Linear Programming and its Extensions

Prof. Prabha Sharma
Department of Mathematics and Statistics
Indian Institute of Technology, Kanpur

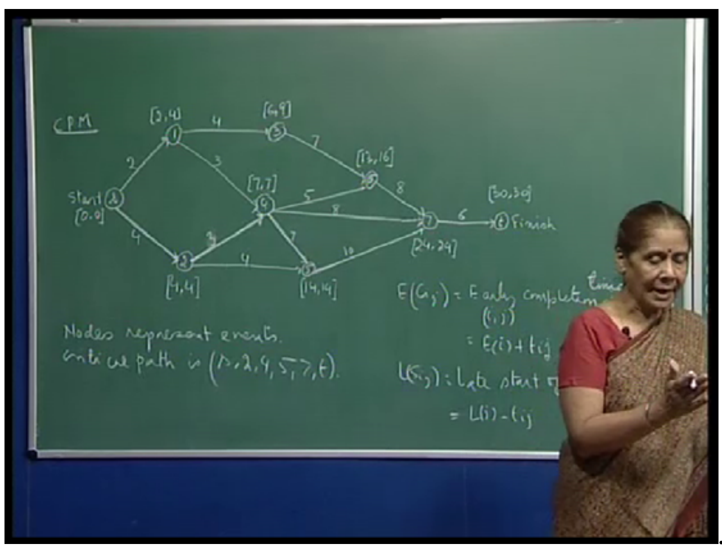
Module No. # 01

Lecture No. # 37

Critical Path Method (CPM)

So, in the last lecture, I had shown you how to calculate given the c p m network and then gave the durations of each activity which we denote by t i j. Then, I had shown you how to calculate the early starts of nodes and late starts right.

(Refer Slide Time: 00:17)



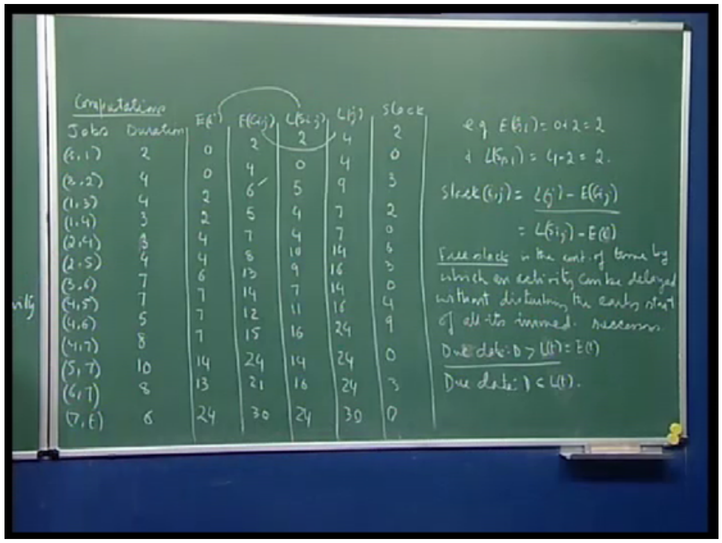
So, you remember that calculation and then, I had also shown you that once you come to the finish point, then of course, we take the late finish also here, as same as early start. So, for this node it will be 30 30 and so, for example, and then you can trace back the critical path; I showed you that also. So, here for example, for node seven, both the early start and late starts are same because, this is a critical activity and then, this way we go on. So, therefore, I told you that to compute the critical path, you just have to show, see how this calculation came about or this calculation came about. So, for for which ever activity, the the early start or late start got calculated, that will constitute the one of the critical activities. So, then your critical path actually came out to be.

So, this was your critical path right and then 4 5 this is 2 7 because, you see each of them has the same early start and late start are the same. So, this was your critical path, we I showed this calculation. Of course, network I have chosen is little different from what we had last time. Then I also said that you see the nodes are events. For example, if you look at node four, then it tells you that this is the completion of activities 1 4 and 2 4 right and of course, 1 4 and 2 4 cannot begin, like for example, 1 4 cannot begin till 1 2 is s one is complete and similarly 2 4 cannot begin till s 2 has been completed. So, in a sense, node 4 indicates the completion of all these four activities. So, node node nodes represent events.

And. So, Then we computed the early start and late starts. For example, this tells you that you must not be here later than seven because, it is a later start because, if if you delay the point this number here, then the other activities, the following the successive activities will get delayed. So, these are events but and. So, the critical path we showed you, but the manager also needs to know the the early start, late start of activities because, he has to concentrate, he or she has to concentrate on the completion all the activities of the projects. So, we need some more tools for the manager to be able to manage the project properly and so, we will define this. For example, e c i j is the early completion time, sorry completion time of activity i j right and this would be, e i; that means, early start of node i because, the activity i j will begin at node i.

So, the early start of node of i plus the completion time of the activity i j. So, this will give you the early completion time of activity i j and similarly, you want to compute the late start of activity i j. So, that will be late start of activity i j, which will be l j So, because, j denotes the completion of activity i j. So, late start late completion of or late reaching of node at node j minus t i j. So, that will give you the late start and you can immediately see that. Now, you can have this concept of slack for each activity because, the manager needs to know which of the activities, which of slack and so, maybe you can, the manager need not concentrate so much on the activities which of slack, but has to concentrate on activities which have no slack or comparatively less slack, than the others.

(Refer Slide Time: 04:16)



So, this is the kind of a tool that the manager needs and so, we are here, we will define. So, if you want to just get a feeling for these numbers, then for example, I am compute computing this. Here, I am saying that the early completion time of activity s, one will be see early start plus the duration 2. So, this is e c i j right and similarly, the late start would be L 1 minus 2 because, L 1 is the s 1. So, L 1 is 4 and then this minus the. That is what I am defining, the yeah t i j l j minus t i j. So, this would be 4 minus 2, which will be 2, which will be 2. So, 2 things should be the. So, this will be, therefore, the activity s. One can actually begin on the sec[ond]- after the second day.

