into specific job assignments. Short – term or detailed scheduling comes into play. At this point individual jobs are assigned to specific work centers.

System Differences

All firms do not need the same level of scheduling, and indeed many firms do not use formal scheduling method at all Scheduling use is largely a function of the size and type of production system, as summarized below. Operational concerns differ for Continuous, Intermittent and Job shop, and Project type systems.

How Production System Concerns Affect Scheduling Strategy

HIGH VOLUME	INTERMEDIATE VOLUME		LOW VOLUME
Continuous (flow operations) system	**Intermittent** (flow and batch operations) system	**Job shop** (batch or single jobs)	**Project** (single jobs)
Key characterizes			
Specialized Equipments Same sequence of operations unless guided by microprocessors or robots	Mixture of equipment Similar sequence for each batch	General purpose equipment Unique sequence for each job	Mixture of equipment Unique sequence and location for each job
Design Concerns			
• Line balancing • Changeover time and cost	• Line and work-machine balance • Changeover time and cost	• Worker-machine balance • Capacity utilization	• Allocating resources to minimize time and cost
Operational Concerns			
Material shortages Equipment break Downs Quality operations Product mix and value	Material and Equipment problems Setup costs and run Lengths Inventory accumulation (Run-out times)	Job sequencing Work center loading Work flow and work in process	Meeting time scheduling Meeting budgeted Costs Resource utilization

Scheduling strategies

Scheduling strategies differ widely among firms and may range from “no scheduling” to very sophisticated approaches. It is convenient to classify the strategies into four groups:

1) detailed,
2) cumulative,
3) cumulative-detailed, and
4) priority decision rules

Scheduling Time Frame

We have seen that the master production schedule as well as the material and capacity requirements plans, are typically expressed in weekly time buckets. As requirements are translated into shop orders, the weekly loadings are preserved, but more precision is sometimes needed for controlling the efficiency of operations However, even though the jobs are given priorities on a daily basis and the precise times to do a job are measured (and recorded) in minutes. It is common practice to accept weekly precision for scheduling purposes in job shops. This gives the shop flexibility to balance high and low-priority work orders.

Forward versus Backward Scheduling

Figure-4 illustrates two approaches to scheduling. Forward scheduling starts as soon as requirements are known and often results in the completion of the component before the required due date. The result is more work-in-process inventory and higher inventory carrying costs. Backward (or “set back”) scheduling uses the same lead-time offset logic as MRP. Components are delivered “when needed” rather than “as soon as possible.”

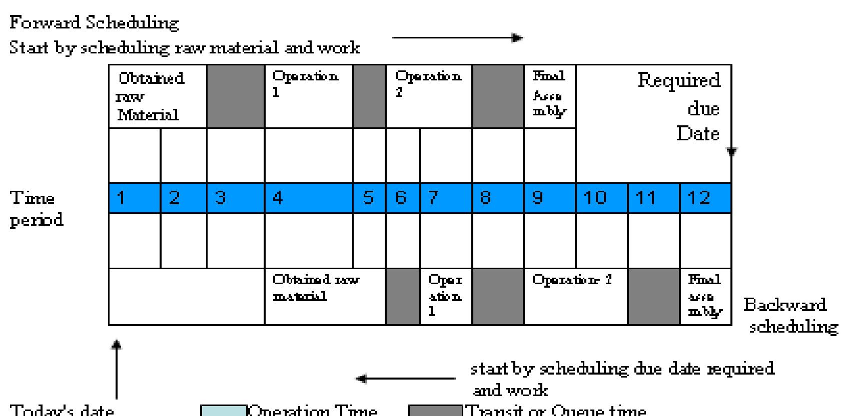


Figure-4

Forward and Backward Scheduling

With finite capacity loading, some work centers may already be too loaded to accept more work in a given time period. For example, in Fig. 4 (backward scheduling) suppose the work center for operation 1 is unavailable during periods 6 and 7. Operation 1 could be moved ahead to period 5, with raw materials acquisition moved up to period 2. Some firms combine both approaches by starting "today' with plans to finish "when due. "Their rationale is to create some slack time in the intermediate weeks

Scheduling and loading guidelines

Successful firms have developed some guidelines for scheduling jobs and loading work centers.

- Provide a realistic schedule.
- Allow adequate time for operations.
- Allow adequate time before, between, and after operations.
- Don't release all available jobs to the shop.

Scheduling and loading guidelines

- Don't schedule all available capacity in the shop.
- Load only selected work centers.
- Allow for necessary changes.
- Gear shop responsibility to the schedule

Scheduling Methodology

The scheduling methodology depends upon the type of industry, organizations, product and level of sophistication required. There are three general classes of methodology:

- Charts and Boards
- Priority decision rules
- Mathematical programming methods

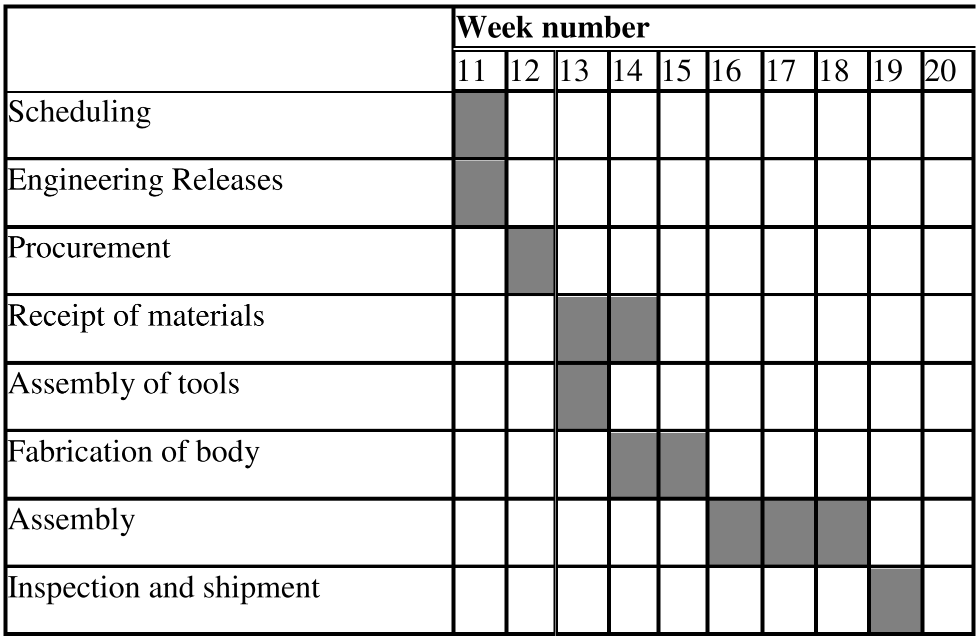
Gant Charts, Schedule Boards, and Computer Graphics

Gantt charts and associated scheduling boards have been extensively used scheduling devices in the past, although many of the charts are now drawn by computer. Gantt charts are extremely easy to understand and can quickly reveal the current or planned situation to all concerned. They are used in several forms, including

1. Scheduling or progress charts, which depict the sequential schedule,
2. Load charts, which show the work assigned to a group of workers or machines, and
3. Record charts, which are used to record the actual operating times and delays of workers or machines.

