فصل پانزدهم

سیستم های فازی ، شبکه های عصبی و آلگوریتم های ژنتیک

Fuzzy Systems ,

Fusion with Neural Networks and Genetic Algorithms

SECTION 1:
FUZZY SYSTEMS AND NEURAL NETWORKS

Introduction

- In this chapter, we study the fusion of fuzzy systems and neural networks.
- The two methods are complementary.
- The neural networks can learn from data while the fuzzy systems can not;
- The fuzzy systems are easy to comprehend because they use linguistic terms but the neural networks are not.
- Many researches have been devoted to the fusion of them in order to take their advantages.

Basic Concepts of Neural Networks

- Neural networks(NN) are a computational model of the operation of human brain.
- Neural network (NN), is a class of adaptive systems consisting of a number of simple processing elements, called neurons, that are interconnected to each other in a feed-forward way.
- A neural network is composed of a number of nodes connected by links. Each link has a numeric weight associated with it.
- Weights are the primary means of long-term storage in neural networks.

Basic Concepts of Neural Networks

- An important contribution of NN is the ability to learn to perform operations, not only for inputs exactly like the training data, but also for new data that may be incomplete or noisy.
- Learning usually takes place by updating the weights.
- NN has also the benefit of easy modification by retraining with an updated data set.
- For our purpose, the significant advantage of NN is the speed of operation after it is trained.

Basic Concepts of Neural Networks

The artificial neuron simulates the behavior of the biological neuron to make a simple operation of a weighted sum of the incoming signals as

$$y = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n \qquad (3.1)$$

where $x_1, x_2, \cdots, x_n$ are inputs, $w_0, w_1, w_2, \cdots, w_n$ are weights, and y is output. Figure 3.1 illustrates a neuron.

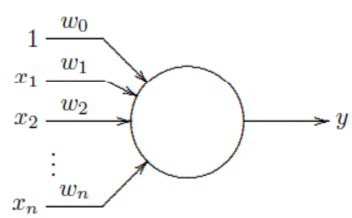


Figure 3.1: An Artificial Neuron

shavandi@sharif.edu

Basic Concepts of Neural Networks

- Neural networks can be categorized into feed-forward and feedback.
- The feed-forward neural networks have only feed-forward links, i.e. neural networks which do not have feedback cycle. The output of a node will not directly or indirectly be used as an input of that node.
- In the case of the feedback neural network, there is no guarantee that the networks become stable because of the feedback cycle.

Fuzzy Sets. Chapter 6- Fuzzy Logic
shavandi@sharif.edu

Basic Concepts of Neural Networks

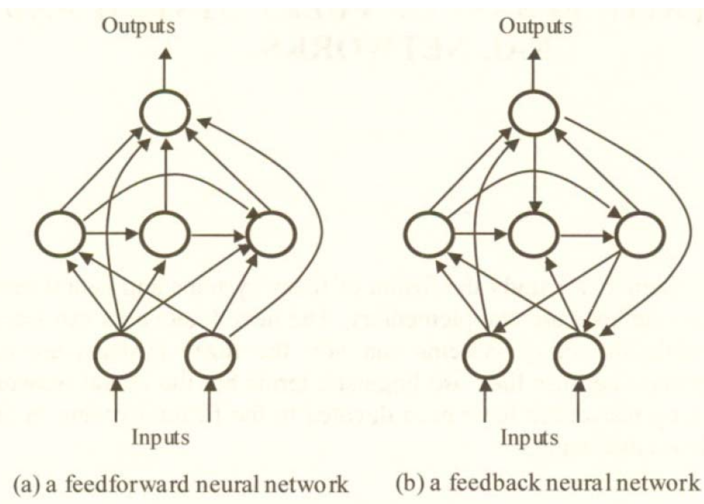


Fig. 11.2. Neural Networks

Sharif University of Technology
Industrial Engineering Dept.

Multilayer Perceptrons and Error Back-propagation Learning

- Among the neural networks and learning algorithms, multilayer perceptron network and its learning algorithm are widely used.
- This learning algorithm is called error back-propagation method. Multilayer perceptrons are layered feedforward neural networks.
- They consist of several layers. A layer is a set of nodes which do not have inter-connection links, i.e. the nodes in the same layer are not connected to each other.
- One more characteristic is that the nodes in a layer are connected to only the nodes in the neighboring layer. (Fig 11.3) shows a three layer perceptrons network. The first layer is the input layer, the second is the hidden layer and the third is the output layer.

