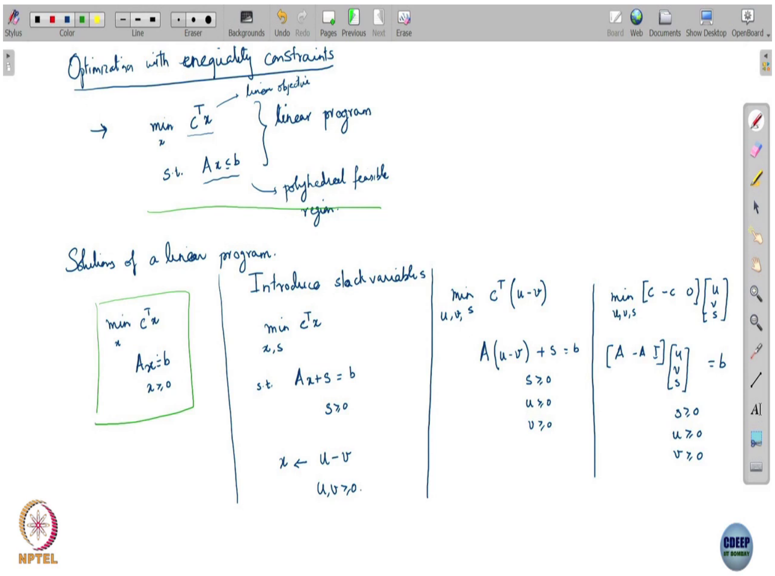
Optimization from Fundamentals
Prof. Ankur Kulkarni
Department of Systems and Control Engineering
Indian Institute of Technology, Bombay

Lecture - 9C
Linear Programming Problems

(Refer Slide Time: 00:19)



So, we started this we went on this a journey because we were talking of Optimization with inequality constraints and the first kind of problem in that the first kind of. So, simplest sort of you can say problem of optimization with inequality constraints is when you are minimizing a linear function over linear inequality constraint.

So, minimizing C transpose x where the x is your variables subject to the constraint that A x is less than equal to b ok. So, what are you doing? Here you have your objective now is a linear function. The constraints are what are the constraints? The constraints are linear inequalities, what do its common region follow of that lies in all of these inequalities linear. So, it is a common region of all of these half spaces.

So, this the feasible region is a polyhedron right. So, we are optimizing a linear function over a polyhedron the and. So, that is what is called a linear optimization or a linear program; a linear program is. So, you have a linear objective the polyhedral feasible region ok. So, this sort of problem is what is called a linear program.

So, let us look at what we let us try to see what we can say about solutions of a linear program ok. So, firstly, the way this problem has been posed you are we are optimizing a linear function over a polyhedral region ok; polyhedral feasible region. Can you put can you show me how this problem can be transformed into a problem that looks like this?

So, how do I transform it from this to this, can does one everyone understand what it means to transform? So, we would like to; we would like to take this a problem like this and do some transformation that we were that we had discussed earlier. Introduce variables change type of constraints etcetera do whatever transformation you can to bring it to this sort of form.

Now, when I say this form I do not necessarily mean that I am only concerned about the form of the problem. Means this A here does not does not have to be the same A here similarly this b and this b does not have to be the same this c this c does not have to be the same. In fact, this x and this x also does not have to be the same right, but I should be able to recover the solution of one from the other ok.

So, can you tell me how do I do this transformation? Ok. So, the first thing is we if we have seen was the introduction of slack variables that lets you convert inequality constraints into equality constraints right. So, the first let us try to do this. So, first so introduce a slack variable s and that then gives you this equivalent optimization you are minimizing C transpose x over the constraint.

Now, A x plus S equals b s is greater than equal to 0 and now your object your this is the variables over which you are optimizing is both x and S ok. So, now is this problem question now is, is this problem in the form of this problem? What is the difference in the form of these two problems? Right. So, the key difference is that here only the if here remember, what is the role of x in this on the problem on the left hand side?

The role of x here is that of the decision variable.

So, it is representing all the decision variables together all of them are required to be greater than equal to 0 in this problem; whereas, if you look at it on the if you look at the problem on the right hand side the decision variables now comprise of both x and s with these together, but the problem only has some of them greater than equal to 0, which is the s component of it is greater than equal to 0.

