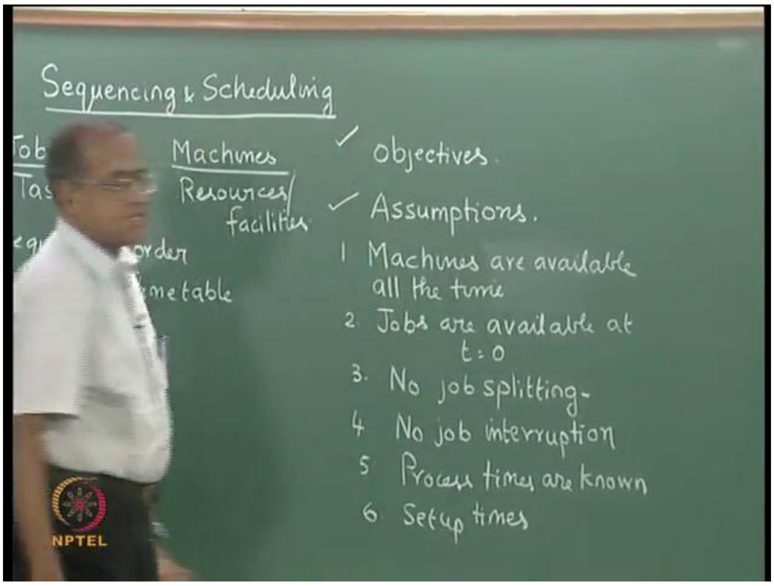
Operations and Supply Chain Management

Prof. G. Srinivasan
Department of Management Studies
Indian Institute of Technology, Madras

Lecture - 24

Sequencing and Scheduling - Assumptions, Objectives and Shop Settings

(Refer Slide Time: 00:20)



In this lecture, we consider the Sequencing and Scheduling problem. The sequencing and scheduling problem talks about, a set of jobs or a set of tasks, that have to be performed on given machines or a given set of resources or facilities, in it is simplest term the problem deals with performing a certain given jobs on a set of given machines. The jobs have to be carried out on the machines, such that certain objectives are met, certain constraints are satisfied. Now, the scheduling and sequencing is to try and meet certain objectives, we will get into detail about these objectives, but simple objectives would be like to try and finish the jobs before they have to be delivered.

Now, there is a difference between scheduling and sequencing. Sequencing essentially refers to the order, in which a sequence represents an order or a list or an order, in which the jobs are sent. The schedule talks about the time table or gets into detail saying in between this time and this time, this particular job is getting processed on a particular machine. So, we need to understand the essential difference between the two, while sequence represents the order, in which things have to be done, schedule represents the timing or time table of the activities.

In certain sequencing and scheduling problems, we will use the word sequencing and in certain other problems, we will use the word scheduling, when the time table automatically follows the order, in which the jobs are sent then we use the term sequence, where sequence also includes the schedule. In situations where the time table has to be explicitly stated, we use the word schedule, as we move along, we will see places where, we use the word sequence and places where, we use the word schedule.

Now, sequencing and scheduling problems in the context of what we are looking at is essentially about performing a set of jobs on a set of machines, now there are certain assumptions in sequencing and scheduling.

Students: ((Refer Time: 03:17))

So, we start with the assumptions and these assumptions are in a given set of machines are all available, machines are available continuously all the time, there is no machine interruption or machine down time or anything like that, machines are available as they are. The jobs or tasks that have to be performed, the jobs that are available all the jobs that or tasks that have to be performed are available at the start of the scheduling period and are continuously available.

So, at time t equal to 0, we assume that all the jobs that have to be processed are available, which also means at this point in time, we are making an assumption that the jobs do not come late. There could be situations where, we might start the sequencing and scheduling process with the available jobs and as we move along some other jobs may arrive and then they may join the system, at present we are not looking at that assumption.

The assumption is that all the jobs are available at time t equal to 0, once a job is started on a machine, we are going to assume that the job will be completed and only then another job will be taken up for processing, on a particular machine. So, we could call that as, no job splitting or the first assumption of course is no job splitting, which means that, if a particular job or a task requires say 10 minutes on a particular machine. We do not and there are 2 machines that are capable of doing, it we do not split it into first 6 on 1 machine and the next 4 on the other.

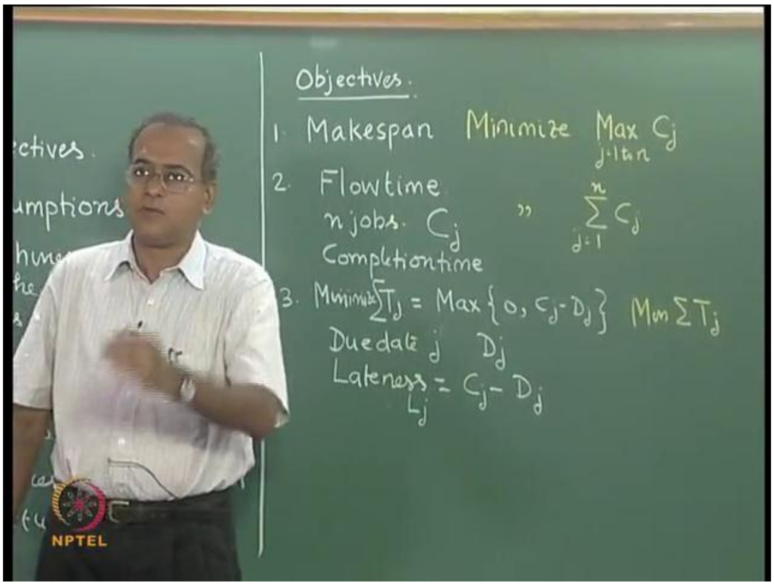
What we assume here is once we pick a machine to do that job or a task, the entire task is completed on that machine, so that is an assumption, then we also have no job interruption. Now this assumption means that, if I have taken up a particular job on a machine and let us say it requires 10 minutes, once I take it up, I complete this job and only then I pick up another job for processing, I am going to assume that in between if there is a more important job that has come.

