Introduction to Soft Computing
Prof. Debasis Samanta
Department of Computer Science & Engineering
Indian Institute of Technology, Kharagpur

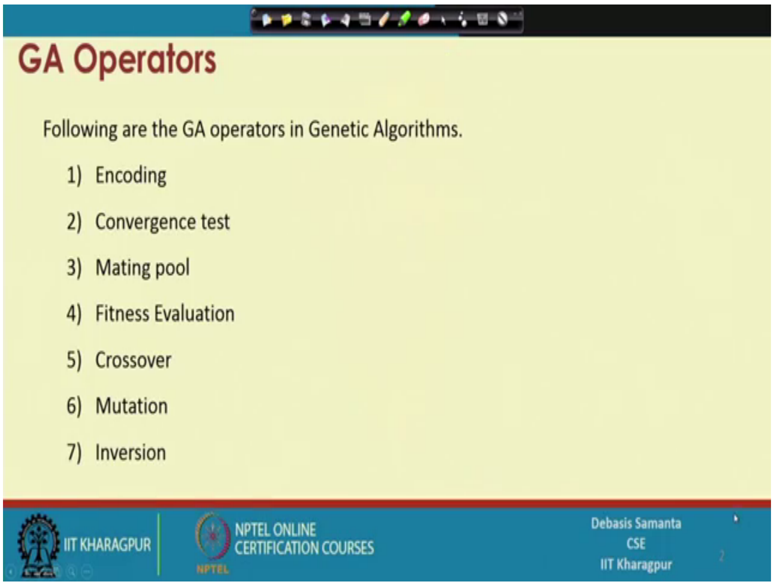
Lecture - 16
GA Operator: Encoding schemes

We have learned the basic architecture of genetic algorithm. And in the basic architecture, there are many operations are involved, one operation is creating the initial population and then convergence testing and then selection operation and finally, the reproduction operation reproduction include cross over mutation and inversion.

Now, so far, the first operation namely, how to create initial population? You basically require to learn about the encoding scheme. Encoding scheme implies how a problem can be encoded. That the GA architecture can use it and then follow it is operation to produce the output result.

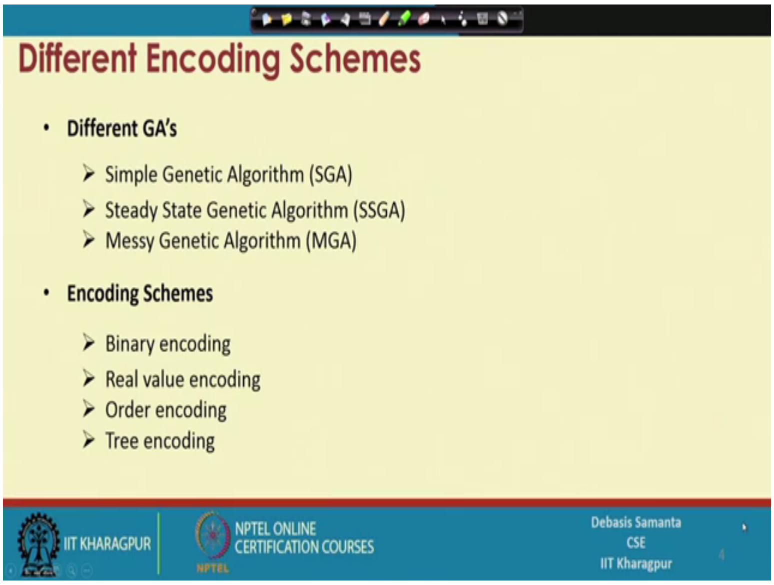
Today we will discuss about, in this direction the encoding operation.

(Refer Slide Time: 01:11)



Now, different operations again I repeat it the encoding the convergence testing then creating mating pool fitness evaluation it is basically part of the selection and then. For the reproduction is called the operation or crossover mutation and inversion.

(Refer Slide Time: 01:33)

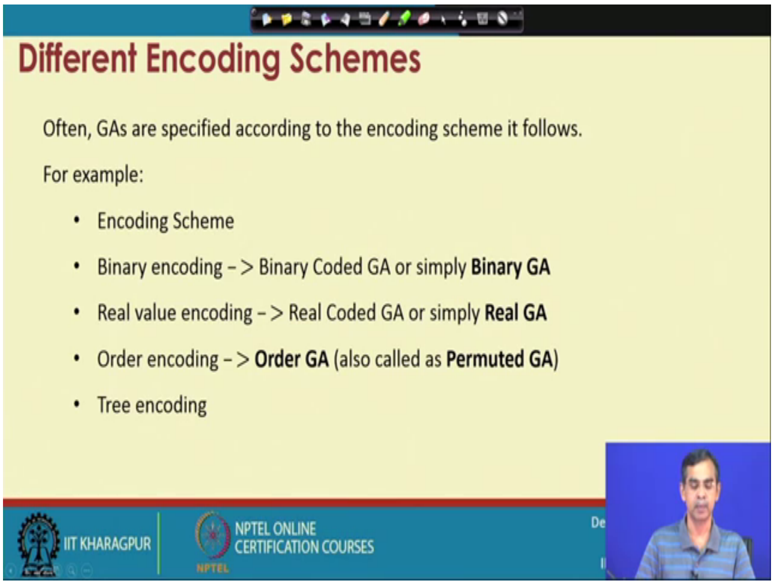


Today, we will discuss about encoding and fine different genetic algorithm we have discussed about a brief about the different genetic algorithm like say simple genetic algorithm and steady state genetic algorithm and also messy genetic algorithm. In fact, they are different from the point of view how the different operations can be carried out. And then also there is one important difference between among this is basically what are the different encoding scheme that they can follow.

