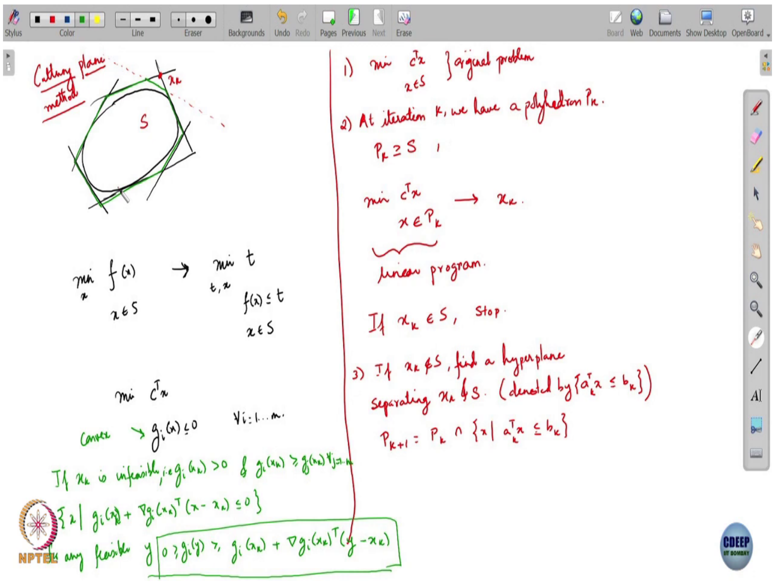
Optimization from Fundamentals
Prof. Ankur Kulkarni
Department of Systems and Control Engineering
Indian Institute of Technology, Bombay

Lecture - 21C
Cutting Plane Methods

(Refer Slide Time: 00:26)



What is the, what is the idea of a cutting plane method? The idea of the cutting plane method is suppose you have an optimization problem in which you have a feasible region that looks like this sort of a convex set. What it would do is, it would try to approximate this convex feasible region using hyperplanes. It would try to approximate this convex feasible region using a hyperplanes.

And at each iteration try to improve the description or the description of these hyperplanes; in other words, it will keep adding hyperplanes to this feasible region to the point where the approximation starts getting to the point where it starts looking almost like the like an like the original constraint.

Now, the, beauty is because you are adding hyperplanes at each step what you would what you what your minimizing over at each step is not the original convex problem a convex region, but rather just a polyhedron. Now, if, so this sort of a problem is ok is potentially much easier because you would you can use techniques that for that are used that are available for linear programming in order to in order to address this particular problem.

Now, the now you would; obviously, ask how would you use linear programming here? Because your objective is not necessarily linear, but then there is a very simple trick to actually convert any optimization problem to a problem where the objective is linear alright.

So, let me first tell you this. So, suppose if I have an objective an optimization problem like this where you are minimizing f x subject to x in s, is there a way by which I can convert this problem to a problem where the variable now is x. Is there a way by which I can convert this problem to a problem where the objective is linear? So, the answer to this is yes. You can do this. You can introduce a new variable t and do a minimization over x as well as t, and add an additional constraint, which says simply that f of x is less than equal to t in addition to x lying in s.

Now, you any of you can check that this problem is these problems are actually equivalent minimizing this over t and x is the same as minimizing just simply f x over x right. So, what has happened as a result? As a result your objective function which could have been any non-linear objective function here has now become the objective has now become linear.

Which is, so this is one of the reasons why it is actually somehow a you know in a in optimization, people tend to think that the difficulty in optimization is always to do with the objective, whereas, in reality the difficulty is all in the constraints you know any kind of complication in the objective you can always be pushed into the constraints.

So, the geometry of the constraints is what makes the problem hard not so much the geometry of the objective. So, without loss of generality, any objective, any optimization problem can be written as an optimization of a linear function over some constraints and that is what we have done here.

So, as a result of this, what we can do is we can start off assuming that this here is the form in which we have been given a problem. We have been given a problem in which you are minimizing some linear function. So, we let us say you are minimizing some linear function c transpose x subject to convex constraints g i of x less than equal to 0. There are say m convex constraints i going from 1 to m alright.

Now, the cutting plane method which is what I am talking about here. So, let us call it write that here cutting plane method. So, what a cutting plan method would do is this. So, in an abstract form what the general form is so, would given a polyhedron you at each iteration k you have a polyhedron p k.

So, your, so it, so we are going to assume that the object assume that you are minimizing c transpose x subject to x in s; and k at each iteration k, this is your original problem at each iteration at each iteration k, you have a polyhedron P k that outer approximates s. We have a polyhedron P k that outer approximates s alright.

