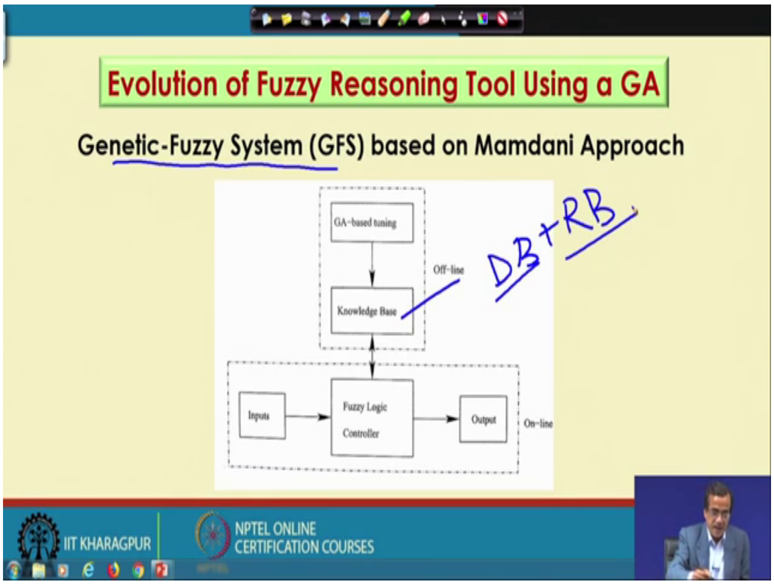
Traditional and Non-Traditional Optimization Tools
Prof. D. K. Pratihar
Department of Mechanical Engineering
Indian Institute of Technology, Kharagpur

Lecture – 41
Genetic Algorithm as Evolution Tool (Contd.)

Evolution of fuzzy reasoning tool, using a genetic algorithm, now fuzzy sets are sets with imprecise boundary now there are several applications of the principle of fuzzy sets.

(Refer Slide Time: 00:36)



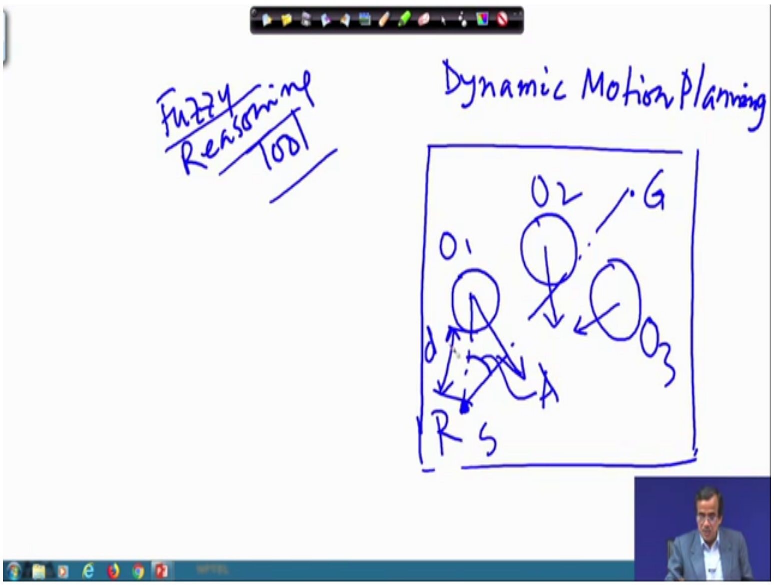
And fuzzy reasoning tool is one of them, now the purpose of developing this fuzzy reasoning tool is to establish the input output relationship of a particular the process. Now to establish this input output relationship the first thing we will have to do is for this fuzzy reasoning tool. So, we will have to design and develop a suitable knowledge base.

Now, knowledge base consists of the Data Base and the Rule Base now to define this data base and rule base I am just going to take one practical example now before that let me tell that our aim is to develop this genetic fuzzy system now here the purpose of using genetic algorithm is to design or evolve a suitable knowledge base for this particular the fuzzy reasoning tool.

Now, to define the concept of this data base and the rule base now I am just going to take the example a practical example. Now supposing that say I have got a process having say

2 inputs and one output and this particular problem is related to one physical scenario now let me try to explain in brief that particular physical scenario.

