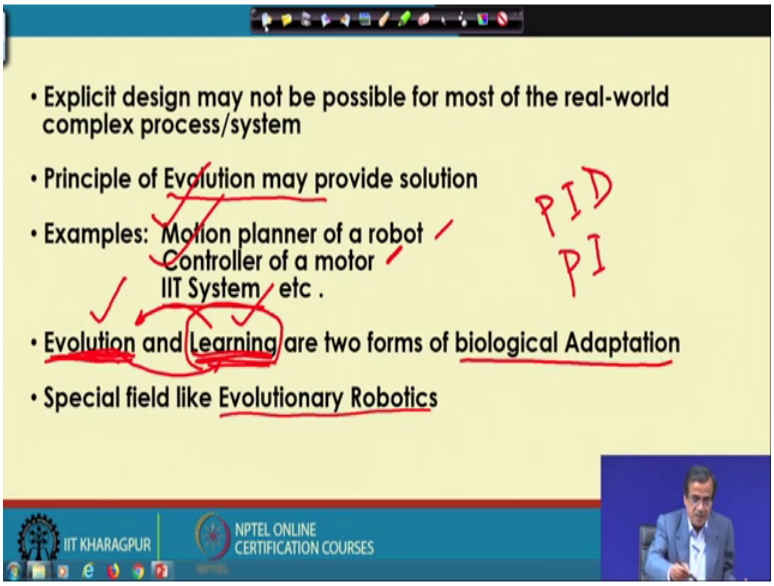
**Traditional and Non-Traditional Optimization Tools**
**Prof. D. K. Pratihar**
**Department of Mechanical Engineering**
**Indian Institute of Technology, Kharagpur**

**Lecture – 37**
**Genetic Algorithm as Evolution Tool (Contd.)**

Let us start with the next topic. It is on genetic algorithm as evolutionary tool. Now if you see the real world problem or the real world process, these are bit difficult and explicit design could be little bit difficult to achieve and that is why we will have to use the principle of evolution.

(Refer Slide Time: 00:30)




Now, if you see the principle of evolution. Now we can see the way we can evolve the most effective design. Now let me take a few simple examples; for example, say the motion planner of a robot, now supposing that we are going to design and develop one say intelligent robots, and for this particular robot we will have to design and develop the suitable motion planner, the adaptive motion planner.

Now, based on our experience we are able to say that it is bit difficult to explicitly design, what should be the most suitable muscle panel for a particular, the intelligent robot and that is why to design and develop the suitable muscle plan planner. So, we will have to take the help of the principle of evolution. Now let us take another example like the controller of a motor. Now if you see in a robotic joint, each of the robotic joint we use a particular the motor and for this motor we have got some controller. For example, sometimes we use like PID controller; like proportional integral derivative controller or sometimes we use PI controller like proportional integral controller.

Now, if I want to design and develop an adaptive controller for the robot adaptive controller for the motor, it is bit difficult and that is why we generally go for the principle of evolution, just to evolve that particular, the adaptive controller for the motor. Now let me allow to take another example; that is the example of our IIT system. Now if you see the curriculum of IIT system or the academics of IIT system. So, it has not come through the explicit design or the direct design, and it has come through the principle of evolution. Now nobody could design explicitly what should be the status of IIT after a few years, and this particular status, the academic status of IIT system has come through the principle of the evolution, and that is why the principle of evolution is little bit a is a thing which we should understand very carefully.

Now, if you see the principle of biological adaptation. Now in biological adaptation, there are two terms; one is called the evolution and another is called the learning. Now this evolution and learning are two forms of biological adaptation, but they walk on two different timescale; for example, learning takes place, you know within one's lifetime, whereas, the evolution takes place through a large number of generations. Now these two terms like evolution and learning, they are going to help each other.

Now, let us see; how do they can help each other; for example, say if you see the principle of learning, actually we use the principle of optimization. Now here if you see one optimization tool, the principle of evolution is generally copied in the optimization tool, and that is why the evolution is going to help in learning. On the other hand let us see how this particular learning can help, the rate of evolution. Now if I can learn some good things, now those good things will be passed to the next generation and these good habits, good things if we can cultivate. Now those things will be pass it to the next generation, there is a chance the rate of evolution is going to increase.

