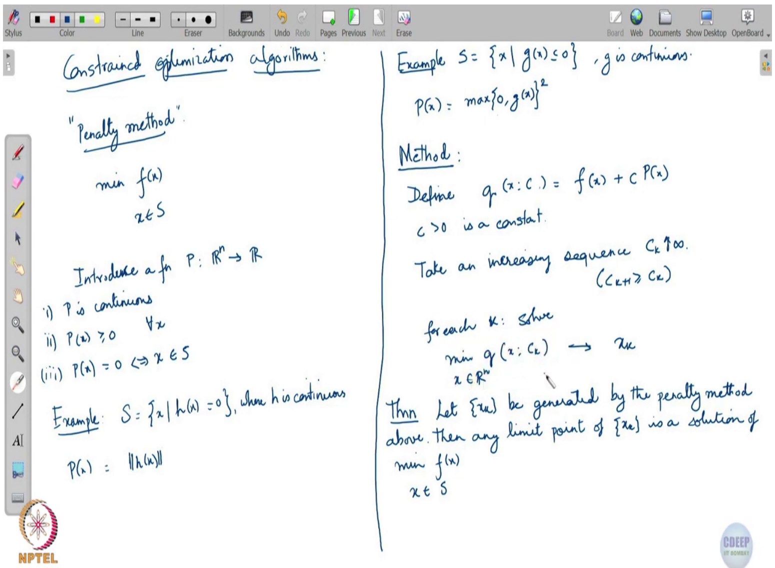
Optimization from Fundamentals
Prof. Ankur Kulkarni
Department of Systems and Control Engineering
Indian Institute of Technology, Bombay

Lecture - 20C
Penalty methods

(Refer Slide Time: 00:21)



So, let us now start on with constrained optimization, algorithms for constrained optimization, constrained optimization algorithms. Now, there are the constrained optimization the methods for constrained optimization are again extremely varied and there are many different types of a types of methods out there. By the simplest, I will, let us start with the simplest method, which basically attempts to convert a constrained optimization problem into an unconstrained one.

And it is a very elegant and nice method. So, let us just see what this idea here is. So, this is what is called a Penalty method. So, the in a constrained optimization problem what we are faced with is a problem like this. You want to minimize a function f, subject over a set S. Now, what a penalty method does is it says, it looks for a, it looks for a function P. So, introduces a function P, introduce a function P such that the following holds.

1st P is continuous. ii P is non-negative for all x. And iii so P is a function now from R n to R. So, it is defined on the entire space. So, it is continuous. It is greater than equal to 0. And it is 0 if and only if, it is it is 0 if an; so, P of x is equal to 0 if and only if x is in S. So, when you are in the set it is in the when you satisfy the constraints or you are in the feasible region, then P is 0 and everywhere else it is positive.

But, unlike the kind of function that we had seen in when we were talking out duality, where we had defined this sort of; if you recall we had defined these i 0 and i plus functions, which went from 0 to infinity outside the feasible regions, these the function P is a continuous function. So, it is not just, it is not that it is not an ill behaved function, ok. So, it is a continuous function, it has it is 0 on the on the feasible region and outside its positive.

But because it is continuous it will bring with it a little bit of baggage, means that it is you cannot it is not going to be an exact, it is not going to give you an exact reformulation of the constrained optimization. You would need to do a little bit more work, in order to solve the constrained optimization problem using the penalty function, ok.

So, first let us look at a few examples of this sort of function. So, let us say suppose my constrained S. So, I write here examples. So, suppose my constraint S is just x or my feasible region S is just x, such that h of x is equal to 0, right. So, it is some one or more equality constraints. So, in that case what is a correct penalty function for this sort of set? So, one can take for instance P of x as simply norm h of x; now, what is; or norm h of x square if necessary.

So, why can we do this? The reason we can do this is that you can see that h of x is equal to 0 if and only if norm x is equal to 0. So, so property 3 holds for free. So, b of x is 0 only on the set S everywhere it is obviously, greater than equal to 0. And if your h is continuous P is continuous, right; if h is continuous P is continuous right. So, as a consequence we get that;

so, let me write this here, so where h is continuous. So, if P h is; so, if h is continuous P is continuous so therefore, this is now a penalty function.

So, how can one do this for an inequality function, inequality constrained? So, another example would be say S, which is x such that g of x is less than equal to 0. What would be; what would be an appropriate penalty function for this? Well, one can do take for example, P as the maximum of 0 comma g of x.

So, what does this do? So, this is a max of f and g again is continuous. Now, what does this P of x do? P of x is the maximum of 0 and g of x. So, when so, it is a maximum of two continuous function; one is just a constant function. See it is a maximum of two continuous function, so it is also a continuous function.