Multilayer Perceptrons and Error Back-propagation Learning

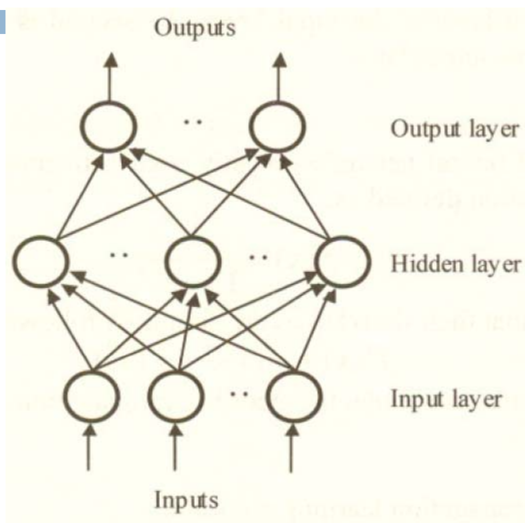


Fig. 11.3. Multilayer perceptrons

Fusion with Neural Networks

- Neural networks and fuzzy systems are two complementary technologies.
- Neural networks can learn from data, but the knowledge represented by the neural networks is difficult to understand.
- In contrast, fuzzy systems are easy to comprehend because they use linguistic terms and if-then rules, but it does not have learning algorithms.
- So, many researches have been devoted to fusion of them.

Fusion with Neural Networks

- The researches on fusion of neural networks and fuzzy systems can be classified into four categories:
- (I) Modifying fuzzy systems with supervised neural network learning,
- (2) Building neural networks using fuzzy systems,
- (3) Making membership functions with neural networks,
- (4) Concatenating neural networks and fuzzy systems.

Neuro Fuzzy Systems

Modifying Fuzzy Systems with Neural Network

- Research in this category represents fuzzy systems with neural networks.
- These systems are called neuro fuzzy systems, and the neural networks are used to improve the performance of fuzzy systems.
- Neuro fuzzy systems have learning capability like neural networks and can perform inference like fuzzy systems.
- In the ordinary neural networks, nodes have the same functionality and are fully connected to the nodes in the neighboring layers.
- But in a neuro fuzzy system, nodes have different functionalities and are not fully connected to the nodes in the neighboring layers

Neuro Fuzzy Systems

- This differences come from the fact that the nodes and links in a neuro fuzzy system usually correspond to a specific component in a fuzzy system.
- That is, some nodes represent the linguistic terms of input variables, some nodes are for those of output variables, and some nodes and links are used for representing fuzzy rules.

For example, the neuro fuzzy system proposed by Kwak, Lee and Lee-Kwang consists of five layers as shown in (Fig 11.4.). In the followings, the function of a node *f* is presented. Function f_j^k represents the *f* of node *j* of layer *k*.

Neuro Fuzzy Systems

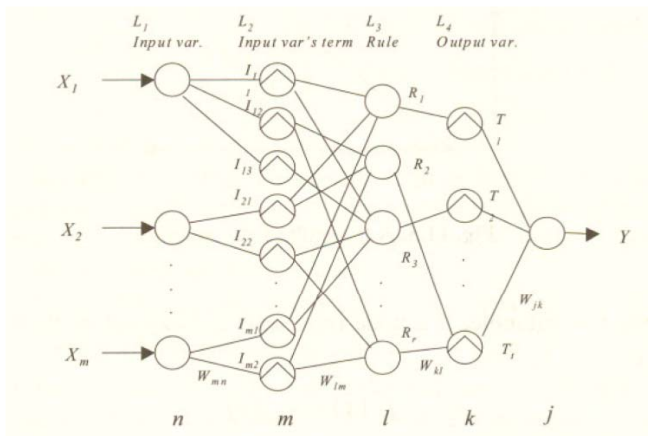


Fig. 11.4. Structure of the neuro fuzzy system proposed by Kwak

Neuro Fuzzy Systems

(1) First layer of the network(Inputs)
The nodes in the first layer takes inputs and just pass them to the second layer.

$$f_j^1(x) = x$$

(2) Second layer of the network(Input linguistic terms)
A node in this layer represents a linguistic term of an input variable. It has parameters which represent the membership function of linguistic term. For example, in (Fig 11.4.). the input variable X1 is connected to three nodes in the second layer and X2 is connected to two nodes. It means that there are three linguistic terms defined on X1 and two linguistic terms on X2. The node in the second layer outputs the members degree of input.

