Fundamentals of Operations Research

Prof. G. Srinivasan

Department of Management Studies

Indian Institute of Technology Madras

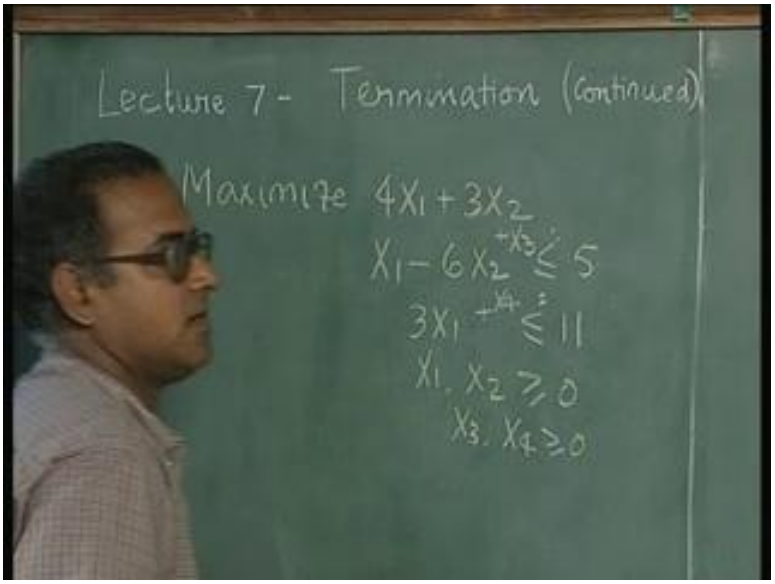
Lecture No # 7

Simplex Algorithm Termination

(Refer Slide Time: 01:1)

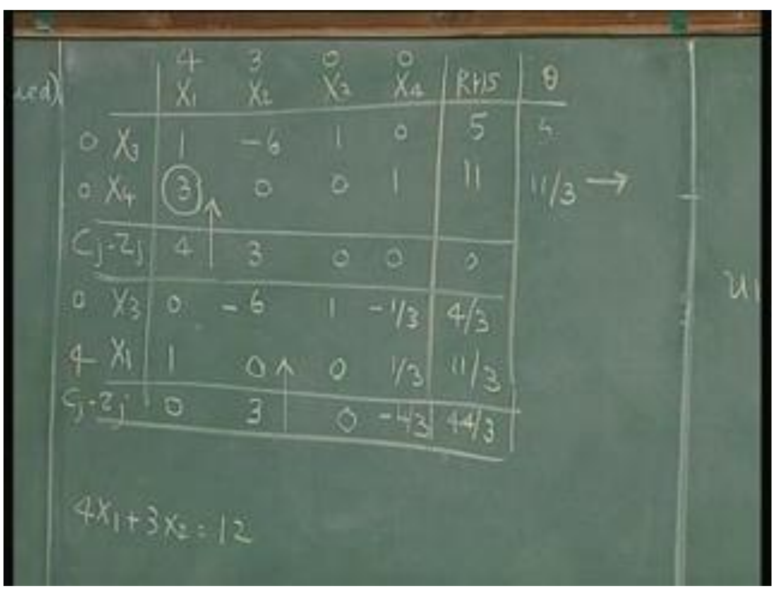
We continue our lecture on Termination. We look at two more aspects of terminations which are Unboundedness and Invisibility. We look at unboundedness with an example.

(Refer Slide Time: 01:33)



We take this example. Maximize $4X_1 + 3X_2$ subject to $X_1 – 6X_2$ less than or equal to 5; $3X_1$ less than or equal to 11; X_1, X_2 greater than are equal to 0. As usual we add the two slack variables + X_3 = 5; + X_4 = 11; X_3, X_4 greater than are equal to 0 and we create the simplex table.

(Refer Slide Time: 02:11)

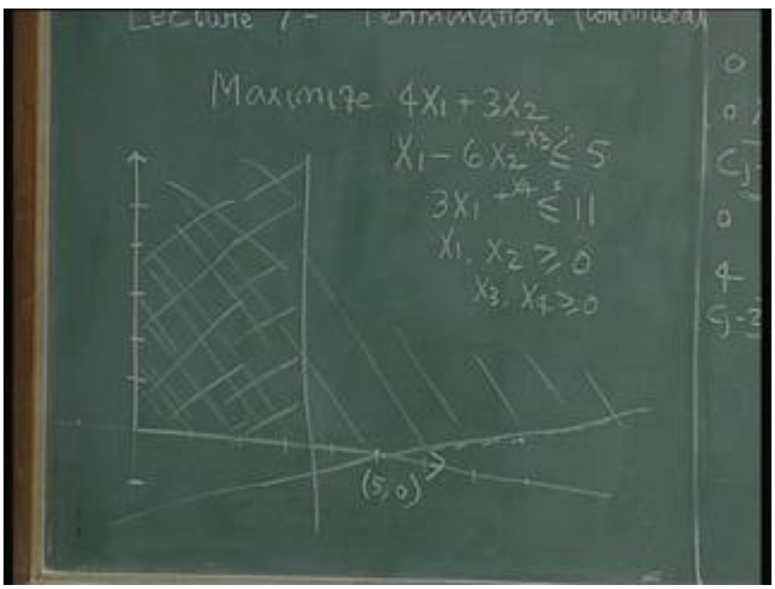


With X_3 and X_4 as starting basic variables, X_1, X_2, X_3 and X_4 right hand side. Start with X_3 and X_4; $4X_1 + 3X_2$ 0 0 these values are 0 and 0. 1 – 6 1 0 5 3 0 0 1 11 $C_j – Z_j$. These two are 0 also X_3 and X_4 are basic so they will have $C_j – Z_j$ or 0.
So 0 into 1 + (0 into 3) is = 0 and we get 4 and 3 respectively. This will become 0. Now variable X_1 with the largest value of $C_j – Z_j$ enters. X_1 enters. To find out the leaving variable we compute

theta. 5 divided by 1 is = 5 and 11 divided by 3 is 11/3. Now 11/3 is smaller than 5 so variable X_4 leaves the basis, X_1 enters and this is the pivot element. So now we have X_3 and X_1 replacing X_4. Now divide by the pivot element to get (1 0 0 1/3 11/3). This (Refer Slide Time: 4:11) is 4 and this is 0. We need a 0 here so this minus this would give 0 so 1 – 1 = 0; – 6 – 0 is = – 6; 1 – 0 is = 1; 0 – 1/3 is = – 1/3; 5 – 11/3 is = 4/3.
$C_j - Z_j$, X_1 and X_3 are basic variable so we get 0. 0 into (– 6 + 4) into 0 = 0; 3 – 0 is = 3; 0 into (– 1/3 + 4) into 1/3 is = 4/3 you get a – 4/3. 0 into 4/3 + (4 into 11/3) is = 44/3.

Now variable X_2 with a positive value of $C_j - Z_j$ enters. You have to find out a leaving variable. We try and compute theta. You realize that this is negative. Therefore you cannot compute this theta. This is 0, you cannot compute this theta. There is no theta. So the algorithm will terminate because the algorithm is unable to find a leaving variable. It is able to find an entering variable but it is not able to find a leaving variable. So the algorithm will also terminate. This phenomenon where the algorithm terminates when it is unable to find the leaving variable is called unboundedness. What exactly is this unboundedness? We will see by drawing the graph corresponding to this problem.

