Advanced Operations Research

Prof. G. Srinivasan

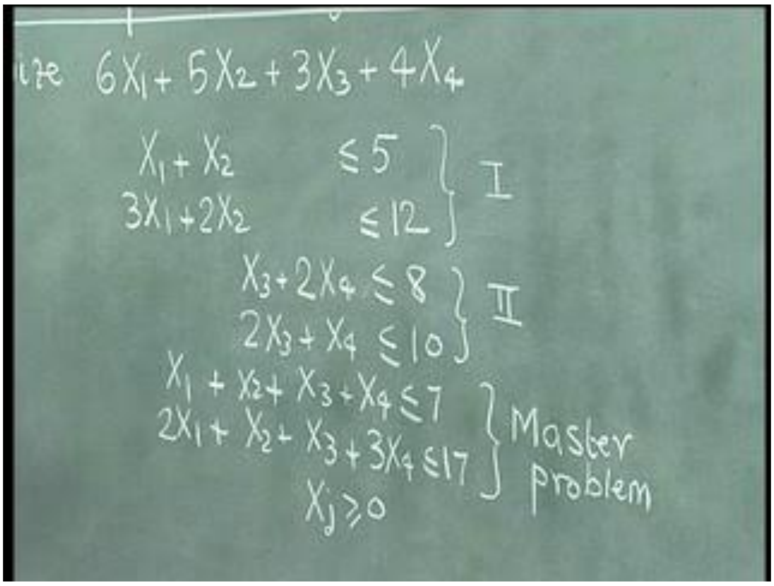
Department of Management Studies

Indian Institute of Technology, Madras

Lecture-6
Dantzig-Wolfe Decomposition Algorithm

Today, we consider the decomposition algorithm and we explain this decomposition algorithm using this example.

(Refer Slide Time: 00:23)



This problem is the linear programming problem, maximization objective with four variables and six constraints and all X_j greater than or equal to 0. We can solve this using any of the techniques of linear programming that we have learnt till now, but we are going to introduce a newer version of simplex or a new technique which is called the decomposition of algorithm which is going to be used to solve this problem.

Let us take a closer look at this set of constraints. There are six constraints, but we observe that the first two constraints involve only the first two variables. The next two constraints involve the next two variables, X_3 and X_4, and these two constraints have all the variables. One of the things we can do is that, if we relax the problem which means we remove some of the constraints by relaxing the problem. If we relax these two constraints and look at only the other four, though we realize that this problem can be split into two linear programming problems: one of which will be maximize $6X_1$ plus $5X_2$ subject to these two constraints; The other will be maximize $3X_3$ plus $4X_4$, subject to these two constraints, by removal of certain number of constraints from the problem, or by relaxing the problem by removing the constraints.

The given problem can be decomposed into more than one linear programming problem, more than one independent linear programming problem. So, we exploit that idea. The first thing is that, if we remove these two and solve the resultant linear programming problem and if it turns out that the optimum solution to the resultant problem satisfies this, then it is also optimum to the original problem. More often that will not happen. We will also have to consider these two constraints.

The method that we will adopt is first, remove these two constraints and treat them as two independent problems and then ensure that the solution obtained to those independent problems, also in some way or the other meet these constraints so that the optimum solution can be obtained.

Now, before we even proceed further, let us see, why we are doing this?

If we were to solve this as a linear programming problem, every iteration of the simplex method, would involve all the six constraints, so it would involve six basic variables and every iteration is to invert a 6 into 6 matrix. But, if on the other hand, we remove these two and then decompose the problem into two problems, each having two constraints then in each iteration in principle we are inverting a 2 into 2 matrix. We are actually inverting two 2 into 2 matrices. We also have to have some relationship between these constraints and the other one; therefore, there is one more binding constraint that is added.

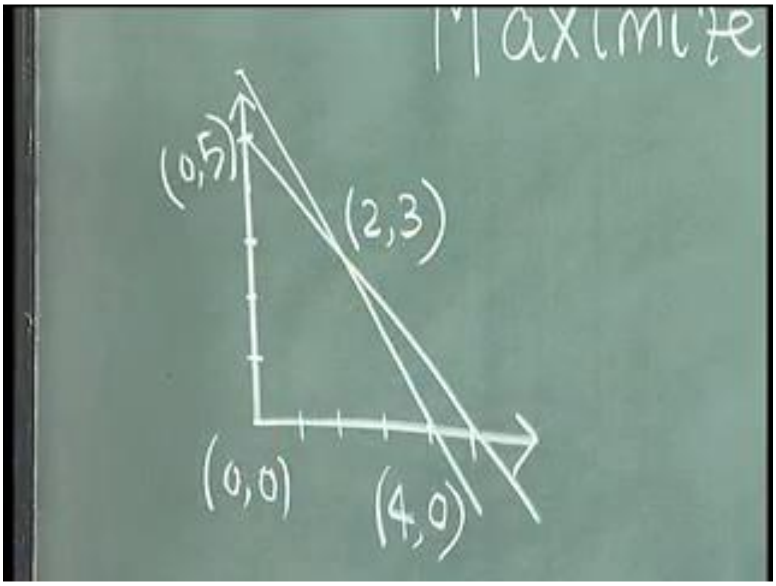
Effectively, we will be looking at 2 plus 2 plus 3, original problem has 6, but then we will be looking at 2 plus 2 plus 3. We also understand that inverting a 6 into 6 matrix is a little more computationally intensive compared to inverting a 2 into 2, another 2 into 2 and another 3 into 3. So, computing time required to solve this problem becomes lesser, particularly when we write computer programs and solve.

These two constraints which are taken out are called the master problem constraints, so these are called master problem constrains. These two are the sub problem constraints which we call as I and II. So, this is a sub problem constraint, this is another sub problem constraint and so on. Now, the two sub problems are maximizing $6X_1$ plus $5X_2$, subject to this and maximize $3X_3$ plus $4X_4$ subject to this.

Actually in every iteration we need to solve this as well as this. Instead, what we will do to make our computation simpler, we will simply note down the corner points associated with this and the corner point associated with this so that depending on the objective function which we will see we will modify slightly depending on the objective function by a simple substitution of the corner points. We can get the optimum solution quickly, because we are illustrating this on the blackboard, in a classroom environment.

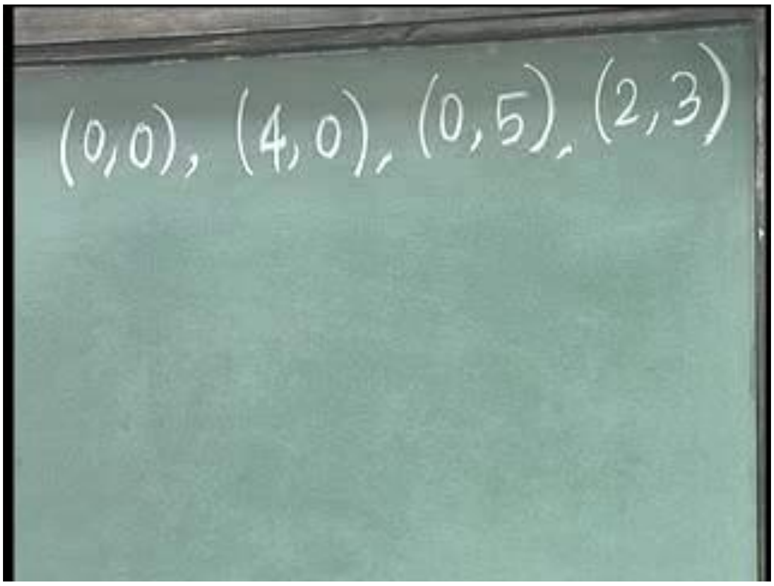
Ordinarily, every time the sub problem is solved, it should have been solved by the simplex algorithm. Let us first identify the corner points associated with this and the corner point associated with this because it is a 2 into 2 problem. We can actually solve it by the graphical method and try to find out the corner points.

(Refer Slide Time: 06:15)



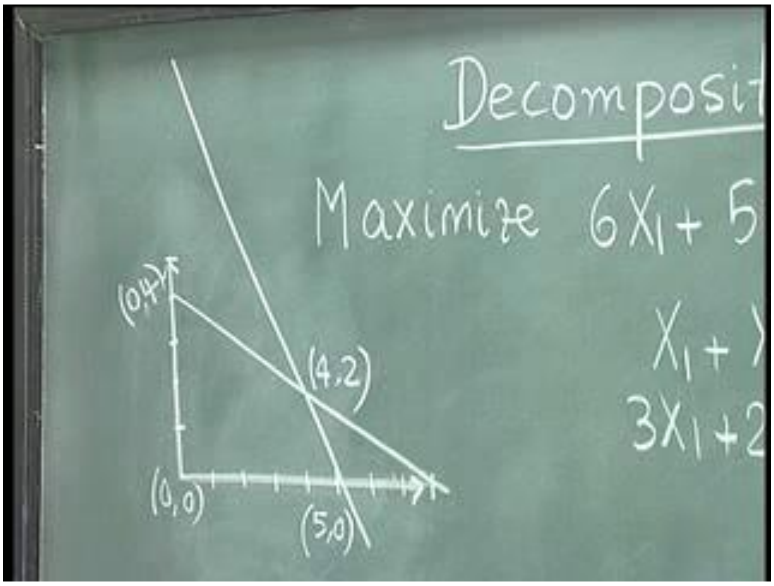
So, the first set of corner points. To do that, we have X_1 plus X_2, so this is X_1 plus X_2 equal to 5. $3X_1$ plus $2X_2$ equal to 12 will look like this. This is the corner point (4,0), this is the corner point (0,0), this is the corner point (0,5) and this will be the corner point (2,3). These are the four corner points associated with this.

(Refer Slide Time: 07:06)



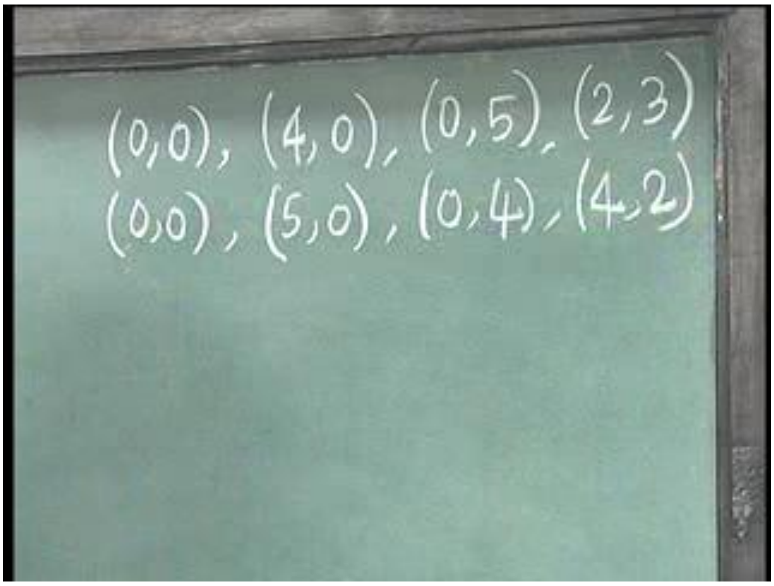
So, we will simply write the four corner points for our reference here; (0,0), (4,0), (0,5) and (2,3) are the four corner points associated with this. Now, let us find out the corner point associated with the second one.

