Business Analytics for Management Decision
Prof. Rudra P Pradhan
Vinod Gupta School of Management
Indian Institute of Technology, Kharagpur

Lecture – 52
Prescriptive Analytics 3 (Contd.)

Hello everybody. This is Rudra Pradhan here welcome to BMD lecture series. Today we will continue with prescriptive analytics and that too coverage on integer programming. In fact, in the last couple of lectures we have discussed the concept of linear programming where the requirement is to optimize the business problem and to find out the values of decision variable through which we can address the business problem more effectively and as per the management requirement.

And typically in the last two lectures, we have discussed the concept of integer programming where the requirement is the something more than the kind of general linear programming problem. So, it is the case where we are not optimizing the kind of models; where we can take a kind of decision, but the decision in such a way that the values of the decision variable should be integer type.

Because in reality some of the business problems. So, you know the focus or the kind of decision will be more effective and the more efficient; if the values of the decisions variable will be coming you know and the kind of integer type. So, as a result we have discussed these problems what is the need or the kind of scope of integer programming. And in fact, we have also solved some of the problems typically by using cutting plane mechanism C P M and then branch and bound mechanism. And in fact, we have also used the solver package to solve problems and look for the integer type solutions.

In fact, manually a it is like you know simple linear programming problem we start with the kind of optimal solutions. So, once we reach the solution by graphical technique or simplex technique then we will we like to see whether the particular problem is optimal and satisfy all constraints all conditions and also the values of decision variables are integer type.

If that is not the case then integer programming has a role and that too we can use either cutting plan mechanism or branch and bound mechanism to again you know revised the particular you know optimal solutions. And then look for the further optimal solutions or alternative optimal solutions where the values of decision variable will be integer type. And the problem is also optimal one and what we have discussed in the last couple of lectures.

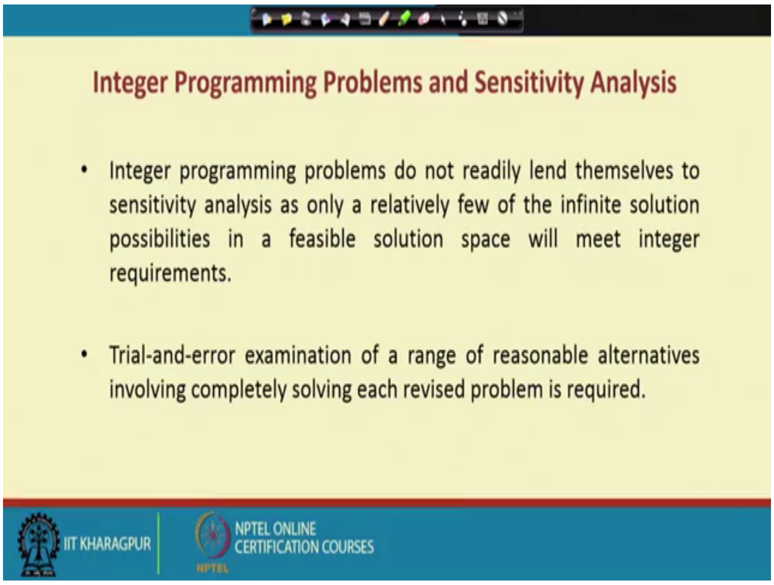
So, with respect to original problems and the kind of original optimality and which is not integer type then this is a kind of additional constraints or additional conditions as a result. So, we can look for an alternative solution where the values of the decision variable will be integer type and we have seen. So, when we put the restriction of integer type then the value of the objective function will be little bit you know compromised.

So, as a result; so, original optimal solution the values of the objective function and the solution where the values of decision variable are integer type where the objective function value will be little bit you know lower compared to the original optimal solution. That means, a since it is a kind of additional constraint a you know then we have a kind of solutions which can satisfy the business requirement, but the values of the object objective function will be compromised.

So, that is how the typical you know structure and in the last lecture we have discussed about the cutting plane mechanisms. Now here we are discussing the concept by branch and bound mechanisms and the kind of 0 1 integer programming. And so, for that this typical structure is like this. So, what I have already mentioned it is the structure of sensitivity analysis and there are two parts if you go by manually then simple problems go by primal kind of solution by using simplex mechanism.

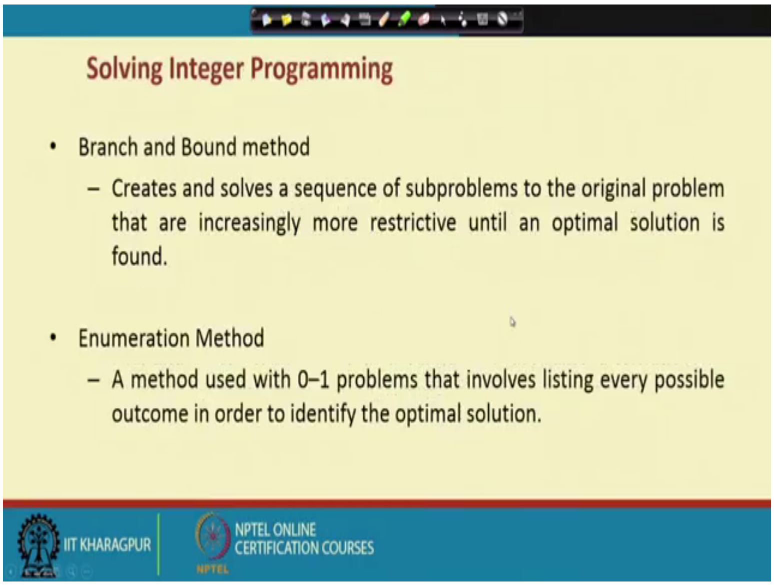
And then you have to check whether the type of solution is optimal and then integer type if not then against we have to add a constraints either particular mechanism or branch and bound mechanism then again you know start the iterative process till you get optimal solution which is integer type again.

(Refer Slide Time: 05:04)



So, for that the second part or the second phase of this particular process is nothing, but you know sensitivity analysis structure and mostly we use actually dual simplex mechanism again to solve the second part of the problem to solve the a you know solution to solve the problem again to get the solution which is actually integer type. So; that means, it is a kind of mixture of primal simplex dual simplex and the application of sensitivity analysis and it is as per the business requirement and the kind of management requirement.

(Refer Slide Time: 05:40)



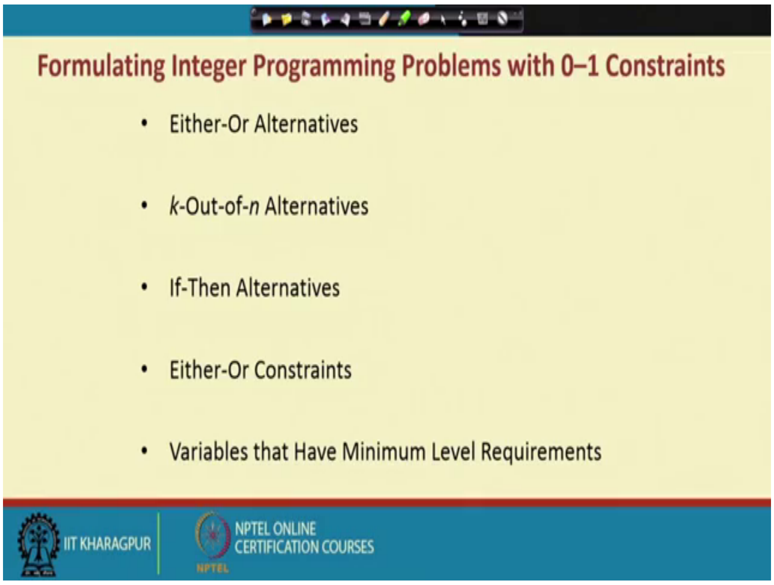
So, corresponding to the kind of discussion which we had in the earlier; so the branch and bound mechanisms so; that means, we you know we like to check whether the values of decision variable are integer type or not; if not then the particular variable which you can find out of all the variables. So, can have you know two different branch then against there are two different constraints and accordingly there are two different models to check and then look for the optimal solution.

