Fundamentals of Operation Research

Prof. G. Srinivasan

Department of Management Studies

Indian Institute of Technology – Madras

Lecture No. # 10

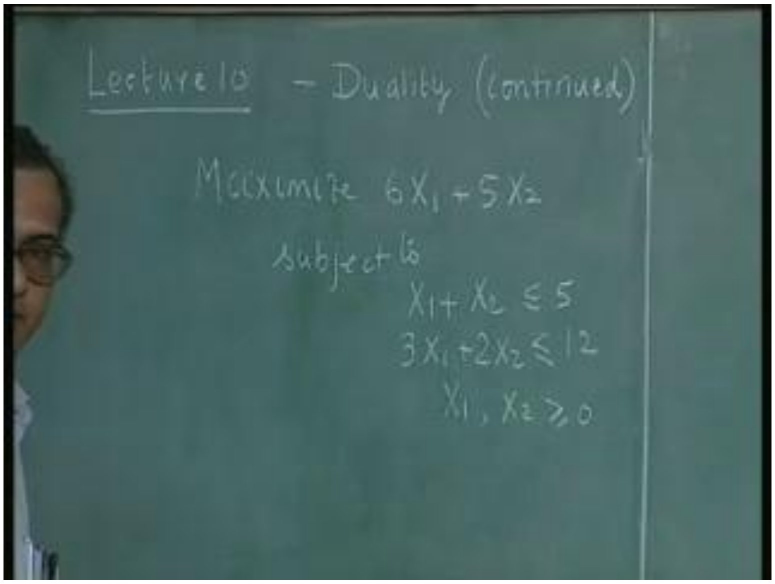
Dual Variables and the Simplex Tables

Dual Simplex Algorithm

(Refer Slide Time: 01:11)

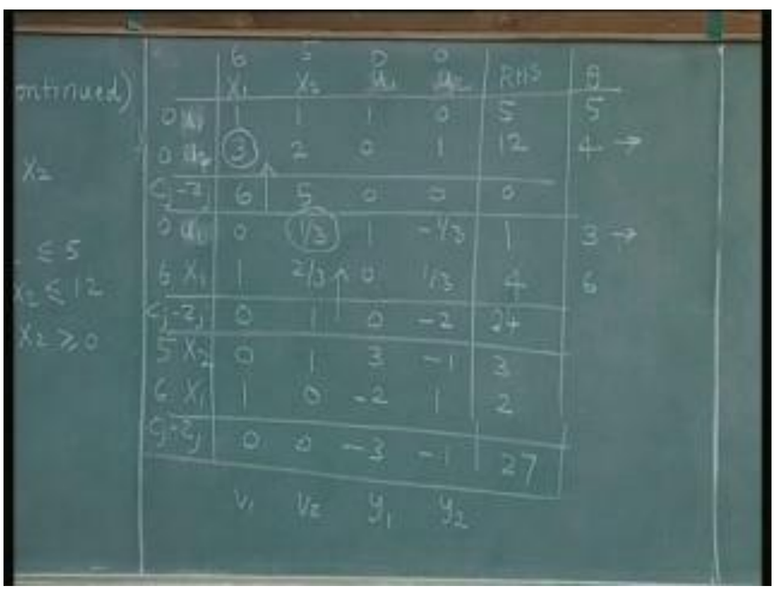
In the last lecture we looked at the complementary slackness condition and we also said that the simplex method also gives the solution to the dual at the optimum. We also wanted to find out how the $C_j – Z_j$'s in the intermediate iterations represent something or do they even represent anything? To consider that further, we will take the familiar example and try to interpolate the $C_j – Z_j$'s corresponding to the intermediate iterations of this simplex algorithm.

(Refer Slide Time: 01:53)



So the example is maximize $6X_1 + 5X_2$ subject to $X_1 + X_2$ less than or equal to 5 $3X_1 + 2X_2$ less than or equal to 12; X_1 X_2 greater than or equal to 0

(Refer Slide Time: 02:30)

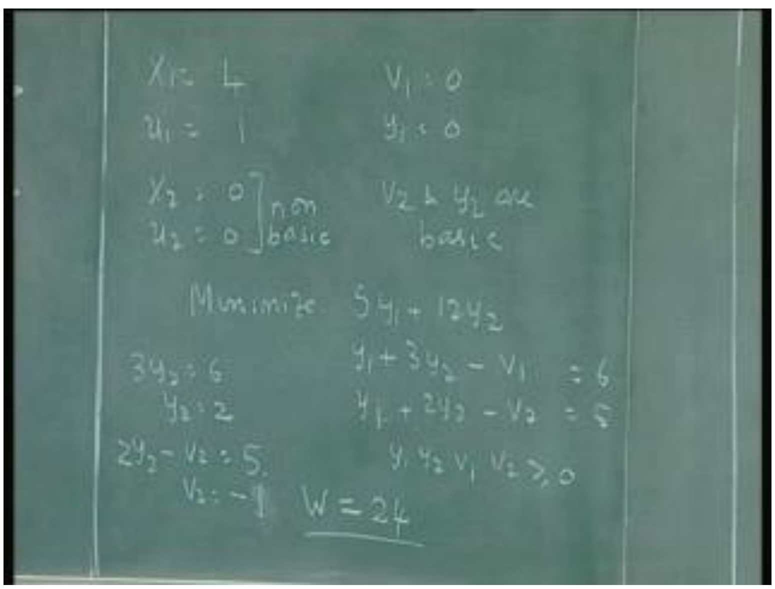


Very quickly performing this simplex iteration, we get X_1, X_2, X_3, and X_4 at the right hand side. We start with the X_3, X_4 1 1 1 0 5, 3 2 0 1 12, 6 5 0 0.
Now variable X_1 with the largest positive $C_j – Z_j$ enters. Corresponding theta values are 5 and 4 is a minimum theta. Variable X_1 enters to get X_3, X_1 and $C_j – Z_j$, Dividing by the pivot element, we would get 1, 2/3, 0, 1/3, 4. This minus this would give us 0, 1/3, 1 – 1/3, 1 is t = 24. 0 0 6 into

2/ 3 is 4. 5 – 4 is = 1. 6 into 1/3 is = 2. We get – 2 here. Variable X_2 enters the basis so this is 1 divided by 3 is = 3; 4 divided by 2/ 3 is = 6. This leaves pivot element.
X_2 replaces X_3 in the basis, so the final table will look like this. Dividing by the pivot element we get 0, 1, 3 – 1, 3. This – 2/3 times this will give 1, 0 – 2, 1/3 + 2/3 which is = 1; 4 – 2/3 into 3 is 2. $C_j – Z_j$ values are 5 into 3 = 15 – 12 = 3; I get a – 3 – 5 and 6 which will give us – 1 and 27.

In the last lecture we saw that this – 3 and – 1 when multiplied with the – 1 again would give us 3 and 1 which are the values of the dual at the optimum. We also said that amongst X_1, X_2, X_3, and X_4 now X_3 and X_4 are our u_1 and u_2 which are the primal slack variables. We replace them by u_1 and u_2 here. X_3 also becomes u_1. From the complementary slackness we understand that there is a relationship between X and V and y and u.
Whatever comes under X, we can write it as V_1 and V_2. Primal decision variables have a relationship with the dual slack. Primal slack have a relationship with the dual decision variables. So we say that with the minus sign, $y_1 = – 3y_2 = – 1$ represents the value here $y_1 = + 3y_2 = + 1$ represents that the values are dual at the optimum. Now let us look at an intermediate iteration here.

