Semester – VI

17CTU603A SOFT COMPUTING 3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0 **Marks:** Int : **40** Ext : **60** Total: **100**

SCOPE

Soft computing refers to principle components like fuzzy logic, neural networks and genetic algoritm which have their roots in Artificial Intelligence

Healthy integration of all these techniques has resulted in extending the capabilities of the technologies to more effective and efficient problem solving methodologies.

OBJECTIVES

Upon completion of the course, you should be able to

- Identify and describe soft computing technique and their roles in building intelligent machines
- Recognize the feasibility of applying a soft computing methodology for particular problem
- Apply fuzzy logic and reasoning to handle uncertainity and solve enginnering problems
- Apply genetic algorithms to combinatorial optimization problems.
- Apply neural networks to pattern classification and regression problems
- Effectively use existing software tools to solve real problems using a soft computing approach
- Evaluate and compare solution by various soft computing approaches for a given problem

UNIT I

Neural Networks – 1 (Introduction a Architecture)

Neuron, Nerve structure and synapse, Artificial Neuron and its model, activation functions, Neural network architecture: single layer and multilayer feed forward networks, recurrent metworks. Various learning techniques; perception and convergence rule, Auto associative and hetro-associative memory.

UNIT II

Neural Networks – II (Back propagation networks) architecture: perceptron model, solution, single layer artificial neural network, multilayer perception model, back propagation learning methods, effect of learning rule co-efficient; back propagation algorithm, factors affecting back propagation training, applications.

UNIT III

Fuzzy Logic – 1 (Introduction)

Basic concepts of fuzzy logic, Fuzzy sets and Crisp sets, Fuzzy set theory and operations, Properties of fuzzy sets, Fuzzy and Crisp realtions, Fuzzy to Crisp conversion

UNIT IV

Fuzzy Logic II (Fuzzy membership, Rules) Membership functions, interference in fuzzy logic, fuzzy if-then rules, Fuzzy implications and Fuzzy algorithms, Fuzzyfications & Defuzzifications, Fuzzy Controller, Industrial applications.

UNIT V

Genetic Algorithm (GA)

Basic concepts, working principle, procedures of GA, flowchart of GA, Genetic representations, (encoding) Initialization and selection, genetic operators, Mutation, Generational Cycle, applications

Suggested Readings

1. S.Rajasekaran & G.A. Vijayalakshmi Pai. Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis and Applications. Prentice Hall of India.
2. N.P.Padhy. Artificial Intelligence and Intelligent Systems. Oxford University Press.
3. Siman Haykin. Neural Networks. Prentice Hall of India
4. Timothy J.Ross. Fuzzy Logic with Engineering Applications. Wiley India.
5. Kumar Satish. Neural Networks. Tata McGraw Hill.

Karpagam Academy of Higher Education

(Deemed University Established Under Section 3 of UGC Act 1956)

Coimbatore – 641 021

Department of CS, CA and IT

SUBJECT NAME: SOFT COMPUTING

SUBJECT CODE: 16CTU603A **SEMESTER: VI**

STAFF NAME : Dr SIVAGAMINATHAN P.G

LECTURE PLAN

S.No.	Lecture Duration (Period)	Topics to be Covered	Support Materials
		Unit – I	
1.	1	Introduction – Neural Networks, Neuron, Synapse	T1: 11-13, R1:2-11
2.	1	Artificial Neuron and its Model	T1:13-16
3.	1	Activation Function, SLFF & MLFF networks	T1:15,17-18
4.	1	Learning Techniques, Convergence Rule	T1:19-22
5.	1	Auto-associative memory, Hetero associative memory	T1:87-88
6.	1	Recapitulation and Discussion of Important Questions	
		Total No. of Hours Planned for Unit-I	**6**
		Unit II	
1.	1	Neural Networks –II(BPN)-Introduction &Architecture	T1:35-38
2.	1	Perceptron Model,Single layer, ANN	T1:39-41
3.	1	Multilayer perceptron Model	T1:41-42
4.	1	Back propogation learning Methods	T1:42-49,W1
5.	1	Learning parameters, factors for BPN, Training and Applications	T1:51=-53
6.	1	Recapitulation and Discussion of Important Questions	
		Total No.of hours planned for Unit II	**6**
		Unit III	
1	1	Introduction –Fuzzy Logic, Basic Concepts	T1:157-158
2	1	Fuzzy sets, Crisp sets and Operations	T1:159-167
3	1	Properties of fuzzy sets	T1:176-179
4	1	Fuzzy set and Crisp Relations	T1:179-184
5	1	Fuzzy set to Crisp Conversion	T1:205-210
6	1	Recapitulation and Discussion of important questions	
		Total No.of Hours Planned for Unit III	**6**
		Unit IV	
1	1	Fuzzy logic- Membership functions	T1: 169-170
2	1	Inference in fuzzy logic,if then rules	T1:202-204
3	1	Fuzzy implications and algorithm	T1:198-201,w1
4	1	Fuzzyfication and Defuzzification	T1: 159,205-209

5	1	Fuzzy controller and Applications	T1:210-219
6	1	Recapitulation and Discussion of important questions	
		Total No.of hours Planned for Unit IV	**6**
		Unit V	
1	1	Genetic Algorithms – Basic Concepts	T1:225-230
2	1	Working Principle, Flowchart & Procedures	T1:230-232
3	1	Initialization, Selection, Mutation & Applications	T1:261-263,W1
4	1	Recapitulation and Discussion of important questions	
5	1	Recapitulation and Discussion of important questions	
6	1	Recapitulation and Discussion of important questions	
		Total No.of hours Planned for Unit V	**6**
		Total No.of Hours Planned for this Course	30 Hours

Suggested Readings

T1: S.Rajasekaran & G.A.Vijayalakshmi Pai, Neural Networks, Fuzzy logic and Genetic Algorithms:Synthesis and Applications, PHI

R1:Siman Haykin, Neural Networks, PHI

Website

W2:http:/en.wikipedia.org/wiki/backpropagation

W4:http://en.wikipedia.org/wiki/fuzzy_logic

W5: https://www.tutorialspoint.com/genetic-algorithms/genetic-algorithms-mutation.html

Journal

J1:Nishchal Sharma, Dr Chaman S.Chauhan, "Improving Heikin-Ashi transformation data learning in neural network using volume weight in stock market data", IJSRCSEIT, 2018

J2:Werner Kranth, Marc Mezard, "Learning algorithms with optimal stability in neural networks"

FACULTY **HOD**

Unit I- Introduction to Neural Networks

Neuron- Nerve Structure and synapse, Artificial Neuron and its model, activation function, Neural network architecture: Singlelayer, Multilayer feed forward networks, recurrent networks, various learning techniques; perception and convergence rule, Auto associative and hetero associative array.

Artificial intelligence (AI) is an area of computer science concerned with designing intelligent computer systems that is, systems that exhibit the characteristics that associate with intelligence in human behavior. Lotfi Zadeh [1], father of fuzzy logic, has classified computing as hard computing and soft computing [2]. The computations based on Boolean algebra and other crispy numerical computations are defined as hard computing, whereas fuzzy logic, neural network and probabilistic reasoning techniques, such as genetic algorithm and parts of learning theory are categorized as soft computing. Soft computing differs from conventional (hard) computing in that, unlike hard computing, it is tolerant of imprecision, uncertainty and partial truth. Soft computing is more analogous to thinking of human mind.

Hard computing methods are predominantly based on mathematical approaches & therefore demand a high degree of precision & accuracy in their requirements. But in most engineering problems, the input parameters cannot be determined with a high degree of precision & therefore, the best estimates of the parameters used for obtaining solutions to problems .soft computing techniques which have drawn their inherent characteristics from biological systems. Neural network are simplified model of the biological nervous system. Therefore it has drawn their motivation from the kind of computing performed by a human brain.