Neuro Fuzzy Systems

For example, if a node represents a fuzzy set A, its output is $\mu_A(x)$.

$$f_j^2(x) = \mu_{A_j}(x)$$

where A_j is the fuzzy set represented by the node.

Neuro Fuzzy Systems

(3) Third layer of the network(Antecedent parts)

This layer corresponds to the antecedent parts of fuzzy rules. For example, in Fig. 11. the inputs of R_1 are the outputs of I_{11}, I_{21} and I_{m1}. It represents the antecedent part of the rule such as :

if X_1 is I_{11} and X_2 is I_{21} and $\cdots$ and X_m is I_{m1} then "

The output of the node is the matching degree of given inputs to the antecedent part. When evaluating the matching degrees, the minimum or product operators can be used. The connection weights between the second and third layers are fixed to 1.0.

$$f_i^3(x_1, x_2, \cdots, x_p) = \begin{cases} \min_{i=1}^{p}(x_i) & \text{if minimum used} \\ \prod_{i=1}^{p}(x_i) & \text{if product used} \end{cases}$$

Neuro Fuzzy Systems

(4) Fourth layer of the network(Consequent parts)

This layer represents the consequent parts of fuzzy rules. Like in the second layer, a node in this layer represents a linguistic term of output variable. For example, node T_t has two inputs from R_2 and R_r. It represents two rules whose consequent part is T_t:

$$\text{"if the antecedent part is } R_2 \text{ then } Y \text{ is } T_t\text{"}$$

$$\text{and "if the antecedent part is } R_r \text{ then } Y \text{ is } T_t\text{"}$$

The output of the node is the maximum matching degree of an input to the rules which are represented by the node. For example, the output of the node T_t is the maximum output of nodes R_2 and R_r. The weights between the third and fourth layers are used as the importance degree of rules, or fixed to 1.00.

$$f_j^4(x_1, x_2, \cdots, x_q) = \max_{i=1}^{q}\{w_{ji}x_i\}$$

where w_{ji} is the weight between node j in the fourth layer and node i in the third.

Neuro Fuzzy Systems

(5) Fifth layer of the network(Defuzzification)

A node in this layer gathers the outputs of all rules and defuzzifies them. A defuzzification method similar to the center of gravity method is used. The weight of links between it and the fourth layer is 1.00.

(6) Learning algorithm

The learning algorithm of this model is based on the error backpropagation. During the learning process, the weights between the third and fourth layers, and the parameters representing membership functions in the nodes of the second and fourth layers are modified based on the error backpropagation method.

Building neural networks using fuzzy systems

- In the neuro fuzzy systems, neural networks were used to improve the performance of fuzzy systems.
- But the methods in this category use fuzzy system or fuzzy-rule structure to design neural networks.
- This model is a kind of divide and conquer approaches. Instead of training a neural network for the whole given input-output data, this model builds several networks:
- (1) Builds a fuzzy classifier which clusters the given input-output data into several classes,
- (2) Builds a neural network per class,
- (3) Trains the neural networks with the input-output data in the corresponding class.

