Optimization Methods for Civil Engineering
Dr. Rajib Kumar Bhattacharjya
Department of Civil Engineering
Indian Institute of Technology, Guwahati

Search Methods
Lecture - 07
Bracketing Method

Hello student, welcome back to the course on Optimization Methods for Civil Engineering. So, already we have learned the necessary and sufficient conditions for optimality. So, I can apply necessary and sufficient conditions to find an optimal solution of a problem.

So, what we are doing in case of single variable function, we are taking the first derivative and that is the necessary condition. And first derivative should be equal to 0, and then we are obtaining the stationary points.

And once we are getting the stationary point, so we apply the sufficient condition and that is in case of single variable function we are taking the second derivative, third derivative, fourth derivative something like that. So, we are trying to find out the first nonzero derivative value ok.

And then we are looking at the what is the derivative if it is second derivative, then n is even. And in that case, we look at what is the derivative value ok, whether it is positive or negative. If it is positive, then that stationary point is a minimum point; and if it is negative, then stationary point is a maximum point. Or if it is an even, so in that case that is an inflection points.

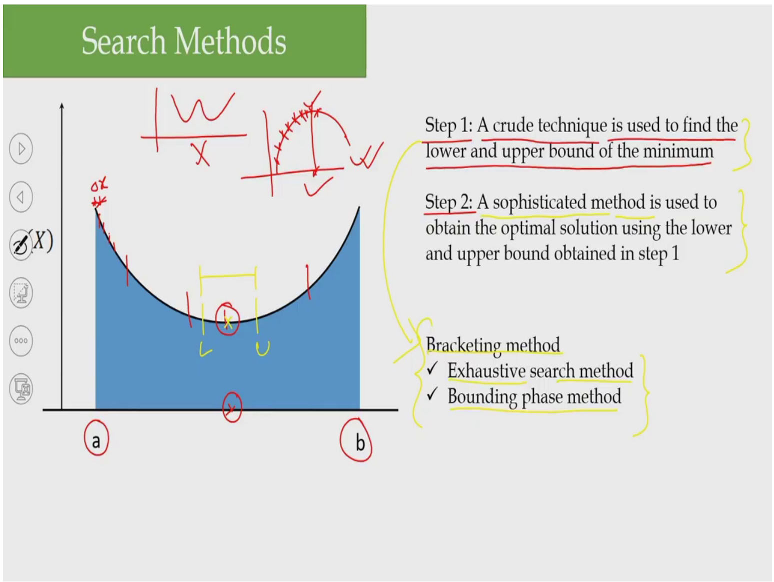
So, similarly, we can solve a problem with multiple variables. So, we will apply the first derivative, and then we will equate to 0 and 1. And once we are getting the stationary points, then we calculate the Hessian matrix. So, in case of multivariable function, the second derivative is a Hessian is a matrix, and that matrix is known as Hessian matrix.

So, what we will do, then after that we are checking whether that Hessian matrix is a positive definite matrix or a negative definite matrix. If it is positive definite, then the solution or stationary point is a minimum point; and if it is a negative definite matrix, then that is a maximum point. So, we can apply this necessary and sufficient conditions, and we can find out the stationary points as well as we can also determine whether that particular solution is a minimum or a maximum.

So, the problem is that for a bigger function, suppose smaller smaller function one or two variable, or three variable, four variable ok, you can do that, but you will just look at a problem of thousand variable or problem of hundred variable fifty variable. So, calculation of Hessian matrix looking at whether it is a positive definite, negative definite, I think this is quite difficult. So, therefore, we need some other algorithm to find out the optimal solution ok.

So, today we will discuss about some of the algorithms to find out the optimal point. So, in this case basically I will discuss about a function that is unimodal function having a minimum. So, unimodal function that means there is only one minimum ok. So, I will take that function, but this can be applied for a function having maxima also, but I will take a function having one minima, that means, unimodal function.