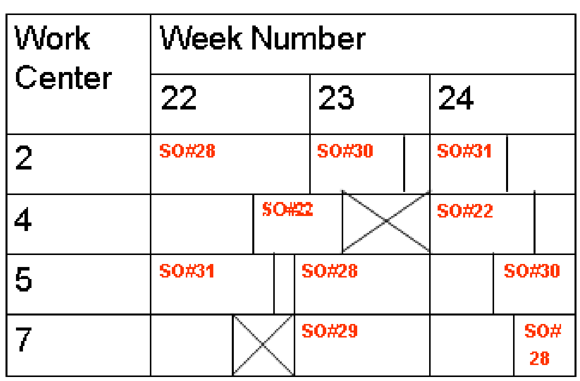
Figure below illustrates a simplified (a) scheduling chart and (b) load chart. The load chart depicts the specific customer orders (SO) to be worked on in work centers 2, 4, 5, and 7. The **X** signifies an unavoidable delay (for maintenance, and so on). Load charts can also be used to show the scheduled workload, maintenance and idle time

Scheduling Chart



b) Load Chart

b) Load Chart

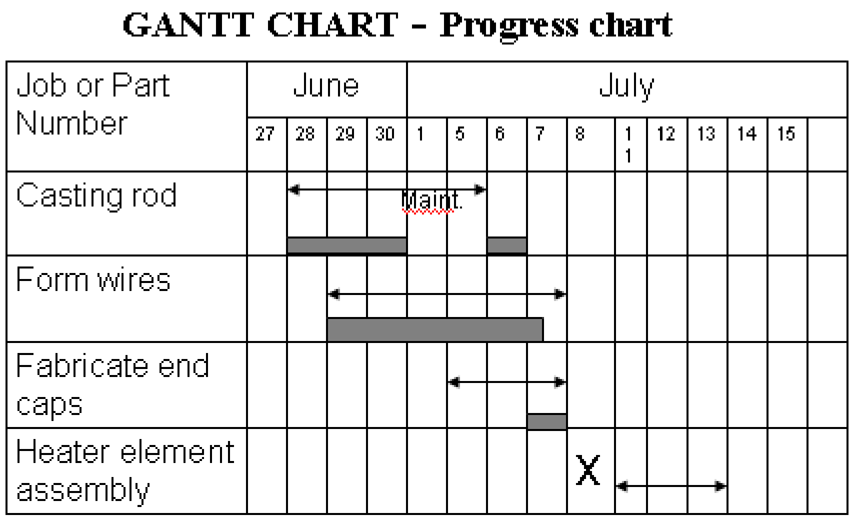


Gant Charts

Another Gantt chart is show in Fig 5, where the Vee indicates updating through july7. This is a progress chart showing which activities must be done prior to other activities.

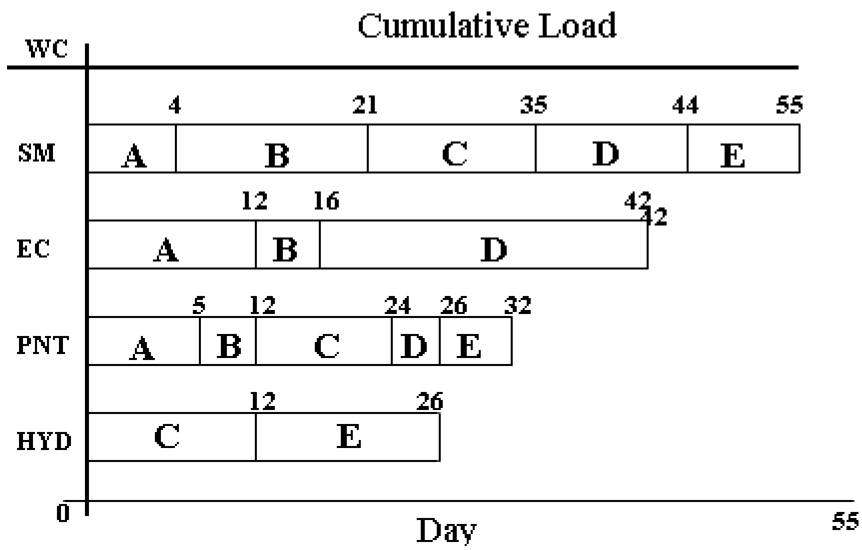
It shows that the rod casting were delayed a day, probably due to extra maintenance on the asting furnace. The wires are half a day behind scheduling, and end-cap fabrication was 2 days late in getting started with a planned 3-day scheduling It appears that the end caps will probably not be completed before the heater element assembly is begun on Monday, July 11, even though an extra day of slack was provided on July 8. Keeping Gantt charts up to date has always been a major problem, especially as the number of jobs and work centers increases. Numerous mechanical and magnetic charts and boards are available to facilitate the revision process

GANTT CHART - Progress chart



Gantt load charts and visual load profiles can be helpful for evaluating the current loadings. The Gantt load chart "A graph showing workloads on a time scale" Gantt charts are extremely easy to understand and can quickly reveal the current or planned situation to all concerned. A Gantt chart shows how each machine or work centre is planned for work on different job orders. The scheduled start and finish times, the delay or production bottlenecks that have occurred and the time lost for that reason. This information could be very useful for diagnosing the problems if any and taking appropriate rescheduling action whenever needed.

Example: The graphical procedure is shown in fig below. An aircraft repair facility has four work centers through which five jobs must be processed. Aircraft A,B,C,D and E require sheet metal and paint work. A,B and D require electronics work, and C and E require hydraulics work. The chart shows the total estimated work load that the jobs require at all work centers.



Thus 55 days of cumulative work lie ahead for the sheet metal centers, the paint centers faces a 32-day load and so on. The chart does not specify which job will be completed at which time not, it shows the sequence in which the jobs should be processed.

PRIORITY SEQUENCING

A systematic procedure for assigning priorities to waiting jobs, thereby determining the sequence in which jobs will be processed.

Some of the major criteria used for priority sequencing are

1. Setup costs
2. In process inventory costs
3. Idle time.
4. Number or percent of jobs that are late.
5. Average time jobs are late.
6. Standard deviation of time jobs are late.
7. Average number of jobs waiting in the queue.
8. Average time to complete a job.
9. Standard deviation of time to complete a job.

Some priority Sequencing Rules:-

The following rules are representative of the many used today in manufacturing and service industries.

First come first served (FCFS): Priority rule that gives top priority to the waiting job that arrived earliest in the production system.

Earliest due date (EDD): Priority rule that gives top priority to the waiting job whose due date is earliest.

Shortest processing time (SPT): Priority rule that gives top priority to the waiting job whose operation time at a work centers is shortest.

Least Slack (LS): A priority rule that gives top priority to the waiting job whose slack time is least.

Slack Time:- It is the difference between the length of time remaining until the job is due and the length of its operation time.

Example: Let's examine some of these rules and illustrate how they work. Consider the Aircraft Repair facility processing jobs and evaluate the sequences created according to the FCFS and SPT rules.

Waiting job (in FCFS sequence)	Processing Time (in days)	Due Date (in days from now)
A	4	6
B	17	20
C	14	18
D	9	12
E	11	12
	Total 55 days	

FCFS: We assume jobs arrived in alphabetical orders, so that according to FCFS, job A goes first, job B next and so on. Customers requested their order be completed by the due dates listed in table below.

Waiting job (in FCFS sequence)	Processing Time (in days)	Flow Time (in days)	Due Date (in days from now)
A	4	4	6
B	17	21	20
C	14	35	18
D	9	44	12
E	11	55	12
	Total 55 days		

FCFS Sequence Results are as follows.

1. Flow Time:- The total time that a job is in the system the sum of waiting time and processing time. Job B for example waits 4 days, while A is being processed and then takes 17 days operations time. Job B is therefore completed in 21 days.