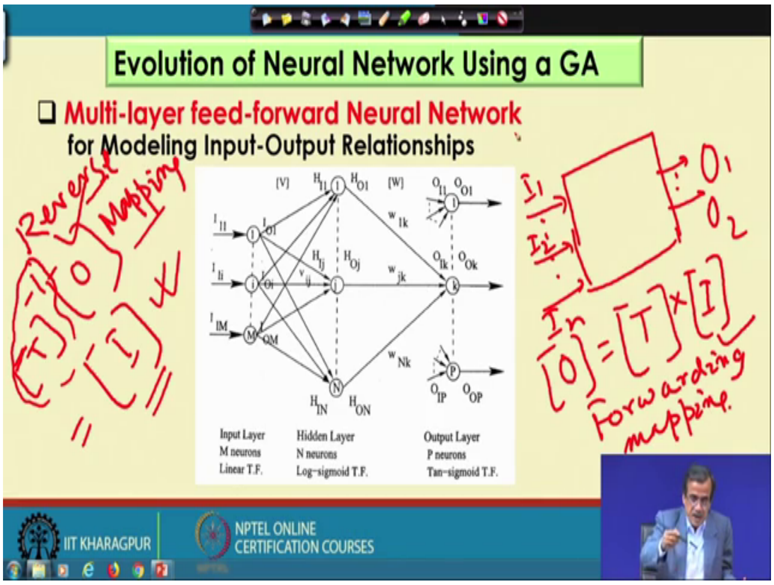
So, it is going to accelerate the rate of evolution. So, evolution is going to help learning and at the same time learning is a way also going to help this particular the evolution. So, they are going to help each other, and consequently it accelerates the, that the rate of biological adaptation. The same principle we can use here in the artificial way to design and develop like the complicated the system or the process like and if you want to modulate in a very efficient way. So, this is the way we should do.

Now, if you see the evolution tool for example, we have already discussed the working principle of genetic algorithm as a very efficient optimization tool. Now this genetic algorithm can also be used as the evolution tool. Now on the other hand if you see we have got a few tools; like the learning tools for example, we have got the neural networks, the fuzzy logic system which I have not yet discussed. So, those things could be very efficient learning tools. Now here in today's class, actually I am just going to give some sort of brief introduction to the learning tools like that the neural network and we will see how to combine this particular genetic algorithm with the principle of your this neural network.

So, that we can evolve a very efficient learning tool and that is going to accelerate your the rate of adaptation. And if you see the robotics, the current field of robotics. This evolutionary robotics is actually one of the most current fields of robotics research. Now here in this particular field, we use the principle of evolution and that of learning, and we merge them together just to evolve some suitable muscle planner for the robot or just to evolve like adaptive controller for the robot used in the adaptive controller for the motor used in robot.

Now, let us see how to combine and how to use genetic algorithm as a evolution tool like, so that we can evolve the suitable neural network, and this particular neural network is going to solve some real world problem. Now let us see this particular principle in details. Now before I go for that, let me tell that we use some artificial tool for evolution that is nothing, but genetic algorithm, and we use some other artificial tool for learning that it could be either neural networks or fuzzy logic technique, and we will try to evolve this particular, your the neural network with the help of genetic algorithm. Now, let us see how can we combine and we can take advantage of both these two techniques.

(Refer Slide Time: 07:50)




Now, let me concentrate on this; that is like how to evolve a suitable neural network using a genetic algorithm. Now if you see the literature of this particular network, the neural network a huge literature is available. Now I am not going to discuss on all these particular neural networks instead. I am just going to concentrate on a particular design of the network and that is called this multi layered feed forward network. Now as I told we have got some other networks like radial basis function network, then comes your recurrent network, the self organizing map, the back propagation algorithm, counter propagation algorithm and as I told, I will be concentrating only on this multi layered feed forward network.

And our aim is to evolve this particular network so that it can, it can serve some purpose. Now before I go for discussing the principle of this particular network and how to evolve this network using a GA. Now let me just spend some time on some preliminaries. Now let us try to understand; what is the application of this type of network. Now let me take a very simple example, supposing that I have got a process, I have got an engine process, say and here I have got a few inputs like I 1 I 2 up to say I n. So, I have got a few inputs and I have got a few outputs also.