(Refer Slide Time: 06:32)



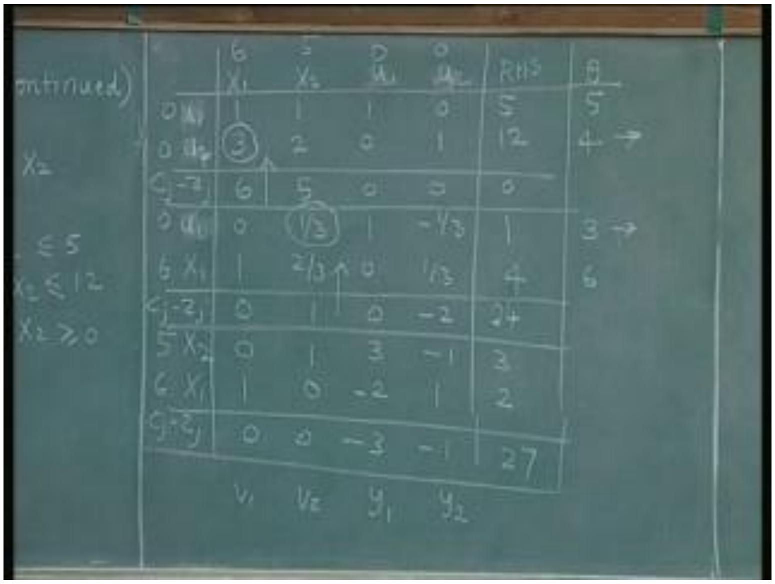
In this, intermediate iteration, solution to the primal is $X_1 = 4$; $u_1 = 1$. Let us apply the complementary slackness and see what happens. When we apply complementary slackness to this, X_1 basic indicates $V_1 = 0$. u_1 basic indicates $y_1 = 0$ and non basic. Here X_2 and u_2 are non basic indicating that V_2 and y_2 are basic for the equivalent solution to the dual if we apply complementary slackness condition. We are writing the dual for this problem. Dual is to minimize $5y_1 + 12y_2$, subject to $y_1 + 3y_2 – V_1 = 6$; $y_2 + 2y_2 – V_2 = 5$; y_1, y_2, V_1, 1 V_2 greater than or equal to 0. When we apply complementary slackness to this we have to solve for V_2 and y_2 which are the basic variables. This solution will become $3y_2 = 6$ from which $y_2 = 2$ and from here we have $y_2 – V_2 = 5$ from which $V_2 = – 3$. When we apply complementary slackness conditions two and intermediate iteration of the simplex algorithm and evaluate the corresponding dual as we have done here, the corresponding primal solution is $u_1 = 1$; $X_1 = 4$; Z

= 24. When we apply complementary slackness and solve we get y_2 = 2; V_2 = – 3; Z or W = $5y_1 + 12y_2$ W is also = 24. Let us go back to this intermediate iteration and see whether we have this solution reflected somewhere in this $C_j – Z_j$. If we go back now we have y_2 = 2; we realize this is y_2 so the value here under the $C_j – Z_j$ once again multiplied with the – 1 gives us the value of y_2 = 3 and from this and I have $y_1 + 2y_2 – V_2$.
So $2y_2 – V_2$ is = 5; V_2 is – 1; V_2 is here which is shown under X_2 and we realize the negative of this number is – 1.

What we observe here is if we take an intermediate iteration of the simplex algorithm, write the complementary slackness corresponding to that and then evaluate a corresponding dual, we realize that the solution to that dual is also seen under the $C_j – Z_j$ of the intermediate iteration.
Simplex satisfy complementary slackness conditions at every iteration of the algorithm not only at the optimum. We see some more interesting things happening. It satisfies the complementary slackness conditions. It gives an equivalent solution to the dual after the complementary slackness conditions are applied. We get the same value of the objective function for the primal and dual respectively. We also realize that the dual is infeasible here because V_2 is – 1 we would want y_1, y_2, V_1 and V_2 all greater than or equal to 0. Now V_2 is – 1.
The infeasibility of the dual which is V_2 = – 1 is now reflected in the corresponding non optimality of the primal. You can see that V_2 = – 1 is actually represented as a $C_2 – Z_2$ = + 1 which enters the basis to get the next iteration.

Whichever dual variable is infeasible or whichever dual constraints are infeasible. After all V_2 = – 1 comes from the fact that $y_1 + 2y_2$ is not greater than or equal to 5.
This V_2 is nothing but the extent to which $y_1 + 2y_2$ is greater than 5 because the actual constraint is y 1 + $2y_2$ greater than or equal to 5. If $y_1 + 2y_2$ is strictly greater than 5 then V_1 will take a positive value. In this case because $y_1 + 2y_2$ is less than 5; V_1, V_2 takes a negative value.
V_2 taking a negative value indicates that the second constraint is violated. A dual is infeasible when either the decision variable takes negative value or the slack variables take negative value.
Slack variable taking negative value represent the extent of not satisfying a particular constraint so the extent of infeasibility of the dual represents the extent of non optimality of the primal. If the second dual constraint is violated by one unit then it implies that in the simplex algorithm, the corresponding digestion variable (if this slack variables take a negative value) which is non basic because of the complementary slackness condition will now try to enter the basis and will indicate the rate of increase of the objective function which is 1 which is the same – 1 which we see here (Refer Slide Time: 13:49) Simplex not only satisfies the complementary slackness conditions at the optimum. At every iteration of this simplex algorithm, the complementary slackness conditions are satisfied. In every intermediate iteration a non optimal iteration with respect to the primal, is an infeasible solution to the dual and the extent to which the entering variable in the intermediate iteration corresponds to the infeasible dual slack if a decision variable or a dual decision variable or slack variable enters.

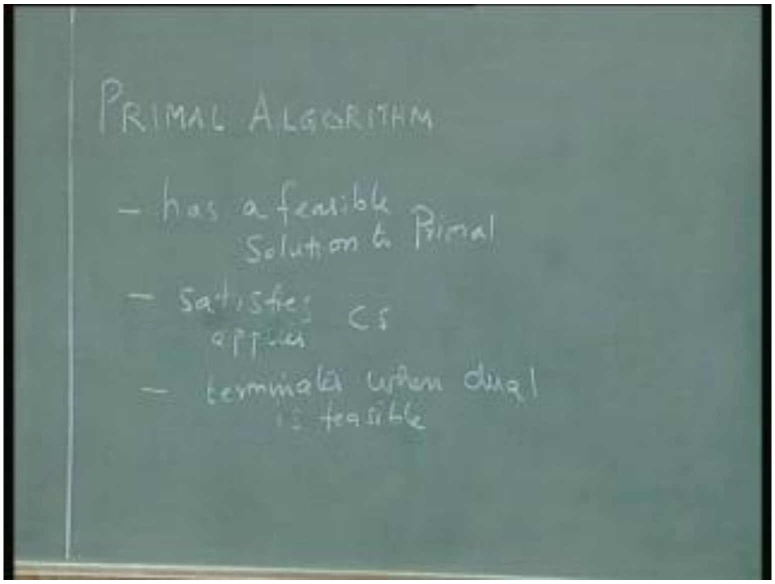
(Refer Slide Time: 14:02)



The rate of the $C_j - Z_j$ corresponding to the entering variable is actually the extent to which the dual is infeasible. Non optimality of the primal which is represented by an intermediate iteration represents infeasibility to the dual. There is a different way of looking at this $C_j - Z_j$. What this simplex algorithm does is remember that the right hand side is always greater than or equal to 0. The rule for the leaving variable will ensure that the right hand side will never become negative. A simplex as an algorithm will always have a primal solution which is basic feasible. It will not have an infeasible solution at all. Now this represents the feasibility of the primal now this represents either the optimality of the primal or the feasibility of the dual. This plus value here indicates that the corresponding dual is infeasible with the negative. What simplex tries to do is it starts with a feasible primal and an infeasible dual. This would mean both X_1, V_1 and V_2 are negative. So it starts with a feasible primal. It has an infeasible dual and it tries to make the dual feasible. The extent of infeasibility of the dual is given by the $C_j - Z_j$ of the corresponding value. It tries to take that variable which is least feasible by picking up this 6 here which is nothing but the maximum rate at which I can increase objective function for the primal. In successive iteration it tries to make the dual feasible.