(Refer Slide Time: 07:25)



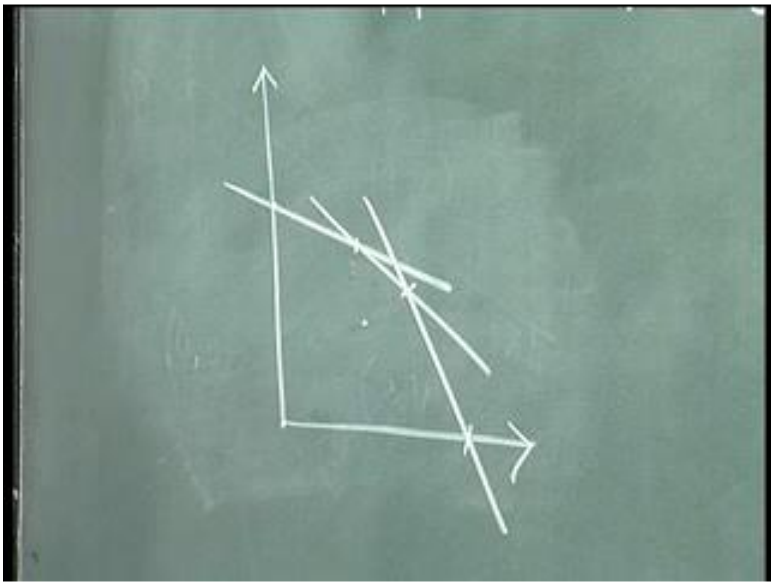
This will be the X_3 axis, this will be X_4 axis, X_3 plus $2X_4$ less than or equal to 8. $2X_3$ plus X_4 less than or equal to 10. This will be the point (5,0), this is the point (0,0), this is the point (0,4), this will be the point (4,2).

(Refer Slide Time: 08:31)



The four corner point associated with this are: (0,0), (5,0), (0,4) and (4,2), so, these are the corner points associated with. Now, let us understand one more principle before we apply the decomposition algorithm.

(Refer Slide Time: 08:53)



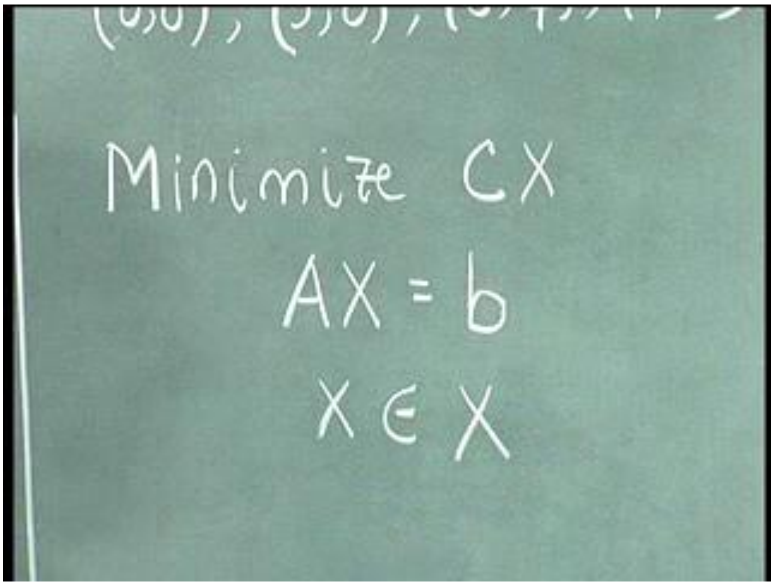
Now, suppose we have a feasible region like this for a given problem, let us assume that this is the feasible region. Now, if we add a constraint into this problem, the first thing that can happen is the feasible region can reduce or the feasible region will remain the same. These two things can happen. Let us assume that the feasible region reduces, so let us say, the feasible region comes somewhere here, because of the additional constraint. If this were the optimum solution before the introduction of the constraint and because of the introduction of the constraint this optimum solution becomes infeasible. Once again, it has to be in one of the corner points, now the new optimum solution has to be in one of these corner points, in these 5 corner points now that we have.

Now, if this were the optimum solution, then this was not a corner point in the first case, when the new constraint was not added, because the corner point was here. Now this point, if it is the optimum solution can always be represented as a convex combination of this point as well as this point because of the convexity property of the feasible region to a linear programming problem. Any point on the feasible region can always be represented as a convex combination of one or more corner points. If we expand it to higher dimension then it becomes convex combination of one or more corner points.

For example, this point can always be represented as $lambda_1$ into this plus $lambda_2$ into this where $lambda_1$ plus $lambda_2$ is equal to 1. Similarly, any point inside, can also be represented as a convex combination of the given corner points. As we add more and more constraints in the feasible region can reduce and the final optimum solution can even be a point which is inside this need not be on the boundary. Since, any point inside the region can be represented as a convex combination of corner points, it is enough to look at those corner points and try to identify which of the corner points are finally going to be part of that combination, which will eventually give the optimum solution.

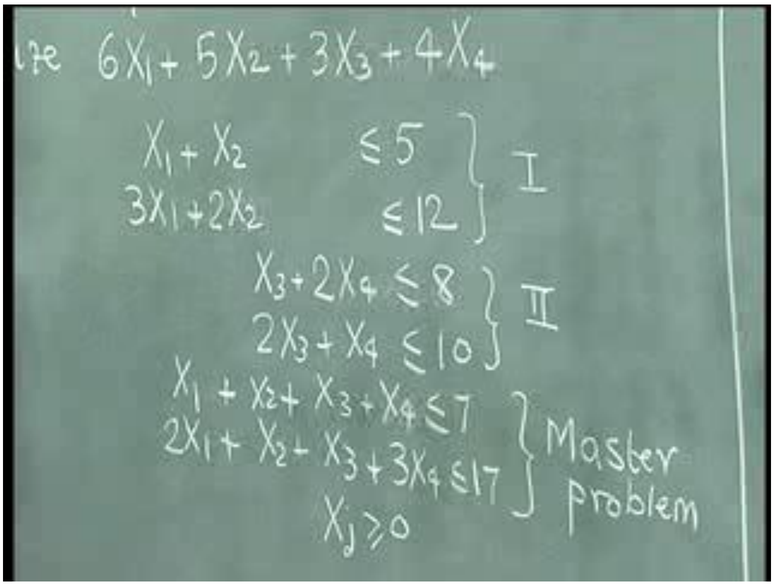
What we can do is let us say, we kind of remove this; we get this feasible region, whose corner points are known here. By the addition of these two, the optimum solution is going to come into some point inside the feasible region, which is bounded by these corner points. But please note that these corner points are given as X_1, X_2, X_3 and X_4, even though, the solution to this each point has its own X_1, X_2, X_3. X_4. Fact is by the addition of these two, if the optimum shifts and comes to a point inside the region, it still can be represented as a combination of these corner points that we are going to have. That is another idea we will use in the decomposition algorithm. Now, Let us look some little bit of theory with respect to the decomposition and then we proceed to solve this problem.

(Refer Slide Time: 12:54)



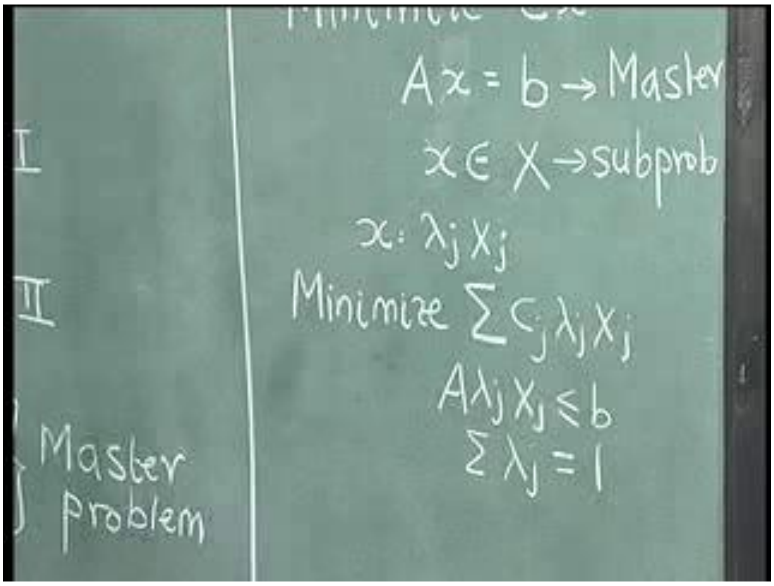
Let us assume that the given problem is actually a minimization problem, minimize CX, our example is maximization; we will see later how we modify this. But right now let us look at a minimization problem which is of the form minimize CX subject to AX equal to b and X is an element of X. What do we mean by that?

(Refer Slide Tim: 13:31)



What we mean by this is if we take this example, this is the maximization problem, but this can be written as a minimization problem by simply saying minimize minus $6X_1$ minus $5X_2$ minus $3X_3$ minus $4X_4$.

(Refer Slide Time: 14:05)

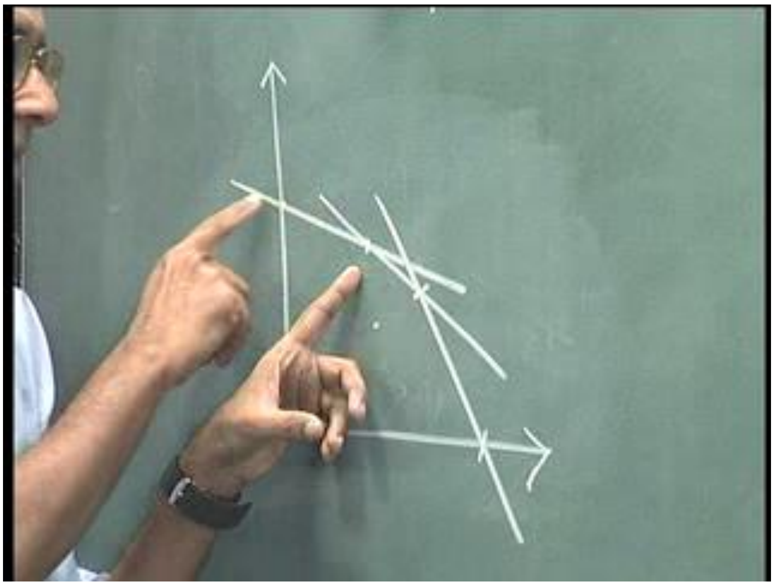


I have not even written X greater than or equal to 0, I also have to explain, what is this x and what is this X. So, we could use, let us call this as some small x minimise Cx, Ax equal to b, x belongs to X, let us use this notation that way. Now, when we apply this to this the minimize Cx is intact, because that is the same as this, minimize minus $6X_1$ minus $5X_2$ minus $3X_3$ minus $4X_4$. Now, Ax equal to b or actually the master constraints so these two are the ones that are associated with this Ax equal to b.

Now, x belongs to X, it simply means that this X represents the set of corner points of these two. In some sense x belongs to X, comes out of this. If we consider this problem, minimize Cx, subject to Ax equal to b, x belongs to X, this is the master problem constraints. These are the corner points of the sub problems.

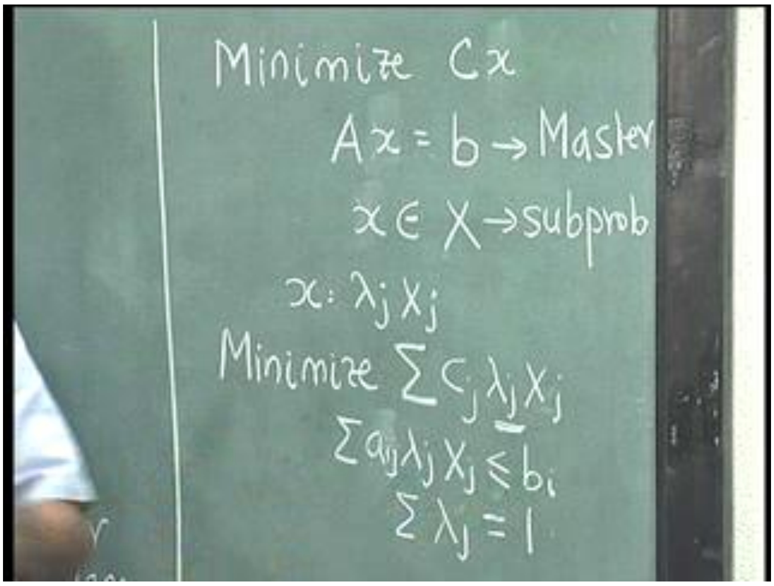
Now, we have also said that, every point is going to be represented as a convex combination of the corner points. What we are going to do is this x will be written as some lambda_j X_j, where X_j are these corner points. When we write this the problem will now reduce to minimize sigma C_j, lambda_j, X_j subject to A lambda_j X_j less than or equal to b, sigma lambda_j is equal to 1, because it is represented as a convex combination of the corner points.