2. Total Completion Time. After 55 days, all jobs are completed.

3. Average Flow Time. It is calculated by summing the flow time for all jobs and dividing by the numbers of jobs.
   ie (4+21+35+44+55)/5 = 31.8

4. Average numbers of jobs in the system each day.

The average no of jobs flowing in the system (waiting or processing) from the beginning of the sequence through the time when the last job is finished is calculated as follows. For the first job 4 days, 5 jobs are in the system. For the next 17 days, 4 jobs are in the system. For days 22 to 35, 3 jobs are in the system and so forth. There are 55 days for the sequence, Hence.

[5 (4) + 4 (17) + 3(14) + 2(9) + 1 (11)] / 55 = 2.89 jobs in the system /day

5. Average job latencies:-

The lateness of each job is the difference of its flow time and its due date. For job A, Flow time is 4 days. Since the Due date is 6 days, the difference is 4 - 6 = - 2. Job a was finished 2 days earlier than required no lateness like wise. Average lateness is calculated as follows.
(0+ 1+17+32+43) / 5 Avg. lateness = 18.6 days,

SPT Sequence Results are as follows.

The SPT rule assigns highest priority to the job order whose processing time is shortest. The SPT rule yields the data in the Table below the sequence is A, D, E, C, B

Waiting job (in SPT rule)	Processing Time (in days)	Flow Time (in days)	Due Date (in days from now)
A	4	4	6
D	9	13	12
E	11	24	12
C	14	38	18
B	17	55	20
	Total 55 days		

1. Total Completion Time.

After 55 days all jobs are completed.

2. Average Flow Time
   (4+13+24+38+55) / 5 =134/5 =26.8 days.

3. Average no of jobs in the system each day
   (5 94) +4(9) + 3(11) +2(14) + 1 (17)) /55 = 2.44 jobs.

4. Average job lateness
   (0+1+12+20+35) % 5 Avg. lateness = 13.6 days.

Characteristics of the sequencing rules.
Comparison of SPT and FCFS rules.

Criterion				
Rule	Total Completion Time in days	Average Flow Time (in days)	Average jobs in system Each day	Average lateness (in days)
FCFS	55	31.8	2.89	18.6
SPT	55	26.8	2.44	13.6

From the table we can see that SPT is superior. It gives lower average flow time, so less inventories are tied up and quicker service can be provided to customers. Average no job in the system is reduced Finally, since average lateness is reduced, deliveries to customers are more prompt.

Example – 2: Shown here are the time remaining (number of days until due) and work remaining (number of days) for five jobs. Which are assigned a letter as they arrived. Sequence the jobs by priority rules

a) FCFS b) EDD c) LS d) SPT e) LPT

Job	Number of days until due	Number of days work remaining
A	**8**	**7**
B	**3**	**4**
C	**7**	**5**
D	**9**	**2**
E	**6**	**6**

Solution – The numerical included in parenthesis are for reference

SEQ.	FCFS	EDD	LS	SPT	LPT
1st	**A**	**B (3)**	**B(-1)**	**D(2**	**A(7)**
2nd	**B**	**E (6)**	**E(0)**	**B(4)**	**E(6)**
3rd	**C**	**C (7)**	**A(1)**	**C(5)**	**C(5)**
4th	**D**	**A (8)**	**C(2)**	**E(6)**	**B(4)**
5th	**E**	**D (9)**	**D(7)**	**A(7)**	**D(2)**

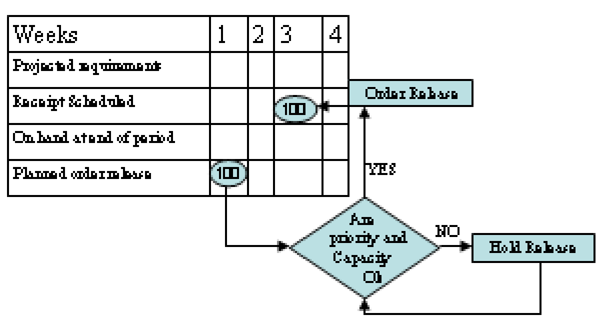
PRIORTY CONTROL

A good Priority control system should reflect true needs, rank the importance of those needs, and be capable of updating those ranking as quantities or due dates change. We turn now to the three major priority control activities necessary for a good priority control or "dispatching" system: Order release, Dispatching and Status control

ORDER RELEASE

Order release is the crucial step that convert an order from a planned status to reality in the shop or on order with a vendor. When an open order is created, actual material and labour may be charged against the job. Figure below illustrates the order release function. Recall the planned order release and scheduling receipt entries on the MRP forms.

As the planned order release date moves in to the current period, the release actually converts what was a "planned" receipt into a "scheduled" receipt . It makes the order official. Order release should not be automatic whenever an order arrives; it should be a conscious human (not Computer) activity because it requires a judgmental assessment of several factors. Two key considerations identified in Fig below are the *validity of the priority and the availability of the capacity.* Is the due date an accurate refection of need? If so, will the necessary material and components be available to support work on this shop order? Are routing and tooling data provided? With respect to capacity, orders should be released to the shop at a planned (and steady) workload if possible.



DISPATCHING

The dispatch list is probably the most widely used tool for priority control. It lists all jobs available to a work center and ranks them by a relative priority (or due date). When priorities have been assigned to specific jobs. The scheduling methodology (studied earlier in the chapter) becomes a reality. Scheduling is implemented via the dispatch list.

CAPACITY CONTROL

We take up capacity control in this section to emphasize the importance of separating priority and capacity problems. Many firms fail to correctly identify capacity problems. As a result, their Fix is no solution at all. For example, assume that a shop is so loaded that some customers are getting upset with delays. Calling their orders **hot** or putting a red tag on them may placate some customer's, but it won't do much to improve relations with others whose orders are dispatched.

Priority techniques do not solve capacity problems.

Capacity control ensures that the hours of labor and machine capacity actually delivered are in conformance with the capacity plans. This requires measurement of actual capacity output, feedback to a database, comparison with planned levels, and provision for the corrective action.

JOB SEQUENCING

Introduction:

In sequencing, we determine an appropriate order (sequence) for a series of jobs to be done on a finite number of service facilities, in some pre – assigned order, so as to optimize the total involved cost (time). A practical situation may correspond to an industry producing a number of products, each of which is to be processed through different machines, of course finite in number.

Definition:

Suppose there are 'n' jobs (1,2,3…4) each of which has to be processed one at a time at each of 'm' machines A,B,C…… The order of processing each job through machines is given (for example, A, C, B). The time that each job must require on each machine is known, The problem is to find a sequence among (n!)m number of all possible sequences (or combination) (or order) for processing the jobs so that the total elapsed time for all the jobs will be minimum.

Mathematically let

Ai = Time for job í on machine A

Bi = Time for job í on machine B etc.

Ti = Time for start of first job to completion of the last job.

Then, the problem is to determine for each machine a sequence of jobs (i1, i2, i3 ….. in), where (i1, i2, i3 .. in is the permutation of the integers) which will minimize Total elapsed time T.

Terminology, Notation and assumptions:

The following terminology & notations will be used.

1. Aij = Processing time required by the ith job on the jth Machine (i = 1, 2,…n; j= 1,2……m)
2. T = Total elapsed time (including idle time if any) for processing all the jobs.
3. Xij = idle time on machine j from end of (i -1)th job to start of the ith job. (i = 2,3, 4…….n) (j = 1,2………m)

Terminology

- Number of machines
- Processing order
- Processing time
- Idle time on a machine
- Total elapsed time
- No passing rule

Principal assumptions:

1. No M/c can process more than one operation at a time.
2. Each operation once started, must be per formal till completion.
3. A job is an entity ie even through the job represents a lot of individual no lot may be processed by more than one machine at a time.
4. All machines are of different types
5. Time intervals for processing are independent of the order in which operation are performed.
6. All jobs are completely known and are ready for processing before the period under consideration begins.