For example, for the simple genetic algorithm and steady state genetic algorithms are concerned. They follow the constant length encoding, where is the messy genetic algorithm constant the variable length encoding; that means, anyway let us see exactly what are the different encoding schemes are there then we will be able to understand about the difference between messy genetic algorithm then others.

Anyway, for the different encoding schemes are concerned we have listed here few important encoding scheme which are most popular here. The first is binary encoding and then real value encoding, order encoding and then tree encoding. All these encoding scheme will be covered 1 by 1.

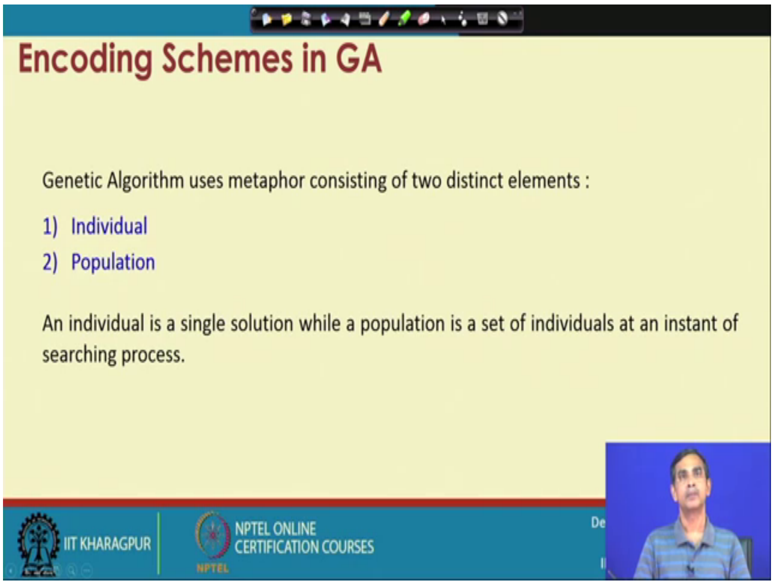
(Refer Slide Time: 02:43)



First, we will start about the binary encoding and before going to this if a particular genetic algorithm follow a particular encoding scheme, then according to the encoding scheme that this algorithm follows it is termed as that GA. For example, if your genetic algorithm follows the binary encoding to create the population then it is called the binary GA.

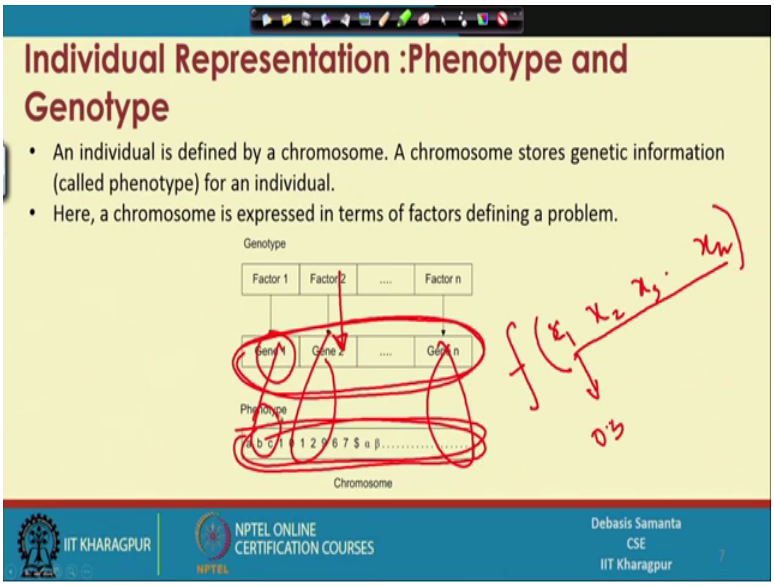
The real value encoding is another approach if it is followed then the GA is called real GA. Order encoding if it is followed in genetic algorithm then it is called order GA. Sometimes order GA is also called permitted GA. And then if it is follow tree encoding mechanism then it is called the tree encoded GA.

(Refer Slide Time: 03:27)



Now, encoding scheme, for the encoding scheme in genetic algorithm is concerned. Basically, there are 2 different things involved in genetic algorithm architecture one is individual and another is population. Individual basically related to a solution a possible solution or a perspective solution and a set of perspective solution, if it is included then it is called a population. Population is basically is a set of individuals and individual is a particular solution and population usually at any instant. What are the different at any instant of the searching process searching for the best solution. In fact, at any instant of the searching process the set of solutions are basically the population of that instant.

(Refer Slide Time: 04:23)



Now, let us first discuss about before going to the encoding scheme the 2 concepts in order to encode a particular solution it is called the phenotype and the genotype. Now, as I told you earlier, the genetic algorithm follows the concept of genetics. And in the concept of genetics, the chromosomes play an important role. A chromosome is basically is a collection of genes and a particular genes gives a particular DNA for an individual. Every individual has it is own DNA code that is the gene combination.

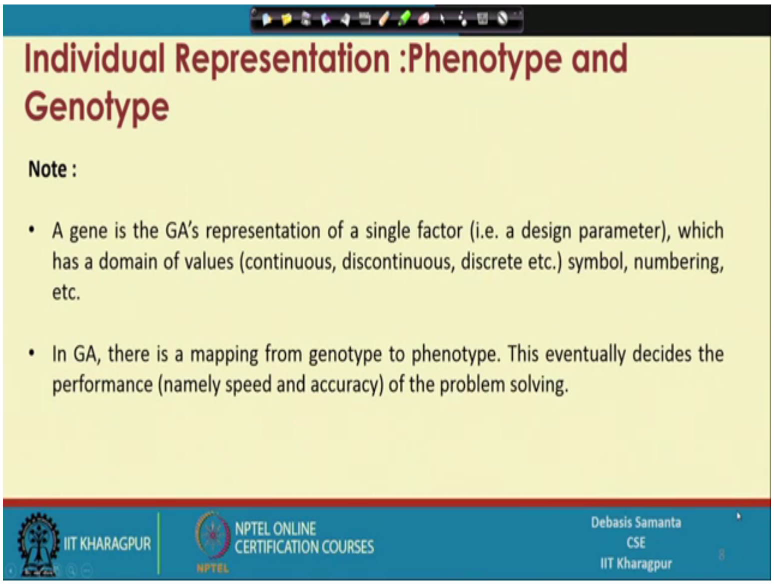
Now, the basic structure of all these chromosomes is called the genotype. Basic structure means, as you know genotypes is basically the different from one problem to another; that means, for your problem you have to decide exactly the genetics genotype.