(Refer Slide Time: 02:13)



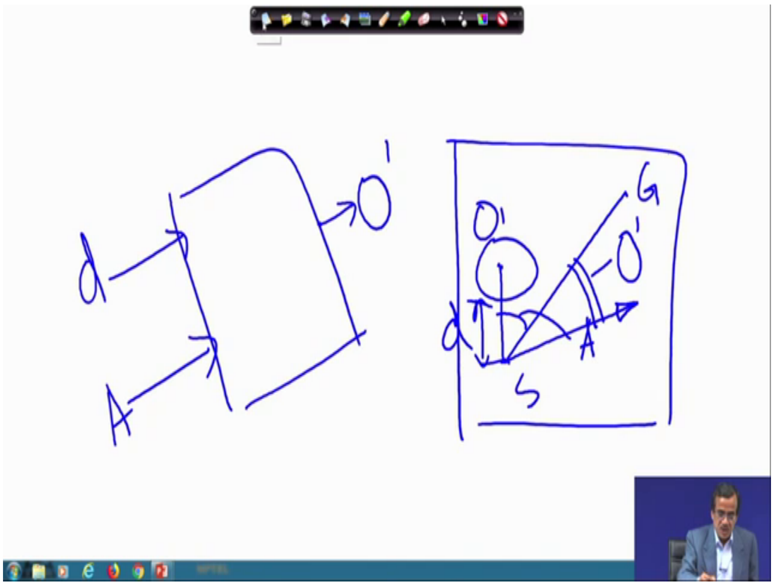
Now, supposing that I will have to solve a Dynamic Motion Planning problem of a robot. Now, here the purpose is to find out a collision free time optimal path for the point robot. So, for simplicity let me assume this is the scenario of a particular robot mobile robot and for simplicity let me consider that this is the point robot denoted by R, now supposing that so, this indicates is starting position and the goal is here. Now m is starting from here to reach this particular the goal now supposing that there is no such obstacle here. So, very easily it will start from here and it is going to reach this particular the goal.

But fortunately or unfortunately supposing that there are a few obstacles like say I have got obstacles like say here I have got a point this type of circular obstacle there is another obstacle here, there is another obstacle here. So, this particular obstacle is moving in this direction, this is moving in this particular direction, this is moving in this particular the direction. Now if you see the situation now this particular path will not remain as a feasible one and I will have to find out a collision free path and to decide the collision free path I can take the help of one Fuzzy Reasoning Tool.

Now, let us see how to design and develop one fuzzy reasoning tool to solve this particular the problem, now out of these 3 obstacles say this is obstacle 1, obstacle 2 and obstacle 3 and this is the starting position of this particular robot. Now from here it will try to look towards the goal and this particular obstacle will be treated as the most critical obstacle. Now to take care of that and to avoid collision with this particular the most critical obstacle so, what we do is, we try to find out 2 inputs for the fuzzy reasoning tool and there will be a one output that is nothing, but the deviation angle.

Now, let us see how to find out that, now what we do is the 2 inputs is one is the distance between the robot and the most critical obstacle. So, this particular distance is decided by this. So, this is nothing, but the distance input set d, now similarly I we also try to find out another input that is the angle input that is the angle between the goal the present position of the robot and the center of the most critical obstacle and this particular angle is this so, this is nothing, but say it is denoted by A.

(Refer Slide Time: 05:34)



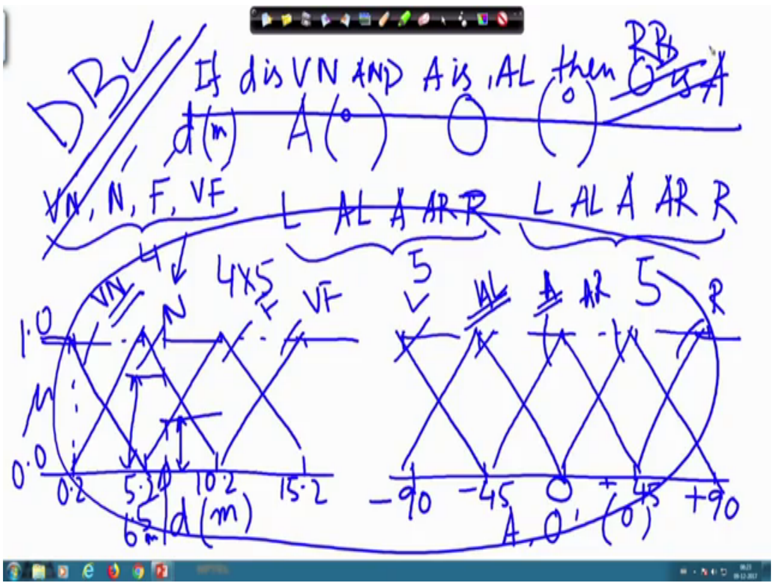
Now this d and A are the input; that means, if I just consider so, this particular thing say this is the starting position and this is the goal and this is the most critical obstacle say O 1 now here. So, one input is this. So, this is the distance input and this is the angle input now our aim is to find out what should be the deviation?