Building neural networks using fuzzy systems

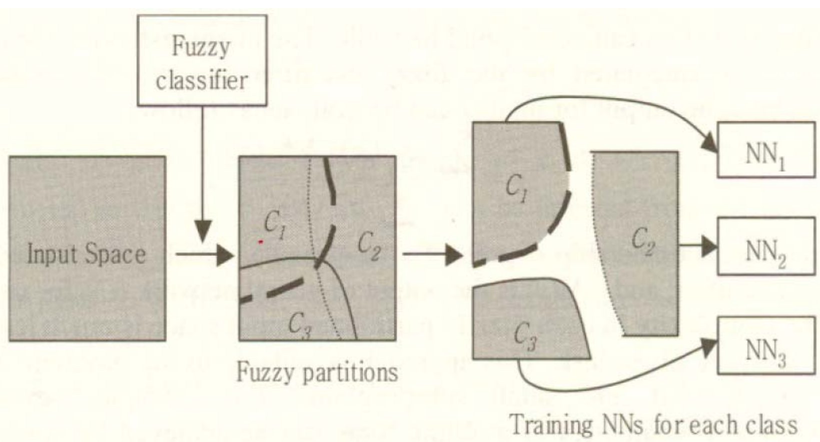


Fig. 11.9. Building neural networks using fuzzy systems

Building neural networks using fuzzy systems

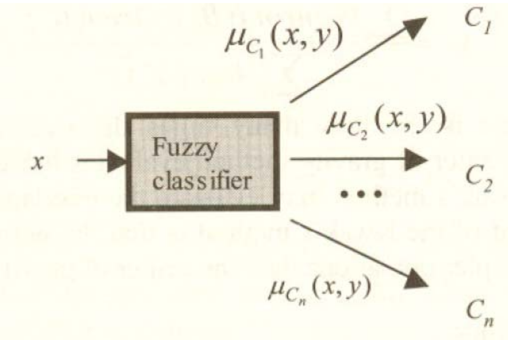


Fig. 11.9. A fuzzy classifier

Building neural networks using fuzzy systems

- Thus, we can say that this model has the following fuzzy rules:

$$\begin{aligned}
&\textbf{if}\ x\ \text{is}\ C_1,\ \textbf{then}\ \ y = NN_1(x)\\
&\textbf{if}\ x\ \text{is}\ C_2,\ \textbf{then}\ \ y = NN_2(x)\\
&\qquad\qquad\cdots\\
&\textbf{if}\ x\ \text{is}\ C_n,\ \textbf{then}\ \ y = NN_n(x)
\end{aligned}$$

Concatenating Neural Networks and Fuzzy Systems

- This category includes the methods equally using fuzzy systems and neural networks to improve system performance.
- **(1) Parallel combination**
- This combination is for correction of the output of a fuzzy system with the output of a neural network to increase the precision of the final system output. Fig. II.15(a) shows this combination.
- If a fuzzy system exists and an input-output data set is available, this model can be used to improve the performance without modifying the existing fuzzy systems

Concatenating Neural Networks and Fuzzy Systems

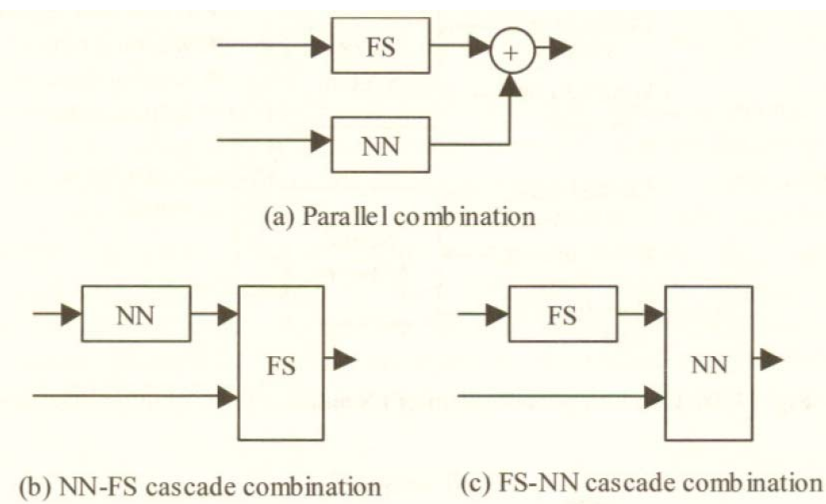


Fig. 11.15. Possible concatenations of NNs and FSs

Concatenating Neural Networks and Fuzzy Systems

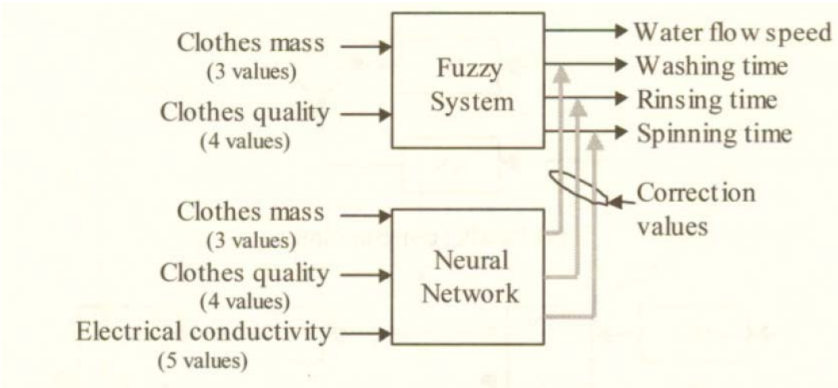


Fig. 11.16. Correcting mechanism of FS and NN for a washing machine

SECTION 2: FUZZY SYSTEMS AND GENETIC ALGORITHMS

shavandi@sharif.edu

Genetic Algorithms

- (Genetic algorithms can be viewed as a general-purpose search method, an optimization method, or a learning mechanism.
- Their basic mechanisms are similar to Darwinian principles of biological evolution: reproduction and "survival of the fittest".
- Genetic algorithms maintain a set of candidate solutions. The set is called a population and candidate solutions are called individuals or chromosomes.
- Chromosomes are usually represented in binary strings of a fixed length.
- The genetic algorithms have been shown to be an effective search technique on a wide range of difficult optimization problems.