An NN, in general is a highly interconnected network of a large number of processing elements called neurons in an architecture developed by the brain. NN exhibit characteristics such as mapping capabilities or pattern association, generalization, robustness, fault tolerance, & parallel & high speed information processing. NN learned by examples. They can be trained with known examples of a problem to ‘acquire’ knowledge about it. Once appropriately trained, the network can be put to effective use in solving ‘unknown’ or ‘untrained’ instances of the problem. Neural networks adopt various learning mechanisms of which supervised learning & unsupervised leaning methods have turned out to be very popular. In supervised learning a ‘teacher’ is assumed to be present during learning process whereas in unsupervised learning there is no ‘teacher’. NN architectures have been broadly classified as single layer feedforward networks, multilayer feedforward networks & recurrent networks.

Some of the well known NN systems include backpropagation network perceptron, ADALINE, associative memory, adaptive resonance theory, self-organizing feature map, Boltzmann machine & Hopfield network. Neural networks have been successfully applied to problems in the fields of pattern recognition, image processing, data compression, forecasting, & optimization to quote a few.

Basic Concepts of Neural Networks

Neural networks, which are simplified models of the biological neuron system. Derives it's origin from human brain. It is a massively parallel distributed processing system made up of highly interconnected neural computing elements. The elements have the ability to learn and thereby acquire knowledge and make it available for use. NNs are simplified imitations of the central nervous system. Therefore it have been motivated by the kind of computing performed by the human brain. Neurons perform computations such as cognition, logical inference, pattern recognition and so on. Hence the technology, which has been built on a simplified imitation of computing by neurons of a brain, has been termed Artificial Neural Systems technology or Artificial Neural Networks or simply Neural Networks. Also neurons are referred to as neurodes, Processing Elements (PEs), and nodes.

Human Brain

The human brain is one of the most complicated things which, on the whole, has been poorly understood. However the concept of neurons as the fundamental constituent of the brain, attributed to Ramon Y.Cajal(1911), has made the study of its functioning comparatively easier. Brain has a highly complex, nonlinear & parallel computing. Brain contains about 10^{10} basic units called neurons. Each neuron in turn, is connected about 10^4 other neurons. A neuron is a small c that receives electro-chemical signals from its various sourcesand in turn responds by transmitting electrical impulses to other neurons. An average brain weighs about 1.5kg and an average neuron has a weight of 1.5 10^{-9} gms.

While some of the neurons perform input x_n and output operations. A neuron is composed of a nucleus a cell body known as *soma* shown in figure 2.Attached to the soma are long irregularly shaped filaments called *dentrites*. The dentrites behave as input channels,(i.e) all inputs from other neurons arrive through the dentrites. Another type of link attached to the soma is Axon. Unlike dendrites links, the axon is electrically active and serves as an output channel. The axon terminates in a specialized contact called synapse or synaptic junction. The synapse connects the axon with the dendritic links of another neuron. A single neuron can have many synaptic inputs and synaptic outputs.

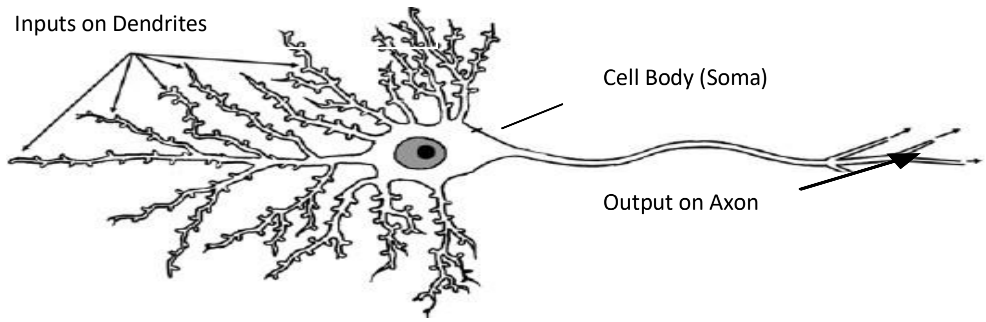
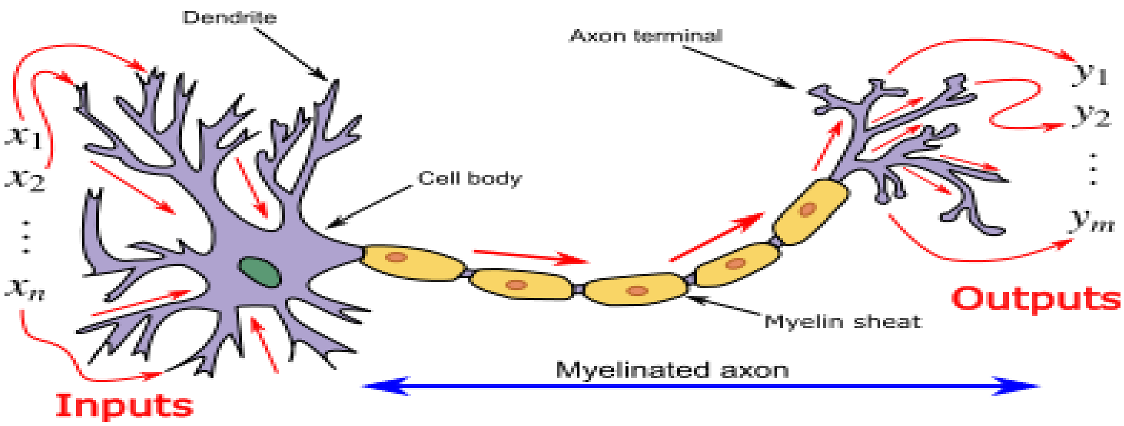


Fig.1 Structure of a neuron

Model of an Artificial Neuron

Artificial neural network motivated from biological analogy. The human brain is a highly complex structure. It is viewed as a massive, highly interconnected network of simple processing elements called neurons. The behavior of a neuron can be captured by a simple model as shown in figure 2.Every component of the model has a direct analogy to the actual component of a biological neuron and hence it is termed as artificial neuron.

What is a Neuron? A **neuron** (also called neurone or nerve cell) is a cell that carries electrical impulses. **Neurons** are the basic units of the nervous system and its most important part is the brain. Every **neuron** is made of a cell body (also called a soma), dendrites and an axon. Dendrites and axons are nerve fibres.



Type of Neurons?

For the spinal cord though, we can say that there are three types of neurons: sensory, motor, and interneurons.

Sensory neurons. ...

Motor neurons. ...

Interneurons. ...

Neurons in the brain.

Role of Neuron

Neuron. **Neurons** (also known as neurones, nerve cells and nerve fibers) are electrically excitable cells in the nervous system that **function** to process and transmit information. In vertebrate animals, **neurons** are the core components of the brain, spinal cord and peripheral nerves.

What are the functions of Neuron?

The neuron is also called nerve **cell** and is the fundamental unit of the **nervous system**.The function of neuron is to transmit electrical messages and signals throughout the **body**. There are three types of nerve cells. **Sensory** neurons collect and transmit information regarding stimuli like sound, light and temperature.

Synapse

In the central **nervous** system, a **synapse** is a small gap at the end of a neuron that allows a signal to pass from one neuron to the next. **Synapses** are found where **nerve** cells connect with other **nerve** cells.

Purpose of synapse

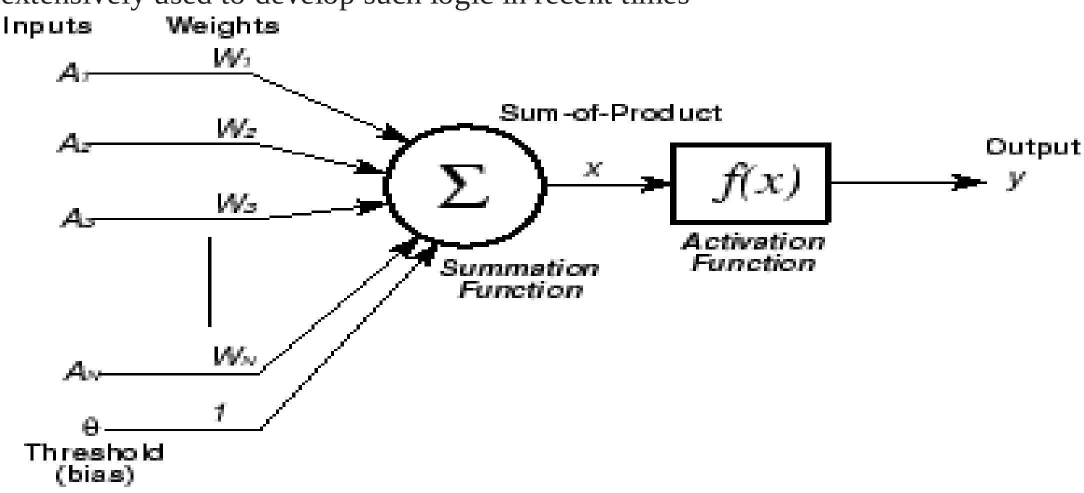
Synapses are the junctions between neurons in the nervous system. A neurotransmitter is released there - a chemical that allows one neuron to talk to the next neuron and continue sending the impulse. They make sure that the flow of impulses is in one direction only.