(Refer Slide Time: 05:44)



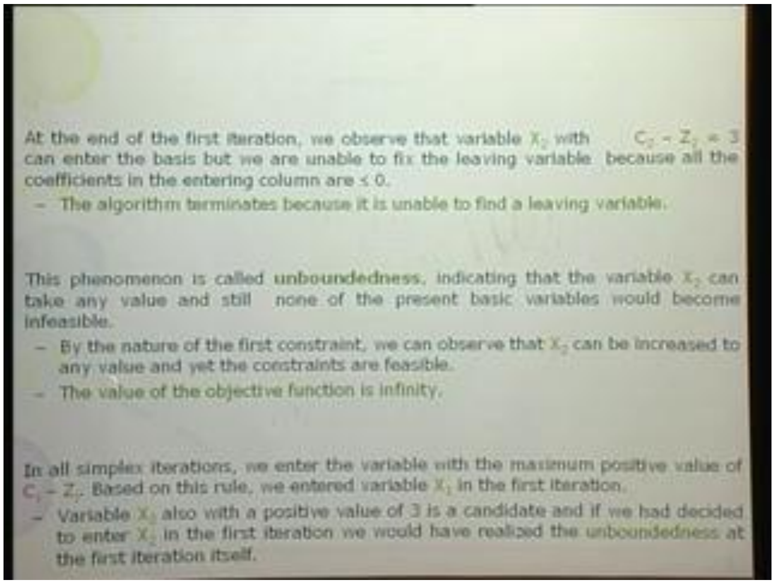
This is (5, 0) one of the points. We need to draw $X_1 - 6X_2 = 5$. So if we put $X_2 = -1$ here then X_1 will also be – 1 so this will be the point. So we have another point. One point is (5, 0) the other point would be for example, if X_2 is 1/2 then X_1 will be 8. (6 7 8 and1/2) so the line is somewhat like this. This is the line $X_1 - 6X_2 = 5$. This divides it into two regions. So (0, 0) is less than this. This is the feasible region corresponding to this constraint. Now $3X_1$ less than or equal to 11 would give us a point $X_1 = 11/3$ so, (1 2 3 4), roughly, this is 11/3 so this is, 1 less than or equal to 11/3 is all this Refer Slide Time: 7:27). So the region that satisfies all the constraints is actually this region which goes right up to infinity and we have this. Therefore the first important thing is that this region is not bounded. So this set of constraints represents an unbounded region in the first place and then an unbounded solution.

Now what is the difference between the unbounded region and the unbounded solution? Let us also draw the objective function. Suppose we draw $4X_1 + 3X_2 = 12$, we have three points (3, 0) (0, 4), this is the Objective function line. As we maximize we realize that the objective function is moving in this direction and it will not get the optimum point because the region is unbound. Both X_1 and X_2 can go up to infinity. On the other hand if the problem happened to minimize $4X_1 + 3X_2$ subject to same constraints, the objective function line is now going to move in this direction and then it will terminate by giving the point (0,0) as the optimum point.

So there are two aspects to it. One is that the feasible region can be unbounded so the problem has an unbounded region and it can have an unbounded solution if the objective function is moving in this certain direction. So this is an example where we have both an unbounded feasible solution as well as unbounded region. Depending on the objective function, sometimes a region that is unbounded may still have an optimum. In this example we are seeing a situation where the feasible region is unbounded as well as the solution is unbounded. So unboundedness is indicated by simplex algorithm being able to identify an entering variable and not being able to identify a leaving variable. In fact we can do one more thing which is this. If we had looked at the very first iteration now, we see both the non basic variables X_1 and X_2 having positive value. So far we have been very consistent in trying to enter that non basic variable with the largest positive $C_j – Z_j$. By now we can understand that any variable with a positive $C_j – Z_j$ can enter and can improve the objective function. So instead of entering this 4 if we had tried to enter the variable X_2 with 3 in the very first stage we wouldn't be able to find out the leaving variable and therefore we could have indicated unboundedness right here.

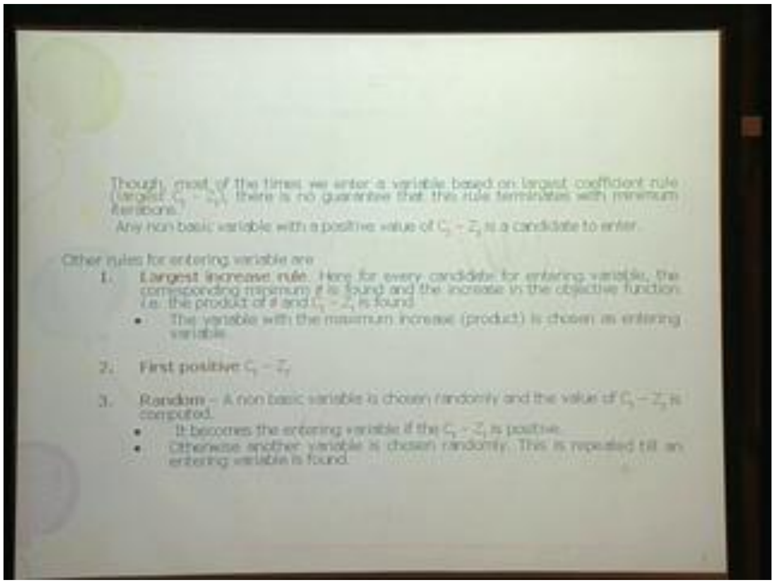
What we are actually trying to do is we could have seen the unboundedness right here but then from this point we move to another point here and then we realize that we are going to get unbounded region. So it is not absolutely necessary that we always need to enter the variable to the largest positive $C_j – Z_j$. We could enter any variable with a positive $C_j – Z_j$ in which case for this example, we could have detected unboundedness earlier than what we arrange. Now let us go back to this. So right here end of the first iteration we observed.

(Refer Slide Time: 10:52)



The variable X_2, $C_2 – Z_2 = 3$ can enter the basis. But we are unable to fix a leaving variable. Algorithm terminates. It is unable to find the leaving variable. Such a thing is called unboundedness, indicating that variable X_2 can take any value and still none of the present basic variable should become infeasible. Now the value of the objective function is infinity in this case. Now as I had explained in all simplex iterations it is customary to enter the variable to the maximum possible value of $C_j – Z_j$ based on which we entered X_1 in the first iteration but if we had tried to enter this we could have still found the unboundedness at the first iteration. This is also interesting that most of times we enter a variable based on the largest coefficient rule which is called the largest $C_j – Z_j$ rule. There could be other rules which you could use to enter variables.

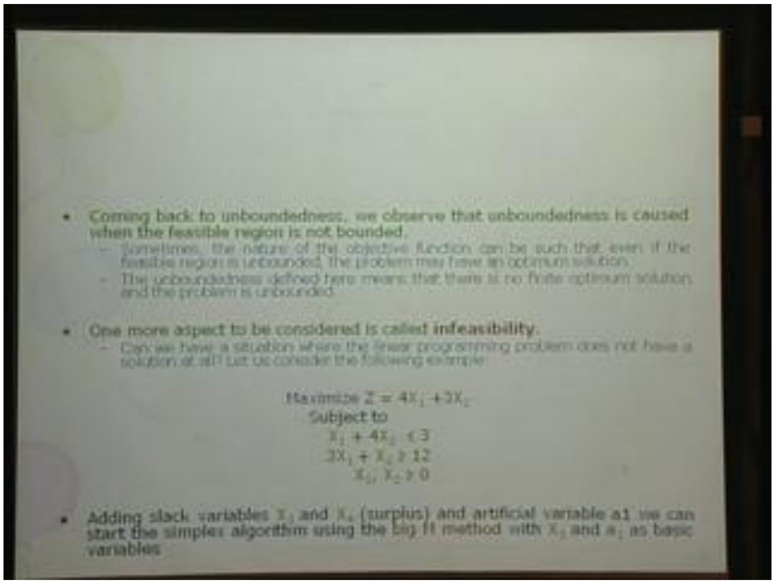
(Refer Slide Time: 11:49)



There is something called the largest increase rule. For example for every candidate that enters you can find out the corresponding theta. We know that the increase in the objective function is the product of the $C_j - Z_j$ and theta. So for every positive value of $C_j - Z_j$ we can go back and find out the corresponding theta and multiply the $C_j - Z_j$ and theta to find out the increase. We could then choose the variable which gives a largest increase in the objective function rather than the largest coefficient. So the first rule is called the largest coefficient rule. The $C_j - Z_j$ rule. We could think of something called the largest increase rule as well. We can think of something like the first positive $C_j - Z_j$ enters. Now these things become important when we are solving large size linear programming problems.