Now, what do you do is, we are going to design and develop one fuzzy reasoning tool d is one input and angle is another input and the deviation say this is nothing, but say O, or say O prime is the deviation. So, that particular deviation angle we will have to find out. So, this particular deviation angle is nothing, but the output of this fuzzy reasoning tool.

So, our aim is to design and develop this fuzzy reasoning tool. So, that it can solve the dynamic motion planning problem and to solve this dynamic motion planning problem. So, the first thing is we will have to design and develop this fuzzy reasoning tool and then we will see how to modify and how to optimize with the help of this particular the genetic algorithm.

So, I am just concentrating on this particular problem, but let us try to see in more details now what you will have to do is. So, we will have to find out a suitable membership function distribution or suitable representation for d that is the distance and a suitable representation for the angle, the input angle and of course, a suitable representation for this particular the deviation.

(Refer Slide Time: 07:05)



Now, what we do is. So, we try to take the help of some membership function distribution, now the membership function distribution I will be discussing in much more detail, now what I am going to do is supposing that. So, this particular distance say unit could be say in meter, the angle the unit could be in degree and this particular deviation the input in could be unit could be in degree, now let us try to design some membership function distribution.

Now once again let me repeat, our aim is to define what do you mean by database and rule base of a fuzzy reasoning tool that I am going to define. Now for this particular distance we are going to use a few linguistic terms for example, say it could be very near distance it could be near distance, the far distance and very far distance. Similarly this angle it could be your, the ahead angle then, ahead right then, comes ahead left and the left. So, there are 5 linguistic term similarly this particular the deviation we are going to define with the help of ahead, right, then comes ahead, left and left.

So, there are 4 linguistic terms for distance, 5 linguistic terms for the angle input and 5 linguistic terms for this particular the deviation now. So, 4 multiplied by 5 there will be actually 20 rules now how to design that particular database and the rule base that I am going to discuss.

Now, to design these, what you can do is. So, I am just going to saw the membership function distribution for this particular the distance; that means the membership function distribution for very near far and very far. Now let me take so, this type of distribution and for simplicity let me consider the triangular distribution. So, here this shows actually the very near distance. So, this particular triangle is nothing, but the very near then this is nothing, but the near then this is nothing, but the far and this is nothing, but the very far and here actually we will have 2 put. So, this is the distant and definitely there will be some unit and let me put some numerical value. So, might be 0.2 meter then 5.2 meter then 10.2 meter 15.2 meter and here we have got the membership function value mu and it is range is 0.0 to 1.0.

Now here a particular distance. So, if I concentrate on a particular numerical value for the distance say I am here now if I am here so, this particular distance. So, might be this is say 6.5 meter. So, 6.5 meter can be called near with this much of membership function value and it can also be called far with this much of membership function value; that means, the same distance can be called in here with one membership function value it can also be called far with another membership function value. So, this particular the definition of the description of the distance is not crease it is fuzzy in nature this is the concept of the fuzzy sets.

Now here now I will have to design the membership function distribution for this angle and this deviation for simplicity let me assume that they are having the similar type of membership function distribution and we can also consider the different types of membership function distribution also. Now here actually what we do is. So, we try to we try to draw the membership function distribution for these ahead. So, this could be the ahead then ahead right. So, this could be the ahead, right and this is the right now this is ahead left and this is the left. So, this is left this is ahead, left, this is ahead, this is ahead right and right and numerical value. So, this is the angle A and this particular deviation and I am just using the same membership function distribution.

Now here so, the unit is actually the degree. So, let me put 0 degree here that is ahead then ahead right let me plus 45; that means, this side and like clockwise 45 say plus 90 similarly minus 45 and minus 90. So, these numerical values we can we can assign, now once I have assign the numerical values for this particular angle input and this particular deviation and the distance. So, these information whatever I am just putting it here is known as the database of these particular the inputs and the output of this process this is what you mean by the database and once we have got this particular database, now we are in a position to design the rules for example, the first rule could be if distance is very near and angle is ahead left then the deviation is nothing, but ahead. So, this is the rule.