What is the genotype? I can give an idea. Suppose you have to your objective function consist of a different variable like x 1 x 2 x 3 and x n so; that means, we have to find these are the basically input parameters and we have to optimise one function which is f for given values of these parameters. Now, here x 1 x 2 x 3 and are xn are the n number of design parameters. It is also called the factors for example, factor 1 is x 1 factor 2 is x 2 and factor n is xn.

Now, if x 1 has a typical value say 0.5 then it is called the gene value for the parameter x 1. Similarly, for every factor there is a gene value that mean the value of this parameter at that instant. These combined is called the gene values for the different factors or the different design parameters in the problem. At any instant, the values of all the factors constitute what is called the phenotype; that means, gene 1 this part gene 2 is this part and gene n this part is constitute the entire what is called the things. It is called the phenotype.

Genotype and phenotype is involved in this way. And so, encoding is basically how a factor can be encoded to give some value? That means, gene value. We will see the different method for the encoding scheme.

(Refer Slide Time: 07:10)

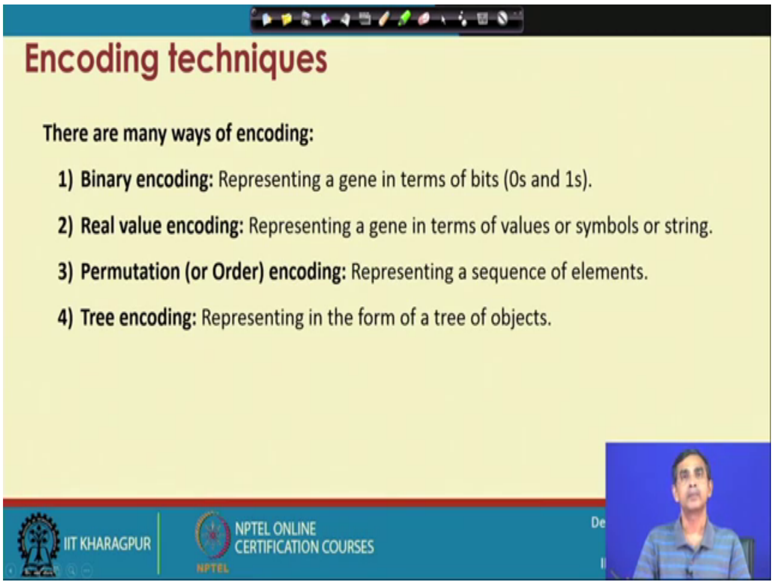


Fine, before going to understand about the binary encoding. We can understand or we can say that a gene is the GA's representation of a unit that is a single factor that is a design parameter.

The design parameter may have different values, they can be defined in the discrete domain, they can be of continuous or or some symbolic values or number etcetera. That mean gene value can be anything and that is according to the requirement of the problem that you are going to solve.

In GA, In fact, there is a need to mapping from genotype to phenotype, that mean for each factors or design parameters that is there in the phenotype. How to map to some values of it? That is the genotype this eventually decides the performance of the algorithm; that means, the efficiency and accuracy of your problem solving. If you design if you design the encoding scheme effectively and properly, then you can get the result at the earliest and also the correct result.

(Refer Slide Time: 08:35)



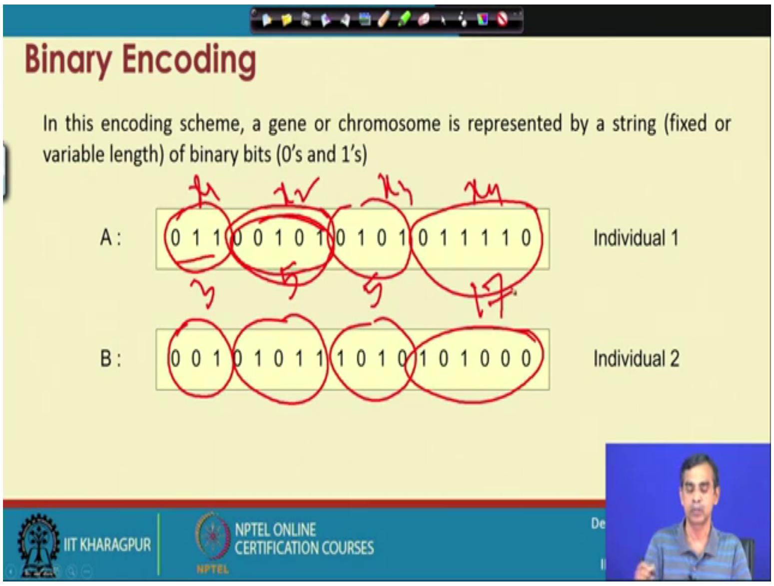
Now, we will come to the different encoding things techniques I have already told 4 different techniques. The binary encoding, the real value encoding and then order encoding or permuted encoding and tree encoding. Now, binary encoding as the name implies it follows the gene value in terms of 0s and1; that means, the binary representation. You know anything whether it is a number it is a real number, it is a symbol or it is a some other representation can be represented using binary encoding scheme.

