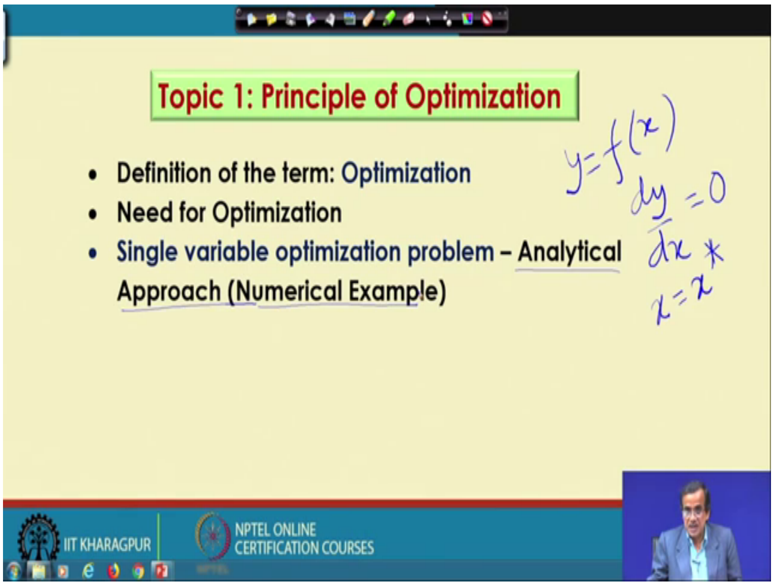
Traditional and Non-Traditional Optimization Tools
Prof. D. K. Pratihar
Department of Mechanical Engineering
Indian Institute of Technology, Kharagpur

Lecture – 43
Summary – 1

Now, I am going to summarise the topics which I have taught in this course. Now there are 14 topics which have been covered in this particular course. So, I am just going to concentrate one after another.

(Refer Slide Time: 00:39)



The, first topic that is the principle of optimization.

Now, here we started with the very definition of the term optimization like what do you mean by optimization? Now optimization is the method of finding the best solution out of the different feasible solutions available. Now let me take one very simple example supposing that say using the traditional the principle of design, say I am just going to design and develop, say one mechanical item, say one gear box.

Now, if I principal the traditional method of this particular the design there is a possibility that we will come up with one design, which has got no practical meaning. Now if I want to get one competitive design which will be able to gain much popularity in the market.

So, we will have to link with the principle of optimization along with your the traditional principle of machine design. And that is why the principle of optimization has become so, much popular. Now this is the method of optimization has to be adopted, if you want to be in the dynamic and competitive market.

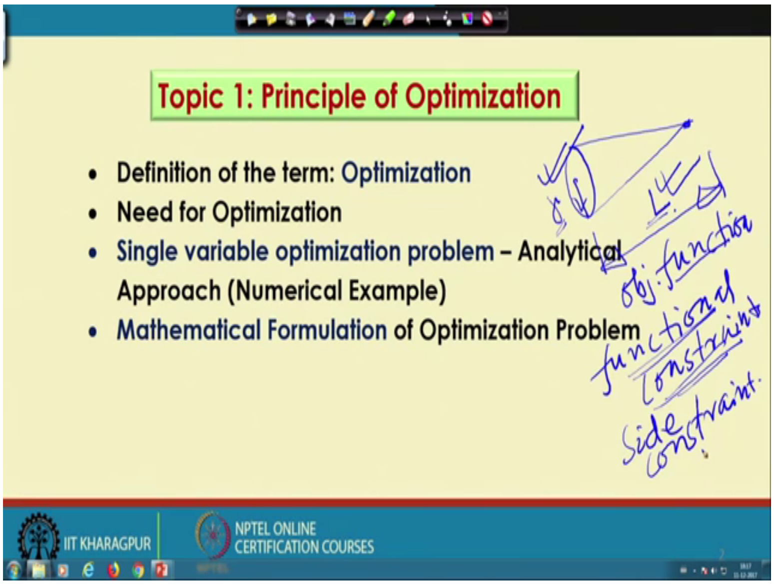
Now, the need for so, this particular the optimization is actually if you want to been competition we will have to go for cost effective and efficient design and the principle of optimization is going to help us to ensure that cost effective and efficient design and that is why so, we should use the principle of optimization. Now, we started with the single variable optimization problem and this is a the method is a very well-known one that is we use the analytical approach, that is the method based on the calculus like supposing that I have got a function y is a function of only 1 variable.