See, here we are saying that s 1. Because of this activity, s 1 this number is e 1 is e s is 0 right, but, when you look at the early start of or late start of activity s 1, it can begin after the second day because its duration is 2 days. So, therefore, it’ll still be over on the fourth day and so, this activity can begin right. This will not get the late, this is the idea what I am trying to say. So, therefore, you need to know these numbers also. So, here l s i j's 2 and you see. Then, we define the slack slack is l j minus e c i j; that means, the late completion or late time for node j minus the early completion or late start of activity i j minus the early start right. So, therefore, this would be, say, for example, here you see, this would be if you subtract these two numbers or you subtract this.

So, 4 minus 2 and 2 minus 0, that is your slack and if you want to, this is a good check for you if you want to make sure that your calculations are correct. Here, then, this is a good check that this number should be equal to this number and that gives you the slack for each of the activities. And this is a very useful tool for the manager. So, you can check for some more activities. For example, if you want to look at 1 2 1 3, this difference is 3 and this difference is also 3 right and you want to look at the early start for activity 1 3. So, this is 1 2 right and we said that early start plus the duration right early start plus duration which is 6. So, that is 6 right and then you want to look at, so, this for 1 3 we are looking at. So, this is 6 and the late start would be 5 because, here the l l 3 is 9. So, 9 minus 4 would give you 5, right which is difference from this here.

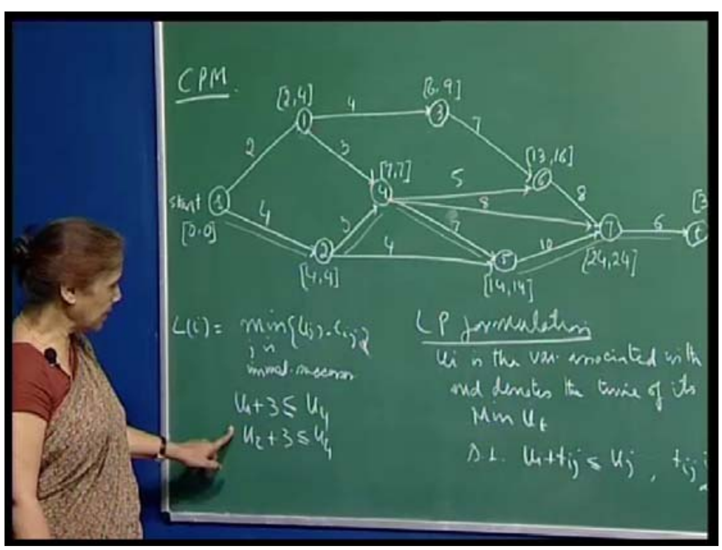
So, for example, activity 1 3 can actually begin after the fifth day right and so on and then of course, you check the numbers 5 minus 2 is 3 and 9 minus 6 is 3. So, this gives you a feeling, a little more feeling about which activities to concentrate on and which can be slack and little bit, so that, the manager can, you know, use all his energies and so on and trying to see that the project gets completed on time. Now, of course, one can also, I did nothing that it was necessarily to go into it. But anyway, I will just define this term for you. you See, you can fine tune the slack because, there is also this concept of free slack which is the amount of time by which an activity can be delayed without disturbing the early start of all its immediate successors, so, which would be something like, you know, you can look at the slacks for each. So, of course, defining this would require little more effort and so on. But, one can look at it, you know, while you are reading a some more literature, but any way, So, you can fine tune this concept of slack time, and you there something like free slack.

Because the manager may also not want to delay the early start of certain activities, so, then the immediate predecessor, the amount of free amount of slack that you can use for that particular activity should be is is equal to the free slack. So, this concept. Now, another computation that I wanted to, I mean, another point I want to make was, see here, when we make these computations, I took the due date to be 30 days. For example, for this particular project, the project is ending on the thirtieth day. So, then I took it to be other due date also, but it is possible that the due date may be different from the project completion date and so, if if your due date, which we denote by d, is greater than your l t, which is also equal to e t in our calculation, right then you see projects that is number d will be bigger than 30. So, then, there will be a slack for each of the critical activities also, because, if this is more than 30, this will be more than 24 and so on. So, the the slack will percolate on the difference between the due date and this number that percolate to all the critical activities also and of course, for the other activities also, the slack will go up that amount.

So, that is happy situation because, the manager then has event slack time for critical activities also, but, in case the due date is less than your project finish time, then you see there will be negative slack and then there is more cause for tension and so on. So, any way, you can still talk of slack being negative, it is possible right, but, obviously, the manager cannot, for example, for this project, the manager cannot in any way complete the project before the thirtieth day right. So, if you due date is less than 30, then there will be some penalty or something, but then, even then, the manager about try to see that the project does not delay, get delayed beyond this number because, then the penalty will increase.

And of course, and in the next lecture, I will talk to you about project crashing and I will show you that, with you know, one one can try to reduce the duration of a project by you using more resources and capital to expedites an activities. So, that part I will discuss later on with you. So, then, it one can also handle the case when the due date is less than your l t, that can also be possible.

