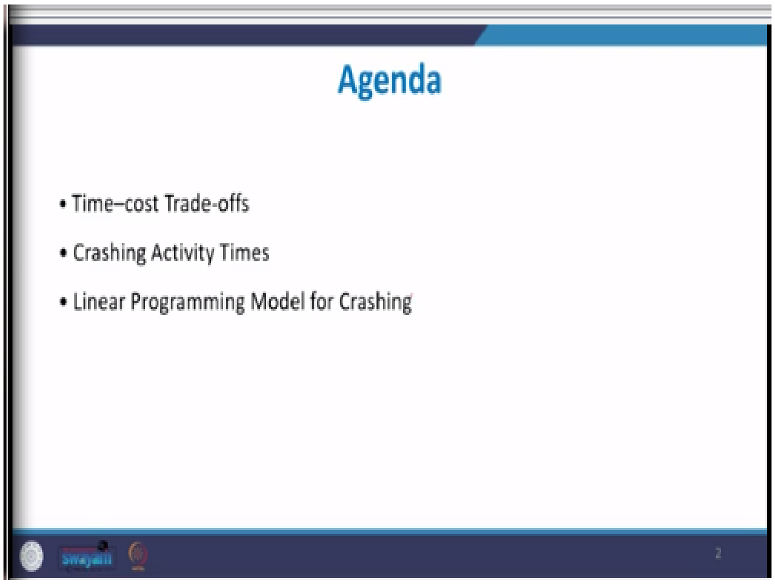
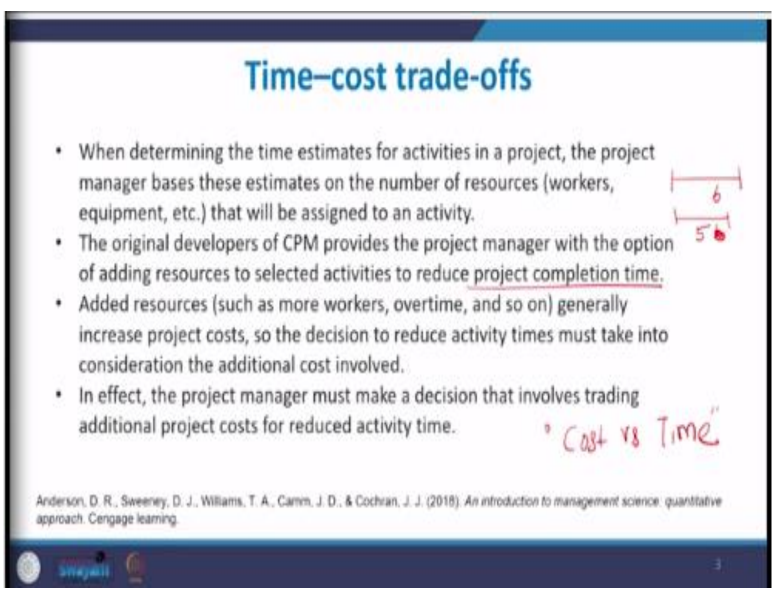
Decision Making With Spreadsheet
Prof. Ramesh Anbanandam
Department of Management Studies
Indian Institute of Technology-Roorkee

Lecture-32
Project Scheduling: PERT/CPM-IV

Dear students, in the previous lecture, I discussed scheduling a project where that has uncertain activity durations. In this lecture, I am going to discuss the time-cost trade-off.



So, the agenda for this lecture is time-cost trade-off and crashing activity times, and a very important thing is how to use linear programming for crashing a project network.

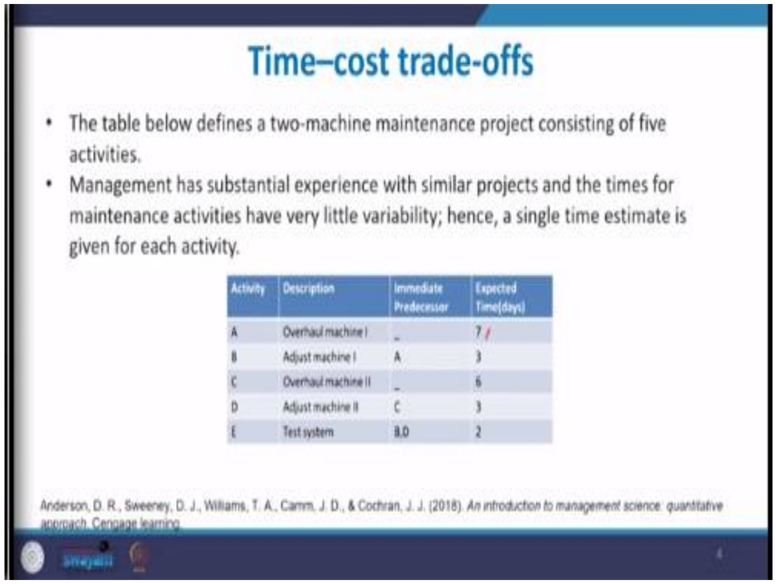


What is this time-cost trade-off? When determining the time estimates for an activity in a project, the project manager bases these estimates on the number of resources that will be assigned to an activity. What are the resources? Resources like workers, equipment, availability of supplies, and so on. The original developer of CPM provides the project manager with the option of adding resources to selected activities to reduce the project completion time.

Suppose I have a project that is taking this many days, for example, 6. So, what will happen when you add more resources? So, what if there is a possibility I can reduce this project activity by, say, from 6 to 5? So, what can we do when you supply more resources? The activity duration can be minimized. The added resources generally increase the project cost, so when you add more resources the project cost increases.

So, the decision to reduce activity time must consider the additional cost involved. In effect, the project manager must make a decision that involves trading additional project costs versus reduced activity times. So, what will happen? We need to have a trade-off between cost and time, what will happen? When you compress the time, the cost will increase; instead of finishing this activity in 6 days by supplying more resources, if you are planning to reduce it within 5 days, the time will decrease.

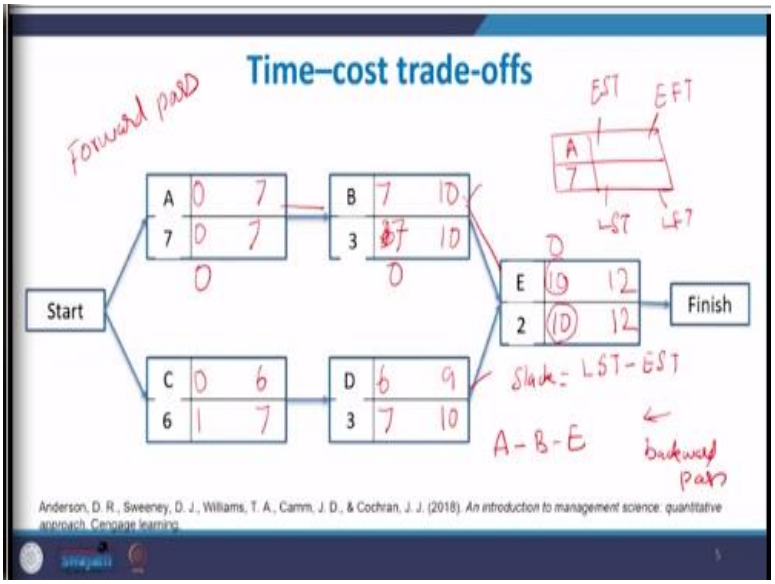
But the cost of finishing this activity within 5 days will increase. So, the project managers need to understand the trade-off between cost and time. So, in this lecture, we are going to discuss this cost versus time. So, what are we going to do in this lecture? What is the optimal project completion time so that the cost can be minimized? That is the objective of this lecture.



Activity	Description	Immediate Predecessor	Expected Time(days)
A	Overhaul machine I	–	7
B	Adjust machine I	A	3
C	Overhaul machine II	–	6
D	Adjust machine II	C	3
E	Test system	B,D	2

Now I am going to explain this time-cost tradeoff with the help of a problem. The reference for this problem is the book Anderson et al. What does this problem say? There is a 2-machine maintenance project consisting of 5 activities. They want to do maintenance for 2 machines for that there are 5 activities. Management has substantial experience with similar projects and the times for maintenance activities have very little variability. Hence, a single time estimate is given for each activity.