It basically gives a gene value in terms of only 2 bits called 0s and 1s. On the other hand, real value encoding scheme is very convenient to understand from a programmer point of view. It is basically state to a exactly what is the value for a particular parameters is to be stored there. That is why real value encoding. It is if you want to I mean say factor is a name. The name can be can consist of 20 characters alpha numeric whatever it is that you can store like that.

Now, order GA is a special, special case of encoding, it requires not to solve all program, but there are some programs where the sequence of elements matters. If it is a problem like this then we can encode that kind of problem using the order encoding scheme.

And tree encoding scheme is a special form of the encoding mechanism where, it is stored in the form of a tree. T is a concept it is a structure with which we can solve we can represent many what is called the problem and we will learn about tree encoding with an example that understand it.

(Refer Slide Time: 10:32)



Now, I will first start with binary encoding. As I told you in this encoding scheme a gene the collection of genes basically constitute a chromosome. A gene is represented by a string of 0's and 1 is a binary string. Now if we follow the fixed length of the chromosome, then they are basically sca or ssj.

On the other hand, if it is a variable length chromosome then it is called the messy GA. Now, first let us see what is the length of the chromosome? Now, in this example if we see the number of 0's and 1's altogether is 18th. The length of the chromosome is 18^{th}.

Now, here for example, another this is the length of the chromosome. Length means this is the length of the chromosome and this is the solution 1; that means, 1 individual and this is the individual 2. 2 chromosome are represented here we have name this chromosome as a and b or individual 1 and individual 2.

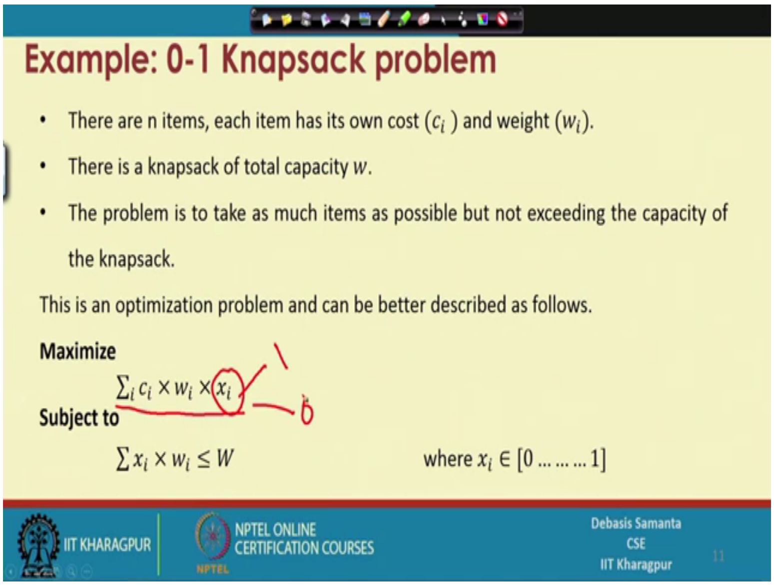
Now, here if this chromosome can be like this we can think about it. Say, these are the first 3 bits for 1 factor and these are the 5 bits for another factors and then this is the 4 bits for another factor and then finally, these are the 9 bits to another factor. We can say this is the parameter x1 x2 x3 and x4. Likewise, this is another gene for another factors representing these value.

Here, we can see the 4 factors or 4 design parameters have been encoded with their binary value. Now, if we say the binary value this represents the 4. For the x1 is concerned and these represents this is basically 3 right and this is 5 and so on. Basically, the binary equivalent a decimal equivalent of this binary representation it basically. It is a 3 5 and this is also 5 and this is basically 2 4 6 8 8 6 14 to 16 and 17. These basically represent 17 and so on.

These are different value that has been represented by this combinations. Whole the things constitute one what is called the chromosomes. This is an essential thing in the binary encoding. Basically, each design parameter can be represented by means of some binary string. And as you know, binary encoding is powerful to represent any value whether it is a integer. Integer can be coded using decimal to binary conversion formula. Real value also can be because there is also formulation by which any real number can be converted in the binary one and then any symbolic representation any characters also can be converted, a string of character also can be converted.

Anything can be converted using the binary representation of the gene. The binary representations follows in a binary encoding and this is the idea, I just want to give one example so that you can understand about the application of binary encoding to solve some optimization problem.

(Refer Slide Time: 14:04)



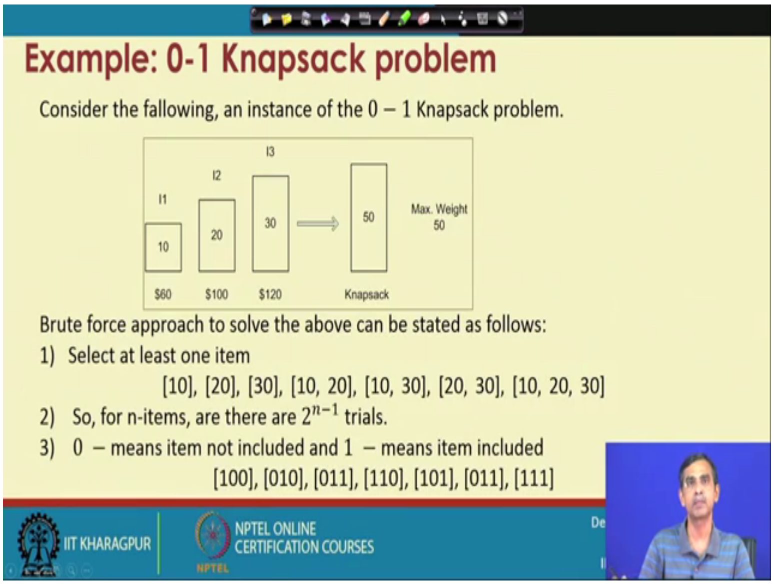
I will discuss one optimization problem it is called Knapsack problem and more precisely it is called 0 1 Knapsack problem why the name 0 1 we will understand about it.

