**Design and Optimization of Energy Systems**
**Prof. C. Balaji**
**Department of Mechanical Engineering**
**Indian Institute of Technology, Madras**

**Lecture No. # 09**

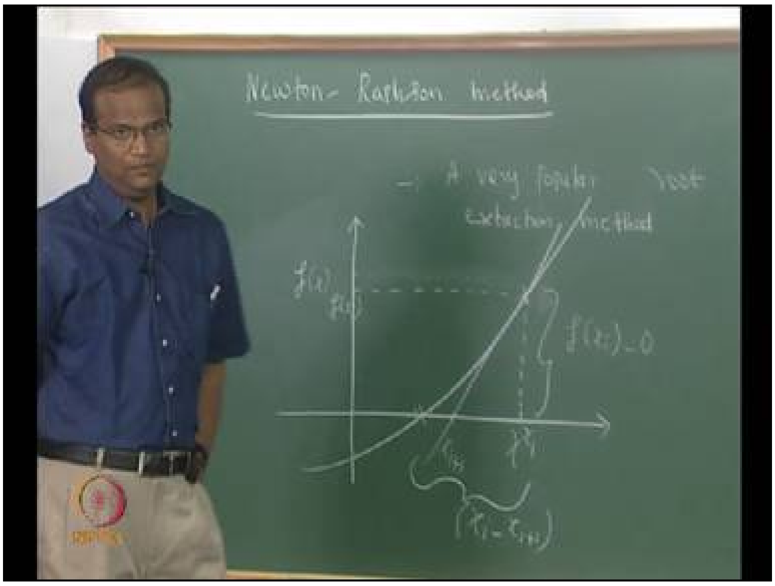
# Newton-Raphson Method Contd…

We will continue with our discussion on the Newton-Raphson method. I quickly derived the Newton-Raphson method on the blackboard by using graphical interpretation. We will go through the derivation little more slowly. Then, we will see the important points considering the Newton-Raphson method. As is our practice, we will solve some problems using the Newton-Raphson method. First, with one variable; and then, we will go on to solve the two variable problem using Newton- Raphson method. In order to see the potency and the power of the method, it would be better if we take a problem, which we have already solved using the method of successive substitution. So, we may decide either on the fan and duct problem or the… What is the other thing?

Student: The engine characteristics…

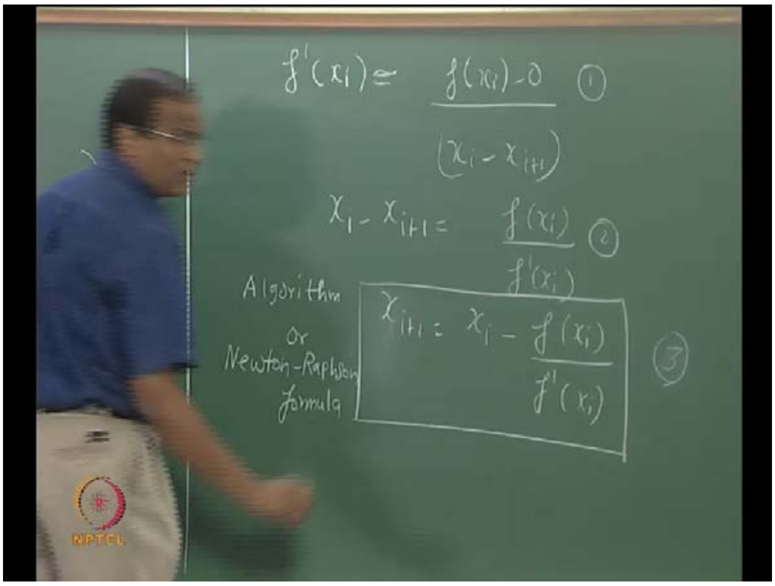
The engine characteristics problem. And then, we will see how if we apply the Newton-Raphson method for multiple unknowns, whether it fares better or not.

