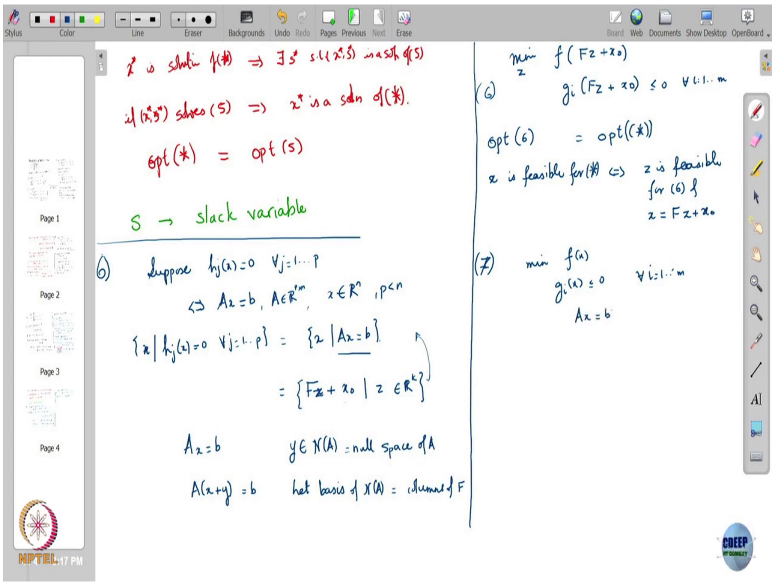
Optimization from Fundamentals
Prof. Ankur Kulkarni
Department of Systems and Control Engineering
Indian Institute of Technology, Bombay

Lecture – 7C
Transformation of optimization problems - III

(Refer Slide Time: 00:16)



Let me show you another example of by another example of a problem. Suppose, so let us look at problem number 6. So, suppose you have ok; suppose you have the let us consider the case where, suppose my equality constraints these equality constraints right these are actually linear suppose ok.

So, this is actually the same as h j of x equal to 0 for all j equal to 1 to p is equivalent to doing say A x equals b. Suppose I can, so what this means is so if I more precisely, if I look at all the x's for which h j of x equals 0 for all j running from 1 to p. Then that is actually the same as the solutions of this linear equation system of equations A x is equal to b ok.

And now of course, this will be this sort of thing which has an interesting solution only when p is less than the number of variables which is n. So, if you have if my x is in R n here and my; so then this, so my number of rows here. So, A has to be in p cross n sorry, number of rows is p ok. So, if you have this sort of a problem where you have equality constraints like this.

Now, when you have equality constraints so are that is basically another in another language the system of linear equations. We can always, we can always you know find their solution set ok. So, the solution set of a problem like this would look like this, it will look like some F x sorry, it will look like some F z plus x 0. Where, z ranges over say some R k.

How do I get it to this sort of form? So, we are, remember we are talking of p less than n right. You would get you get this in this sort of form by you need a particular solution which is x 0 and then you need a. So, any solution x that of A x equal to b whenever you have a solution like that you can always add to it. And take up take some vector y is a in the null space of A, take the vector y which is in the null space of A then that x plus y is also a solution of this system of equations.

Now, the null space of A has it is own basis right the null space of A has it is own basis. So, let the basis of the null space, let the basis of the null space be the columns of the columns of f right. So, if I take the basis of the null space as the columns of f or what I need take any solution x 0, a particular solution x 0 and then add to that the entire null space of this of a that would become my that then gives me the say that is the same as the solution set of my original set yeah.

No, I am saying suppose that is the case. So, I it is not a it is not equivalent suppose it is yeah. So, we are considering only that case where this is where we think where you have linear equality constraints ok. So, in that case then now what happens is because you are able to do this your entire problem can now be transformed.

So, you can now do you can get rid of your equality constraints altogether and do the substitution. What is my variable now? Now, my variable is z right. So, I can so and z is just anything in R k and there is no constraint on there is no sign constraint or anything like that on z it is the dimension of my null space you can say ok. So, this problem as now what have we done as a result every x that satisfies A x is equal to b has been written as a equivalently as an F z plus x 0.

And I have substituted for all such x's as f z plus x 0 and put them back in. Once I do that substitution this is if this is equivalent right to the original. So, all the x is that satisfy A x is equal to b are can be written in the form F z plus x 0 likewise every x z everything that is in the form f z plus x 0 satisfies A x equal to b. So, as a result my equality constraints can be removed altogether ok.