shavandi@sharif.edu

Operations in genetic algorithms

- 1) Initialize a population of chromosomes (population size = *n*).
- 2) Evaluate the fitness of each chromosome in the population.
- 3) If the stop condition is satisfied, stop and return the best chromosome in the population
- 4) Select *n*/2 *pairs of chromosomes from the population*. Chromosomes can be selected several times.
- 5) Create new *n chromosomes by mating the selected pairs by* applying the crossover operator.
- 6) Apply the mutation operator to the new chromosomes.
- 7) Replace the old population with the new chromosomes.
- 8) Goto (2).

Evolution of Genetic Algorithms

- **(1) Fitness function and evaluation**
- We need a function to evaluate the fitness of candidate solutions during the operations.
- The fitness of chromosomes are the driving force of genetic algorithms.
- It represents how much good solution a chromosome is. Thus, the fitness function should be designed so that it gives higher fitness values to better solutions.
- If there is a function to be optimized, it is usually used as a fitness function.

Evolution of Genetic Algorithms

- **(2) Selection**
- Selection is an operation which prepares reproductions.
- The selected chromosomes are called parents.
- For the selection, first the possibility for each chromosome to be selected is evaluated.
- This possibility largely depends on the fitness value; the higher fitness value, the higher selection possibility.
- The reason is that it is expected that better offspring can be generated from better parents.
- There are several selection methods.

Evolution of Genetic Algorithms

- **(2) Selection Methods: Roulette Wheel Selection**

1) Roulette wheel selection
The roulette wheel selection is a typical one. The selection probability of a chromosome is the ratio of its fitness value to the sum of those of all chromosomes. That is, this method gives the selection probability to individuals linearly proportional to their fitness values. For example, there are five chromosomes, I_1, I_2, I_3, I_4 and I_5, and their fitness values are 1, 4, 3, 6, and 2, respectively. The summation of fitness is 16. Thus, the selection probability of I_1 is 1/16=0.0625, I_2 is 4/16=0.25, I_3 is 3/16=0.1875, I_4 is 6/16=0.375 and I_5 is 2/16=0.125. Each chromosome will be selected based on these probabilities.

Fuzzy Sets. Chapter 6- Fuzzy Logic shavandi@sharif.edu

Evolution of Genetic Algorithms

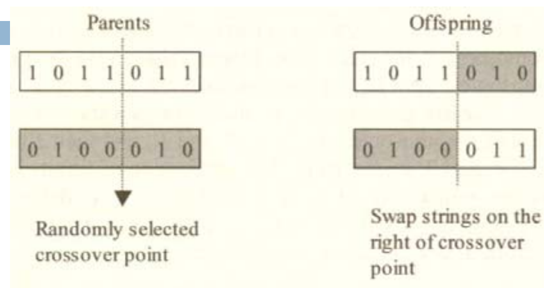
- **(3) Crossover**
- Crossover operators produce two new chromosomes by exchanging information of the selected chromosomes.
- The most typical crossover operator is the one-point crossover. The selected chromosomes are cut on the randomly chosen point, and the cut parts are exchanged. It is shown in (Fig 12.1(a)).
- An extension is the multi-points crossover in which several points are chosen. (Fig 12.1(b)) shows the two points crossover.
- The crossover operations are not performed on every selected chromosome. Genetic algorithm decides, based on a given probability, whether it performs the crossover operation on the certain pair of chromosomes or not. It is called the crossover probability and given by users.

Sharif University of Technology Industrial Engineering Dept.