(Refer Slide Time: 16:22)



For example here, when we said that this can be written as $\text{lambda}_1 X_1$ plus $\text{lambda}_2 X_2$, where X_1 and X_2 are corner points, lambda_1 plus lambda_2 equal to 1. This can also be written as $\text{lambda}_1 X_1$ plus $\text{lambda}_2 X_2$ plus $\text{lambda}_3 X_3$, lambda_1 plus lambda_2 plus lambda_3 equal to 1. When we start represent it, introduce the lambda and start representing it as the combination of the corner points, we need the linking constraint which is sigma lambda_j equal to 1.

(Refer Slide Time: 16:57)



Since every point inside this region also has to satisfy the master constraints so the point itself is lambda_j X_j, so it will satisfy the constraint sigma A lambda_j X_j less than or equal to b or equal to b depending on how the problem is defined. When we say here, it is equal to b, it means the inequalities have been converted to equations.

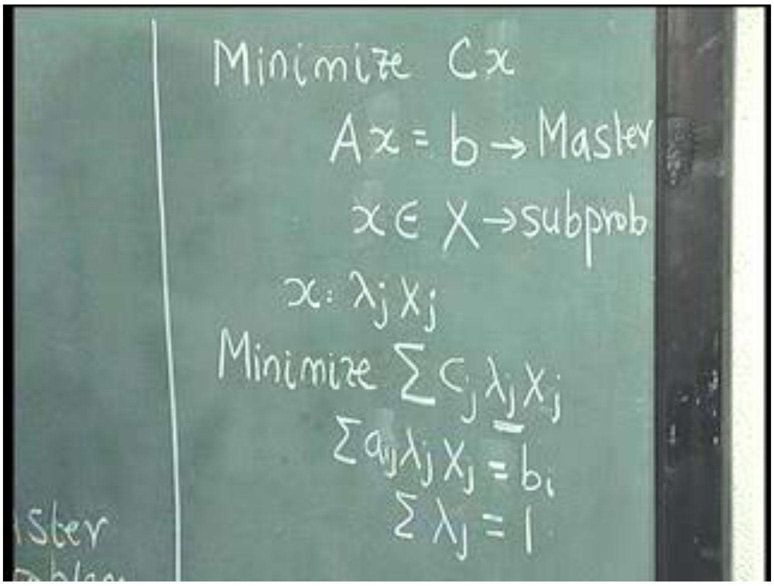
Here, we simply retain the inequalities as it is, we may say plus a slack variable equal to b or plus a surplus variable if it were greater than or equal to constraint; Therefore, the problem reduces to something like this. Now, for this problem, we are actually trying to solve the only difference being all these X_js are corner points, which are given here. Any point X, inside this region, which actually satisfies this set of constraints, optimizes this objective and has this lambda.

Now, this is the solution to the problem. The problem now becomes one of not solving for the X_js, but solving for the lambdas. The lambdas now become decision variables, because the X_js are known corner points, these are the corner points. So, the lambda has become the decision variables for this. Now, this one I have written in a certain form, we may put a summation here in which case we can put sigma a_{ij} lambda$_j$ X_j is equal to b_i. We can either write it as individual constraints or keep the whole thing as the vector.

In principle, what I am trying to explain here is that these X_js are the corner points, so at any point inside, which will be the solution, is now represented as sigma lambda$_j X_j$ and therefore this problem gets, rewritten in this format. I have also explained this equation versus inequality.

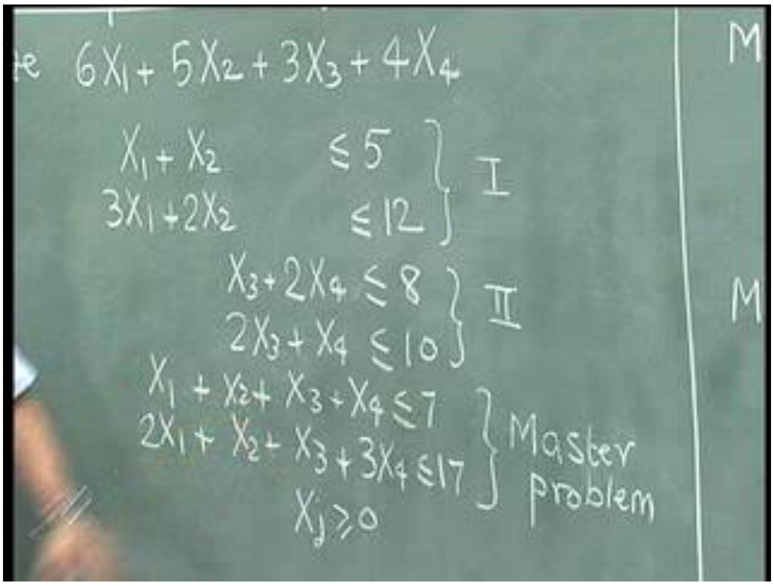
When I write an equation here, I assume that the inequalities have been already converted to an equation, where the addition of suitable slack variables or surplus variables. When I write an inequality here, I assume that we are yet to add, the slack variable or the surplus variable. If we want to be consistent with these two, we can even say that, X_j equal to b_i, which means, that we have added the slack variables and surplus variables associated with this.

(Refer Slide Time: 19:38)



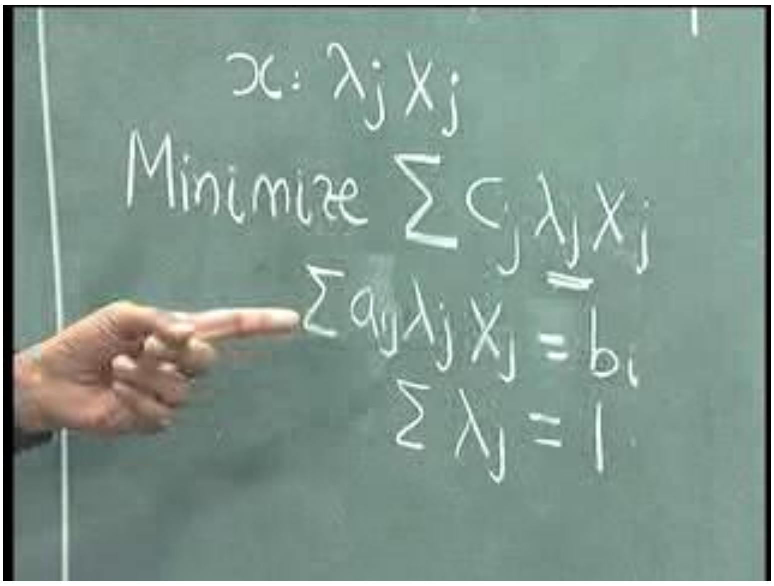
Now, as I said the focus now shifts on finding out the lambda_j which become the variables or which are essentially the weights that are associated with each corner point.

(Refer Slide Time: 20:15)



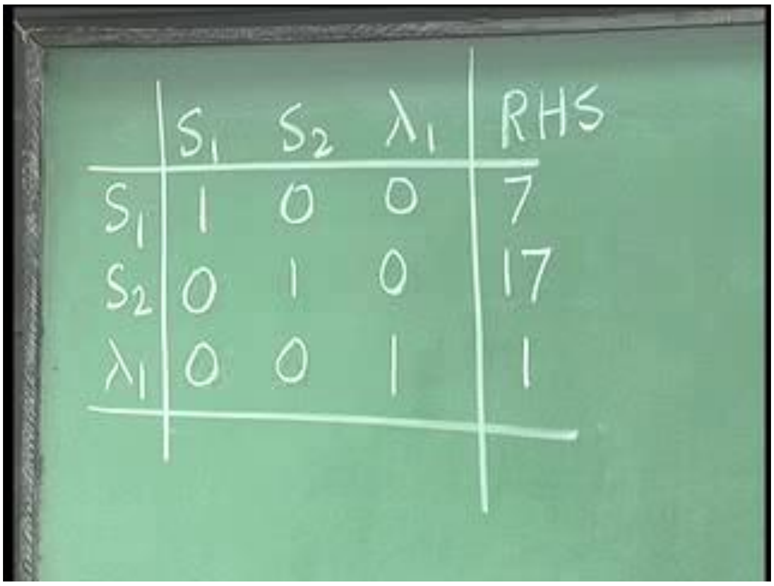
Now, we can start solving this problem by doing a few things, if we are able to get a feasible solution to the problem, for example, this is the convenient example, where all these constraints are less than or equal to constraints. Therefore the point 0, 0, 0, 0 is basic feasible for this. Particularly, when we leave out both these, we have four constraints on the point 0, 0, 0, 0 is basic feasible. We could start with lambda_1 equal to 0, 0, 0, 0, which is a corner point. So, we can start with lambda_1 for this. Now, these two are the master problem constraints and these two constraints are also less than or equal to constraints. The corresponding slack variables can be added here. We can add an S_1 here and we can add an S_2 here, so S_1 and S_2 automatically qualify to be basic variables.

(Refer Slide Time: 21:09)



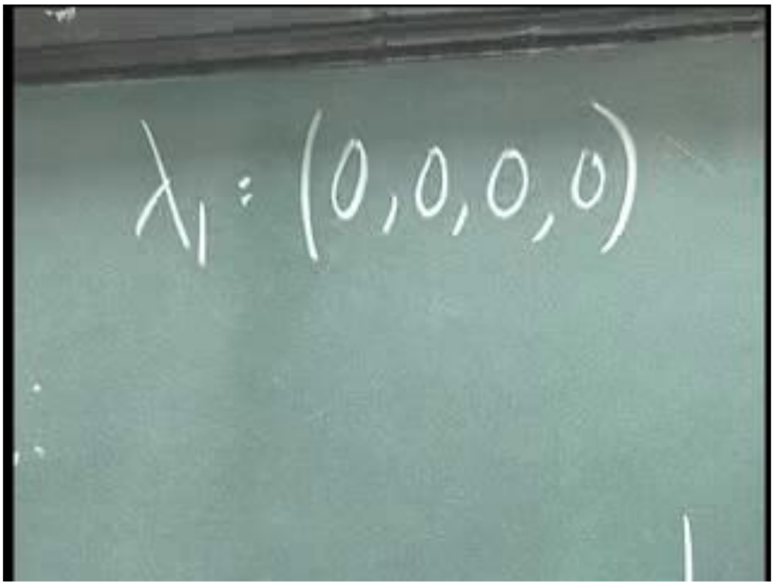
So, we can start solving this problem, with the basis, which comprises of S_1, S_2 and $lambda_1$. The first basis we can have for this is as follows.

(Refer Slide Time: 21:18)



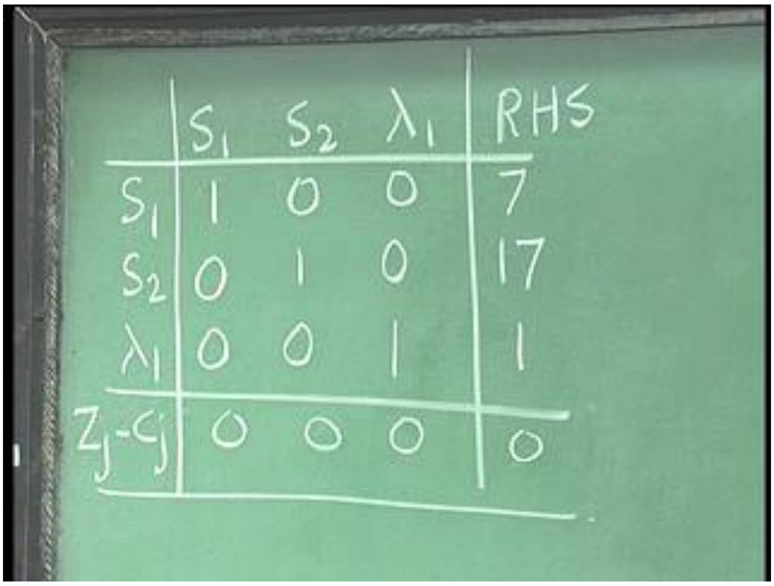
It is given by S_1, S_2, lambda$_1$ still have S_1, S_2, lambda$_1$ and right hand side we will have the identity matrix 1 0 0, 0 1 0, 0 0 1, with values b_i, b_i are 7 and 17. The right hand side values are 7 and 17 and for this constraint, the right hand side value is 1, so it is 1.