Define ANN - **ANN** is a soft-computing tool that can learn patterns and predicts. Learn more in: Biometric Identification System Using Neuro and Fuzzy Computational Approaches. An artificial neural network **defines** a mathematical model for the simulation of a network of biological neurons (e.g. human nervous system).

Artificial Neuron and its Model

A Simple **Artificial Neuron**. Our basic computational element (**model neuron**) is often called a node or unit. It receives input from some other units, or perhaps from an external source. Each input has an associated weight w, which can be modified so as to **model** synaptic learning.

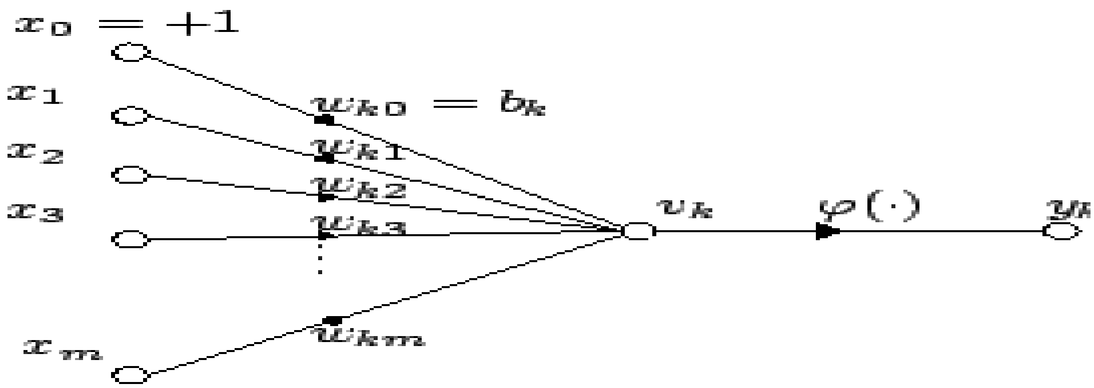
An **artificial neuron** is a mathematical function conceived as a model of biological neurons, a neural network. Artificial neurons are elementary units in an artificial neural network. The artificial neuron receives one or more inputs (representing excitatory postsynaptic potentials and inhibitory postsynaptic potentials at neural dendrites) and sums them to produce an output (or activation, representing a neuron's action potential which is transmitted along its axon). Usually each input is separately weighted, and the sum is passed through a non-linear function known as an activation function or transfer function[*clarification needed*]. The transfer functions usually have a sigmoid shape, but they may also take the form of other non-linear functions, piecewise linear functions, or step functions. They are also often monotonically increasing, continuous, differentiable and bounded. The thresholding function has inspired building logic gates referred to as threshold logic; applicable to building logic circuits resembling brain processing. For example, new devices such as memristors have been extensively used to develop such logic in recent times



For a given artificial neuron, let there be $m + 1$ inputs with signals x_0 through x_m and weights w_0 through w_m. Usually, the x_0 input is assigned the value +1, which makes it a *bias*input with $w_{k0} = b_k$. This leaves only m actual inputs to the neuron: from x_1 to x_m.

The output of the kth neuron is:

Where (phi) is the transfer function.



The output is analogous to the axon of a biological neuron, and its value propagates to the input of the next layer, through a synapse. It may also exit the system, possibly as part of an output vector.

It has no learning process as such. Its transfer function weights are calculated and threshold value are predetermined.

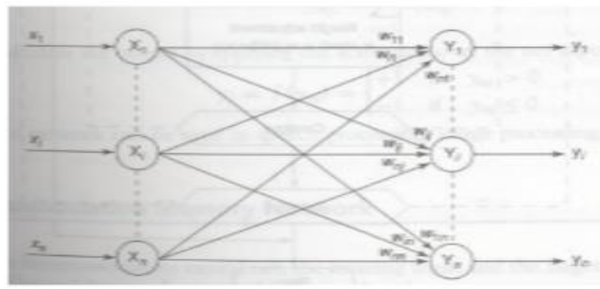
Artificial Neural Network

Artificial neural networks (ANNs) are biologically inspired computer programs designed to simulate the way in which the human brain processes information. ANNs gather their knowledge by detecting the patterns and relationships in data and learn (or are trained) through experience, not from programming. An ANN is formed from hundreds of single units, artificial neurons or processing elements (PE), connected with coefficients (weights), which constitute the neural structure and are organised in layers. The power of neural computations comes from connecting neurons in a network. Each PE has weighted inputs, transfer function and one output. The behavior of a neural network is determined by the transfer functions of its neurons, by the learning rule, and by the architecture itself. The weights are the adjustable parameters and, in that sense, a neural network is a parameterized system. The weighed sum of the inputs constitutes the activation of the neuron. The activation signal is passed through transfer function to produce a single output of the neuron. Transfer function introduces non-linearity to the network. During training, the inter-unit connections are optimized until the error in predictions is minimized and the network reaches the specified level of accuracy. Once the network is trained and tested it can be given new input information to predict the output. Many types of neural networks have been designed already and new ones are invented every week but all can be described by the transfer functions of their neurons, by the learning rule, and by the connection formula.

What is activation in machine learning?
A neural network without an **activation function** is essentially just a linear regression model. The **activation function** does the non-linear transformation to the input making it capable to learn and perform more complex tasks. ... Linear transformations would never be able to perform such tasks.
Why activation function is used in neural network?
An **activation function** is a decision making **function** that determines the presence of particular **neural** feature. ... Non-linearity is **needed** in **activation functions**because its aim in a **neural network** is to produce a nonlinear decision boundary via non-linear combinations of the weight and inputs.
What is perceptron?
Perceptron is a **single layer** neural network and a multi-layer **perceptron** is called Neural Networks. **Perceptron** is a linear classifier (binary). Also, it is used in supervised learning. It helps to classify the given input data.
Perceptron convergence rule?
The **Perceptron convergence** theorem states that for any data set which is linearly separable the**Perceptron** learning **rule** is guaranteed to find a solution in a finite number of steps.
PERCEPTRON CONVERGENCE THEOREM: Says that there if there is a weight vector w* such that f(w*p(q)) = t(q) for all q, then for any starting vector w, the perceptron learning rule will converge to a weight vector (not necessarily unique and not necessarily w*) that gives the correct response for all training patterns, and it will do so in a finite number of steps.
IDEA OF THE PROOF: The idea is to find upper and lower bounds on the length of the weight vector. If the length is finite, then the perceptron has converged, which also implies that the weights have changed a finite number of times.
Auto-Associative Memory Network: **Auto associative neural networks** were feed forward **nets** trained to produce an approximation of the identity mapping between **network** inputs and outputs using back propagation or similar learning procedures. The key feature of an **auto associative network** are a dimensional bottleneck between input and output.

AUTO ASSOCIATIVE MEMORY NETWORK

- **Theory**
- Training input and target output vectors are same.
- Determination of weight is called storing of vectors
- Weight is set to zero.
- Auto associative net with no self connection.
- Increases net ability to generalize.
- **Architecture**
- The Fig gives the architecture of Auto associative memory network.
- Input and target output vectors are same.
- The input vector has n inputs and output vector has n outputs.
- The input and output are connected through weighted connections.



5

What is meant by associative memory?