Now, we try to find out the rate of change of this particular function that is d y d x and put equals to 0; that means, we try to find out a value of x for which the rate of change of this particular function will become equal to 0.

So, there is no change at that particular value of x and we try to find out. So, that value of x which is nothing, but x equals to x star and we just gone investigating further, like whether there will be a maximum point or a minimum point or there will be an inflection point.

Now, this particular this single variable optimization problem I have discussed and will I took the help of one numerical example, and with the help of this numerical example we discuss like how to get and when to get the maximum point or the minimum point or the settle point or the inflection point.

(Refer Slide Time: 03:58)



Now, this single variable optimization problem is a very simple one and then we took the help of one multivariable optimization problem. And we just try to model so, that particular problem in a as an optimization problem. So, what we did is we took the help of a very practical example.

Now, that example was just like one pointer which we generally use while delivering this particular the lecture. Now this pointer is actually having so, on the gripping end the gripping end the radius is say r and the pointing end the radius is 0 and supposing that the length of this particular the this wooden pointer is say L. So, I can find out the volume of this particular pointer and I can calculate what should be the mass or the weight.

Now, our aim was to get one design of this particular pointer. So, that it becomes light and weight at the same time the very purpose of using this particular pointer will be sub; that means, we will be able to point whatever we are going to point during the presentation. And at the same time there should not be too much reflection and there should not be any such mechanical breakage.

Now, what we did. So, we try to find out like what should be the objective function. So, we try to define the objective function and we try to find out what should be the functional constraints.

So, the this functional constraints we try to define for example, like for this type of problem the functional cool constraint should be the develop stress, the develop stress should be less than the permissible value of the stress for this particular the material and that is nothing, but the functional constraint and of course, it has got a few of the design variables.

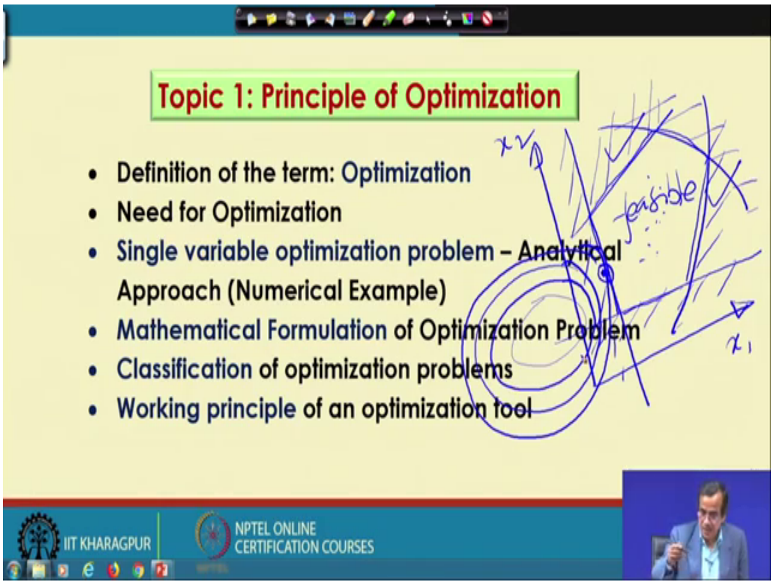
Now, this particular design variables are nothing, but so, the radius of this particular gripping end that is r and the length of this particular the pointer that is L. So, this r and L are the design variables they are having some ranges and that is nothing, but those are known as the side constraints. So, if you want to explain or if you one 2 describe 1 optimization problem in the mathematical form.