Now, our aim is to find out. So, this particular input output relationship. Now the way we do the conventional method is like this. So, we try to express this particular output as a function of the input parameters; that is output o is nothing, but the transformation matrix multiplied by the these input parameters. Now what we do is, we supply a set of inputs and if I know; these particular transformation matrix, I can find out the set of output.

Now this is known as the forward mapping, and this is required actually just to find out the input output relationship of a particular the process. So, this is known as the forward mapping.

Now, if I want to find out the relationship in the reverse direction; that means, if I want to find out what should be the set of inputs so that I can reach a predefined set of output? Now let us see how to solve that particular the problem. Now to solve this particular problem, so here on both the sides. So, I multiplied by the inverse of this transformation matrix that is nothing, but. So, T inverse multiplied by o that is the output is nothing, but the identity matrix multiply and that is nothing, but i.

So, if I know the transpose and if I know the inverse of this particular transformation matrix, and if you multiply with the output; so, I will be getting this particular the input and as I told our aim is to find out what should be the set of input, so that I can achieve one desired set of output, so this is known, and if I know this particular the inverse of the transformation matrix. So, very easily I can find out what should be the set of inputs. Now this problem is known as the reverse mapping. So, this is nothing, but the reverse mapping. So, this is known as the reverse mapping.

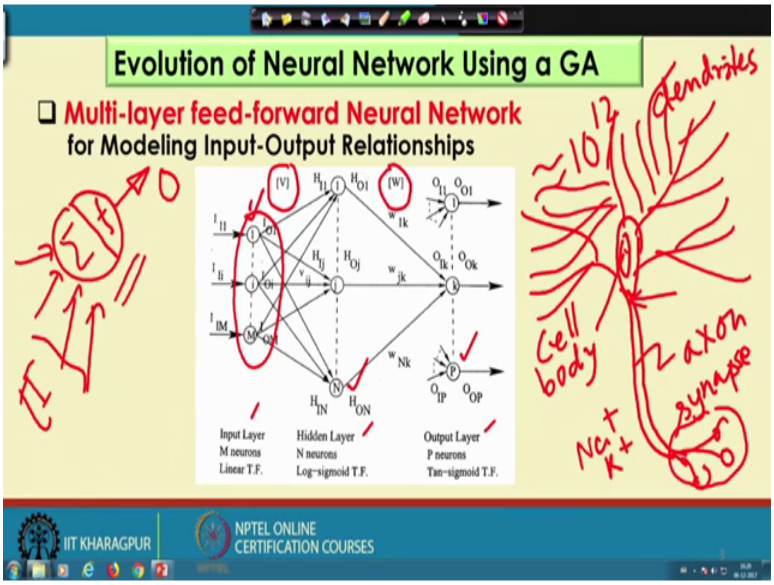
Now, this reverse mapping problem we can solve, provided we know the inverse of this particular the transformation matrix, but for most of the problem the transformation matrix may not be the square matrix, and it may not be invertible. Now if it is not invertible, so very accurately we cannot find out. So, this particular input output relationship in the reverse direction, then how to solve this. To solve this particular problem we can take the help of say this type of network. So, I can take the help of this multi layered feed forward network. So, this is one of the application.

Similarly, there could be a number of applications or this type of network. Now so we have understood; why should we study this type of network? Now let me see; what is there inside this particular the network. Now to see what is there inside this particular network. So, what we will have to do is, the first thing we will have to understand what are the different elements of this particular the network. Now let me read out the name of this network once again, this is multi layered feed forward neural network. Now here we have got more than one layers. For example, say I have got input layer then comes the hidden layer and the output layer and for simplicity I have considered only three such layers.

Now so, this particular hidden layer is actually going to dictate what should be the architecture of this particular network, now that I am going to discuss after some time, but before that let me try to understand what is there inside one neuron. So, this particular circle is going to represent one neuron, and this is one artificial neuron. Now here actually what we do is, we copy everything from the biological neuron. Now let us try to see in short, the principle of this particular, the biological neuron first and then I will try to design this particular the artificial neuron and then gradually I will just going, I am just going to constitute a layer of the network and then this the whole neural network and we will see how to evoke this particular network with the help of a genetic algorithm.