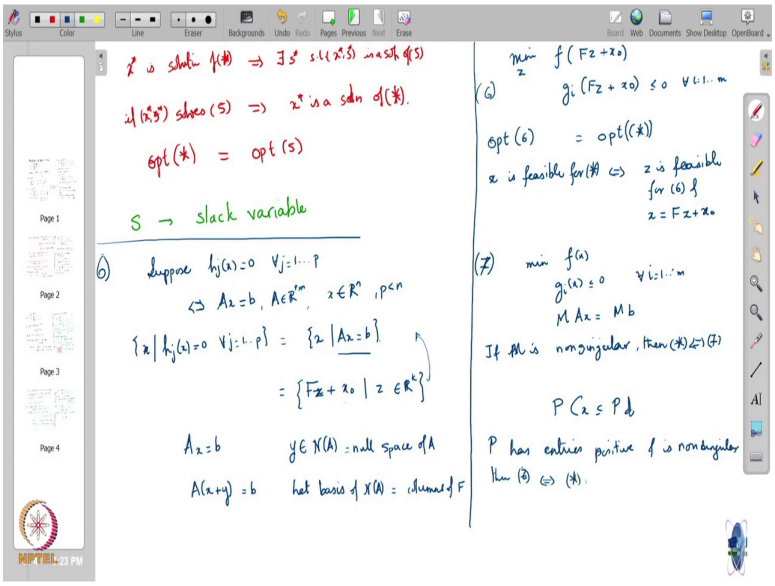
So, if you this kind of way if you have a way by which you can solve for one set of variables using the equations or the constraints that you have you are welcome to do so it is that gives you an equivalent optimization problem. So, this problem here this problem let us call this problem 6; 6 is equivalent to problem star, in what sense? What about the optimal values they would be equal optimal value of star is equal to the optimal value of 6.

How do I relate the feasible regions? So, 6 is written in the z space which is R k star was written in R n right, but still they can be related they can be related because of this relation. So, through this relation you can so for every x that is feasible for star there is a z that is feasible for 6 and z is equal to and x is equal to f z plus plus x 0 for every, so x is feasible for.

Ah sorry it is for every x that is feasible for star into yeah; x 0 is the well it is what is called a particular solution a particular solution of this linear equation. Let me show you one case where again cautionary note while you are eliminating equations and so on. So, this so, the one case where you should be careful is this sort of case.

Suppose, I have let me suppose I have this kind of optimization problem. So, this is not exactly elimination, but it is rather some other form of transformation you can say. Suppose, I have again linear equality constraints for simplicity; linear equality constraints some A x equal to b. This is my this is one problem , and now what I do is; I let me suppose I multiply these constraints the equality constraints by a matrix M ok. So, I let me left multiply it by M.

(Refer Slide Time: 09:53)



If I am left multiplying it by M what can I say about this problem 7 compared to the original problem? Yeah, so if M is; if M is non singular then of course, you can just multiply again back by M inverse and you will get back the original problem. Now, the reason I brought this up is so is because there are in many cases M is not singular.

If M is not singular like for example, M is suppose just a vector ok, just a row vector a row vector that you are multiplying on the left by and like on the left hand side and likewise on the right hand side. What you are effectively doing is; doing a weighted sum of all your constraints.

One of the things that students tend to do many often when they are trying to solve optimization problems they see these equality constraints as equations and then they go about so why do not I add these equations let us see what happens this cancels that cancels etcetera etcetera. And in the process what they are doing is applying linear transformations ok.

Now, they may get they may get a solution of these; of these equations you really need to carefully ask have you captured all the solutions. You may get one solution by while you do this, but you may not get all.

Because generally you have more variables than equations and what you will miss out as many is this whole range of solutions in the null space all right. So, if M is not singular then this kind of elimination whatever you do you know by adding; adding equations subtracting equations etcetera that is all those transformations are effectively amount to multiplying by some M right.