So, what will have to do is will have to first identify what are the design variables for example, for this type the problem the design variables are r and this L and we will have to find out the functional constraint that is what is our exact aim the main aim the main aim is to design one pointer which is having less weight. So, our aim is to minimise the mass or the weight of this particular the pointer, subject to the condition that the develop stress should not exceed the permissible limit and that is nothing, but the functional constraint.

Now, if you want to solve one optimization problem using any technique say either traditional or non-traditional, the first thing will have to do is will have to develop the mathematical formulation of this particular the problem; that means, we will have to identify the design variable what should be the objective function what should be the functional constraint what should be the side constraint or the variable bounds.

So, all such things are to be defined first and that is actually what you mean by the mathematical formulation of the optimization problem. Now once I have got so, that particular the mathematical formulation.

(Refer Slide Time: 07:55)



So, now we are in a position to use some optimization algorithm to solve it; that means, to find out the optimal solution.

Now, if you see this particular the literature a huge literature is available on this particular the optimization problem. Now these optimization problems have been classified in a number of ways. For example, say depending on the nature of the objective function and the functional constraint we can classify the optimization problem as the linear, optimization problem and non-linear optimization problem.

And by linear optimization problem you mean that both the objective function as well as the functional constraint should be the linear function of the design variable. On the other hand the non-linear optimization problem is such where either the objective function or any one of the functional constraint becomes non-linear function of the design variable.

So, this particular classification is the linear optimization problem and non-linear optimization problem. Now for solving the linear optimization problem we solve a particular technique and for solving the non-linear optimization problem we will have to use some other type of the techniques. The next classification could be depending on the nature of the design variables.

Now, if you see the design variables the design variables could be integer it could be real values and there could be sometimes a combination of some integer variable and some a real variable and those are known as the mixed integer the programming problem. So, depending on the nature of this particular your your the design variables the optimization problems are classified into 3 groups like one is called the linear programming problems, another is called sorry one is called the integer programming problem, another is called the real valued problem another is called the mixed integer the programming problem.

Now, if you see the nature of these particular design variables. So, so the optimization problem can also be classified as static optimization problem and dynamic optimization problem. And another classification of the optimization problem could be the constrained optimization problem and unconstrained optimization problem.

So, by constrained optimization problem we mean that at least one functional constrain is to be there and by unconstrained optimization problem we mean the objective function should be there, but there is no such functional constraint. So, this is the way actually we can classify actually the optimization problem.

Now, we classify the optimization problem and to solve each of this particular problem. So, I will have to use a particular type of optimization algorithm, and that is why you see the literature of traditional tools for optimization there exist a huge a large number of optimization tools. Now these optimization tools can be broadly classified into 2 groups one is called the gradient based method and another is called the direct search method.

Now, I am just going to briefly discuss after sometime the gradient based method and the direct search method. Now before I go for that. So, let me spend some time on this working principle of the optimization tool. Now here actually what we do is supposing that I have got one constrained optimization problem.

Now, if I plot the constrained optimization problem it is objective function and the functional constraint and the variable bounce. So, there is a possibility that we will be getting some feasible zone and let me take a very simple example here. Now supposing that I have got one problem so, this is so, if I just plot say it is a function of say 2 variables x 1 and x 2 and if I just plot the variable bounce might be this is one variable bound this is another variable bound.

So, this side is the infeasible zone and this side is the infeasible zone. So, considering this particular variable bound. So, this side is a feasible considering this particular feasible this particular variable bound supposing that the upper side is feasible. So, this is nothing, but the feasible zone. Now what we do is now we take the help of some functional constraint supposing that I have got a functional constraint like this. So, this side is infeasible and this side is feasible.