Now, before I go for that. So, let me try to concentrate on this particular the neuron, the artificial neuron. And as I told if I want to understand the principle of this particular artificial neuron, the first thing I will have to understand the principle, the working principle of the biological neuron, and we know that in our nerves system we have got a large number of neurons.

(Refer Slide Time: 14:34)




So, the number of the neurons could be in the order of say 10 raised to the power 12; say it is a huge number and all such neurons are working in parallel in our brain. So, our brain is a highly complex parallel computer, but before that let me try to see; what is there in a particular the biological neuron.

Now, in biological neuron if we see, so we have got one cell body or the soma. For example, say this is the cell body or the soma and we have got one fiber, the long fiber and this is known as actually the action. So, this is actually known as the action and this is the cell body or the soma, and here we have got a few dendrites. So, we have got a large number of dendrites and with the help of this large number of dendrites. So, these are all thin fibers. So, this neuron is going to collect information from the neighboring neuron, so these are all dendrites. So, we have got the dendrites here and with the help of dendrites, it will collect information and all the collected information will be summed up here in this particular the cell body.

And this collected information from the cell body, it will pass through the action and this is known as axon hillock. Now through this axon hillock, so this will enter the action and this region is known as the synapse region. Now in the synapse region actually so, this is connected to the dendrites of the neighboring neuron. So, here I have got another neuron and these dendrites are going to be in contact with this particular the synapse zone. So, this neuron is going to collect information from the neighboring neuron with the help of dendrites. Those collected information will be there in the cell body and it will pass to the action to the end; that is the synapse and through that and dendrites, so it will go to the next neuron.

And in fact, here there will be some change in ion concentration of sodium ion potassium ion concentration, and due to this particular the change in ion concentration the information the collected information is going to pass through the next neuron. So, this is the way the information is processed from one neuron to the next neuron ah. In biological nervous system the same thing has been copied here in the artificial way now as I told that one artificial neuron. So, that is represented with the help of a particular the circle. Now here if you see, so all such inputs are coming through this particular the artificial neuron and here inside the artificial neuron. So, we have got the summing junction and we have got a transfer function.

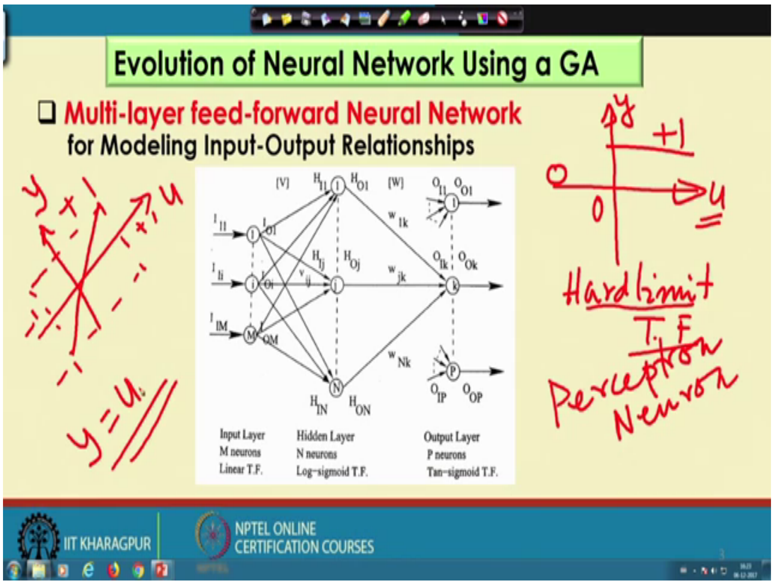
So, this transfer function is going to help here just to pass the information through the synapse from one neuron to the next neuron. So, here we have got the transfer function and I will be getting some output here. So, the moment I pass the inputs, the set of inputs it will pass through the summing junction and the transfer function, and here I will be getting this particular the output. So, the working principle of this biological neuron has been copied in the artificial way in this particular the artificial neuron. Now if I concentrate on a particular layer say input layer. So, I have got a large number of neurons here. For example, here I have considered say 1 to up to m. So, there are capital M number of neurons lying on this particular the input layer.