So, for activities A, B, C, D, and E, there is a predecessor for each activity given and the expected time. When you look at this expected time, for example, for completing activity A, it will take only 7 days since the maintenance activity is a repetitive in nature. Based on their experience, they are able to provide a single time estimate, which is why it is exactly 7 days. What is the first task? The first task is we have to draw the project network, and then we have to find out how much time it is going to take to complete all 5 activities.



So, what I have done? I have drawn this network; now I am going to find out how much time it is going to take. So, I am going to use the same notation that we have followed in the previous class to see, for example here, the activity name and activity duration. Here it will be the earliest starting time, here it will be the earliest to finishing time, here is the latest starting time, here is the latest finishing time.

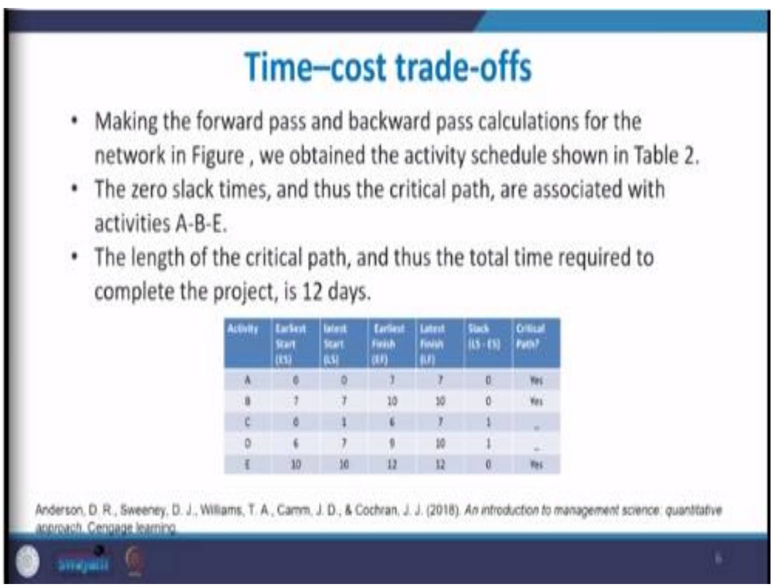
So, first, I am going to use a forward pass; what is this forward pass? So, for activity A, there is no precedence, so it can start on the 0th day, which will be the earliest starting time. What will be the earliest finishing time? The duration is 7, so 0 + 7 it is 7. Similarly, for activity C, there is a known predecessor, so it can start on the 0th day and it will finish 0 + 6 = 6th day. Activity B will start on the 7th day. You see that the time I am writing is a cumulative scale.

So, why I have written 7th day because the earliest finishing time of the previous activity is 7 days; activity B can be started only after the 7th day, so 7 + 3 = 10. So, here, activity C will be completed only on the 6th day, so activity D can be started only after the 6th day, so the duration of activity 3 is 9, so it is 9. Now we have to find out the earliest starting time for activity E. What happened?

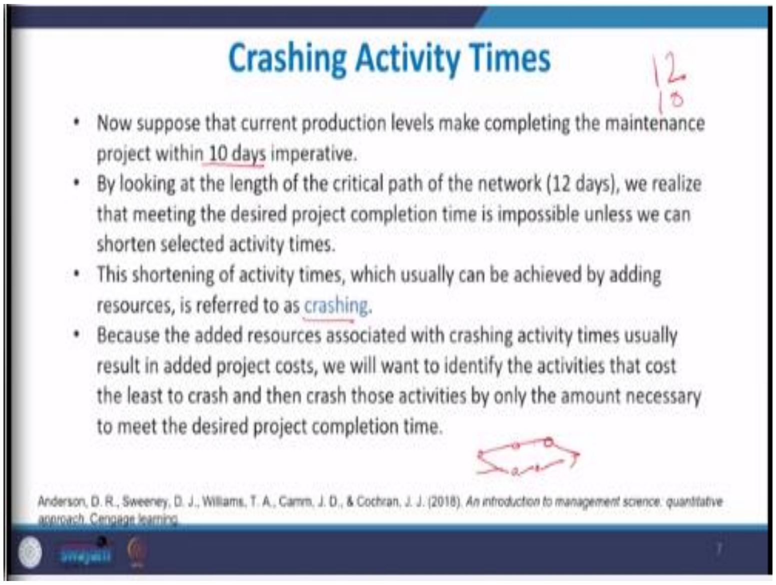
You see there is a 10, there is a 9. So, as per the forward pass algorithm, we have to consider the largest value of the earliest starting time; what is the largest value? 10, the 10 is the largest value between 10 and 9, so 10 + 2 is 12, so it will take 12 days to complete the whole project. Now, I am going to discuss this backward pass. So, 12 is your latest finishing time; if 12 is the latest finishing time, what will be the latest starting time?

So, 12 - 2, 10. Now for activity D, the latest finishing time is 10 because the latest starting time of activity E is 10, which will be equal to the latest finishing time of activity D, which is the preceding activity, so 10 - 3 is 7; now this is 7, and 7 - 6 is 1. For activity B, the latest finishing time is 10, so 10 - 7 is 3, 10 – 3 = 7, so it is 7, and 7 - 7 is 0. How to find out the critical path, the path which is taking the longest duration otherwise if you connect all critical activities.

What is the critical activities? Activities that have 0 float or 0 slack. So, what is the slack? We have discussed in the previous class that slack is the difference between the latest starting time and - the earliest starting time. That means the difference between this value and this value is 0, so here, 7 - 0, 0, 0 - 0. So, there are 3 critical activities activity A, B, and E, so this path, which one? This path is A-B-B-E, so the path A-B-E is your critical path. What is the project duration? 12 days.



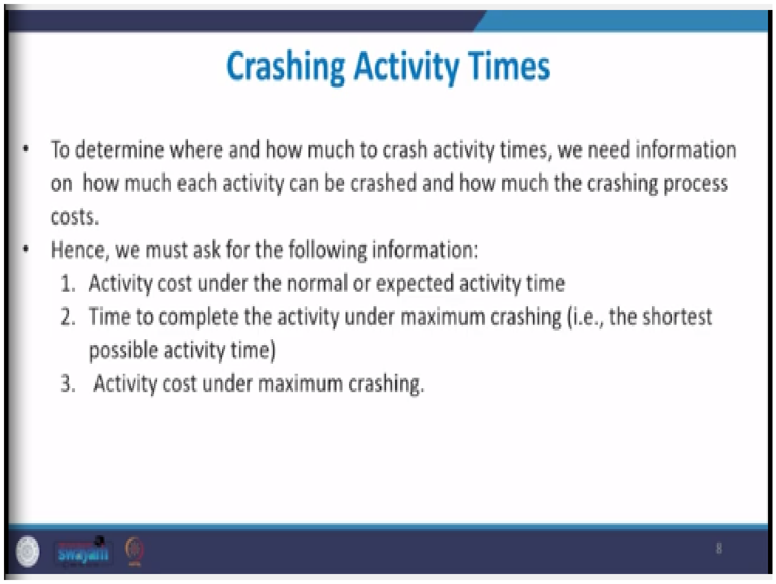
Making the forward pass and backward pass calculation for the network in the figure which I have shown in the previous slide we obtained the activity schedule shown in table 2 like this. We found the earliest starting time for each activity and the latest starting time, earliest finishing time, and latest finishing time; we got slack; wherever slack is 0, that path is a critical path. Then the 0 slack times and thus the critical path associated with activities is A-B-E. So, the length of the critical path and, thus, the total time required to complete the project is 12 days, which I have shown you in the previous slide.