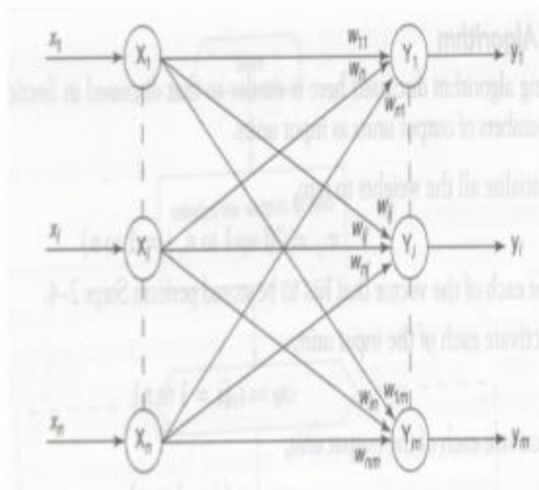
Associative memories can be implemented either by using feedforward or recurrent **neural networks**. ... The aim of an **associative memory** is, to produce the associated output pattern whenever one of the input pattern is applied to the **neural network**.

What is hetero associative memory?

Definition. A neural network that stores input–output pattern pairs to recall a stored output pattern by receiving a noisy or incomplete version of a stored input pattern paired with that output pattern. In each of the pairs, an input pattern should differ from an output pattern

HETERO ASSOCIATIVE MEMORY NETWORK

- **Theory**
- The training input and target output vectors are different.
- Determination of weight is by Hebb rule or Delta rule.
- Input has ‘n’ units and output has ‘m’ units and there is a weighted interconnection between input and output.
- **Architecture**
- The architecture is given in the Fig.



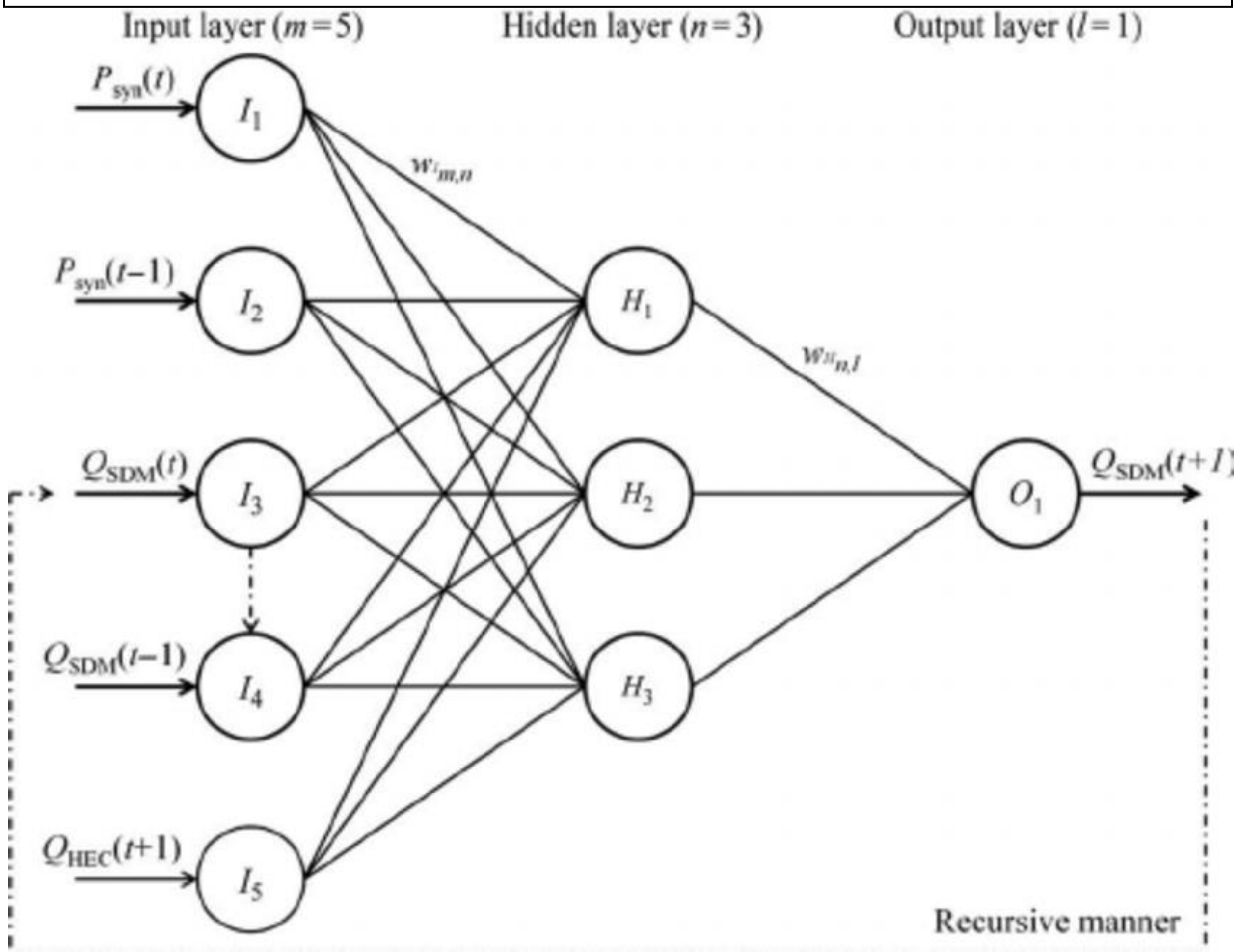
8

What is associative recall?

In psychology, **associative** memory is defined as the ability to learn and remember the relationship between unrelated items. ... A normal **associative** memory task involves testing participants on their **recall** of pairs of unrelated items, such as face-name pairs.

Unit II- Neural Networks -II

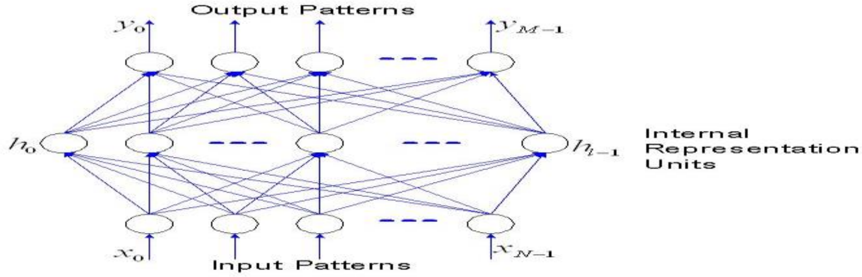
Neural Networks II (Back Propogation Networks) architecture: perceptron model, solution, single layer artificial neural network, multilayer perception model, back propagation learning methods, effect of learning rule co-efficient; back propagation algorithm, factors affecting back propagation training, applications.