(Refer Slide Time: 11:55)



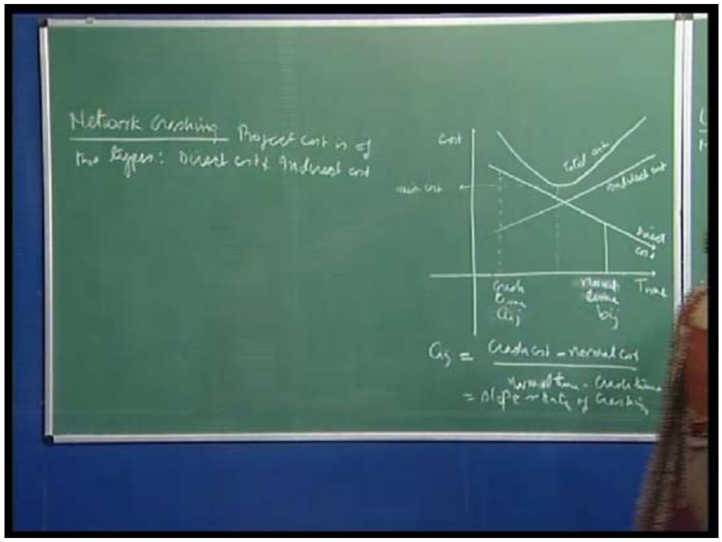
The next question that comes is that, is it possible that if you if you add more resources, for example, you put in more labor, you have more, you bring in more equipment and so on. Whatever it is, can you try to shorten the project and of course, shortening the project could mean that you're going to spend more money because, you hire more labor, you will hi[re]-, you will use more resources whatever you're using and so, the question becomes that. But then, if if the project takes a long time, like we will shorten the duration, see associated with the duration of the project error, other indirect pass I will explain in a minute. So, then the question arises, is there are optimal duration of the project? So, I will talk about it in a minute, but here, for example, as I said that you can if if you if you the linear programming formulation linear programming formulation can also be linear programming formulation would be here. If you say that u i is the completion time of u i denote is the variable associated with node I, right that is and denotes the and denotes it is the denotes the time of its occurrence occurrence.

That means, the when will this, for example, when will this event take place, that is when when will all these four activities get completed. So, u i is the variable associated with node I and denotes the time of its occurrence. So, then what do you want? to, So, here yeah You want to find out, let us say because, u s, we will always take as 0. So, you want to find out minimize u t subject to. So, now, here the thing is that you will right u i plus p i j less than or equal to u j let t i j denotes the. So, t i j is the time for activity, time taken by activity i j or the job i j job activity, whatever you want to call it subject u i plus t i j less than or equal to u j. So, here, this will take care of this thing. For example, when you want to look at node four when is the corresponding constraints (( )) just for node four, I want u to, I want to write down for u.

So, here, this will be that u 1 plus 3 is less than or equal to u 4, but we will also have this constraint to say that u 2 plus 3 is less than or equal to u 4. see you See, it will take care by itself. For example, here, this cannot starts, see this, 1 if you look at this node, it will be 2 plus 3 5, but this 1 is 4 plus 3 7. So, your u 4 should not be less than 7. And these two constraints together will take care of the fact that u fold is satisfied this and this will this is the reasonable. But, you saw that we really do not need the linear programming method of solution because, the network is the cyclic and so, simple labeling procedure gave you the labels u i's that you require by just going along and labeling the nodes right. So, this is what it is.

And now, let me talk to you about the crashing; that means, so, this is what we call network crashing. I will work out the, I mean, I try to give you what are the issues involved and then we try to come out with the method for come to finding the optimum duration of the project.

(Refer Slide Time: 15:25)



So, let us continue with networking crashing. So, what I am saying here is that project cost is of two types, this is a direct cost and indirect cost. So, as I was suggested earlier, direct cost would be the cost associated with the particular activity; that means, the number of lab[or]- the amount of labor and other resources, whatever machinery you require that is specific to a job right. And the indirect cost is actually the cost related with the whole project because, you may high premises, you may higher (( )) skills you might subcontract and so on. With the subcontract will certainly because, if you subcontract certain job, then they will come under the direct cost for a for a particular activity, but the indirect cost will be basically the cost related to the whole project which you then divide because, the project, the indirect cost will be related with the duration of the project.

And the idea here and. So, what we are suggesting is that, see if the activity takes place at his normal. So, this is, sorry, this is not cost, this is normal time. If an activity takes place gets completed in its normal time, then the cost involved, the direct cost could not be as high compared to the cost when actually gets completed during its crash time.

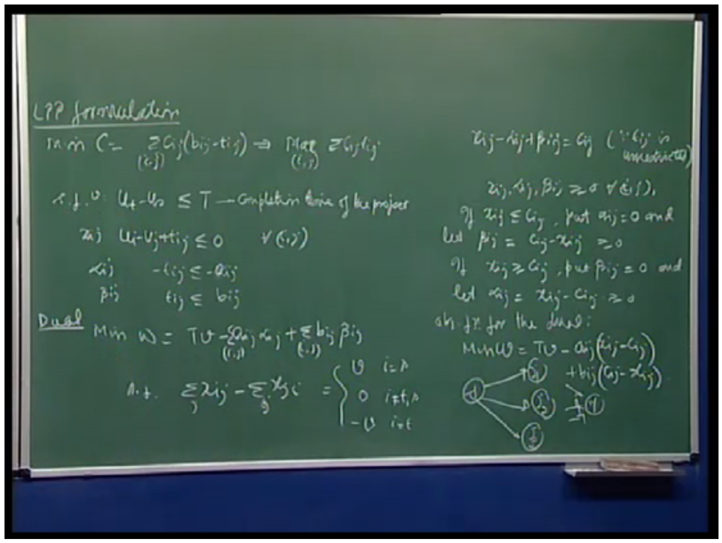
So, that means, there will be certain a time associated with an activity to which, I mean the time we can we can actually try to [compute/complete] complete the activity within that time, but certainly, you cannot reduce it further maybe crash time. So, you cannot perform, complete the activity in the time less than the crash time and normal time is the normal. So, you have some concept of what resources, what labor and other things you would require to complete the job.

And so, this is this and the indirect cost of course, because the if you complete the activity in earlier time, then the direct cost would be less, sorry, in in crash time, then the direct cost would be less and the direct cost. So, the indirect cost gets start increasing with the duration of the project and the direct cost gets reduced when the project time gets delayed right. So, you have to think of, you have to come up with the duration of the project, which is the best trade of between direct cost and the indirect cost right. So, when you have these, now this is the very simple presentation of the cost, and then you see the your total cost. So, this is your total cost curve right this is your total cost curve and the minimum will rise some way here, between the crash time and normal time for each activity. And then of course, will add up when find out for the full project, what is the optimum duration of the project?