(Refer Slide Time: 22:05)



We have also defined lambda$_1$, this is our first starting point, it will be 0, 0, 0, 0, which comes out of these two. Now, because we have a basic feasible solution with 0, 0, 0, 0, we can start this, otherwise we may have to introduce an artificial variable for this and then begin with the artificial variable. Right now, we are going to solve this numerical example and the point 0, 0, 0, 0 is basic feasible. We simply start with the lambda$_1$ which is 0, 0, 0, 0, which takes weight equal to 1. The current basic feasible solution to the problem is actually the solution 0, 0, 0, 0.

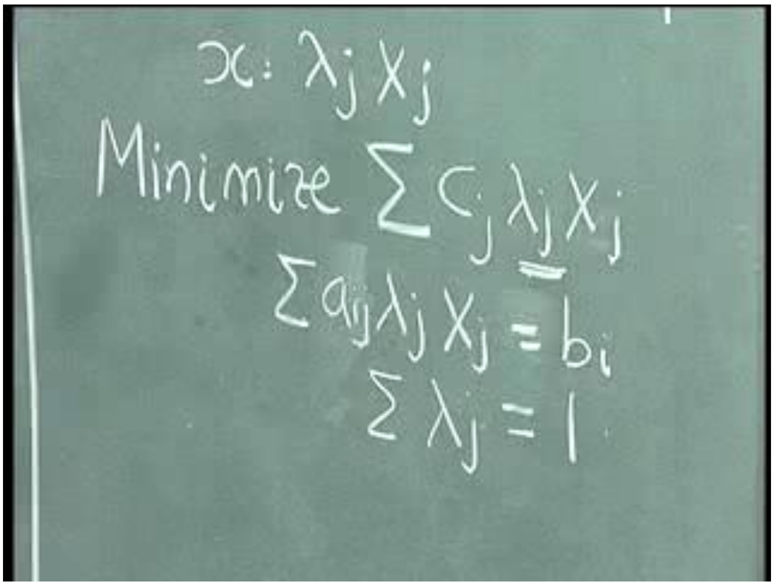
(Refer Slide Time: 22:54)



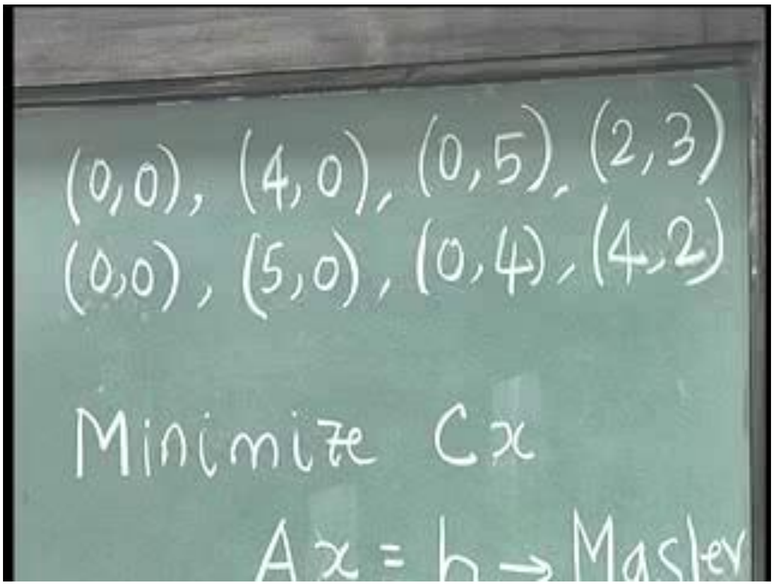
We have to find out C_j minus Z_j or Z_j minus C_j, which will be 0 0 0 and 0 here. Now, this is the starting basis as far as this problem is concerned, which means this basic feasible solution has one corner point which is the point 0, 0, 0, 0. Right now, this has a solution with Z equal to 0, so if we substitute 0, 0, 0, 0 here, we will get the solution Z equal to 0.

We need to verify whether this solution is optimum. Now in order to verify whether this solution is optimum we need to find out whether there is an entering variable. We have not written the entire table, we have written only that part of the simplex table which contains the basis matrix b, the right hand side and the Z_j minus C_j under b which can always give us the dual variables. We know by now, that Z_j minus C_j the way the notation that we have used, particularly in the earlier course on the fundamentals of O.R, where we covered linear programming extensively. We also mentioned that if we use Z_j minus C_j and if we do not have artificial variables, exactly under the identity matrix, the beginning identity matrix we can always read the solution to the dual of this problem. This helps us in reading the solution of the dual directly.

(Refer Slide Time: 24:59)



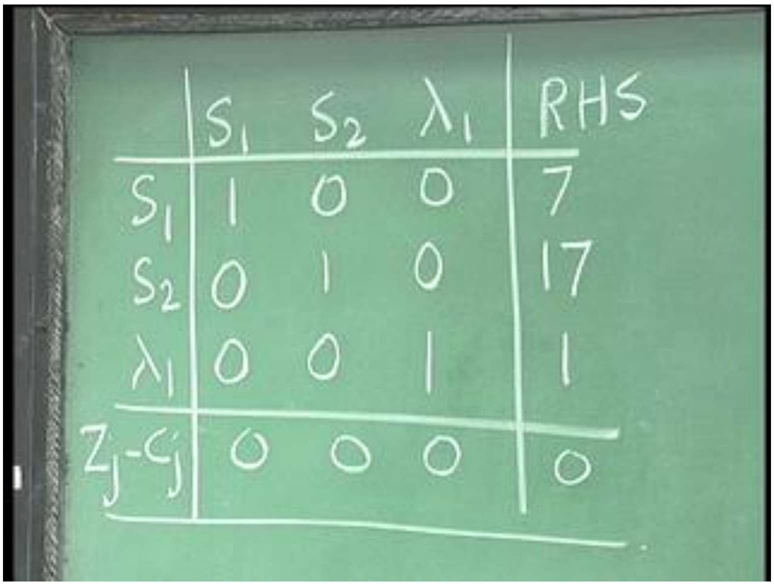
(Refer Slide Time: 25:10)



Now, several corner points are possible, in fact, right here, there are four here and there are four here, sixteen corner points are possible. Now, we have to check for all the sixteen C_j minus Z_js. If they cannot enter, then we have reached the optimal solution. So, one way is to generate all the sixteen and then try and find out the C_j minuss Z_j associated with all the sixteen.

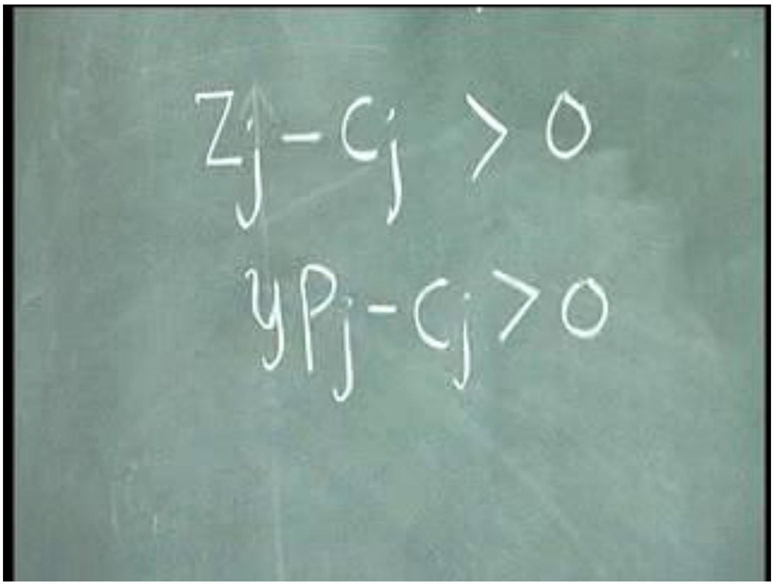
For example, we had a much larger problem, it was decomposed into three sub problems and let us say, there were five corner points in each, then we are looking at 5 into 5 into 5, 125 variables. We do not want to store all the corner points and then verify, whether these corner points can enter. Instead, what we do is we try and follow the column generation idea that we saw in the cutting stock problem. Then check that if there is a corner point that can enter the basis, if there is a lambda that can enter the basis, because each corner point is associated with the lambda. If there is a lambda that can enter the basis, we now generate that corner point and say that this corner point can enter the basis with the certain lambda. We follow the column generation idea and we do not explicitly store all the combinations of the corner points.

(Refer Slide Time: 26:36)



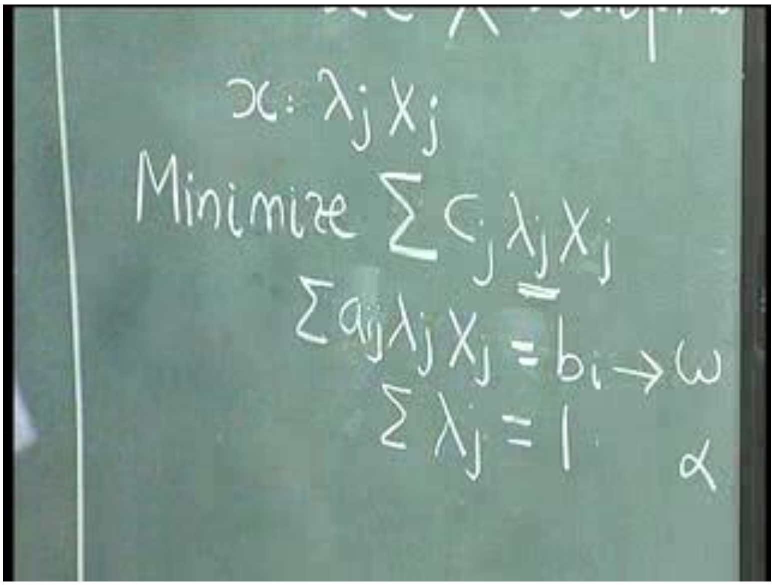
How do we generate, an entering corner point or entering lambda. In order to enter, we have we need to make sure that Z_j minus C_j, so, it is a minimization problem.

(Refer Slide Time: 26:49)



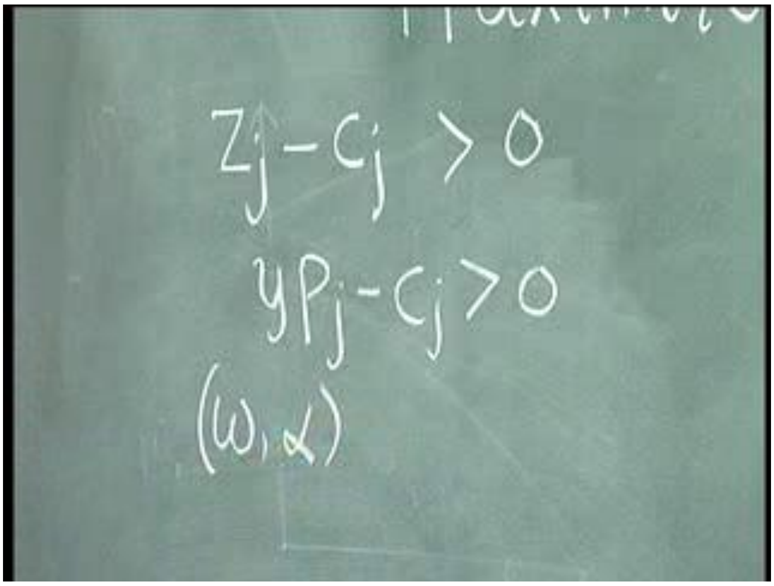
Z_j minus C_j, since it is a minimization problem, a positive Z_j minus C_j will enter the basis. We have seen for maximization problem, a Positive C_j minus Z_j will enter. For a minimization problem, a positive Z_j minus C_j will enter. Please remember, we have defined this problem as the minimization problem. It is only incidental, our numerical illustration is a maximization, but this problem is a minimization problem where a positive Z_j minus C_j will enter. If there is a j, if there is a corner point, whose Z_j minus C_j is greater than 0, then such a j can enter. Now, what is this Z_j. Z_j equal to yP_j, Z_j is always equal to C_B b inverse P_j, so that will be equal to yP_j minus C_j is greater than 0 where y is the associated dual variable.