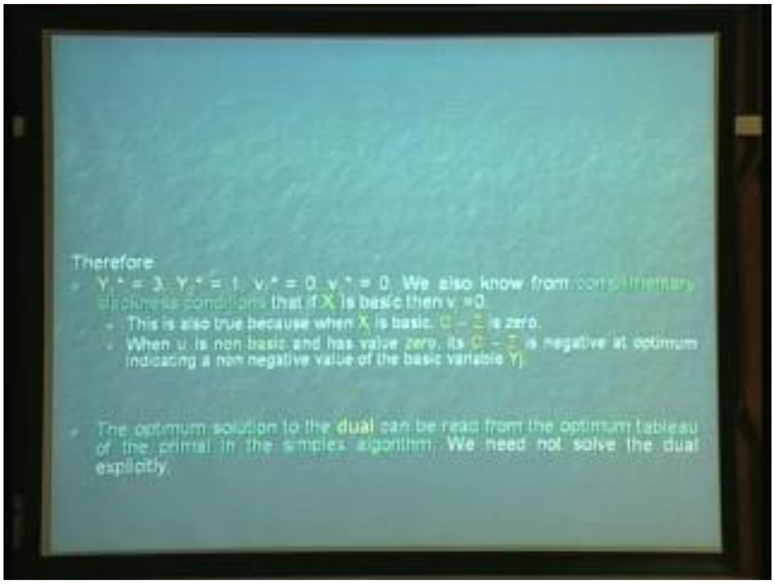
Here you have two dual variables that are infeasible. We have one dual variable that is infeasible. It tries to make the dual also feasible by applying complementary slackness. In the end when both the primal and the dual are feasible, then the optimal is reached after applying the complementary slackness. The optimality can be looked at in this way. We can go back and try to even interpolate all the duality theorems. One, you have a basic feasible solution here with an objective function value of 27 that is feasible to the primal. You also have a feasible solution to the dual with a same value of the objective of function therefore it is optimal to both primal and dual respectively. A simplex as an algorithm is always feasible to the primal. It applies complementary slackness and tries to evaluate the corresponding dual in every iteration. The moment it finds that the corresponding dual is also feasible then it is optimum. Simplex can be seen in an algorithm which actually solves both the primal and the dual, but it keeps a basic feasible solution to the primal and works towards getting a feasible solution to the dual there/making it optimum. Now simplex comes under a category of what are called primal algorithms.

(Refer Slide Time: 17:21)



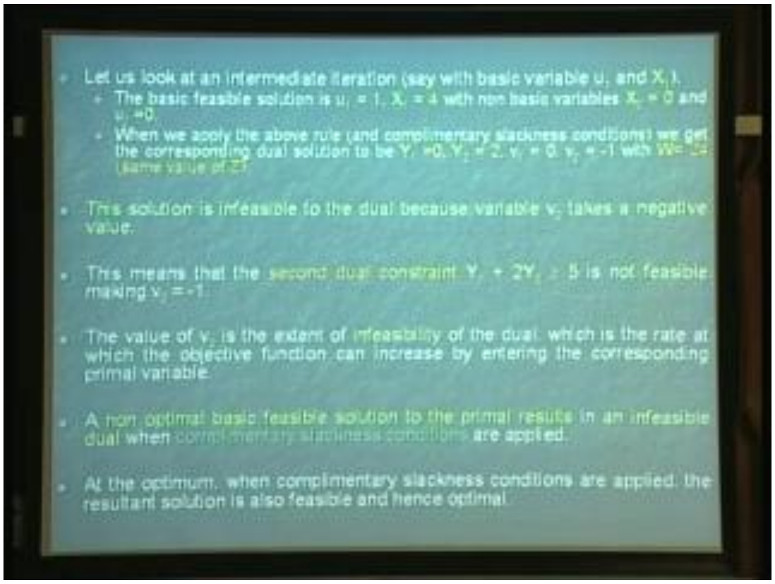
A primal algorithm always has a feasible solution to the primal. It satisfies complementary slackness or applies complementary slackness and terminates when the corresponding dual is feasible indicating that both primal and dual are feasible and optimum. We do not need to solve a dual at all. We now realize that when we solve the primal we are automatically solving the dual and we can always get the solution to the dual of the problem that we are solving from the primal iterations. This is the primal that we have solved. Solution to the primal is here. Solution to the dual of this problem is here to the $y_1 = 3$; $y_2 = 1$; Z is = 27. Let us go back and see this.

(Refer Slide Time: 18:37)



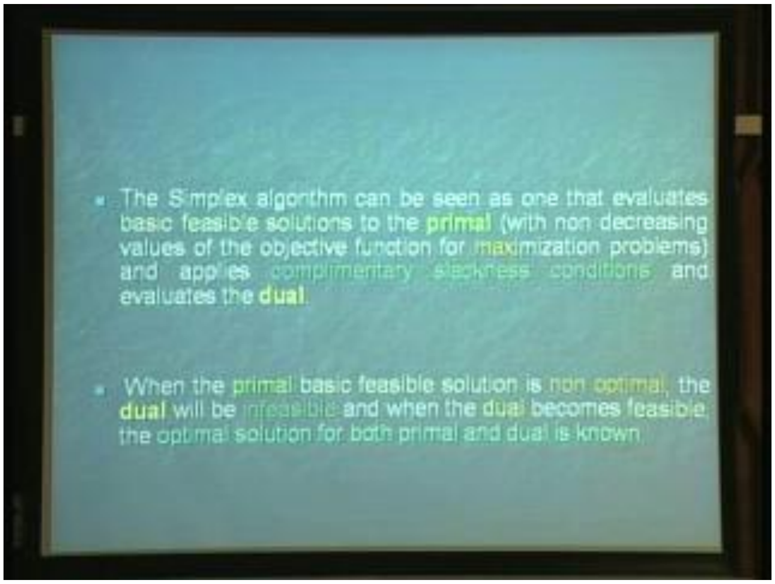
The optimum solution to the dual can be read from the optimum tableau of the primal in the simplex algorithm and we need not solve the dual explicitly at all.

(Refer Slide Time: 18:48)



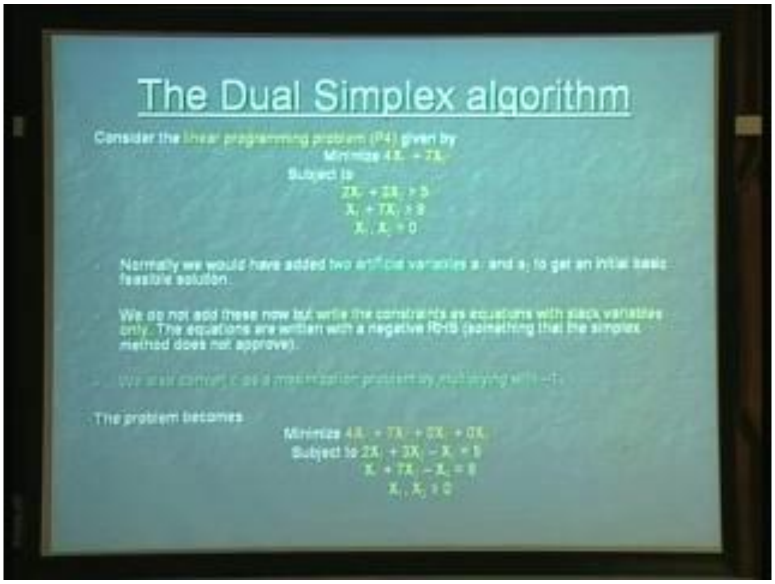
Now we look at the intermediate iteration. Whatever we have tried to show here is what is explained here. This solution is infeasible to the dual because variable V_2 takes a negative value which is here. We have to remember constantly that we need to multiply this with the – 1. This is actually the negative dual variable not this (Refer Slide Time: 19:07). It just shows us that this constraint is violated by the same quantity of one which is seen here and the optimum when the complementary slackness conditions are applied resultant solutions is feasible to the dual and hence optimum.