So, these are the two costs involved. and you can yes, So, the the (( )) Skills are required to, you know, really find out what would be the, what would be the how would you compute the cost for each activity and then of course, the cost of the the indirect cost. These figures are most of the time available, because, the activities are deterministic in nature and so, you know what all is needed and so on. Ok

So, then yes. So, now, we have this diagram and then we can first try to formulate this problem as a linear programming formulation. I would call it linear programming formulation if I do not have the integer requirements for t i j the completion time. If it can be treated as a continues variable, fine, in its if the requirement is in terms of the t i j's are required to be integer, then, it will certainly be integer linear programming problem. So, here, now we are going to define c i j as the crash as the you know, if your given the crash cost for the activity and the normal cost the difference because, crash cost will be higher than the normal cost and then normal time, normal duration of the activity and the crash time; so, the difference. So, this ratio will give you the rate of, so, this, this we will say slope of slope or or rate of crashing. So, this will be the rate of crashing for each activity i j. Once we have this, then I can and let me denote this time by a i j, sorry, this let let me this denote this by b i j because, I have yeah oh no. So, this is b a i j because, a t i j is greater than or equal to a i j this is a i j a i j ok. So, the crash time we denote by a i j for activity i j and this is b i j in normal time right. So, therefore, if I define this rate of crashing of activity i j by c i j.

(Refer Slide Time: 21:20)



Then if the actual t i j is the variable denoting the actual duration, because, now that I am going to crash the activities, I do not know what the actual completion time for each activity will be. So, with this, what you want find out through this linear programming formulation; where we want to minimize the total cost. Ah. So, this would be a t i j is actual duration and since b i j is the normal one. So, you have crash the activity by this many days and so, this into c i j will give you the [cast/cost] cost of crashing the activity i j right And then, if you have u t minus u s you can treat this as 0. So, then, u t minus u s. So, t is the completion time of the project. So, suppose I want to say that I want to minimize the cost given, that I want to [complish/complete] the complete the project in time less than or equal to t right.

So, this is u t minus u s less than or equal to t and then your usual constraint linear programming, constraint that your u j must be greater than or equal to u i plus t i j for i j and then your t i j should lie between a i j b i j duration, you can do not want to go beyond normal time of the comp normal completion time and this you cannot go beyond this also in the left. Now, here, this objective function, you can write as because, this is the constant. So, I can write this as I will write this as max summation c i j t i j. So, because minus of c i j t i j. So, that becomes max and that constant is, I mean, we can just keep it aside fine, and then see if this is the maximization problem. Then, my constraints are less kind fine and here. I will write this constraint as. So, this come on the top that t i j minus t i j is less than minus a i j right the constraint was t i j greater than or equal to a i j i multiplied by minus sign; this is what it becomes. right

Now, just out of academic interest, sense we have whenever we have talked of the linear programming formulation, we have also looked at its dual and I just want to show you even though we will not use this particular formulation here. But, any way this is of interest in the, so, sort of academic interest right. So, I am trying to write the dual to this l p p formulation and you see that your right hand side of course. So, we will associate the variable v with this constraint and of course, t is your completion time of the project then x i j with these constraints and alpha i j beta i j with these two sets of constraints. So, then, if you want to write the dual, you see the objective function will be t v right Then, you have only with respective, these because this is 0.

So, minus a i j alpha i j summation this and beta i j b i j summation this, right that is your objective function and the constraints, you see, if you they should look familiar because, these are, you are a flow constraints. Remember, for the max-flow and min-cost flow problems we are written down and you see the dual, but you can also check for yourself. For example, for u s see u s is appearing here, so, therefore, that is a v there because, it will be minus v. So, I have shifted to to that side and then how else will u s appear?

Because this is i j, so, for example, you have like this, suppose you have a j 1 j 2. So, I will just explain for 1. So, then, you see you have constraints like u s minus u j 1 less than or equal to t s j 1 right u s minus you will have x x s j because, the corresponding variable will be s j s j 1. So, we will have x j 1, s j 2 x s j 3. So, here you'll have summation this and this and then of course, this appears with the minus sign. So, here of course, this is the only constraint in which it will appear. So, that'll this will be 0 and so, this will be v, so, minus v which goes here right.

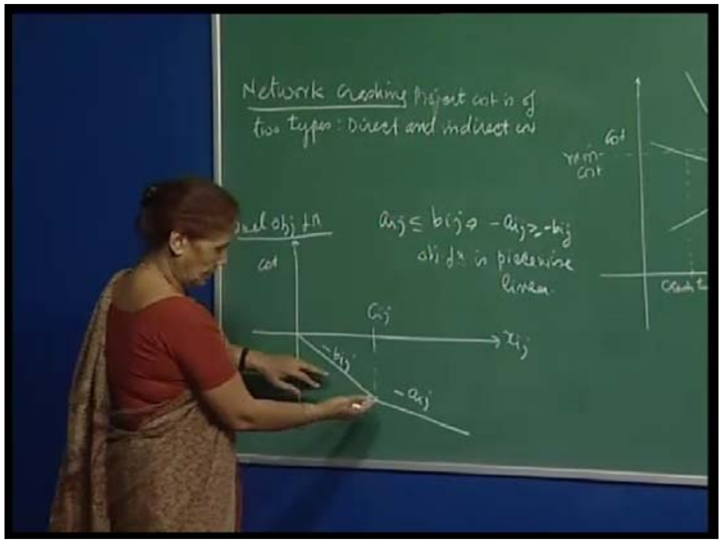
Because associated with this, it is minus v v i shifted to this side right for all other nodes this thing it will be, you can see that i j. So, for arcs going out of a node you will add up the x i j's and for arcs coming into node I, you will subtract the usual constraints we have been handling right. So, this will be 0 for I, not equal to t comma s and similarly, here, it will be v plus cosine. So, when I shifted to this side, it will become minus v. So, for I equal to t, your constraint, of course, will be minus this because, these will not be there and minus this, wherever, whichever nodes are connected to t.

