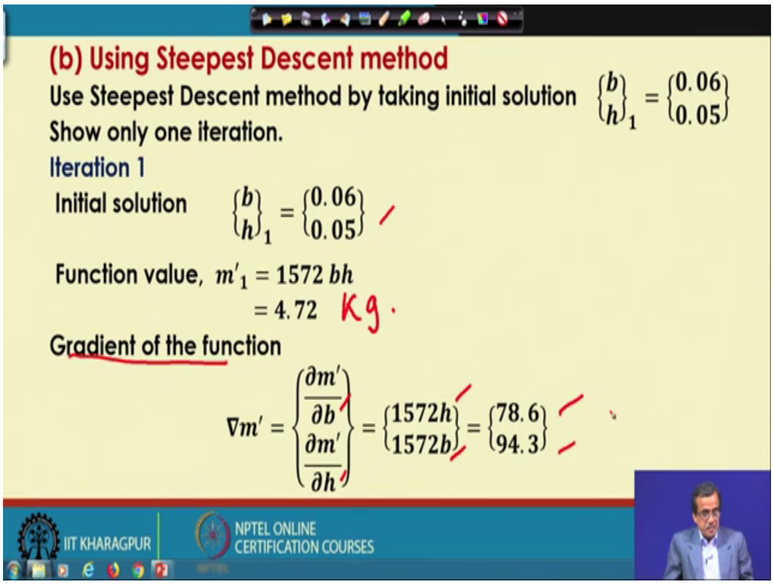
**Traditional and Non-Traditional Optimization Tools**
**Prof. D. K. Pratihar**
**Department of Mechanical Engineering**
**Indian Institute of Technology, Kharagpur**

**Lecture - 29**
**A Practical Optimization Problem (Contd.)**

Now, I am going to discuss how to use the concept of the steepest descent method to solve the same optimization problem.

(Refer Slide Time: 00:27)




That is how to find out the optimal design of this particular the single point cutting tool. Now as I told that steepest descent method is actually one of the most popular Traditional Tools for Optimization.

Now, let us see how to use the principle of steepest descent method to solve this real world optimization problem. Now here we start with one initial solution selected at random that is b h 1 that is nothing, but 0.06 0.05. Now, this initial solutions is selected at random and let us see how to proceed with this particular the iteration, iteration 1.

So, we will start with the initial solution we start with the initial solution that is b h 1 that is nothing, but 0.0 6 0.05. We calculate the function value m 1 prime, that is 1572 b h and if we just substitute this part b b is 0.0 6 h is 0.05.

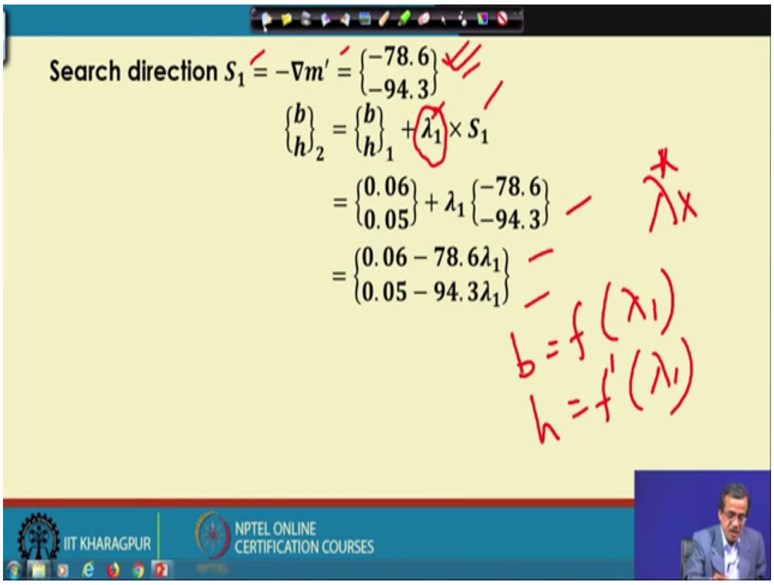
So, I will be getting 4.7 2 kg. So, this is the mass of this particular the single point cutting tool and here the search direction is opposite to the gradient.

So, what will we have to do is we will have to find out the gradient of this particular objective function that is nothing, but delta m prime that is the partial derivative of m prime with respect to b partial derivative of m prime with respect to h.

And here if I use the partial derivative so, with respect to b I will be getting 1572 h with respect to h I will be getting 1572 b. And if I substitute the values of this b and h so, I will be getting 78.6 and 94.3.