(Refer Slide Time: 01:16)



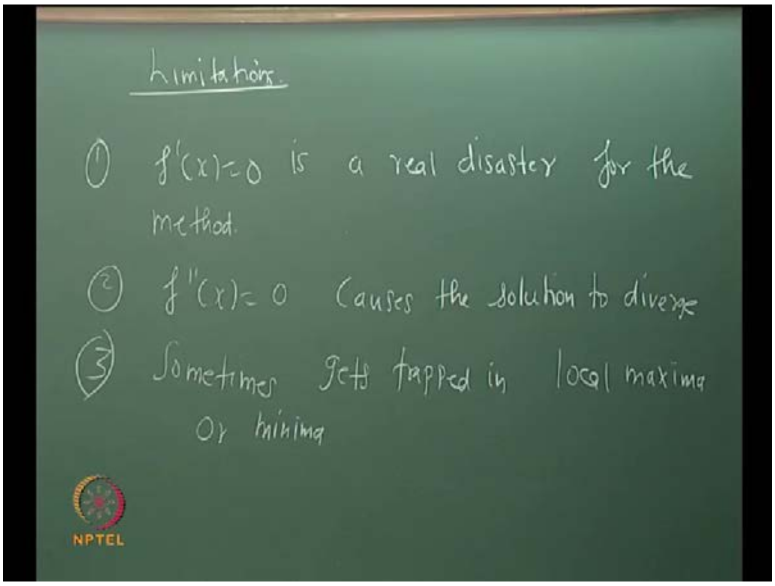

Newton-Raphson method is basically a root extraction method. Why do we say it is a root extraction method? Because we are trying to set f of x is equal to 0. You are trying to find the values of x at which f of x tends a numerical 0; because these equations cannot be analytically solved. Invariably, we encounter non-linear equations. So, if you look at the… Closer part of f of x was an x for a typical function, whose roots we are seeking to obtain. As you all know; as you are all able to see, the root is here. So, this is basically again iterative technique. So, we start up from somewhere. I call this as x i. This is xi. Corresponding to this we have f of x i. The root is somewhere far off. How does this method work? Draw a tangent at x i. Wherever this tangent intercepts the x axis, we call it as x i plus 1; the next iterate. This is f of x i minus 0; f of x i minus 0. So, this is x i minus x i plus 1. You are able to see Anand? Please change your place. All right?

(Refer Slide Time: 04:03)




Now, what is the algorithm? f dash of x i is approximately equal to… 3. We can call this as the algorithm. Just call it as the Newton-Raphson formula. These are the most important and powerful techniques for solving a single variable problem. We can extend it to multiple variables. It is a very very powerful technique. As engineers, you must not leave this place without knowing the Newton- Raphson method; this you should never forget – x i plus 1 equal to x i minus f of x i divided by f dash of x i. So, what happens? The next one is x of i plus 1. So, you can go here (Refer Slide Time: 05:57) and draw a tangent. Hopefully, it will converge to this within a few iterations. The major difference between this method and the method of successive substitution is it is using information of the derivatives. Since you are using the information of the derivatives, the convergence hopefully should be very fast. But, we cannot say that the convergence is always guaranteed. Why do I say that? Look at the formula and tell me what could be the possible fit for the limitations or short comings for the Newton-Raphson method?

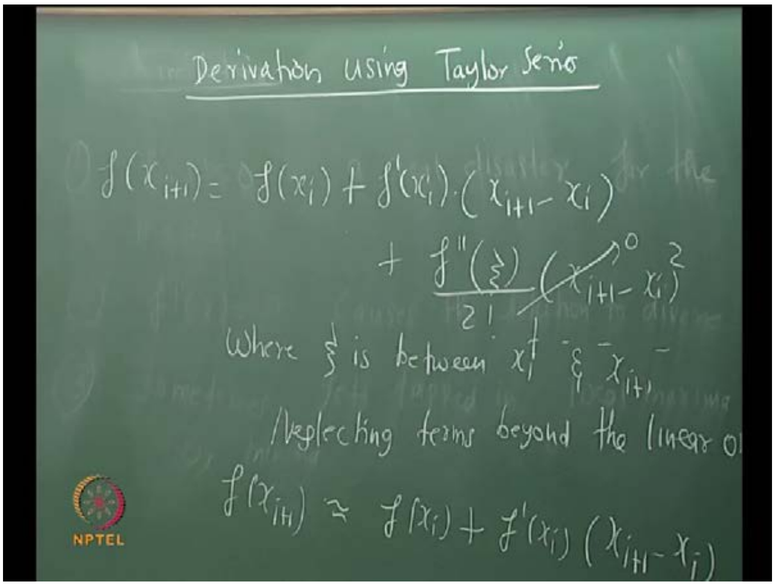
(Refer Slide Time: 06:43)