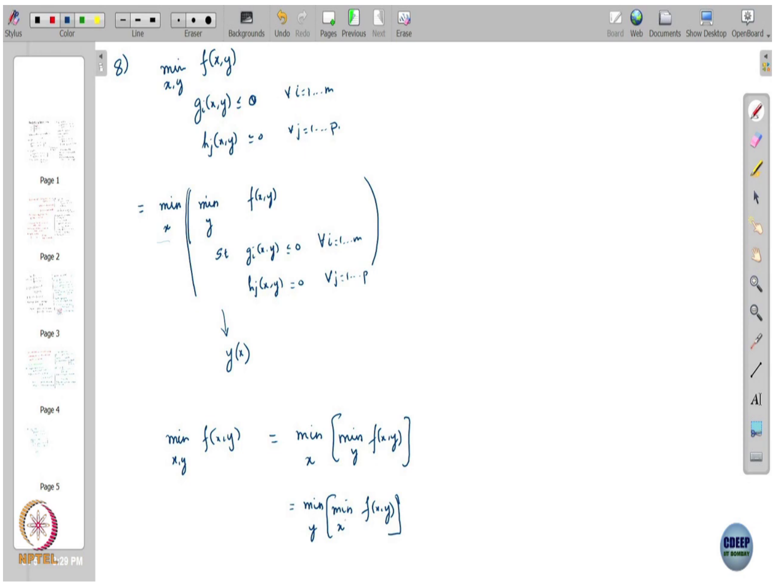
So, be careful whether your transformations are actually giving you a all the solutions or not ok. So, that was that is the reason for multiple explaining this. Does the sign of M matter in all this? Can you subtract equations can you add equations freely? Sign is immaterial, yeah because you can always add equations subtract equation no problem what matters is non singularity right.

Now, what if I had to do the same thing with my inequality constraints? If I suppose I had another these inequality constraints let us suppose we write them for simplicity as sum C x less than equal to d. I had inequality constraints like this and if I multiplied both sides by another matrix say, multiplied both sides by another matrix suppose P, is that valid? What do I need on P? So, for me to for this to be valid means that I firstly inequality has to be preserved when I multiply ok.

So, P so it has to be the k, what you are what are you doing effectively you are multiplying each of the inequalities by a scalar and each of them needs to be preserved otherwise this whole operation is not going to be at preserve right. So, you need that P is all has all entries non negative actually as all entries positive. And of course, you need then non singular so that this is this is equivalent to the original ok.

So, with when you have inequality constraints as other danger of flipping the direction of the inequality and changing the entire region over which you are optimizing. One last point let me make about again this is not exactly related to transformation, but this is; this is again at an important trick.

(Refer Slide Time: 15:13)



Suppose, you have an optimization problem like this which is over two variables some variables x and y and you have constraints also over x and y. So, you are minimizing over both x and y so this is less than equal to 0. The question here is is this the same as optimizing first over x first over y and then over x. So, fixing an x I optimize this over y or vice versa fixing optimized first over x and then over y.

Question is does the order matter or when we are faced with an optimization problem we can freely we can optimize in any order we like. No, you are eventually minimizing over both question is, can I do piecewise when I do minimize over one first keeping one fixed then minimize the resulting thing over the other variable. May not be; but I am after that I am revising over the first one also. See, I am minimizing over x after I have minimized over y.

No, so it is not it is incorrect to say that y will be fixed. So, let us understand what we are doing here. So, when we are minimizing over x and y the objective is to goal is to pick jointly a pair of x and y ok. So, you have to pick that pair combination correctly. So, when I am doing saying first minimize over y then over x, what I am doing is fixing a value of x finding a y. So, my necessarily my y is going to be a function of x ok.

So, it is not so this is something to be remembered when you do it analytically of course, you will get y as a function of x, but when you are doing numerical computations this will this is now often not evident, because you will get back some number you will just input that number and forget that it is dependent on the x anymore right ok.

So, y will is a function of x, I put that y back as a function of x, then I will get some complicated function of x in which y has been eliminated and then I will I optimize that over x. This is all this so this can always be done does not matter what the order is, but there are some practical issues though. Remember whether you can whether a pair of x comma y is feasible or not is not evident until you test the pair itself.

See, that pair whether it is satisfying all the constraints or not is not evident until you test the pair. So, it may not be that for every x you can get a feasible y so you fix an x outside you search you may you satisfied the first constraint second constraint third constraint you do not satisfy. In what does it mean? You cannot satisfy that constraint there is no y that satisfies first second and third together which means that the x itself needed a change.

So, you need to start with a different x and then you search again over y or maybe then again start trying different x etcetera. So, in the presence of constraints this whole procedure gets messy. If you do not have constraints in the in the unconstrained case if you just simply had this sort of problem this is absolutely nope no issue you can always do this. Minimize first over y then over x or minimize first over x then over y ok.

As, I said in the presence of constraints also it is valid, but it is messy this is sometimes the easier thing to do. When you get when you notice that your objective has some structure that it is simpler in one set of variables than in the other target those variables first. Optimize over those get those in terms of the others substitute back solve alright, fine.

So, this is also another case of you can say roughly speaking transforming optimization problems, you are getting rid of some variables the optimizing them out alright. So, we will stop here.