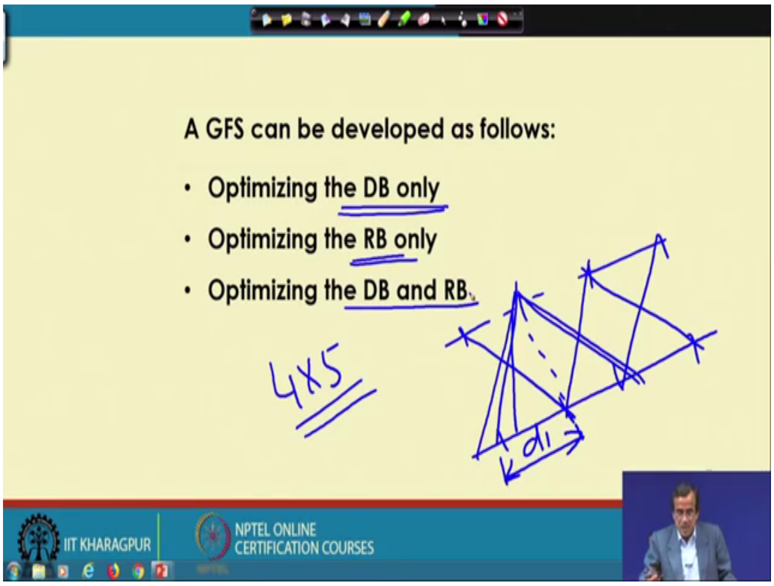
So, let me just write here a particular rule if the distance d is very near and the angle A is a ahead left then the output that is the deviation is nothing, but ahead. So, this is a particular rule this is a rule similarly I can design 4 multiplied by 5 20 rules in the rule base and this is what I mean by the rule base of a fuzzy reasoning tool. So, I hope the idea behind this particular the data base and the rule base is clear, now I have been a position to proceed further to discuss the working principle of this.

So, let us try to concentrate here. So, our aim is to design and develop the knowledge base which is nothing, but a combination of data base and rule base which I have already discussed. Now this particular database and rule base can be actually optimized with the help of a genetic algorithm, but genetic algorithm is computationally expensive. So, this particular training has to be implemented offline and once it is trained now we can use for some test scenarios online; that means, I can pass say one set of inputs to the GA optimized or the GA tuned or the GA evolved fuzzy reasoning tool or fuzzy logic controller and for these set of inputs.

So, I will be getting this particular output and of course, this is online because within a fraction of second for a set of inputs I will be getting this particular the output, but the training has to be provided offline because GA is computationally expensive and this particular training is provided with the help of some pre determine or pre collectorate training scenarios.

Now, let us see how to proceed further.

(Refer Slide Time: 15:31)



Now, if you see the literature now this genetic fuzzy system actually has been defined in different ways; that means, we can optimize the database only, now how to optimize this particular the database now if we take the same example of this particular the triangular membership function distribution now for example, say this is say the very near this is the near distance and this is the far and this is very far.

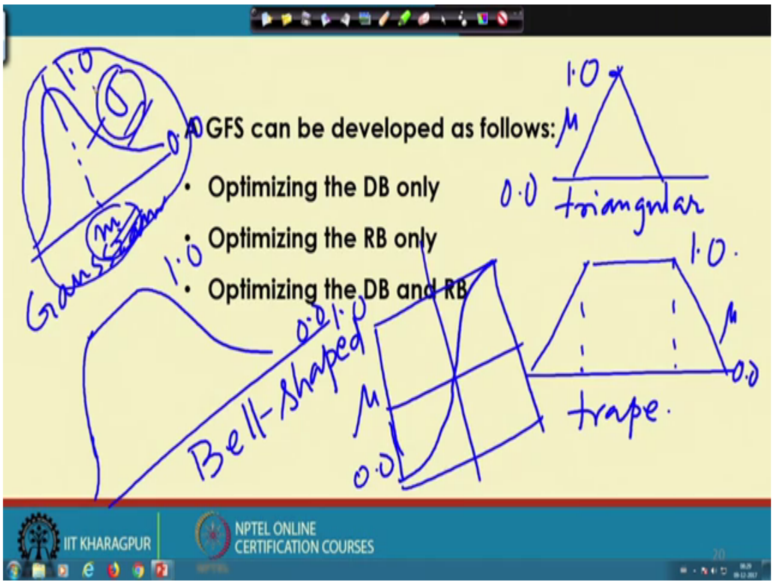
Now, if you see. So, this is nothing, but the half base width of this particular triangle say d 1 and what you can do is so, during the optimization. So, I can find out what should be the optimized value for this particular d 1. So, d 1 is going to vary in a range and it will try to find out what should be the optimal value. So, if d 1 is more. So, this will be some sort of flatter distribution for the triangle and if d 1 is small then it will be steeper. So, d 1 is large we will be getting the flatter distribution and if d 1 is small. So, you will be getting that particular the steepest distribution for this particular triangle.