So, you for example, here you have to t, you will have like this and for these arcs, it will be minus sign. So, this summation will be equal to minus v for i equal to t and then you have the last column with respect to t i j's. So, the column here would be, see this will be x i j then, minus alpha i j and minus beta i j plus beta i j, sorry, for this, it is plus beta i j right and since the we have, we do not have any sign constraints on t i j. Remember, we have made all these explicit constraints the known and non-negativity constraints on t i j's. So, therefore, the corresponding this constraint will be equality here and here also, we are no no restrictions on the signs of u i's and therefore, thought, of course, when you start with your u s as 0, all other u i's will also be positive so, but we are not mentioning the constraints explicitly. So, therefore, all these constraints are as equality right and now we just can simplify the. So, any way, these constraints show you that there is a flow going out of may. You can look up on v as a flow which is going out of a node v and it is reaching node t and the other nodes act as transshipment points.

So, these constraints are definitely your flow constraints. Now, this constraint I can simplify and make it in the implicit manner and what I will do is, see if x i j is less than c i j. Then you can put alpha i j equal to 0 because, then beta i j being non-negative, this some can make up for c i j i can just dispense with alpha i j and put it equal to 0.

In that case, my b i j will become c i j minus x i j which is non-negative. right Similarly, if x i j is greater than or equal to c i j, then I will put b i j equal to 0 right because, this will act as a surplus variable and then I will define alpha i j is x i j minus c i j, which is also non-negative because, x i j right and then the objective function for the dual. See now, I can remove my alpha i j be beta i j. see You see, I have taken care of these constraints already and. So, then, I will substitute for alpha i j alpha i j is x i j minus c i j right. So, this is minus a i j this is x i j minus c i j right here, and then plus b i j c i j minus x i j and you can look at this constraints. So, you see here, the objective function therefore, is not linear function completely, it is a piece-wise linear as we call it and you can see from here.

(Refer Slide Time: 28:45)



That, see, yes, because for x i j less than c i j your slope is minus b b i j minus b i j and here the slope is minus a i j for x i j greater than c i j right this is this and which is which is valid because, you have a i j less than or equal to b i j. So, this implies minus a i j is greater than b minus b i j and tan theta is a increasing function right. So, therefore, this slope would be, so, you have minus a i j greater than minus b i. So, the corresponding angle will be bigger right. Because stand is an increasing function, even though they are negative. is it. So, this is the idea. So, the therefore, therefore, the objective function for the dual problem is a piece-wise linear function and we can solve this problem as a min-cost flow problem all right.

We are We have already learned the method. So, my idea is to show you that, see trying to show you that, where all you can get these kind of formulation that we have studied under the banner of linear programming and the how versatile, the simplex algorithm is because you can use it anywhere. Of course, it is possible that you may have some better algorithm for solving this problem. But, you see that it can be formulated as a min-cost flow problem, except that the objective function is not linear and we have methods available for for handling piece-wise linear functions, because, all you have to take care is the moment your x i j is becoming bigger than c i j, then you have to take this as the cost coefficient and for x i j less than c i j you have to. So, is small modification in your current network simplex algorithm that we [dod\did], we can handle this situation also. So, you can solve this problem this way, but of course, I will give you a more direct method in the next lecture. Now, just try to look at this also. See here, of for example, if your completion time of the project is given to be very large, in that case, you know this constraint will always be satisfy will never be satisfied as equality. If your t is large, in that case, your v will be 0 right and so what it is means is that, you can look up on v actually because v is not the flow in the network.

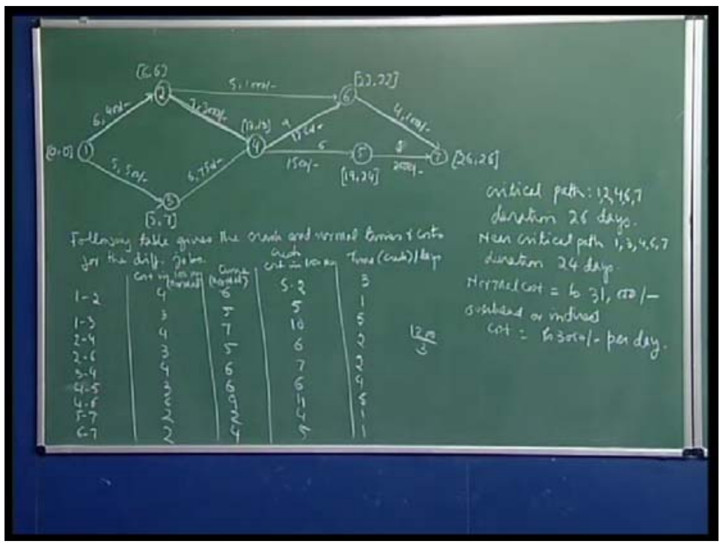
But, when you look at the dual, we are treating it as a flow value, but what it will say is that, v will be 0; that means, you do not need to spend any money on having extra resources to shorten the project duration, because, it is, I will anyway very large right, but if t is very small and this constraint is, you know, it is not even feasible. That means, this number a because, when the u i is you compute the come satisfy these constraints, but they this this constraint is not satisfied in, that means, the primal is infeasible and therefore, the dual would be unbounded. That means, any amount of money that you spend in trying to shorten the project, would be would be would do not help you, because, anyway you cannot meet the current t value. So, by shortening t, you will again, what is money you spent, you will not be able to complete the project in that time. So, this is the kind of interpretation. Again, the idea was that you see what meanings the dual variables can acquire and they can give you an understanding of the problem.