So, like means that is how this particular mechanism is called as you know branch and bound mechanism. So, it is like you know these entries; so, it has lots of branching till you get an optimal solution which is actually integer type. And the second mechanism is we have to solve the problems where you know the values of the decision variable will be binary in nature that is you know 0 1 you know interval only.

So, that 0; that means, either 0 or 1s not in between so; that means, technically so, we can use the branch and bound mechanism and then we use a kind of mechanism where the values of the decision variable will be integer and that too strictly a you know within the kind of structure of 0 and 1 only that is the binary numbers only.

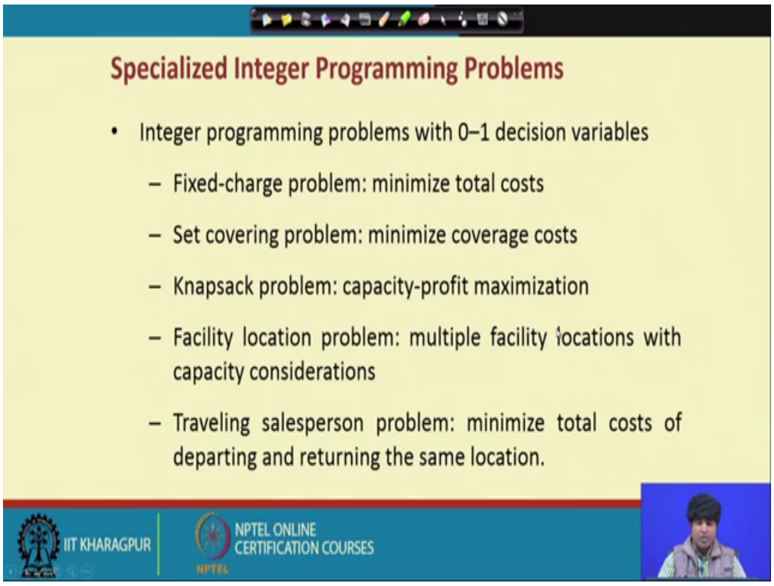
So, let us see how is the kind of structure and so, far as formulating integer programming is concerned. So, we have a either or you know alternatives.

(Refer Slide Time: 07:08)



And k out of n alternatives if then alternatives either or constraints and variables that have minimum level requirements.

(Refer Slide Time: 07:18)

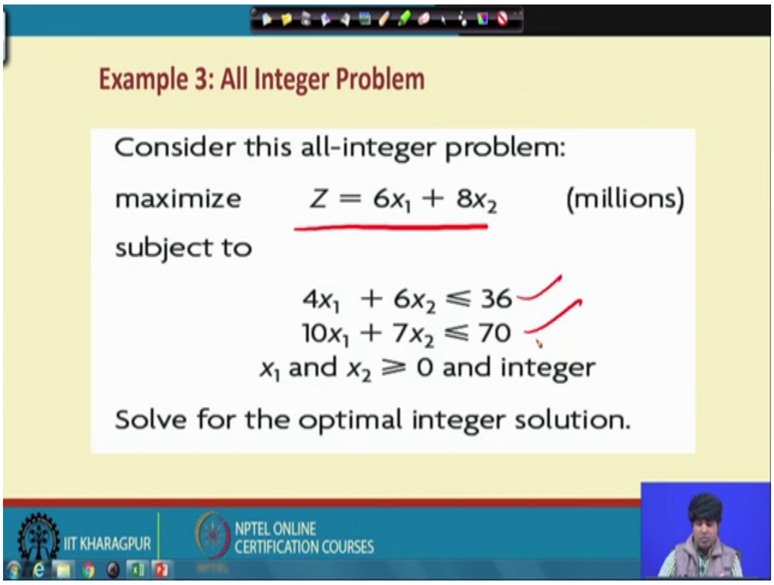


So, these are all actually kind of structure. So, the some of the specialized integer programming problems are you know fixed as problem where we like to minimize total costs set covering problems where we minimize coverage cost.

So, likewise we have a couple of you know specialized integer programming. That means, this will give you the kind of snap shot that you know integer programming problem is having a special kind of entity in the prescriptive analytics where you know some of the business problem you know highly requires.

And without such kind of tool it is very difficult sometimes to predict the business requirement or to predict the management requirement and then you may not be in a position to come with a kind of management decision through which you can address the problem more effectively.

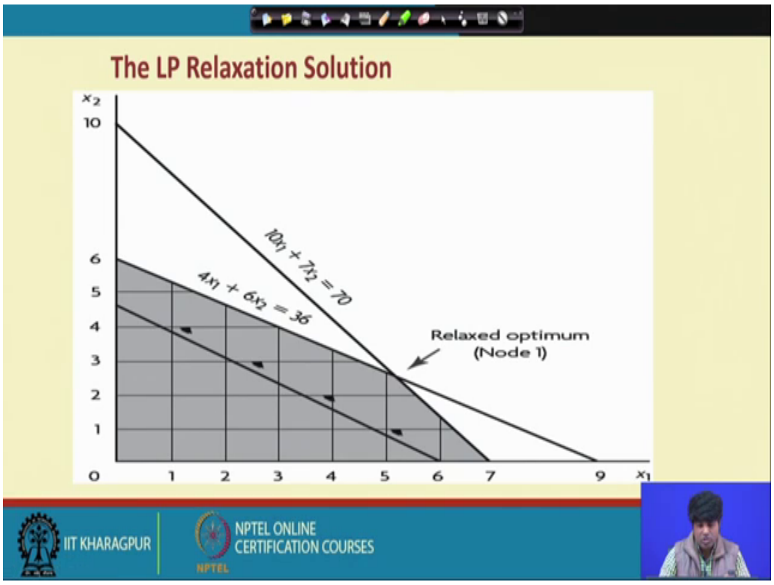
(Refer Slide Time: 08:10)



So, as a result we like to check the kind of structure integer programming. So, what we have already discussed here; so, this is a simple example of integer programming where we have a maximization maximizing objective function Z equal to 6 x 1, 8 x 2 and we have two you know constraints less than type a you know both are less than type and x 1 x 2 greater than 0.

So, that is the non negativity restriction and then the solution must be integer type so; that means, the in the optimal you know stage. So, the values of the decision variable must be a integer type.