Processing 'n' jobs through two machines

The problem can be described as (i) only two machines A and B are involved (ii) each job is processed in the order AB and (iii) the exact or expected processing times A1, A2, A3, - - - An, & B1, B2, B3, - - - Bn, are known as shown in table below.

Processing Times	**Job (i)** 1 2 3…….n

Ai	A1 A2 A3…..An
Bi	B1 B2 B3…..Bn

The problem is to sequence (order) the jobs so as to minimize the total elapsed time T. The solution procedure adopted by Johnson is given below.

Solution procedure:- JOHNSON PROCEDURE

Step 1:- Examine the processing time occurring in the list A1, A2, A3, - - - An, and B1, B2, B3, - - - Bn. select the min/least processing time; if there is a tie either of the smallest processing times should be selected. ie min(i) [Ai Bi]

Step 2:- If the min processing time is Ar, select rth job first. If it is Bs, do the sth job last (as the given order is AB)

Step3:- There are now (n-1) jobs left to be ordered. Again repeat steps 1 and 2 for the reduced set of processing times obtained by deleting the processing times for both the machines corresponding to the jobs already assigned.

Continue till all jobs have been ordered. The resulting ordering will minimize the elapsed time T.

Example 1.

There are five jobs, each of which must go through the two m/c's A and B in the order AB processing times are given below

Processing Times					
Job	1	2	3	4	5
M/c A	5	1	9	3	10
M/c B	2	6	7	8	4

Determine a sequence for five jobs that will minimize the elapsed time T

Job	A	B
1	5	2
2	1	6
3	9	7
4	3	8
5	10	4

Apply steps 1 and 2 of solution procedure it is seen that the smallest processing time is one hour for job 2 on the machine A so list the job 2 at first place as shown below

2			

Now the reduced list of processing times becomes

Job	A	B
1	5	2
3	9	7
4	3	8
5	10	4

Again the smallest processing time in the reduced list is 2 hours for job 1 on the machine B. So place job 1 last.

2				1

Continue in the like manner, the next reduced list is obtained

Job	A	B
3	9	7
4	3	8
5	10	4

2	4			1

And the reduced list

Job	**A**	**B**
3	**9**	**7**
5	**10**	**4**

Leading to sequence

2	**4**	**3**	**5**	**1**

Finally the optimal sequence is obtained

2	**4**	**3**	**5**	**1**

Further, it is also possible to calculate the minimum elapsed time corresponding to the optimal sequencing, using the individual processing time given in the statement of the problem. The details are given in table below

Job sequence	**Machine A**		**Machine B**	
	Time in	**Time out**	**Time in**	**Time out**
2	0	1	1	7
4	1	4	7	15
3	4	13	15	22
5	13	23	23	27
1	23	28	28	**30**

Thus, the minimum time ie the time for starting of job 2 to completion of the last job 1 is 30 hours only. During this time the machine a remains idle for 2 hours (from 28 to 30 hours) and the machine B remains idle for 3 hours only.

Example 2:

Following table shows the machine time (in Hours) for 5 Jobs to be processed on two different machines.

Job	:	1	2	3	4	5
Machine A	:	3	7	4	5	7
Machine A	:	6	2	7	3	4

Passing is not allowed. Find the optimal sequence in which jobs should be processed.

Job	Machine A	Machine B
1	**3**	**6**
2	**7**	**2**
3	**4**	**7**
4	**5**	**3**
5	**7**	**4**

The optimal sequence is:

1	3	5	4	2

Job Sequence	Machine A		Machine B		Idle Time
	Time In	Time out	Time In	Time Out	
1	0	3	3	9	3
3	3	7	9	16	0
5	7	14	16	20	0
4	14	19	20	23	0
2	19	26	26	28	3
					6

Total elapsed Time (T) = 2hrs
Idle time for Machine A = 2hrs
Idle time for machine B = 6hrs

Example -3

Find the Sequence that minimizes the total elapsed time required to complete the following Talks

Tasks	:	A	B	C	D	E	F	G	H	I
Time on I m/c	:	2	5	4	9	6	8	7	5	4
Time on I m/c	:	6	8	7	4	3	9	3	8	11

The Optimal Sequence:

A	C	I	B	H	F	D	E	G
A	I	C	B	H	F	D	G	E
A	I	C	B	H	F	D	G	E
A	I	C	H	B	F	D	E	G
A	C	I	H	B	F	D	E	G

Elapsed time:

Job Sequence	**Machine I**		**Machine II**	
	Time In	**Time out**	**Time In**	**Time Out**
A	**0**	**2**	**2**	**8**
C	**2**	**6**	**8**	**15**
I	**6**	**10**	**15**	**26**
B	**10**	**15**	**26**	**34**
H	**15**	**20**	**34**	**42**
F	**20**	**28**	**42**	**51**
D	**28**	**37**	**51**	**55**
G	**37**	**43**	**55**	**58**
E	**43**	**50**	**58**	**61**

Example: 4 the following table gives machine times for the six jobs and the two machines.

Job	:	1	2	3	4	5	6
Machine I	:	5	9	4	7	8	6
Machine II	:	7	4	8	3	9	5

Find the sequence of jobs that minimizes the total elapsed time to complete the jobs. Find the min time by using Gantt's chart.

The Optimal Sequence:

3	**1**	**5**	**6**	**2**	**4**

Sequence	**Machine I**	**Machine II**	**Idle Time**

	Time In	Time out	Time In	Time Out	
3	**0**	**4**	**4**	**11**	**4**
1	**4**	**9**	**11**	**18**	**0**
5	**9**	**17**	**18**	**27**	**0**
6	**17**	**23**	**27**	**32**	**0**
2	**23**	**32**	**32**	**36**	**0**
4	**32**	**39**	**39**	**42**	**3**
					7

Total elapsed Time (T) = 42hrs
Idle time for Machine 1 = 3hrs
Idle time for Machine 2 = 7hrs

Processing n jobs through Three machines

The problem can be described as

(i) only three machines A, B and C are involved
(ii) each job is processed in the order ABC
(iii) transfer of jobs is not allowed
(iv) the exact or expected processing times A1, A2, A3, - - - An, & B1, B2, B3, - - - Bn, are known as shown in table below.

Machines	Processing Times Job (i) 1 2 3…….n
M/C A	A1 A2 A3…..An
M/C B	B1 B2 B3…..Bn
M/C C	C1 C2 C3…..Cn

The problem is to sequence (order) the jobs so as to minimize the total elapsed time T. The solution procedure adopted is given below.