Now, similarly on the hidden layer. So, we have got the capital N number of neurons lying on this particular, the hidden layer and on the output layer. So, we have got capital P number of neurons, and as I told that here in this particular multi layered feed forward network. So, we have got only 3 layers input layer hidden layer and output layer, and as I told that at each of these particular the your artificial neuron, we have got the summing junction and we have got the transfer function.

Now, let us see the different types of transfer function which we generally use in this five type of network, and let us try to see the different types of your; this transfer function and before I go for this particular transfer function. So, let me tell that we use some connecting words between the input layer and this particular hidden layer, and those connecting wires are denoted by the V matrix. Similarly the connecting wires between the hidden layer and this output layer are denoted by, so this particular the W matrix.

Now, let us see, like what are the different types of the transfer function which we generally use. Now the transfer function which is generally used in this particular network are of different types. So, I am just going to discuss the different types of transfer function which we generally use now here. Now let me see this particular the transfer function which we generally use. For example, say we use the hard limit transfer function. So, let me see the principal of this particular the hard limit transfer function supposing that, so this is the input and the output is denoted by y.

(Refer Slide Time: 20:18)



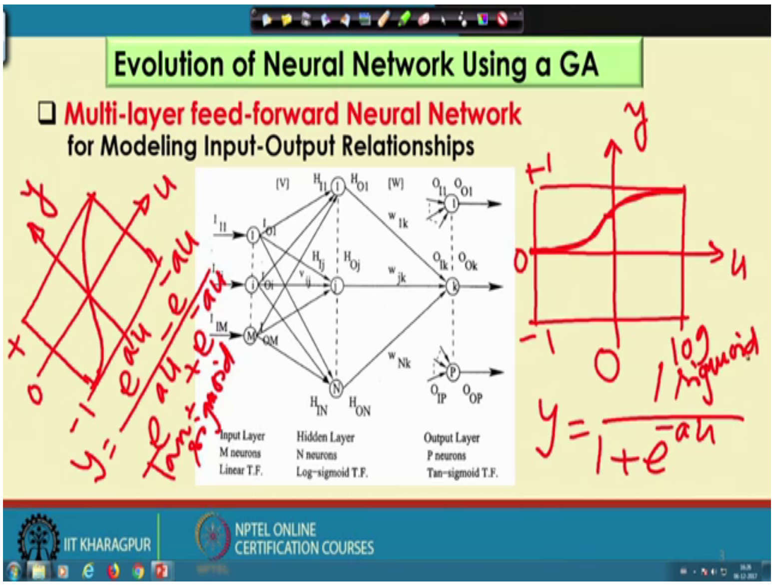

Now this particular the hard limit transfer function is such, the hard limit transfer function is such that if u the input, if it is found to be greater than 0 then the output will be plus 1 and if it is less than 0 less than equal to 0. So, the output will be actually your, the 0.

So, here the output is either 0 or 1, and if the input u is greater than equal to 0; so, output will be 1; otherwise the output will be 0. So, there are two outputs, either it is 1 or 0. So, it is some sort of ON, an OP type of a network sort of thing and here actually this type of hard limit transfer function we generally use for perceptron neuron. So, for this perceptron neuron we generally use. So, these types of perceptron neuron, we generally use this type of hard limit transfer function. So, this is actually the hard limit transfer function. Similarly sometimes we use some set of linear transfer function.

For example say this is the input u and supposing the output is denoted by y and here the output is nothing, but the input, and here I will be getting actually a line which is 45 degree line sort of thing. So, this is a plus 1 and this is minus 1 and this is plus 1 and this is minus 1. So, I will be getting a 45 degree line; that is y equals to u. So, this is nothing, but a linear transfer function. So, these are the hard limit transfer function output is either 1 or 0 and for this particular linear transfer function. So, output is nothing, but equal to the input.

Now, I am trying to find out there are some non-linear transfer function, so which is generally used very frequently. So, I am just going to take the example of this type of non-linear transfer function. For example, say I have got this log sigmoid transfer function.