Limitations – let us not write the advantages. We are able to see the advantages. So, it uses information of first derivative. It should be fast; it should rapidly converge; all that. Let us see the limitation, which are not so apparent upon first sight. And, f dash of x is equal to 0 for any iterate is the real disaster for the method. f double dash of x is equal to 0 causes the solution to diverge. f dash of x need not even be 0. If it is very small also, this fellow will oscillate too much; this fellow will shake too much. That is the problem. So, it depends on that f of x divided by f dash of x should be descent such that it is small compared to x of i, so that you get the new x of i plus 1. So, sometimes causes the solution to diverge; sometimes gets trapped in local minima or maxima. For each of this, we can take a particular example. Each of this we can take a particular example and show when it converges slowly; when f of dash tends to 0, it becomes disaster and all that. But, again, please be reminded that, this is not the course MA401 numerical analysis or something. So, we are looking at optimization. So, I am going to stop at certain stage and those people are interested; you can always look at some books on numerical analysis to get more ideas about this method.

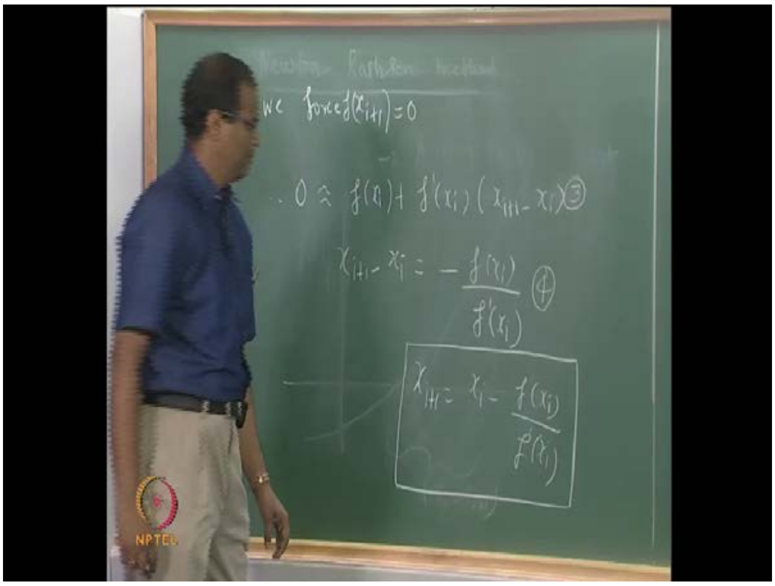
Now, how do we get this algorithm using Taylor series? This is a graphical depiction. It is possible for us to get the same algorithm using the Taylor series expansion.

(Refer Slide Time: 09:46)