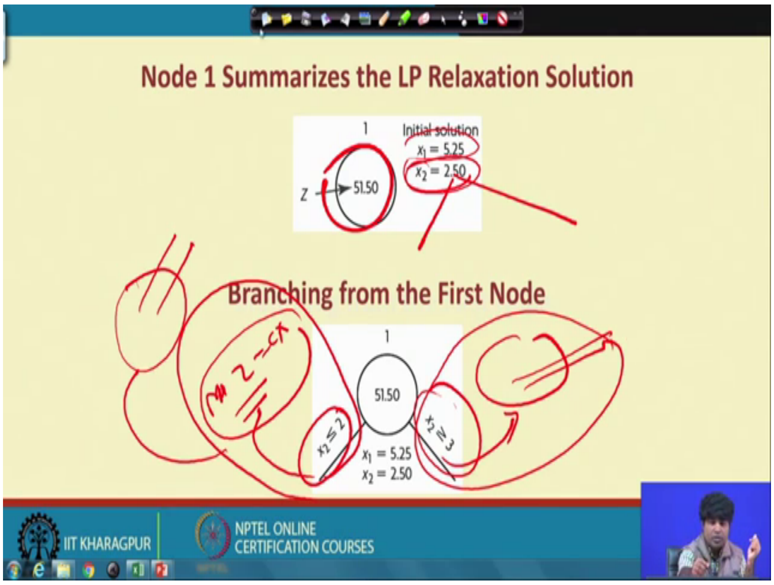
(Refer Slide Time: 08:52)



So, with this we can start the particular you know process and then check how is the kind of. So, we can have also graphical structure since it is actually with respect to two variables and if it is the case of more than two variables then we can go by a you know algebraic method or simplex method to solve the problems and look for the optimal optimality and then the integer type solution.

And in fact, you know after solving this particular you know problems.

(Refer Slide Time: 09:20)



So, the solution is the like this and the initial solution is here like this you know the Z value is 51.50 which is not actually our you know requirement our requirement is to you know check whether the values of the decision variable is integer type or not.

If that is the case then we can look the value of the objective functions; if the values of the decision variables are not integer type then no point to check the value of the objective function because this is our requirement and this, our the condition. So, since both the variables are not integer type for this corresponding problem. So, as a result; so, we like to, so corresponding to this particular you know problem.

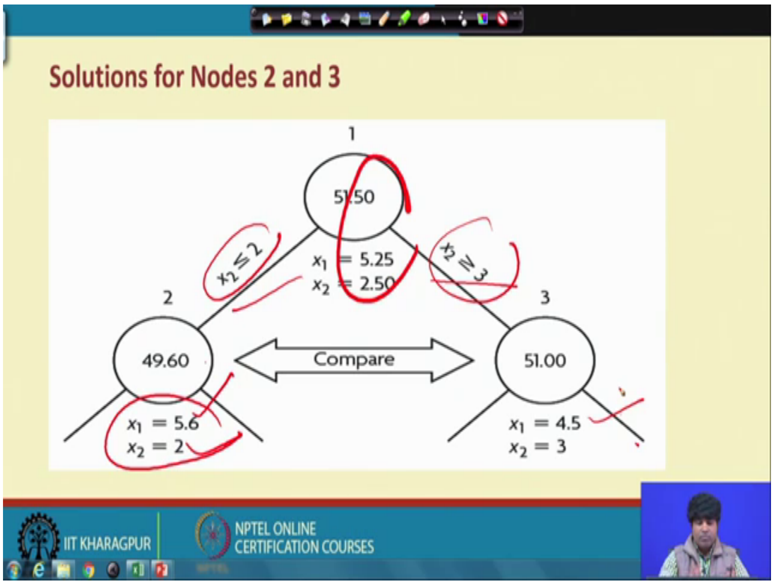
So, this is problem. So, the solution will be x 1 equal to 5.25 and x 2 equal to 2.5. So, as a result both are having actually non integer; so, then we start the branching. So, the branching can be with respect to either x 1 or x 2, but there is a you know system through which you can choose a particular variable.

And as a result let us start with you know choosing x 2 you know because it the kind of the non integer value is the 0.50 which is higher than compared to x 1 which is 0.25. So, as a result; so, x 2 is 2.5 and so. So, what is happening here? So, we like to you know put the kind of structure here. So, this will be giving you two different branching and that too one is with respect to a x 2 less than equal to corresponding x here 2.50 and another constraint will be x 2 greater than and greater than equal to 3.

So, that is how the two different branching can be done so; that means, the original problem maximizing Z equal to c x and we have two different constraints then this will be third constraint to this problem. Again the same original problem will be here and this is the third constraint will be added. So, now, technically we have two different sub problems; so, sub problem 1 and then sub problem 2; against we solve this problems and they get the optimal solutions and check whether they are integer type.

Again we will solve here and look for the optimal solutions and check whether the solution is integer type or not. If both the cases it is integer type and we reach the optimality then that could be final solution and we can have a two best alternatives and then find out which one is the best as per the particular you know problem requirement or you know business requirement. So; that means, technically corresponding to these problems we have a two different you know sub problems. So, now, we look for the solution of these two sub problems.

(Refer Slide Time: 12:24)

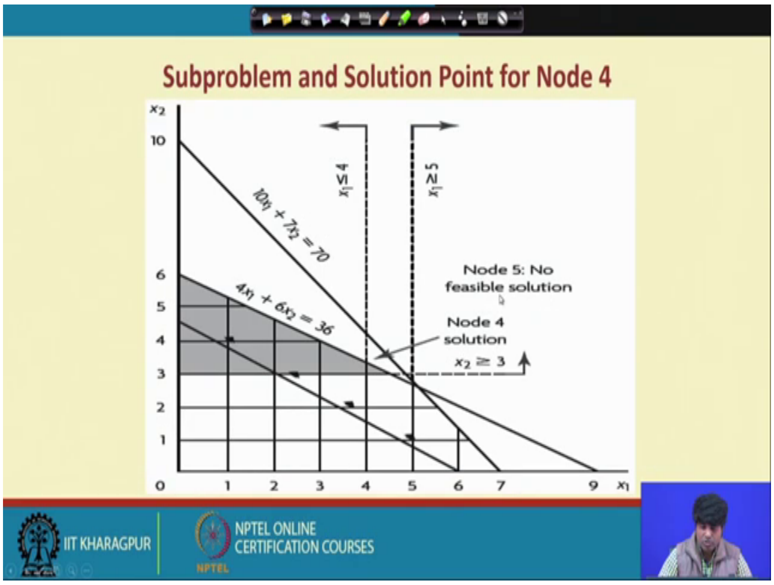


So, where x x 2 less than equal to 2 and x 2 greater than equal to 3 so; that means, corresponding to original you know graphical plotting. So, now, we have to add actually two different constraint here then we look for the solution. So, now, the corresponding this is the original problem; this is what the original problem now. So, the first partition will be x 2 less than equal to 2 and second partition is x 2 greater than equal to 3.

So, now, this is additional constraints and corresponding to these. So, we have a solution here and that too x 2 is integer type x 1 is not integer type again corresponding to x 2 greater than 3. So, we have x 1 4.5 and x 2 equal to 3. So; that means, in this case x 1 is not integer and x 2 is integer again here x 2 is integer and x 1 is not integer. So, that means; so, now x using x 1s then we may we will have here two different branching; one is the x 1 less than equal to 4 and again another one x 1 greater than to 5.