Now, in this problem basically there are n items given to you and each items is specified by a by it is cost and then weight. N items and then n items has their cost and weights known to you. And there is a Knapsack and then total capacity of Knapsack also given to you.

Now, here the problem is to take as much as items as possible, but not exceeding the capacity of the Knapsack we have collected. As an example, suppose a thief enter into a showroom with a knapsack in his and knapsack his own capacity and the different cost of the different items are known there, and the thief has to collect the maximum elements from there, that he can put all the elements into his knapsack. And so that he can also I mean gain maximum I mean. For the thief is concerned he can take the maximum amount from there. Basically, within the limited weight he has to collect the items and each items having their own cost. And so that he can maximise the cost from there.

Now, so regarding these things, this problem can be expressed in the optimization problem statement which is here. Basically, the idea is that. This is the objective function maximize, maximize is basically here. See I if the ith item is selected and w I/ is the cost and xi is basically how many of that element is selected or it is selected yes or no. Xi is basically if it is selected then 1 if it is not selected then 0 and ci and this one. This is basically the objective function in this regard. And then constant is that constant is. Basically, xi into wi should be less than equals to w. That is the constant, because if you select one item xi then weight of these things and summation of all the selected items should be within the maximum limit of the limit of the Knapsack that is a w. These are the problem and the statement of the optimization problem and having this optimization problem, let us see how we can solve it.

(Refer Slide Time: 16:54)



First, we will see that how it can be solved using new approach and then finally, how the same thing can be solved using GA approach or basically we will decide about encoding scheme. Now, here this pictorially once simple instance where the different items for example, item 1, item 2, item 3, 3 items only are there for this item 1 the weight is 10 20 and 30 units and the cost of the item 1 is 60-dollar 100 dollar and 220 dollar and this is the this this is the Knapsack this is the Knapsack the capacity of the Knapsack is 50. We have to take some items selectively from there where. That we can have the maximum this one.

Now, as an example if we select 10 and 20 then; obviously, within this capacity, but the solution gives the cost is 100 and 60. Or say, 20 and 30 it is possible and then it will give the cost is 100 and 120. This is like this the different solutions.

Now, what are the different solutions are possible it is listed here. We can select single item at a time and then 2 item at a time and then 3 see item at a time; however, this is not a feasible solution. Because, if we can include the item 1, item 2 and item 3 the cost may be 10 20 30 but it will exceeds the maximum quota. Because, the maximum weight is 50. It is not the feasible solution.

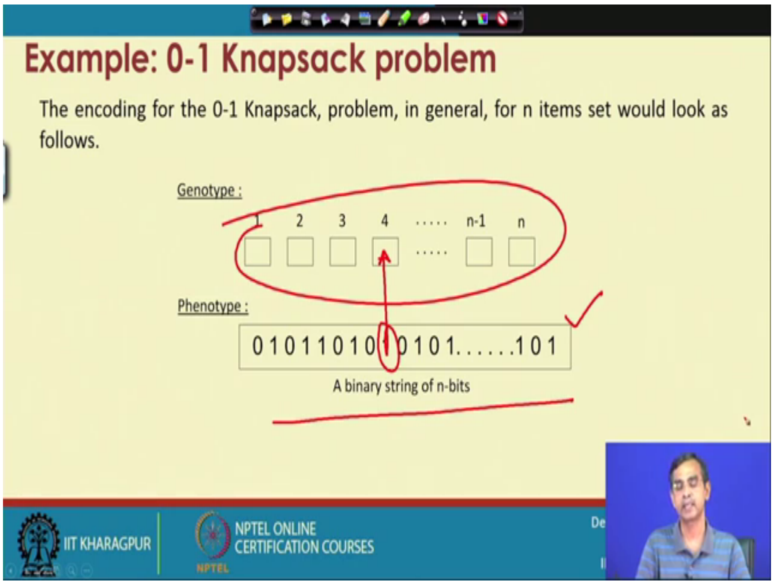
Now, what I can understand is that is basically the problem, if we solve in a new approach. Out of the n n item we have to select all possible subsets of this n item. The number of possible subsets that is there is basically 2 to the power n minus 1; that means, for this problem there are 2 to the power n minus 1 solutions are possible. N my if the n items are given to you. Out of this 2 to 2 to the power n minus 1 there maybe 1 or more solutions are the optimum solution; that means, it will give the maximum the objective values.

Now, here as a genetic algorithm task is to search for this optimum value. Now, how we can decide the encoding scheme? We can decide the encoding scheme like this if one item is selected then we can represent. There are 3 n items. We can decide the n items we can decide the n items and then the length of the chromosome is n. It is basically item 1 and item 2, item 3 and so on, on n item.

Now, if an item is selected then we can press 1 if it is not selected 0 it is like this. This way the n out of n items the subset of n items which have been selected for the solution is a 1 chromosome. This way you can represent the chromosome.