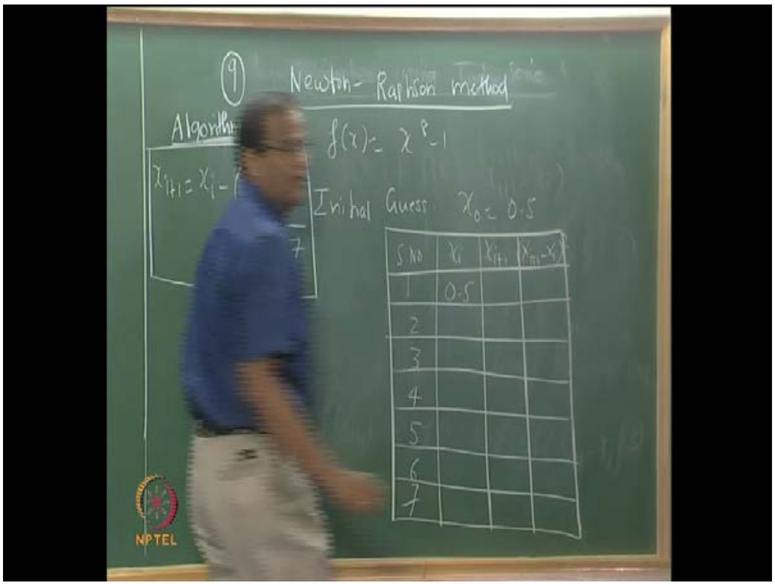

Shall we get the same thing using the Taylor series? It is always confusion whether it is Taylor series or Taylor's series, that is, Taylor apostrophe s or Taylors series or Taylorses apostrophe. So, there are several versions. I am using the plain vanilla; simple Taylor series. Even if I make mistake, it is all right. There is a level of simplicity associated with this Taylor series expansion. Now, how do we do the Taylor series expansion on f of x of x i in the vicinity of x i? f of… Can you proceed? f of x i plus 1 is equal to f of x i… Can you proceed? We can call it as x of i plus 1…; where, xi is somewhere between x i and x i plus 1. This term plus higher order terms. Of course, in close to the true solution, it does not matter and all that. When x i and x i plus 1 are also close to the true solution x of x true; it does not matter. Now, this becomes highly mathematically involved question; why it should be xi; where you want to evaluate this second derivative and so on. Let us not get into also those debates. Now, if x i plus 1 and x i are sufficiently close or the second derivative is small; then, we neglect the second and higher order term. Neglecting… Does that lead to the same solution? What are we trying to make 0? Which is that we are trying to make 0? 1, 2.

(Refer Slide Time: 13:28)




Minus? Is that a minus? Using Taylor series expansion also, we get the same result. Since both the graphical methods in the Taylor series expansion give the same result, this must be the correct version of the algorithm. Now, we will start off with an example, where the Newton-Raphson method is not powerful at all. So, I will demonstrate you the simple example that all is not too well. Everybody says all is well and all is well; all is not too well with the Newton-Raphson method. Let us take an example like this. So, it is got to be problem number 8, is it? Good.

(Refer Slide Time: 15:34)



Problem 9. It is not exactly equal. At some place, we are putting exactly equal. But, I want to remind you that, we are chopping up so many terms – higher order terms.

Student: (( )) not true.

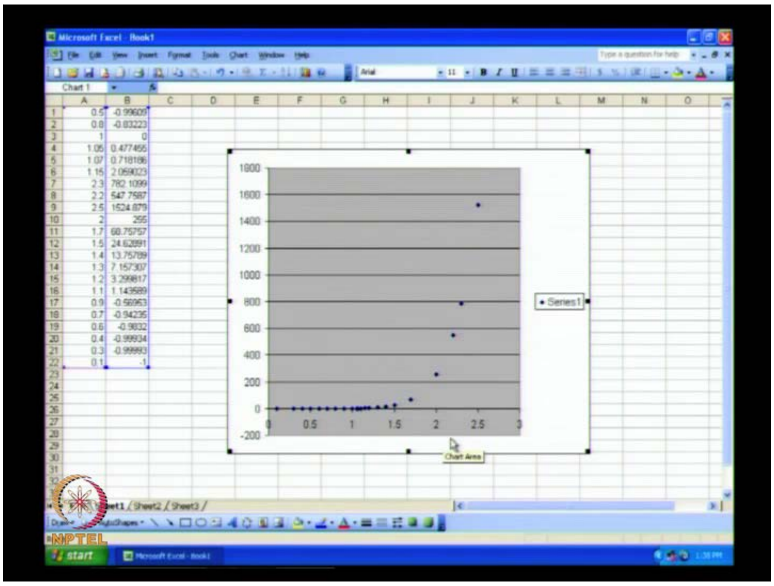
No, when I use this, graphical method is all right. You can put that; that is no problem. But, here you have to be careful; graphical method – there is no problem. But, in the Taylor series expansion, I want you to remember that, there are higher order terms, which we neglect. So, we have to talk about the convergence of this method, which involves some little more advance math. That 5 o'clock we will do it. Please be reminded that, we have class at 5 o'clock. Anyway, we will be working on lot of problems on Newton-Raphson method. So, it should be fun. Problem number 9 – determine the root of the equation; determine the root of the equation, f of x equal to x to the power of 8 minus 1; x to the power of 8 over minus 1; it is not x to the power of 7.