So, this was the idea behind by, even though I know that we will not be solving this problem in this way, but I still wanted to show you the how far the applications of linear programming problems can go. yes and now here. So, I think, that tells you that the crashing of the network project and finding out the optimal duration can be formulated as a linear programming problem, but here again, and I have shown you the dual which is a interesting problem by itself, but again, there are ad-hoc methods which are faster because, as I said that in case your require your t i j is to be integer value, then of course.

We have learnt the techniques for handling such problems and also solving integer programming problems is not a very happy situation, because, the things get complicated. So, we have fortunately, nice ad-hoc method which will also give you optimal solution and. So, I will like to discuss, but, because this is a linear programming and its extension course, I thought I should show you limitations of linear programming also, because, it always helps to be able to exploit this structure of the problem and this is and. So, as you saw, that when finding out the critical path, because, the underlying network was a cycling, we could with the simple labelling technique, find out the longest path and the critical path ah. So, we did not need the linear programming formulation and here also we can formulate the problem as a linear programming problem, but we will not do it because, we can, we will exploit the structure of the problem and find out, come up with a faster method for computing the optimal duration of the project.

So, I will indicate that through the through a simple algorithm and preview the method. So, to show you how network crashing is done or duration of the project, I will go through the procedure showing you the steps for an example.

(Refer Slide Time: 35:42)



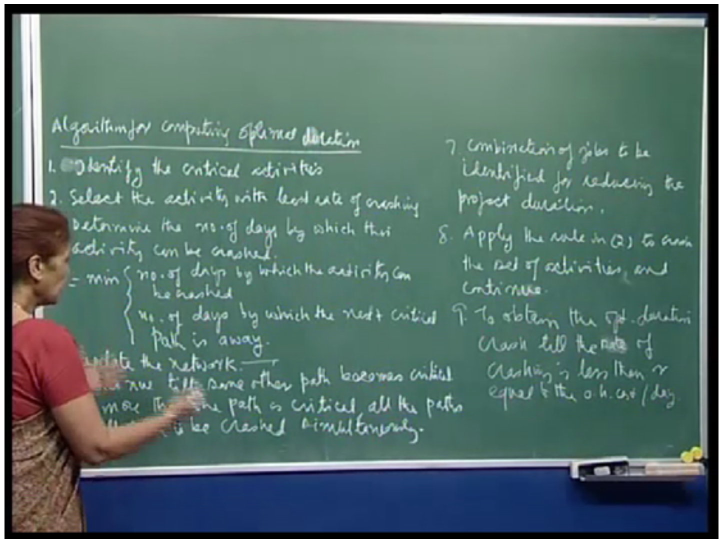
So, this is here, the example that I have for written down. So, network flow problem and I have given you the see this cost and this time. So, this is also for normal normal duration. When the normal duration of the projects, for example, for project 1 for the job 1 2 activity 1 2 and so, and the normal cost is 1000 of rupees. So, this is this is the cost. So, this is 4000 rupees is the normal cost for the activity 1 2 then, normal time taken to complete the activity would be 6 days, this is in days and then this will be crash ok. I could have written here, a crash cost. So, this is crash cost in 1000 of rupees again. So, to crash the activity by one day or total to reduce the time from 6 days to 3 days for activity 1 2, the cost of crashing it into 3 days would be 5.2 1000; this is how we are going to read it.

So, for then for example, what I have done is here, this is the normal time, this is the crash time. So, 3 days, you can crash the activity by 3 days and the difference in the cost is 1200, So, you can so that means, 1200 divided by three, which is 4000 rupees. So, here, I have written the rate of crashing the activity by 1 day. So, that is this is the rate of crashing. Similarly, you can verify that for job 1 3, the rate of crashing per day is 500, because, here the cost is 3 and here this is 5. So, 2000 is the total cost of crashing and this is from 5 to 1. So, 4 days, you can crash the activity by 4 days and therefore, the cost of crashing per day would be 2000 divided by 4 which is 5000.

So, you can you know the understand data given here and as we see that the critical path, I have already pointed out to you; I mean, this is the example we discussed earlier. So, 1 2 2 4 4 6 and 6 7, so, this is your critical path and the total duration is 26 days, the near critical path is 2 days away which is, so, there path is 1 3 4 6 7 and its duration is 24 days. Now, the total normal cost if you just add up; that means, if you do not crash any activity, then this adds up to 31000 and then the over head cost that we said indirect cost, is 3000 per day. That means, reduce the project duration by 1 day, then you save a rupees 3000 in terms of indirect cost.

So, you have to now balance the cost of crashing and the which will of course, increase the cost. But then, since you are reducing the duration of the project, the over head cost would come down write. So, this is what we going to do. So, first let me just outline the algorithm and this, as we we formulated the problem as a linear programming problem. But, I want to know show you that because network is very simple, we can use an ad-hoc method to get to an optimal solution and this is where you know, you can see that optimization is not simply applying an algorithm to a problem; it is also trying to see what is the structure of the problem and if you exploit it in some way or if you can use some other information to get to an optimal solution, you you should do it.

(Refer Slide Time: 38:09)