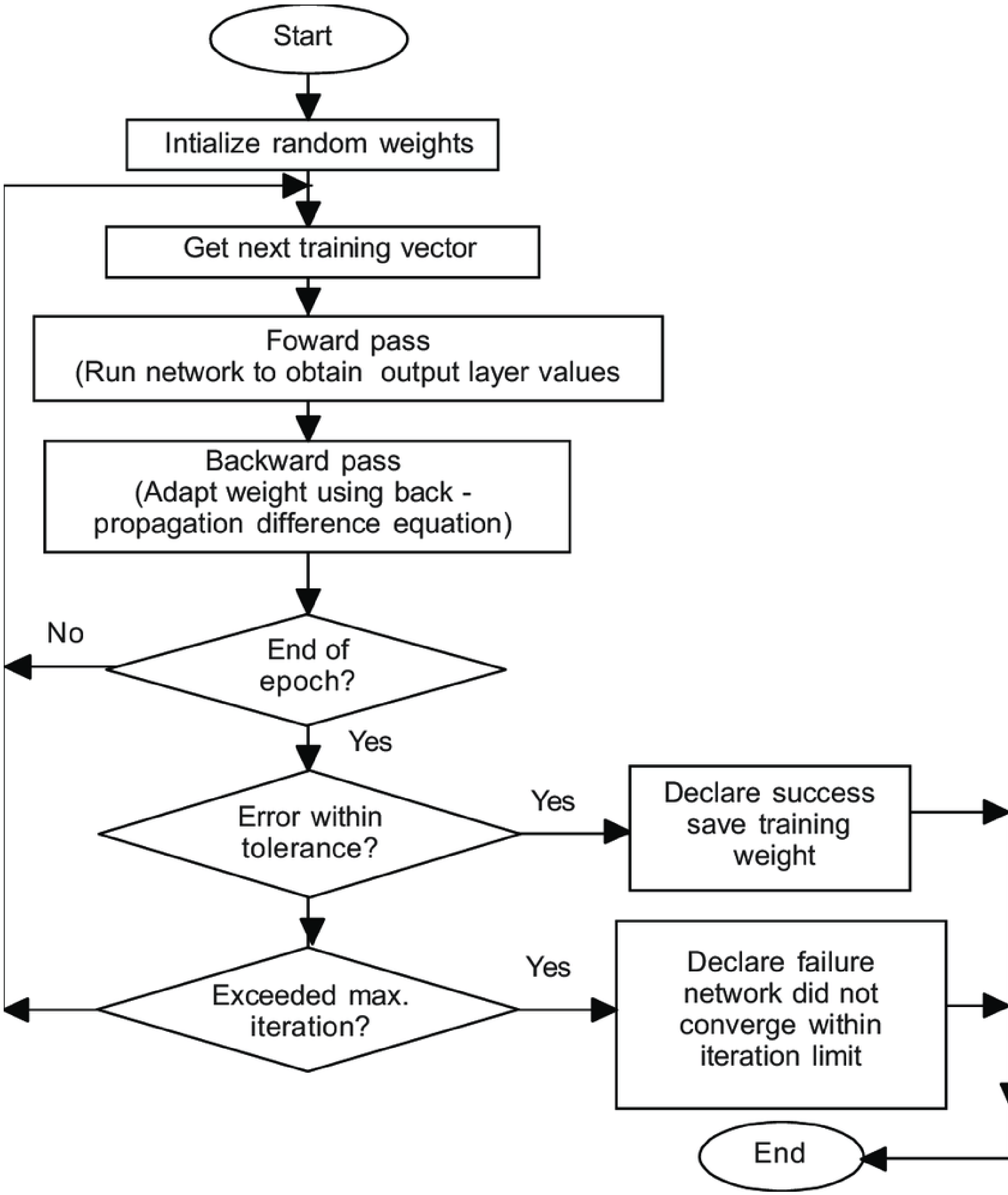
Perceptron model

Multi layered BPN

Multi-Layer Perceptron Model



BPNAlgortihm:

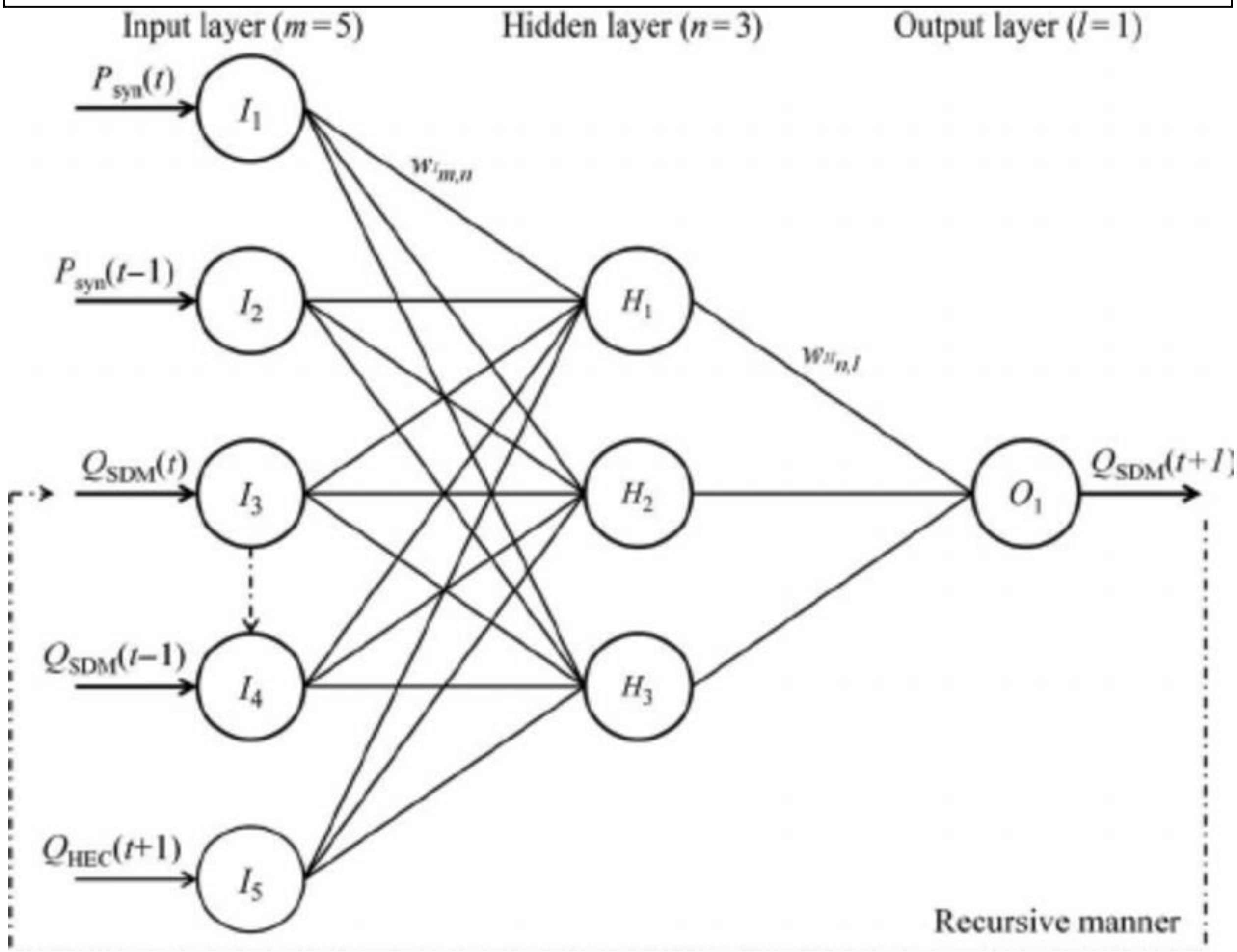


BPN Applications: Page (59-68)

1. Design of Journal Bearing
2. Classification of Soil
3. Hot extrusion of steel

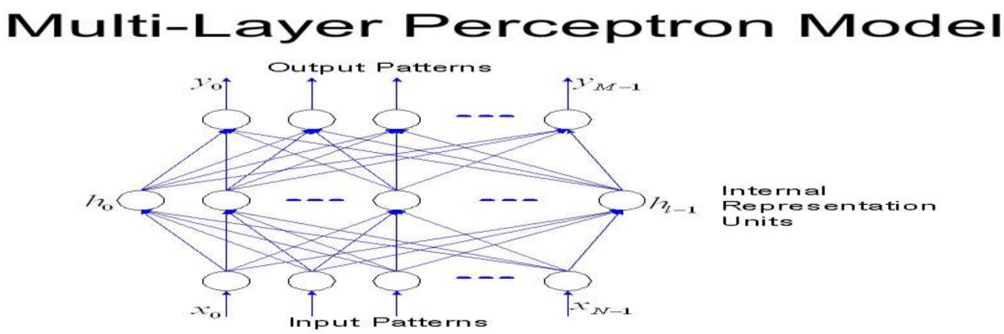
Unit II- Neural Networks -II

Neural Networks II (Back Propogation Networks) architecture: perceptron model, solution, single layer artificial neural network, multilayer perception model, back propagation learning methods, effect of learning rule co-efficient; back propagation algorithm, factors affecting back propagation training, applications.