Evaluate the root of the equation f of x equal to x to the power of 8 minus 1 using the Newton-Raphson method with an initial guess of x equals 0.5; with an initial guess of x equals 0.5. Many people call it as x naught; initial guess is x naught. Answer is so obvious. But… So, it is very painful. What is the answer?

Student: 1

1. The illusive 1. We start with 0.5. You see the fun. Shall we first write the algorithm? Before start; before you begin to draw the tabular column; first, write the algorithm – x i plus 1 is equal to x minus… Bracket is not required. That is the algorithm.

(Refer Slide Time: 20:08)



You can. Enough? Is equal to? What happened? Does it go like this?

Student: (( ))

I am not… I am just showing how this function behaves. This is too much, we have to now rigid. What do we do now?

Student: (( )) divided by (( ))

Not; 3.2 is also bad man. All values are decent or some indecent values are also there? So, let us plot. Now, tell me why is this fellow misbehaving?

Student: The slope at the guess point (( ))

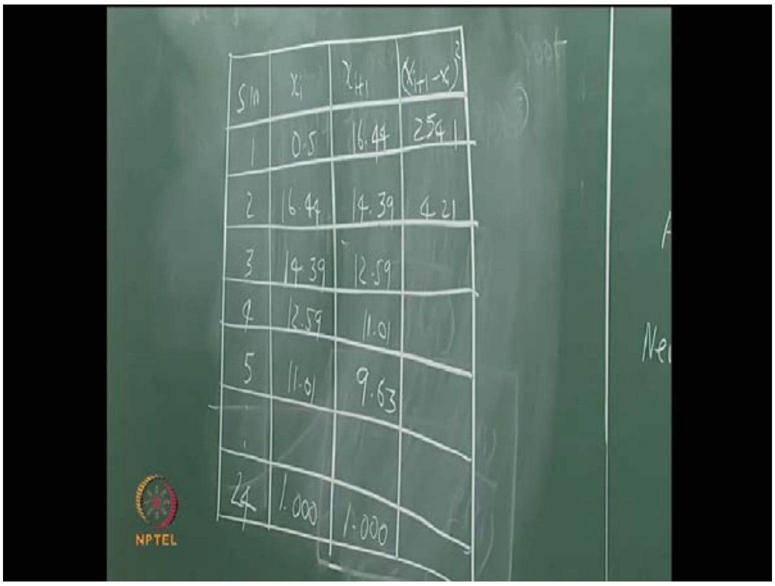
Why are we getting that? Why do you think… Where did we start? Then, it is ok, no? This slope… So, the algorithm becomes very slow when it is like this. It nicely converges if we have something like this; when it is moving very gently; if the initial guess is in that region, it takes forever for it to converge. Should I put x i plus 1 x i and all that and show it to… No need right? Now, we will… So, the solution is of course x 25 or x 30. What is the…

Student: x 24

x 24 is the… What is the value of x 24?

Student: 0.0000…

(Refer Slide Time: 23:30)




We will have to necessarily do it, because this stuff is going outside. Now, we will just say 1, 2, 3 and we will go to 26 in the interest of time. 0.5 leads to?

Student: 16.4

It is correct or… Please be reminded that, we have to fill it up. I have the freedom not to, but you guys have to fill it. Now, 20?