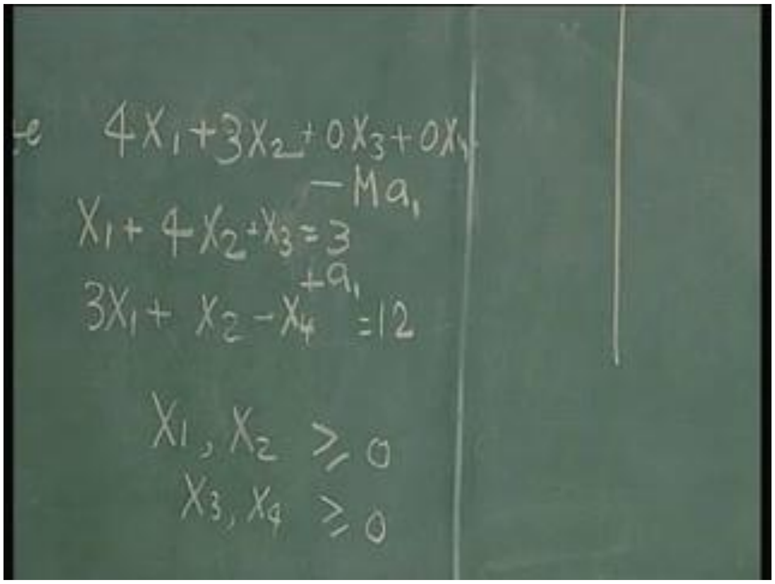
It may not be necessary to evaluate a large number of $C_j - Z_j$'s to find out the maximum hence one could just enter the first positive $C_j - Z_j$. One could also think of some kind of a random allocation. Pick a non basic variable randomly, compute it $C_j - Z_j$. If it is positive, enter it or keep doing this till you get the first randomly picked non basic variable with the positive $C_j - Z_j$. So these things are also available. It is also interesting to note that none of these rules consistently are able to outperform the others in terms of number of iterations. So far based on many experiments and researches that have been conducted, the largest coefficient rule that we commonly use seems to work better than the rest of the others.

(Refer Slide Time: 13:33)



Coming back to unboundedness we observe that their unboundedness is caused when the feasible region is not bounded as shown in this example. Sometimes the nature of the objective function can be such that even if the feasible region is unbounded the problem may have an optimum solution which was explained by a minimization function for the same example. Another aspect of termination is there is something called infeasibility which we will again see in another example.

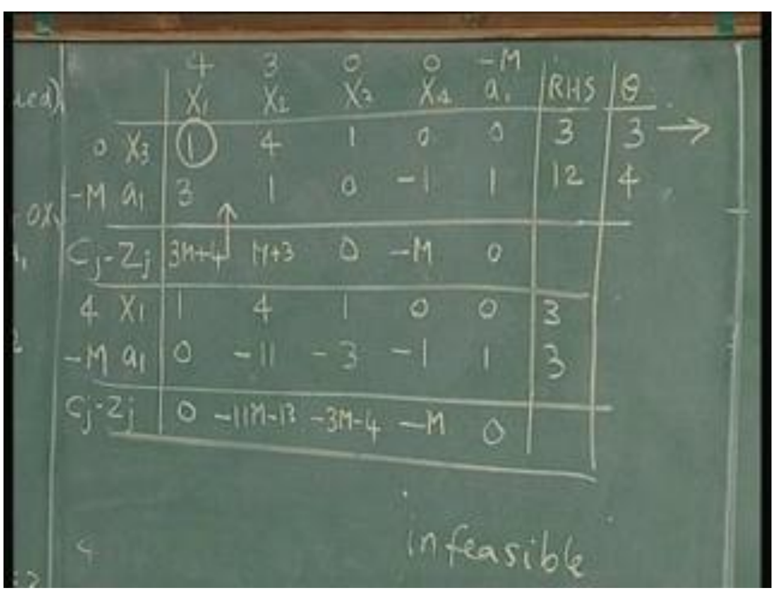
(Refer Slide Time: 14:05)



Now let us consider this example for infeasibility. Maximize $4X_1 + 3X_2$; $X_1 + 4X_2$ less than or equal to 3; $3X_1 + X_2$ greater than or equal to 12. X_1 X_2 greater than or equal to 0. Now we add a positive slack variable X_3 here to convert this to an equation. This is a greater than or equal to type inequality, so we add a negative slack $– X_4 = 12$; X_3, X_4 greater than or equal to 0

Now X_3 will qualify to be a basic variable and X_4 will not. Therefore we need to add an artificial variable so we add $+ a_1 = 12$. We now try to solve this problem using the big M method.

(Refer Slide Time: 15:22)



It is a maximization problem. So we have $+ 0X_3 + 0X_4 – Ma_1$. So we get a $– M$ here. We now start with X_3 and a_1. So, a_1 has a minus here; X_3 has a 0; $X_1 + 4X_2 + X_3 + 0X_4$ 3;

$3X_1 + X_2$; 0; – X_4; + a_1 = 12; $C_j – Z_j$. 0 into 1 – (M into 3) is = – 3M; 4 – (– 3 M) is = 4 + 3M So this is 3M + 4, 0 into 4 – (M into 1) is = – M; 3 – (– M) is = M + 3; 0; 0 into 0, – M into – 1 is = + M; So – M and a_1 has 0.

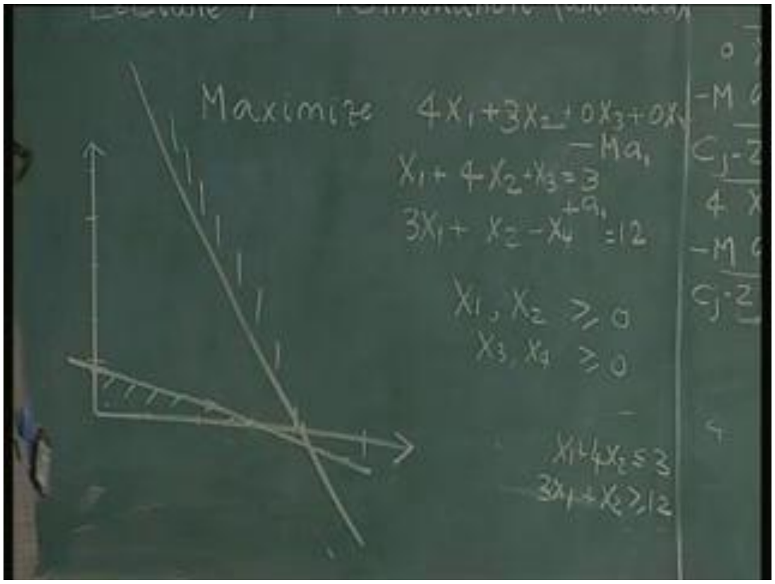
Now between these two 3M + 4 is bigger than M + 3 because M is large and positive so once again variable X_1 with the larger $C_j – Z_j$ will enter. Now we need to find theta.

This is a right hand side value. 3 divided by 1 is = 3; 12 divided by 3 is = 4. Being a smaller variable, X_3 leaves the basis. This is the pivot element so variable X_1 would replace X_3 in the basis. So we have X_1 and a_1; X_1 has 4 and a_1 has – M. Now divide by the pivot element to get (1 4 1 0 0 3). We need a 0 here. So this – 3 times this would give a 0. 3 – (3 times 1) is = 0; 1 – (3 into 4 = 12) is – 11; 0 – (1 into 3 is) = – 3; – 1 – (3 into 0) is = – 1; 1 – (3 into 0) is = 1; 12 – (3 into 3 = 9) is = 3

Now $C_j – Z_j$, variables X_1 and a_1 are now basic so we will get a 0 here 16 + 11M; 3 – 16 – 11M. So this is – 11M – 13. This is 4 + 3M. So 0 – 4 – 3M, so – 3 M – 4; 4 into 0 = 0; – M into – 1 is = + M; 0 – (+ M) is = – M. 4 into 3 = 12 but we have an artificial variable here and we still do not write the value here. Now let us go back and check whether there is an entering variable in this case. All the non basic variables X_2, X_3, and X_4 now are clearly negative because M is large and positive. They all have negative coefficients forever so the algorithm terminates. Algorithm terminates because it is not able to find an entering variable. When the algorithm terminates what we have found is that the artificial variable has still not left the basis. It continues to hang on and stay in the basis. Now what does it mean? We said when we introduce an artificial variable we are introducing it for the purpose of getting a basic feasible solution. We also said that the artificial variable should not lie in the solution because if an optimal solution exists, then the value of the objective function corresponding to that solution will always be for a maximization problem higher than any basic feasible solution that involves an artificial variable.