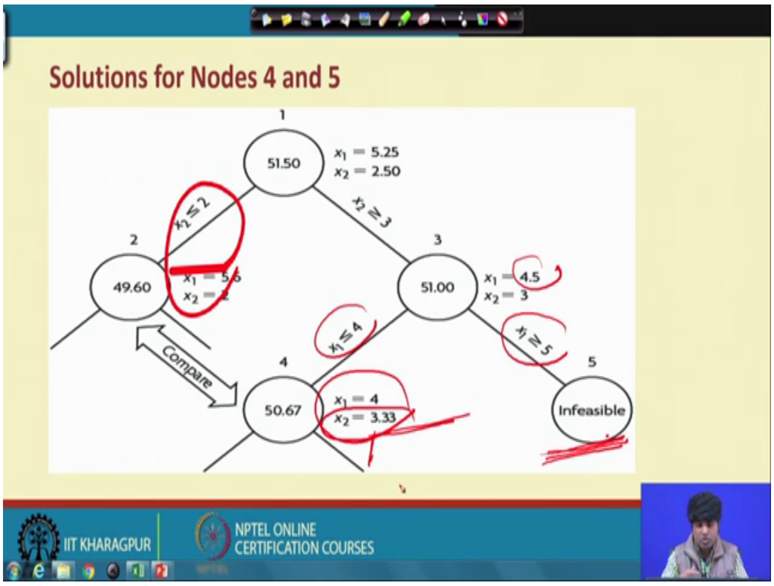
Similarly greater than to 6 and similarly here one constraint will be x 1 less than equal to 4 and the other one will be x 1 greater than equal to 5. So, then; that means, technically corresponding to original problem and the optimal solution we have a two different branch problems. And then against corresponding two different branch we have two different you know sub problems again so; that means, technically now we have a four different sub problems and against we like to check the optimal solutions and then we like to compare. And finally, conclude which one is the best requirement for this you know business problem.

(Refer Slide Time: 14:00)



So, as a result we like to move the kind of procedure so obviously. So, the next procedure is the; so we like to you know add this two constraints and then look for the optimal solution.

(Refer Slide Time: 14:10)



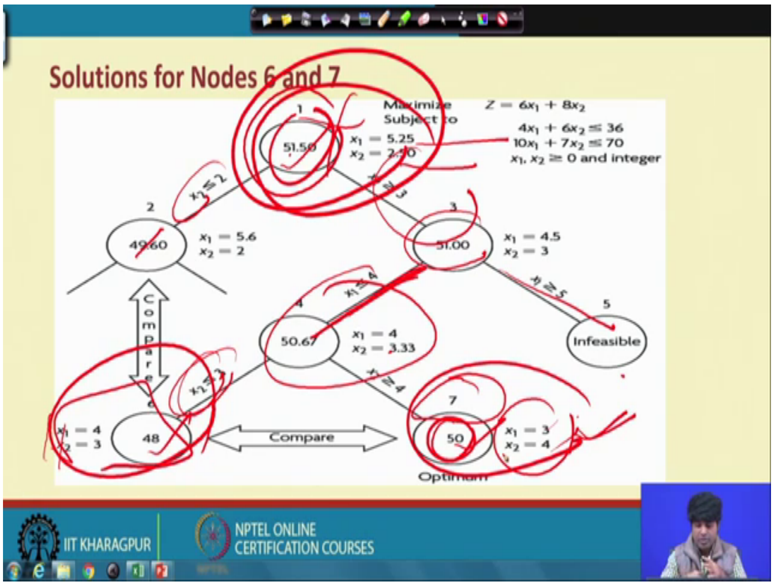
And this is what the particular you know structure here. So, corresponding to this original problem here and this is what the solutions now; so, keeping this one constraints we can look for this one and here this is 4.5. So, one will be x 1 less than equal to 4 and another one is x 1 greater than equal to 6.

So, now, against this is the fourth constraint to the original problem. So, as a result we look for again optimal solution. So, after the optimal solution we have x 1 again integer type and x x 2 is non integer type again; so in this case putting x 1 greater than to greater than equal to 5; so, our indication is that you know infeasible one.

So; that means, this; so, the loop will be close here in this side and the loop will continue here so; that means, it can have a two different branching again. So, one will be x 2 less than equal to 3, another will be x 2 greater than equal to 4. So that means, again two different sub problems then we look for the again optimal solutions as per the particular you know requirement.

So, now again we add that particular constraint then look for this solution.

(Refer Slide Time: 15:21)



And then this is what the final you know structure. So, this is what the original problem and this is first branching, this is second branching a compared to this ones we like to start with this side. So, we have a two different sub branching since it is infeasible. So, this could be a again you know solve and then the final solution will be like this. So, corresponding to non integers we have again two different sub problems and then we look for the particular you know solution.

And here so, putting this particular additional constraint the final solution will be here like this. So, x 1 equal to 4, x 2 equal to 3 that the where Z equal to 48 and again x 1 equal to 3 and x 2 equal to 4. So, this is a you know where x Z equal to 50. So; that means, original in the original solution we have a Z equal to 51.50 and x 1; x 1 where x 1 equal to 5.25 and x 2 equal to 2.50. So; that means, you compare this one and now the final optimal solutions we have here and we have here. So, now these three will be the mirror image where we can you know compare and then finally, you know fix which one is the best requirement.

So, now if you look into the objective function value, so this is the highest and this is second highest, this is third highest. So, as a result you know we like to pick up a particular you know solution where the value of objective function will be highest and in the same time values of the decision variable will be integer type. So, keeping this two you know requirements; so, this is this can be rejected and again.

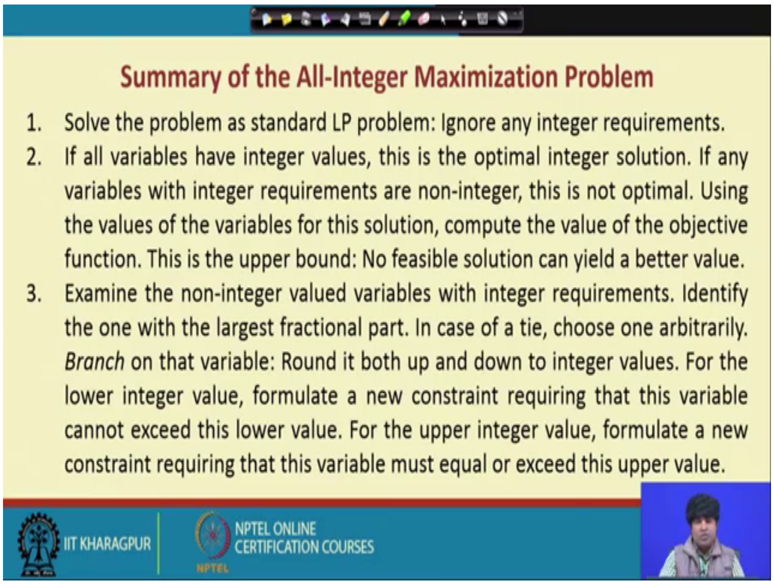
So, this is integer type the values of the decision variable here integer type here the values of the decision variable are integer type so; that means, both can be accepted and in that contest. So, we like to check which one is having high you know Z value. So, that is objective function value; so, in that contest. So, this is highest Z value and where the values of decision variable are also integer type. So that means, technically; so, this is not the optimal case because here the values of decision variable are not integer type.

So, this two can be, but with respect to high Z value. So, finally, the option is the; so, means final the optimal solution will be you know where the Z equal to 50 and x 1 equal to 3 and x 2 equal to 3. So, this is how the kind of typical model and so, these are the kind of structure through which you can take the kind of decisions as per the particular you know business requirement; that means, if you look into this figures.

So, you will find it is a very interesting and in fact, you know it is a kind of continuous process the entire linear programming you know problem is the kind is the structure of iterative process. Again within the iterative process this branch and bound is you know again further you know kind of interesting you know iterative steps and you will continue till you get a kind of final remarks to corresponding to the particular you know business problem, where we have to reach the optimal solutions and the values of the decision variable should be integer type.

So, these are the kind structure through which you can analyze the particular problem.