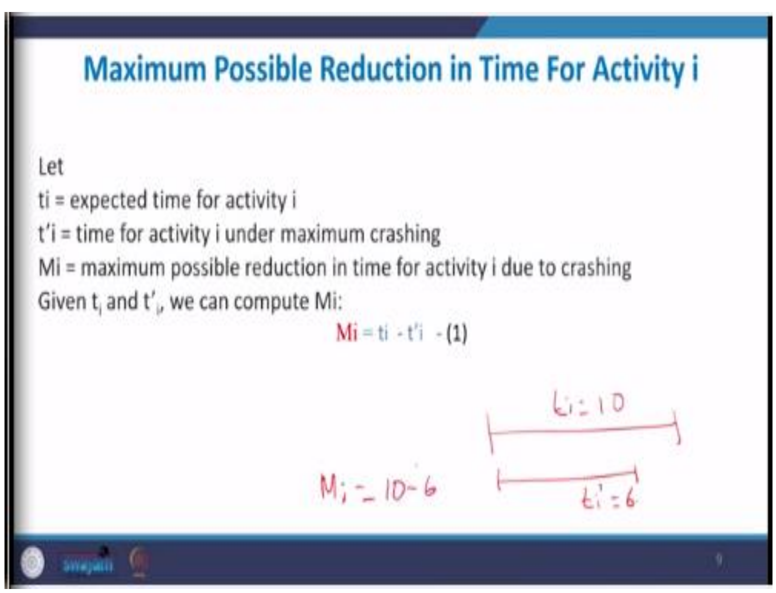
Now, we are going to discuss the concept of crashing. Suppose the current production level makes completing the maintenance project within 10 days we have to finish the maintenance activities. It will take 12 days, but the management has decided to finish this maintenance activity within 10 days. So, when you reduce 12 days to 10 days, time will be decreased, but there will be an increase in cost; that is what we are going to do.

By looking at the length of the critical path of the network, 12 days, we realize that meeting the desired project completion time is impossible unless we can shorten selected activity times. The only possibility to reduce the project duration to 10 days is for some of the activities we have to compress the time. So, this shortening of activity times, which usually can be achieved by adding additional resources, is referred to as crashing.

Because the added resources associated with the crashing activity times usually result in added project costs. We will want to identify the activities that cost the least to crash and then crash those activities by only the amount necessary to meet the desired project completion time. So, there will be different activities, so what we have to do? We have to crash, or we have to reduce the time of activity, which takes the minimum cost; that is the logic here. There will be a limitation, so for each activity, only this much duration can be crashed; that constraint also needs to be considered.

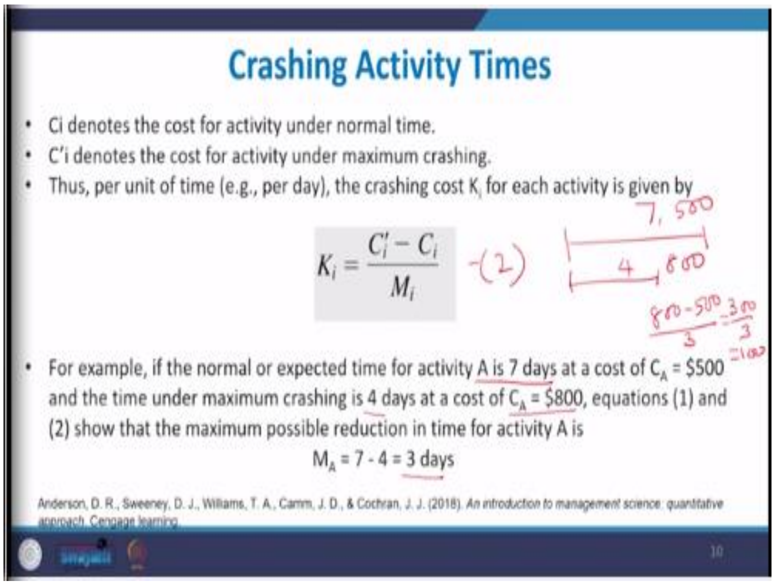


So, to determine where and how much to crash activity times, we need information on how much activity can be crashed and how much crashing processes cost. So, we need to have the time and the cost. Hence, we must ask for the following information, first information is activity cost under normal or expected activity time. Second, the time to complete the activity under maximum crashing that is the shortest possible activity time. And activity cost under maximum crashing, so this information is required for completing the crashing activity.



Now I am going to explain the maximum possible reduction in time for activity i let ti = expected time for an activity. For example, I have an activity like this, say it is taking 10 days, so the ti is nothing but this 10 expected time for activity i, ti dash time for an activity i under maximum crashing, for example, I can crash up to say 6 days. So, the maximum possible reduction in time for an activity i due to crashing is so what I can write M i is nothing but this 10 - 6, so ti - ti dash.

Mi = ti - t'i - (1)



Let us Ci denote the cost of activity under normal time C i dash denotes the cost of activity under maximum crashing. Thus, per unit time that is the crashing cost Ki for each activity is ki = Ci dash - Ci upon Mi.

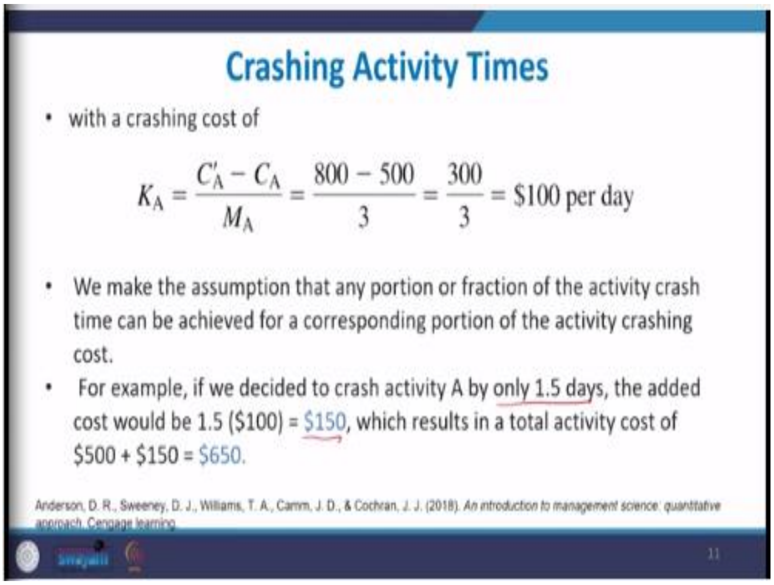
$$K_i = \frac{C'_i - C_i}{M_i}$$

For example, if the normal or expected time for activity A is 7 days, for example, say this is 7 days, and the cost of C_A is 500 dollars, and the time under maximum crashing is 4 days, and the cost of crashing it to 4 days is 800 dollars.

For example, if the normal or expected time for activity A is 7 days at a cost of CA = $500 and the time under maximum crashing is 4 days at a cost of CA = $800, equations (1) and (2) show that the maximum possible reduction in time for activity A is

MA = 7 - 4 = 3 days

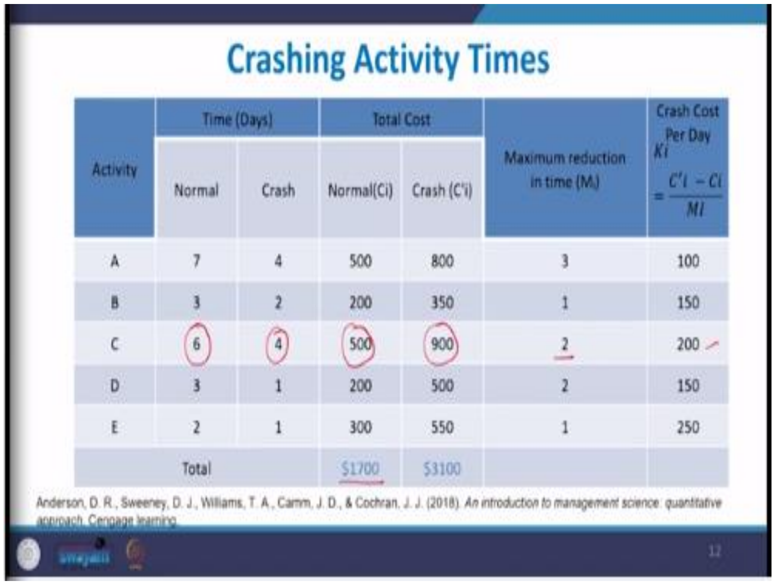
So, from the equation 1 and 2, so this is equation 2, the maximum possible reduction in time for activity A is 7 - 3, 3 days. So, there is an activity like this, so normally we can finish in 7 days, and the cost for this is 500 dollars; if you compress this into 4 days, the cost is 800 dollars. So, first, I am finding out how much time we can crash it. That is 7 - 4 is 3. And I am going to find out per day crashing time.