This is actually the gradient information of the objective function.

(Refer Slide Time: 03:02)




Now here the search direction should be opposite to the gradient, because this is this particular algorithm can solve the minimization problem and it moves in a direction opposite to the gradient that is why that is called steepest descent method.

So, the search direction is nothing, but S 1 is minus delta m prime and that is nothing, but minus 78.6 minus 94.3 and if I substitute the values. So, I can find out what should be this particular b h 2; that means, what will be the next solution.

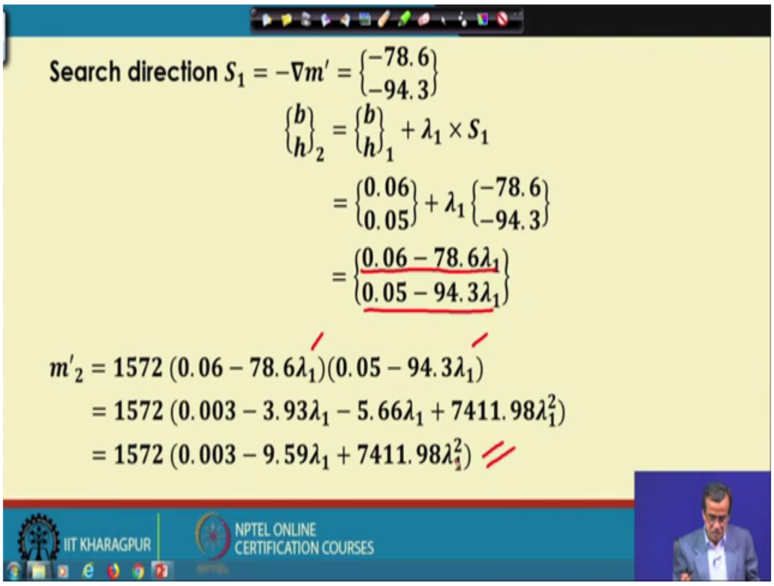
So, b h 2 is nothing, but b h 1 plus lambda 1 multiplied by S 1 lambda 1 is actually the step length and here we will try to find out what should be the optimal step length. And S

1 is nothing, but the search direction which I have already calculated. So, S 1 is known, but I will have to find out what is the optimal value for this particular the lambda 1?

Now, to determine the optimal value of this particular lambda 1 so, what we do is we substitute the values of this b and h b h 1 is nothing, but 0.06 0.05 plus lambda 1 into S 1 S 1 is nothing, but this. So, we substitute and we will be getting this and if you simplify. So, we will be getting this particular the expression.

Now, you see that both b and h are function of only 1 variable that is nothing, but lambda 1. So, b is nothing, but a function of lambda 1 and your h is actually another function of your this particular lambda 1. Now what you can do is we can actually take, we can find out what should be this particular lambda 1 star that is yours.

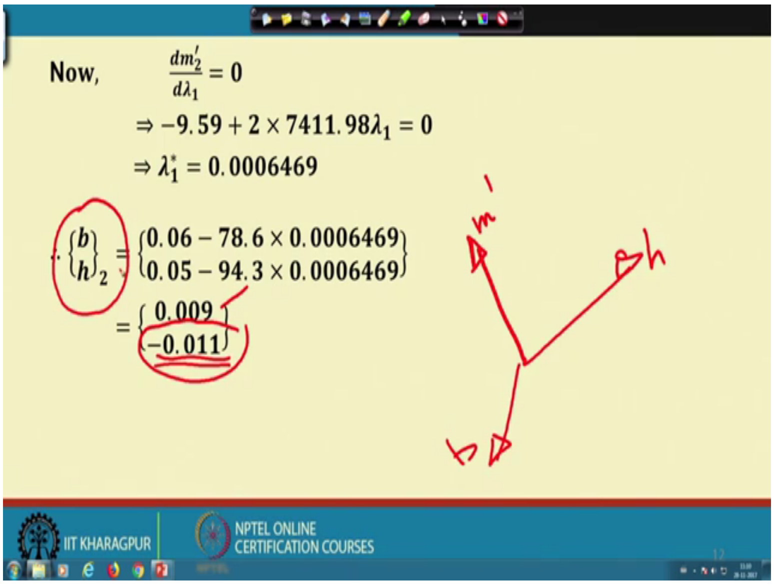
(Refer Slide Time: 05:28)




Now to find out the lambda 1 star actually what we do is we will substitute the values of this b and h in the expression, we substitute the value of b and h in the expression of objective function.