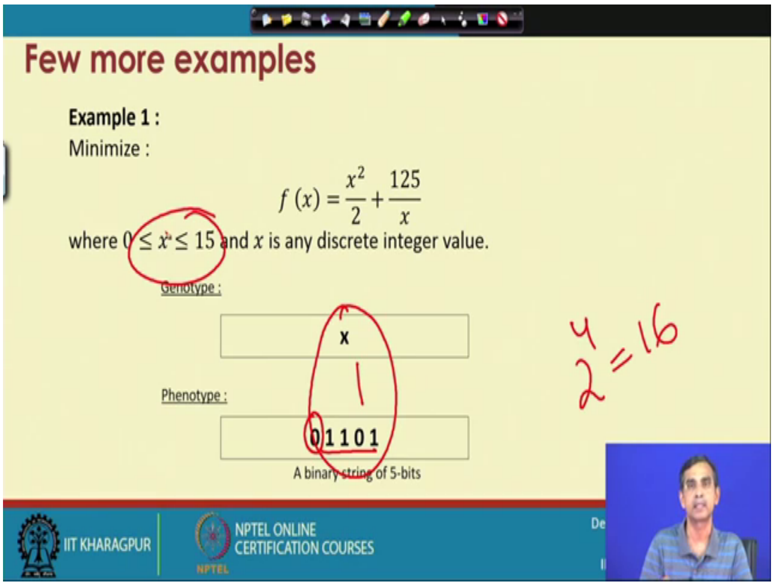
Now, for example, in this case the number of item is 3. The chromosome length is 3 and the different chromosome that is possible in this context is shown here. As the n is 3, the different number of chromosome in this case is basically yeah basically 2 to the power n all possible permutations you can see. This 1 are the different, different chromosomes representing the different solution. What we can say this is the 1 solution. The solution is basically same as this 1 this another solution represent this 1 and similarly this is the another solution is this 1. Different solutions can be encoded and this is basically the encoding scheme in this case this is a Knapsack problem.

(Refer Slide Time: 20:56)



Now, the encoding scheme for the 0 1 knapsack problem in general for n items has the length of the chromosome. Chromosome is n, it is a binary string of n bits and this is the structure. This is basically a particular instance; that means, a particular individual we can say or a solution and the solution is called phenotype and this is the genotype this means that any ith bit is basically represent the ith parameters. Here n parameters basically to solve the problem whether 1 particular items is to be selected or not is the concern.

(Refer Slide Time: 21:42)

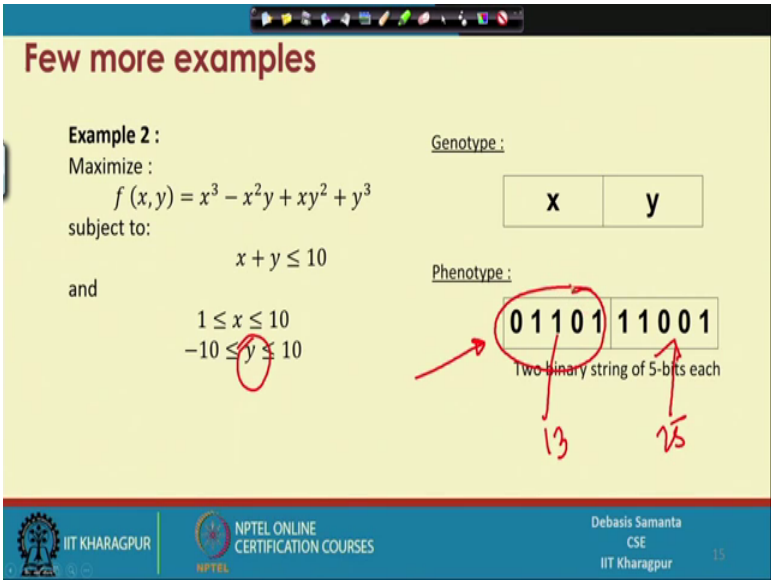


Now, let us practice this concept with few more example, I want to give one simple example here. Suppose, this is the one optimization problem where objective function is defined here and this objective function is to minimise x square by 2 plus 125 by x. That mean we have to solves the values of x for which this function fx has the minimum value. And suppose the range of the values is in this range x is greater than or equals to 0 and less than equals to 15. And suppose any discrete integer value that mean 0 1 2 3 are the values not the real values.

Now, if this is a problem as the it contents only 1 parameter namely x. It is genotype consist of only x. This is a genotype only very simple and so far, the phenotype is concerned we have represented one phenotype it is like this. It has the value say 1 and then 4 it 13. This is basically represent 13 the equivalent value in decimal.

We have understood it. Now you can see one thing that you can note it. For x within this range 0 to 15. There are different values is basically 16 and for the 16 any discrete values integer values representation we need maximum 4 binary bits I mean 2 to the power 4 this equals to 16. In order to represent 16 numbers uniquely, we need the 4 bits or binary string of length 4. Here, we have considered; however, 5 absolute no problem in that case only this is the MSB most significant bit will be 0. Within this 4 bit we can represent. Minimum requirement is 4. And we have considered here 5 we can consider of course, but minimum if it is possible then we should go for this; that means, this x can be represented with 4 bit. For this constant is concerned.

(Refer Slide Time: 23:58)



Now, as another example this is another example little bit complex compared to the previous one. And here if this is the objective functions to be minimised and you can note that this objective function is in terms of 2 parameters x and y. Your design parameters is 2 namely x and y and this is the type of the objective function we have to write; that means, we have to find the good values of x and y for which this f can gives us the minimum value.

Now, here again there is a constant, the value of x and y should be chosen in such a way that it will say it will satisfy this inequality. That mean x plus y should be less than equals to 10. Now, without any special thinking and here also we see other different what is called the range of values that the x and y should be x should be within 1 and 10. 10 different values and y should be minus 10 to 10 so 21 different values.