So, with the crashing cost of K_A, we got the K_A representing per day crashing cost, so the difference in cost is 800 - 500, you see this, this 800, difference in cost is 800 - 500, the difference in time is 3, 7 – 4 = 3. So, when you simplify this, so 300 upon three this is 100, so this 100 represents per day crashing cost. So, these 100 dollars represent the crushing cost per day; we make the assumption that any portion or fraction of activity crash time can be achieved by the corresponding portion of activity crashing cost.

$$K_A = \frac{C'_A - C_A}{M_A} = \frac{800 - 500}{3} = \frac{300}{3} = \$100 \text{ per day}$$

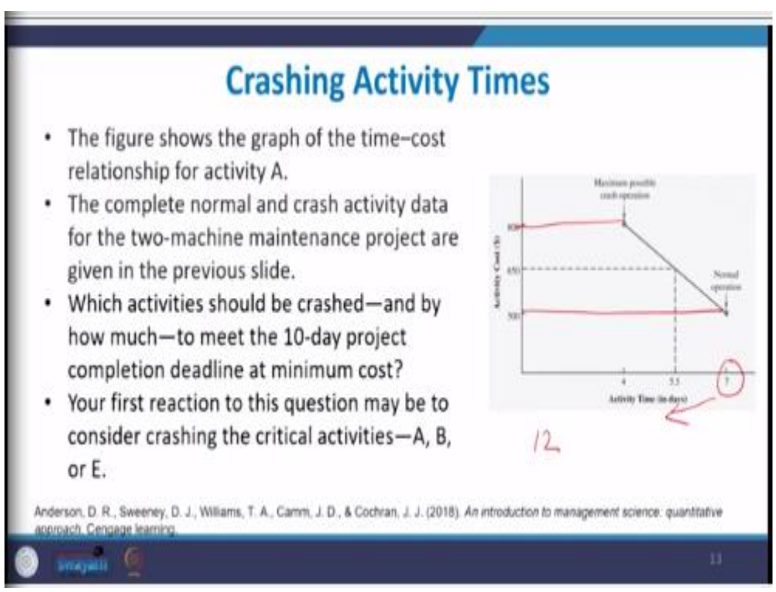
So, what is the meaning of that? If you are crashing that means if you are compressing the time that is achievable by adding the additional cost, that is the crashing cost. For example, if we decided to crash activity A by only 1.5 days, the added cost would be 1.5 multiplied by 100, 150 dollars only for crashing cost. Apart from this, there is a 500 dollar for normal cost, so the total cost would be 500 + 150 = 650 dollars.



Activity	Time (Days)		Total Cost		Maximum reduction in time (M_i)	Crash Cost Per Day $K_i = \frac{C'i - Ci}{Mi}$
	Normal	Crash	Normal(Ci)	Crash (C'i)		
A	7	4	500	800	3	100
B	3	2	200	350	1	150
C	6	4	500	900	2	200
D	3	1	200	500	2	150
E	2	1	300	550	1	250
	Total		$1700	$3100		

Now, in this table, for each activity, the normal time and crash time are given, and the normal cost and crashing cost are given. For example, Activity C can be normally completed within 6 days; the cost for finishing the 6 days is 500 dollars. If you compress it from 6 days to 4 days, the crashing cost is, say, 900 dollars. So, now, how many days can we crash? 6 – 4 = 2 days we can crash it. What is the per-day crashing cost?

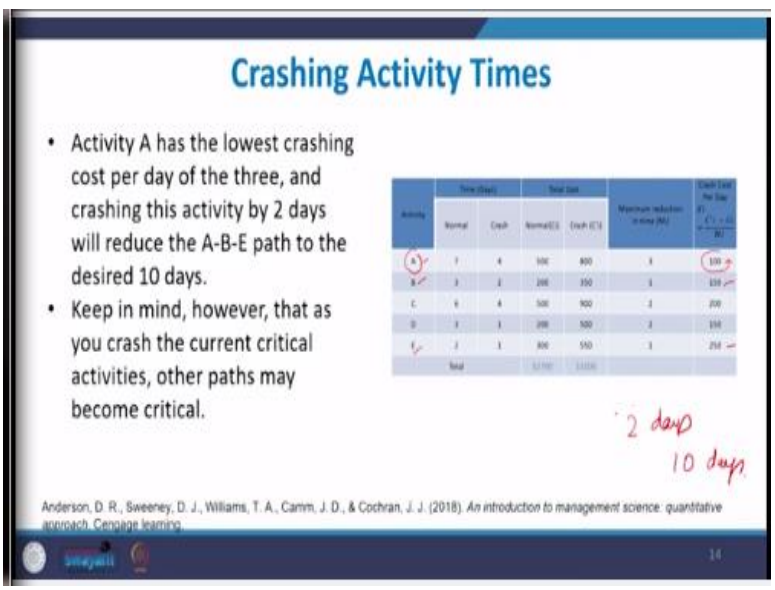
So, what is the difference in cost? 900 - 500 is 400, 400 divided by 2 is 200, so 200 dollars is a crashing cost per day. That means you see that if it is 6 days, if you are making it to 5 days, there will be 200 dollars, which will be extra, so 500 + 200 = 700. For each activity, a per-day crashing cost is found. Then apart from this, there is a total cost of 1700 dollars.



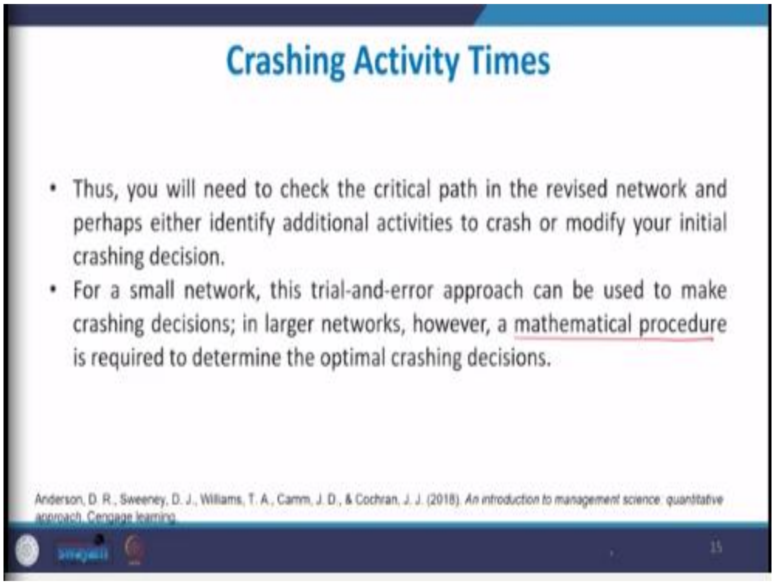
This figure shows the graph of the time-cost relationship for activity A. For activity A, the normal duration is 7 days, which is a normal operation; the cost for completing this activity for 7 days is 500 dollars. When you decrease this cost, when you compress, this does not cost the time; for example, in 4 days, the cost will be 800 dollars. So, what we understand is that when you compress the time, the cost increases.

The complete normal and crash activity data for the two machine maintenance projects are given in the previous slide I have shown you which activity should be crashed and how much to meet the 10-day project completion deadline at minimum cost because the management is expecting the maintenance activity to be done within 10 days, but actually, it is taking 12 days. So, what will be your first reaction to this question?

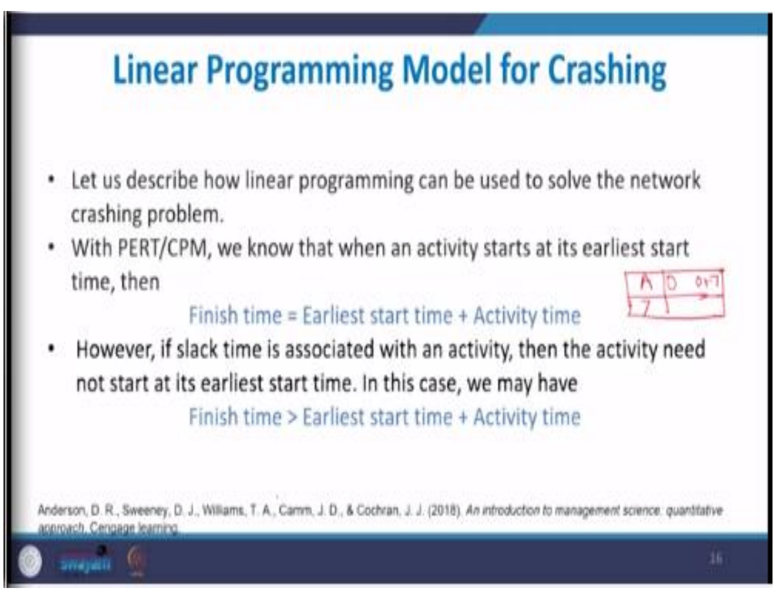
Maybe we should consider crashing critical activities. What are the critical activities? A, B, E because if you reduce only on critical activity, the project duration can be compressed; if you do the crashing for noncritical activity, that is not required because that will not affect your total project completion time.