(Refer Slide Time: 28:01)



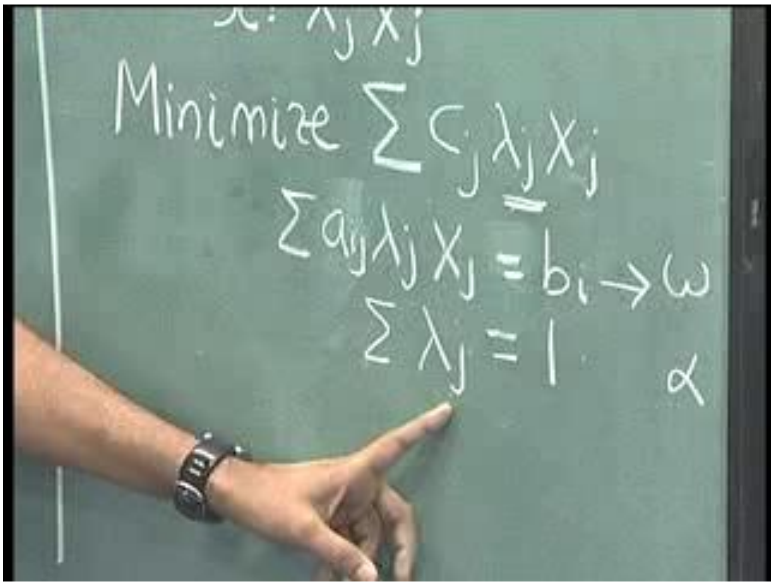
Now, in this problem, if we look at this primal, here we have as many constraints as the master constraints. In our illustration two constraints, this will give rise to two dual variables. This is always a single constraint, so this will give rise to one dual variable. We will now, instead of calling the dual variable by y which is the customary notation, we would now introduce dual variables w and alpha where since there are two constraints here we will have w_1 and w_2 and there is one constraint here we will have alpha.

(Refer Slide Time: 28:45)



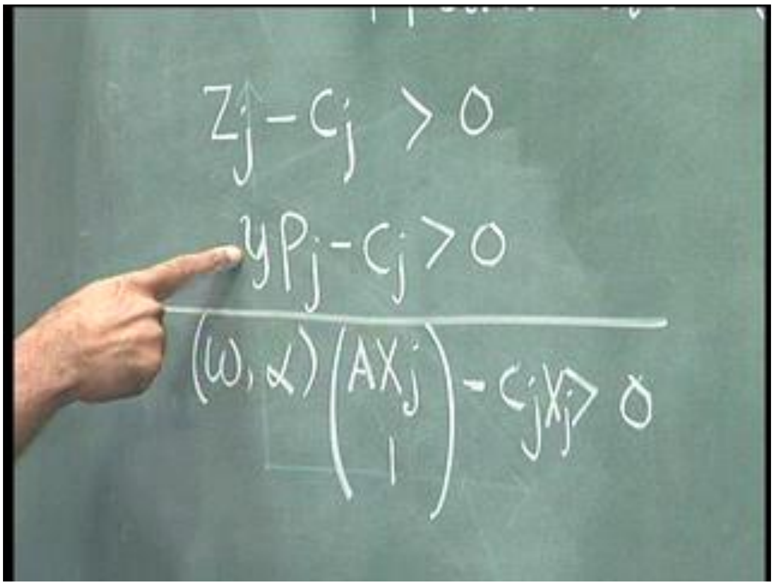
What happens here is this y becomes (w, alpha). What is P_j? P_j is an entering column yP_j, C_B b inverse P_j, so yP_j is entering column.

(Refer Slide Time: 29:09)



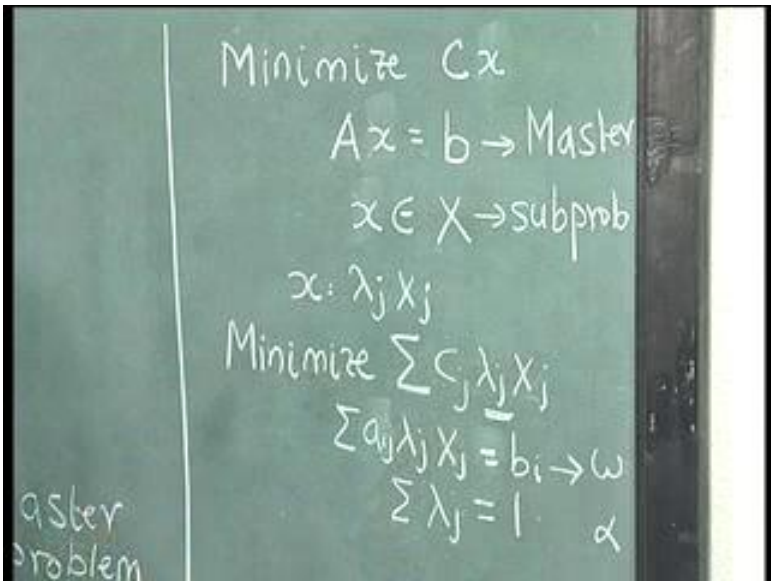
The entering column right now, any entering column here, will be of the form, aX_j, 1 because the $lambda_j$ is the variable. So you need to take this variable outside, so the entering column will be of the type aX_j, 1.

(Refer Slide Time: 29:23)



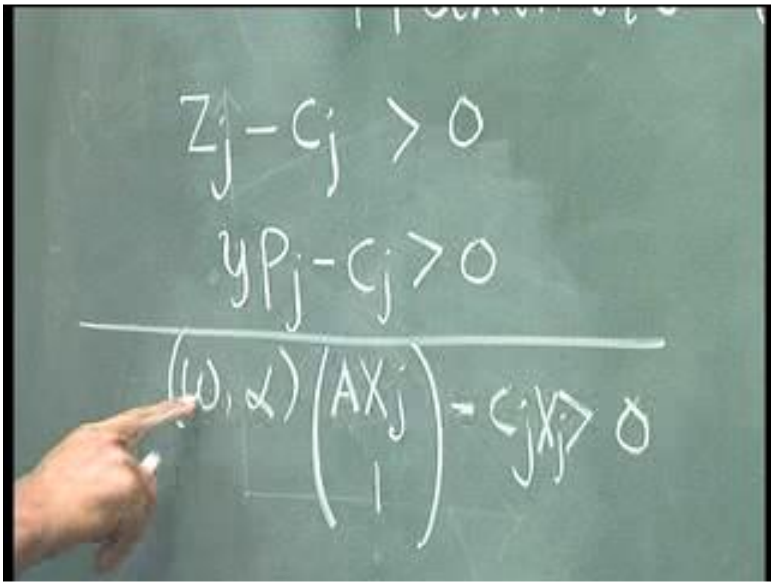
You will have w alpha into AX_j, 1, minus C_j should be greater than 0 and this is C_j now becomes C_jX_j from here, because $lambda_j$ is the variable. So, C_j becomes $C_j X_j$ greater than 0. Let me explain this again. In order for a new variable to enter the basis, because it is minimization problem, in general, Z_j minus C_j should be greater than 0. In general Z_j is written as yP_j minus C_j greater than equal to 0, where y is the dual variable, P_j is the entering column and C_j is the objective function coefficient of the entering variable, now, all these are general equations. We are adapting this to the problem on hand.

(Refer Slide Time: 30:28)



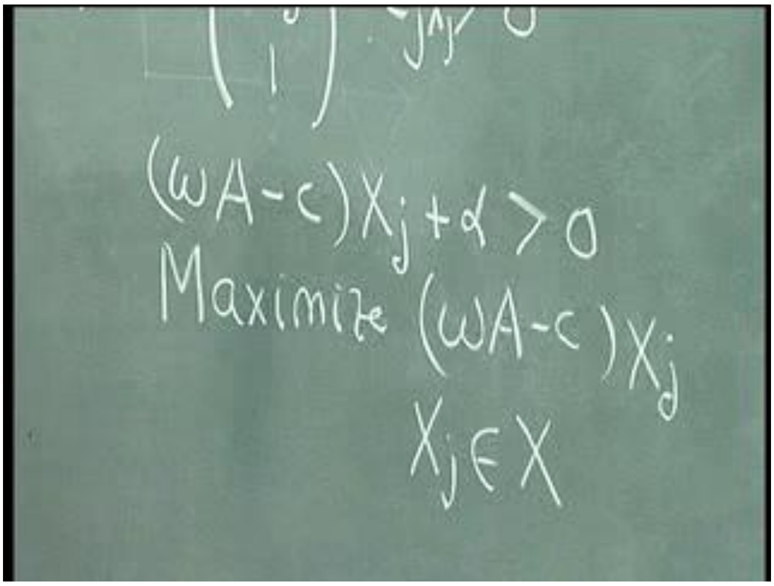
Now, because our problem is structured specifically this way there are as many constraints here as in the master problem where we have to introduce dual variable w, this will be w_1, w_2, w is the vector, this is always a single constraint, so introduce an alpha. So y becomes w alpha. If there is an entering lambda_j, lambda_j is the variable, X_j is not, so if there is an entering lambda_j, then the coefficient will be aX_j 1, so we will get aX_j 1.

(Refer Slide Time: 31:01)



Remember that, this is in matrix, this is the single thing so there will be two ws. There will be two values coming here, there is 1 alpha, there is just a 1, minus C_j is the objective function coefficient of the entering variable. The entering lambda will have an objective function coefficient of CX_j or C_jX_j greater than or equal to 0.

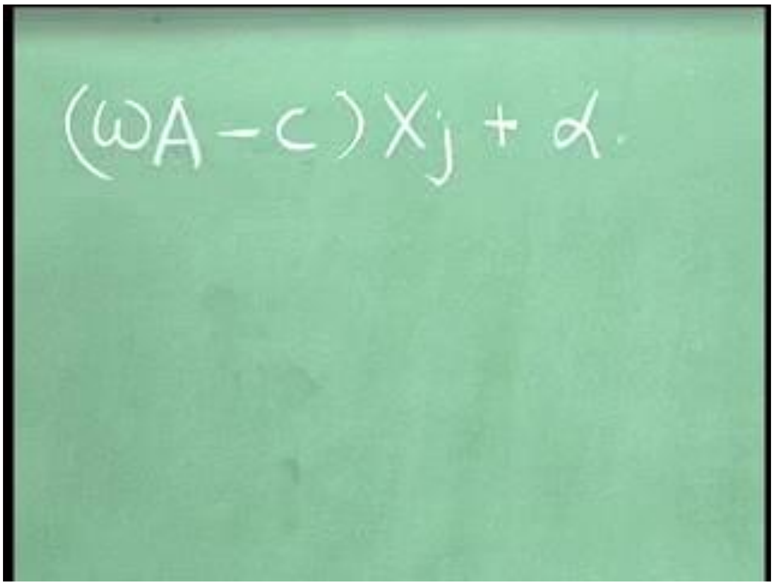
(Refer Slide Time: 31:29)



Now, this can be expanded to wA minus C into X_j plus alpha greater than 0. We also know that this X_j is the corner point corresponding to this and this, because every X is written as lambda$_j$Xj, where X_j is the corner point. Therefore, because X_j is the corner point, X_j has to satisfy this and this, set of constraints. What we want to do now is if there is an X_j, which satisfies this and this and has a value wA minus CX_j plus alpha greater than 0, then such a corner point can enter the basis with an associated lambda$_j$.