(Refer Slide Time: 19:28)



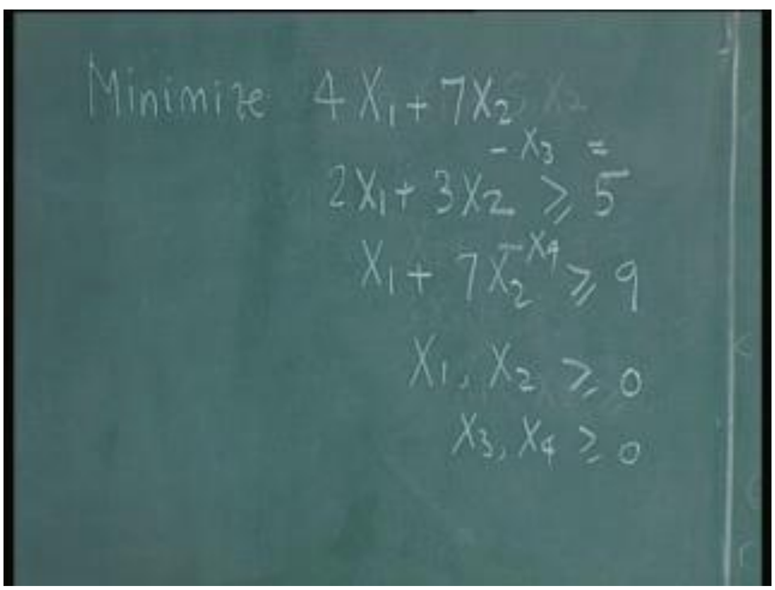
Simplex can be seen as one that evaluates basic feasible solutions to the primal and applies complementary slackness conditions and evaluates a corresponding dual. When the primal basic feasible solution is non optimal as in an intermediate iteration, a dual will be infeasible and as and when the dual becomes feasible, we get the optimal solution for both the primal as well as a dual.

(Refer Slide Time: 19:50)



We next look at what is called as dual simplex algorithm and try to see another version of simplex which is quite interesting and different from the version that we have seen.

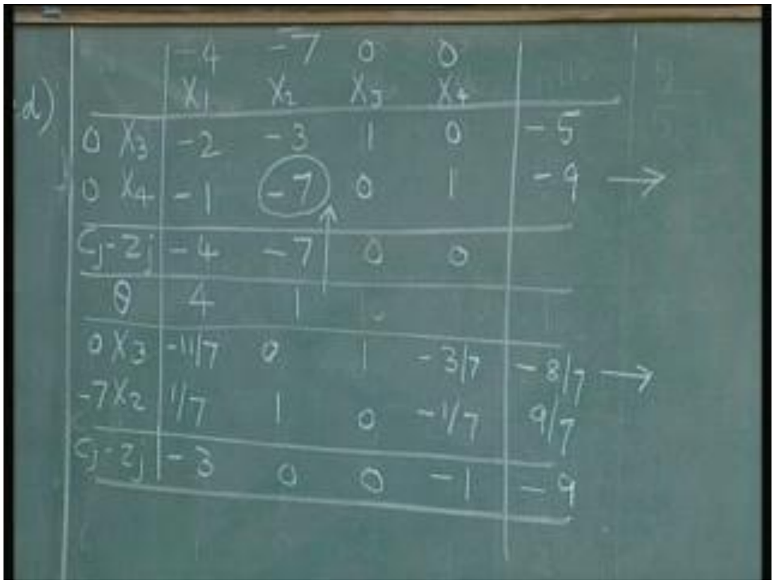
(Refer Slide Time: 20:23)



We take an example. Minimize $4X_1 + 7X_2$ subject to $2X_1 + 3X_2$ greater than or equal to 5; $X_1 + 7X_2$ greater than or equal to 9; X_1, X_2 greater than or equal to 0.
We add slack variables because of the greater than or equal to. We have negative slack
So we have $– X_3 = 5$; $– X_4 = 9$; X_3, X_4 greater than or equal to 0.
Normally we would added two artificial variables $a_1$and a_2 to get the basic feasible solution because X_3 and X_4 by themselves are not capable of giving us a starting basic feasible solution

because $X_3 = -5$; $X_4 = -9$ is infeasible but now let us see what happens if we still start with X_3 and X_4.

(Refer Slide Time: 21:41)



Now we first thing we do is we convert this problem to a maximization problem. We have X_1, X_2, X_3 and X_4. We are not going to add artificial variables at all. The problem becomes maximize $-4X_1$, $-7X_2$, $0X_3$ and $0X_4$. We now still keep X_3 and X_4 as basic variables.
We multiply this because this is equal to 5. We multiply this with the -1 to get $-2X_1 - 3X_2 + X_3 = -5$. Hence, $-2X_1 - 3X_2 + X_3 + 0X_4 = -5$
$-X_1 - 7X_2 - 1X_1 - 7X_2$, 0, 1, $X_4 = -9$. We have 0 and 0
Now we have violated a very important assumption in simplex that the right hand sides should be greater than or equal to 0. Now we have a solution which is not basic feasible but has a completely negative right hand side values. But let us look at $C_j - Z_j$ values. $C_j - Z_j$ for this would give us a -4 here -7 here 0 0. We have 0s here and they do not contribute at all $C_j - Z_j$ would simply become C_j so we have this.

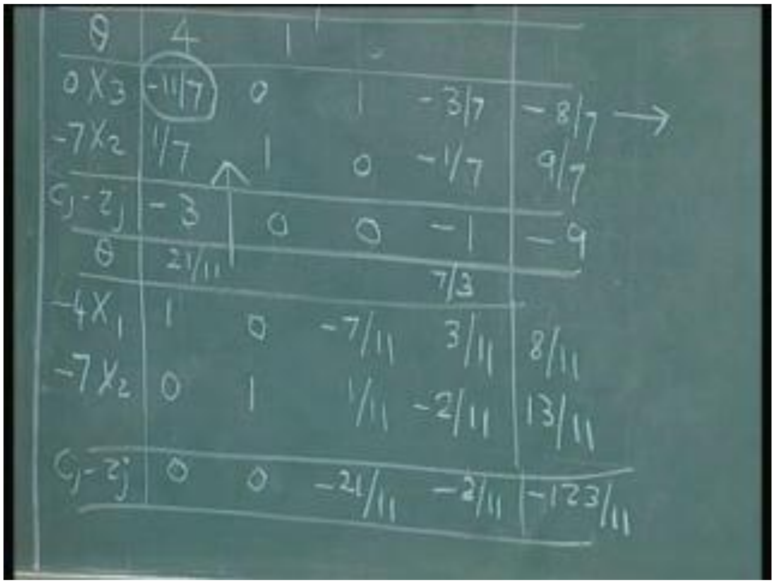
Because of the problem we will now come back to what is peculiar about this problem. But we definitely observe that the optimality condition is satisfied. The optimality condition is satisfied, feasibility condition is not satisfied or just extending whatever we had seen, the dual corresponding to the solution $X_3 = -5$; $X_4 = 9$ is feasible because I have a completely negative $C_j - Z_j$. The dual is feasible. So we have a simplex iteration where the primal is infeasible but the dual is feasible in all our earlier simplex versions. In any first iteration or intermediate iteration the primal will be feasible. Dual will be infeasible. Now we have exactly the opposite happening. The primal is infeasible. The dual will feasible. Now we can still workout simplex.
If we can maintain the feasibility of the dual and slowly make this feasible then it will become optimum. In a normal simplex what we did is we maintained this feasibility. We tried to bring the dual feasible and we got the optimum. Now here if we can keep the dual feasible consistent and then make this one feasible then it will be become optimum. Let us do that. Now we can have an entering variable here because all the variables are with a negative $C_j - Z_j$. The first thing