I will not stop processing on this keep it out take the more important one finish it and then get back to this, that assumption we that kind of a thing, we will not do and that is an assumption, which means once taken up a job will be completed and only then the next job will be taken up for processing, so these are some of the simple assumptions. Next one is the processing times are known and they are deterministic, we are not assuming that, processing times follow any distribution or anything like that, processing times are known and deterministic.

Setup times, we also are aware that, if we take up a job for processing on a machine, then we need to setup the machine to process a particular job. So, we are going to assume that setup times are small and setup times are included in the processing times etcetera, so there is also an assumption about the setup times. So, these are some simple assumptions, which lead to very basic simple scheduling problems, as we relax some of these assumptions scheduling problems become more involved and little more complicated.

And in this lecture series most of the times, we are going to make all these assumptions and then we are going to proceed to develop some algorithms, which will solve sequencing and scheduling problems. Now having seen some of the assumptions, we will now see some of the objectives that are there in sequencing and scheduling.

(Refer Slide Time: 08:45)



Now the first objective and the most popular objective that, we will be looking at is called Makespan in simple terms Makespan represents the span of time, that is required to make a set of jobs or to process or manufacture a set of jobs. So, if a certain set of jobs have to be done, on a certain set of machines, now there is a point t, let us say a point at which we start, which we call as t equal to 0 and then at some point all the jobs are completed.

Now that point from t equal to 0 to that point is the amount of time that, we have consumed, to do or perform all the tasks or to do all the jobs. Now that is called the Makespan, that is the amount of time that we have taken from t equal to 0, to complete all the jobs, now that Makespan we want to keep it as small as possible. So, that all the jobs are completed at the earliest, so the first objective is to do is to is to look at the Makespan and then try to minimize the Makespan, we want to have the Makespan as small as possible.

The second objective is called the flow time, now if we have n jobs to be done, on a set of machines, now all the jobs are available at time equal to 0, they may start processing at time equal to 0 or they may start processing later. But, each and every one of these jobs gets completed at a certain point, the completion indicates, the time at which the last operation or the last the machine that the job has to visit last has been visited and the operation has been completed on that machine.

So, if we call completion time for the j-th job as C j, the time at which the j-th job is completed that is called C j, then automatically the Makespan is the maximum of this C j values. If we have ten jobs, each job has a completion time C 1 C 2 up to C 10, it is not necessary that C 2 has to be bigger than C 1 and so on. Now, at some time all of them are over, so the maximum of C 1 to C 10 or C 1 to C n in general indicates the Makespan and we want to minimize the Makespan, which is essentially to minimize the maximum of C j, j equal to 1 to n.

There are n jobs each is going to have a completion time, so C j is the completion of the j-th job, the maximum of the completion times is the Makespan and we want to minimize the maximum of C j. Now, each job is completed at time C j, so the completion time is called C j, let me write flow time, as well as completion time, completion time is C j, now here what we want to do is all of them start at time equal to 0. So, in some sense, we want all the jobs to be completed individually, but then we want the completion times of each one of the job to be as small as possible.

So, instead of minimizing the maximum of the C j's, here we end up minimizing the total of the C j minimize sigma C j j equal to 1 to n, so the completion time of job j is C j. Now, we want to minimize the sum of all the completion times, which is an indication that each job in some sense has spent as less a time as possible in the system, at the same time has got all the tasks completed and is ready to leave. Now, flow time as such by the very name it suggests the amount of time the job flows in the system.

So, if all the jobs are available at time equal to 0, then the completion time represents the flow time, if all of them are available at time equal to 0 and for example, a particular job finishes in 10 hours, then 10 hours is the time, at which it is there in the system. So, its completion time is 10 hours, it is flow time is also 10 hours, but on the other hand, if we relax one of the assumptions, that all jobs are available at time equal to 0 or the job is available at time equal to 0, but the job is taken up for processing first say at time equal to 2.

And then it finishes at time equal to 10, the completion time is still 10, but the flow time is 8, because it started at 2 and finished at 10. So, we would like to in this particular objective, we could define it as some of the completion times that, we try to minimize, so we minimize the mean completion time or we minimize the total completion time, the mean completion time or average completion time is the total completion time divided by n, which is the number of jobs.

There is another reason to look at this particular objective, if we start looking at inventories in the system, inventories in any manufacturing system can be categorized into 3 inventories. One is called the raw material inventory, the other is called work in progress or work in process inventory and then the finished goods inventory, now when we talk of sequencing and scheduling problems, we are talking of a manufacturing facility or a plant or a shop where jobs come get processed and they leave the system.

So, we are talking about work in process inventory, when we look at sequencing and scheduling problems, so the time at which each job spends within the manufacturing facility or within the shop floor as it is called represents the amount of inventory that the shop floor is having. So, by minimizing the sum of completion times, we essentially try to minimize the work in process inventory inside the shop floor.

So, this is an objective, which primarily aims at minimizing the inventory in the system, inventory as in work in process inventory in the manufacturing shop floor. Then we look at a third objective, where we start looking at the jobs from a different angle. Now these jobs and tasks are carried out and they are carried out, they are finished and then they ultimately go to a customer, these jobs are there, because a customer has ordered, these jobs.

So, let us say that each job j, when the job is taken up for processing, there is a time, which is called the due date, which indicates the time, at which this job is expected to be finished and delivered. So, we would say that, there is a due date associated with each job j, which is called D j, now this due date can be internally created or this due date can be externally created. Now, when a particular task is taken up for manufacture and supply to a customer, there is a due date, which, at which time this is to be delivered to the customer.