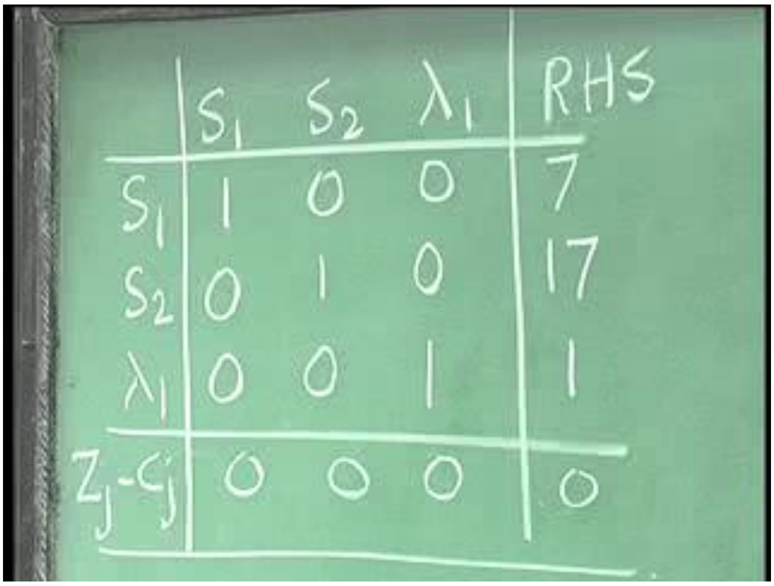
We do the usual thing of this being strictly greater than, we say that we now want to maximize wA minus C into X_j, subject to X_j belongs to some X, where this X is the set of corner points. Then, we find out such an X_j and then enter the corresponding lambda$_j$ into the solution. For a given objective function this is equivalent to solving the two sub problems. With this, bit of theory, let us go back and see whether there is an entering corner point whose lambda has to enter the basis.

(Refer Slide Time: 33:44)



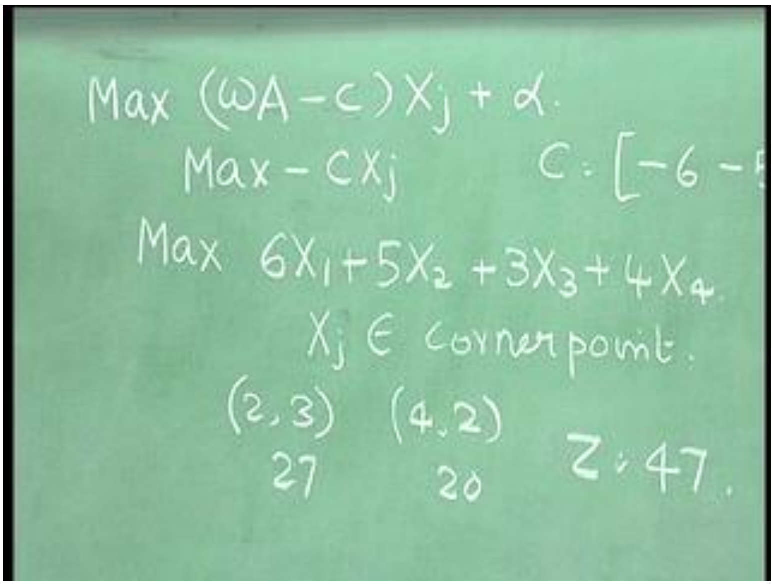
To do that, we go back and now check. We want to find out wA minus C into X_j plus alpha.

(Refer Slide Time: 33:57)



So, let us write this wA minus c…. Now, from this solution. We also said that this being a simplex table and this being the identity matrix, there are no artificial variables and we are computing Z_j minus C_j, Z_j minusminus C_j will automatically give us the values of the dual. These two values will be w_1 and w_2, because they correspond to S_1 and S_2. S_1 and S_2 were obtained from this and w is the dual variable for this, so w_1 and w_2 are seen here, alpha is seen here, alpha corresponds to this. Alpha corresponds to this lambda$_1$. Alpha is seen here so from this right now, the ws are all 0.

(Refer Slide Time: 34:40)



We want to maximize this right now, the ws are all 0, so, wA minus C X_j, they are all 0, plus alpha. Right now, we will have maximize minus CX_j. Alpha is also 0. Now, remember the original problem is minimize Cx, this problem has to be written as minimize minus 6 minus 5 minus 3 minus 4. The C vector is minus 6 minus 5 so this is C. Because the given maximization problem has to be written as a minimization problem, because the theory this is derived for a minimization problem, so the C vector becomes minus 6 minus 5 minus 3 minus 4.

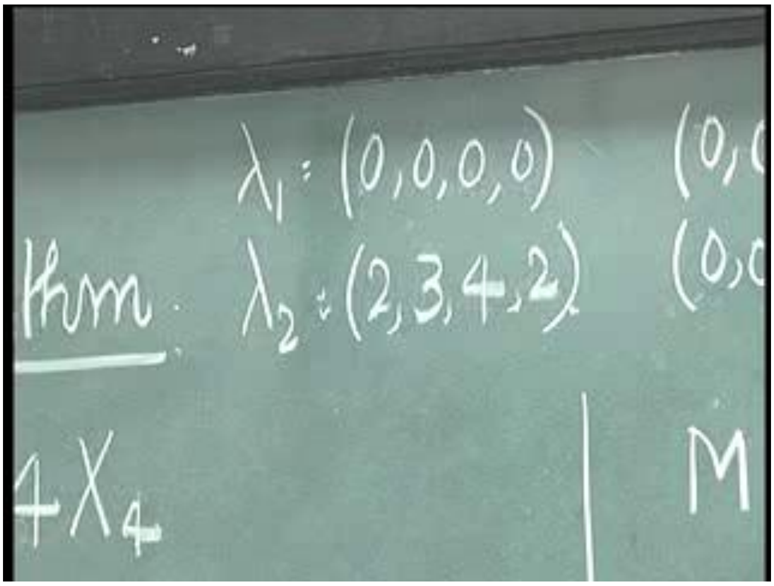
Maximize minus CX_j will now become maximize $6X_1$ plus $5X_2$ plus $3X_3$ plus $4X_4$, subject to X belonging to this. So, this is decomposed into two problems. The first problem being maximized $6X_1$ plus $5X_2$, subject to this constraint and the second one is to maximise $3X_3$ plus $4X_4$, subject to this constraint. So, we now have to solve two linear programming problems.

One is a 2 into 2 problem, the other is a 2 into 2 problem, to get the solution to this, subject to the condition, that X_j belongs to corner point. Because these two are linear programming problems and we know the corner points of the feasible region. Ordinarily, this is not the best way to solve this; each one has to be solved as an LP as we go through iterates. Because we are solving a much smaller problem and each of the sub problems has only two constraints and it is a 2 into 2, we have simply stored the corner points, just to make the ease of computation possible for us.

In practice when we are actually solving a very large LP, we will not a priori compute all the corner points and keep it. At every stage, we will be solving a linear programming problem within the decomposition. Let me again repeat, only to make the computation simple, I have just written down all the corner points.

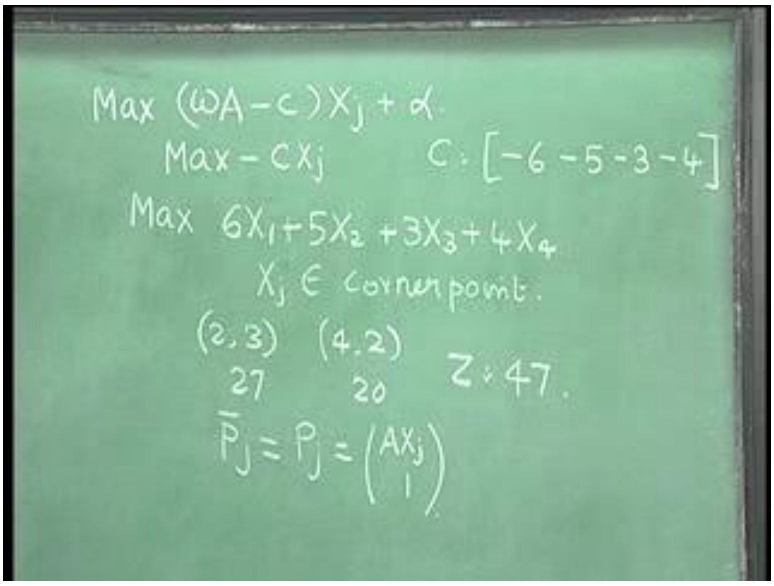
Since we know all the corner points, we can go back and find out what is the best value. So (0, 0) will give 0. (4, 0) will give 24. (0, 5) will give 25 (2, 3) will give 27, 6 into 2 is 12 plus 5 into 3 is 15. This gives the optimum solution (2, 3). Second one is $3X_3$ plus $4X_4$, so $3X_3$ is 15, here it is 16, now 4 into 3 is 12, plus 2 into 4 is 8, is 20. So, (4, 2) is optimal, this is 27, this is 20, so Z equal to 47. So value of Z is equal to 47 and the corner point (2, 3, 4, 2) enters the basis. So, the first corner point that will enter the basis is lambda_2 which is (2, 3, 4, 2).

(Refer Slide Time: 38:44)



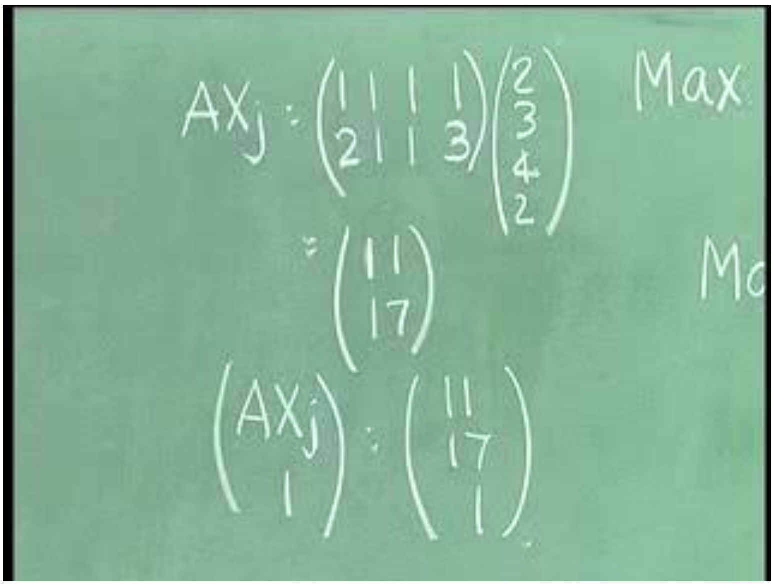
We now know the entering corner point which is 2, 3, 4, 2, but we have to find out the leaving variable.

(Refer Slide Time: 38:59)



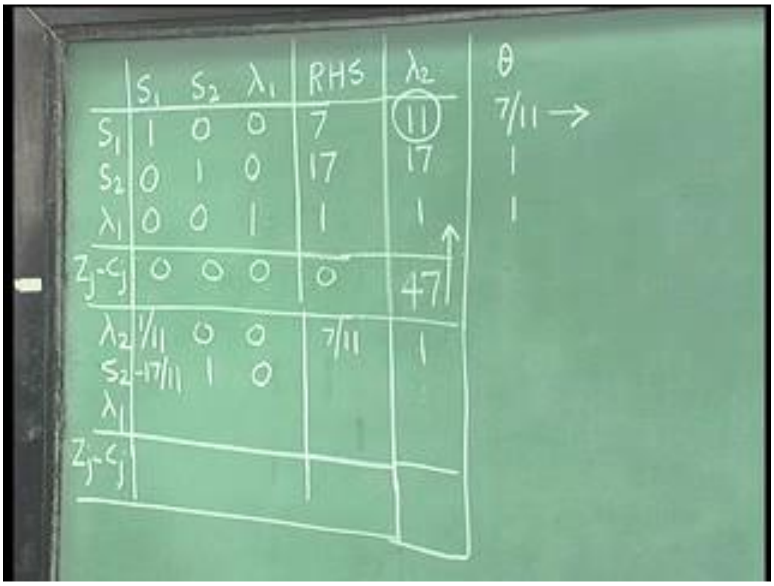
In order to find out the leaving variable, we need to find out $Pbar_j$. So, $Pbar_j$ is equal to b inverse P_j and this is always b inverse, so $Pbar_j$ equal to b inverse P_j. Right now, b inverse is I, so $Pbar_j$ equal to P_j. What is P_j? Any entering P_j is AXj, 1. So, this is P_j, this is equal to AXj 1. Now we have to compute AXj and 1.