(Refer Slide Time: 18:50)

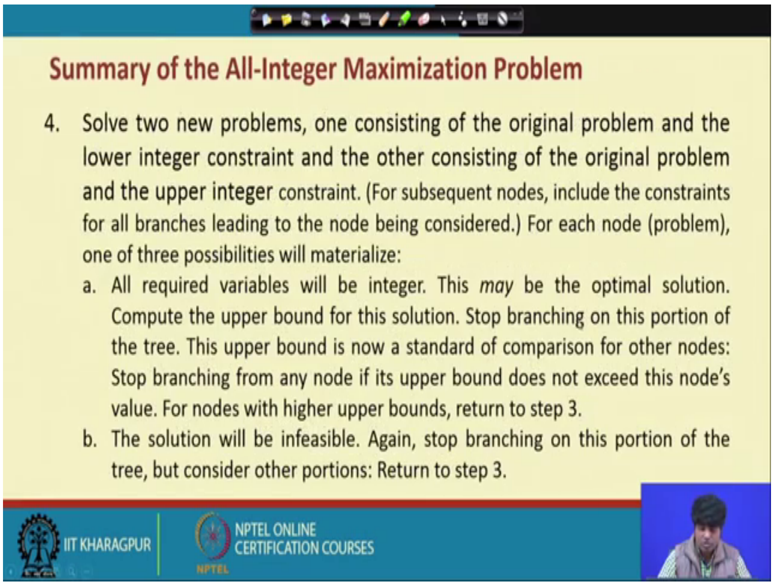


And now; so, if you sum up then you will find a you know you will find there are two different you know steps all together. In the first steps corresponding to original problem use either graphical mechanisms or you know simplex mechanism to get the solution that is the optimal solution and check whether the values of the decision variable are integer type or not. So, if a they are coming integer type then you can stop and no need to go further and analyze the problem as per the particular business requirement, but if the values of the decision variables are not integer type then you have to proceed you know further kind of iterative you know process to get the alternative solution, where the value of the optimum you know optimal solution or the values of the decision variable will be integer type. So, we have discussed you know two different mechanisms and you know both are you know very interesting.

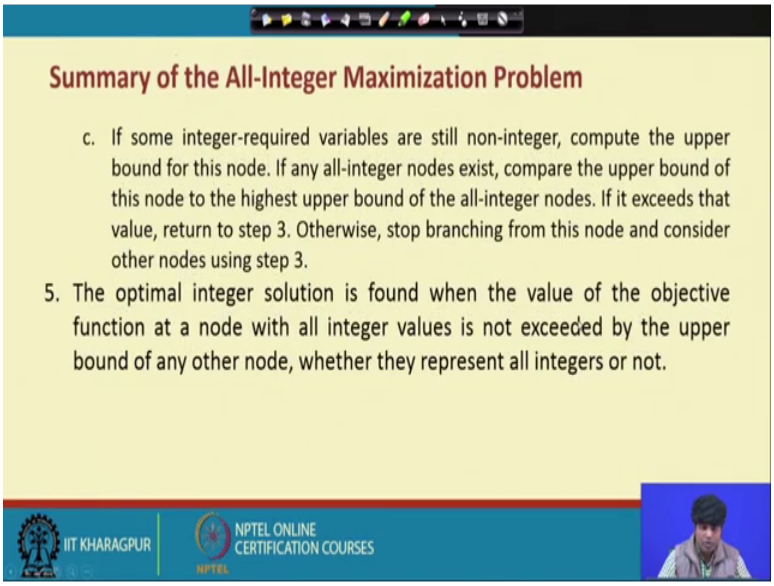
And it will give you some kind of confidence and different alternatives through which you know every stage you will find the improvement with respect to values of the objective functions and corresponding to the constraints and conditions. So that means, you know corresponding to a particular business problem or management problem we have several kind of alternative several kind of structure.

So, it is you to decide how you have to you know fix up a particular you know solutions corresponding to management requirement. And once you know select a particular you know solution corresponding to the management problems. So, you can take the decision effectively; so, this is how the kind of structures.

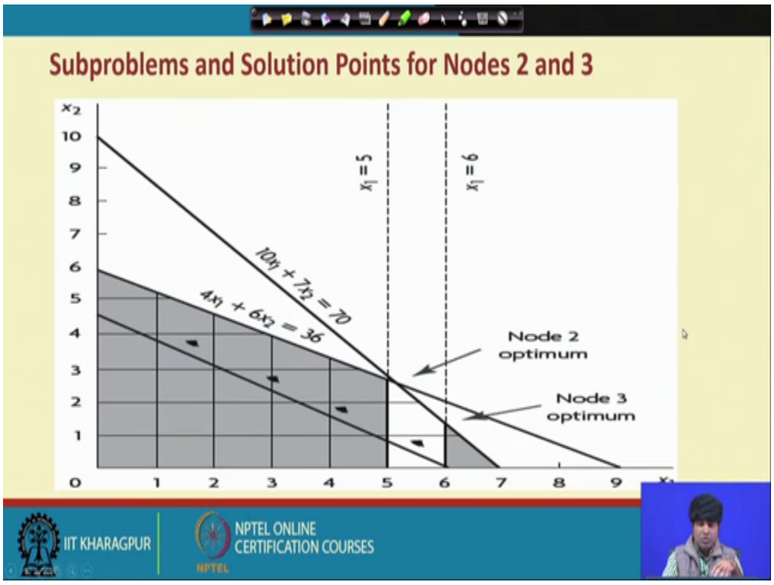
(Refer Slide Time: 20:25)



(Refer Slide Time: 20:27)



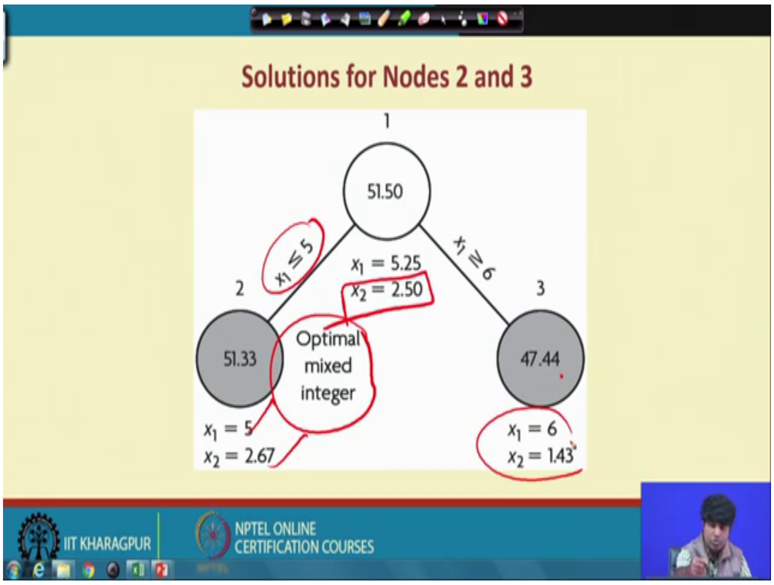
(Refer Slide Time: 20:29)



So, the other aspect of integer type solution is the you may have pure integer type, mixed integer type and then kind of 0 1 integer type. So, which we have already discussed and in the case of mixed integer some are in integer type and some are not necessarily integer type.

So, it depends upon you know the problem to problem and the kind of game to game. So, depending upon our requirement we have to you know operate the process; then we will look for the optimal solution and the values of the decision variable through which you can address the business problem and this is another kind of structure where you know the particular problem is having you know kind of operation.