Now from manufacturing, it might take some more time for the finished goods to move and ultimately reach the customer, so from the customer due date, we could always derive the manufacturing due date or the due date or time, at which we expect this job to be completed by this manufacturing facility. So, in all our discussion due date would simply mean, the time at which this job is expected to be finished by the manufacturing facility.

So, each job has it is due date, now obviously, we want to meet the due dates, because not meeting the due date is going to create a lot of trouble for manufacturing, any delay the customer is not willing to wait, we have already seen that the customer is extremely demanding. And therefore, it is the job of manufacturing to try and finish all their activities within time, so the primary objective is to avoid delays. And then if we start looking at delay can be defined as, if the completion time C j exceeds the due date D j then there is a delay.

Now what we do is we define a very generic term called lateness, where lateness is how late the job is, so lateness is defined as C j minus D j, C j is the time, at which the job is completed D j is the time, at which it is due. So, C j minus D j is called lateness and it is used general term L j is used to represent lateness, now by the very definition of L j, L j can be positive or L j can be negative. Now L j becomes positive, when C j is greater than D j, which means there is actually a delay, now if a job is completed before the time, it is due, then C j minus D j becomes negative and it is early.

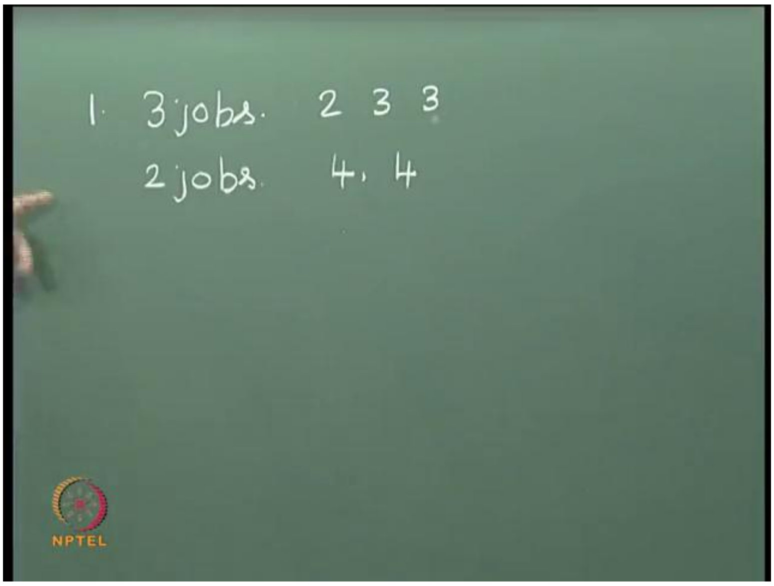
So, lateness has 2 components, which is called earliness and tardiness, earliness is when the job is completed ahead or before the due date, tardiness is when the job is completed after the due date. So, tardiness would mean that C j minus D j is positive, to begin with we would say that, we are if we are able to complete the job early, but then we do not want the job to be tardy.

So, there is a tardiness T j associated with jobs that exceed the due dates, so using the definition of lateness, tardiness T j is max of 0 comma C j minus D j, now this is not difficult to follow, because if C j minus D j is negative and the job is early, then the maximum value is 0, therefore, the tardiness is 0. If C j minus D j is positive, the job is late and tardy then this will take a positive value and therefore, T j will be equal to that, so each job, now has a T j.

Now this T j can be either 0 or this T j can be positive. Now, we are concerned about this T j being positive, and that positive T j represents the delay or the extent, to which the job goes behind the schedule. So, now, the objective would be to try and minimize, the total the sum of the individual tardiness, so the 3rd objective is to minimize sigma T j, which is sum of the tardiness, so minimize T j and note that this T j can only be 0 or it can be a positive quantity, T j cannot be a negative quantity by the very definition here.

So, the third objective that we can think of in sequencing and scheduling is to try and minimize the sum of the tardiness of the individual jobs, which is commonly mentioned as minimizing total tardiness. Then we move to a 4th objective, now let us assume that, we can some jobs are tardy, we are unable to complete them in time some jobs are tardy, let us look at two situations.

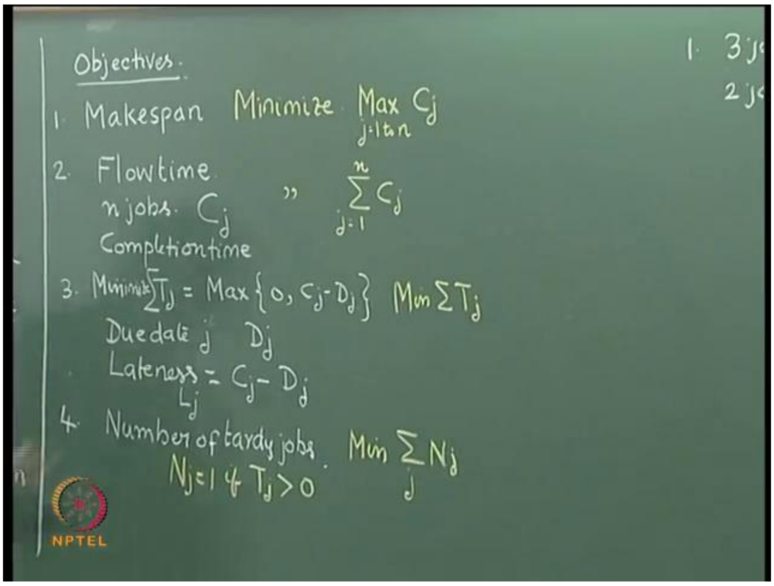
(Refer Slide Time: 22:33)



One situation where, there are 3 jobs that are tardy and the sum of the tardiness are 2 3 and 3, which means they are behind by let us say 2 days, 3 days and 3 days respectively, there is another situation, where a 2 jobs that are tardy, let us say the situation is 4 and 4, the 2 jobs both are 4 and 4. Now, if we go by minimizing the total tardiness, then we would calculate that the total tardiness is 8 but then 3 jobs are tardy 2 jobs are tardy.