(Refer Slide Time: 04:07)



Now, let us see what are these algorithms. As I said so I have taken a function having one minimum. So, this is a unimodal function. So, that means, there is one somewhere here. So, this is unimodal. Suppose, I may have a function of maybe like this, so this is not unimodal function. So, in that case, we have more than one minima. And so for this function, this is an unimodal function and there is only one minimum ok. So, I have taken this type of and this is this function has only one minima.

So, I can apply the same algorithm for a function like this. This is also in unimodal function having one maximum. So, this is the maximum of this particular function. So, I can also apply the algorithm on that function also, but let us take this example, so where the function has one minimum unimodal function having one minimum. And search space is between a and b. So between a and b, there is a minimum, and I would like to find out this particular minimum. So, I need an algorithm to do that ok.

Now, before that suppose for this function. So, this function I can compare. So, I am telling about the function where we have unimodal function where we have one maximum. So, this problem can be compared with a hill climbing problem. Suppose, this is the top of the hill. So, this is the top of the hill, and I would like to I would like to find out the top of the hill. So, what I will do basically?

Suppose you are here. So, what you will do? Then you will give one step here, one step here, one step here, one step here, one step here. And so basically so every step, you have a feeling that ok you are going up basically, so that means, you have not reached the top of that particular hill.

So, once you are reaching here and you are giving another step. So, you are going down basically. So, in that case, you can say because this is a unimodal function there is only one maximum. So, in that case you will feel or you will say that ok I have reached the optima. Now if I go further, so in that case, I will go down basically. So, I am already at the top of this particular hill.

So, this problem can be compared with a hill climbing problem. Suppose if I put a blindfold on your eyes, and then I ask you that ok you go, and find the top of that hill. So, somewhere you are here. So, what you will do basically, so you will give one step you are going up, second step you are going up, third step, fourth step, fifth step like that.

So, once you are going down, then you will say that I have crossed that particular peak. And that particular peak is between your previous, previous step and this step, current step.

So, in that case, so this same concept I can apply here, and I can find out minimum or a maximum of a unimodal function. So, whatever I am telling please note that this is a unimodal function, that means, this is a unimodal and this is not a unimodal function. So, this algorithm is only applicable for a unimodal function ok.

So, now, suppose you are away from the optima. Suppose, I am here, and you are optimized somewhere here. So, you are away from the optima. So, if you are doing this, suppose if you are going step by step. So, what you will do?

Basically, you will take one step, one step, one step, something like that. So, I have to fix the step length ok. So, maybe I can say this is delta x. Now, what should be the step length? Suppose, if you say the step length is very small, so in that case, you need more iteration to reach this particular point.

Now, you may take a larger step length. So, larger step length means you are here, you are here, and next step will be here. So, you will cross the optima. So, therefore, so what type of algorithm you will apply? So, if you are applying a very sophisticated algorithm, so that will give you the optimal solution, but it may take lot of time because you are away from the optima. So, therefore, we are solving this problem in two steps ok.

So, we are solving this problem in two step. So, in the first step my interest is not to find the exact optimal solution of this particular problem. So, what I would like to do, I would like to bracket the optima. So, basically a crude technique is used to find the lower and upper bound of the minimum. So, what I am doing in the first step, I am applying a crude algorithm.

So, it is not a very sophisticated algorithm, but this algorithm is very efficient in terms of computational time, that means, or you can say with a very less number of iteration I can bracket the optima. So, I can say that optima is between these two points basically, so that is the objective of step 1.