Student: 24

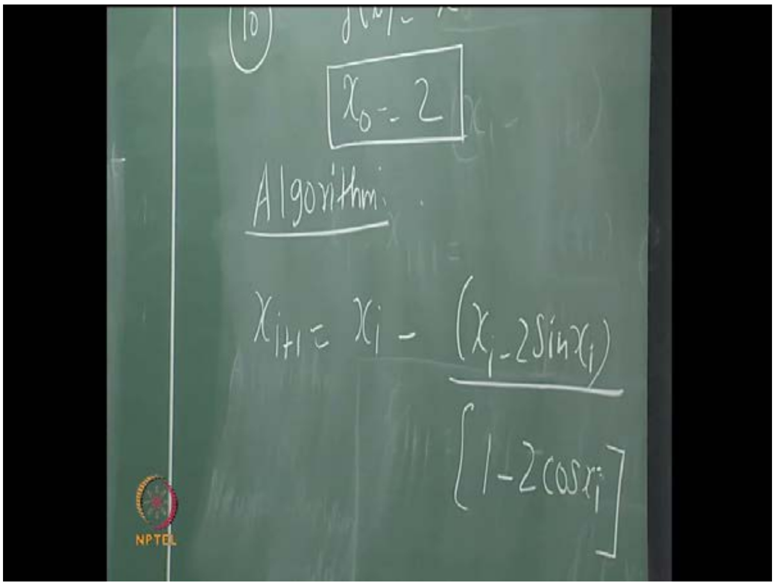
24. What is the value?

Student: 1.0000

So, it is slowly converging. So, all the big talk about Newton-Raphson method, because it is using f dash of x very powerful and all that; it is not; it is not really. It does not come, but the convergence is very slow.

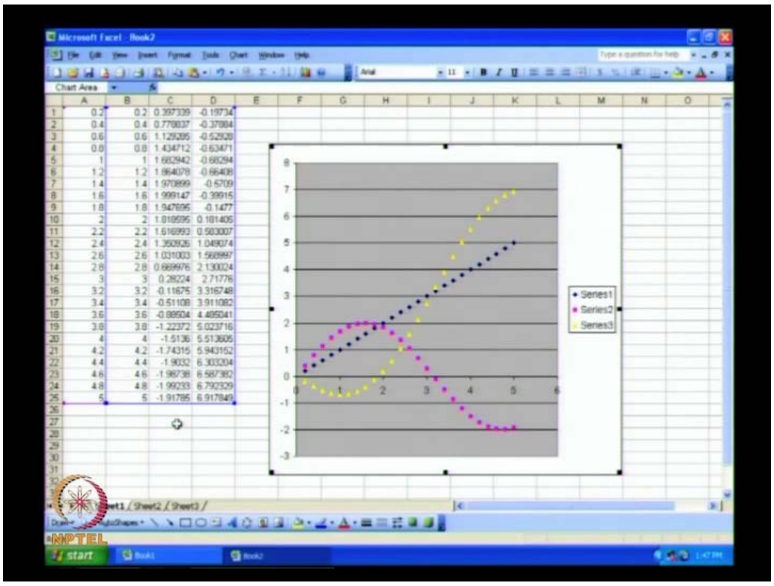
Now, let us workout a problem in which the Newton- Raphson method really works. Problem number 10 – use the Newton-Raphson method; use the Newton-Raphson method; use the Newton-Raphson method; use the Newton-Raphson method to determine the first positive root.

(Refer Slide Time: 26:32)




Use the Newton-Raphson method to determine the first positive root of the equation, f of x is equal to x minus 2 sin x; where x is in radians. f of x equal to x minus 2 sin x; where, x is in radians. Starting guess – x equal to 2 radians; starting guess – x is equal to 2 radians; x naught. So, what is the algorithm? x i plus 1 is equal to x i minus x i minus 2 sin x i divided by 1 minus 2 cos x i. So, I will plot this.