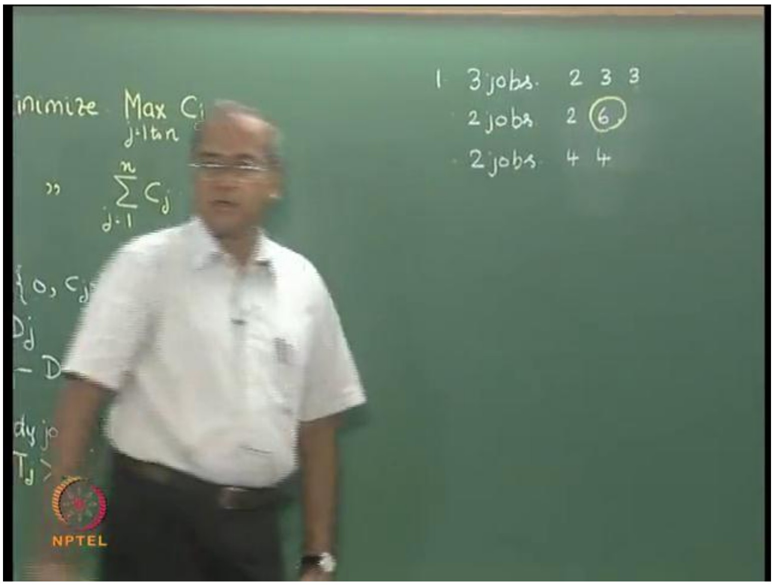
Now, it depends on the situation, if these 3 jobs go to 3 different customers, then in some sense 3 customers are unhappy, because the jobs are late, whereas if these goes to 2 customers then two customers are unhappy. So, essentially in addition to worrying or being concerned about the delays, we should also look at the number of jobs, that are late or that are tardy. So, number of jobs that are tardy also becomes a meaningful objective and that at time, which depends on the number of customers, who are getting affected, because of the tardy jobs that are there in the scheduling system.

(Refer Slide Time: 24:14)



So, the fourth objective will be to minimize the number of tardy jobs, so N j represents the N j equal to 1, if T j greater than 0, which means this job is tardy, if it has a positive tardiness. And then we try and minimize N j, the number of tardy jobs or number of jobs that go behind the schedule, going back to the same example, sometimes we might even look at a situation like this.

(Refer Slide Time: 25:03)

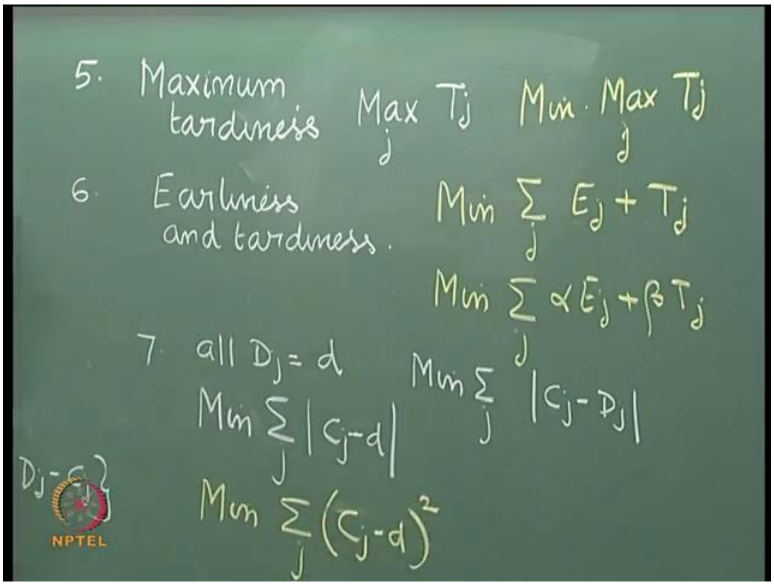


There are 2 jobs that are tardy, but the tardiness is 4 and 5 in terms of total tardiness, this is preferred, but in terms of minimizing the number of tardy jobs, this would be preferred. So, it depends on the situation, it depends on the number of different customers to whom it is going, now let us look at another situation like this. Now, I have one situation with 2 and 6 and I have another situation, where there are 2 tardy jobs with 3 and 5.

Once again based on total tardiness is 8, we could choose any one of them based on minimizing the number of tardy jobs, this has 2 jobs this has 2 jobs, so let us say, we would prefer these 2 schedules to this, which has 3 jobs, which are tardy. Now, once again between the 2, if you see very carefully, here is the situation, where one job is late or tardy by 2 days, another job is late and tardy by 6 days, now this is a situation, where it is 3 and 5.

In fact, let us make it 4 and 4, which also adds up to 8, now based on minimizing number of tardy jobs, we could take this or we could take this, but then which one would we prefer. In a way, we would prefer, this because there are two customers, who are going to be delayed again two customers, let us say who are going to be delayed assuming that, these jobs go to different customers, but then this person is going to be extremely happy, because the extent of delay is very, very high. So, in addition to minimizing the total tardiness and minimizing the number of tardy jobs, we should also try and reduce this number to the extent plus, so the objective also moves to trying to find out.

(Refer Slide Time: 27:21)



So, we go to the fifth objective, which actually tries to look at the maximum tardiness, which is max over j T j and then we try to minimize max over j T j. So, minimizing the maximum tardiness also becomes another objective. Then we look at the sixth objective, write from 3 4 and 5, we assumed that, it was to complete a job early. So, we complete the job early, let it go to the finished goods inventor area, let it stay there wait till the due date is reached and let it be shipped to the customer depending on the due date.