Similarly, I have got to one functional constraint like this this side is infeasible. So, this side is feasible I have got another functional constraint like this. So, this side is say infeasible and this side is feasible. Now so, to obtain this particular optimal solution so, I have to concentrate on this particular the feasible zone only.

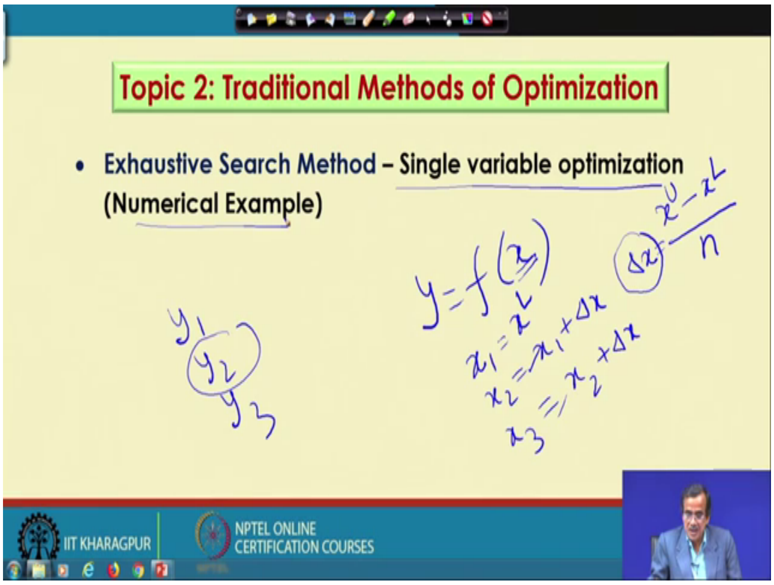
Now, here so, this is nothing, but the feasible zone now on this feasible zone actually there will be a free points a few free points and there will be a few bound points the free points are those points, which are lie inside the feasible zone and the bound points are those point which are lie on the boundary of this particular the feasible zone.

Now, what we do is we try to find out we try to draw the control plot of this particular the objective function. Now supposing that I have got one mathematical expression for this particular the objective function and if I find out the control plot supposing that say for a particular value of this functional constraint say I am getting so, this type of control plot.

Now, for another value supposing that I am getting so, this type of control plot, another value I am getting so, this type of control plot, for another value of the objective function supposing that I am getting this type of control plot, the moment this control plot touches the feasible zone. So, that is going to indicate actually the optimal solution corresponding to the constrained optimization problem.

So, this is the way actually one optimization toolbox just to find out the optimal solution. So, more or less so, this particular method is followed by most of the optimization algorithm. So, this is actually what we mean by the working principle of one optimization algorithm.

(Refer Slide Time: 15:01)



Now, once I have got these the next we concentrated on some traditional methods of optimization. Now here if you see the literature we have got a huge literature available and as I told, that these traditional methods are broadly classified into 2 groups one is called the gradient based method another is called the direct search method.

Now, in gradient base method we try to take the help of the gradient information of the objective function. And the search direction is decided by the gradient direction of this particular the of the objective function. On the other hand for this particular the direct search method we do not take the help of the gradient information of the objective function instead we consider, the numerical value of the objective function.

So, this particular numerical value of the objective function is going to control the search direction of this particular the algorithm. Now here in this course we discuss the working principle of a few traditional tools for optimization. Now we started with one exhaustive search method and that we did for one single variable problem. So, this exhaustive search method is used for the single variable problem and if you remember we took the help of a 3 points just to find out the search direction like whether the solutions will be accepted or not.