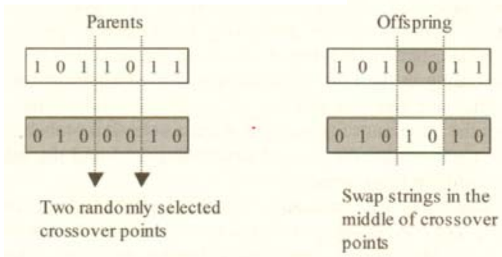
shavandi@sharif.edu

Evolution of Genetic Algorithms

- **(3) Crossover**



(a) one point crossover



(b) two point crossover

shavandi@sharif.edu

Evolution of Genetic Algorithms

- **(4) Mutation**
- Mutation operators change some randomly selected bits of chromosomes.
- If the chromosomes are binary strings, then '0' are changed to '1', and '1' to '0'.
- It plays a secondary role after the crossover operator in genetic algorithms.
- The changing bits means making an offspring genetically different from its parents.
- Since the crossover operator mixes the information of only two parents, the information of off springs may not be much different from that of its parents.
- Thus, applying only the crossover operator may make a population trapped in a local optimum.

Evolution of Genetic Algorithms

- **(4) Mutation**

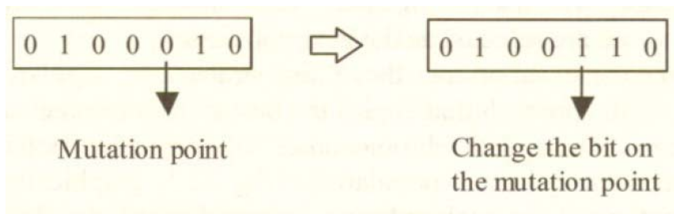


Fig. 12.2. Mutation operator

Fuzzy Sets. Chapter 6- Fuzzy Logic
shavandi@sharif.edu

Evolution of Genetic Algorithms

- **(5) Replacement**
- A typical genetic algorithm totally replaces the old population with the newly created chromosomes, but it is not mandatory.
- There could be many variations. For example, after reproduction, the old and new populations are taken together, and among them the best n chromosomes are selected as the next population.
- Among these variations, the elitist strategy is popular. The elitist strategy is an approach that copies the best k chromosomes into the next population.
- The other chromosomes of the new populations are reproduced from the old population. (Fig 12.3) graphically shows the elitist strategy.
- If the total replacement is performed, the chromosome of the best fitness in the new population may be worse than that in the old population. This is why the elitist strategy is useful.

Sharif University of Technology
Industrial Engineering Dept.

Evolution of Genetic Algorithms

- **(5) Replacement**

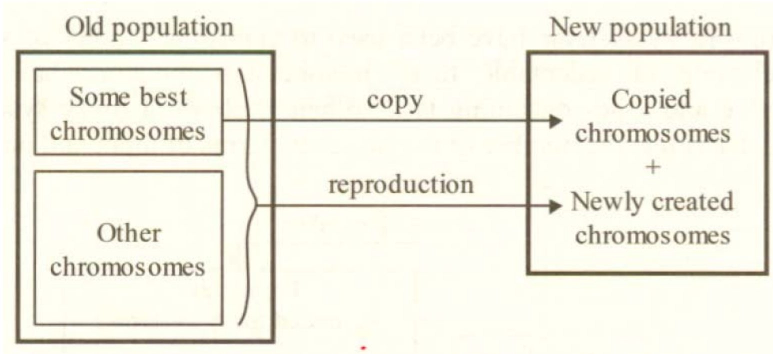


Fig. 12.3. Elitist strategy

Fusion with Genetic Algorithms

- Like in the fuzzy systems and neural networks, the fuzzy systems and genetic algorithms can complement each other.
- Researches on their fusion of them can be classified into two categories:
- (I) Identifying fuzzy systems with genetic algorithms
- (2) Controlling parameters of genetic algorithms with fuzzy systems.
- As mentioned, the fuzzy systems do not have learning algorithms, so the genetic algorithms can be used as a learning algorithm of the fuzzy systems.
- Genetic algorithms have some parameters to be set, so fuzzy rules can be used to change these parameters during the searching process.

Identifying Fuzzy Systems with Genetic Algorithms

- Although fuzzy systems have been used to control a number of systems,
- the selection of acceptable fuzzy membership functions has been a subjective and time-consuming task.
- When we build a fuzzy system, We should determine the number of the linguistic terms of input and output variables, their membership functions and the consequence parts of fuzzy rules.