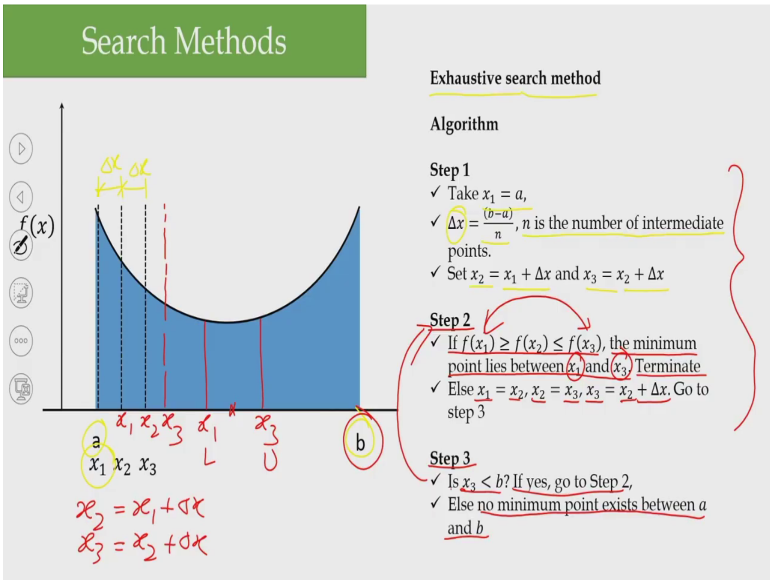
So, in step 1, so I am applying an algorithm and you can say this is not a sophisticated algorithm, it is a crude technique. And to find out lower and upper bounds of the minimum ok. So, you are getting. Suppose, here this is the optima this is the optima. So, I would like to bracket the optima. So, that means, I would like to find out this is lower bound, and this is upper bound, and then I can say that my optima is between this narrow region.

And then I can apply a little bit sophisticated, I can apply a sophisticated algorithm to find out the exact optimal solution of this problem, so that is the objective of step 1.

And in step 2, what you are doing you are applying a sophisticated method to obtain the optimal solution using the lower and upper bounds obtained in step 1. So, whatever lower and upper bounds you have obtained in step 1 that will be used in the step 2, and we will be able to find out the exact optimal solution of this problem with desired precision certainly ok.

So, now whatever step 1 algorithm, so we call it bracketing algorithm ok. So, we call it bracketing algorithm or bracketing method. So, here we will discuss two methods ok. One is exhaustive search method and other one is bounding phase method. So, we will apply these two algorithm to bracket the optima. So, let us discuss these algorithms ok.

(Refer Slide Time: 10:59)



So, the first algorithm as I said this is exhaustive search method ok. So, here what we are doing, so we have that lower bound and upper bound. So, lower bound is a; upper bound is b. Now, so we will take we will start from the lower bound. So, we will consider that x 1 is a point which is equal to a, that means, this particular point we will consider x 1.

Then we will take a delta x that is the step length. So, delta x is b minus a by n. So, where n is the number of intermediate points ok. So, number of intermediates point. So, I need n numbers of intermediate points. So, you can say, so I will get the step length ok.

So, once you are getting your step length, then take another two points that is x 2. So, x 2 equal to x 1 plus delta x, and x 3 equal to x 2 plus delta x. So, what we are doing basically, so this is your delta x. So, you can say that this is delta x, and this is delta x ok.

So, we are getting that x 2, x 2 equal to x 1 plus delta x. And similarly, I can get x 3 which is equal to x 2 plus delta x. So, x 1 is a, and then I am getting another two points that is x 2 and x 3, where x 2 equal to x 1 plus delta x, and x 3 equal to x 2 plus delta x ok.

Then what you are doing in step 2? We are checking these conditions. So, whether f of x 1 is greater than equal to f of x 2, and f of x 3 is greater than equal to f of x 2, so if these conditions is true or if it is satisfied, the minimum point lies between x 1 and x 3. So, it is between x 1 and x 3. Then we will terminate the search process. So, we will say that lower bound is x 1, and upper bound is x 3. So, then we will terminate the search.