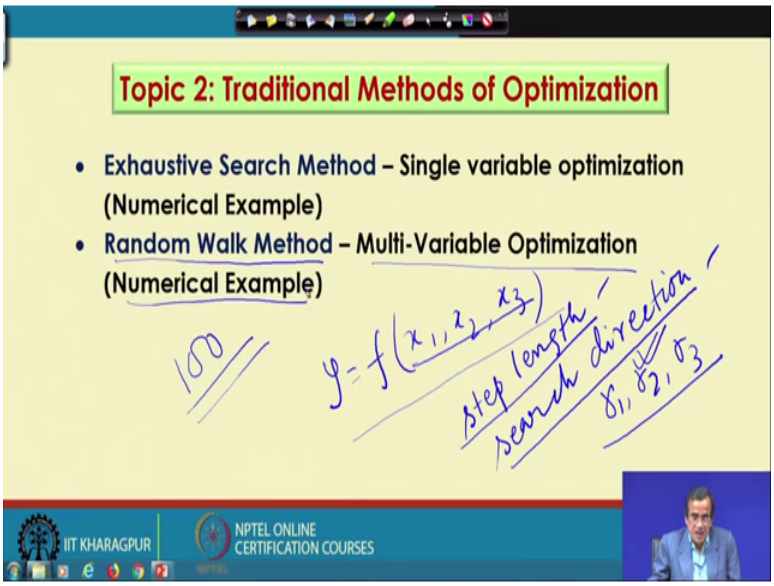
So, supposing that I have got a function. So, y is a function of only 1 variable. So, what we do is so, we try to design the 3 values for this particular x like your x 1 x 2 and x 3 depending on the lower end the higher limit of this particular the design variable that is x we consider. So, initially we consider this x 1 is equals to x lower value and we try to find out the intermediate values like x 2 and x 3 depending on some small range already defined that is nothing, but your delta x.

Now, delta x is nothing, but your the x the upper limit minus x the lower limit divided by the equal number of division that is denoted by small n. And once we have known this particular small x that is the delta x i can find out x 2 and x 2 is nothing, but x 1 plus delta x and x 3 is nothing, but x 2 plus delta x. Then what we do is we substitute the values for this x 1 x 2 and x 3 and we try to find out the numerical values of these particular objective function that is nothing, but is your say y 1 y 2 and y 3 and we compare.

Now, if it is a maximization problem. So, we try to find out whether this particular solution is line within that particular the range or not. So, by using the either the greater than or the less than sign we can control; and we can solve either the maximization problem or the minimization problem.

Now, this method the exhaustive search method we discuss with the help of one numerical example. And this is a very simple method and this method is having some limitations and that is why we actually concentrated on some other more complex method and those are also suitable for the multivariable problem.

(Refer Slide Time: 18:45)



For example say we try to concentrate on this particular the random walk method.

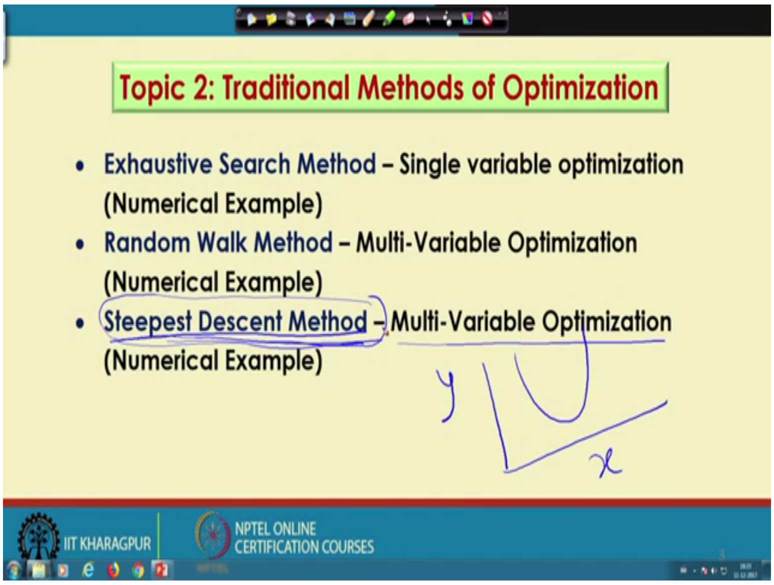
Now, this random walk method can tackle the multivariable optimization problem. For example, say the problem could be a function of. So, many variables like x 1 x 2 say x 3. Now here there are 3 design variables and this type of function. So, we can actually optimise with the help of this random walk method. Now for any optimization tool there are 2 things, which have to be decided one is actually the step length. So, what should be the step length in one iteration and what should be the search direction in a particular the iteration.

