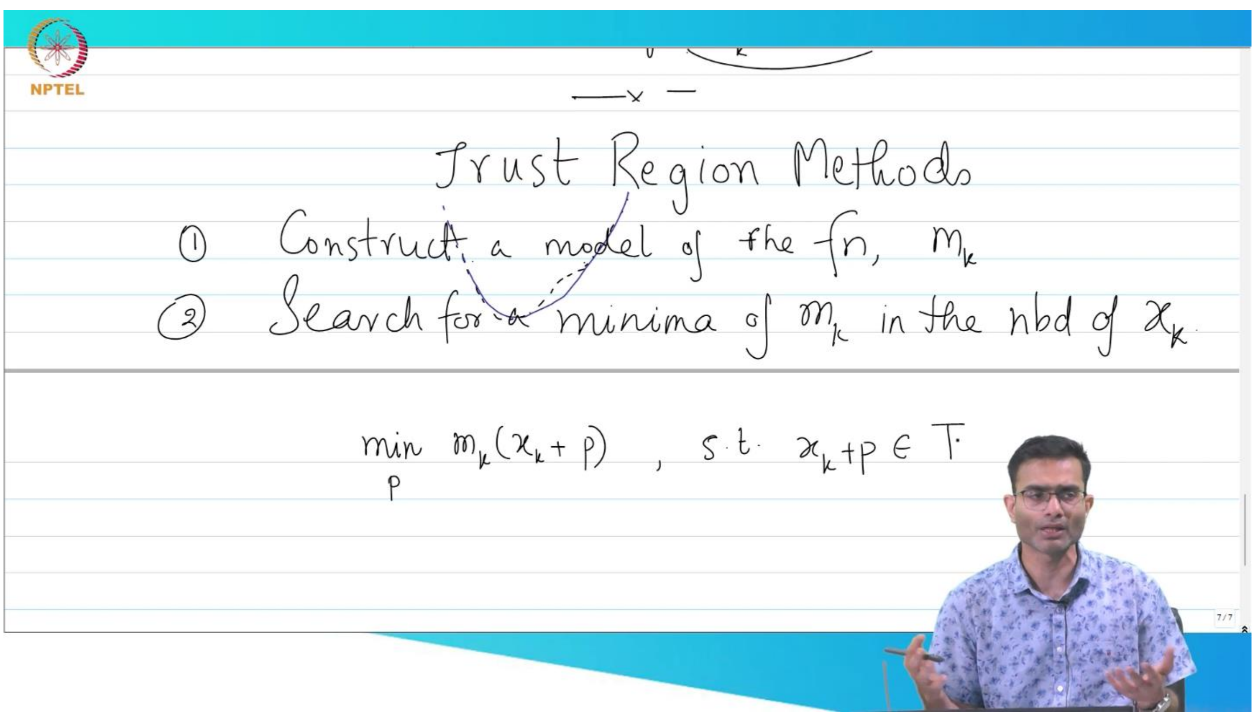
Course Name: Optimization Theory and Algorithms
Professor Name: Dr. Uday K. Khankhoje
Department Name: Electrical Engineering
Institute Name: Indian Institute of Technology Madras
Week - 03
Lecture - 21

Unconstrained Optimization - 7 - Trust Region Methods

So, this was as far as we wanted to talk in a general way about what are the qualities of a descent direction. The next thing that we want to briefly mention, but we will not cover in detail in this course, is the second family of algorithms. The first family is line search methods, and the second family is trust region methods. The trust region approach is kind of a complement to the line search. In the line search, I am trying to find the correct direction to go in. In a trust region approach, what I do involves two steps.

The first step is to construct an approximate model of the function. So, what does that mean? Let us say I have a complicated function that I cannot even write as sin, cos, tan, log, or whatever, right? It is something that is coming from data. Supposing I give you a bunch of points like this. Let us simplify and suppose I give you a set of points like this and ask you to make a model from these points. What would your first instinct be if you wanted to create the simplest possible model that explains this data? You would fit a quadratic, right? If I take a quadratic expression, I will get something like this—there will be some error somewhere, but this reasonably explains the data.

Once I have the model, what is the advantage of building it? I can do continuous math with it: I can take derivatives, second derivatives, and so on.



All the time, I have to keep in mind how accurate this model is, as it translates into how accurate my analysis is. So, step 1 is quite simple: construct a model, let's call it m_k. Obviously, the objective is that it should match the function's behavior in some region. Just like in 1D, where I can fit a parabola, I can fit a quadratic form in n-dimensions—the idea is the same.