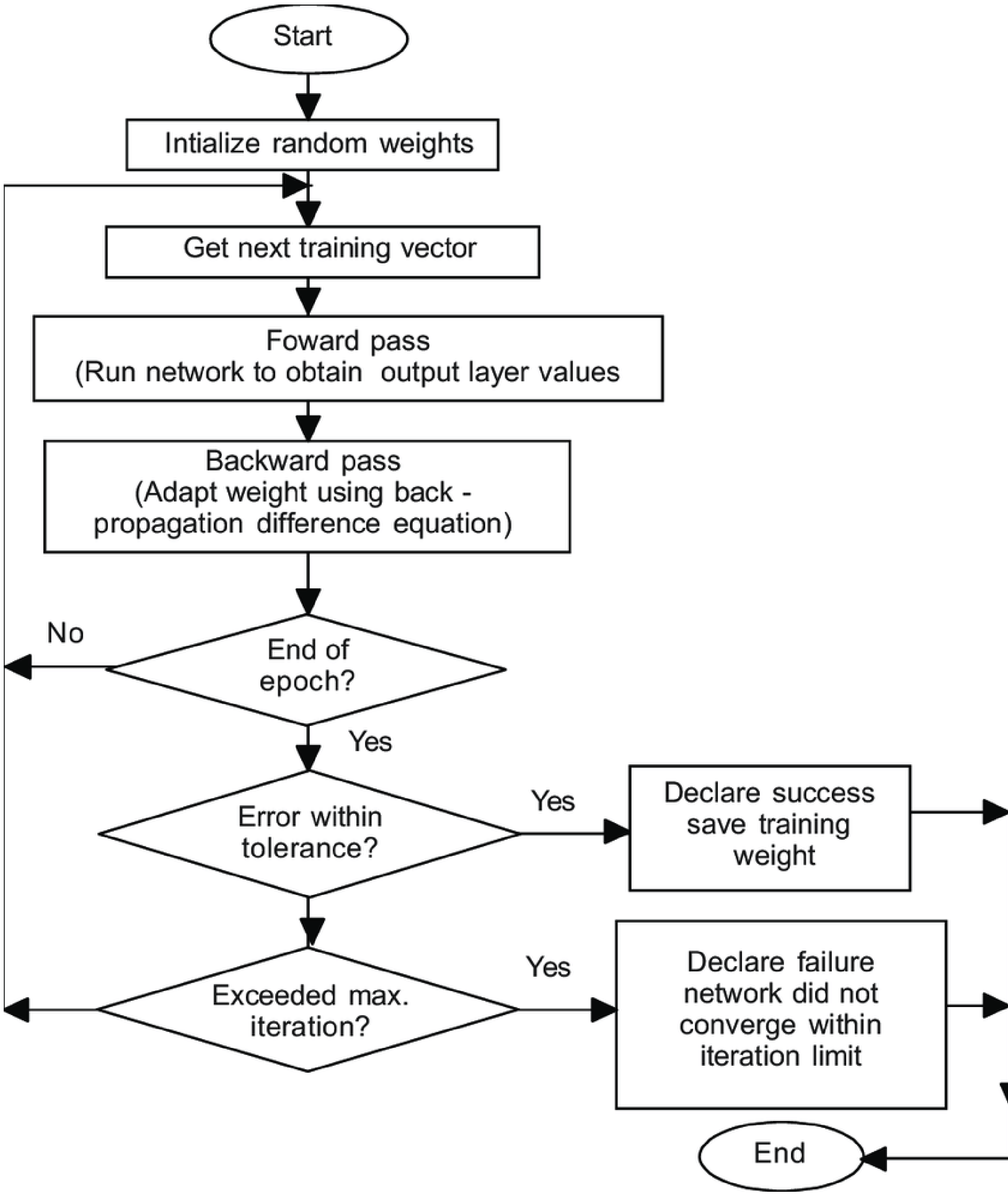


Perceptron model

Multi layered BPN


Multi-Layer Perceptron Model
Output Patterns
Internal Representation Units
Input Patterns

BPNAlgortihm:



BPN Applications: Page (59-68)

1. Design of Journal Bearing
2. Classification of Soil
3. Hot extrusion of steel

What is the objective of "back propagation alogorithm"
The backpropagation law is also known as generalized data rule is it true?
What is the true regarding back propagation rule?
Is there a feedback in final stage of backpropogation algorithm
What is true regarding back propagation rule?
What is meant by generalized in statement " backpropagation is a generalized data rule"?
What are general limitations of back propagation rule?
What are the general tasks that are performaed with back propagation algorithm?
Does propagation learning is based on gradient descent along error surface?
How can learning process be stopped in back propagation rule?
Which is the simplest pattern recognition task in a feedback network?
In a linear autoassociative network,if inputs is noisy than output will be noisy?
Does linear autoassociative network have any practical use?
What can be done by using non-linear output function for each processing unit in a feedback network?
When are stable states reached in energy landscapes, that can be used to store input patterns?
The number of patterns that can be stored in a given network depends on?
What happens when number of available energy minima be less than number of patterns to be stored?
What happens when number of available energy minima be more than number of pattern to be stored?
How hard problem can be solved?
Why there is error in recall, when number of energy minima is more the required number of patterns to be store
How can false minima be reduced in case of error in feedback neural networks?
What is a boltzman machine?
What is objective of linear autoassociative feed forward networks?
Is there any error in linear autoassociativenetworks?
If input is " a(I)+e" where 'e' is the noise introduced then what is the output in case of autoassociative feedbacck r
If input is " a(I)+e" where 'e' is the noise introduced then what is the output if system is accretive in nature?
If input is " a(I)+e" where 'e' is the noise introduced then what is the output if system is interpolative in nature?
What property should a feedback network have to make it useful for storing information?
What is the objective of the pattern storage task in a network ?
Linear neurons can be useful for applications such as interpolation is it true?
What are the tasks that cannot be realised or recognized by simple networks?
Can data be stored directly in associative memory?
If the weight matrix stores the given patterns, then the network becomes?
If the weight matrix stores the association between a pair of patterns, then the network becomes?
If the weight matrix stores multiple associative among several patterns,then network becomes?
If the weight matrix stores association between adjacent pair of patterns,then network becomes?
Hetro associative memory is also known as ?
What are some of the desirable characteristics of associative memories?
What is the objective of BAM?
BAM is a special case of MAM,is that true?
What is the use of MLFFNN ?
What is the advantage of basis functon over multilayer   feedforward neural networks?
Why is the training of basis functions is faster than MLFFNN?
Pattern recall takes more time for?
In which type of network training is completely avoided?
What does GRNN do?
What does PNN do?

The CPN provides practical approach for implementing?
What consist of a basis counterpropagation network?
How does the name counter propagation signifies its architecture?
An auto associative network is ?
What is true about sigmoidal neurons?
The bidirectional associative memeory is similar in principle to?
What does ART stands for?
What is the purpose of ART?
What type learning is involved in ART?
ART is made to tackle?
What type of input does ART-1 receive?

to develop learning algorithm for multilayer feedforward neural network
yes
is is also called as generalized data rule
yes
it is a feedback neural network
because delta rule can be extended to hidden layer units
local minima problem
pattern mapping
yes
there is convergence involved
heteroassociaton
yes
yes
pattern classifaction
mean of peaks and valleys
number of units
pattern storage is not possible in this case
no effect
by providing additional units in a feedback network
due to noise
by providing additional units
a feedback network with hidden units
to associative a given pattern with itself
yes
a(l)
a(l)
a(l)
accretive behaviour
to sore a given set of patterns
yes
handwritten characters
yes
autoassociative memory
autoassociative memory
autoassociative memory
autoassociative memory
uni directional memory
ability to store large number of patterns
to store pattern pairs
yes
to realize structure of MLP
training of basis functon is faster than MLFFNN
because they are developed specifically for pattern approximation
MLFFNN
GRNN
function approximation task
function approximation task

pattern approximation
a feedforward network only
its ability to learn inverse mapping functions
network in neural which contains feeback
can accept any vectors of real numbers as input
hebb learning model
automatic tresonance theory
take care of approximation in a network
supervised
stability problems
bipolar