Today manufacturing organisations, more and more tend to follow, just in time manufacture, so if a job is completed ahead of schedule or early you cannot immediately ship, it to the customer or send it to the customer, because the customer does not want it. If the customer is practising just in time the customer would say, if you are early by 2 days, do not send me, today send it to me exactly on my due date, therefore earliness is also not a desirable thing.

So, the objective now shifts to computing, both earliness and tardiness, so here it is to minimize E j plus T j, E j is earliness of j, which has been defined earlier, but not written here. So, earliness of j E j will now be E j will be max of 0 D j minus C j not over j max of D j minus C j, so you try and minimize both earliness, as well as tardiness. Now, we also know that, for any given job j, it will have either a positive earliness or it will have a positive tardiness or it will have 0, which is completed exactly at the due date, earliness is 0 tardiness is also 0.

Now, earliness represents the additional cost of holding finished goods inventory, the job is completed ahead of schedule, it is going to stay in finished goods inventory and then it is going to be transported. So, it represents some kind of an inventory cost, T j represents some kind of a cost associated with a late delivery and the cost associated with late delivery, essentially have 2 important components. One is when the job is completed later than the due date, what organisations normally try to do is to try and use different modes of transport.

So, that the transportation time can be reduced, so many times instead of sending something by road people may end up sending it by air, so there is an increased transportation cost, because of this tardiness T j. The second and the most important cost is the loss of customer goodwill, which gets reflected in the customers business being affected and then at some point eventually, the customer is so unhappy, that the customer decides to leave.

So, loss of customer goodwill is a very large component of the cost associated with tardiness and the difficult part is that, the loss of the cost associated with loss of customer goodwill is not easy to measure and compute. So, T j is right now, T j represents time as far as our objective function is concerned, but there is a cost associated with that T j, now when we try and add E j and T j, when E j and T j are actually time measures, it is to add them as they are.

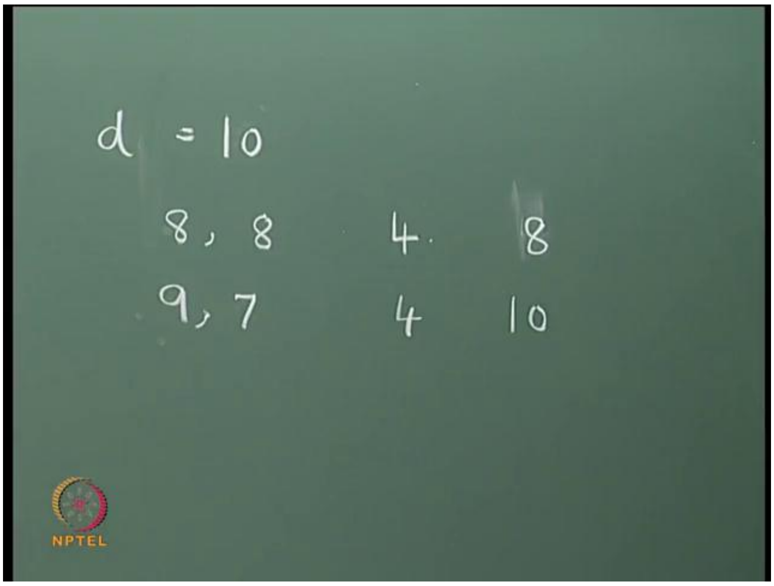
But, if we are going to add the cost associated with them, we will quickly understand that though both earliness and tardiness are not desirable the cost associated with earliness is a little cheaper than, the cost associated with tardiness. So, you do not want to add both of them instead, you want to do a weighted objective function, which is to minimize over j something like an alpha E j plus the beta T j, where alpha and beta are the weights.

But, beta is bigger than alpha, because the difficulties associated with tardiness are far more than the difficulties associated with earliness, but also the moment we start doing this alpha and beta then we get into issues of however, we are going to define the weights alpha and beta. Then we move to the seventh objective, which also deals with earliness and tardiness, but then looks at it slightly differently, now let us look at this, let us first look at this definition of minimize E j plus T j, under the assumption that, both earliness and tardiness are equally undesirable.

So, both alpha and beta the weights become 1, so you minimize earliness plus tardiness, now let us go back and go to these definitions, so what we do here is minimize summed over j, E j is max of 0 comma D j minus C j and tardiness is max of 0 comma C j minus D j. So, essentially either earliness or tardiness is nothing but the absolute value of C j minus D j, if it is 0, it is not going to contribute, if the term inside is positive, which means completion time is after the due date, then it is tardiness the absolute value will be positive.

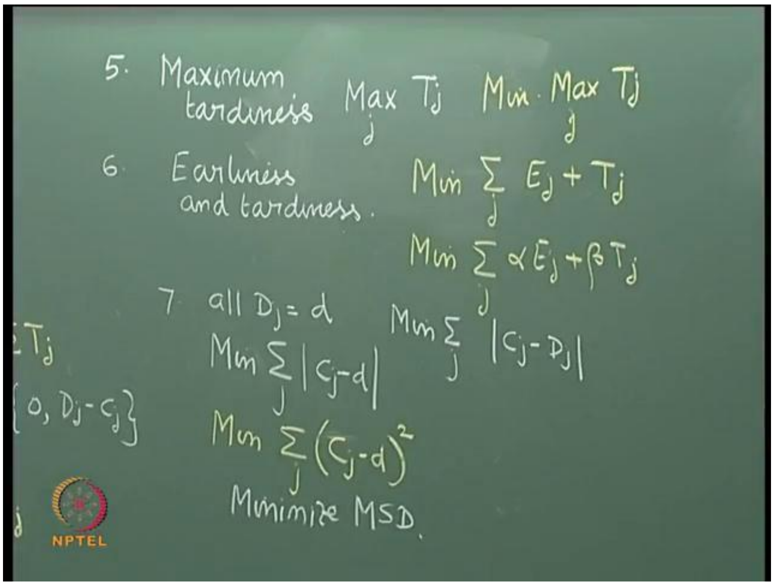
If C j minus D j is negative, which means it is early the absolute value would still be positive, so this adequately represents this, so this is nothing but if there is a due date D j associated with job j, then the objective is to try and minimize the sum of the absolute value or absolute difference between C j and D j. Now, if all due dates are equal, if all D j equal to d, which means all the jobs have the same due date, which means these are all jobs that have to be sent by today evening or by a Friday evening say then with this becomes minimize C j minus d. If we make a further assumption here that, we want to give larger penalties for larger deviations then this can be approximated to minimize, now when we square the differences, we give a larger penalty for a larger deviation.