So, m 2 star is nothing, but your 1572 into b b is nothing, but 0.06 minus 78.6 lambda 1 h is nothing, but your 0.05 minus 94.3 lambda 1 and if you simplify. So, I will be getting this particular expression. And now actually we can see that your this m 2 is a function of lambda 1.

(Refer Slide Time: 06:27)




Now, to find out the optimal value of this particular lambda 1 what we do is we find out the derivative of m 2 prime with respect to lambda 1 and this is put equals to 0. If you remember the way we tackle the single variable optimization problem at the beginning of this particular the course. The same principle I am using it here and if you carry out this differentiation put equals to 0. So, I will be getting the optimal value for this particular the lambda 1 star.

Now, what is this this is nothing, but the optimal step length. Now till now we have considered we have determined the optimal search direction and the optimal step length. And once you have got those 2 things, now I am in a position to find out this b h 2 and here the b h 2 if you see the earlier expression of this. So, 0.0 6 minus 78.6 lambda 1 0.05 minus 94.3 lambda 1, so those things if you put here so, I will be getting what should be the values for b and h in the next iteration.

And if you calculate so you will be getting the values like b is 0.0 0 9 and h is nothing, but minus 0.0 1 1. So, this type of possible solutions we are getting for this particular b h 2.

Now, we will have to check the feasibility of this particular solution. Now to check the feasibility first let us concentrate on b b is coming to be equal to 0.0 0 9. So, this is lying within it is range, but you can send it on h is minus 0.0 1 1. Now that means, this is negative, now this h cannot be negative first of all.

And moreover using this particular principle whatever h we got that is also coming out of this particular the range of the variable.

This is actually one of the serious drawback of this particular the so, called very popular traditional tools for optimization that is steepest descent method there are some other drawbacks, which I already discussed, but let me once again mention that particular the drawback.

Now supposing that that we are solving a problem optimization problem having said 2 variables like b and h. And there is a possibility that there could be there could be some local minima problem and this local minima problem occurs due to the search direction.

Now, if you see the surface of the objective function as this is a function of 2 variable. So, the surface is in 3 dimension. So, we can visualize we can plot that particular the surface of the objective function as a function of b and h.

For example, if we want we can just draw the b and h as a function of this particular your for example, say if I consider say this is nothing, but say b this is nothing, but h and this is your. So, this particular objective function let me consider this is m prime.

Now, we can visualize the surface of this particular objective function which is in 3 dimensions and this particular gradient is actually a local property of the objective function. Now supposing that there are some undulation of this particular objective function, it is hypothetical there are some undulation ups and downs and all such things there is a possibility, that this particular algorithm is going to a get stuck at the local minima, because gradient is a local property.

So, in at a particular point in a particular basin there could be a particular search direction, but that will vary from point to point your basin to basin. And there is a possibility that this particular algorithm is going to hit your that particular the local minima problem and it may not be able to reach that globally optimal solution.

But on the other hand this particular algorithm is very fast particularly for the unimodel function, but fortunately or unfortunately for the real world problem there is no guarantee that you will be getting a very well defined in unimodal surface for this particular the objective function.

And that is why our experiences most of the time this particular algorithm is going to fail to solve the real world optimization problem. Now supposing that say here we did not get the feasible solution, but supposing that we are getting some feasible solution. And the values of b and h are lying within the respective ranges, but till now whatever we have solved is the unconstrained optimization problem. Like single point cutting tool, we want to find out the optimal design, but we did not consider the functional constraint till now.

So, what you will have to do is once this particular solution is found to be better we are going to keep it if an only if the functional constraint is not violated. So, once we have got some feasible solution for b and h. Now we check it is feasibility further through this particular the functional constraint, if that particular constraint is fulfilled then only we declare that this is a valid solution for the next iteration.

And this completes actually 1 iteration and this will go on through a large number of iteration and ultimately through a large number of iteration there is a possibility that you will be getting some optimal solution, but of course, there is no guarantee that we will be getting the optimal solution and even if we get there is once again no guarantee that will be the globally optimal solution or the globally minimum solution.

So, till now to solve this particular real dual problem we tried with 2 traditional tools for optimization. Now you see these both the traditional tools for optimization showed some difficulty in solving even this particular very simple real world problem. And now I am just going to show you how to use the non-traditional tools for optimization and we will see that these type of problems can be tackled very easily and efficiently using some non-traditional tools for optimization.

Now, what I am going to do next the same real world problem, I am just going to use some non-traditional tools for optimization to solve it and the performance of these non-traditional tools for optimization to solve the same real world problem I am just going to discuss and compare 1 after another.

Thank you.