Now what happens is that, the optimum solution is found but the artificial variable is still hanging on. This means that the problem does not have an optimal solution because if the problem had an optimal solution without the artificial variable (because the artificial variable is not part of the original problem) then obviously that solution will be without the artificial variables and that will definitely have a higher than any basic feasible solution which has an artificial variable. Since we have the optimal containing the artificial variable, it goes back to show that the problem does not have an optimum solution or to put it more in general terms, the problem does not have even a single basic feasible solution. If the problem had a single basic feasible solution, then that would not have an artificial variable then that would have had an objective function value higher than any solution like this which has an artificial variable. Therefore this indicates that the problem is infeasible. Problem is not feasible or the problem is infeasible. Now let us see how this problem is infeasible by drawing the graph corresponding to these constraints.

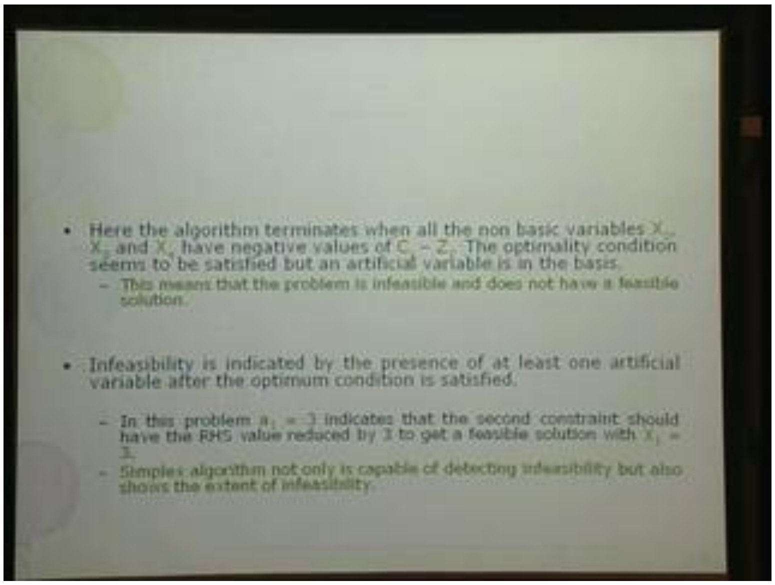
(Refer Slide Time: 21:24)



So the graph will be like this. $X_1 + 4X_2$ less than or equal to 3, so 1, 2, 3, 4, and 5 and the two constraints are $X_1 + 4 X_2$ less than or equal to 3; $3X_1 + X_2$ greater than or equal to 12.
So $X_1 + 4X_2$ less than or equal to 3 would give (3, 0) and (0, 3/4). This will be the line. $3X_1 + X_2$ greater than or equal to 12 is (4, 0), here and (0, 12) which is somewhere else. So you will find a line coming from somewhere here and this side up.
So you realize that there is no feasibility. Reason one being, constraint is moving in this direction the other constraint is moving in the other so there is no feasible region at all and that is reflected by this. Now this is the situation were the simplex is able to show that the linear programming problem may not have an optimal solution at all. If the problem does not have a feasible region then obviously it cannot have an optimal solution. So simplex is able to detect if a given linear programming problem does not have a feasible region. That is done by the artificial variable continuing to remain in the basis even after the optimality is met. Simplex also does something interesting. It not only says that this is infeasible but indirectly it also tries to give something like an extent of infeasibility of this problem. Now what does this a_1 = to 3 indicate?

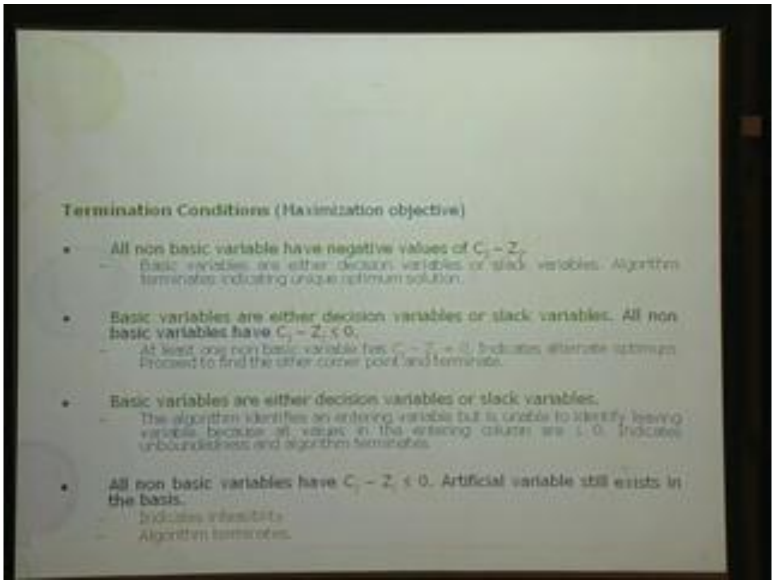
Now let us go back to this constraint. This is a constraint which had a_1. So if for example this constraint had been $3X_1 + X_2$ greater than or equal to 9 instead of 12 then what would have happened is the second constraint would have moved a little bit here and would have exactly touched this point. So as the bare minimum, you need to move one of the constraints so that you get at least one feasible point which is indicated by the value that the artificial variable takes at the optimum. It also tells you how much these right hand side has to be adjusted so that you get at least one feasible point which would have become optimum. So simplex is not only capable of detecting infeasibility it also gives a clue as to what should be done to make it feasible. When we apply this to practical situations, if a practical problem formulated as a linear programming problem indicates infeasibility, the first thing we need to do is to check whether the right hand side values are entered properly or estimated properly. So this is going to give us a clue that something may have been wrong in the estimation of this 12, as a result of which it would have become infeasible. Simplex normally does not look at this, it only gives a feel about the artificial value for all that may be possible. If this constraint had been for example, 6 or 7 or some large number then we may still have got a feasible solution. Simplex does not look at that part, it will look only at the artificial variable part and gives part wise answers to the fact that this system is infeasible. So these are the basic four aspects of termination which we have seen. The four aspects that we have seen are Alternative optimum, Unboundedness, Infeasibility and then one more which is cycling that we need to see.

(Refer Slide Time: 25:52)



Let us go back to this. Simplex not only is capable of detecting infeasibility but also shows the extent of infeasibility. This $a_1 = 3$ forces us to make this 12 as 9 so that we could get a solution.

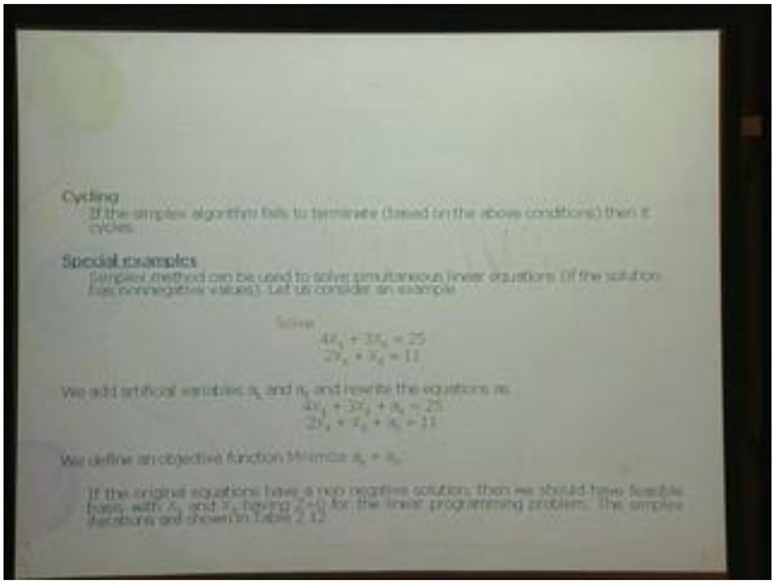
(Refer Slide Time: 26:04)



Now what are the termination conditions? Basically for maximization problems there are these termination conditions. All non basic variables have negative value, $C_j - Z_j$. That is the ideal situation that represents a unique optimum and simplex will terminate giving a unique optimum and in this case, this is the termination condition with which we started. The basic variables are either decision variables or slack variables. Algorithm terminates indicating a unique optimum solution. Second condition is basic variables are either decision variables or slack variables. All non/slack variables $C_j - Z_j$ less than or equal to 0 but at least one non basic variable as $C_j - Z_j = 0$ indicates alternate optimum.