(Refer Slide Time: 35:57)



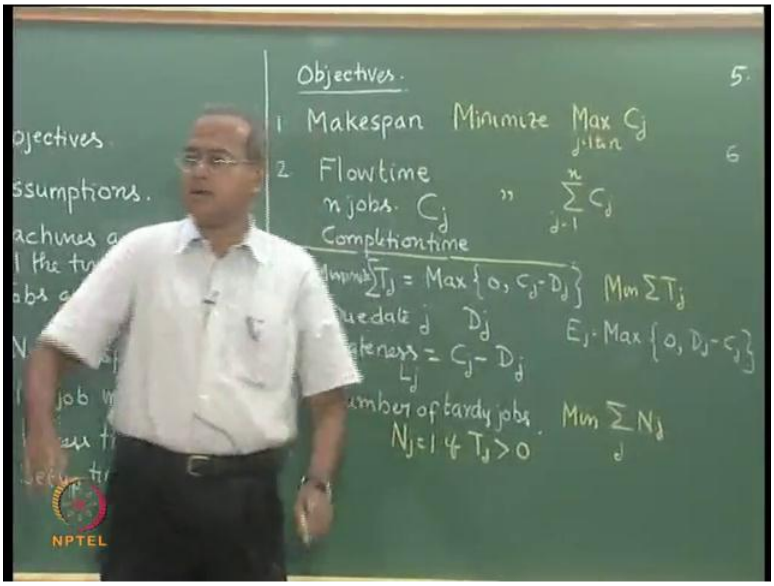
For example, If due date is equal to 10 or if D equal to 10 D equal to 10, if we have two completion times, say which is 8 and 8 then some of the absolute differences would still be 10 minus 8 plus 10 minus 8, which is 4. If we have 2 difference, 2 due dates, which is 9 and 7, some of the absolute differences would be 1 plus 3, which is 4, but if you do the square, then this will become 2 square plus 2 square, which is 4, which is 8, here it will become 1 square plus 3 square, which is 10. So, we would prefer this to this, because this gives larger penalty for larger deviation, so if we want to give larger penalty, for larger deviation instead of using the absolute value, you could use the square.

(Refer Slide Time: 37:13)



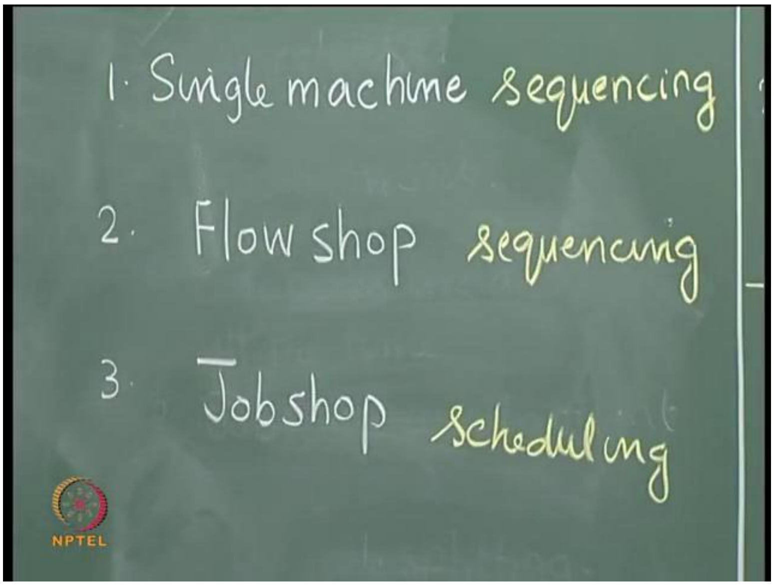
So, this is called minimizing mean square deviation of completion times, minimize mean square deviation of completion times, about a given common due date. So, right now we have seen about 7 different objectives, there are other objectives, sometimes the organizations would also like to maximize the utilization of machines. So, maximize machine utilization is another objective that we could think of, now if we look at these 7 objectives that, we have right now seen.

(Refer Slide Time: 37:48)



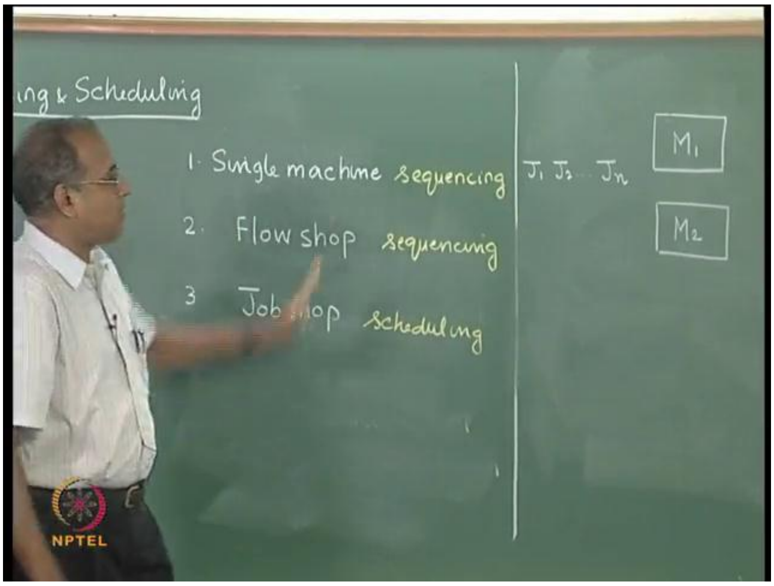
Now you realize that the first 2, do not involve due dates and the rest of them involve, due dates in some form, so the remaining five, 3 to 7 are called external objectives or external measures, where the role of the customer comes in the form of a due date. The first two are internal measures, where the organization within itself is trying to optimize a few things like, early completion less inventory and so on, now these are more important from a customer point of view ok. Now, having seen the objectives and the assumptions, we will also try and spend some time on the type of scheduling models and scheduling and sequencing models that we are going to see. So, we would see three types of scheduling and sequencing problems in this lecture series.