Similarly we can also optimize the rule base, now if you take the same example. So, out of these 4 multiplied by 5 rules which are present in the rule base all the rules may not be equally good. So, using GA we can find out what should be the optimal rules and that is actually what you mean by optimization of rule base.

Now, the performance of this fuzzy reasoning tool depends on both database as well as rule base and that is why we can also optimize the data base and rule base the both the things either in stages or in or simultaneously. So, using any one of these we can find out what should be the optimal the knowledge base of this particular the fuzzy reasoning tool.

Now, before I proceed further. So, let me tell that we use different types of membership function distribution.

(Refer Slide Time: 17:49)



Now, as I told the triangular membership function distribution we use very frequently because it is computationally efficient and it is faster. So, this is actually the linear distribution and this is nothing, but the triangular membership function distribution. Now similarly there are some other membership function distributions like we use some set of trapezoidal membership function distribution. So, this type of trapezoidal membership function distribution also we use.

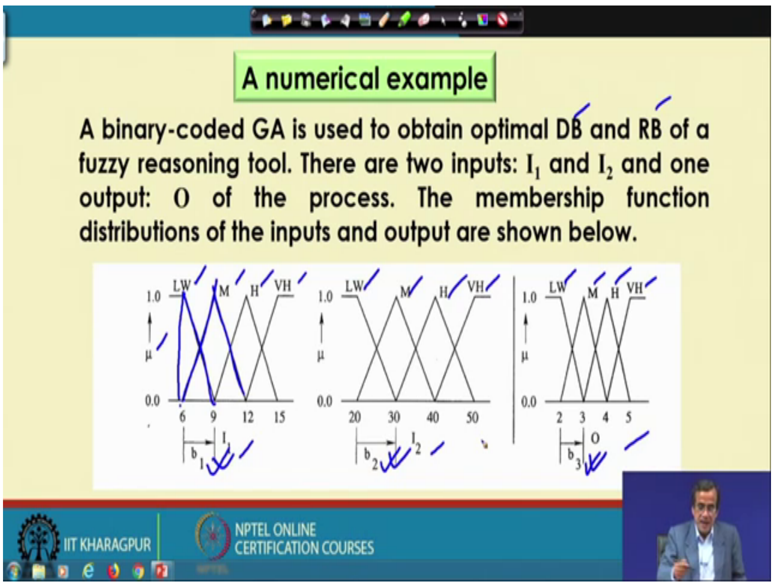
Next we use some sort of Gaussian distribution for example, the Gaussian distribution could be something like this and this particular Gaussian distribution is decided by the mean and the standard deviation value. So, this is nothing, but the Gaussian distribution of the membership function distribution and sometimes you use the bell separate curve like for example, it will be something like this. So, it is almost similar to these. So, this is nothing, but the bell separate curve and sometimes we use some sort of sigmoidal membership function distribution and it is looking like this. So, the sigmoidal membership function distribution.

So, this type of distribution so, mu equals to 0.0 and this is 1.0. So, everywhere here actually mu is equals to 0.0 these corresponds to 1.0 similarly these corresponds to 0.0 and this is 1.0 similarly here it is 0.0 and mu equals to 1.0 0.0 and here it is 1.0. So, this is the way actually we can design the different types of membership function distribution and we just take a particular the distribution and if required we can further optimize the shape and size of this particular the membership function distribution.

For example, say if I want to optimize the membership function distribution which is nothing, but Gaussian. So, what you can do is, we can keep this m that is the mean and the standard deviation sigma as the design variables inside the GA stage and GA is going to supply what should be the suitable value for this particular m and sigma. Similarly the different other types of membership function distribution we can optimize with the help of this genetic algorithm.