(Refer Slide Time: 39:54)



AX_j is equal to, A comes from the master problems, please note this is the master constraint; this is the AX_j equal to b. So A is 1 1 1 1, 2 1 1 3, 1 1 1 1, 2 1 1 3 into X_j is 2 3 4 2 so this gives us 2 plus 3 is 5 plus 4 is 9 plus 2 is 11, 4 plus 3 is 7 plus 4 is 11 plus 6 is 17. So AX_j 1 is 11 17 1.

(Refer Slide Time: 41:06)

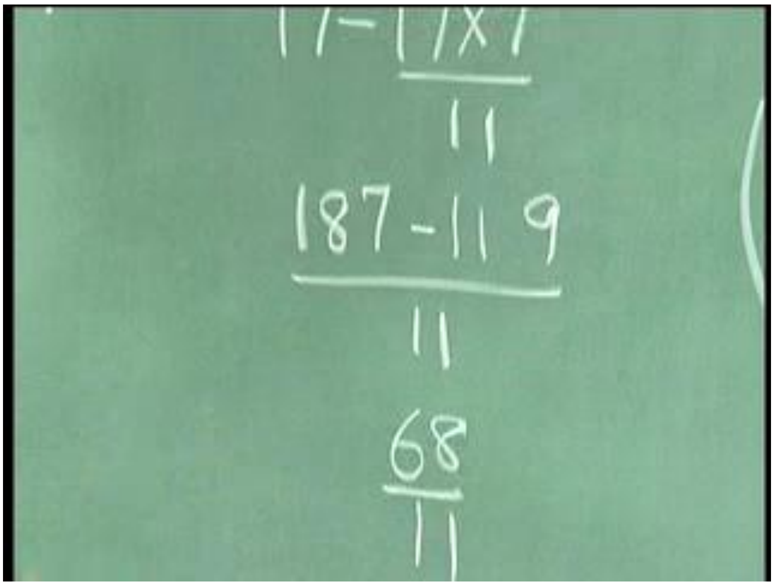


We have found out the entering column, which is Pbar$_j$, so we just write the entering column here. This is our lambda$_2$, which is 11 17 and 1 with value equal to 47. The positive Z_j minus C_j will enter the basis, so this lambda$_1$ with lambda$_2$ with 11 17 and 1 enters the basis. Now, we have to find out the leaving variable and to find the leaving variable, we will compute a theta. Note that, theta is right hand side divided by the entering column. So, it is 7 divided by 11, 17 divided by 17 and 1 divided by 1, so minimum theta leaves the basis. Therefore this will leave the basis and this is your pivot element.

We need to perform one simplex iteration. So we will go back here, to continue with the simplex iteration. Now, the variable S_1 is replaced by the variable lambda$_2$, so you have lambda$_2$ S_2 and lambda 1 with Z_j minus minus C_j. Now, divide every element of the pivot by the pivot element, so we need 1 0 0 here, so divide by the pivot element, you get 1 by 11 0 0, 7 by 11, 1.

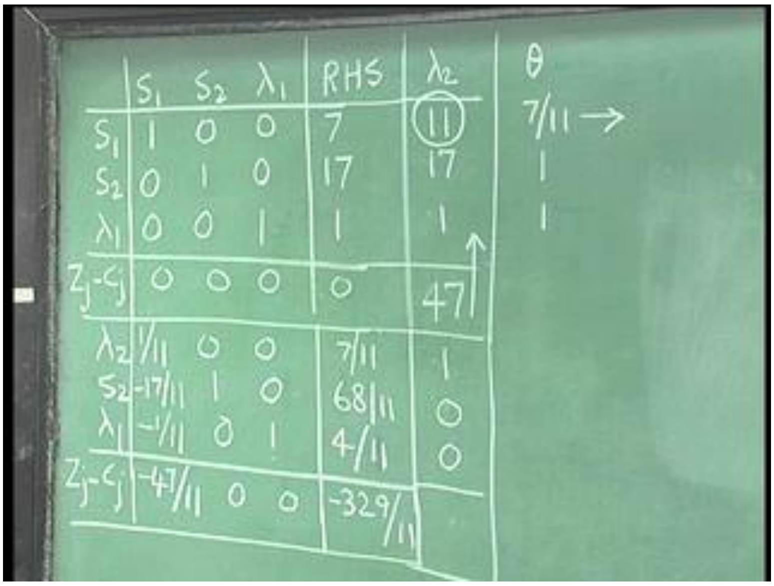
Now we need a 0 here, so this minus 17 times 1 will give me 0. So 0 minus 17 into 1 by 11, is minus 17 by 11 1 0. This minus 17 times this, 17 into 7 is 119. So, 17 minus 119 by 11, 17 into 11 is 187.

(Refer Slide Time: 43:45)



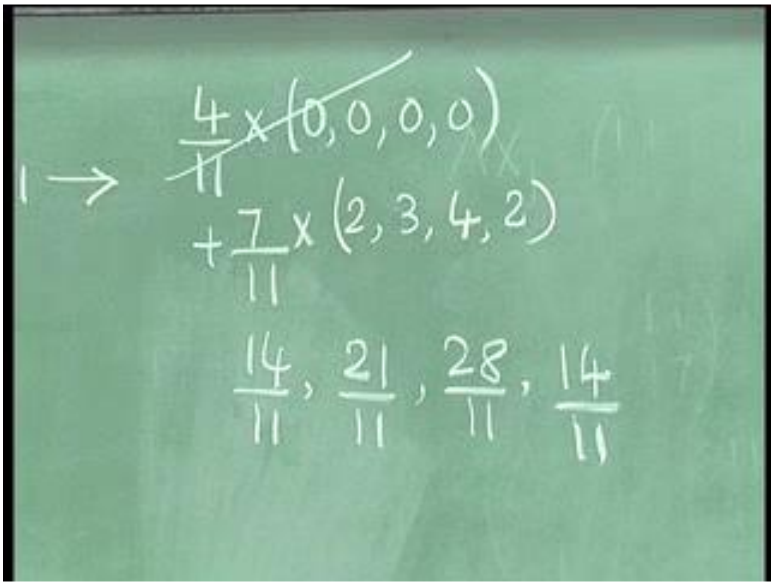
17 minus 17 into 7 by 11, this is 187 minus 119 by 11 which is 68 by 11.

(Refer Slide Time: 44:10)



We get 68 by 11 here. This will become 0 and we need another 0 here, so, this minus this will give us 0. So, minus 1 by 11 0 1, 1 minus 7 by 11 is 4 by 11 and a 0 here. We also know that, we can do the same kind of row operation to try and get a new Z_j minus C_j, what we need is a 0 here. This minus 47 times 1 is 0, so this minus 47 times, is minus 47 by 11 0 0. This minus 47 times, so 47 into 7, is 329. We get minus 329 by 11. This is the solution.

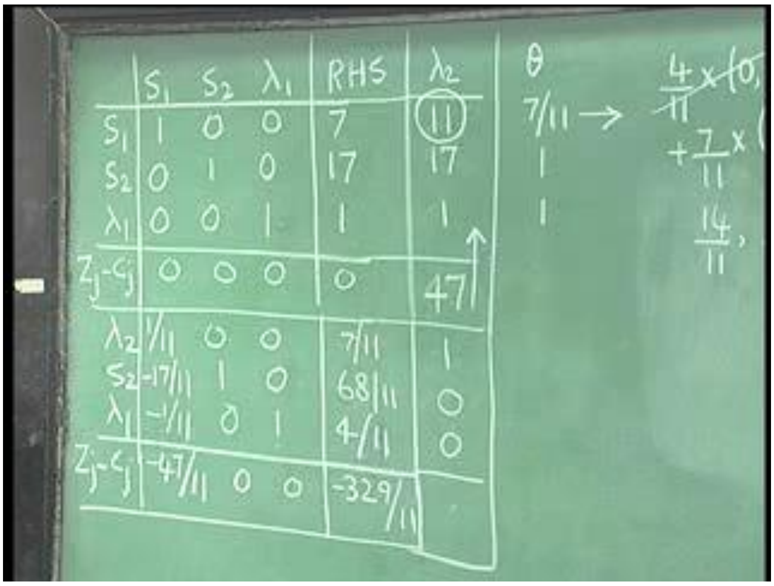
(Refer Slide Time: 45:40)



Now, what is this basic feasible solution tell us, right now from 0 0 0 0, this has moved to a new point, which is given by $lambda_1$ is equal to 4 by 11, $lambda_2$ is equal to 7 by 11, which, the present point that we are looking at is 4 by 11 into (0, 0, 0, 0) plus 7 by 11 into $lambda_2$, which is (2, 3, 4, 2). This is all 0, so the point is 14 by 11, 21 by 11, 28 by 11 and 14 by 11. This is the point we are right now looking at. What is the value of the objective of a function? $6X_1$, 14 into 6 is 84, $5X_2$, 155 plus 84 is 189, 28 into 3 is another 84, so, 189 plus 4, is 193, 273 plus 56 is 329. So we get 329 by 11, which is the value of the objective function, right now.

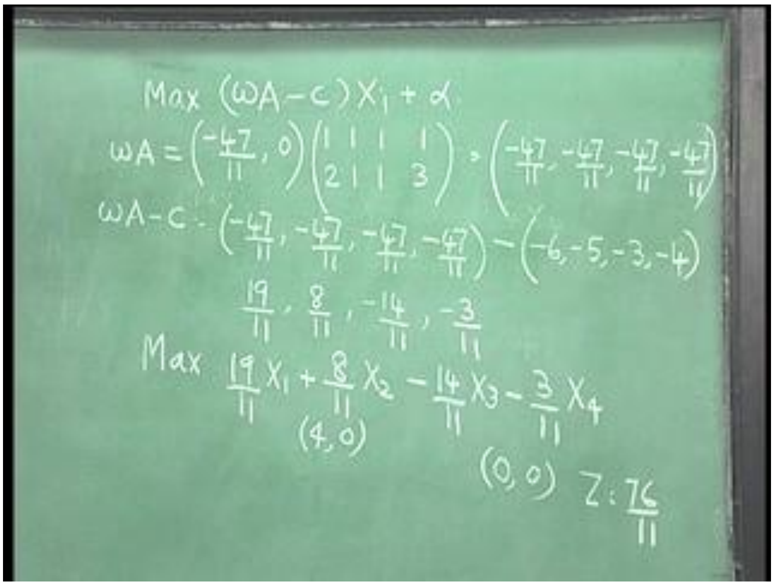
From here we have moved on to one iteration and which means from the corner point (0, 0, 0, 0), we have now moved to a corner point 14 by 11, 21 by 11, 28 by 11, 14 by 11, if we look at the entire problem. But we have not look at the entire problem at a time, we have arrived this corner point in a slightly different way, by getting this as a convex combination of two corner points (0, 0, 0, 0) and (2, 3, 4, 2), with weights 4 by 11 and 7 by 11.

(Refer Slide Time: 47:25)



When we reached here, we have actually generated an entering column by solving a sub problem. The sub problem turned out to be solving two independent linear programming problems in this case. We will not store all the corner points now. We need to check whether this corner point is optimum, in order to do that, we need to check whether any other lambda can enter the basis. For any lambda to enter the basis, we again need to find out maximize wA minus cX_j plus alpha. Subject to these two sets of corner points or subject to the X_j satisfying this and this, both means the same. We now go back and try to solve these two sub problems. From this, we know that w is minus 47 by 11, 0. So, w is minus 47 by 11.