(Refer Slide Time: 38:59)



One is called single machine scheduling, we have flow shop scheduling and we will see job shop scheduling. Actually we will be seeing single machine sequencing problems, we will be actually seeing flow shop sequencing problems and we will see job shop scheduling problems. Earlier, I had mentioned that, when the order alone is enough to represent or describe the entire thing, then we uses the word sequence, but then if the order and the timetable have to be mentioned then we use the word schedule. Now, we will also try and understand the difference between each of them, what is the single machine problem, what is a job shop problem, what is the flow shop problem.

(Refer Slide Time: 40:36)



So, single machine problem as the name suggests has a single machine, which you could call as M as 1 machine and there are several set of jobs, which could be J 1 J 2 J n and these jobs have to be processed on the machine. Sometimes in single machine problems, we could have actually, 2 machine, now it is called parallel processes, so if we have more than one machine in a single machine problem, it may again repeat, if we could have more than one machine in what are called single machine systems.

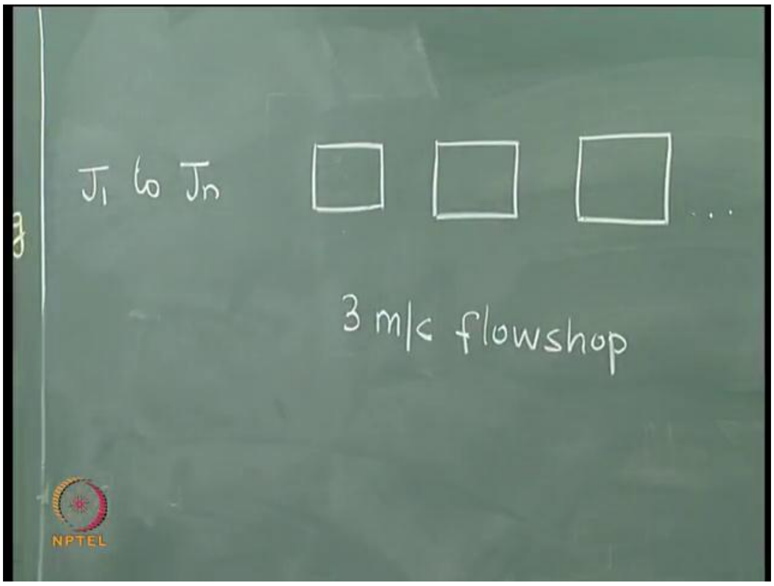
Then those problems are called single machine with identical or with parallel processes, which means it is like the simple example from a non-manufacturing context is it is like someone going to a to a place to pay electricity bill or to book a railway ticket. So, there could be 2 counters, so these 2 counters would be like 2 machines, so in manufacturing for example, this could be a particular machine, which is some kind of a pressed and then there could be another press, which is there, so you could have parallel machines.

In the parallel machine case a job can go to either of them, it can either go to, let us now call look at it in the parallel machine case, so same machine, I could have M 1 as well as M 2. But, then each of these jobs can go to either M 1 or M 2 and they can complete all that is required in either M 1 or in M 2 and then the further assumption is that, in this case, it will go to either M 1 or M 2, it will not visit both, it will be allowed to go to only one of them, complete all the operations and leave.

If these 2 machines are such that, the processing time taken, for a particular job J 1 whether it goes to M 1 or whether it goes to M 2, if the processing is the same, then it is called identical parallel process. If for example, this takes ten minutes, it can go to M 1 or it can go to M 2, it takes 10 minutes on M 1, it takes only 8 minutes on M 2, to do the entire thing. Even then we could send it to M 1, even though it takes a little longer, due to optimizing other objectives, but such situations are called parallel machine scheduling with non identical processing.

If you use the word identical processing, it means the processing time is the same, on either of the machines or both the machines and if you use the word non identical, it means the processing times are different. So, single machine problems can be seen either as the single resource or as a multiple resource with identical and non identical machines, now flow shop is like this.

(Refer Slide Time: 43:50)