Now if this is the situation now we are in a position to design the fuzzy reasoning tool and let us see how to evolve or how to design the optimized fuzzy reasoning tool.

(Refer Slide Time: 20:30)



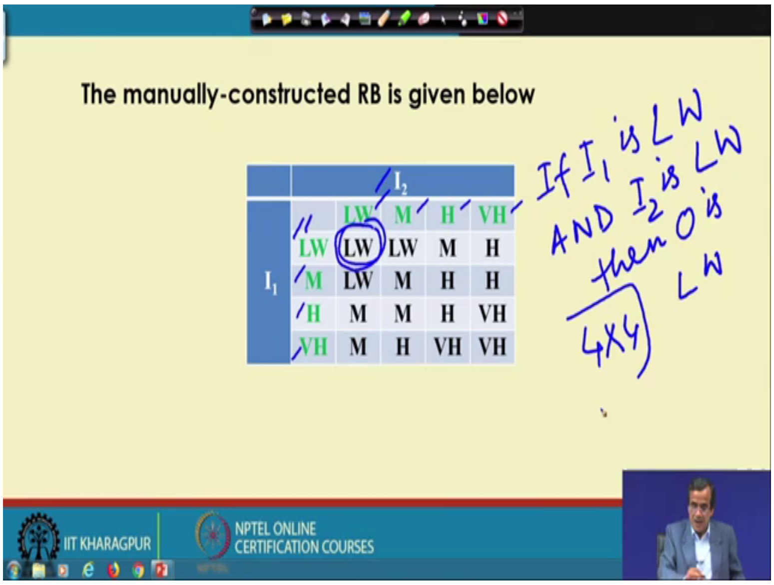
Now here the, I am just going to discuss the working principle like how to evolve a fuzzy reasoning tool with the help of a genetic algorithm. So, that particular problem I am just going to discuss with the help of a numerical example. Now let me state the numerical example first, a binary coded g is used to obtain optimal database and rule based optimal database and rule base of a fuzzy reasoning tool. So, we know the meaning of database and rule base there are 2 inputs I 1 and I 2 and one output that is O of the process. So, here if you see so, these are the membership function distribution for the 2 inputs I 1 and I 2 and further the output and for simplicity we have considered the triangular membership function distribution.

Now, for this particular the first input that there are 4 linguistic terms like low, medium, high and very high similarly for I 2 we have got low, medium, high and very high and for this output you have got low, medium, high and very high. Now this b 1 is going to decide like your the size of this particular your the triangular membership function distribution.

So, b 1 is nothing, but the half base width of this particular isosceles triangle or the base width of this particular right angled triangle. So, this b 1 is nothing, but the design variable for this I 1 and here similarly b 2 is the design variable for I 2 and b 3 is the design variable for this particular output O and for simplicity we have considered the symmetrical membership function distribution and as I told that for simplicity we have considered the triangular membership function distribution and mu is nothing, but the membership function distribution which will vary from 0 to 1.

Now here we can see for I 1 we have put some numerical values like 6, 9, 12 and 15 and as I told we have considered the symmetrical membership function distribution, similarly for I 2 the numerical values are 20, 30, 40 and 50 and for this particular output the numerical values are 2, 3, 4 and 5 they are having their respective units and respective ranges.

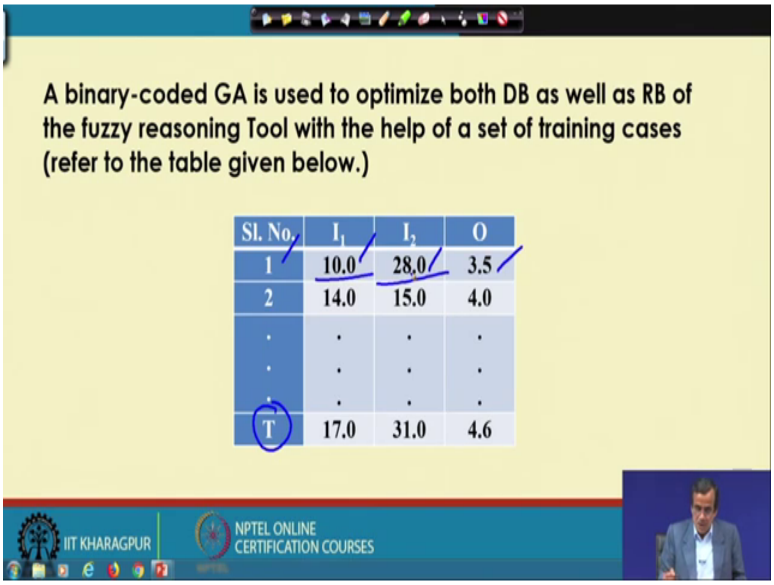
(Refer Slide Time: 23:23)