(Refer Slide Time: 22:44)

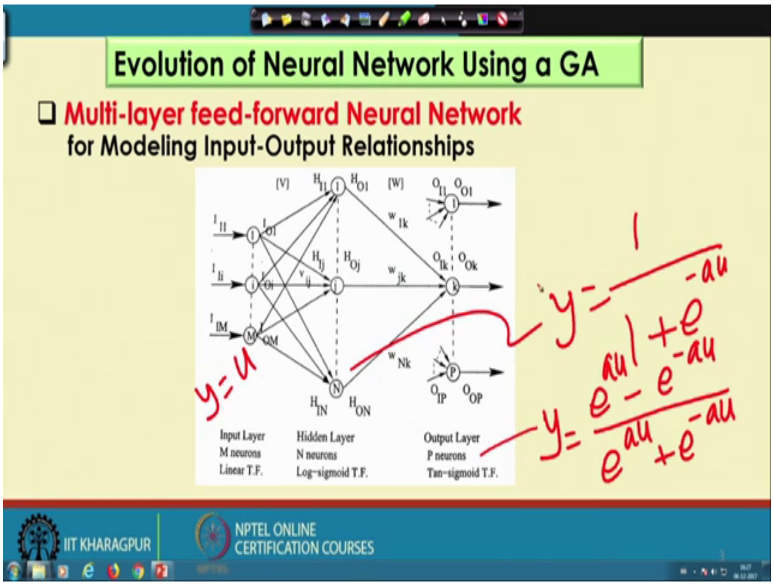



Now, if I plot the log sigmoid transfer function. So, this is say u and this is y; say u equals to 0 for u equals to 0, the value of y will be 0.5. So, I will be getting, I will be getting this type of plot and the here y is equals to plus 1 and here it is 0, and here it is minus 1. So, this is nothing, but your, the log sigmoid transfer function and mathematically this can be represented as y equals to 1 divided by 1 plus e raised to the power minus au. So, this is nothing, but the mathematical expression for this type of the log sigmoid transfer function.

Now, if I see the output of this particular log sigmoid transfer function. So, output will be either a 0 or 1 or in between. So, this will not go to the negative side, because this is the log sigmoid. Now similarly if I see the tan sigmoid transfer function. So, this is actually the tan sigmoid transfer function. So, this is the input u and output is y and here if I see the plot. So, the plot will be something like this. So, this is actually the tan sigmoid transfer function. So, this is minus 1 0 plus 1 and this is actually the positive direction of u, and if you see the mathematical formulation. So, y is nothing, but. So, e raised to the power au minus e raised to the power minus au divided by e raised to the power au plus e raised to the power minus au.

So, using this particular mathematical expression. So, you can represent. So, this type of your, the tan sigmoid transfer function. So, this is nothing, but the tan sigmoid transfer function and this is your the log sigmoid transfer function, log sigmoid transfer function. Now this transfer function is actually used just to find out the output corresponding to the input and generally we use either linear or non-linear transfer function. Now if you see the literature we use some other type of non-linear transfer function. For example, we can use some set of polynomial. Now, here actually what we do is. So, we are going to use some special type of transfer function.

(Refer Slide Time: 25:57)




Here for example, say on this, your the input layer we are generally using some separate linear transfer function. So, this is here we have got the linear transfer function; like your output is nothing, but the input that is like y is equals to u. Now similarly in the hidden layer we are generally used ah. We generally use this type of log sigmoid transfer function where this is nothing, but the expression I have already given like y is nothing, but 1 divided by 1 1 plus e raised to the power actually minus au.

So, this is nothing, but the log sigmoid transfer function and on this output layer. I am just going to use the tan sigmoid transfer function and for this tan sigmoid transfer function, the expression I have already written that is e raised to the power au minus e raised to the power minus au divided by e raised to the power au plus e raised to the power minus au . So, this is the way actually we can select this transfer function, and once we have selected the transfer function and once we have generated disconnecting wires v and w.

Now, let us see like if I pass one set of inputs. So, how to get this particular the output? Now first we will have to find out how to get the particular output for a set of inputs and those things we are just going to put inside the genetic algorithm so that the genetic algorithm can evolve one network through a large number of iterations.

Thank you.