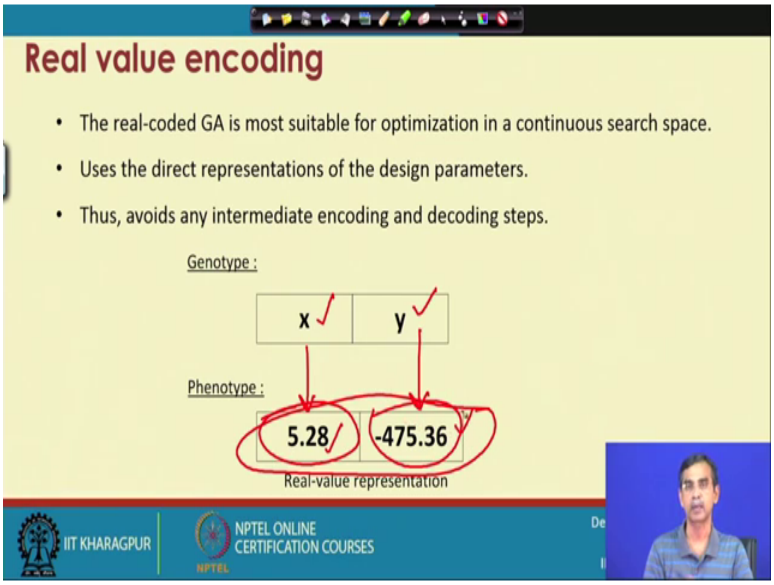
Now, here 10 different values we can easily use I mean for the representation of x the 4 binary bits; however, I have used the 5 binary bits absolute no problem. Similarly, for the 21 we can represent the 21 different numbers within the range minus 10 to 10 by another 5 binary bits it is the 5 represents.

Now, at any instance it basically at any instance this is basically 1 instant or 1 individual or 1 solution. At any instant, the value of this binary bits here is basically 1 and then 4 and 8 13 this basically 13th. And similarly, this basically 16 20 this is basically 25 25 it is 1 2 this is not correct anyway. This is this represent some 25 may be. This represent 20;

however, this is not in this range when you will check it this will be excluded. Anyway, these are the different genotype and then concern is the phenotype is there we have discussed it.

Now, we have discussed about the different encoding there the binary encoding scheme.

(Refer Slide Time: 26:20)

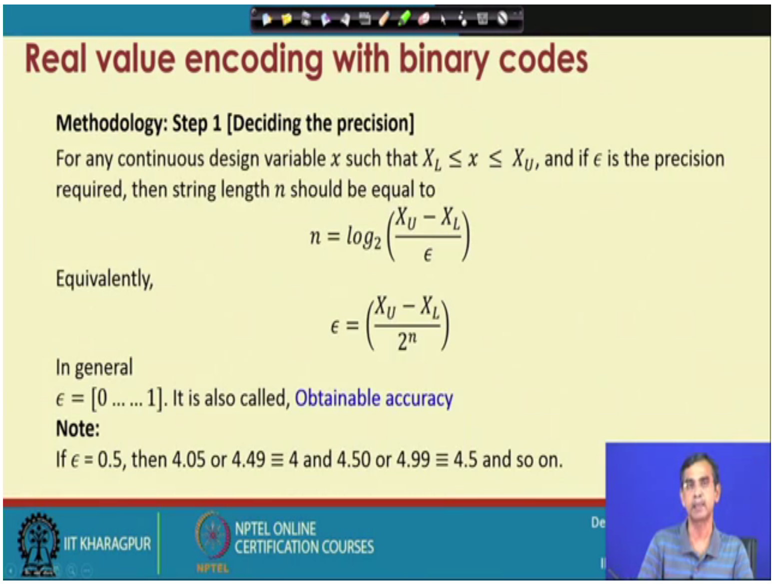


And now, we will come to the discussion of real value encoding scheme. The real coded GA is more suitable for optimization in a continuous search space. And it basically use the direct representation of the design parameter unlike in case of binary encoding scheme, there is a need to convert any value into their binary equivalent. But, here no need to represent in a binary it is a straightforward.

Now, for an example if an objective function is consisting with 2 design parameters namely x and y. And they are values at any instant say x is 5.28 and y is this 1. These constitute the 1 what is in combination or phenotype or is a chromosome. It consist of 2 values right one is 5.28 and this one. This constitute a solution; that means, at any instant the solution is that x equals to 5.28 and y equals to minus 475.36.

Now, the real value encoding scheme is very simple.

(Refer Slide Time: 27:28)



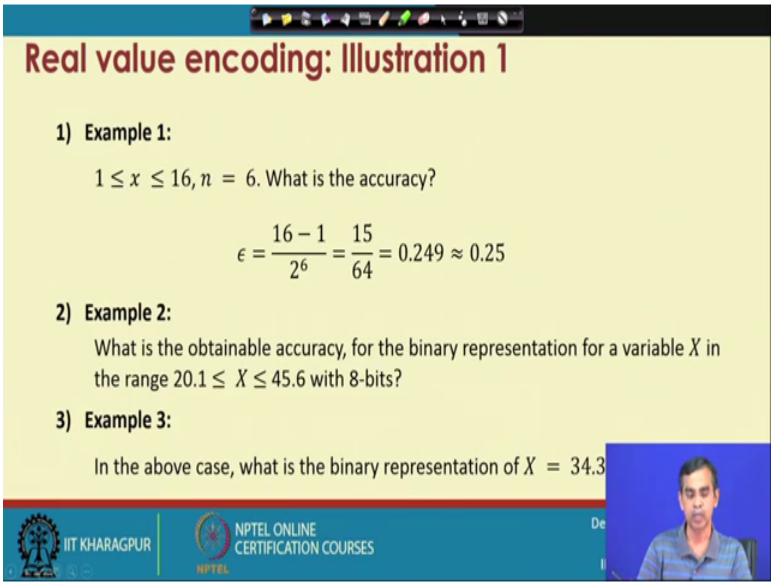
However, real value encoding also can be in the form of a binary course it is the more usual practice basically. This is because, the binary encoding scheme is more faster and gives more accurate results. That is the many users many programmer they prefer the binary encoding scheme. And in 2 3 slides I will quickly give an idea about how the binary encoding can be adapted in the real value encoding scheme. It is basically use some formula.