(Refer Slide Time: 21:13)



So, let us say; so, this is second you know iterative process so; that means, you know in the last solutions which you target is with respect to you know x 2. So, now, we are targeting here x 1s so; that means, in the last particular you know last solutions we have targeted the x 2 branching. So, now you can go through x 1 branching; so, since x 1 is 5.25. So, there are two different constraint again; so, one will be x 1 less than equal to 5 and another one is the x 1 greater than to 6, again we start the process.

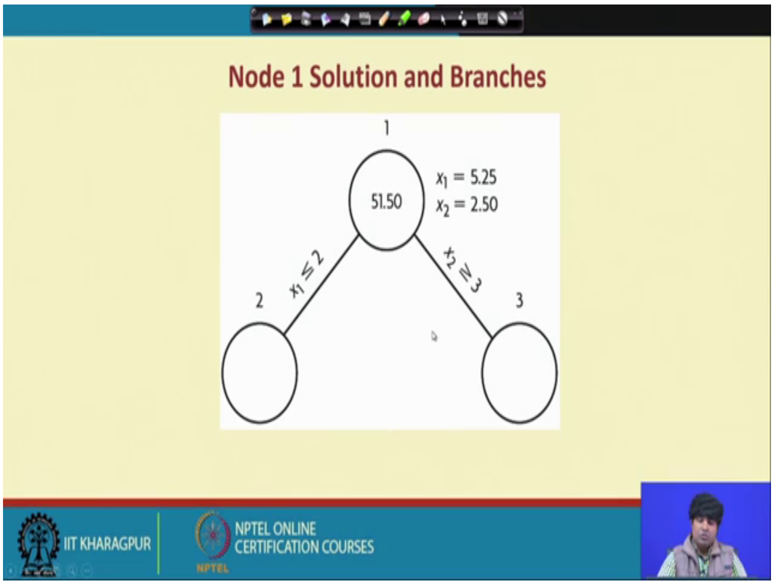
And here you will find a this is actually solutions corresponding to this additional constraints and where x 1 is integer type and x 2 is non integer type and that is what it is called as you know mixed integer programming. And against in this case we have value of the optimize optimum function x 1 equal to 6 and x 2 equal to 1.43 and; that means, technically x 1 is integer type and x 2 is not integer type; it is again you know optimal mixed integer solutions.

So; that means, technically if you if you if you go through the previous structures corresponding to the previous structure; you will find. So, these are all you know kind of mixed integer kind of things. So, they this is this is a mixed integer and this is this is pure integer, this is pure integer, this is mixed integer and this is actually completely non integer. So, we have a different kind of flexibility.

But depending upon the particular business requirement or the kind of problem requirement; you have to fix a if for instance if this is a case where x 2 is integer and x 1 is the non integer. So, you look into the problems if x 1 you know corresponding to x 1s if there is you know such requirement of integer then we can actually fix there because you know that value may be compare you know higher compared to this value.

Of course, we have another solution here where the value of the objective function will be highest higher compared to the previous one. So, as a result this could be the best solutions; so, now, coming through the particular you know structures we will find you know the other part of the problem which is a having two different branching again.

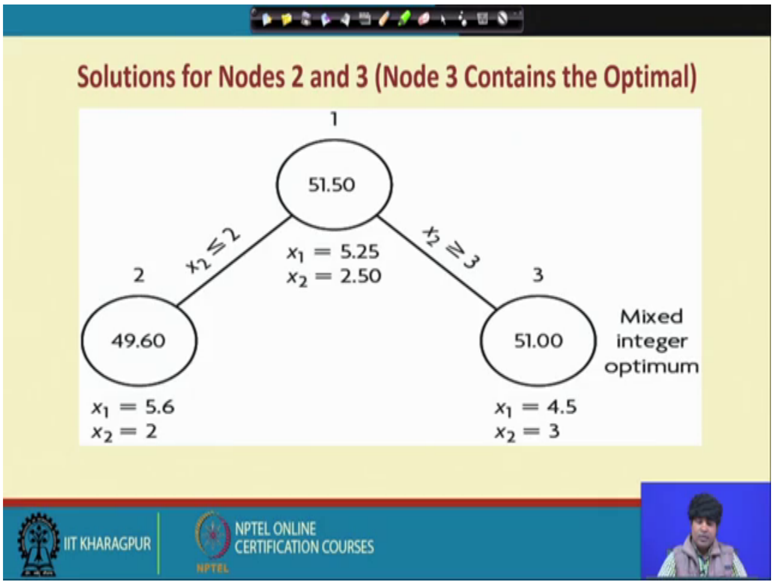
(Refer Slide Time: 23:40)



And then against we look for the solutions and against. So, so here you know we have actually a x 1 equal to 5.25 and x 2 equal to 2.50.

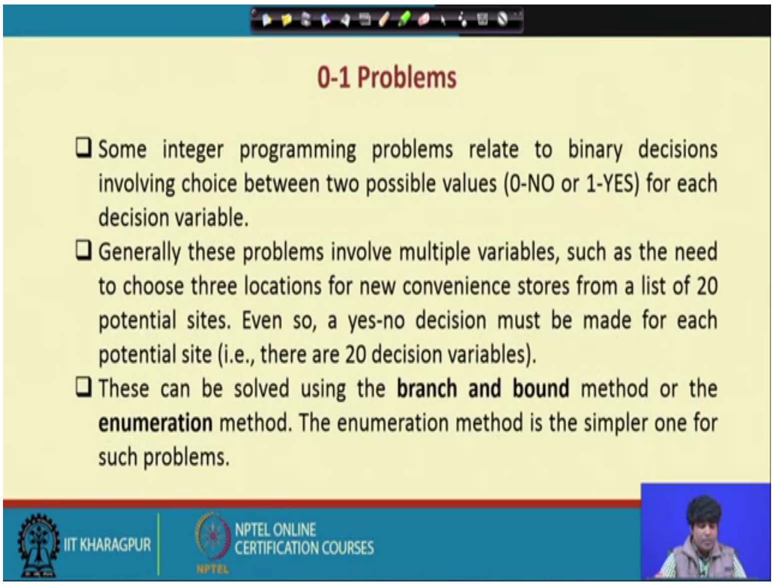
Then x 1 can be actually having different you know divisions one is less than equal to two another is greater than equal to 3.

(Refer Slide Time: 23:58)



Then against we have a solution and this is what actually a called as you know mixed integer solutions.

(Refer Slide Time: 24:05)

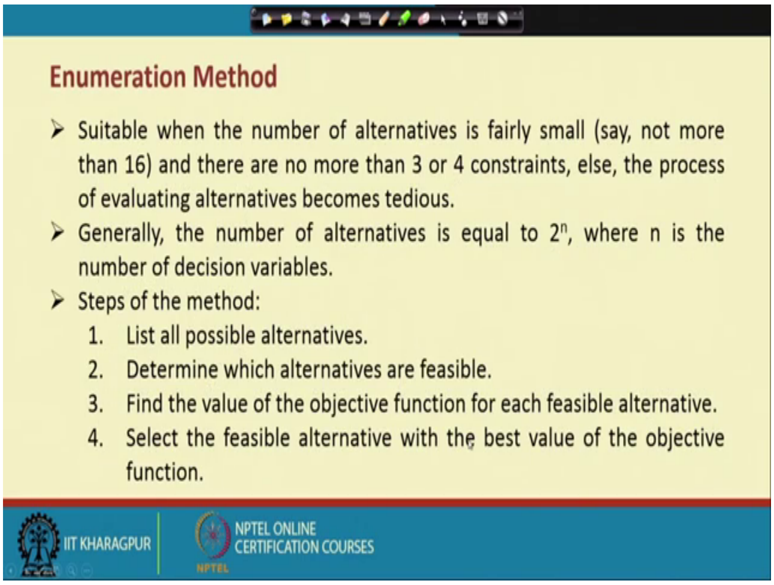


So, now whatever we have discussed till now it is the kind of a integer type solution and you know mixed integer type solutions and then you know kind of completely non integer type of solution; that means, we have a three different situation pure integer, mixed integers and completely a non integers. So, now, we have fourth different kind of structure where the values of the variables decision variables are integer type and that too restrictive on 0 1 only.