So, it satisfies 1. It is a maximum of 0 and something. So, it is always greater than equal to 0, so it satisfies 2. And when is this equal to 0? Well, this would be equal to 0 only when g x is itself less than equal to 0. So, when this is less than equal to if; so, P of x is equal to 0 if and only if g of x is itself less than equal to 0. So, consequently it satisfies 3 also, right. So, this gives us another a penalty function, another kind of penalty function.

Now, obviously, this is not as nice as this function because it is non-differentiable, but, so you can even do a little you can make this a little smoother. For example, you can also do this the whole squared and that will ensure that the derivative, that that will give you a little bit more differentiability also at the at points where P of x is equal to 0, alright.

So, what is the method now? What is the penalty method? The method itself is this. So, what does one do? You define, so define this function q the function q of x given c k, ok. I will explain what this is. So, this is or let us say x given c. So, this is defined as f of x plus c times P of x. So, you have your original function plus c times the penalty function, ok. Well, what is c? c is any constant, c greater than 0 is a constant. So, for a given c you will define this sort of function.

So, what has this done? It has defined it has given you a new objective, it is given you a new function, where you have the original function plus a constant times penalty. Now, when you are on the in the feasible region the penalty function by property 3, if the penalty function should be 0, so in that case therefore, the q is actually identical to f. So, in the feasible region minimizing q is actually the same as minimizing f.

But then what we will be doing is we are not going to put constraints any more. The whole idea of a penalty method was to convert a constrained problem to an unconstrained problem. So, what the method would do is it would do the following it would. So, it will take an increasing sequence c k going to plus infinity, ok.

So, it has increasing sequence of constant c k going to plus infinity. And by increasing I mean c k plus 1 has to be greater than equal to c k, right. So, this it is an sequence like this, where it goes to plus infinity. And then what does when do at each step? For each k solve minimize q of x given c k, over x and R n. So, what are we doing at each step? At each step you take your constant and minimize q of x given c k.

So, what is going to happen with as your k increases? As your k increases your c k increases and becomes increasingly large. So, what this tends to do is that whenever the c k is becoming extremely large, if your x goes into the region where P is not 0; that means, where P is positive, that is the region outside your feasible region, it tends to blow up the value of c times p.

So, if your x wanders into this region, where P is positive because you are multiplying it outside by a large constant c k, the value gets blown up. So, when you are minimizing q here, when you are minimizing q, the better value for better value for x can be found where; can be found where in the region where P is actually 0; that means, in the region S itself.

So, what it does is as, k becomes larger your, the minimization tends to gravitate more and more towards this minimization, this unconstrained minimization in effect tends to gravitate more and more towards the feasible region S. And it starts producing, looking for iterations in looking for solutions in the region S itself, although you have not actually imposed them as constraints.

Though you have not imposed them as constrained it sort of implicitly looks for those for solutions in that region because outside that region the objective value itself becomes very large. So, it becomes; so, the iteration will tend to look for those for solutions there, alright.

So, the result that we have for this is the let me state it as a theorem. So, let x k; so, this suppose the iterate generated here is x k. So, let x k be generated by the penalty method above, then any limit point of the sequence x k is a solution of the minimization of f over x in S. So, what does this mean? So, you keep doing this for over k and keep increasing it, increasing your c k.

Eventually you get a sequence the sequence will gravitate towards the towards, if it has a limit point it will that limit point will be a solution; will be a solution of this particular problem. Now, the thing that you that is interesting here is that when you are solving; when you are solving this problem for when you are minimizing sorry; when you are solving this particular problem you are minimizing f plus something else over this entire region, right.

So, what you actually end up doing which is somewhat in counter intuitive may seem counter intuitive is that you are approx you end up approximating f from below. So, you end up; so, your objective value actually increases to the optimal solution, rather than decreasing to the optimal solution which is what you would otherwise do in an optimization problem.

And that that happens because of the because you are penalizing the constraints and it is the nature of the penalty function that creates this effect, ok, alright. So, with this actually let me also mention to you that there is an slightly related type of method which is called a barrier method.

A barrier method only what it does is instead of penalizing the entire the entire constrained it tends to penalize the approach towards the boundary. So, as you go towards the boundary the barrier function blows up and so, it tends to keep you within the constrained.

So, in the penalty method you are searching all over the space, in the in the entire space R n without going out of the without imposing the constrained. And then eventually you end up inside the constrained. Whereas, in a barrier method you do not impose the constrained, but you impose, you penalize movement towards the barrier towards the boundary of the constrained. So, that gives you a different type of method, alright.

So, with this I think I will end today lecture. We can then, we will continue on with more optimization algorithm in the next lecture.