The formula is that first you decide to represents a real value. How many binary bits is required at least? So, this basically gives a formula. This formula gives that how many binary bits is required to represent a value. And if the value has it is range from XL to XU. Where XL is the lower range and x is the upper; that means, if a value is lying within the value range XL to XU then n can be decided by this one. And here one important factor epsilon, epsilon decides there what is the obtainable accuracy? I mean how much accuracy that you want to have it.

Now, from this expression epsilon also can be calculated within this formula now. Within this thing obtainable accuracy can be calculated and then based on this obtainable accuracy, we can decide n the number of bits that is required to represent a real value. For example, if epsilon equals to 0.5 then 4.05 and 4.49 all the values will be within this range will be represented by 4 it is this one.

On the other hand, if it is epsilon equals to say 1 then 4.00 to 4.99 should be represented by 4 like this one. Depending on the obtainable accuracy the range or precision will change and accordingly the number of bits will be decided. Now, here this is the formula that we should follow in order to understand how many bits is required and with a desirable accuracy it is there.

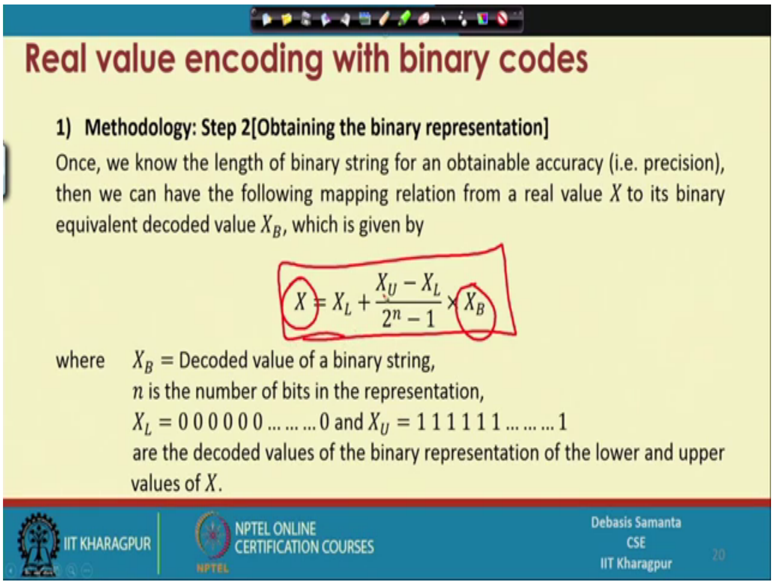
(Refer Slide Time: 29:34)



Now, this can be followed to solve it as an example, we say suppose x is within this range and n is the 16; that means, we decide the number of bits that is 6 then obtainable accuracy if x is within this range then it can be call 0.25. Within the range of the values and then n we can decide the obtainable accuracy and finally, we can use this one to calculate the number of bits. For example, you can easily calculate, With 8 bits and if the number is within this range and what will be the optimal accuracy?

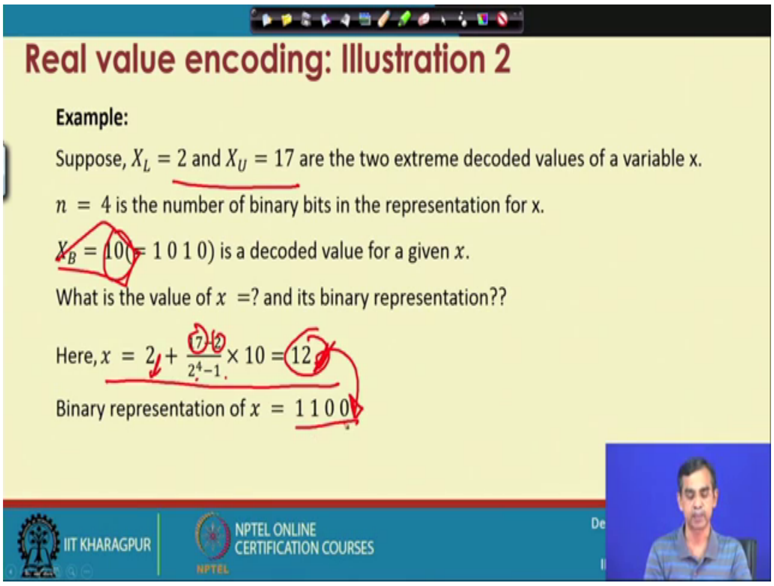
Now, for example, say suppose x equals to 34.35 is a representation then what is it is corresponding binary scheme, binary value? Now, let us see how this can be obtained it can be obtained easily.

(Refer Slide Time: 30:27)



Now, this is the formula that can be used to understand that if XB is the current value and then X is basically it is a binary representation. How it will be this is the standard formula that is followed here and n is the number of bits in this case.

(Refer Slide Time: 30:45)



Now, for an example we can understand about say XL equals to 2 and x equals to 17; that means, lower limit and upper limit of the parameter x. And n is the 4 that means, number of bits that is required and say suppose XB the binary value that is 10. Which is basically in the binary representation can be like this 1 0 1 0 0 2 then 4 8. It is 10 this is basically the decimal equivalent, this 10 is the decimal equivalent of this binary values.

Now, having this one what will be the x that can be represented if it represent xb. This can be obtained using this formula. Here, this is basically the XL and this is XU minus XL 2 to the power n minus 1 into this is the current value XB. It is 12; that means, this this XB which is 10 actually it is 12. For the real value is concerned in the range 2 to the power 17 and this 12 can be represent using binary it is called 1 1 0 0 representation.

Here, is the idea about that within a particular range XL and XU and given the obtainable accuracy will be able to represent any binary value in the binary encoding scheme. With this things we have learned about the binary encoding scheme and in the next we learn about the other encoding scheme binary encoding scheme and the real value encoding scheme is cover, the order encoding scheme will cover in the next lecture.

Thank you.