So, either 0 or 1 so; that means, it is again strict restriction or strict kind of requirement. And there are certain business problems such kind of requirement is also highly required and then we again go for the branch and bound mechanism to solve the problems. And then look for the optimal solution where the values of the decision variable will be either 0 or 1.

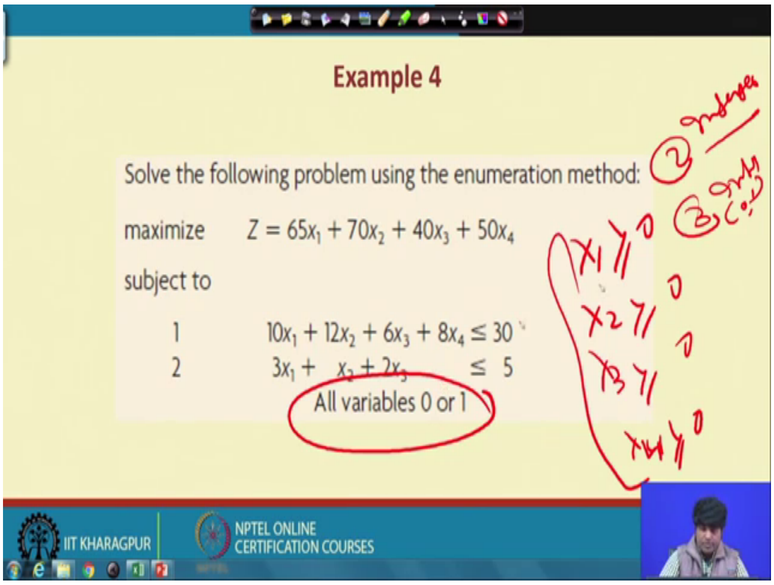
So, like you know we have discussed the concept like binary choice model or linear probability model in the case of you know u kind of predictive analytics.

(Refer Slide Time: 25:09)



So, now this is a mechanism through which actually you will find different kind of constraints and the kind of condition through which you know we look for the various alternatives and the kind of optimal solution. And then look for a kind of solution where the values of the decision variable will be 0 1 type only and that too integer again you know it is a strict kind of requirement either 0 or 1.

(Refer Slide Time: 25:32)

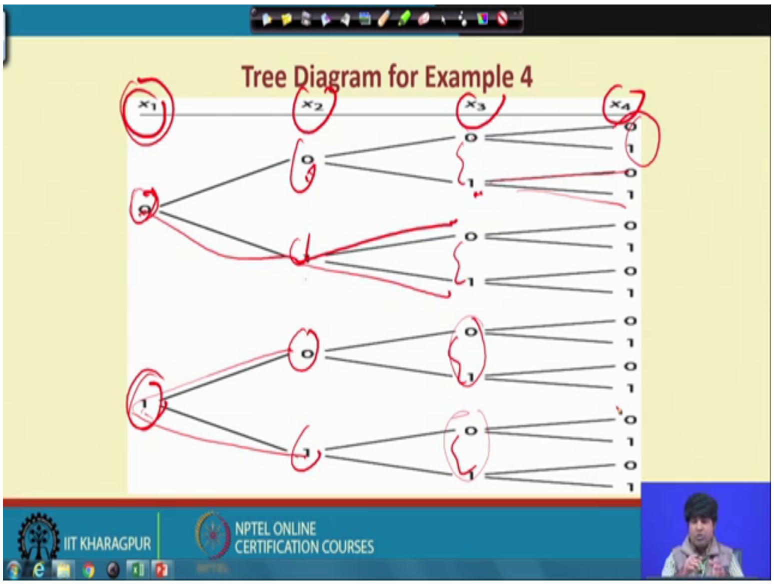


So, let us say this is a problem here and here the problem is with respect to four variables and then we have a two different constraints and all are less than type constraints. So; that means, technically earlier structure is that you know x 1 and x 2. So, we first start with the simple problem that you know x 1 greater than equal to 0, x 2 greater than equal to 0 or you know corresponding to these problem.

So, x 3 greater than equal to 0, x 4 greater than equal to 0 so, this is the first hand conditions and the second hand condition is that you know. So, it must have integer type integer type and now this is second condition then the third requirement is the it is the integer type and against it will be either 0 or 1. So; that means, it is actually one additional constraints we have to put to you know solve the problems and that too as per the particular you know business requirement.

So, this is what you know the restriction we which you have to put here. So, now, corresponding to this we can actually go to the particular you know structure and then look for the solution and here these typical structure will be like this.

(Refer Slide Time: 26:42)



So, let us start with you know the tree diagram and here. So, we have actually four variables and then; so, we start with first x 1 then connect with x 2 then connect with x 3 then connect with x 4 like you know step wise regression. So, it is actually it is a kind of chain and then you know we can you know look for the kind of solution.

So, if you start with x 1s then there are two option only a; so, either 0 or 1. So, there is no other alternative. So, as a results then you know if you connect you know x 2 now x 2 can be either 0 1 either 0 either 0 or 1 so; that means, technically corresponding to x 1 0. So, x 2 can have option 0 1 and a corresponding to x 1 1s. So, x 2 option again have a 0 1 so; that means, technically when we have only x 1. So, we have two different option only; so now, when we have a two variables x 1 and x 2.

So; that means, technically we have a four options. So, either 0 0 or 0 1s or 1 0 or 1 1; so, these are the four options against if we add a x 3 so; that means, corresponding x to x 1 and x 2 for every x 2 where equal to 0. So, it will be 0 1 again then again 0 0 0 1 again 0 1 you know 0 1. So, this is how the case again 1 0 for that you know x 3 will be either 0 1 then 1 1; so, then for you know x 3 0 1.

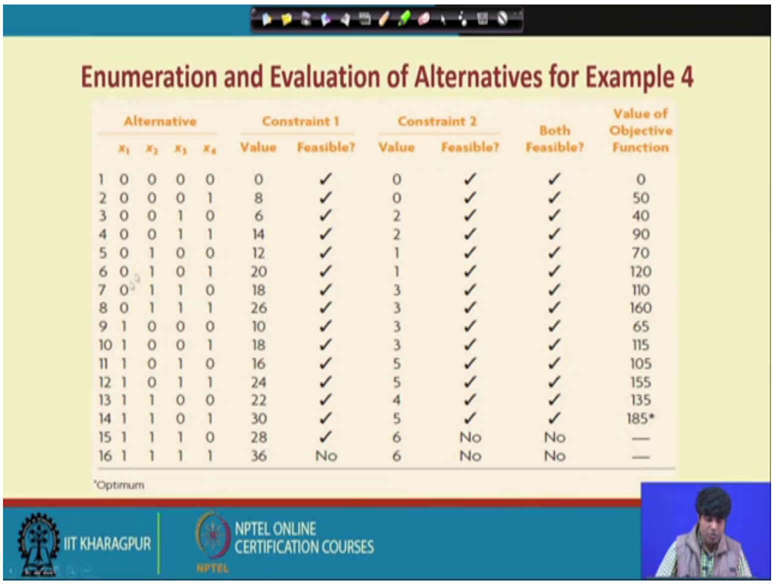
So; that means, corresponding to x 1 and x 2; so, now, we have we have two different option here two different option here two different option here two different option here so; that means, so we have actually eight options now.