We need to proceed to find the other corner point which simplex is able to detect and terminate and then understand that simplex, being capable of looking only at corner points has given us only two alternate optima but there exists many numbers of alternate optima. Any line any point in joining these two would indicate optimum for a 2/2 problem. This is alternate optimum. Then what is unboundedness? Once again algorithm identifies an entering variable but it is unable to identify a leaving variable because all values in the entering column are less than or equal to 0. This indicates unboundedness and the algorithm terminates. The last one would be all non basic variables have $C_j - Z_j$ less than or equal to 0. Artificial variables still existing in the basis indicates infeasibility and the algorithm terminates. So these are the four termination condition that we have for linear programming problems.

(Refer Slide Time: 27:46)

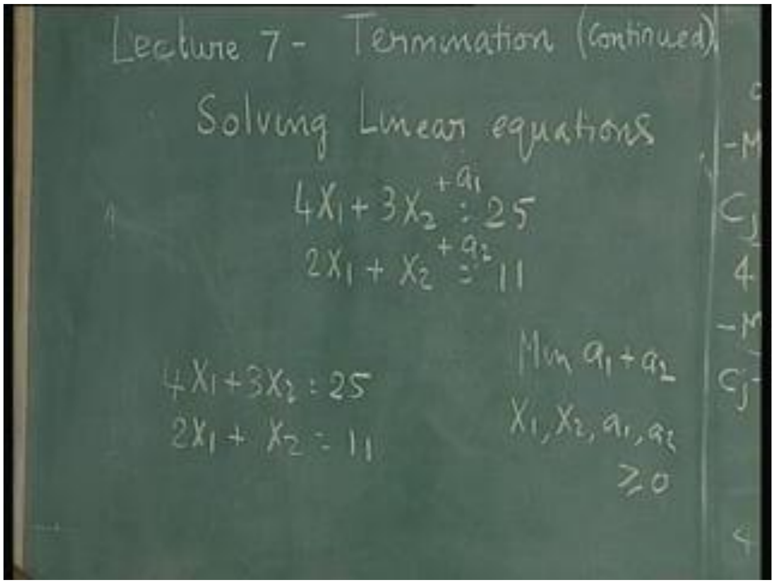


The last aspect is if the algorithm still does not terminate then something can happen and that is called cycling. If a simplex algorithm fails to terminate based on the above conditions then it will cycle. We are not going to explain cycling through an example here. But I will tell you what cycling is all about. In all the simplex iterations that we have seen till now, we have seen that every iteration is characterized by a set of basic variables and more importantly a unique set of basic variables. So far in no simplex iteration did we go back to the same set of basic variables. The only time we came very close to doing it was when we had an alternate optimum and then we said we move to the other solution and if we had iterated we could have come back to one more solution. Now is what is called cycling.

Cycling is a phenomenon in which you are in the middle of simplex iteration with certain set of basic variables. You go through some 3 or 4 or certain number of iterations and you have realized that you have come back to the same set that was there earlier and you are not satisfying any of the termination conditions. Such a thing is called cycling. Cycling is a very rare phenomenon. We do not have too many reported instances or problems that cycle. In fact so far there has been no report of a practical problem or a linear programming problem formulated out of a practical situation which cycles. There have been reports where people have said that they have encountered problems that cycle but we still do not have an example taken from a practical situation which cycles.

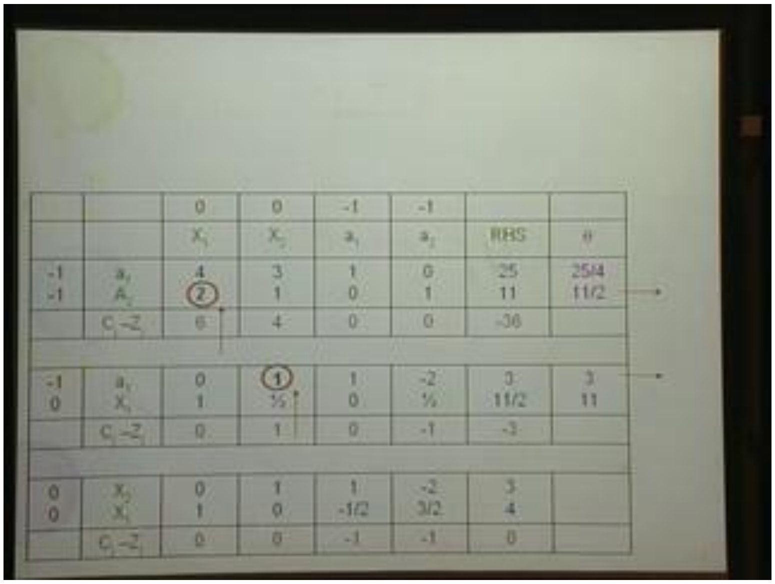
There are a few examples that are available in the books which show the cycling phenomenon. But cycling is not a very common phenomenon. There are also some interesting results like for the problem to cycle it should have at least three constraints, six variables and so on but we will not go deeper into cycling in this part of the operation research course. So we would enter discussion on linear programming with the termination conditions for linear programming, with these four conditions, plus cycling and say that if the linear programming problem does not terminate then it should cycled. We have seen all the three aspects of the simplex algorithms, Initialization, Iteration and Termination. We will now look at a couple of more examples to see some other things that simplex can do other then solving a linear programming problem. We will take some examples to do each one of them. Now the first one is that this simplex can be used to solve simultaneous equations or linear equations.

(Refer Slide Time: 30:48)



So we take an example to explain that we are into solving linear equations. We just take a simple example to solve linear equations. So without an objective function we just take $4X_1 + 3X_2 = 25$; $2X_1 + X_2 = 11$. Remember we do not have an objective function in this case whereas every linear programming problem is characterized by an objective function. We right now are going to assume that the solution to this has X_1, X_2 greater than or equal to 0 or if X_1, X_2 is greater than or equal to 0 then simplex will detect it. So we first convert this to a case by adding artificial variables. So we have $+ a_1 = 25$; $+ a_2 = 11$. Now if the original problem, $4X_1 + 3X_2 = 25$; $2X_1 + X_2 = 11$ has a solution then that should not have this a_1 and a_2. So what we do now is we create an artificial objective function. We now take this problem. We create an objective function; minimize $a_1 + a_2$ subject to the conditions X_1, X_2, a_1, a_2 greater than or equal to 0. So if this problem has an optimal solution then a_1 and a_2 will not be there. If this problem has an optimal solution or a single. Let us assume, it has a unique solution then this will be able to detect if a_1 and a_2 do not appear in the basis. So we formulate a problem with a_1 and a_2 as starting artificial variables and then we proceed. Now we can see that here. Now we add artificial variables. We define an objective function, minimize $a_1 + a_2$. If the original equation has a non negative solution then we should have a feasible basis with X_1 and X_2 having $Z = 0$ for the linear programming problem. The simplex iterations are shown in the next table.