we need to do is we need to somehow make this feasible which means we need a positive value or a non negative value on the right hand side. The most negative of this will leave first so we first find out the leaving variable and leave out the variable with the most negative value of the right hand side. So this will leave. We are trying to do some steps. Some part of the simplex which is the opposite and some part of the simplex which is common to the earlier one. Here first find out the leaning variable. We need a variable that can enter and substitute this X_4. In order to do that, we compute a theta but theta now comes in the row because we have to find out an entering variable. Next, to do that, we do something very similar. Take this value and divide here. – 4 divided by – 1 is = 4; – 7/– 7 is = 1; we have 0. We can leave this because these are the basic variables. We want only one of the non basic variables to enter. We do not have to evaluate the theta for what are presently the basic variables.

You will evaluate the theta only to represent non basic variables. These are the theta values.
Once again the smallest theta will enter so we have this. Now what happens, as a result of this, in this algorithm the $C_j – Z_j$'s will always be negative and we need a negative pivot because when we do the simplex iteration next, we divide this by the pivot element we will then get a positive value here. In this case we would need a pivot that is negative and this value will always be negative. So you will compute thetas only for negative values in the row corresponding to the leaving variable. If there is an element with the positive value in the row corresponding to the leaving variable you will not compute the theta even if that is a non basic variable. For example if this had been a + 1 we would not have computed this theta at all which is very similar to situations where we do not compute theta in the earlier version of this simplex. (in the earlier version of the simplex if that number is negative you would not compute theta or if that number is 0 you would not compute theta). Here if that number is positive or 0 you will not compute theta. So you will not compute theta here because of this 0. In any case this is a basic variable so you won't do that.

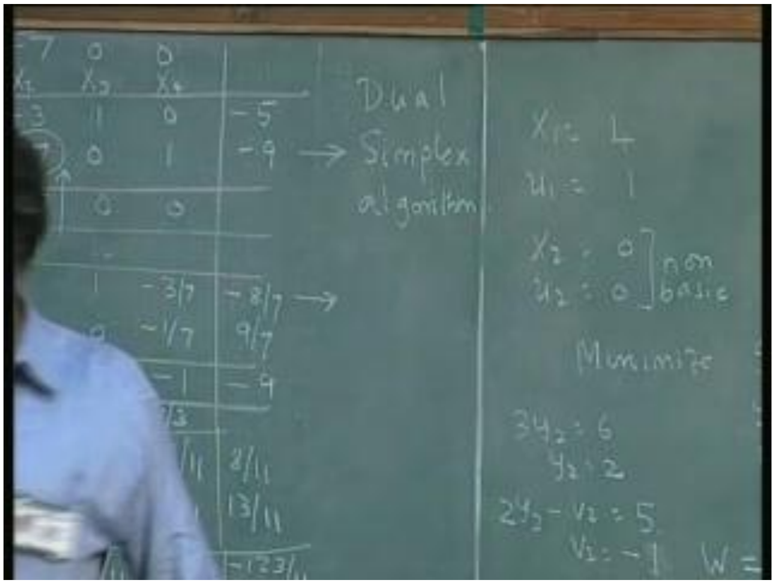
Now variable X_2 enters and variable X_4 leaves I have X_3, X_2. This is – 7 this is 0, so divide by the pivot element to get 1/7, 1, 0, – 1/7, 9/7. We need a 0 here. This + 3 times this would give a 0. This + 3 times this (Refer Slide Time: 27:49) will give a 0 – 2 + 3/7 is = – 11/7, 0, 1 this + 3 times this would give us – 3/7. This + 3 times this – 5 + 27/7 is – 8/7. The – 8/7, 9/7, 9/7 and we have Z is = – 9 here. Now we need to find out $C_j – Z_j$, X_3 and X_2 are basic variables.
We get 0 here. This 0 into – 11/7 + (– 7) into 1/7 – 1 – 4 – (– 1) is = – 3. Here I have a + 1
So 0 – (+ 1) will gives me = – 1. Once again I have $C_j – Z_j$ less than or equal to 0. The optimality condition is satisfied. How is it satisfied? It is satisfied because of the minimum theta rule that we followed to define the entering variable. We made sure that the optimality condition is satisfied. The feasibility condition is not satisfied because this as a negative right hand side. This has to go now. We need to find out the entering variable corresponding to this leaving variable. How do we do this?

(Refer Slide Time: 29:59)



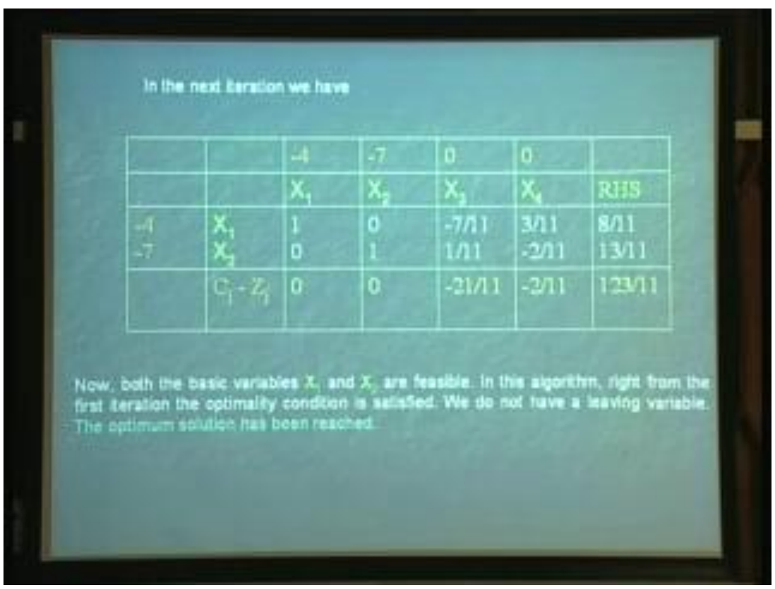
We compute theta again. – 3 divided by – 11/7 is = 21/11.
– 1 divided by – 3/7 is 7/3; 21/11 is smaller than 7/3. Variable X_1 enters and this is the pivot. Remember that the pivot element has to be negative in this case. Only when the pivot element is negative, the right hand side will become non negative in the next iteration. We continue with this simplex. The simplex iteration part of it is same. It does not change. The row operations are all the same. We will have X_1, X_2. X_1 you get – 4; X_2 you get – 7. So dividing by the pivot element or multiplied by – 7/11 which is 1, 0, – 7/11 + 3/11. This is 8/11, this – 1/7 times this is = 0, so 1/7 – this is = 0. This is 1. This – 1/7 times gives me + 1/11 – 1/7 – 1/7 into 3/11.
So – 1/7 – 3/77 is – 14/77 which is – 2/11, 9/7 – 1/7 into 8/11. 9/7 – 8/77, so 99 – 8 is 91/77 which is 13/11.So we get 13/11 here and then the value here will be 32 + 91 is = 123/11 with the minus sign, 123/11. $C_j – Z_j$ values are all 0 and this is + 28/11 – 7/11 is + 21/11. 0 – (+ 21/11) is = – 21/11 – 12/11 + 14/11 is = 2/11. So 0 – 2/11 is – 2/11. Once again the $C_j – Z_j$ values are negative indicating that the optimality condition is satisfied. Now we realize that the solution is feasible. Feasibility condition is also satisfied therefore this is optimum. You can also show that, for example (21/11 into 5) + (2/11 into 9) will also gives us 123/11, 105 + 18 will give 123. So the optimal solution to the primal is X_1 = 8/11; X_2 = 13/11; Z = + 123/11. The minus comes because we have converted a minimization problem into maximization problem by multiplying with the – 1. Optimum solution to the dual will be Y_1 = 21/11; Y_2 = 2/11; W = 123/11.