Solution procedure:- Optimal Solution

So far no general procedure is available for obtaining an optimal sequence in this case. However, the earlier method adopted by Johnson can be extended to cover the special cases where either one or both of the following conditions is satisfied

Conditions:

i) The minimum time on machine A ≥ maximum time on machine B
ii) The minimum time on machine C ≥ maximum time on machine B

The procedure explained here is to replace the problem with an equivalent problem involving n jobs and two imaginary machines denoted by G and H and corresponding time of Gi and Hi are denoted by

$G_i = A_i + B_i$ and $H_i = B_i + C_i$

If this problem with given ordering GH is solved, the resulting optimal sequence will also be optimal for the original problem

Example 1. There are five jobs, each of which must go through the m/c's A, B and C in the order ABC. Processing times are given table below

Machines	Processing Times Job (i)				
	1	2	3	4	5
M/C A	8	10	6	7	11
M/C B	5	6	2	3	4
M/C C	4	9	8	6	5

Solution:

Here, Min Ai = 6 and Max. Bi = 6 Min. Ci = 4
Since, one of two conditions is satisfied by Min. Ai ≥ Max. Bi so earlier procedure can be applied. The equivalent problem, involving five jobs and two imaginary machines G and H becomes

Job	**Processing Times**	
	Gi = (Ai+Bi)	**Hi = (Bi+Ci)**
1	**13**	**9**
2	**16**	**15**
3	**8**	**10**
4	**10**	**9**
5	**15**	**9**

The Optimal Sequence is: (Alternate solution exists)

3	**2**	**1**	**4**	**5**

3	2	4	5	1
3	2	1	5	4
3	2	4	1	5
3	2	5	4	1
3	2	5	1	4

Job seq.	Machine A		Machine B		Machine C	
	Time in	Time out	Time in	Time out	Time in	Time out
3	0	6	6	8	8	16
2	6	1	16	22	22	31
1	16	24	24	29	31	35
4	24	31	31	34	35	41
5	31	42	42	46	46	51

Total elapsed Time (T) = 51 hrs
Idle time for Machine A = 9 hrs
Idle time for machine B = 31 hrs
Idle time for machine C = 19 hrs

Example 2. Find the sequence that minimizes the total elapsed time required to complete the following tasks. Each job is processed in the order ACB.

Machines	Processing Times Job (i) 1	2	3	4	5	6	7
M/C A	12	6	5	11	5	7	6
M/C B	7	8	9	4	7	8	3
M/C C	3	4	1	5	2	3	4

Solution:

Machines	Processing Times Job (i) 1	2	3	4	5	6	7
M/C A	12	6	5	11	5	7	6

M/C C	**3 4 1 5 2 3 4**
M/C B	**7 8 9 4 7 8 3**

Solution: Here, Min Ai = 5 and Max. Ci = 5 Min. Bi = 3

Since, one of two conditions is satisfied by Min. Ai ≥ Max. Ci so earlier procedure can be applied. The equivalent problem, involving five jobs and two imaginary machines G and H becomes

Job	**Processing Times**	
	Gi = (Ai+Ci)	**Hi = (Ci+Bi)**
1	**15**	**10**
2	**10**	**12**
3	**6**	**10**
4	**16**	**9**
5	**7**	**9**
6	**10**	**11**
7	**10**	**7**

The Optimal Sequence is: (Alternate solution exists)

3	5	2	6	1	4	7
3	5	6	2	1	4	7

Job seq.	**Machine A**		**Machine C**		**Machine B**	
	Time in	**Time out**	**Time in**	**Time out**	**Time in**	**Time out**
3	0	5	5	6	6	15
5	5	10	10	12	15	22
2	10	16	16	20	22	30
6	16	23	23	26	30	38
1	23	35	35	38	38	45
4	35	46	46	51	51	55
7	46	52	52	56	56	59

Total elapsed Time (T) = 59 hrs Idle time for Machine A = 7 hrs
Idle time for machine C = 37 hrs Idle time for machine B = 13 hrs

Example 3.

A shop has eight shop orders that must be processed sequentially through three work centers. Each job must be finished in the order wc1,wc2,wc3. Processing times in hors is shown in table below. Develop the job sequence that will minimize the completion time.

Machines	Processing Times Job (i)							
	A	B	C	D	E	F	G	H
WC 1	4	8	5	9	3	4	9	6
WC 2	6	4	7	1	4	2	5	2
WC 3	8	7	9	7	9	8	9	7

Solution: Here, Min WC1 = 3 and Max.WC2 = 7 Min.WC3 = 7 since, one of two conditions is satisfied by Min. WC3 ≥ Max. WC2 so earlier procedure can be applied. The equivalent problem, involving five jobs and two imaginary machines G and H becomes

Job	Processing Times	
	Gi = (wc1+wc2)	Hi = (wc2+wc3)
A	10	14
B	12	11
C	12	16
D	10	8
E	7	13
F	6	10
G	14	14
H	8	9

The Optimal Sequence is: (Alternate solution exists)

F	E	H	A	C	G	B	D

Job seq.	WC 1		WC 2		WC 3	
	Time in	Time out	Time in	Time out	Time in	Time out
F	0	4	4	6	6	14
E	4	7	7	11	14	23
H	7	13	13	15	23	30
A	13	17	17	23	30	38
C	17	22	23	30	38	47
G	22	31	31	36	47	56
B	31	39	39	43	56	63
D	39	48	48	49	63	70

Total elapsed Time (T) = 70 hrs   Idle time for WC 1 = 22 hrs
Idle time for WC 2 = 39 hrs   Idle time for WC 3 = 6 hrs

Flow shop Scheduling

In flow shop scheduling problem, there are n jobs; each requires processing on m different machines. The order in which the machines are required to process a job is called process sequence of the job. The process sequences of all the jobs are the same. But the processing times for various job on a machine may differ. If an operation is absent in a job, then the processing time of the operation of that job is assumed as zero.

The flow-shop scheduling problem can be characterized as given below.

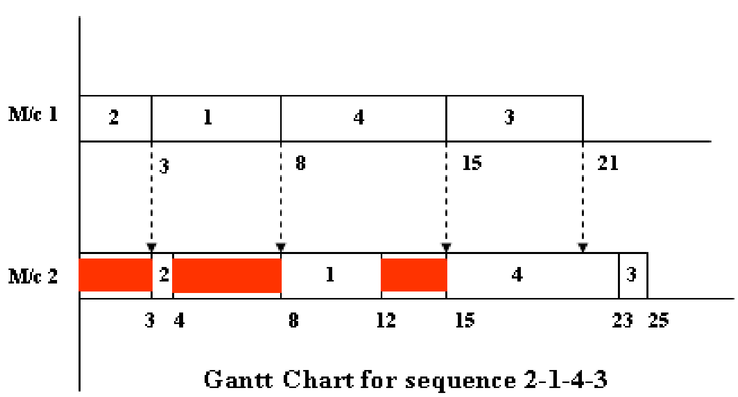
1. A set of multiple-operation jobs is available for processing at time zero (Each job require m operations and each operation requires a different machine).
2. Set-up times for the operations are sequence independent, and are included in processing times.
3. Job descriptors are known in advance.
4. *m* different machines are continuously available.
5. Each individual operation of jobs is processed till its completion without break.

The main difference of the flow shop scheduling from the basic single machine scheduling is that the inserted idle time may be advantageous in flow shop scheduling. Though the current machine is free, if the job from the previous machine is not released to the current machine we cannot start processing on that job. So, the current machine has to be idle for some time. Hence, inserted idle time on some machine would lead to optimality.

For example consider the following flow shop problem

Job	Machine Number	Machine Number
1	5	4
2	3	1
3	6	2
4	7	8

If the sequence of the job is 2-1-4-3, then the corresponding makespan (total elapsed time) is computed as shown in figure below. In Figure, the makespan is 25. Also, note the inserted idle times on machine 2 are from 0 to 3, 4 to 8 and 12 to 15.



Gantt Chart for sequence 2-1-4-3

Consider another sequence say 3-4-1-2. The makespan for the schedule is 26. This problem has 4 jobs. Hence, 4! Sequence are possible. Since, n! grows exponentially with n, one needs some efficient procedure to solve the problem. For large size of n, it would be difficult to solve the problem. Under such situation we can use some efficient heuristic.