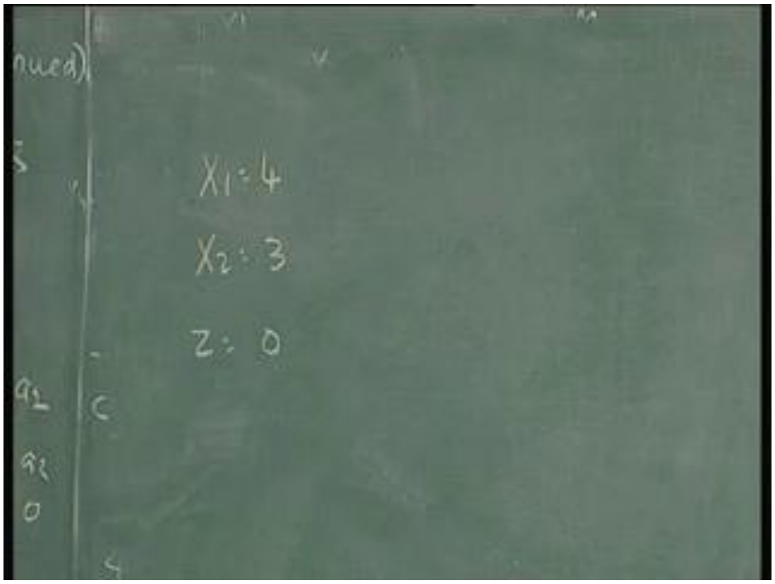
(Refer Slide Time: 33:08)



		0	0	-1	-1		
		X_1	X_2	a_1	a_2	RHS	θ
-1 -1	a_1 a_2	4 2	3 1	1 0	0 1	25 11	25/4 11/2
	$C_j - Z_j$	6	4	0	0	-36	
-1 0	a_1 X_1	0 1	1 ½	1 0	-2 ½	3 11/2	3 11
	$C_j - Z_j$	0	1	0	-1	-3	
0 0	X_2 X_1	0 1	1 0	1 -1/2	-2 3/2	3 4	
	$C_j - Z_j$	0	0	-1	-1	0	

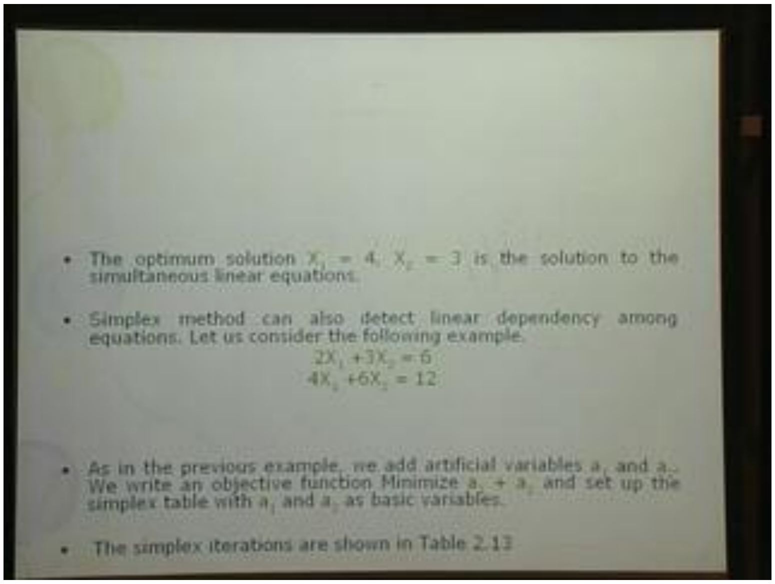
So we now have the simplex table that is given here with a_1 and a_2 as the starting basic variables. Variable X_1 will enter now with a most positive $C_j - Z_j$. Remember it is a minimization function minimizing $a_1 + a_2$. So a_1 and a_2 now have a – 1 because in simplex we convert it to a maximization problem. We have a – 1 here in the basis. Variable X_1 enters with value 6. Find theta 25/4, 11/2. 11/2 is smaller than 25/4, so variable a_2 leaves the basis. We do the Simplex iterations here with a_1 and X_1. In this table after computing $C_j - Z_j$ we realize that variable X_2 now enters the basis and a_1 leaves the basis and finally we have a solution with $X_1 = 4$; X_2 equal to 3 and Z = 0.

(Refer Slide Time: 34:24)



So now the linear programming solution for this is given by $X_1 = 4$; $X_2 = 3$; $Z = 0$; a_1 and a_2 do not appear in the solution. Now this is optimal to a given set of equations $4X_1 + 3X_2 = 25$; $2X_1 + X_2 = 11$. We can use simplex to solve a set of linear equations 2/2, two variables, two constraints or any three variables, any M variable, M constraint problems, provided they are all equations, provided they are greater than or equal to 0, by adding artificial variables here, by creating an objective function, minimize some of the artificial variables. Finally if there is a solution then all those will come into the basis. Every single artificial variable will be eliminated. Finally we will get a basic feasible solution with X_1 equal to something X_2 equal to something. In this example $X_1 = 4$; $X_2 = 3$ with $Z = 0$; the objective function tries to minimize $a_1 + a_2$ in this case. So this is an additional thing that simplex algorithm can offer.

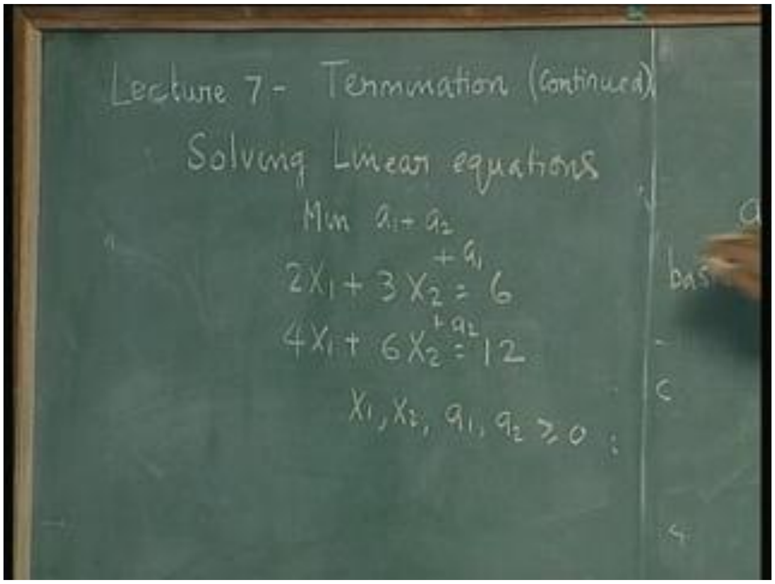
(Refer Slide Time: 35:30)



Simplex can also do another interesting thing. Now given a set of equations we know from algebra that 3 things are possible.

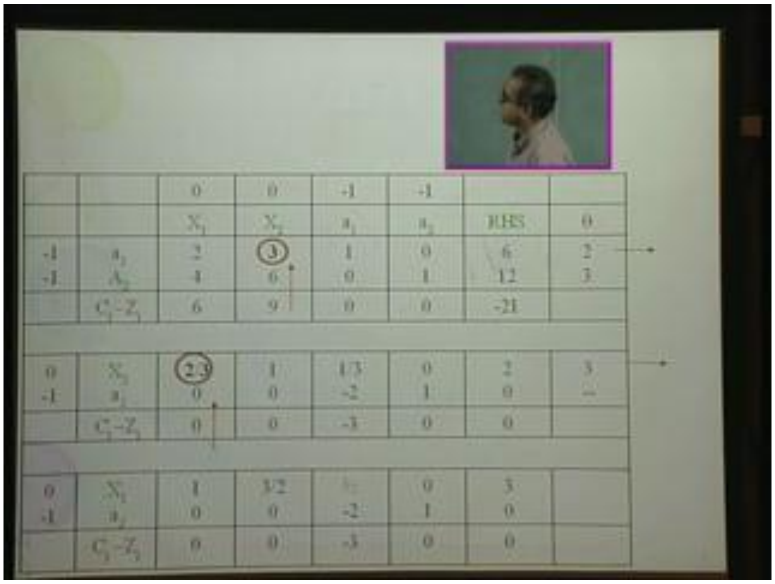
We can get a unique solution we could have infinite number of solutions and it may not have a solution at all. When a system does not have a solution at all it is infeasible. We have seen how simplex can detect infeasibility. When a system of equations has a unique solution we have just seen how simplex can identify that unique solution. So the last thing is if the system is linearly dependent then it will have a infinite number of solutions. We will see how simplex is used to detect a system of linearly dependent equations. So we again take an example with that. The example that we can consider is like this.