Now, this particular the search direction and the step length are decided in a particular fashion in this particular the random walk method the step length is actually is a user defined. So, we try to define some fixed value as a step length and the search direction is decided at random using the random number generator.

Now, if this particular algorithm is working for a large number of iteration say 100 iteration. Now at the at the beginning of each iteration. So, we will have to decide what should be the search direction and to decide this particular search direction we take the help of some random numbers. Now there are 3 design variable. So, we consider 3 random numbers like r 1 r 2 and r 3 lying in the range of say minus 1 to plus 1 or say 0 to 1 and we try to find out like what should be the random search direction.

And the working principle of these particular the random walk method we discussed with the help of one numerical example. Now this random walk method it has got some merits and demerits it has got a merit in the sense that it does not required the gradient information of this particular the objective function. So, it can a handle the problem having discontinuous objective function and it is got some demerits in the sense sometimes it may take unnecessary unnecessarily a very large number of iteration. So, this is actually one of the drawbacks of this particular the random walk method.

(Refer Slide Time: 21:20)



Now, next we try to concentrate in this particular course on a method which is known as the steepest descent algorithm. Now here so, this particular algorithm can tackle the multivariable optimization problem and here the search direction is decided by the gradient of this particular the objective function, now here the search direction is nothing, but opposite to the gradient direction.

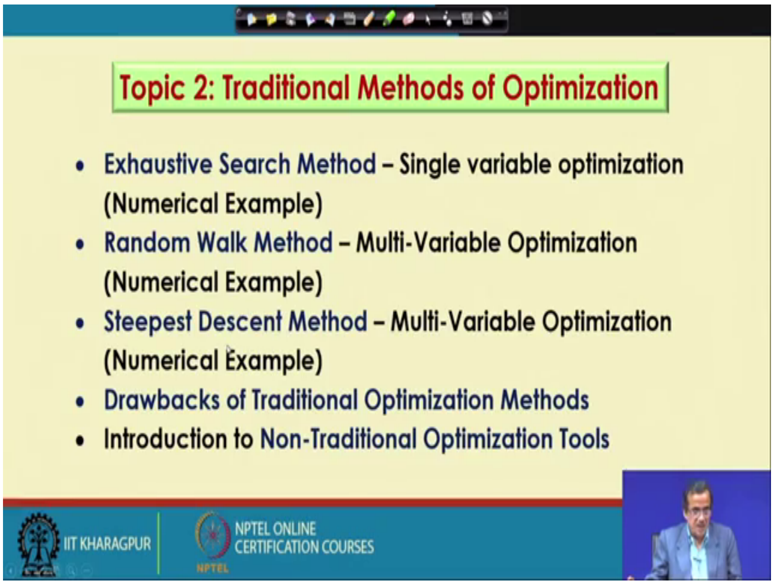
Now, we consider the gradient direction because the rate of change of the function is the maximum along this particular gradient direction. So, we try to consider the we try to move in this particular algorithm in a direction opposite to the gradient and on principal it can solve the minimization problem. Now if you just compare the computational complexity of this particular algorithm. So, this particular algorithm is the first test of all the traditional methods of optimization and this particular method is having some merits and it is also got some demerits.

Now, it has got the merit in the sense that it is very fast particularly for the unimodel function for example, say I have got a function like this say y is a function of x and I have got a very regular unimodel function. So, for this type of function no algorithm can actually a compete with the steepest descent algorithm, but supposing that I have got a complicated function of say many undulations and there are many such modes and it is a multi-model function.