Activity A has the lowest crashing cost per day of 3; you see, activity A is the lowest because the critical activities are A, B, and E, so the corresponding costs are 100, 150, and 250. The lowest crashing cost per day is 100 rupees, which is for activity A. So, what can we do? The activity A, we can maximum crash for 3 days, now what we can do we can crash it for 2 days so that the project can be completed within 10 days instead of 12 days. Keep in mind, however, that as you crash the current critical activities the other path may become critical. So, what will happen? When you crash this, another path that is noncritical may also become a critical path.

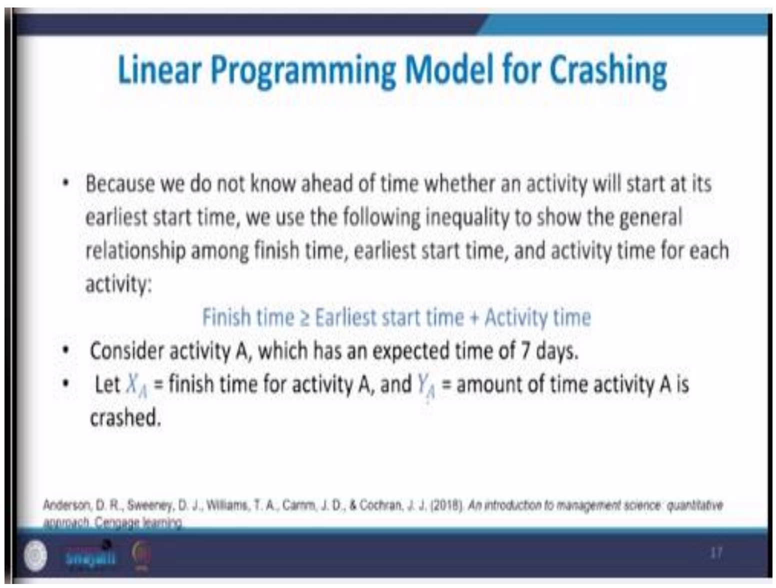


Thus, you will need to check the critical path in the revised network and perhaps either identify the additional activities to crash or modify your initial crashing decision. For a small network, this trial-and-error approach can be used to make crashing decisions. Still, in larger network, however, a mathematical procedure is required to determine the optimal crashing decisions. These mathematical procedures I am going to explain these mathematical procedures as nothing but your linear programming. So, what am I going to do now? With the help of linear programming, I am going to complete this crashing activity.

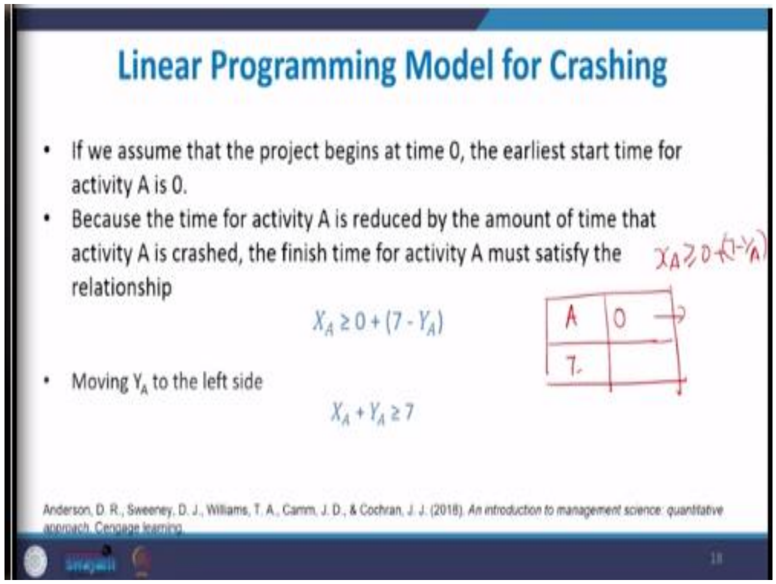


So, the method is called a linear programming model for crashing. Let us describe how linear programming can be used to solve the network crashing problem. With PERT and CPM, we know that when activity starts, it is the earliest starting time, so the finishing time will be the earliest starting time + activity time. You might remember like this, can you recollect? This is A, for example. What is the duration of this? I think 7. So, what will happen?

This finishing time, so here, so the finishing time will be the earliest starting time 0, so 0 + 7, so that will be your earliest finishing time. However, if slack time is associated with an activity, what is the slack time? Sometimes, instead of starting on the 0th day, we can start on the 1st day or 2nd day if there is a slack available. If the slack time is associated with activity, then the activity need not start at it is the earliest start time. So, what can we do? In this case, we may have the finish time be greater than the earliest starting time + activity time.



Because we do not know ahead of time whether an activity will start at its earliest start time, we use the following inequality to show the general relationship between finish time, earliest start time, and activity time for each activity. What is the general relationship? The finishing time is greater than or equal to the earliest starting time plus the activity duration time. Consider activity A, which has an expected time of 7 days. Let X_A = finishing time for activity A, then Y_A is the amount of time activity A is crashed.

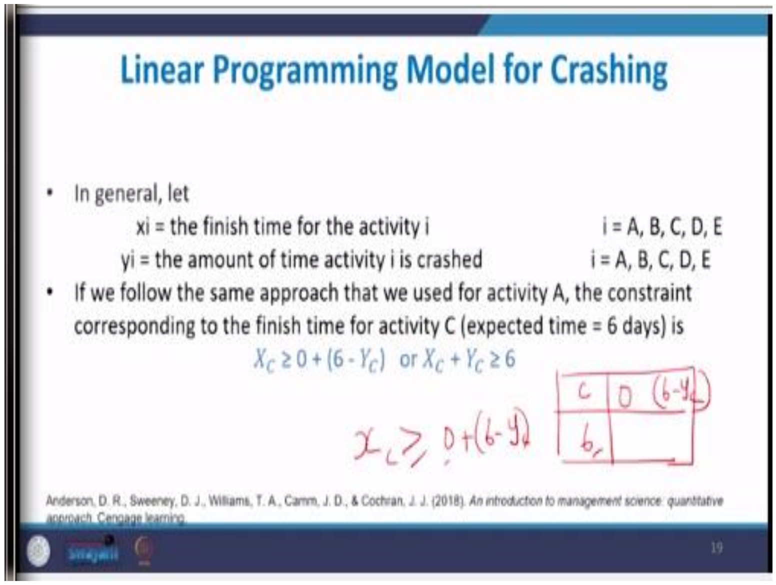


If we assume that a project begins at 0, the earliest start time for activity A is 0. So, I am writing like this, so the earliest starting time is 0. Because the time for activity A is reduced by the amount of time that activity A crashes, the finish time for activity A must satisfy the following relationship: what is that?

$$X_A \geq 0 + (7 - Y_A)$$

The Y_A is how much time we are going to crash it. So, X_A this point, this point is nothing but earliest finishing time X_A should be earliest starting time 0 + 7 that is duration minus how much time we are going to crash it. So, when you bring this Y_A on the left hand side will become X_A + Y_A is greater than 7.