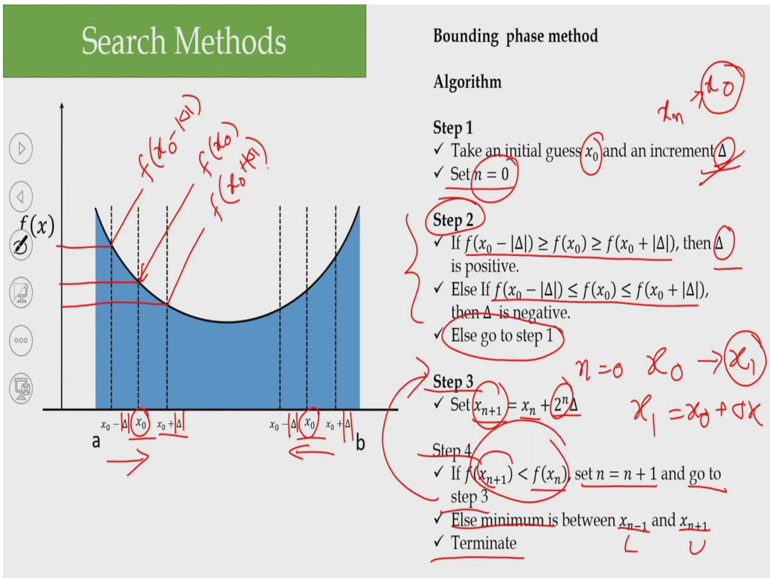
So, if it is not satisfied ok, if this condition is not true, so then what we will do basically, so we will go to the next step. So, in that case, x 1 equal to x 2, x 2 equal to x 3, and x 3 equal to x 2 plus delta x. So, if that condition is not true, so we will take another, another point. So, now, this will be your x 1, this is x 2, and this will be your x 3.

And then go to step 3. So, in step 3, what we will do basically in step 3, we will check whether x 3 is less than b because the upper bound is b. So, we cannot go beyond b. So, if x 3 is less than b, if yes, go to step 2 ok. So, from here, so you go to step 2 if it is yes. If no basically that x 3 is more than b, so in that case, that no minimum point exists between a and b. So, there is minimum point between a and b. So, we have to take a different lower bound and upper bound.

So, after applying this, so you just see. So, this is one algorithm. So, to bracket the optima, so once, so I can use this algorithm and you are optimize somewhere here. So, finally, what will happen, this will be your x 1 and this will be your x 3, or I can say this is your lower bound, and this is upper bound. So, we have bracketed the optima ok.

So, you are getting a very narrow region. And after that, you can apply a different algorithm so which will be very efficient in terms of obtaining the optimal solution, but may not be very efficient in terms of computational time, but anyway so because we have a very narrow region. So, therefore, if we apply that algorithm, so it will not take much time to get the optimal solution of this particular problem ok. So, this is one algorithm to bracket the optima.

(Refer Slide Time: 15:51)



Let us see the other one. So, other one is bounding phase method ok. So, in this method, what we are doing, we are not considering upper bound and lower bound, but in this case we are starting our algorithm from x naught ok. So, we are starting our algorithm from x naught. And x naught is a point between our upper bound and lower bound. So, this is your x naught now set n equal to 0 ok. So, n equal to 0 means I am telling x n. So, now, n equal to 0, so therefore, I am getting x 0 ok.

Set n equal to 0, and also set a delta, delta is the step length or you can say increment ok. So, how you are taking the next point that is delta. So, in the first step what we are doing, we are choosing an initial point. So, in this case, so I have chosen this particular point. So, this is your x naught. And I have also chosen a appropriate value of delta. And I put n equal to 0. So, therefore, x and n equal to 0 means I am getting x naught.

Now, in step 2, what we are doing, step 2 we are checking this particular condition ok. So, what we have done here. So, we have taken; we have taken two points, one is x naught minus delta, one is x naught plus delta. So, we have taken two points. So, one is on the left hand side that is x naught minus delta; and one the right hand side, that is x naught plus delta. So, we are taking two points ok.

Now, after taking these three points, so we are checking this condition whether f of x naught minus absolute value of delta is greater than f of x naught, and f of x naught is greater than f of x naught plus absolute value of delta ok.

So, we are checking this condition, that means, this is the particular condition you can see this is the value of f of x naught minus delta. Is it not? And this is the value of f of x naught, and this is the value of f of x naught plus delta ok absolute value of delta. So, I can find out these values ok from the y-axis.

So, now, in this case, what is happening this condition is true, that means, f of x naught minus delta is greater than f of x naught, and f of x naught is greater than f of x naught plus delta of say absolute value of delta. So, this condition is true ok. So, if it is true, then what you will do, then delta is positive, that means, I will go in this particular direction ok.