So, there so, this type of steepest descent algorithm may fail to give the globally optimal solution now as gradient is a local property. So, there is a possibility that this particular algorithm is going to get start at the locally optimal solution. And the of course, there is no guarantee that it is going to head it is going to get it is going to hit that particular the globally optimal solution.

So, these are actually the merits and demerits of this particular the steepest descent algorithm. Now, after discussing the working principle of these traditional methods of the optimization so, I just try to concentrate on actually the drawbacks of the traditional optimization method.

(Refer Slide Time: 23:46)



Now, these traditional optimization method has got a few drawbacks for example, say we start with one initial solution selected at random and the quality of the solution depends on the way we choose the initial solution.

Now, supposing that say unfortunately we are selecting the initial solution in the local basin and if you select the initial solution in the local basin. So, there is a possibility that it is going to hit that particular the locally optimal solution and we may not be able to reach that particular the globally optimal solution these thing I have discussed in much more details.

Now, there are a few other drawbacks for example, say supposing that I am going to use one say gradient based method the steepest descent method, but that cannot be used for a objective function which is having some sort of discontinuity. So, if there is some sort of discontinuity you cannot determine the derivative and we cannot use this type of the steepest descent algorithm.

Next is your this particular traditional method as it starts with only one initial solution it is not suitable for parallel computing. Now that means, the traditional tools cannot be implemented in parallel computing and by parallel computing actually we can minimise the effects it is few time for this particular the computation.

Now, these are all drawbacks and there are a few other drawbacks for example, say if there is any such discontinuity in the objective function we cannot handle effectively or sometimes we face a problem optimization problem, for some of the variables are integer summer some other variables are real and those types of problems are known as the mixed integer programming problem.

And to solve this particular mixed integer programming problem I will have to use a very special type of algorithm; that means, so, these traditional methods may not be robust to solve a variety of problem and a particular tool could be suitable to solve only a special type of problem. And these are all drawbacks of this particular the traditional methods of optimization and that is why the concept of the non-traditional tools actually came into the picture.

Now, if you see the non-traditional tools for optimization once again we have got actually a large number of non-traditional tools for optimization. Now before I name a few you now let me tell you once again the philosophy behind going for so, this type of non-traditional tools for optimization.

Now, as I told the traditional tools may not be robust. So, a particular tool can solve only a special type of problem and that is why we have got a large number of traditional methods for optimization. And just to overcome that particular problem. So, we try to design and develop some robust optimization tool and that is why the concept of non-traditional optimization tools came.

Now, if you see this particular the non-traditional optimization tool we have got in fact, a large number of non-traditional tools for optimization. Now if you see if a name few for example, say we have got like the non-traditional optimization tools like genetic algorithm, then we have got genetic programming, then evolution strategies, evolutionary programming, simulated annealing, particle swarm optimization, and colony optimization, be colony optimization and so on there are a large number of non-traditional tools for optimization.

Now before I start the discussing little bit or before I summarise the principle of a very famous non-traditional tool for optimization known genetic algorithm. Now let me let me tell you once again the reason behind going for this particular the genetic algorithm.

Now, we have already seen the working principle of these particular random walk method and your the steepest descent algorithm. Now these random walk method here there is no such fixed search direction on the other hand the steepest descent algorithm is having a very well defined search direction and both this algorithms are having it is own merits and demerits.

Now, actually in non-traditional optimization tool like genetic algorithm so, we tried to actually get the merits of both this 2 algorithms and we try to overcome and we try to delete their demerits, and that is why if if in one side we have got the random walk method on other side we have got the steepest descent algorithm and genetic algorithm is actually in between.

Now, let us try to summarize the working principle of genetic algorithm in in details we have already discussed, but let me try to summarise that.

Thank you.