(Refer Slide Time: 27:51)




Enough? This – I have just shown the graphical variation of the two functions: f of x is equal to x and f of x is equal to 2 sin x. So, the solution is about 1.895. Correct?

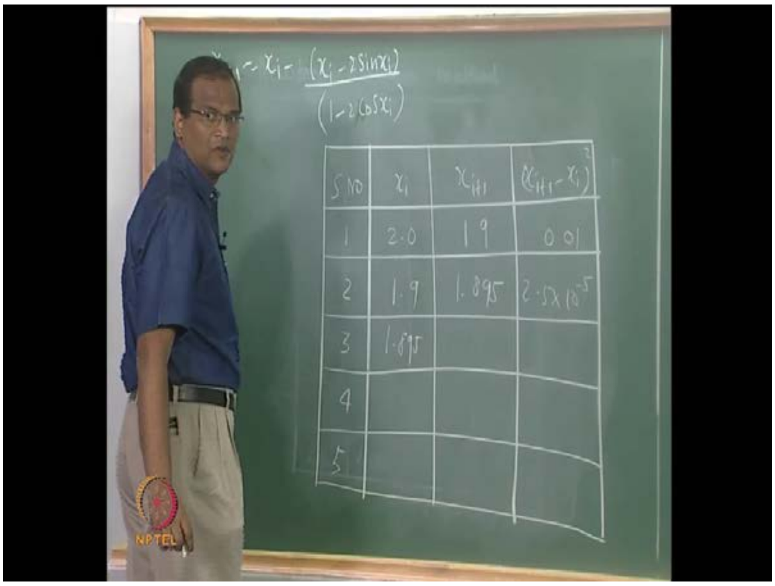
Student: Yes, sir.

Solution is x is equal to 1.895. So, which is the sinusoidal function here?

What about Jayaraman? Are you still there?

You can see exactly with the yellow one. Yellow one is the difference between the two. That is actually f of x; you can see where f of x becomes 0. I asked you to get the first positive root. It will also have negative roots. f of x is equal to x will keep on going on the other side also; it will also have negative roots. So, it is pertinent; it is imperative that, I will tell you whether I want the first positive root or whatever.

(Refer Slide Time: 31:39)



If you workout the algorithm and if you workout the steps; what is x i plus 1?

Student: 1.90

What is the…

Student: 2.5 into 10 to the power of minus 5

So fast; absolutely fast; there is no comparison with the successive substitution. Generally, if you are in the right region; your initial guess is in the right region; and, this function is not slowly changing; f of x is not slowly changing with x; is a good chance but, it will get… What do you say? You can get a very quick and fast convergence. Now, please retry this problem with an initial guess of 1 radian. So, 10 b… 10 a – initial guess is… If we go to 1.895, what do we get here? 895. So, 10 b will be x naught equal to 1 radian. I have also shown it on… I have shown it graphically. Now, when you work out the parts a and part b; part a – we started with 2 radians; it converged. Part b – we started with 1 radian; you are going to tell me whether it converged or not. When we look at the graph and then you allow your own understanding as to why it works for some case, one case, and it does not work for the other case.

Student: 10 steps

10 steps; you are to doing with the calci? When you start with 1, what is the next one?

Student: minus 7.47

Minus 7.? Somewhere it goes. And then?

Student: (( ))

Then, totally off. It comes back?

Student: It comes back.

It comes back. Now, graphically, explain what is the difference between starting at 1 n (Refer Slide Time: 35:20)? What happens when it is… What happens to f dash of x at 1?