(Refer Slide Time: 48:24)

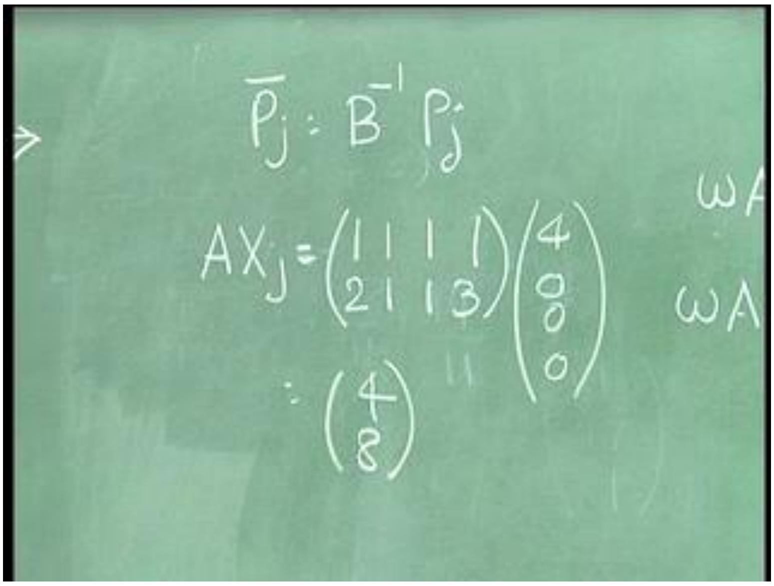


Then minus 47 by 11, 0 into A is 1 1 1 1, 2 1 1 3, so we are first finding out wA, this is (minus 47 by 11, minus 47 by 11, minus 47 by 11, minus 47 by 11), this is wA because 0 into 2 1 1 3 is 0. Now, wA minus C equal to minus 47 by 11, minus 47 by 11, minus 47 by 11, 47 by 11, minus c is (minus 6, minus 5, minus 3, minus 4) plus alpha and alpha is right now at 0.

We go back and simplify this. So, this is 6 minus 47 by 11, this is 19 by 11, 66 minus 47 is 19. Now this is 5 minus 47 by 11, which is 8 by 11. This is 3 minus 47 by 11, which is minus 14 by 11 and 4 minus 47 by 11 is minus 3 by 11. We now have to solve a problem which maximizes 19 by 11 X_1 plus 8 by 11 X_2 minus 14 by 11 X_3 minus 3 by 11 X_4, subject to these two constraints. The sub problem that we generate now is decomposed into two linear programming problems, which again have to be solved separately by using the simplex algorithm. But as explained earlier, since these are all small sized problems, we have stored the corner points and by substituting the corner points into this, we can find out the objective function value and the optimum solution.

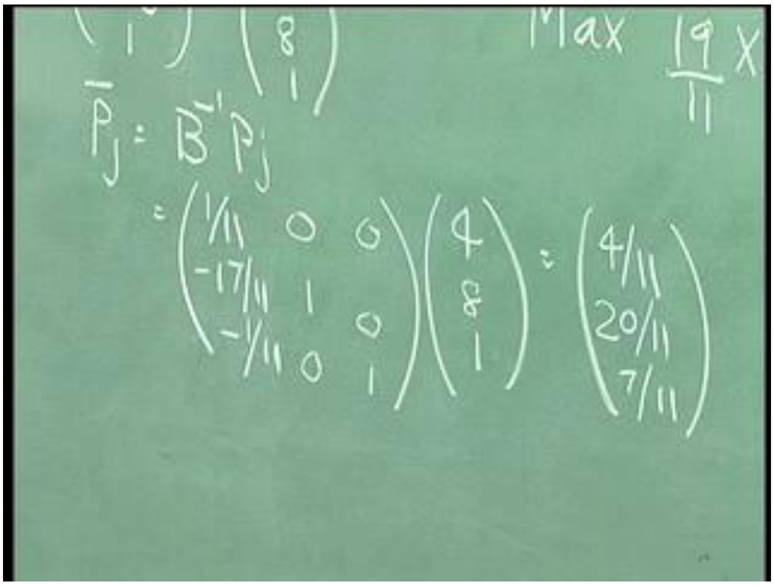
When we look at $19X_1$ plus $8X_2$ and minus $14X_3$ and minus 3 $11X_4$, it is fairly obvious that (0, 0) is an optimum solution to this. You are maximizing minus 14 by 11 X_3 minus 3 by 11 X_4, which is minimizing that, therefore (0 0) is optimum here. For this is concerned, we look at $19X_1$ plus $8X_2$ to begin with. This would give us 0, this is 19 into 4 which is 76, this is 8 into 5 is 40, this is 19 into 2 is 38. So, 38 plus 24 is 62. The point (4, 0) is optimal here. This has 76 by 11, this is 0, so with Z equal to 76 by 11 we had. So the point 4 0 0 0 enters the basis with Z equal to 76 by 11. What we do now here, is we enter this point 4 0 0 0 into the basis, but before we enter that, we have to find out $Pbar_j$ and then we need to enter.

(Refer Slide Time: 52:51)



Pbar$_j$ equal to B inverse P_j and in order to find out P_j we need to do AX_j 1, so P_j will be, so first we do AX_j, so AX_j is 1 1 1 1, 2 1 1 3 into 4 0 0 0. 1 into 4, plus 1 into 0, plus 1 into 0, plus 1 into 0, is 4. 2 into 4 is 8, so we get 4 8. Now, AX_j 1 which is P_j is 4 8 1. Now Pbar$_j$ equal to B inverse P_j.

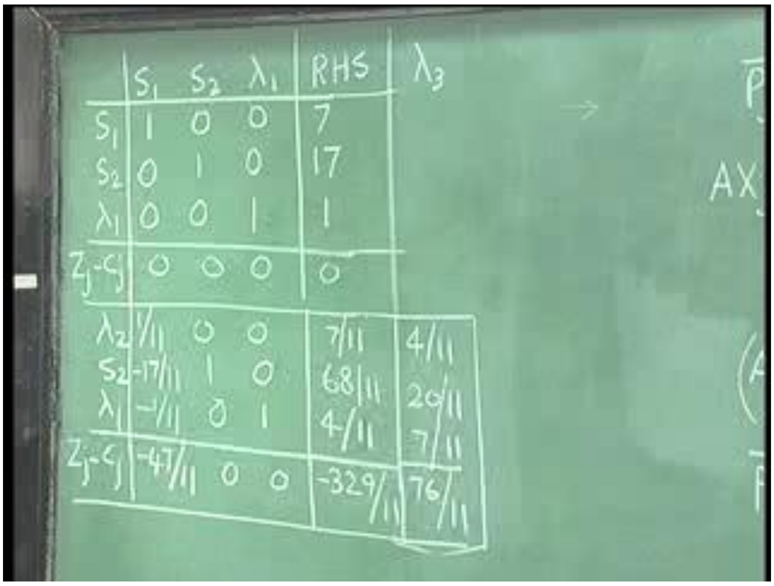
(Refer Slide Time: 53:53)



Pbar$_j$ is equal to B inverse P_j. This is B inverse, can always be read from this, 1 by 11 minus 17 by 11 minus 1 by 11, 0 1 0, 0 0 1 into 4 8 1. This would give us 4 by 11, minus 68 by 11 plus 8 which is 20 by 11, 88 minus 68 is 20 by 11. This is minus 4 by 11 plus 4, which is 40 by 11.

So, minus 4 by 11 plus 0 plus 1 equal to plus 7 by 11. You get plus 7 by 11. Now, the corner points 4 0 0 0, which will take value lambda$_3$, will enter the basis with the entering column 4 by 11, 20 by 11, 7 by 11 and with Z_j minus C_j value of 76 by 11. Let us enter that, so what we do now is the kind of keep this column as some kind of dynamic column, because what we got from here is already available here, I will erase this as well and I will notionally say ,that another lambda$_3$ is entering the basis here.

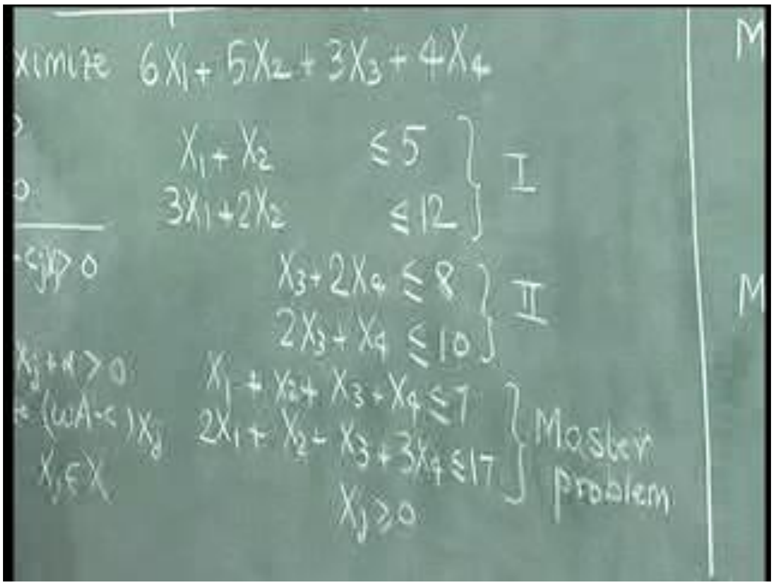
(Refer Slide Time: 55:44)



Lambda$_3$ is entering the basis, with values 4 by 11, 20 by 11, 7 by 11 and with a positive Z_j minus C_j value of 76 by 11. Now, we know that variable lambda$_3$ are corner point 4 0 0 0, enters the basis using a variable lambda$_3$. We now need to the find out the leaving variable, where the leaving variable could any of these three, which is found out by the minimum theta role. Minimum theta is right hand side divided by the entering column, so 7 by 11 divided by 4 by 11 is 7 by 4. 68 by 11 divided by 20 by 11 is 68 by 20, which is 34 by 10, which is 17 by 5 and this is 4 by 7.

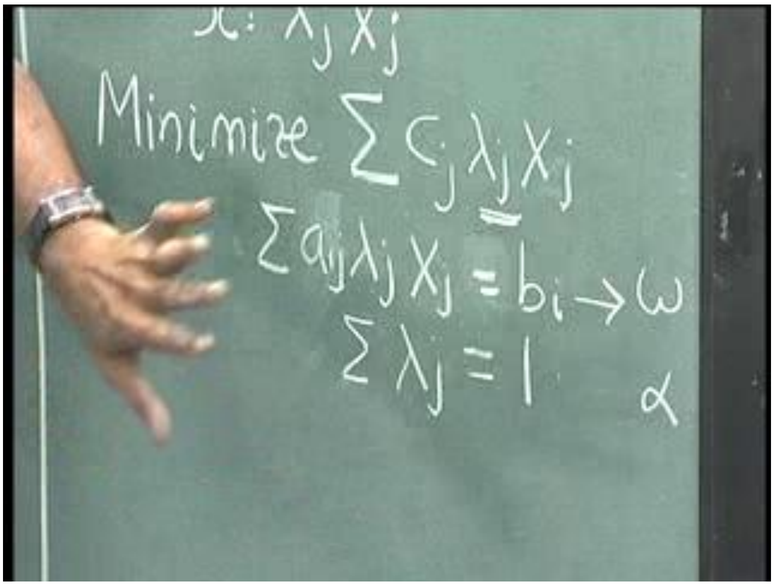
Now, clearly 4 by 7 is the minimum of the three, because this is more than 1, this is more than 3 and this is less than 1 so this variable leaves the basis. When this variable leaves the basis, we need to perform one, more simplex iteration, where we replace lambda$_1$ with lambda$_3$ and continue with the simplex algorithm. But before we continue with the simplex algorithm let us again take a very quick recap of what actually has happened.

(Refer Slide Time: 57:41)



What we did was, in this we started with this large linear programming problem and we realised that by leaving out a certain sets of constraints, the problem is decomposable into smaller size problems. Those constraints that go out constitute what is called the master problem and the other constitutes the sub problem. We also exploited the convex combination property and then we re-wrote the formulation.

(Refer Slide Time: 58:10 min)



Now based on this formulation, we had been carrying out the simplex algorithm and we have come up to the variable lambda$_3$. We will continue the discussion in the next session.