(Refer Slide Time: 33:46)



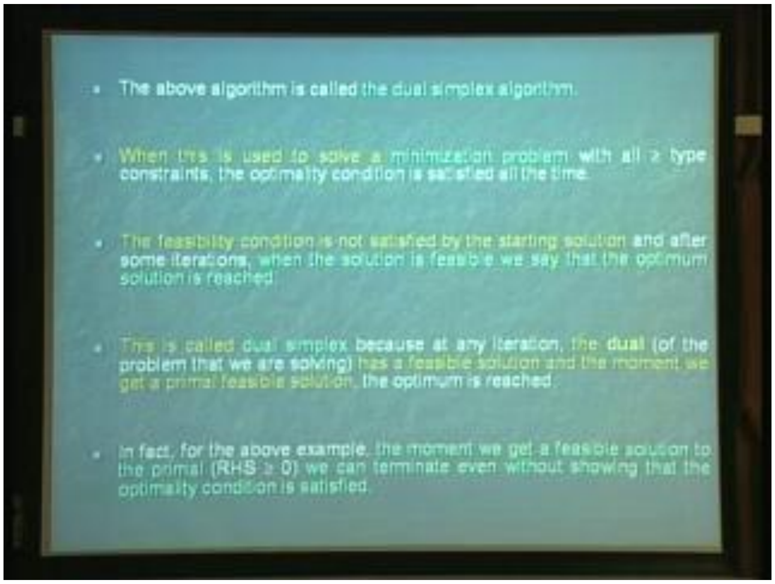
This algorithm is called the dual simplex algorithm. Now let us go back to this.

(Refer Slide Time: 34:00)



Both X_1 and X_2 are feasible. Right from the first iteration, the optimality condition is satisfied. We do not have a leaving variable. The algorithm terminates to give us the optimum.

(Refer Slide Time: 34:10)



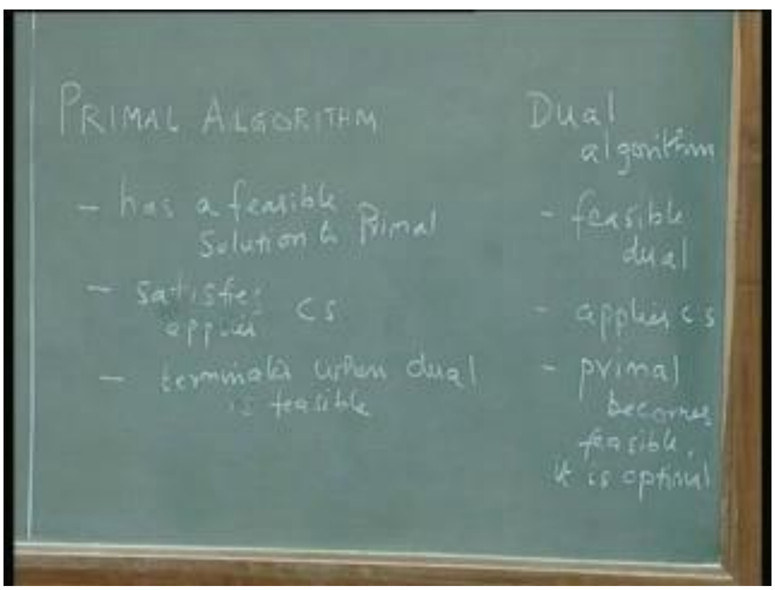
The above algorithm is called the dual simplex algorithm. What is special about the dual simplex algorithm? A dual simplex algorithm is very well suited when you have all greater than or equal to constraints. We are not unduly worried about the objective. But it is very well suited when you have a minimization problem with all positive coefficients here. So that the equivalent maximization will have all negatives and the first $C_j - Z_j$ will have all negatives. It is very well suited for a minimization problem with all positive coefficients in the objective function and all constraints of the greater than or equal to type and with a non negative right hand side. A non negative right hand side is granted when we formulate the problem. So in all greater than or equal to constraints with non negative right hand side minimization function, all non negative or positive coefficients, this algorithm is very well suited.

We do not need to introduce artificial variables at all. We do not need to do the big M method. We can still use the dual simplex and so on. The condition for the dual simplex is we will have to have the $C_j - Z_j$'s negative to begin with. This will also be the negative indicating infeasibility. We will retain the optimality condition by a careful choice of the entering variable and theta by following a similar minimum theta rule by making the pivot element negative or by pivoting on a negative element corresponding to the leaving variable. We make the corresponding right hand side non negative. We ensure that we are converting in every iteration at least one negative value to a non negative value. Once again it is not absolutely necessary that within two iterations this has to converge. We may encounter a situation where for example here by doing one iteration, we are making sure that we have non negative number here. But this could turn out to be negative and then it could go on. But if the algorithm has a solution then it will terminate. In the dual simplex algorithm, we have to make sure that the pivot element is negative.

Only when the pivot is negative we can get a positive here. Therefore the theta is computed in such a way that it is computed only for those non basic variables which have negative coefficients or are not completely negative coefficients in the leaving row. Then the minimum theta will ensure that the optimality condition is satisfied. Feasibility is not satisfied by the starting solution. If the solution is feasible we say that the optimum is reached. It is called dual simplex for a very specific reason. There are two ways of looking at it. So far we are used to having a maximization problem with less than or equal to as our primal. Then we might be tempted to believe that the dual of the primal by the way we have defined is a problem that fits into this structure which is correct. The dual of the problem that we defined as primal through fits into this structure. We might be tempted to call this dual simplex, because it can solve the dual of our primal while that is not the real reason. The real reason is whatever problem you chose to solve, moment you start the dual simplex algorithm, if you look at this carefully the optimality condition being satisfied indicates that solution to the dual of the problem that you are solving is feasible. So right through the dual simplex algorithm you have a dual feasible solution, primal infeasible and the moment primal become feasible it is optimum.