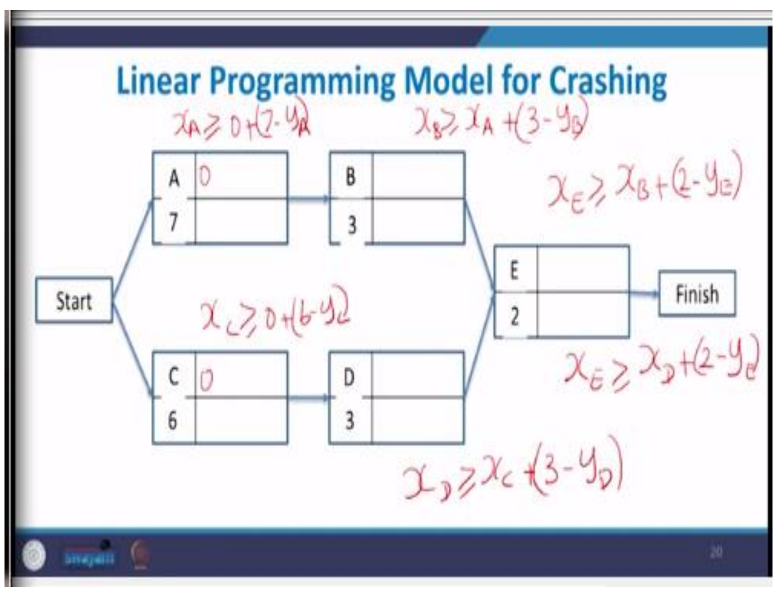
$$X_A + Y_A \geq 7$$



If we follow the same approach that we used for activity A, the constraint corresponding to the finish time for activity C, what is activity C? Activity C is like this activity C, the duration is 6 days, the earliest starting time is 0, then

$$X_C \geq 0 + (6 - Y_C) \quad \text{or} \quad X_C + Y_C \geq 6$$

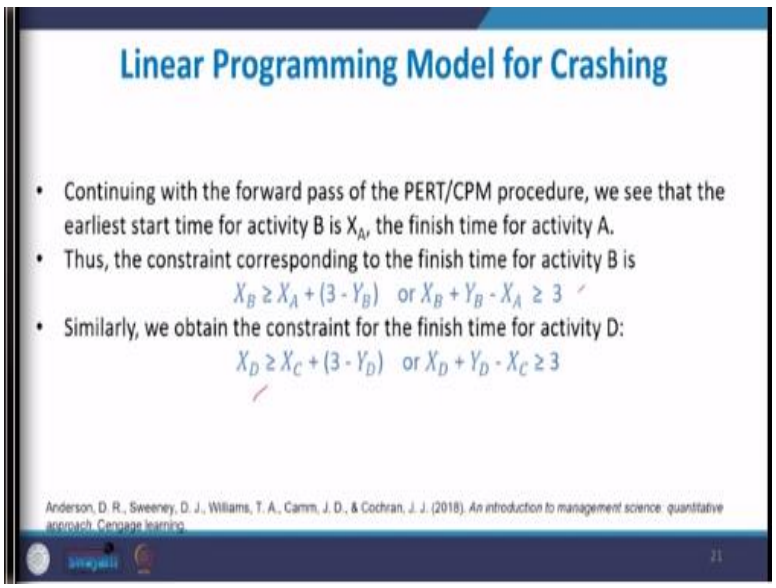
.



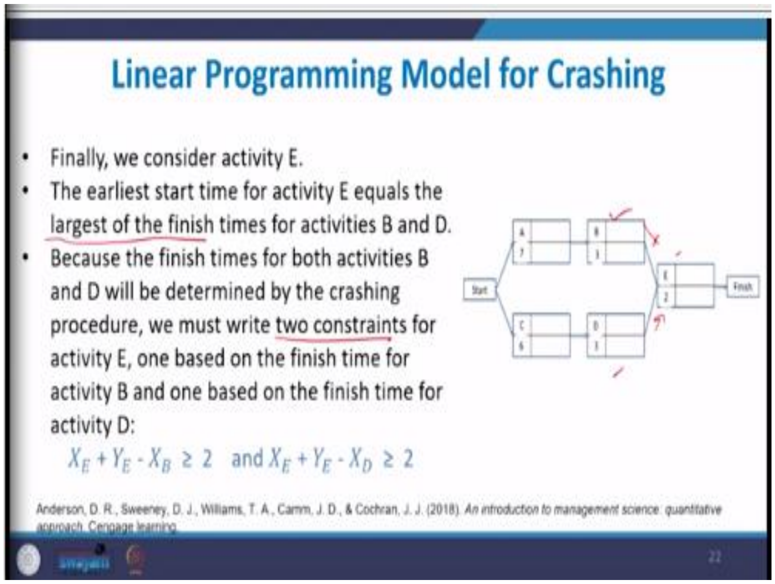
So, what am I going to do? I am going to write about the equations for all the activities. For example, here it is 0, so how can I write it? So, $x_A >= 0 + (7 - y_A)$, and for activity B, the x_B's earliest finishing time will be greater than. Now you see here that the earliest starting time will be x_A; for activity B, the earliest starting time will be x_A. So, $x_A + (3 - y_B)$. Now we come to activity C, so here it is 0, so $x_C >= (0 + 6 - y_C)$

Now, for activity D, so x_D, I am writing here that x_D is >= $x_C + (3 – y_D)$; what is the y_D, and how much time this activity D can be crashed? Now we have to write for activity E, so generally, what are we used to do? The largest value of the latest finish time will be the earliest starting time. So, we are going to write it for both activities because we do not know which is the largest one. So, there will be 2 equations for this. What is the 2 equations?

So, x E will be greater than or equal to the earliest starting time. It may be $x_B + (2 – y_E)$, which is 1 possibility. Otherwise the x E is greater than or equal to the earliest start starting time may be x_D also, $x_D + (2 - y_E)$. So, now I have written for all the activities the equation which connects the earliest finishing time and how many days the crashing can be done for each activity.



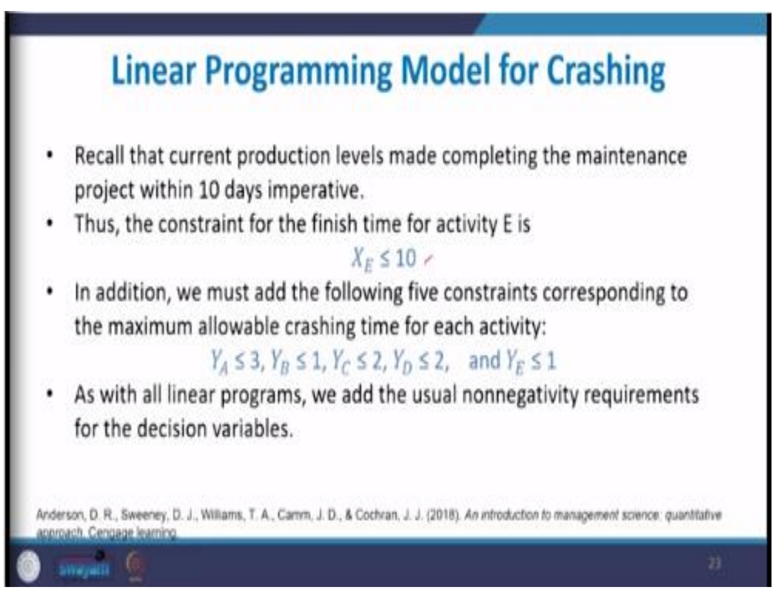
Continuing with the forward pass of the PERT/CPM procedure, we see that the earliest start time for activity B is X_A is the finishing time for activity A. Thus, the constraint corresponding to the finish time for activity B is, so X_B is greater than or equal to X_A. What is an X_A? It is the earliest starting time of activity $X_A + (3 - Y_B)$ what is the 3? It is the duration of activity B; what is the Y_B? How many days can the crashing be done, so this is activity B; similarly, I have written for activity D.