The next and final step is to search for a minimum of this model in a neighborhood of x_k.

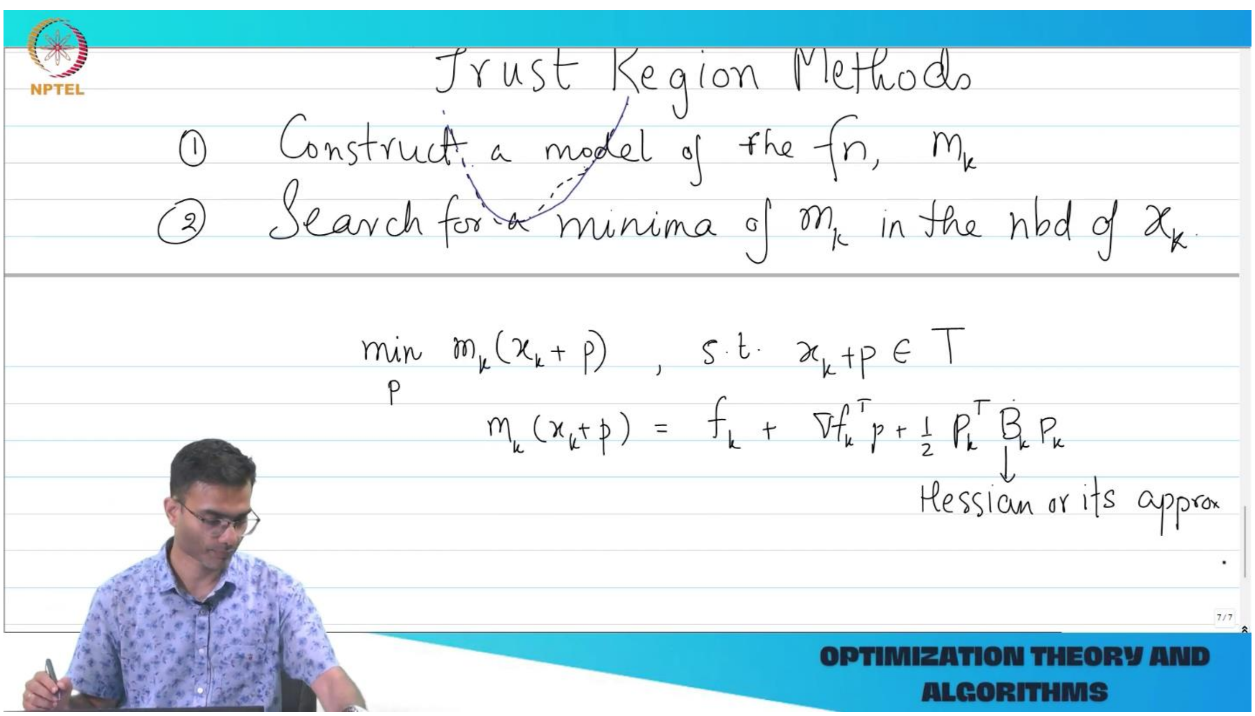
Search for the minimum of m_k, not f_k, in the neighborhood of x_k. So, what does the optimization problem look like? Now, I am trying to search for the best p such that

$$m_k(x_k + p)$$

I have my model m_k, which explains the function behavior near x_k, for example, a parabola or whatever. I then say, let me add a little bit of p to this and find the best p. In some sense, I am looking for the direction. It could be any p, but if I do not put any constraints on this, I will be in trouble because the model is accurate only near x_k.

So, what I will do is search for p such that $x_k + p$ belongs to some trust region T. That is where the term "trust region" comes from. Trust means I trust the model in that region, and I search for p's in that region.

So, this is, in a nutshell, the trust region approach. It does not look radically different from a line search approach, right? I mean, there are lots of similarities. That is why we will not cover this in detail in the course—if you understand line search, you can easily apply the same analysis to trust region methods. What are the ingredients we are working with? Essentially, Taylor's theorem and the legitimacy of descent directions, which also come from Taylor's theorem. That is really the only thing here.



I will just make one note: the most common model that people use in the literature is the quadratic model.

So, the most common model would be:

$$m_k(x_k + p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T B_k p$$

Now, what is this B_k? B_k is the Hessian or an approximation of it. I may not be able to get the exact thing, so this is the Hessian or its approximation.