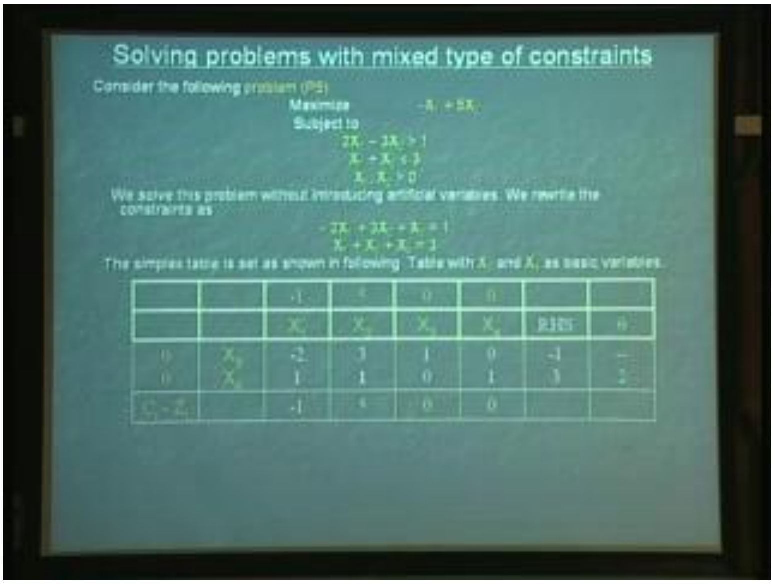
It is called dual simplex because at every iteration of this simplex, that dual of the problem that you are solving is feasible whereas in the regular simplex which you may now call as primal simplex, the primal problem that you are solving is feasible in all iterations. At the moment its dual becomes feasible and it is optimum. We need to understand that it is call dual simplex because at every iteration of this simplex algorithm, the dual to the problem that you are solving is feasible and represented by the optimality condition being satisfied. Therefore it is called a dual simplex algorithm. This comes under the category of what are called dual algorithms.

(Refer Slide Time: 38:49)



Now dual simplex is an example of a dual algorithm. A dual algorithm will have a feasible dual always or dual to the problem that you are solving is always feasible, applies complementary slackness and the moment primal becomes feasible, it is optimal to both primal and dual. The regular simplex also called primal simplex is a primal algorithm. Dual simplex is an example of a dual algorithm.

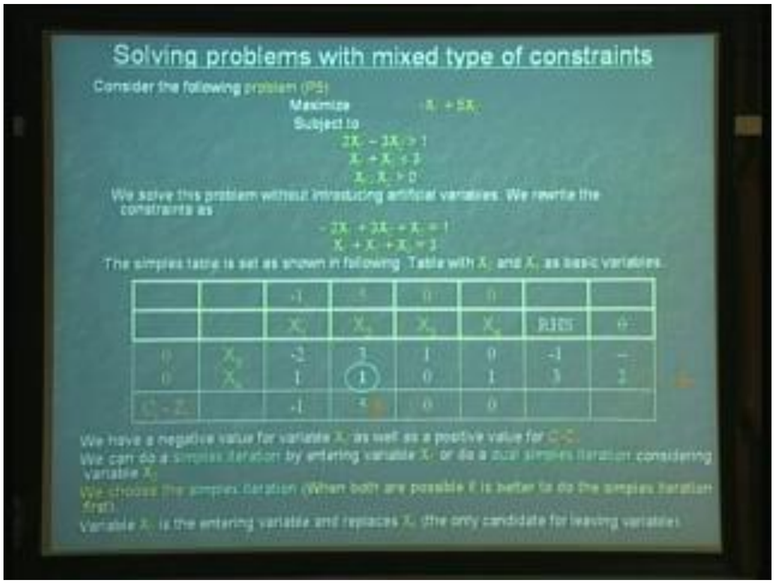
(Refer Slide Time: 39:36)



Let us go to another type of problem. We have taken a problem which has mixed type of constraint. You have a greater than or equal to type constraint. You have less than or equal to type constraint. You have all these. More importantly you have an objective function though it has maximized it does not have all positive coefficients. So far we have not looked at a problem that had a negative coefficient in the objective function. We looked at only after we convert it to a maximization problem, you could get negatives but we never had a mixed set of objective function coefficients. We never add a mixed set of right hand sides. If we have a problem like this then what do we do is, the simplest and the easiest thing to do is the second constraint $X_1 + X_2$ less than or equal to 3 is the kind of constraint that we want because the slack variable $+ X_4$ will automatically qualify to be a basic variable. The first constraint being a greater than or equal to type would now give us a negative slack which would not qualify normally. We would have added an artificial variable a_1 there and it is easy to start the simplex table with a_1 and X_4, use the big M method and solve it. After having learned the dual simplex which helps us to solve problems without introducing artificial variables, can we apply dual simplex to this problem? Right now we cannot apply dual simplex to this problem as it is because, when we when we write a – X_3 here and a + X_4 here, the combination X_3, X_4 has an infeasible right hand side as you can see here (Refer Slide Time: 40:33)

Right hand sides will be 1 0 0 1 with a – 1 and 3. When we write this $C_j – Z_j$, we write it and convert it. We have a maximization problem. If we start with X_3 and X_4, we have a – 1 and 5. So we encounter situation where neither the primal is feasible nor the dual is feasible. This 5 indicates that the dual is infeasible. A positive $C_j – Z_j$ would indicate that this variable can enter which means that the corresponding dual when I apply complementary slackness is infeasible. When we applied the dual simplex algorithm, we made sure that all these were less than or equal to 0 and the dual was feasible in all the iterations.

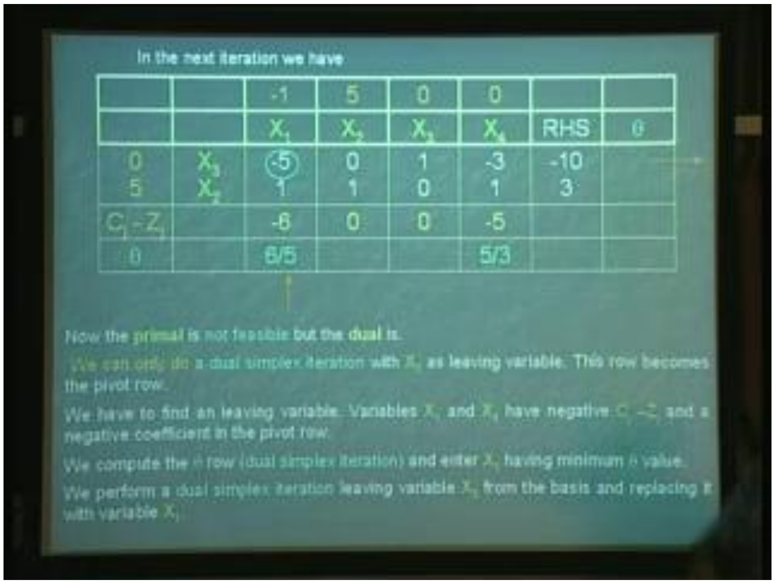
(Refer Slide Time: 42:18)



Now we come into situation where the dual is infeasible but we can still do this. So let us go back and see. You can do two things. You can leave this variable, treat this as a leaving variable first and then perform a dual simplex iteration, or you can enter this as your variable and perform a simplex iteration. You can do either. So what we are trying to show now is we have a negative value for variable X_3 as well as a positive value for $C_j – Z_j$ indicating that both the primal and the corresponding dual are infeasible. So you cannot entirely apply the simplex algorithm or entirely apply the dual simplex algorithm. If you want to solve this problem without artificial variable, you have to judiciously mix simplex iteration and a dual simplex iteration and proceed till the optimum. We show that through this example. You can actually do a simplex iteration by entering variable X_2 here with the 5 or you can do a dual simplex iteration by leaving out this.