Finally, we consider activity E. So, here, activity E, so what is happening here? There are 2 preceding activities. So, the earliest start time for activity E is equal to the largest finishing time of activities B and D, this B and D. Because the finish time for both activities B and D will be determined by the crashing procedure, we must write 2 constraints for activity E. One is based on the finishing time for Activity B, and another one is based on the finish time for Activity D, as I have explained previously. So, when you simplify this will be

$$X_E + Y_E - X_B \geq 2$$

$$X_E + Y_E - X_D \geq 2$$

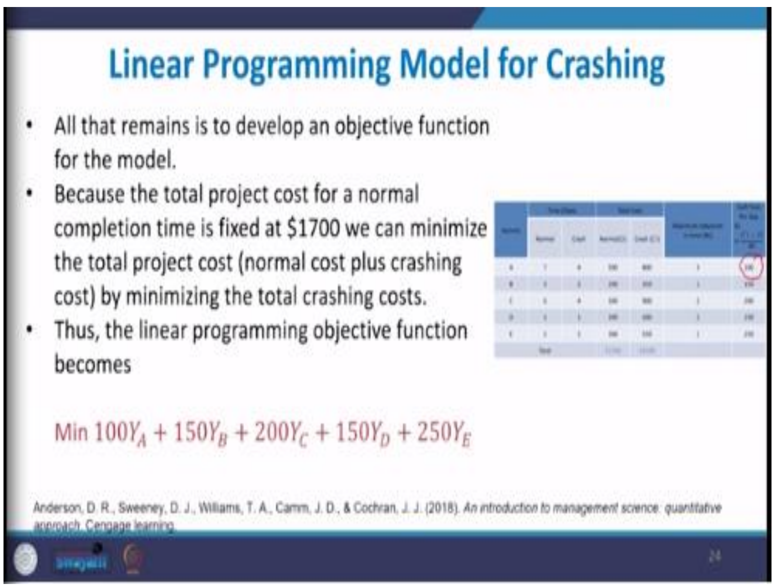


Recalled that the current production level made completing the maintenance project within 10 days. Thus, the constraint for finish time for activity E, X_E should be less than or equal to 10; this constraint needs to be added.

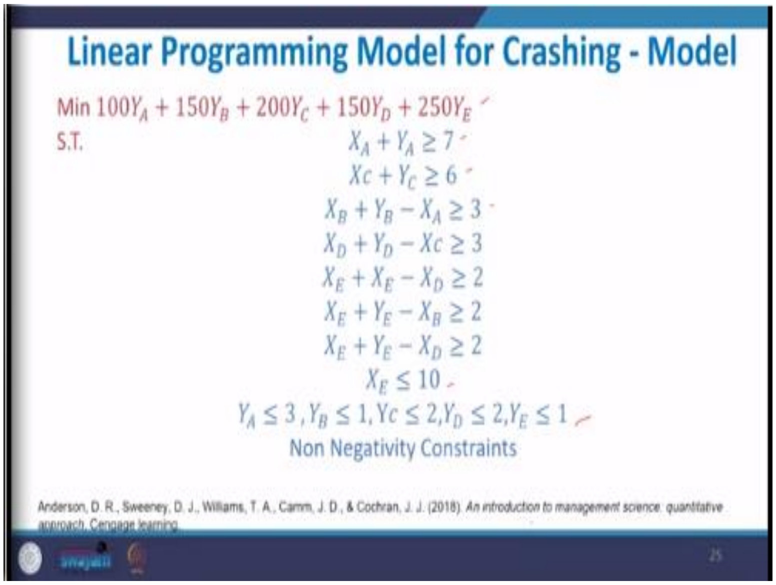
$$X_E \leq 10$$

In addition, we must add the following 5 constraints corresponding to the maximum allowable crashing time for each activity. For example, activity A from the table from the previous slides we have seen a maximum of 3 days can be crashed.

$$Y_A \leq 3,\ Y_B \leq 1,\ Y_C \leq 2,\ Y_D \leq 2, \quad \text{and } Y_E \leq 1$$



Now we have to formulate the objective function. Because the total project cost for a normal completion time is fixed, how much? It is 1700 dollar. We can minimize the total project cost, which is normal cost + crashing cost, by minimizing the total crashing cost. So, the linear programming objective function becomes 100Y A; how did we get this 100? You see this: if you reduce this duration for activity A by 1 day, the cost will be 100, so how many days do we have to reduce Y_A and Y_B, Y_C, Y_D, and Y_E? So, the objective function is minimizing the total crashing cost.

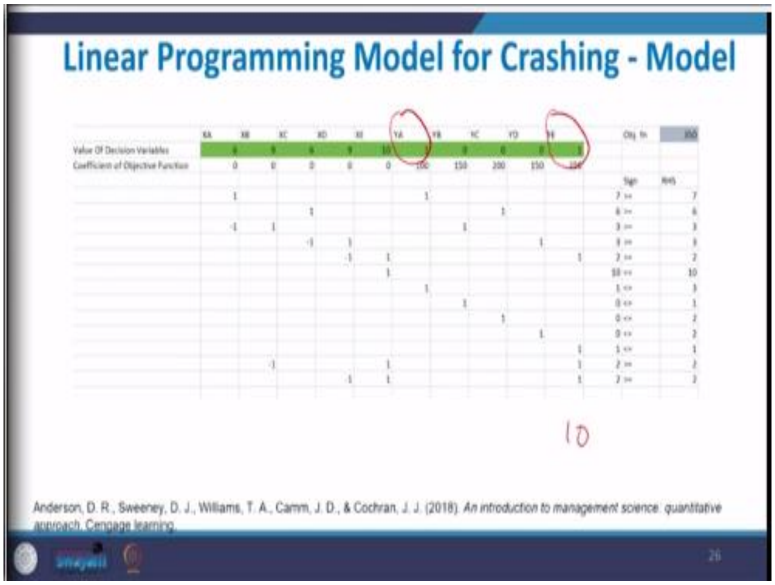


So, now I have brought the complete linear programming models. So, here the first one is objective function and constraints for all activities. And this constraint X_E is less than or equal to 10, it is the constraint provided by the management. The last Y_A and Y_B, Y_C, Y_D, and Y_E is the maximum allowable crashing times for each activity. Now, I am going to solve this problem with the help of a solver; then I am going to interpret the result.

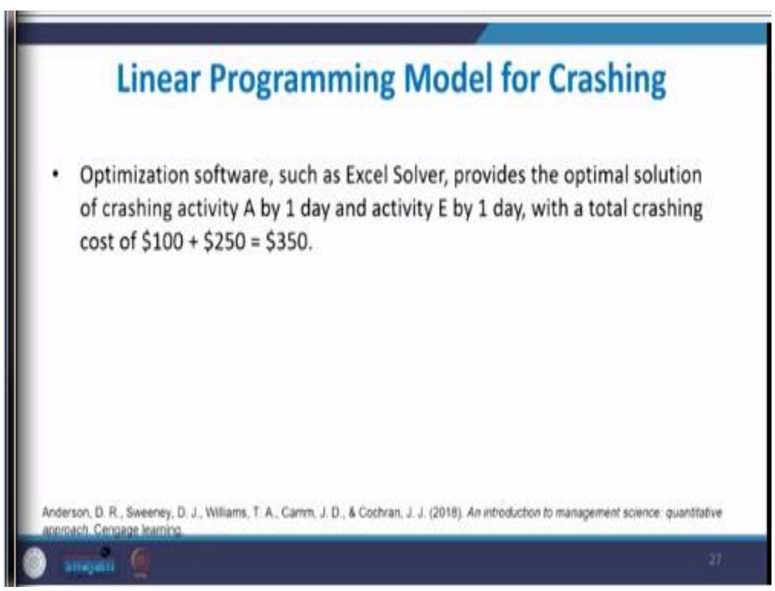
Now, this is the Excel model for this crashing problem; remember that X_A, X_B, X_C, and X_D, these variables are there in our constraint. So, any variables which are appearing in our constraint that need to be reflected on our objective function also. So, we have to include in our objective function X_A, X_B, X_C, and X_D where the coefficient of objective function is (0, 0).

And Y_A and Y_B, Y_C, Y_D, and Y_E variables are already there, and corresponding coefficient also given. So, if I go to N4, that is an objective function. What is the objective function? As usual, the sum product is multiplied by the value of decision variables and the coefficient of the objective function. Then I have included all the constraints, so now I am going to solve it, data solver, see that this problem is a minimization problem, the objective function is N4, and the changing cell is B5 to K5, and all the constraints I have included.