to develop learning algorithm for single layer feedforward neural network
no
error in output is propagated backwards only to determine weight updates
no
actual output is determined by computing the output of units for each hidden layer
because delta is applied to only output and input layers thus making it more simple and generalized
slow convergence
function approximation
no
no heuristic criteria exist
autoassociation
no
no
recall
maxima
strength of connecting links
pattren storage can be easily done
pattern storage is not possible in that case
nothing can be done
due to additional false maxima
by using probabilistic update or using additional units
a feedforward network with hidden units and probabilistic update
to associate a given pattern with others
no
a(l)+e
a(l)+e
a(l)+e
interpolative behaviour
to recall a given set of patterns
no
speech characters
no
hetroassociative memory
hetroassociative memory
hetroassociative memory
hetroassociative memory
bi directional memeory
fault tolerance
to recall pattern pairs
no
to solve pattern classificatoin pattern
training of basis function is slower than MLFFNN
because they are developed specifically for pattern classification
basis function
PNN
pattern classification task
pattern classification task

pattern classsification
a feedforward network with hidden layer
its ability to learn forward mapping functions
network in neural which contains loops
outputs a real numbers between 0 and 1
boltzman model
artificial resonance theory
take care of updates of weights
unsupervised
hard problems
binary

to develop learning algorithm for single layer network,so that network can be trained to capture the mapping imp
no other name
there is no feedback of signal at any stage
only comments
hidden layer output is not all important they are only meant for supporting input and output layers
it has no significance
scaling
predictoin
cannot be said
on basis of average gradient value
can be a hetro or autoassociation,depends on situation
cant be said
cant be said
pattern storage
minima
both number of units and strengths of connecting links
pattern storage problem becomes hard problem for the network
error in recall
by removing units in hiddden layer
due to additional false minima
can be either probabilistic update or using additional units
a feedforward network with hiddden units
to associate output with input
cant be said
could be either a or b
could be either a or b
could be either a or b
both accretive and interpolative behaviour
both of the above
cant be said
image sequences
cant be said
multidirectional associative memory
multidirectional associative memory
multidirectional associative memory
multidirectional associative memory
multi directional memory
able to recall,even for input pattern is noisy
to store a set of pattern pairs and they can be recalled by giving either of pattern as input
cant be said
to solve pattern mapping problem
storing in basis function is faster than MLFFNN
because they are developed specifically for pattern approximation or classification
equal for both MLFFNN and basis function
GRNN nd PNN
function approximation and pattern classification task
function approximation and pattern classification task

pattern mapping
two feedforward network with hidden layer
its ability to learn foraward and inverse mapping functions
network in neural which no loops
they are most common type of neurons
papert model
adaptive resonance theory
take care of pattern storage
both
storage problems
both

none
data rule
all of the above mentioned
both comments and feedback
none of the mentioned
none of the mentioned
all of the above mentioned
all of the above mentioned
it depends on gradient descent but not error surface
none of the mentioned
none of the mentioned
not determined
not determined
all of the above mentioned
none of the mentioned
none of the mentioned
none of the mentioned
none of the mentioned
none of the mentioned
none of the mentioned
none of the above mentioned
a feed forward network with hidden units and probabilistic update
none of the above mentioned
can be determined
e
e
e
none of the mentioned
none of the above mentioned
cant be determined
all of the above mentioned
not yet determined
temporal associative memory
temporal associative memory
temporal associative memory
temporal associative memory
temporal associative memory
all of the above mentioned
none of the mentioned
not yet determined
to realize an approximation to a MLP
none of the mentioned
none of the above mentioned
none of the above mentioned
none of the above mentioned
none of the above mentioned
none of the above mentioned

pattern clustering
none of the above mentioned
none of the baove mentioned
none of the above mentioned
all of the above mentioned
none of the above mentioned
none of the above mentioned
none of the above mentioned
none
none
none

to develop learning algorithm for single layer network,so that network can be trained to capture the ma
yes
all of the above mentioned
no
actual output is determined by computing the output of units for each hidden layer
because delta rule can be extended to hidden layer units
all of the above mentioned
all of the above mentioned
yes
on basis of average gradient value
autoassociation
yes
no
pattern storage
minima
both number of units and strengths of connecting links
pattern storage problem becomes hard problem for the network
error in recall
by providing additional units in a feedback network
due to additional false minima
by using probabilistic update or using additional units
a feedforward network with hidden units and probabilistic update
to associative a given pattern with itself
no
a(l)+e
a(l)
a(l)+e
accretive behaviour
to recall a given set of patterns
yes
all of the above mentioned
no
autoassociative memory
hetroassociative memory
autoassociative memory
autoassociative memory
bi directional memory
all of the above mentioned
to store a set of pattern pairs and they can be recalled by giving either of pattern as input
yes
to realize an approximation to a MLP
training of basis functon is faster than MLFFNN
because they are developed specifically for pattern approximation or classification
basis function
GRNN nd PNN
function approximation task
function approximation task

pattern mapping
two feedforward network with hidden layer
its ability to learn foraward and inverse mapping functions
network in neural which contains feeback
all of the above mentioned
none of the above mentioned
adaptive resonance theory
none of the above mentioned
unsupervised
none
binary

ıpping implicitly

Unit II- Fuzzy Logic-1 Introduction

Basic concepts of fuzzy logic, Fuzzy sets and crisp sets, Fuzzy set theory and operations, properties of fuzzy sets, Fuzzy sets and crisp relations, Fuzzy to Crisp conversion.

FUZZY LOGIC TECHNIQUES

BASIC CONCEPT

Problems in the real world are quite often very complex due to the element of uncertainty. Although probability theory has been an age old and effective tool to handle uncertainty, it can be applied only to situations where the system characteristics are based on random processes. In such situations, fuzzy logic exhibits immense potential for effective solving of the uncertainty in the problem.

Fuzzy logic was develop by Lotfi A. Zadeh and represents a form of mathematical logic. Values between 0 and 1 represent uncertainty in decision-making. 0 indicates a false value while 1 indicates a true value. So within a fuzzy set a value x is not restricted by the values 0 or 1, but from the real interval {0;1}

Fuzzy logic is an extension of Boolean logic which handles the concept of partial truth, where the range of truth value is in between completely true and completely false. In classical logic concept we can express everything in the form of 1 or 0, true or false, or, white or black. But fuzzy logic replaces Boolean truth-values with some degree of truth. This degree of truth is used to capture the imprecise modes of reasoning that play an important role in the ability of human being to make decisions in an environment of uncertainty and imprecision. The process of formulating a mapping from a given input set to an output using fuzzy logic is known as fuzzy inference. The basic elements of fuzzy logic are fuzzy sets, linguistic variables and fuzzy rules. While variables in mathematics usually take numerical values, in fuzzy logic applications, the non-numeric *linguistic variables* are often used to facilitate the expression of rules and facts . The linguistic variables' are words, specifically adjectives like "small," "little," "medium," "high," and so on. A fuzzy set is a collection of couples of elements.

In narrow sense, fuzzy logic can be defined as a logical system that is the extension of multivalued logic. In a wider sense fuzzy logic is almost similar to the theory of fuzzy sets, which talks about the relationship between classes of object with unsharp boundaries and their membership values.

FUZZY LOGIC CONTROLLER

To tackle the load balancing problem, conventional control theory can be applied to restore system equilibrium. Fuzzy logic control attempts to capture intuition in the form of IF-THEN rules, and conclusions are drawn from these rules. Based on both intuitive and expert knowledge, system parameters can be modeled as linguistic variables and their corresponding membership functions can be designed. Thus, nonlinear system with great complexity and uncertainty can be effectively controlled based on fuzzy rules without dealing with complex, uncertain, and error-prone mathematical models.