Now, you what you do is instead of solving the original problem, you solve you do this you minimize c transpose x subject to x in now in p k. Now, this here is a linear program. This is now a linear program alright. So, if you if x k belongs to s that means, if it is feasible for the original problem, you can you one can stop because you have now found a solution of a lesser constraint problem that like that is feasible also for your original problem. So, you can now can stop, and declare this as the solution alright.

But at the, but if x k is not feasible for the original problem, then what do we do? Well, if x k is not feasible for the original problem, then you are in the situation where you have found say a point like this. Say suppose this is your original feasible region here and you have constructed an approximation of it using these using hyperplanes and got to p k, and you have now found a an optimal solution say at this corner point here. And this is suppose your x k at this state.

Now, what this x k is not in the feasible region s, but the feasible region s is convex and x k is out sided, then that what that tells us is that there must therefore, exist a another hyperplane like this. There must exist a separating hyperplane, a hyperplane that separates x k from s right ok. And what one can do then is that will add this hyperplane to your definition of P k ok.

So, what, so effectively what and that would then tighten the original hyperplane P k and generate for you a new sorry the original polyhedron P k, and generate for you a new polyhedron P k plus 1 right. And then you are minimize then c c transpose x over P k plus 1, and then go on right.

So, if x k is not in s, what one does is find a separate find a hyperplane separating x k and s ok, and then define P k say denoted by denoted by say denoted by say a transpose x less than equal a k transpose x less than equal to b k ok. So, suppose, this is your hyperplane, so then you just simply define P k plus 1 as P k intersection x as that a transpose x a k transpose x k is then equal to b k.

So, what is happened is the origin your original hyperplane, the original set was this P k as that has been outlined now, the new set the new constraints would be this, this, this, this and so on. And with each iteration your outer polyhedron will be will continue to shrink and it will continue to better approximate the feasible region.

And so their advantage of this is that at every step you are only solving a linear program. And if solving that linear program is cheap and is something that you can do easily, you can effectively solve a convex optimization problem using linear programming alright.

So, now how do why do we need this to be convex? Because we need the guarantee that that are separating hyperplane exists, so that something that we have only in the case of a convex optimization problem. We the other reason why this convex optimization works very neatly with this is because it all the generation of these new hyperplanes ok defining these new hyperplanes becomes very easy when the problem is convex.

So, for example, in this particular problem that I wrote out here where you have where your constraints are g i of x less than equal to 0 for i from 1 to m. So, if your point is infeasible, so if your x k, if x k is greater than 0, sorry if x k is infeasible which is effectively saying that g i of x k is greater than 0 ok, then what one does is what one can do is simply notice the following.

So, a notice that. So, these are I assume that these are all convex ok. So, if x k is infeasible that is g i of x case greater than 0, and for let us suppose you choose the i which is most infeasible, that means, g i of x k is also greater than equal to g j of x k for all j for all j going from 1 to m ok alright.

So, if so if this is the most infeasible one out of these right, so then you can then in that case the you the new cutting plane or the new hyperplane separating hyperplane is defined as x such that g i of x k plus gradient of g i of x k transpose x minus x k is less than equal to 0 alright. Now, this is a hyperplane in x and it is defined use in terms of x k. So, this is linear in x, and x k is simply a parameter here.

Now, why is this separating hyperplane? The reason for that is well because if I take any if I take any point if I take any point that is feasible. So, for any feasible point, any feasible y, it must be that g i of y is less than equal to 0. And from convexity, it also has to be that this is greater than equal to g i of, by convexity this has to also be greater than equal to g i of g i of x k plus gradient of g i of x k transpose y minus x k right.

So, what does this mean that any feasible y will always satisfy this particular thing, that means, the in particular this equation here, this here should be less than equal to 0 right. Whereas, on the yeah so, this particular thing would always be less than equal to 0. So, what does this mean? So, this means that all feasible the entire feasible region of your problem ok, the entire feasible reason of your problem must be contained in this particular half space this half space that is written that is defined here right.

So, when, so when you find a point x k you just add this particular inequality constraint to your definition of your polyhedron P k, and that gives you an additional half space in that contains ok this defines for you this half space here that contains the original the original feasible region right.

So, in short by this is what the summary is that if you have a convex optimization problem like this, the this particular this simple tangent condition of a convexity also gives us ways of generating hyperplanes that would have of the kind that we require alright ok.

So, with this, I think, I can I will wind up this lecture. And we will take up interior point methods in the next lecture.