Flow shop scheduling includes

- Johnson's rule for 'n' jobs on 2 machines
- 'n' jobs on 3 machines and
- CDS heuristic

CDS HEURISTIC

for large size problems, it would be difficult to get optimum solution in finite time, since the flow shop scheduling is a combinatorial problem. This means the time complexity function of flow shop problem is exponential in nature. Hence, we have to use efficient heuristics for large size problems.

CDS (Campbell, Dudek and Smith) heuristic is one such heuristic used for flow shop scheduling. The CDS heuristic corresponds to multistage use of Johnson's rule applied to a new problem formed from the original with processing times.

At stage 1 $t_{j1}^{1} = t_{j1} \quad and \; t_{j2}^{1} = t_{jm}$

In other words, Johnson's rule is applied to the first and *m*th operations and intermediate operations are ignored.

At stage 2 $t_{j1}^{1} = t_{j1} + t_{j2}, \quad t_{j2}^{2} = t_{jm} + t_{j,m-1}$

That is Johnson's rule is applied to the sum of the first two and the last two operation processing times. In general at stage i,

$$t_{j1}^{i} = \sum_{k=1}^{i} t_{jk} \; and \; t_{j2}^{i} = \sum_{k=1}^{i} t_{j,m-k+1}$$

For each stage I (i= 1, 2, …,m-1), the job order obtained is used to calculate a makespan for the original problem. After m-1, stages, the best makespan among the m-1 schedules is identified. (some of the m-1 sequences may be identical). Find the makespan using the CDS heuristic for the following flow problem.

Job $_j$	t_{j1}	t_{j2}	t_{j3}	t_{j4}
1	4	3	7	8
2	3	7	2	5
3	1	2	4	7
4	3	4	3	2

Solution. Stage 1

Job j	M/c – 1 t_{j1}	M/c – 2 t_{j2}
1	4	8
2	3	5
3	1	7
4	3	2

The optimal sequence for the above problem is show below. 3 - 2 - 1 - 4

The makespan calculation for the above schedule is show below.

Job	M/C – 1		M/C – 2		M/C – 3		M/C – 4	
	In	Out	In	Out	In	Out	In	Out
3	0	1	1	3	3	7	7	14
2	1	4	4	11	11	13	14	19
1	4	8	11	14	14	21	21	29
4	8	11	14	18	21	24	29	31

Stage 2

Job j	M/c – 1 t_{j1}	M/c – 2 t_{j2}
1	7	15
2	10	7
3	3	11
4	7	5

After Applying Johnson's algorithm to the above problem. We get the sequence, 3 – 1 – 2 – 4. the makespan calculation is summarized in the following table.

Processing Time (IN Hour)								
Job	**M/C – 1**		**M/C – 2**		**M/C – 3**		**M/C – 4**	
	In	**Out**	**In**	**Out**	**In**	**Out**	**In**	**Out**
3	**0**	**1**	**1**	**3**	**3**	**7**	**7**	**14**
1	**1**	**5**	**5**	**8**	**8**	**15**	**15**	**23**
2	**5**	**8**	**8**	**15**	**15**	**17**	**23**	**28**
4	**8**	**11**	**15**	**19**	**19**	**22**	**28**	**30**

The makespan for the sequence 3 – 1 – 2 - 4 is 30.

Stage 3

Processing Time		
Job j	**M/c – 1 t_{j1}**	**M/c – 2 t_{j2}**
1	14	18
2	12	14
3	7	13
4	10	9

The application of Johnson's algorithm to the above data yield the sequence 3 – 2 – 1 – 4 . The determination of the corresponding makespan is show below

Processing Time (IN Hour)								
Job	M/C – 1		M/C – 2		M/C – 3		M/C – 4	
	In	Out	In	Out	In	Out	In	Out
3	0	1	1	3	3	7	7	14
1	1	4	4	11	11	13	14	19
2	4	8	11	14	14	21	21	29
4	8	11	14	18	21	24	29	31

The makespan for the sequence 3 – 2 – 1 – 4 is 31.
Summary:

Stage	**Sequence**	**Sequence**
1	3 – 2 – 1 – 4	31
2	3 – 1 – 2– 4	30
3	3 – 2 – 1 – 4	13

The best sequence is 3 – 1 – 2 – 4 which has the makespan of 30.

SINGLE MACHINE SCHEDULING

- Scheduling is the allocation of start and finish time to each particular order.
- Scheduling can bring productivity in shop floor by providing a calendar for processing a set of jobs.
- It is nothing but scheduling various jobs on a set of resources (machine) such that certain performance measures are optimized.
- The single machine scheduling problem consists of n jobs with the same single operation on each of the jobs.
- While the flow shop scheduling problem consists of n jobs with m operations on each of the jobs. In this problem, all the jobs will have the same process sequences.
- The job shop scheduling problem contains n jobs with m operations on each on the jobs; but, in this case, the process sequences of the jobs will be different from each other.

Concept of single machine scheduling

The basic single machine scheduling problem is characterized by the following conditions.

1. A set of independent, single operation jobs is available for processing at time zero.
2. Set-up time of each job is independent of its position in job sequence. So, the set-up time of each job can be included in its processing time.
3. Job descriptors are known in advance.
4. One machine is continuously available and is never kept idle when work is waiting.
5. Each job is processed till its completion without break.

- Under these conditions, one can see one one-to-one correspondence between a sequence of the n jobs and a permutation of the job indices 1, 2, …..n.

- The total number of sequences in the basic single machine problem is n! which is the number of different permutation of n elements.

- [3] = 5 means that the job 5 is assigned to the third position in the sequence. Similarly, d[4] refers to the due date of the job assigned to the fourth position in the sequence.

- The following three basic data are necessary to describe jobs in a deterministic single machine scheduling problem which are as follows.

- **Processing Time (t_j):** It is the time required to process job J. The processing time, tj will normally include both actual processing time and set-up time.
- **Ready Time (r_j):** it is the time at which job j is available for processing. The ready time of a job is the difference between the arrival time of that job and the time at which that job is taken for processing. In the basic model, as per condition 1, $r_j = 0$ for all jobs.
- **Due date (d_j):** It is time at which the job J is to be completed.

- **Completion Time (C_j):** it is the time at which the job j is completed in a sequence. Some, sample performance measures are flow time, Lateness, tardiness etc.
- **Flow time (F_j):** It is the amount of time job j spends in the system. flow time is a measure which indicates the waiting time of jobs in a system.
- This in turn gives some idea about in-process inventory due to a schedule. It is the difference between the completion time and the ready time of the job j.

$$F_j = C_j - r_j$$

- **Lateness (L_j):** It is the amount of time by which the completion time of job j differs from the due date ($L_j = C_j - D_j$).
- Lateness is a measure which gives as idea about conformity of the jobs in a schedule to given set of due dates of the jobs.
- Lateness can be either positive lateness or negative lateness. Positive lateness of a job means that the job is completed after its due date.
- Negative lateness of a job means that the job is completed before its due date. The positive lateness is a measure of poor service.
- The negative lateness is a measure of better service.
- In many situations distinct penalties and other costs are associated with positive lateness, but generally, no benefits are associated with negative lateness.
- Therefore it is often desirable to optimize only positive lateness.

Tardiness (T_j): Tardiness is the lateness of job j if it fails to meet its. Due date, or zero otherwise

$$T_j = \max(0, C_j - d_j)$$
$$= \max(0, L_j)$$

Measures of performance : The different measures of performance which are used in the single machine scheduling are listed below with their formulas.