The rest are not they are not imposing greater than equal to 0 there right. So, this is not in the same form as this right the problem on the left hand side has the form, where all decision variables are greater than equal to 0 the right hand side has only some of them. So, we have to somehow do more work to get it to this form.

Remember focus is on form; of course, x itself make a you cannot ask for x certainly to be greater than equal to 0 because it may not be valid in this particular problem. So, you have to transform x into some other using some other variables in such a way that you get it to this sort of those variables end up having this sort of form.

So, what would be the way to do that? So, one simple way of observing this is notice that x is just can is a real vector it can take any both it can have positive as well as negative component, take any component of x, then that is just a real number it can be positive or negative or 0, but every real number can be written as a difference of two positive numbers right always

So, component wise I can always write x as the difference of two non negative vectors right. So, I can do this change of variables I can replace x; I can replace x by u minus v where u and v are both non negative. Since u and v are non negative the, but I am looking at the difference, the difference can take both positive values negative values can take 0 also, every possible value of x can be generated as a difference u minus v where u and v are non negative.

So, if I make this transformation this will not change the this is this transformation is going to be equivalent.

So, I can make this change minimizing c transpose x now becomes minimizing my x gets replaced now by u minus v; u minus v. A as is going to be multiplied by u minus v plus s equals b, but I am going to keep u and u and v non negative sorry. So, my I will have inequalities s greater than equal to 0, u greater than equal to 0, v greater than equal to 0.

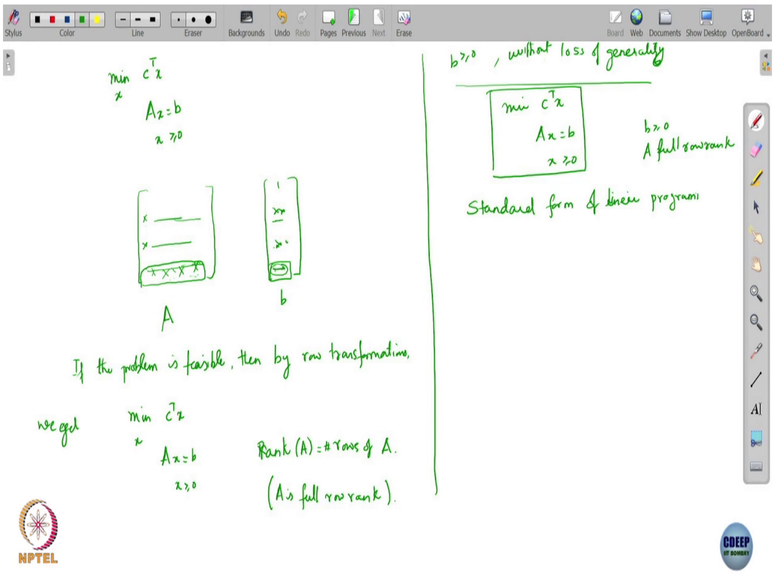
Now, what are my variables now the decision variables of my optimization? s which is earlier remains and in addition to that I have u and v also. So, my I am now optimizing over u, v and s right. So, I can just write this whole thing in a neater form, we write it like this minimizing over u, v, s this sort of function c minus c 0 u v s.

Now, question is, is this in the form of this problem on the problem that I was looking for? Yes, it is in this form because all my variables are now non negative I have equality constraints or the other constraints are all equality constraints and I have a linear objective right. But you see in the process of getting to this form my x is completely disappeared right.

So, I when you are changing the form of the problem it should be open to changing redefining your variables redefining your constraints introducing new variables etcetera right so, but focus is to focus should be on the form not on the on retaining the identity of particular variables ok.

So, what it means is that what the conclusion from this is that if I give you any linear program like this which is optimizing a linear function over a polyhedron, after some transformations I can bring it down to this sort of form. Now, can you tell me if you can make this even more specific?

(Refer Slide Time: 11:58)



So, I have brought the a linear optimization problem to this form, can you make it even more specific say more about remember the transformations I was we had discussed. We had discussed things like eliminating constraints for example. So, are there some constraints that you can trivially eliminate from here?

So, look at we have equality linear equality constraints here. So, if there are rows of A that are linear combinations of row of some other rows of A right, then it has to be the then you can eliminate those. So, if there is a row of a here. So, look at the matrix A, there is some row here that is happens to be a linear combination of these two other rows and you have b on the right hand side here the corresponding b here is your b this is b this is A.