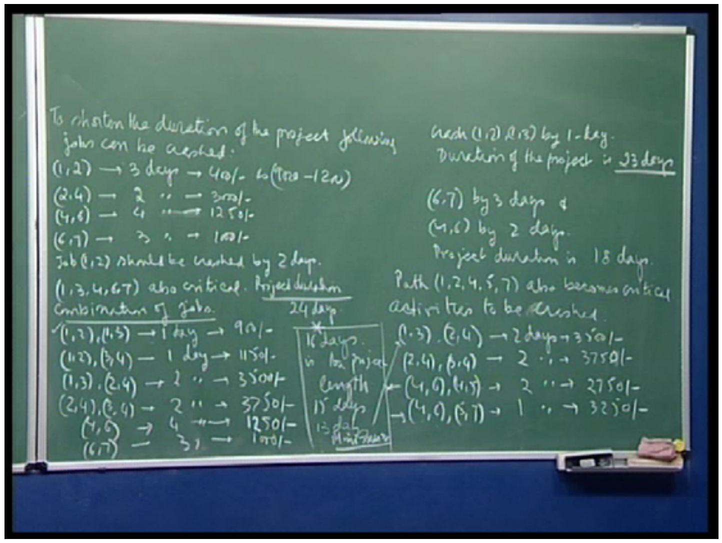
So, therefore, let me just give this ad-hoc algorithm that we called it; but it gets you the optimal solution also. So, first what we will do is, identify oh this is identify Identify the critical activities So, which will currently, our critical path is 1 2 4 6 7. So, these activities 1 2 2 4 4 6 6 7 are the critical activities, then we will select the activity with least rate of crashing. So, out of these 1 2, there 4 4 activities, I will select the one which was the least rate of crashing and then I would have to determine the number of days by which this activity can be crashed. Which means that, see, this will be have to be the number of days by which the particular activity I have selected here, can be crashed and the number of days by which the next critical path is away, because, if I crash this two, whatever number I number of days, I can crash this activity, but, in the mean time, some other path becomes critical, which is longer. Then, it does not help me to reduce the project duration. right

So, the idea here is that, I should be able to reduce the project duration by crashing a particular activity. So, here for example, if you look at 1 2 1 2 can be crashed by 3 days right which means that, I can if I crash the activity 1 2 to its crash time of 3 days, this path length will become 23, but then, the next critical path is twenty four days 24 days. So, that means, my project duration will not come down to 23 days; this is a whole idea. So, this is important you have to always choose the minimum of the 2 numbers. So, then, once you decide by how much the activity can be crashed, then we will update the network, that which means that you can somewhere wrote down note down is to, what the current duration of the for particular activity which you have crashed; you have denoted somewhere and then you also compute the improvement in the cost reduction in the cost right. Then continuing to some other path becomes critical. So, we will continue doing this 2 and 3 till some other path becomes critical and of course, here, I should have mention that, since you are looking for an optimum duration, you will also check that you will decide to crash an activity as long as the crashing cost is less than the over head cost, because, you want to cut down on the cost also right.

So, that part I have not added here, but certainly you will keep doing that. So, then continue to some other path becomes critical and then after if, there is more than one path that is critical. Then, you see, you will have to combine, you will have to crash all the paths which are critical simultaneously, right to be able to reduce the project duration. So, to do that, you will have to pick up on combination of jobs. That means, you will have to identify the combination of jobs, on the on all the critical paths so that, by crashing that set of jobs or that combination of job, you can reduce the project duration. Once we do that, then again, I will apply the same rule that I used here. I will select the set of activities which can, which has the least rate. of You will add up the rates of crashing the activities in a particular combination and you will select the one which has the smallest rate of crashing

So, same thing you will continue and of course, you will to to obtain the optimal duration crash till the cost of crashing is less than or equal to to the. Again, I am not writing it very precisely, but you understand, I should they said the till the rate of cost to obtain the optimal duration crash, till the cost, the till the rate of, I should say here, rate of crashing is less than or equal to the overhead of cost per day, because, you want to balance. So, this is the idea and the moment you find that your cost of rate of crashing exceeds the overhead cost, then your total cost will also start going up. So, this is the idea.

(Refer Slide Time: 48:01)



And so, now, let us just apply this ad-hoc algorithm to this example and will I will try to show you some steps in detail, so that you can continue with the algorithm. So, you see here, to shorten that duration of the project, following jobs can be crashed. So, these critical jobs, critical path is this. So, these are the four jobs on the critical path. So, you see 1 2 can be crash by 3 days and the rate is 400 rupees per day. So, I have written down the rates here and these are the days by which each activity can be crashed. Right,,, But, yes, I told you earlier that job 1 2 should be crashed by 2 days, because, then the path 1 3 4 6 7 1 3 4 6 7, see, that becomes also critical, because, then the duration of the project, once I crash activity 1 2, the project duration comes down to 24 days and so, this path also becomes critical. So, now, I need to combine activities on the two critical paths so that when I crash the two these set of jobs simultaneously, then the two critical paths will also get crashed and so, the project duration will come down right.

So, then, here these are the combinations I have written down. If something is missing you please check for yourself. So, anyway 1 2 and 1 3, see, you can see that this this is the second path 1 3 4 6 7. So, these activities are common and these are the two pa[ths]-, these are the activities, whether two paths differ. So, it can be 1 2 1 3. if I crash both of these, then the total project duration will come down, because then, both the paths will get shorten. Similarly, I take the other combination 1 2 and 4 4 ok. So, 1 2 and 3 4 will help me to again reduce a project duration, then 1 3 and 2 4 will also help me reduce the project duration and then you see here 2 4 and 3 4 yes 2 4 and 3 4. So, I think this takes care of all possible combinations here and then you see if I crash 4 6, it is common to both the paths. So, then again the project duration comes down and similarly 6 7 because, that is also common to both the paths. right

So, here I have written down the rates again and see, you can immediately see that this activity, this set of activity is 1 2 and 1 3 can be crashed and. So, here, I can only crash it by 1 day because, 1 2 now has left only one day slack and this is what I meant that somewhere you will have to note down that once you have crashed activity 1 2 by 2 days, when one more day is left for it to be crash and so, this cannot, this pair cannot be crashed by more than one day, but our criteria is to go by the minimum rate since we are looking for an optimum solution. So, therefore, this we will decide to crash by one day and in that case, the project duration will come down to 23 days right and then you can crash 6 7 by 3 days because, 6 7 is the next 1000 rupees rate of crashing others or higher than 1000. So, I will crash 6 7 by 3 days and then 4 6 can be also crashed by 2 days because, this is 1 2 5 0, all these numbers are less than ah.