Mean flow time : $\overline{F} = \frac{1}{n}\sum_{j=1}^{n} F_j$

Mean tardiness : $\overline{T} = \frac{1}{n}\sum_{j=1}^{n} T_j$

Maximum flow time : Fmax = max (F_j)
= $i \le j \le n$

Maximum Tardiness : Tmax = max (T_j)
= $i \le j \le n$

Number of tardy jobs : NT = $\sum_{j=1}^{n} f(T_j)$

Where

f (T_j) = 1, if $T_j > 0$, and

f (T_j) = 0, otherwise

Shortest Processing Time (SPT) Rule to Minimize Mean Flow Time

In single machine scheduling problem, sequencing the jobs in increasing order of processing time is know as the shortest processing time (SPT) sequencing. Some times we may be interested in minimizing the time spent by jobs in the system. This, in turn, will minimize the in-process inventory. Also we are interested in rapid turnaround /throughput times of the jobs. The time spent by a job in the system is nothing but its flow time, (F bar). Shortest processing time (SPT) rule minimizes the mean flow time

Example : Consider the following single machine scheduling problem.

Job (j)	**1**	**2**	**3**	**4**	**5**
Processing time (tj) (hrs)	**15**	**4**	**5**	**14**	**8**

Find the optimal sequence which will minimize the mean flow time and also obtain the minimum mean flow time.

Solution. No. of jobs =5
Arrange the jobs as per the SPT ordering

Job (j)	**2**	**3**	**5**	**4**	**1**
Processing time	**4**	**5**	**8**	**14**	**15**

Therefore the job sequence which will minimize the mean flow time is 2-3-5-4-1

Computation of $\bar{F}_{\min}$

Job (j)	**2**	**3**	**5**	**4**	**1**
Processing time	**4**	**5**	**8**	**14**	**15**
completion time	**4**	**9**	**17**	**31**	**46**

since the ready time $r_j = 0$ for all j, the flow time F_j is equal to C_j for all j.

$$\bar{F} = 1/5\sum_{j=1}^{5} F_j$$
$$= 1/5(4+9+17+31+46)$$
$$= 1/5(107)$$
$$= 21.4 hours$$

Hence the optimal mean flow time = 21.4 hours.

Sometimes, the jobs in a single machine scheduling problem will not have equal importance. Under such situations, each job is assigned a weight The mean flow time is computed after considering is called weighted mean flow time which is shown below.

$$\bar{F}_w = \frac{\sum_{j=1}^{n} w_j F_j}{\sum_{j=1}^{n} w_j}$$

WSPT RULE

In single machine scheduling problem, sequencing the jobs in increasing order of weighted processing time is know as weighted shortest processing time (WSPT) sequencing the weighted processing time of a job is obtained by dividing its processing time by its weight.

Example 1. Consider the following single machine scheduling problem with weights. Determine the sequence which will minimize the weighted mean flow time. Also find the weighted mean flow time.

Job (j)	**1**	**2**	**3**	**4**	**5**
Processing Time (T_j)	**15**	**4**	**5**	**14**	**8**
Weight (W_j)	**1**	**2**	**1**	**2**	**3**

Job (j)	**1**	**2**	**3**	**4**	**5**
Processing Time (T_j)	**15**	**4**	**5**	**14**	**8**
Weight (W_j)	**1**	**2**	**1**	**2**	**3**
T_j / W_j	**15**	**2**	**5**	**7**	**2.67**

Arrange the jobs in the increasing order of t_j / w_j (i.e. WSPT ordering). From the above table, we get the following relation.

$$t_j / w_2 \le t_5 / w_5 \le t_3 / w_3 \le t_4 / w_4 \le t_1 / w_1$$

There fore optimal sequence which will minimize the weighted mean flow time is, 2-5-3-4-1
F_w Bar Calculation:

Job (j)	**2**	**5**	**3**	**4**	**1**
T_J	**4**	**8**	**5**	**14**	**15**
C_J (F_J)	**4**	**12**	**17**	**31**	**46**
W_J	**2**	**3**	**1**	**2**	**1**
F_J * W_J	**8**	**36**	**17**	**62**	**46**

Therefore,

$$\bar{F}_w = \frac{\sum_{j=1}^{n} w_j F_j}{\sum_{j=1}^{n} w_j}$$

$$= \frac{(8+36+17+62+46)}{(2+3+1+2+1)} \quad = \frac{169}{9} \quad = 18.78 hours$$

Earliest Due Date (EDD) Rule to Minimize Maximum Lateness

The lateness (L_j) of a job is defined as the difference between the completion time and the due date of that job. L_j Can be either positive or negative values

$$L_j = C_j - d_j$$

The maximum job (L_{max}) and the maximum job tardiness (T_{max}) are minimized by Earliest due date (EDD) sequencing. In single machine scheduling problem, sequencing of jobs in increasing order of due date is know as "Earliest due date" rule.

Example 1.

Consider the following single machine scheduling problem. Determine the sequence which will minimize the maximum lateness (L_{max}) also, determine the (L_{max}) with respect to the optional sequence .

Job (j)	**1**	**2**	**3**	**4**	**5**	**6**
Processing Time C_J	**10**	**8**	**8**	**7**	**12**	**15**
Due date D_J	**15**	**10**	**12**	**11**	**18**	**25**

Solution: Arrange the job as per EDD rule (i.e. in the increasing order of their due dates). The EDD sequence is 2-4-3-1-5-6. This sequence give the minimum value for (L_{max}).

Computation of (L_{max}) :

Job (j) (EDD sequence)	**1**	**2**	**3**	**4**	**5**	**6**
Processing Time T_J	**8**	**7**	**8**	**10**	**12**	**15**
Completion time C_J	**8**	**15**	**23**	**33**	**45**	**60**
Due date D_J	**10**	**11**	**12**	**15**	**18**	**25**
Lateness L_J	**-2**	**4**	**11**	**18**	**27**	**35**

.

From the above, the maximum (L_j) is 35. this is the optimized (minimized) value for (L_{max}). The (L_{max}) of any other non-EDD sequence will not be less than35

Minimizing The Number of Tardy jobs

If a job is completed beyond its due date, then it is called tardy job; otherwise it is called non-tardy job. In many organizations, the objective may be to minimize the total number of tardy jobs. If the EDD sequence yields zero tardy or it yields exactly one tardy job, then it is an optimal sequence for minimizing the total number of tardy jobs (NT). If it yields more than one tardy job, the EDD sequence may not yield the optimal solution. An exact algorithm for the general case is given below

a) First, a set (E) of early jobs, in EDD order b) Then, a set (L) of late jobs, in any order

This algorithm gives optimal sequence which will result in minimum number of tardy jobs (NT).

Hodgson's Algorithm to minimize (NT).

Step 1. Arrange the jobs in EDD order and assume this as set E. Let set L be empty.

Step 2. If no jobs in E are late, then stop. Find the union of E and L (Note: The remaining jobs in E should be in EDD order. But the jobs in L can be in any order);

Step 3. Identify the longest job, among the first K jobs in the sequence. Remove this job from E and place it in L. Revise the completion times of the jobs remaining E and return to step 2.
This algorithm is demonstrated using the following problem

Example 1.
A computer systems consulting company is under contract to carry out seven projects, all with deadliness assured in days from now. The consultants are a small group and they work together on each project, so that the project will be started and completed sequentially. Under the terms of contract, the consultants will receive Rs 24,000 for each project completed on time, but they will incur Rs 40,000 in penalties for each project completed late. Each project has an associated duration ,which is the anticipated number of days required to carry out the project as shown below. How should the projects be sequenced in order to maximize net revenues?

Project ID	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Duration (tj)	**2**	**4**	**6**	**8**	**10**	**12**	**14**
Deadlines (dj)	**6**	**12**	**30**	**19**	**12**	**18**	**24**

Solution:

From the statement of the problem, one can identify that the objective is to maximize net revenues. This can be achieved by simply obtaining a sequence which will minimize the number of tardy jobs (NT). Hence we apply Hodgson's algorithm to minimize (NT).