Now, to proceed further so, what we can do is now as there are 4 inputs for this particular I 1 and 4 inputs for this particular 4 linguistic terms for I 1 and 4 linguistic term for this particular I 2. So, we have got 4 multiplied by 4 there are 14 rules and these rules are manually design rules. So, based on the experience of the user the user can define this particular the rule base.

Now, for example, the first rule will be something like this. So, this is the output of the first rule is as follows if I 1 is low and I 2 is low if I 1 is low, I 2 is low so, output is low, then the output O is low. So, this is actually the first rule now similarly there will be 16 such rules and these are all manually constructed rule. So, it may not be optimal it may not be good some of the rules could be good, but all the rules may not be equally good. So, GA we will try to find out what should be the optimal rule based like which one out of these 16 are really good.

(Refer Slide Time: 24:52)

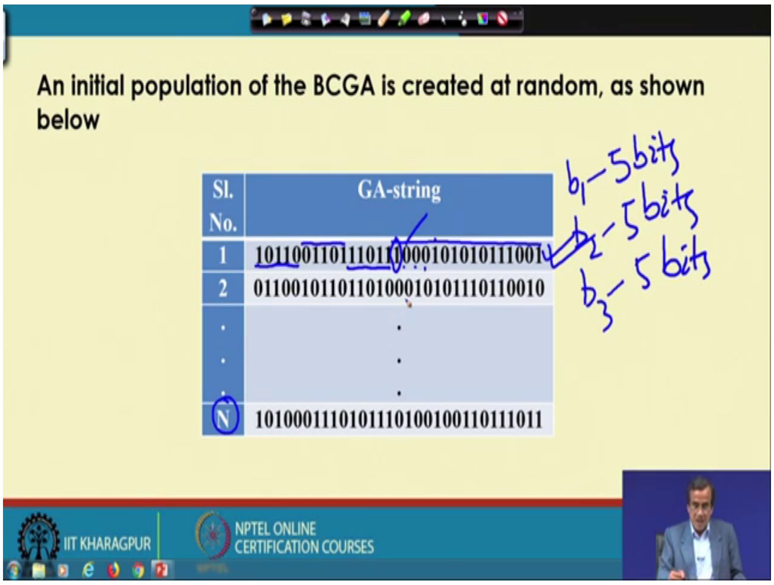


Now, let us see how to how to do it now, a binary coded GA is used to optimize the database and as well as the rule base of the fuzzy reasoning tool with the help of a set of training cases. So, here there are large number of training cases and the training cases are nothing, but the known input output relationship for example, the first the training case is nothing, but these if I 1 is 10.0 and I 2 is 28.0. So, output is 3.5 similarly we have got capital T number of the training scenarios and these are pre collected and the known values.

Now here one thing I just want to mention now unlike the neural networks like here the inputs are actually a sent in the real scale not in normalized scale if we remember in neural network we used to send the inputs in normalized scale, but in fuzzy reasoning tool we can supply the inputs that is I 1 and I 2 in the real scale you need not go for the normalized scale the reason is very simple because there will be fuzzification and during the fuzzification actually there will be normalization.

So, we need not send these particular inputs in the normalized form. So, this is the difference between the neural network and fuzzy reasoning tool.

(Refer Slide Time: 26:24)