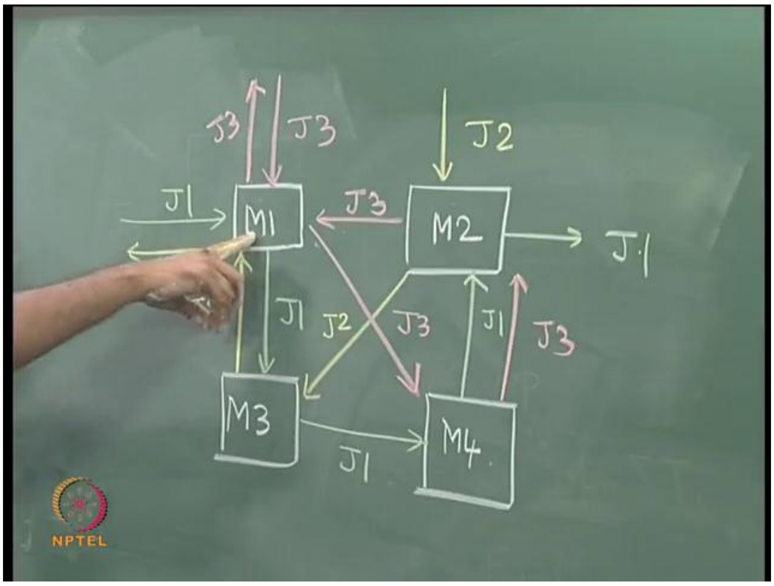


You have a set of jobs, which you call as J 1 to J n and there are a set of machines, which are called M 1 to M m 1 2 3 and so on, so this is called a 3 machine flow shop, which could represent 3 different processes in a manufacturing process. It could represent some kind of a cutting, welding and painting, in sheet metal manufacturing or it could represent some kind of a raw material grinding, mixing and then heating, in other process type manufacturing and so on.

So, depending on the manufacturing situations, we would have all these jobs requiring a set of machines or facilities not only that, they require these facilities in the same order. So, whether it is job 1 or job 2 or job 8, it will first have to go to M 1 get something done from then it goes to M 2 get something done, then it goes to M 3 and so on. So, in a flow shop it is customary to say that the machines or facilities are already arranged in the sequence, in which the processing is required.

So, whatever is the first machine that all the jobs visit is called M 1, whatever is the second machine that all the jobs visit is called M 2 and all the jobs will therefore, visit M 1, first then M 2 then M 3 and then the last machine and then they go out. Now such a system is called a flow shop, we could have sometimes, we could have parallel processes here and so on, then it becomes flow shop with parallel processes, it is also a called a flexible flow shop and so on. But, the basic flow shop problem assumes that, there is only processor in the sequence to represent 1 particular machine then we move to the last one, which is called the job shop.

(Refer Slide Time: 46:15)



So, this could be M 1 this could be M 2, this could be M 3, this could be M 4, now in a job shop each job has a certain route, which it will take for example, say job 1 would first come to M 1 from there, it may go to M 3 from there, it may go to M 4, it may then come to M 2 and then it may leave. You could have job 2 that starts with M 2 from there it may go to M 3, from there it may go to M 1 and from there it may leave you could have a job 3, which starts with M 1, from there it may go to M 4, it may go to M 2, it may go back to M 1 and it may leave.

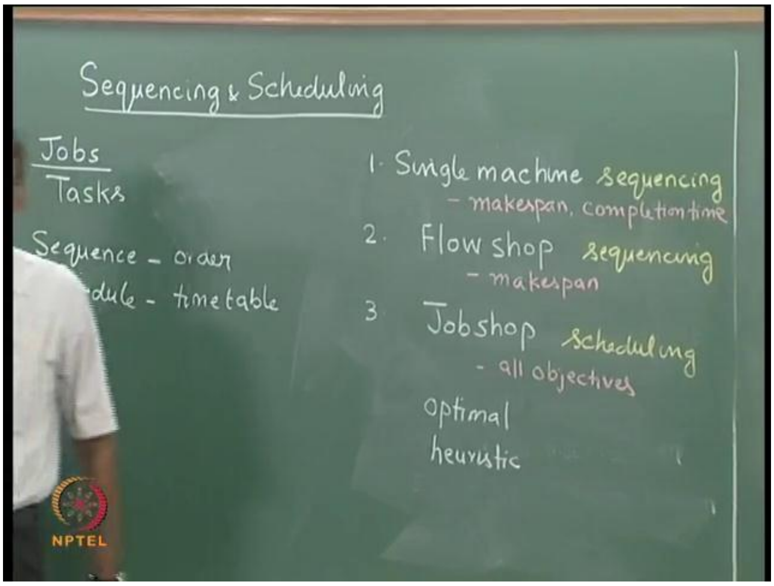
So, like this we could have a different route for each of the jobs and when, we actually define the routes, we did a couple of things, if you look carefully at say job J 2, J 2 starts with M 2 goes to M 3 goes to M 1 and goes out it does not visit M 4, it is not necessary for all the jobs to visit all the machines. Whereas, in a flow shop it is absolutely necessary for all the jobs to visit all the machines, the order is very different for each job here in the flow shop the order has to be the same.

If you look at J 3 very carefully, it comes into M 1 goes to M 4 goes to M 2 goes to M 1 again and goes out, it does not visit M 3, but it visits M 1 twice, now such a thing is called a revisit. So, in a job shop a revisit is also allowed, in a flow shop revisit is usually not allowed, so job shop is the most flexible thing is the is kind of an unconstrained kind of a system where, but then in spite of all of that the route for each job is very clearly defined and each job will follow, it is route.

And there is a process in time associated with each one of it, for example, J 3 on M 1 will have a processing time J 3 on M 4 will have a given processing time and so on. So, we have now kind of described, the three basic shops that, we are going to see as mentioned, there are other shops, there are things like an open shop where, a job can be done on any machine, that is available here. And then it takes a certain time to do then I also mentioned about flexible shops, where you have parallel machines in a flow shop and so on.

So, the and there are many other types of shops, which people have studied over, so many years, but as far as this lecture series is concerned, we would be looking at single machine, we would be looking at flow shop, and we would be looking at job shop. We will also be seeing certain models in this lecture series.

(Refer Slide Time: 50:18)



So, we would look at single machine sequencing and we would look at models that deal with makespan completion time, in flow shop, we will see models for minimizing makespan and then in the job shop, we will see models that minimize almost all objectives. We would also be looking at algorithms, which can give optimal solutions, which means the best value of the objective function, as well as we will be doing heuristic solutions or solutions, which are essentially non-optimal, but close to optimal, we will see some of these algorithms in detail, in the next lecture.