Step 1. The earliest due date order is shown below:

d_j	**6**	**12**	**12**	**18**	**19**	**24**	**30**
Project $_j$	**1**	**2**	**5**	**6**	**4**	**7**	**3**

Place the above sequence of projects which is in EDD order in set E. Therefore,

Set E = (1,2,5,6,4,7,3)

Set L = (Empty).

Step 2. The lateness of the projects are checked as shown below

Project (j)	**1**	**2**	**5**	**6**	**4**	**7**	**3**
Duration (tj)	**2**	**4**	**10**	**12**	**8**	**14**	**6**
Completion time (cj)	**2**	**6**	**16**	**28**	**36**	**50**	**56**
Due date (dj)	**6**	**12**	**12**	**18**	**19**	**24**	**30**
Tardy/Non-tardy (1/0)	**0**	**0**	**1**	**1**	**1**	**1**	**1**

In the above table, in the last row, 0 means that the project is non-tardy and 1 means that the project is tardy. As per the sequence in the set E, there are five tardy projects. The first tardy project is 5 which is in the third position [3].

Step 3. The project with the largest duration among the first three projects in the sequence is 5. Remove this project and append it to L. Therefore

L = {5}

E = {1,2,6,4,7,3}

The completion times of the projects in the set E are revised as shown below

Project (j)	**1**	**2**	**6**	**4**	**7**	**3**
Duration (tj)	**2**	**4**	**12**	**8**	**14**	**6**
Completion time (cj)	**2**	**6**	**18**	**26**	**40**	**46**
Due date (dj)	**6**	**12**	**18**	**19**	**24**	**30**
Tardy/Non-tardy (1/0)	**0**	**0**	**0**	**1**	**1**	**1**

Step 2. From step 3, it is clear that there are three tardy projects. The first tardy project is 4 which is in the fourth position [4] of the sequence in the set E.

Step 3. The project with the longest duration among the first four projects in the sequence is 6. Remove the project 6 and append it to L. Therefore

E = {1,2, 4,7,3}
L = {5,6}

The completion times of the projects in the set E are revised as shown below

Project (j)	1	2	4	7	3
Duration (tj)	2	4	8	14	6
Completion time (cj)	2	6	14	28	34
Due date (dj)	6	12	19	24	30
Tardy/Non-tardy (1/0)	0	0	0	1	1

Step 2. From the table in step 3, it is clear that there are two tardy projects. The first tardy project is 7 which is in the fourth position [4] of the sequence in the set E.
Step 3. The project with the longest duration among the first four projects in the sequence is 7. Remove this job from the set E and append it to L. Therefore

E = {1,2, 4,,3}
L = {5,6,7}

The completion times of the projects are revised as shown below

Project (j)	1	2	4	3
Duration (tj)	2	4	8	6
Completion time (cj)	2	6	14	20
Due date (dj)	6	12	19	30
Tardy/Non-tardy (1/0)	0	0	0	0

Step 2. From the table in the previous step, it is clear that all the projects are non-tardy. Hence we reached the optimal sequence in the set E. Now merge E and L to get the complete sequence.

Final sequence = E U L = {1,2,4,3,5,6,7} In the above optimal sequence, total number of tardy projects is 3 which the minimum value.

JOB SHOP SCHEDULING

Introduction:

In flow shop scheduling problem, we assume that each job has m different operations. If some of the jobs are having less than m operations, required number of dummy operations with zero process times are assumed. By this assumption, the condition of equal number of operations for all jobs is ensured.

- The process sequences of the jobs are not same.
- The flow of each job is not unidirectional in job shop Scheduling.
- Heuristic approaches are popular in this

Types of Schedules

In general, infinite number of feasible schedules are possible for any job shop problem, because one can insert any arbitrary amount of idle time at any machine between adjacent pairs of operations. These idle times are not useful in any sense. In fact, these will lead to non-optimal solution while minimizing makespan measure.

Two jobs and M Machines Scheduling

Two jobs and M Machines Scheduling is a special problem under job shop scheduling. The problem consists of 2 jobs which require processing on M machines. The processing sequences of the jobs are not the same. Since, this is a special kind under the job shop scheduling like, Johnson's problem (n jobs and two m/c). Under flow shop scheduling, we have a graphical procedure to get optimum schedule. The graphical procedure consists of the following steps.

Step 1: Construct a two dimensional graph in which x-axis represents the job 1, its sequence of operations and their processing times, and y-axis represents the job 2, its sequence of operations and their processing times (use same scale for both x-axis and y-axis).

Step 2: Shade each region where a machine would be occupied by the two jobs simultaneously.

Step 3: The processing of both jobs can be shown by a continuous line consisting of horizontal, vertical and 45 degree diagonal lines. The line is drawn from the origin and continued to the upper right corner by avoiding the regions. A diagonal line means that both jobs can be performed simultaneously.

So, while drawing the line from the origin to the top right corner, we should try to maximize the length of diagonal travel (sum of the lengths of 45 degree lines) which will minimize the makespan of the problem. Using trial and error method, one can draw the final line which has the maximum diagonal portion. This concept is demonstrated using a numerical problem.

Processing Two Jobs Through m machines

The problem can be described as

1. There are m machines denoted by A1, A2, A3-----Am
2. One has to perform only two jobs ie job 1 and job 2
3. The technological ordering of each of the two jobs through m machines is known in advance.
   Such ordering need not be the same for both the jobs.
4. The exact or expected processing times are known.

The problem is to minimize the total elapsed time T ie the time from the start of the first job to the completion of the second job. Graphical approach to the two – job m- machines is rather simple to apply and usually leads to good through not necessarily optimal) results. The following example will make the graphical procedure clear.

Example 1. Use graphical method to minimize the time needed to process the following jobs on the machines shown below also calculate the total time needed to complete both the jobs.

Job1	**Sequence of machines time**	**A2**	**B3**	**C 4**	**D6**	**E2**
Job 2	**Sequence of machines time**	**C4**	**A 5**	**D 3**	**E2**	**B6**

Solution:

Step -1: First draw a set of axes, where the horizontal axis represents processing time on job 1 and the vertical axis represents processing time on job 2.

Step -2: layout the machine time for two jobs on the corresponding axes in the technological order machine A takes 2 hours for job 1 and 5 hours for job 2. Construct the rectangular PQRS for the machine A. Similarly, other rectangular for machine B,C,D and E are constructed as shown in fig.

Step -3: make a program by starting from the origin O and moving through various stages of completion (points) until the point marked finish is obtained. Physical interpretation of the path thus chosen involves the series of segments which are horizontal or vertical or diagonal making an angle of 450 with the horizontal. Moving to the right means that job1 is proceeding while job 2 is idle and moving upward means that job 2 is proceeding while job 1 is idle and moving diagonally means that both the jobs are proceeding simultaneously. Further both the jobs cannot be processed simultaneously on the same machine. Graphically diagonal movement through the blocked out (shaded) area is not allowed and similarly for other machines too.

Step 4: (To find an optimal path) An optimal path is one that minimizes idle for job1 (horizontal) and job 2 (vertical movement). Choose such a path on which diagonal moment is as much as possible.

Step – 5: (to find the elapsed time). The elapsed time is obtained by adding the idle time either of the job to the processing time for that job. In this problem the idle time for the chosen path is seen to be 3hrs for the 1 and 0 for the job 2. Thus the total elapsed time, 17 + 3 = 20hrs