So, now, again if you add x 4; so, then again corresponding x 1 x 2 and x 3 that to 0. So, we have a two different options for x 4; again corresponding to x 1 x 2 0 and x 3 1. So, we have a 0 1 option. So; that means, technically; so, that is how you know it is a kind of a similarly like you know branching, but it is more interesting like you know with you know 0 1 clustering.

So; that means, we have plenty of flexibility or plenty of options through which you can address the business problem. And then come with a kind of decision through which you can you know give a some kind of decision and that too be more effective.

So, corresponding to this particular structure; so, we can actually summarize.

(Refer Slide Time: 29:16)


Enumeration and Evaluation of Alternatives for Example 4

	Alternative				Constraint 1		Constraint 2		Both	Value of Objective
	x_1	x_2	x_3	x_4	Value	Feasible?	Value	Feasible?	Feasible?	Function
1	0	0	0	0	0	✓	0	✓	✓	0
2	0	0	0	1	8	✓	0	✓	✓	50
3	0	0	1	0	6	✓	2	✓	✓	40
4	0	0	1	1	14	✓	2	✓	✓	90
5	0	1	0	0	12	✓	1	✓	✓	70
6	0	1	0	1	20	✓	1	✓	✓	120
7	0	1	1	0	18	✓	3	✓	✓	110
8	0	1	1	1	26	✓	3	✓	✓	160
9	1	0	0	0	10	✓	3	✓	✓	65
10	1	0	0	1	18	✓	3	✓	✓	115
11	1	0	1	0	16	✓	5	✓	✓	105
12	1	0	1	1	24	✓	5	✓	✓	155
13	1	1	0	0	22	✓	4	✓	✓	135
14	1	1	0	1	30	✓	5	✓	✓	185*
15	1	1	1	0	28	✓	6	No	No	—
16	1	1	1	1	36	No	6	No	No	—

*Optimum

And here; so, these are all 16 different case corresponding to the particular you know particular you know business problem. So, that too that too the 0 1 integer programming; so, we have a four variables. So, the way which we have already highlighted here the kind of chain that can be summarized here. That means, we have all together 16 different cases if you go through here only. So, these are all the chains.

And then we will find here these are all 16 different cases corresponding to x 1 x 2 and x 3 and different options we have here. So, now, if you actually you know prepare a tables and summarize this results. So, this could be this could be like this and here; so, the first case all are you know 0. So, that is the case here and that is the case here 0 0 0 0 ok. So, this is the 0 x 1, 0 x 2, 0 x 3, 0 x 4 so; that means, typically. So, this case this case this case and this case; so, this one particular cluster; again 0 0 0 1 and then again 0; so, this is another clusters.

Like we will have a different kind of clusters again 0 0 0 and 1; so, this is another cluster. So, like that we will have a different kind of cluster depending upon the particular you know variables involvement. So, now, the summary sheet represents like this; so, we have actually 0 0 0 and that too the value will be 0 because all variables are you know 0. So, again; so, this is a feasible and then constraints this is also feasible and this is both are feasible and value of the objective function will be 0.

So, now, second options 0 0 0 1 and then x 4 equal to 1 so; that means, corresponding to the objective function. So, the then the constraints; so, it will be appearing 8 and this is actually value 0 and this is both are you know feasible and then finally, it will coming 50. So, likewise we have a different you know options and a every options we have a value of the objective functions. And then finally, a we will fix a particular solution we which can give you the best you know optimal value with the condition that you know the values of the decision variable are integer type positive and then 0 1 only.

So; that means, it is actually you know kind of very very restricted kind of game or you know very specific games and where you cannot apply just a each and every you know problems or you know business kind of fields. So, it is a case specific problem specifics where we have a strict three conditions that to non negativity, integer type and then the values of the decision variable will be only binary in nature that too 0 1. So, that is how it is called as you know otherwise called as you know 0 1 integer programming.

So, so; that means, technically if you if you sum up then we have discussed couple of problems here and some problems connecting to primal structure, dual structures then we have discussed with graphical structures to solve the particular problem. Then we have discussed simplex mechanism to solve the problem depending upon the length of the problems.

Then again we have discussed dual and dual simplex methods and again within the particular structure we have discussed integer kind of programmings, where corresponding to a particular problem whether it is a bivariate structure or multivariate structure; where the optimality requirement is you know not to address the business problem more efficiently.

But in the same times the values of the decision variable should be integer type. And the third option is the optimal solutions then the values of the decision variable integer type and then that too they have they are all in 0 1 interval only. So; that means,. So, we have a discussed a case here with respect to four variables and that too x 1, x 2, x 3, x 4. So, as a result we have you know 16 different cases; so, typically the number of cases in such a situation is the nothing, but you know 2 to the power n.

And so, accordingly if you know put you know another variables x 5 then the number of cases will be 35; so, 32 sorry the. So, likewise a if you put you know another a x then it will be it will be again 2 to the power 6. So, likewise you know you will find you know different you know cases and different flexibility and as a result we look for you know different values of the objective functions. And then finally, you fix the particular you know situation where the value of the objective function will be high.

And the particular problem is optimal will address the business problem effectively and where the values of the decision variable will be integer and that to either in 0 1 you know structure. So; that means, the corresponding to this problems x 1, x 2, x 3, x 4. So, the minimum entry will be x 1 0, x 2 0, x 3 0, x 4 0 and the maximum entry will be x x 1 equal to 1 x 1 x 2 equal to 1 x 3 equal to 1 and x 4 equal to 1 and you know corresponding to this one.

So, sometimes you may not get actually you know not feasible solution sometimes you get you know feasible solution. And this is equally true in the case of simple you know integer programming where we are putting you know branch and bound mechanism. So, while doing so, so many branches; so, you will find in a particular case; so, the particular solution is not you know feasible one.

For instance we have a discussion we have a structure here like this is the case here yes this is the case here. So, while doing the kind of branching and we have different options. And in all cases this is one branching, this is one branching corresponding to original problem, this is another branching this is another branching, this is another branching, this is also another branching.

But every case there is optimal solution, optimal solution, optimal solution, optimal solution, optimal solution and in some cases mixed integer some case some cases it is a pure integer, but here there is a case you know it is a it is a sub problems and it is through branching, but it does not give any feasible solution. So, likewise this is also case in the case of you know 0 1 integer programming and so, we have a number of cases.

So, that likewise this is a branching; so, if you apply you know 0 1 game. So, number of cases corresponding to the number of variables and then you will find all such conditions. And then look for the optimal solution or that is with respect to value of the value of the objective function. And that strictly depends upon the objective function structures then you know just put the values of the variables and look for the value of the objective function. And then fix whether you know you will reach the optimality and then you know address the business problem effectively.

So, this is how the kind of requirement and so, we have solved some of the problems in this contest, where you know the optimal solution requirement is the integer type. And that too it is a problem specific, business specific and case specific and it the strict warning is that you know it cannot be you know apply each and every case if there is you such requirement.

So with this, we will stop here.

Thank you very much.