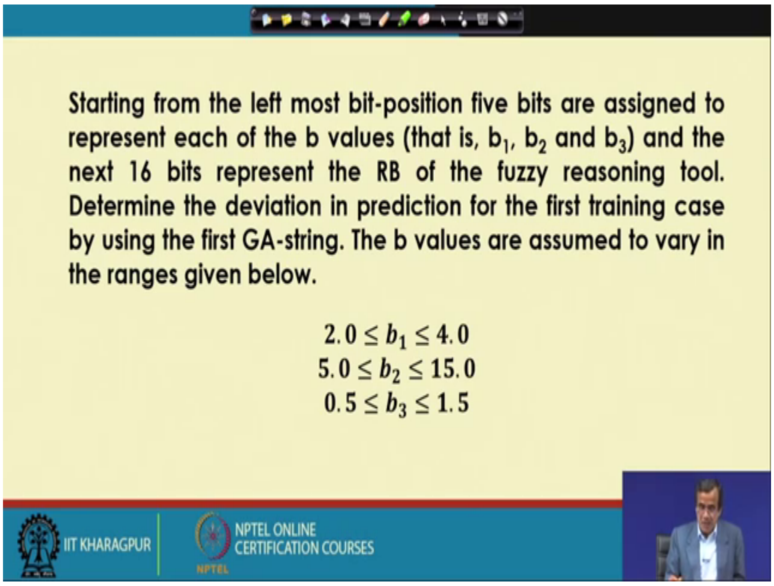
An initial population of the binary coded g is created at random. So, this shows the initial population of this particular the binary coded GA and the population size is capital N and here actually this particular GA- string that is created at random now what should be the length of this particular the GA- string.

Now if we remember we have got 3 actually the real variables like b 1, b 2 and b 3 and they are going to indicate what should be the half base width of the triangular membership function distribution representing I 1, I 2 and the output O and let me assume that we are assigning 5 bits each to represent. So, 5 bits, 5 bits and 5 bits; that means, you are the first 5 bits is going to represent your this particular the b 1. The next 5 bits is going to represent the b 2 and the next 5 bits are going to represent b 3. So, we have we have used 15 bits here 5 plus, 5 plus 5 and there are a few other bits here and. In fact, we are using 16 bits here just to represent the 16 rules which I have already discussed.

So, these 16 manually constructed rules are represented by. So, these particular your the 16 bits now a particular bit if it is 1 then it indicates that that particular rule is present and if it is zero; that means, that rule is absent; that means, so, here we have got a 1 here; that means, the first combination that the first rule or the manually constructed rule base is present, but the second, third, fourth are absent and the fifth is present and so on. So, out of these 16 rules only a few will be present and others will be absent corresponding to this particular the GA- string similarly we have got the other the GA- string.

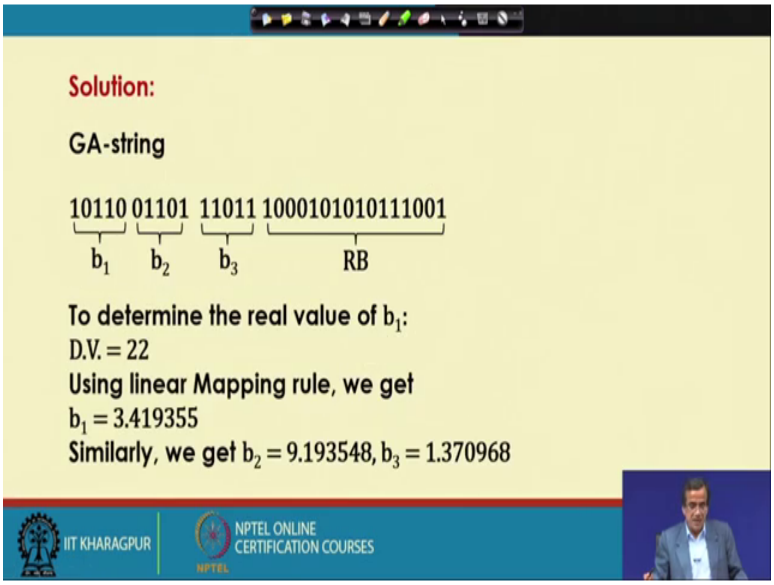
Now, here once we have got this particular GA- string.

(Refer Slide Time: 28:40)



Now, we will have to state the range for the variables now here we can see whatever I mentioned the same thing I have written it here just starting from the leftmost bit position 5 bits are assigned to represent each of the b values like b 1, b 2 and b 3 and the next 16 bits represent the rule base of the fuzzy reasoning tool and our aim is to determine the deviation in prediction for the first training case by using the first GA- string and the b values are assumed to vary within these particular values like b 1 is lying between 2 and 4 then comes. So, b 1 is lying between the 2 and 4 and then b 2 is lying between 5.0 and 15.0 and b 3 is lying between 0.5 and 1.5.

(Refer Slide Time: 29:44)



Now, here so, this is the range for this particular the variables now we are going to solve it.

Thank you.