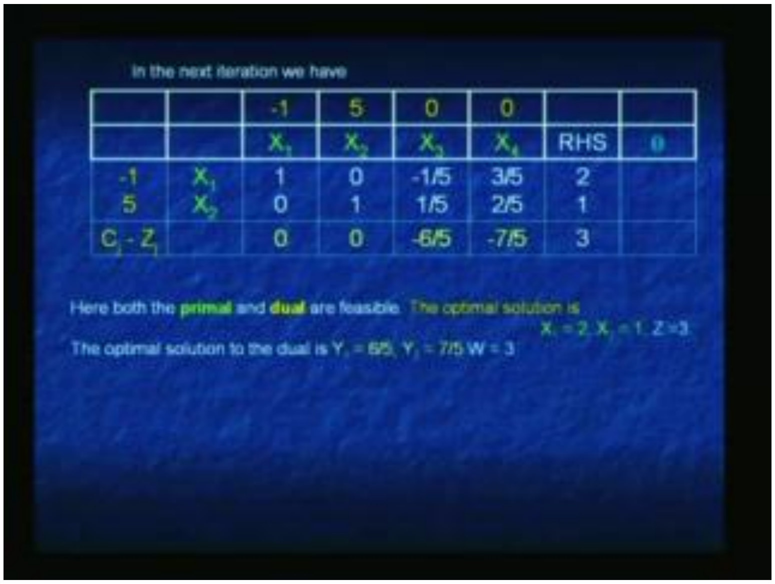
So we choose the simplex iteration. In fact there is a general thumb rule that when both the simplex and dual simplex iterations are possible, the thumb rule is, do the simplex iteration first.
We enter variable X_2 with 5 and perform a simplex iteration. For this you cannot have theta because in a simplex iteration you would want the feasibility to be maintained. You will not compute the theta for this. You will compute a theta only for this. You have right hand side 3, you have $X_2 = 1$, so the theta value is 3 and variable X_4 will leave the basis. This is 3 instead of 2 and variable X_4 will leave the basis. We do the next iteration. Next iteration is with X_3 and X_2.

(Refer Slide Time: 44:25)



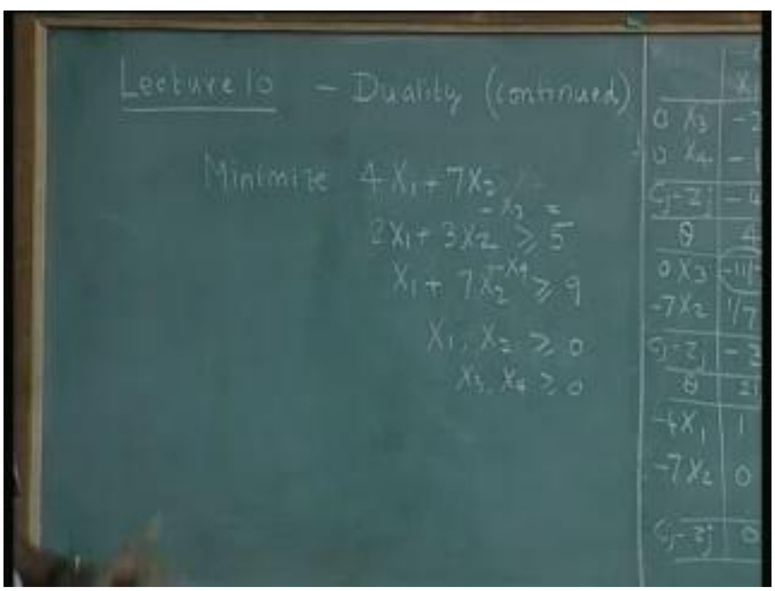
When we do simplex iteration, you now get into situation where, your $C_j - Z_j$'s are negative which means that dual is feasible but then the primal is infeasible because X_3 now has a negative value. Now in this iteration you can do a dual simplex iteration because your optimality condition is satisfied and the feasibility condition is violated. You can do a dual simplex iteration which is the only thing possible. Simplex iteration is not possible here because you do not see an obvious entering variable whereas you see an obvious leaving variable. So you do a dual simplex iteration by leaving out variable X_3. That is what is written here. The primal is not feasible but the dual is feasible. You can only do a dual simplex iteration with X_3 as a leaving variable. When we tried to compute the theta you will get 6/5 and 5/3, 6/5 being smaller. Variable X_1 will now enter. You will do a dual simplex iteration with variable X_1 replacing variable X_3 and that is shown in the next iteration.

(Refer Slide Time: 45:29)



Now you have $X_1 = 2$; $X_2 = 1$ with Z = 3 which is your feasible solution to the primal. The feasible solution to the dual will be $Y_1 = 6/5$; $Y_2 = 7/5$ and W = 3. You have a situation where your right hand sides are non negative. Primal is feasible. We have optimality condition satisfied. So dual is feasible. You have now reached the optimum to both primal as well as dual. The simplex method can also be used to solve situations where we can have mixed type of constraints.

(Refer Slide Time: 46:15)



When have mixed type of constraints and mixed coefficients in the objective function, as we saw in this example, you could have a positive and negative here. You could have a greater than or equal to constraint and less than or equal to type constraints. When you have an equation you may still be forced to use an artificial variable if necessary, because splitting the equation into two constraints complicates further because you are increasing the number of constraints and the effort will increase. We wouldn't choose to do that if you have only inequalities and if you have a mixed type of constraints, you can still use a judicious mix of simplex and dual simplex iterations and solve the problem. The only assumption that we have made here is, the problem has a solution and we have got the optimal solution but we could get into certain situations if for example a problem with mixed type of constraints with positive and negative coefficients in the objective, if the problem is unbounded or infeasible then we need a way to understand whether the problem is unbounded or infeasible. The usual conditions will still apply. For example even when you mix a simplex and a dual simplex iteration or even within a dual simplex iteration you might get into a situation where I can find a leaving variable, I may not be able to get an entering variable that can happen. Now that does not directly ensure unboundedness whereas those things are very well defined in a simplex iteration.

In a dual simplex iteration if we get into a situation where I have a leaving variable and I do not have an entering variable. That can happen. Such a thing would actually indicate infeasibility of the problem. We have to go back and define how we look at unboundedness and infeasibility with respect to the dual simplex or we look at unboundedness and infeasibility with respect to applying simplex and dual simplex alternatively as in the case of problems with different types of constraints and different positive and negative coefficients in the objective function.
Now both the dual simplex algorithm as such and the example that we saw here where we solve problems with mixed type of constraints, we have just tried to show that particularly in the second case when you have problems with mixed constraints, where it is possible to solve using a judicious mix of simplex and dual simplex iterations but then one has to be careful.

If there is an optimum, we will definitely get it but if the problem exhibits other things like unboundedness or infeasibility then we need to carefully define unboundedness and infeasibility rules for those iterations. Here again we need to be little careful to define the unboundedness and infeasibility but the equivalent unboundedness rule actually represents infeasibility in the dual simplex iteration. Let us spend a couple of minutes on understanding the simplex and dual simplex, the differences and so on. The regular problem that we had solved, i.e., the maximization problem which has all less than or equal to constraints and all greater than or equal to variables is very ideal for a straight simplex application because slack variables will qualify to be the basic variables.

We can solve it entirely by the simplex algorithm. A minimization problem with all positive or non negative coefficients in the objective with greater than or equal to constraints is a very ideal situation or ideal problem to use the dual simplex. If we have problems with a mixture of constraints and types of coefficients and so on, one would take the risk of solving it by judiciously applying simplex or dual simplex or one would follow a very safe way of solving it only by simplex algorithm by adding artificial variables. Both simplex and dual simplex algorithms solve linear programming problems extremely efficiently. We have also seen the relationships between the primal and dual that one is the primal algorithm and the other is a dual algorithm and so on. The next thing that we have to see is, if we can efficiently represent the simplex algorithm or the dual simplex algorithm.

For example if we have to write a computer program for the simplex algorithm, we will use the tabular form or are there better ways of solving the simplex algorithm other than using the standard tabular form that we have seen? We will see that in the next lecture