Student: Negative (( ))

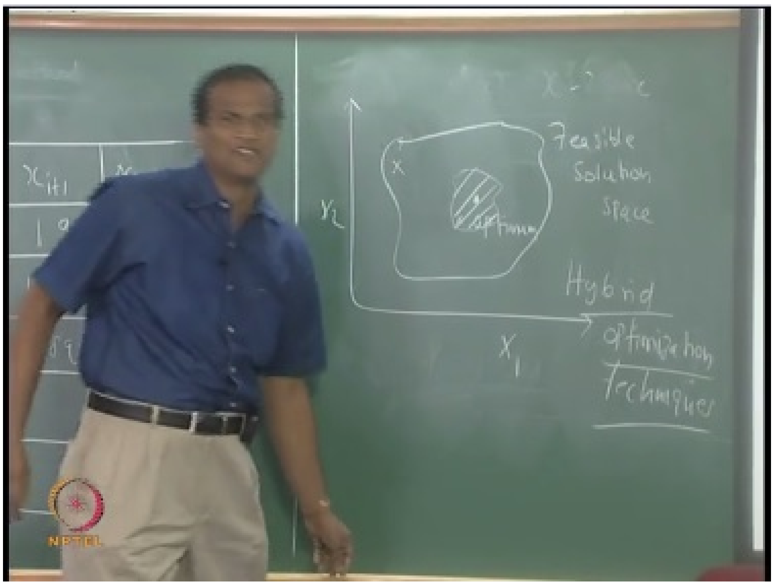
What is f dash of x at 1?

Student: Close to 0.

Close to 0. I told you when it is close to 0, it is a disaster for the algorithm. If it is close to 0; then, suddenly from 1, it swings to? Is it minus 7.5 or plus 7.5? It swings to minus 7.55. From there it swings to some other value. Eventually, it converges, but it takes a… So, that is why the initial guess is very important. So, people will say, then there is a some kind of cheating involved if the initial guess is so important; that is, if we know what the initial guess is, then you already know the answer or something. But, I think we cannot call it as cheating, because often times in engineering, the mathematician can argue anyway – I do not have any idea and all that; but, as an engineer, you will have a good idea of what is the approximate range in which you except the solution. Therefore, in optimization problems, if this involves a maximization or minimization of a particular function, leave the…

For the time being, forget that we are trying to seek roots to a particular equation. Suppose, we are trying to maximize or minimize; in that case, you are using a technique, which is similar to Newton-Raphson. You will use what is called the conjugate gradients or steepest ascent or steepest descent method; where, you will use the information on the derivative – first derivative. So, if it is going to give trouble like this; if you start at the wrong value; then, if it is going to oscillate left and right; and then, it is going to become difficult for you; then, one possibility is to have some other technique, which will localize; I mean which will ensure that, you give an initial guess in the proper region and you direct the solution such that you get accelerated convergence. How do you do that? So, you can use an ant colony optimization or genetic algorithm or simulated annealing or some other technique, where we will… We will see this in the later part of this course.

(Refer Slide Time: 37:26)




Suppose you have got a function, which is… f of x is a function of x 1 and x 2; you are trying to… So, this is what is called the feasible space; that means all combinations of x 1 and x 2 will satisfy all the constraints to the problem; that they will be the working solution to the problem. But, if you have an objective function, each objective will not be equally desirable; some will have a high value of y; some will have a low value of y depending upon the context, whether you want to maximize or minimize; some are more desirable compared to the others.

Now, if you start with the initial guess value; suppose this the optimum point; if you use a gradient method; suppose you start from here, it may work; but, if you start… Suppose you start from here, it may not work; but, if you start from here, it may work. So, what you generally do in these cases is you use a global optimization search technique like genetic algorithm or simulated annealing and then you identify a region, where you expect the solution to occur. Once you are in this region, then you fire the gradient method from anywhere; it will work. Now, you will ask me, sir, if I am able to converge to… If I know from this region, I am able to reduce to this region using genetic algorithm, why not I proceed and converge? Genetic algorithm… Usually these global optimization techniques are extremely slow. So, you initially use a global optimization technique to narrow down the region in which you expect a solution to occur; and then, once you hit upon that narrow region; then, you use a very powerful calculus base technique, which uses information on first derivative. Sometimes you can use the information on the second derivative – Hessian and all that. And then, you can narrow down your search. So, such techniques are called hybrid optimization techniques. They become the norm now. For multivariable optimization problems, non-linear optimization problems involving several variables, people are increasingly trying to use this.