Now, when I do row transformations I can using row transformations I can basically eliminate make these all of these 0. When I am doing row transformations, what am I doing? I am multiplying this row by some; multiplying this the rows by some constants, adding them to other rows and subtracting them from other rows etcetera right.

So, by doing that I end up eliminating this make this entire row 0 so, it is as good as not present. What would what should happen to this after those transformations? If these equations are consistent this should also become 0 right; otherwise these equations are not consistent and in which case there is the feasible region itself is empty right.

So, if this with this and this with this and the original form of these three equation if they are consistent, then after these row transformations not only should this disappear, this should also become 0 right. So, in short it that entire constraint should become irrelevant one right.

So, if the feasible region is non empty, then without loss of generality, I can do these transformations and get rid of all these linearly dependent additional constraint. Now, is this changing the geometry of the problem? Is this changing the feasible region? It is changing the number of constraints, but its not changing the feasible region right.

Because the feasible region is the set of xs that satisfy these constraint and the same xs that satisfy their your earlier set of constraint continue to satisfy the reduced set of constraints also because you have only removed the linearly some linearly dependent constraints, correct? So, the feasible region does not change, but constraints get removed.

Student: (Refer Time: 15:05).

No x the dimension of x remains the same it is only the number of constraints which is changing right. So, remember feasible region is a region in space constraints are just an algebraic way of representing them. There are many redundant ways by which you can represent the same region and we are removing all these redundancy by getting rid of these dependent equations right reading ok.

So, if I if the problem is feasible, then I can do the above transformation, then by row transformations we can reduce the problem to this form; we get something like this you are minimizing a linear objective c transpose x again over the fall of the constraints is still the same A x is equal to b x is greater than equal to 0, form of the constraints is still the same, but what can we say?

I can now say that if I look at the rank of A that is then that has to be equal to the number of rows of A. So, A is full row rank ok. So, by doing, so this is another further reduction. So, I had an general problem like this where I had an equality constraint A x equal to b and x is greater than equal to 0. I have now said well that also I can be more specific, I can take a to be full row rank.

Can you be even more specific then; is there any further reduction possible? This A is not no more the same A that you have start off started with. The A we started with is this the A we ended up with is this sort of A and this may have dependent linearly dependent rows.

So, question is that I asked was can is it possible to make this problem even more specific, can you say something more? Sorry no x is greater than equal to 0 that is the that is just a constraint

Student: (Refer Time: 17:50).

Yeah, so we can make them more specific make it more specific by to the point where we now begin to solve the problem ok. So, the but that is not what we are trying to do we still want to read we do not want to dive deep and begin you know solving the problem, but we want to still looking at the form itself try to be more specific about it.

So, instead of an arbitrary A now we have an A which is full row rank, what about b? In the in all these transformations b has actually left been left untouched. So, b can actually be taken as greater than equal to 0 and this is without loss of generality, can you tell me, why? So, without loss of generality b can be taken as greater than equal to 0 no.

So, if I try to do what you are saying which is that I write b as b 1 minus b 2 take the one of the as difference of two non negative bs you will take one of them to the left hand side, but it is not being multiplied by x yeah. So, the point here is it is the very simple observation actually that, here you have equality constraint. So, they remain valid even if I multiply any one of the rows by minus 1.

So, if I have a b which is negative I can just multiply that particular row throughout by minus 1. The corresponding row of A will get multiplied by minus 1 the corresponding b gets multiplied by minus 1 and I get back on positive A positive b right. So, by doing this I can get without.

So, without loss of generality I can assume b to be greater than equal to 0 right. So, what this means is so this gives us the final form that you can do minimize c transpose x, A x equal to b, x greater than equal to 0, b greater than equal to 0, A full row rank. This is what is called the standard form of a linear program.

Now, why care of why do all these? The reason is because once you have a standard form standardization like this helps in creation of technology on top of it right. Once you have a standard form you can people can create methods for solving that particular problem of that particular form.

If you have too many different forms then the there is no fix form you know do not know which one to choose and so on right. So, all you will see that many solvers usually assumed that you are starting of your linear program in this in the standard form. So, they demand that you enter your problem in a standard form right ok.

Of course, there you can anything that is written in this sort of form is at the end of the day a special case of the first definition that I mentioned. It is a eventually a linear optimization of a linear objective over a polyhedral feasible region. So, if the solver demands that you write it in this form you can also write it in this form clearly right ok. So, we will end here and we will continue next hour.