So, if I go along that direction, I will be able to bracket the optima. So, what you are doing, we are just checking whether delta is positive or negative. In step 2, if this condition is true, that means, delta is positive ok.

Now, if suppose if this is my x naught, so in that case, this is x naught minus delta absolute value of delta, and this is x naught plus delta ok. So, now, if this condition is true, that means, f of x naught minus absolute value of delta less than f of x naught; and f of x naught is less than f of x naught plus absolute value of delta. So, in that case, delta should be negative because I would like to go in this direction. So, delta should be negative in this case.

So, in the second step, what we are doing, we are trying to find out whether you are going in the positive direction or whether you are going on the negative direction. So, basically I would like to find out whether delta will be positive or negative for the initial point you have chosen ok. So, if the initial point is here this is in that case delta should be positive; and if it is on the other side, delta should be negative ok. So, in the second step, you are doing that one.

So, now, you know whether in which direction you should go. So, whether you should go in the positive direction or you should go in the negative direction ok. So, if none of this is satisfied, then go to step 1 ok. So, and you find out a different value of x naught or different value of delta. So, probably you are not getting a taking appropriate value of delta. So, you use some different values of x naught and delta ok.

Now, in a step 3, what you are doing basically that you are finding. So, initially n equal to 0 that is my x naught ok. So, that is my x naught now you are finding n plus 1, that means, I would like to find out what is x 1 point ok. So, x 1 now x 1 equal to x naught plus 2 to the power n into delta ok. So, what I am doing here I am multiplying 2 to the power n, suppose in the first step n is equal to 0 ok. So, n is 0. So, this value is 1.

So, basically x 2 ok or x 1 equal to x naught plus this is delta because 2 to the power n will be equal to 1, 2 to the power 0 that is that is equal to 1 ok. So, with iteration, the idea is that the with iteration you increase the step length. So, that very quickly you reach the optima. Suppose, you are not getting your optima, so in that case, you increase your step length ok. So, that is the idea that is why we are using two to the power n into delta ok.

Now, in step 4, what you are doing that you are getting a new your point, new point is x n plus 1. Now, if f of x n plus 1 is less than the f of x n, that means, previous your point. So, in that case, what is happening that you are still you are not crossing the optima. So, in that case, I will set n equal to n plus 1 that means, you just increase the value of n by 1 and go to step 3 ok.

So, we will go to step 3. So, you are trying to find out a different point. So, if this is not true, if this is not true, that is f of x n plus 1 is less than f of x n, that means, you have cross the optima ok. So, in that case, you can say that minimum is between x n minus 1 and x n plus 1. So, this is basically you can say this is the lower bound, and this is the upper bound.

x n minus 1 is the lower bound, and x n plus 1 is the upper bound. And you terminate the algorithm ok. So, if you are getting that solution. So, I hope this is clear to you. So, basic idea of these two algorithms is to bracket the optima that means I do not want to find the exact optimal solution of this particular problem. So, what I want to do, I want to bracket the optima that optimize between that narrowed region that means that between your lower bound and upper bound.

So, that I can apply a sophisticated algorithm because thus the this algorithm whatever we will discuss later on I said that sophisticated algorithm. So, they are very good in find the optimal exact optimal solution of the problem, but they may not be very efficient in terms of computational complexity ok. So, it may take your some time to get the optimal solution.

So, therefore, I would like to apply this algorithms on a very narrowed region where optima is there. And so this is a very crude algorithm to bracket the optima. So, once you are you have bracketed the optima, so after that you just apply the other algorithm as I said that means, sophisticated algorithm to find out the optimal solution of the problem.

So, whenever I say the exact optimal solution, so exact optimal solution you may not get because you have to rely with some your precision suppose I need a value which is your precise enough suppose 10 to the power minus 3, 10 to the power minus 6, 10 to the power minus 15 ok. So, if my that level of precision is reach, so I can say that is the optimal solution of this particular problems ok. So, that is basically you can find out that is you can find out using the other algorithms ok.

So, I hope this is clear to you. So, here we are not trying to find out the optimal solution of this problem, but we are trying to find out the region, a small region where optima is there ok.