See, and here, once I decide to crash this one, this pair, then, I cannot do this because, this is ruled out. Remember, 1 2 has come to its last, the duration has be reduced to three days, I cannot reduce it further. So, then, this is ruled out and all the other two sets of activities are more than 3000. So, I will not consider them right now. Since, I am looking for an optimal pair, I am trying to get an optimal solution. So, therefore, these are the three activities which I can crash right and so, by doing that, the project duration finally comes down to eighteen days. So, both the paths have been reduce to eighteen days, but in this ca[se]-, in the process, now path 1 2 4 5 7 1 2 4 5. So, this also becomes, you see this here, this is 8 and this has come down to 3, so, 3 plus 8 10 10 plus 8 18

So, this path also now has duration of eighteen days. So, the you have three paths are which are critical right. So, now, in that case, you will have to again look at the combination of activities, which when reduced together, will reduce the project duration; that means, will reduce the duration of all the three critical paths. So, here again, 1 3 and 2 4 if you look at 1 3 and 2 4, yes, because, they have this one is common to 1 2 4 5 7 also right. So, therefore, I can crash 1 2 and 2 4 which will help me to reduce the duration of all the three critical paths. So, this way you can just verify for yourself that these are the four possibilities and which I can crash all the three critical paths and so, here again, I will choose this 1 4 6 4 5, because, this is the cost of crashing is 2 7 5 0, which is less than 3000 rupees; your over head cost is 3000 rupees per day. So, I can still crash this set of activities and improve my total cost of completing the project right. So, therefore, this gets crashed by two days and so, yeah lack of space, maybe I can do it here yeah. So, now, that means, 16 days 16 days is the project length. right Once I crash the set of activities, my project length comes to sixteen days and now you see that after this of course, yeah this pair is a valid because 4 6 can be reduce by how many days; 4 6 can be reduced by 6 and 4, oh. So, no 4 6 can be 9 and 5. So, this 4 days, so, therefore, I can also still for the crash this, but, which is the least one. yeah this is right

So, now, that means, sixteen days will be your project length, which is your optimal, because, after this I can crash this pair by 1 day, but the cost is 3 2 5 0; that means, my cost is going to go up by 250 rupees. So, the moment your cost of reducing or completing the project and by crashing go starts to go up then; that means, you have reached the optimal duration here and so, you can further reduce the project duration to 15 days, for example, by crashing this pair right and then if you want to do it further. For example, sometime for the manger wants to know how far can you go and what is the minimum duration; even though the cost is may go up, but, you still want to know. So, like this is 15 days and then here you will want to crash this 1 by 2 days so, that means, it will come down to 13 days, if you crash this set of activities one 3 and 2 4. So, which we do this by 2 days then, I think this pair is out because, 2 4 2 4 is how much for 2 4; the duration is only 2 days, you can crash the activity by 2 days only. So, you can either choose this pair or this pair; right you can cannot have both. So, therefore, this is way you will stop because 3 5 0 0 is lower than 3 7 5 0. See, you would still want to know what the minimum duration is and then corresponding cost should also be the smallest.

So, therefore, if I decide to crash this pair, my duration will come down to 13 days. So, this is your minimum duration this is your minimum duration and after this sit down and see for yourself that you cannot crash the activities any further, even if you want to increase the cost. So, this is the your minimum duration and sixteen days is your optimal duration. So, essentially I have not made the actual cost computations which you can do it along each of this, because, you know by how much you reducing the cost. So, you started with the initial cost of 31000. Now, you can go and make the reduction in the cost and you'll come up to here, right which means that, for example, this was 3 days, so, 400. So, here you were making a reduction of 9000 minus 1200 right this many rupees, this was the reduction 9000 minus, you will subtract this number from 31000 to get the cost of crashing the project by 3 days. right

Similarly, when you choose this pair here and this was by one day. So, then the reduction here would be 3000 minus 900. So, that is the amount you will reduce the from your total cost. So, please do this computation and you will see that the cost keeps coming down and then you will stop at sixteen days which will give you the optimum cost. So, here what we want to say is that, it may not always be necessary that you to get the minimum duration, you have to crash every activity; that will not be feasible and in fact, it will not reduce the project duration beyond 13 days. So, do not say that if you want to find out the minimum duration of the project, that you just add up the active[ity]- you know the crash duration of each activity and you know say, that will be the minimum duration. That means, you compute the, you you take the crash duration of each activity and then compute the longest path that may not be equal to thirteen days, I mean, that will not be the minimum duration. That is what I am trying to say, that is, that may still not require each activity to be crashed, because, after all, you know that the project duration is determined by the critical path.

So, essentially the activities which become critical they have to be crashed so that you get the minimum duration. So, some activities for example, here 2 6, I do not think figures in any of the reduction thing. So, when you compute the minimum duration, your 2 6 continuous to be because, this path is 6 11 and 11 4 15. So, at 13, the duration, no no you crash this to 3 days and this this activities 6 7, we crash to how much, we crash 6 7 by 3 days ok. So, that means, here this is 1. So, this path will continue to with 3 plus 5 8 8 and 1 9 right. So, you see minus is much below thirteen, therefore, there is no need for activity 2 6 to be crashed. I mean this is one example I can give you right away right. So, therefore, minimum duration does not mean that each activity of the project will be performed at its crash time only, the activities which help to reduce the duration of the project, they have to be crashed.

So, essentially I have now, this is of course, you might say that lot of this thing can be programmed easily, but, you can see what is happening from day to day and so, essentially, and this is what the manager likes to know that ha from day to day, if you are managing a project, how it is happening, and how the cost is improving, what activities need to be crashed and so on. So, nice ad-hoc way, which can be very easily implemented, but of course, you when the project is large, you might say that the number of, see for example, this combinations will go up, but again, you have simple rules to identify the combination of activities, which you will help to reduce the project duration and so on.

So, therefore, what I am trying to say again, which I said in the beginning is that, you know in optimization, you do not have to be strictly go by certain rules. You you may have an algorithm to guide you, to take you towards optimal solution, but in the process, you can also exploit the structure and try to use some common sense and whatever information you can you can have to to go to the optimal solution; this is the whole idea.