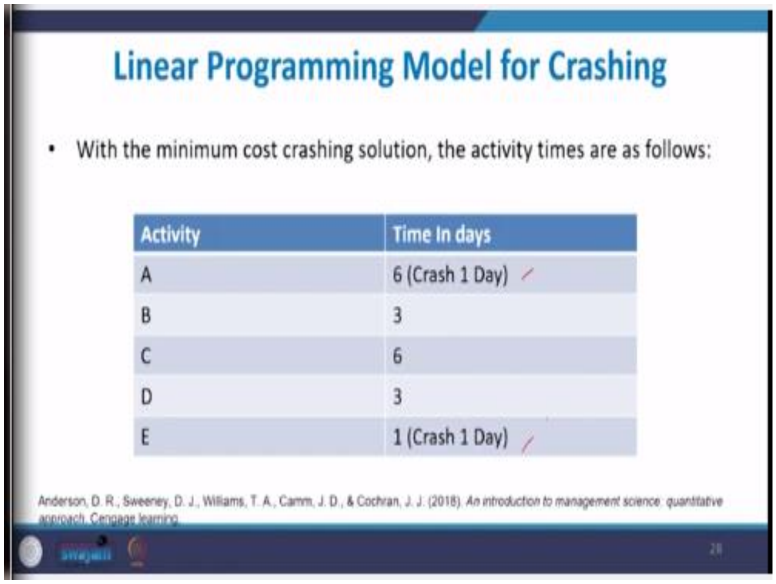
This is a linear model, so I am going to solve by linear programming models. So, when I solve it, now you see the total crashing cost is 350 rupees and Y_A, what is the $Y_A = 1$? The activity A can be reduced by 1 day and you see Y_E; activity E can be reduced by another 1 day. So, it just means by reducing 2 days, it will become 10 days, now I will go to my presentation, and I will interpret the result.



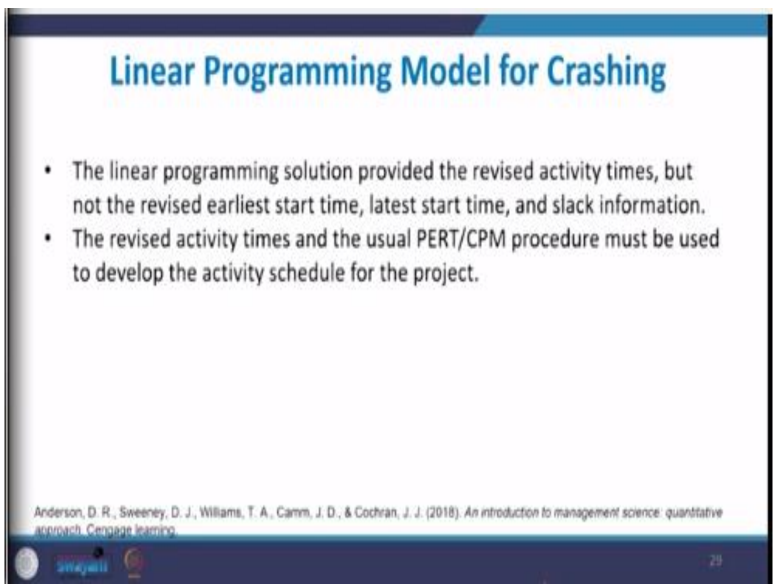
Now this is my final answer from the excel. So, here, the value of $Y_A = 1$, that is, activity A can be reduced by 1 day, Y is 1, and activity E can be reduced by another 1 day, so that it will become a total duration is 10 days.



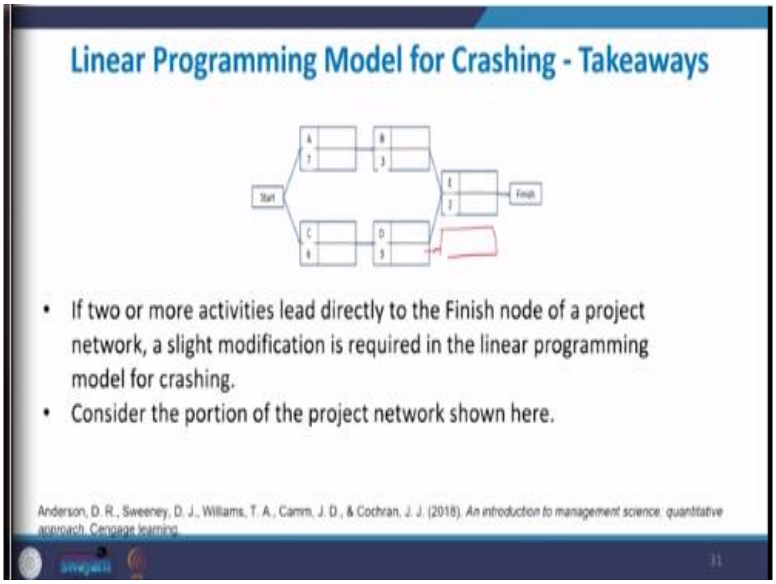
So, optimization software, such as excel solver provides the optimal solution for crashing activity A by 1 day and activity E by 1 day with the total crashing cost of 350, just now I have shown this answer.



So, with the minimum cost crashing solution, the activity times are like this. So, A can be crashed by 1 day because initially it was 7 days now it is 6 days, so activity E can be crashed by 1 day.

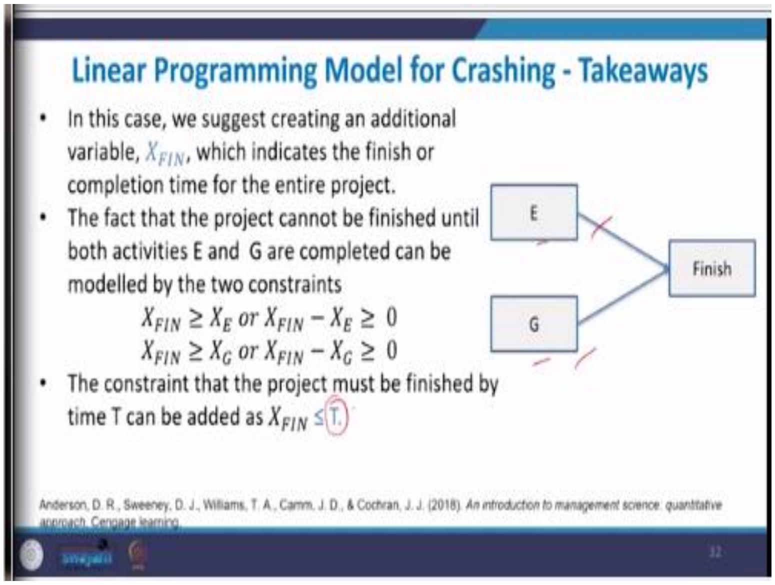


The linear programming solution provided the revised activity times but not the revised earliest start time, latest start time, and slack information. So, what will happen? When we compress our time the whole schedule also will change, so the revised activity times and the usual PERT/CPM procedure must be used to develop activity schedule for the projects. So, what I am saying here is that if you revise when you compress the duration again, the whole schedule needs to be found.



Linear programming model for crashing, some takeaways. Note that the 2-machine maintenance project network for the crashing illustration has only one activity, activity E, leading directly to the finish node. You see that there is only one activity that directly comes to the finishing node. Thus, thus we can directly write the linear programming constraint requiring the project completion in 10 days or less, which could be written as X_E <=10.

Sometimes, what will happen? There will be two activities that will end for finishing. For example, if 2 or more activities lead directly to the finish node of a project network, a slight modification is required; what will happen? There may be one more network here and one more activity here that will end at this finish node. So, what if there is more than one node, and we need a slight modification in our LP model? Now consider the portion of the project network shown here.



There are activity E and activity G that is directly coming to the end node, which is a finish node. So, in this case, we suggest creating an additional variable X_{FIN}, which indicates the finish or completion time for the entire project. The fact that the project cannot be finished until both activities E and G are completed can be modeled by the 2 constraints. So, what can we do that? So, we can write a constraint like this X_{FIN} >= X_E for this activity.

So, for the activity, if the G X_{FIN} is greater than or equal to X G, this X_{FIN} will be provided by the management. So, the constraint that the project must be finished by time T can be added as X_{FIN} <= T. So, whenever there are two nodes that are directly ending to the finishing node, these three equations can be added to take care. Dear students, in this lecture, I explained the time-cost trade-off. What is this time-cost trade-off?

Whenever you compress the time, the cost will increase, so we need to have the optimal reduction in the number of days so that the cost will be minimized. Then, I explained the crashing activity times. The last one that is very important is that I have explained how to use a linear programming model for crashing with the help of Excel. Thank you very much, students.