(Refer Slide Time: 36:27)



We have a system which is like this $2X_1 + 3X_2 = 6$; $4X_1 + 6X_2 = 12$ is a linearly dependent system. Now we want to solve this. We now realize that this matrix is singular and so on but then we can still use simplex to solve this. Now we again add two artificial variables here + a_1 + a_2. We now want to introduce an objective function which minimizes $a_1 + a_2$. We have X_1, X_2 a_1, a_2 greater than or equal to 0. Now we set the simplex table. Simplex iterations are shown in the next table. It is like this. Again it is a minimization problem. We have converted it to maximization by multiplying with the – 1. We start with a_1 and a_2 as the basic variables. So a_1 and a_2 have a – 1 here.

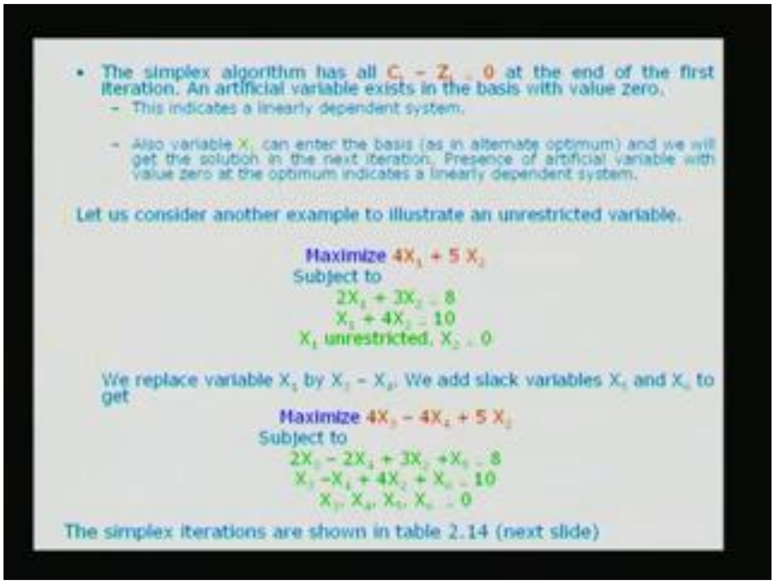
(Refer Slide Time: 37:40)



Now variable X_2 with $C_j - Z_j = 9$ enters the basis and replaces variable a_1. Now in the next iteration we realize that X_2 and a_2 are in the basis. Now we come to the situation were variable X_1 with $0(C_j - Z_j)$ tries to enter and more importantly you realize here that the artificial variable a_2 now takes value 0 here. We perform the next iteration. We realize that X_1 and a_2 are in the basis. Once again $C_j - Z_j$'s are all 0 or negative. The algorithm terminates. The algorithm terminates but does not give a give a unique solution. The algorithm terminates with an artificial variable lying in the basis but more importantly with value 0, we can see here that the artificial variable is in the basis and takes value 0. So when we solve a set of equations and if the artificial variable takes 0 at optimum, it is in the basis and takes 0 at the optimum which indicates that we are looking at a linearly dependent system of equations.

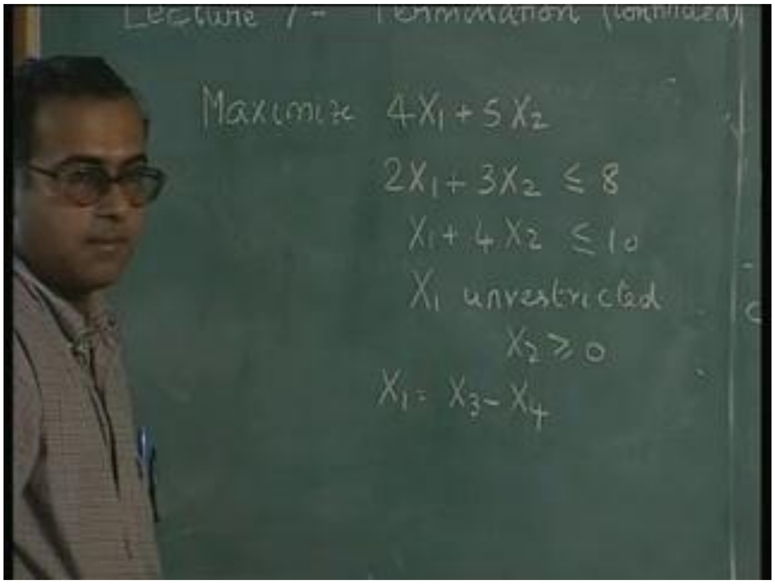
So simplex is capable of also indicating linear dependency among these equations. In addition to solving a linear programming problem, simplex algorithm can do two or three other things. One is, it can detect whether a given linear programming problem is feasible or not. It can detect infeasibility. It can be use to solve a set of linear equations if it has a unique solution and with all greater than or equal to 0. More importantly simplex can be used to detect a linearly dependent system and if the artificial variable is in the basis at the optimum and takes value 0 as in this example here, it indicates that the system we are trying to solve has a linearly dependent set of equations next.

(Refer Slide Time: 39:59)



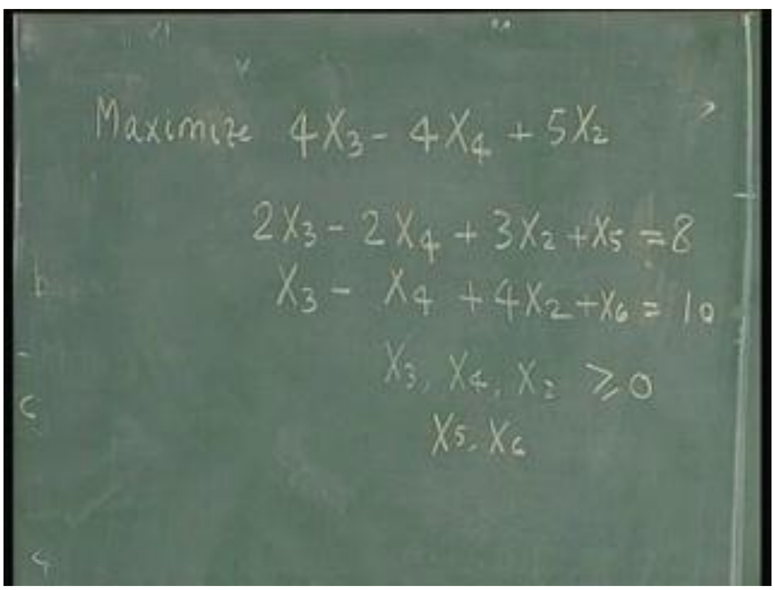
We need to look at one more aspect which we had not dealt with so far. What we need to do is before we wind up, the basic idea of simplex is to see how the simplex algorithm works if we have an unrestricted variable in the problem. We mentioned during our initialization that if we have an unrestricted variable then we replace the variable as a difference of two variables.
So let us take an example here to explain what happens to your simplex algorithm when we are trying to solve a problem with unrestricted variables.