Identifying Fuzzy Systems with Genetic Algorithms

- The IF-THEN structure of fuzzy rules is easy to understand and to build with priori knowledge,
- But many parameters should be specified by experts.
- The identification of these parameters can be viewed as an optimization problem; finding parameters that optimize the performance of the model.
- Therefore, there have been many researches on applying the genetic algorithms to the identification of fuzzy systems.
- These researches encode the parameters of a fuzzy system into chromosomes, and these chromosomes are evolved to find parameters which make a fuzzy system fit to real systems or given data well.

Identifying Fuzzy Systems with Genetic Algorithms

- **(1) Tuning an existing fuzzy system**
- The researches in the first category modify the parameters of an existing fuzzy system.
- The usually tuned parameters are the membership functions and/or fuzzy rules.
- In these researches, the membership functions are encoded into chromosomes and better membership functions are searched by genetic algorithms.
- (Fig 12.5.) shows an example of encoded fuzzy sets into chromosomes.
- In this example, the center of triangular fuzzy set is fixed, and only the left and the right points of each fuzzy set are variable and thus encoded.

Identifying Fuzzy Systems with Genetic Algorithms

- **(1) Tuning an existing fuzzy system**

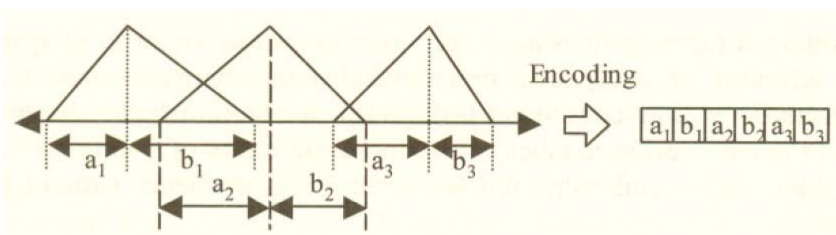


Fig. 12.5. Example of encoding fuzzy sets

Identifying Fuzzy Systems with Genetic Algorithms

- **(1) Tuning an existing fuzzy system**
- To modify the fuzzy rules, their consequent parts are usually encoded. For example, there are four fuzzy rules:

IF X is I_1 THEN Y is O_1
IF X is I_2 THEN Y is O_2
IF X is I_3 THEN Y is O_3
IF X is I_4 THEN Y is O_4

then, these are encoded as a string of linguistic terms like $O_1O_2O_3O_4$. The genetic operators will change the linguistic terms, but not their membership functions. For example $O_1O_2O_3O_4$ may be changed into $O_1O_3O_4O_1$ after genetic operations. This represents the following fuzzy rules:

IF X is I_1 THEN Y is O_1
IF X is I_2 THEN Y is O_3
IF X is I_3 THEN Y is O_4
IF X is I_4 THEN Y is O_1

Controlling Parameters of Genetic Algorithms with Fuzzy Systems

- For example, fuzzy rules for those systems can be described as follows:

- **IF** average fitness is high **THEN** population size should be increased.
- **IF** best fitness is not improved **THEN** mutation rate should be increased.

- One question of this approach is how to obtain the knowledge to build the fuzzy rules.
- It can be solved in the ways; an expert on genetic algorithms can describe his/her own knowledge or an automatic fuzzy design technique can be applied.

Controlling Parameters of Genetic Algorithms with Fuzzy Systems

- **Encoding scheme**

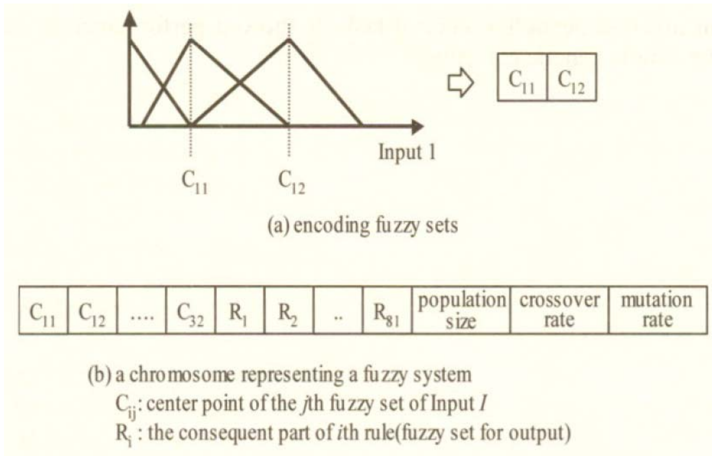


Fig. 12.8. Coding of a fuzzy set and a chromosome of the DPGA