(Refer Slide Time: 40:46)



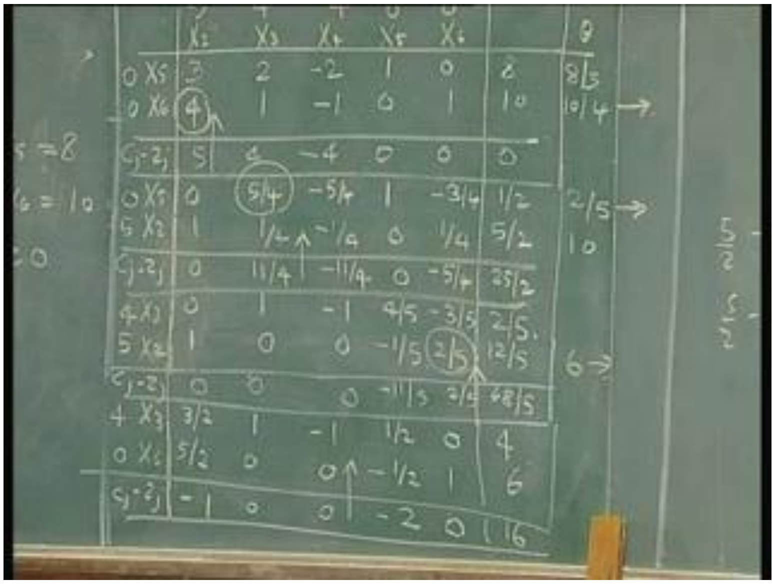
The problem that we will consider is this. Maximize $4X_1 + 5X_2$ less than or equal to 8; $X_1 + 4X_2$ less than or equal to 10; X_1 unrestricted, X_2 greater than or equal to 0. We now we replace this unrestricted variable as a difference of two variables and we write X_1 as $X_3 – X_4$ to get,

(Refer Slide Time: 41:42)



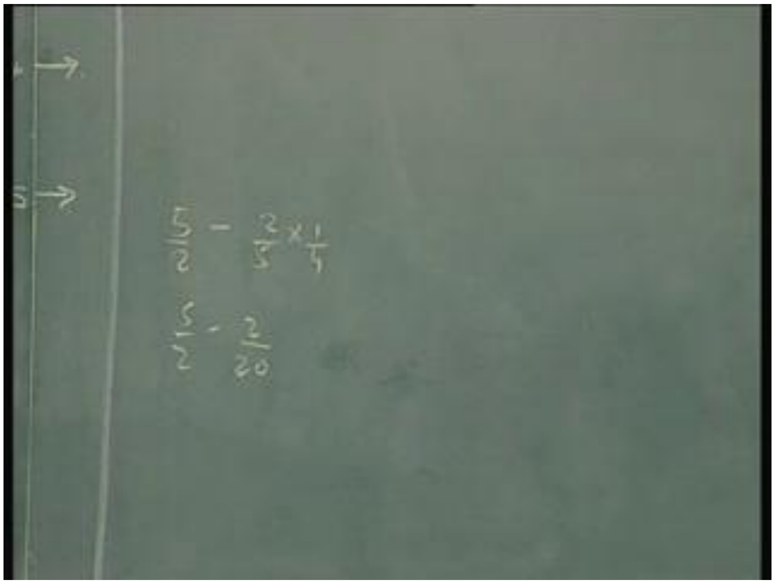
maximize $4X_3 – 4X_4 + 5X_2$ subject to $2X_3 – 2X_4 + 3X_2$ less than or equal to 8
$X_3 – X_4 + 4X_2$ less than or equal to 10; X_3, X_4 and X_2 greater than or equal to 0
Now we convert these inequalities to equations by adding a + X_5 = 8 and + X_6 = 10 and say X_5, X_6 are also greater than or equal to 0.

(Refer Slide Time: 42:49)



Now we set up the simplex table X_2, X_3, X_4, X_5 and X_6. We start with X_5 and X_6 here. We have $4X_3 – 4X_4$, $5X_2$, 0, and 0. So I have $3X_2$, $2X_3$, $– 2X_4$, $1X_5$, $0X_6 = 8$; $4X_2 + X_3 – X_4$, 0, 1 = 10.
$C_j – Z_j$, these are all 0s so I have (5 4 – 4 0 0).
So variable X_2 with the largest $C_j – Z_j$ 5 enters the basis and we will replace X_6 so we have X_5 and X_2 in the basis here. This is the pivot element, so dividing by the pivot we would get 1, 1/4, – 1/4 = 0, 1/4, 10/4 or 5/2. There is a 0 here and there is a 5 here. Now we need to get a 0 here so 3 – (3 times 1) is = 0; 2 – 3/4 is = 5/4; – 2 + 3/4 is = – 5/4; 1, – 3/4, 8 – (3 times 5/2) 8 – 15/2 is 1/2.
Now $C_j – Z_j$ will be 0 for X_2 and X_5 and this would be 4 – 5/4 is = 11/4; (– 4 + 5/4) = – 11/4, 5/4 so I have – 5/4 and Z will be = 25/2. So now variable X_3 with the largest $C_j – Z_j$ will enter.
To find out the corresponding theta 1/2 divided by 5/4 is 1/2 into 4/5 which is 4/10 or 2/5.Now this is 5/2 into 4 is = 10.Variable X_5 leaves. This is the pivot element. Now variable X_3 replaces X_5 so I have X_3 and X_5; X_3 has 4; X_2 has 5 and I perform row operations.
Once again divide this by the pivot element to get this row and this row would look like under X_2 I multiply by (4/5 to get 0 1 – 1 4/5) – 3/4 into 4/5 is = – 3/5 and this is 2/5. We need a 0 here so this – 1/4 times this. 1 – 1/4 into 0 is 1. We get a 0. This will also become a 0. 0 – (1/4 into 4/5) is = 1/5. So I get a – 1/5. This – 1/4 times this so 1/4; (– 1/4 into – 3/5) so 1/4 + 3/20 which is 2/5. This – 1/4 into this we get 5/2 – (1/4 into 2/5) = 5/2 – 2/20 which will give us 5/2 – (– 2/5 into 1/4) 5/2 – 2/20 = 48/20 which is = 12/5.

(Refer Slide Time: 47:12)



$C_j - Z_j$ values for X_3 and X_2 will be 0. This would also be 0. This 16/5 – 5/5 is = 11/5, – 11/5 – 12/5 + 10/5 is = – 2/5. I now get + 2/5; 8/5 + 60 will give us 68/5 so variable X_6 with the positive value of $C_j - Z_j$ enters. There is only 1 theta which is 12/5 divided by 2/5 = 6; so this is the pivot element. Now we have variable X_6 replacing X_2. X_3 remains as it is. X_3 remains at 4 and X_6 is at 0. Divide again by pivot element 2/5 so you get (5/2 0 0 – 1/2 1 and 6) now I need a 0 here. This + 3/5 times this would give a 0 so 0 + 5/2 into 3/5 = 5/2 into 3/5 is = 3/2. I get a 3/2 here, 1 here, – 1 here, so this + 3/5 times this. 4/5 – 3/10 which is = 5/10 which is 1/2. This (Refer Slide Time: 48:02) will be 0. 2/5 + (3/5 into 6) is = 20/5 gives us 4. So $C_j - Z_j$ would be, this is 6, so 5 – 6 is = – 1. This is 0 and this would also be 0. 4 into 1/2 is = 2. 0 – 2 is – 2; 4 into 0 + (0 into 1) is = 0; we get 0 here and the value is 16. Now what we observe here is we have reached the optimum but there is a non basic variable X_4 which has a 0 value of $C_j - Z_j$ which can technically enter and if we try to enter this we observe that there is no leaving variable.

So the question is that does it indicate an optimum? Does it indicate unboundedness because I am unable to find a leaving variable. Does it indicate an alternate optimum because I have a 0 which enters after the termination condition is met?

Now this is something that will happen to all problems that involves unrestricted variable.

This is because one of the variables X_3 which we defined here as part of X_1, (X_1 is now $X_3 - X_4$) X_3 takes value 4 here in the basis therefore the other variable X_4 will now try to enter. Now this is something which we have to recognize in almost every problem when we use this. This actually terminates. This indicates neither unboundedness nor alternate optimum but this is a termination condition when we are solving problem with unrestricted variable.The other variable will always try to enter the basis and we should be aware of this. Now with this we end our discussion on the simplex algorithm, the initialization, iteration and termination conditions. We have also seen an example of how simplex behaves when we are solving unrestricted variables.

In the next lecture we will continue our treatment on simplex algorithm using or learning principles of duality and sensitivity analysis.