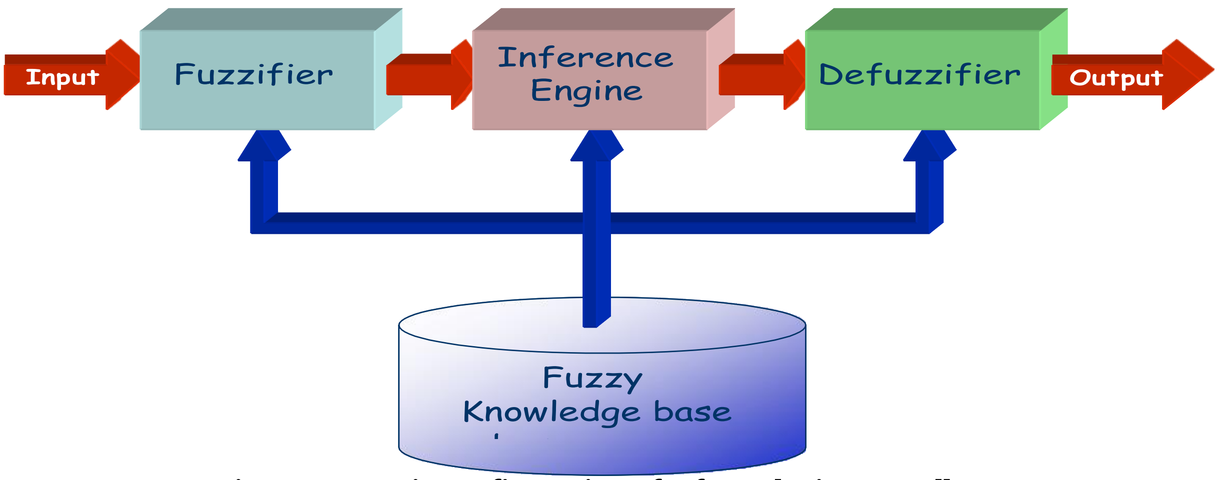


Figure 4.1 Basic configuration of a fuzzy logic controller

The architecture of the fuzzy logic controller shown in figure 4.1 above includes four components: Fuzzifier, Rule Base, Fuzzy Inference Engine, and Defuzzifier.

Fuzzifier: The fuzzifier is the input interface which maps a numeric input to a fuzzy set so that it can be matched with the premises of the fuzzy rules defined in the application- specific rule base.

Rule Base: The rule base contains a set of fuzzy if-then rules which defines the actions of the controller in terms of linguistic variables and membership functions of linguistic terms.

Fuzzy Inference Engine: The fuzzy inference engine applies the inference mechanism to the set of rules in the fuzzy rule base to produce a fuzzy output set. This involves matching the input fuzzy set with the premises of the rules, activation of the rules to deduce the conclusion of each rule that is fired, and combination of all activated conclusions using fuzzy set union to generate fuzzy set output.

Defuzzifier: The defuzzifier is an output mapping which converts fuzzy set output to a crisp output. Based on the crisp output, the fuzzy logic controller can drive the system under control.

The fuzzy rule base contains a set of linguistic rules. These linguistic rules are expressed using linguistic values and linguistic variables. Different linguistic values can be assigned to a linguistic variable. These linguistic values are modeled as fuzzy sets. Based on the linguistic values, their corresponding membership functions can be expressed based on application requirements. So, we can say that the job of a fuzzy logic controller is to carry out the following three steps.

1. To receive one or a large number, of measurement or other assessment of conditions existing in some system we wish to analyze or control.

2. Processing all these inputs according to human based, fuzzy "If-Then" rules, which can be expressed in plain language words.

3. Averaging and weighting the resulting outputs from all the individual rules into one single output decision or signal which decides what to do or tells a controlled system what to do. The output signal eventually arrived at is a precise appearing, defuzzified, "crisp" value.

FUZZY SET THEORY

A fuzzy set is a set which has no crisp, clearly defined boundary. It's elements only have partial degree of membership.

Membership Functions A curve that tells how all of the input points are mapped to the membership value is called a *membership function* (MF). This membership value is also known as degree of membership and its range is between 0 and 1. The input space here is known as the *universe of discourse* and the output-axis has the membership value which is in between 0 and 1. The curve known as *membership function* is represented by the symbol μ.
A fuzzy set is nothing but an extension of a classical set. If the Universe of discourse is X and its elements are represented by *x*, then a set of ordered pair

$A = \{(x, \mu_A(x)) \mid x \in X\}$
defines a fuzzy set where,
$\mu_A(x)$ is the membership function (or MF) of *x* in A. This function maps each element of X to a membership value between 0 and 1.

The only condition a membership function must satisfy is that it must vary between 0 and

1. There are 11 built-in membership functions in the fuzzy logic toolbox. Some of them are: triangular membership function, trapezoidal membership function etc. One example of a membership function for an input variable having three linguistic variables low, medium and high is shown in figure 4.2.

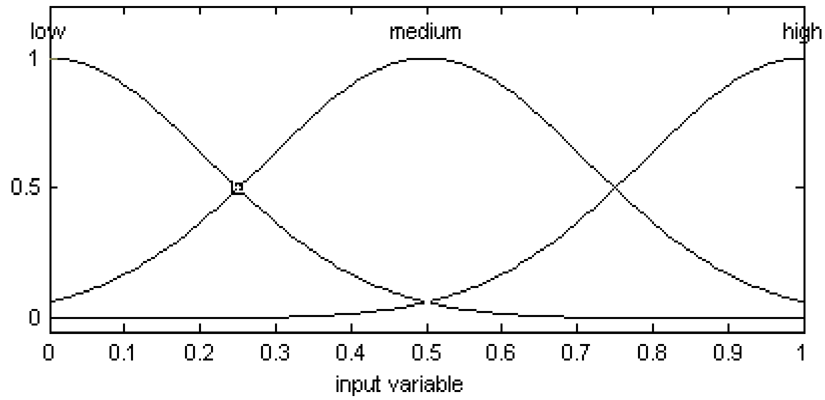


Figure 4.2: Membership function for an input variable with three linguistic variables low, medium and high.

Logical Operations

The fuzzy logical reasoning is a superset of standard Boolean logic. If in fuzzy logic we keep the membership values at the two extremes of 0 (completely false) and 1 (completely true) it becomes the same as Boolean logic.. The statement *A* AND *B* is resolved using the min function, where *A* and *B* are limited to the range (0, 1). Similarly the OR operation is replaced by the *max* function, such that *A* OR *B* is equivalent to *max* (*A*, *B*).Similarly NOT *A* is same as the operation 1− *A*.

A	B	Min(A,B)
0	0	0
0	1	0
1	0	0
1	1	1

Figure 4.3: Truth table for AND operator

A	B	Max(A,B)
0	0	0
0	1	1
1	0	1
1	1	1

Figure 4.4: Truth table for OR operator

A	1-A
0	1
1	0

Figure 4.5: Truth table for NOT operator

If-Then Rules

We have the concept of a subject and a verb in fuzzy logic and if-then rule statements are used to make conditional statements that are the building blocks of fuzzy logic. The form of a single fuzzy if-then rule is

if *x* is *A* then *y* is *B*

where *A* and *B* are the values of the linguistic variables X and Y, respectively. *Antecedent* or premise is the if-part of the rule "*x* is *A*" while *consequent* or conclusion is the then-part of the rule "*y* is *B*".

Interpretation of If-Then Rules

The interpretation of the if-then rule is done in three parts:

1 **Fuzzify inputs**: All the fuzzy statements in the antecedent are resolved to a degree of membership between 0 and 1. If the antecedent has only one part then this is the degree of support for the rule.

2 **Apply fuzzy operator to multiple part antecedents**: If the antecedent has more than one part then we apply fuzzy logic operators to resolve the antecedent to a single number between 0 and 1. This becomes the degree of support for the rule.

3 **Apply implication method**: Now the degree of support for the entire rule is used to shape the output fuzzy set. An entire fuzzy set is assigned to the output by the consequent. This fuzzy set is represented by a membership function indicates the qualities of the consequent. The fuzzy set is truncated according to the implication method if the antecedent is only partially true, (i.e., is assigned a value less than 1).

FUZZY INFERENCE SYSTEM

The process of creating a mapping between input and output using fuzzy logic is known as fuzzy inference. The mapping is the base from which decisions can be made, or patterns discerned. Two types of fuzzy inference systems can be implemented in the toolbox: Mamdani-type and Sugeno-type. The description of these two methods is given in [94, 95]. The most commonly used method is the Mamdani's fuzzy inference system. This was one of the first control systems built using fuzzy set theory proposed by Ebrahim Mamdani [96] in 1975. It was developed in an attempt to control a steam engine and boiler combination by synthesizing a set of linguistic control rules obtained from experienced human operators. Lotfi Zadeh's 1973 paper on fuzzy algorithms for complex systems and decision processes [97] was the driving force behind this work of Mamdani. After the aggregation process, we get a fuzzy set for each output variable which is defuzzified to get the crisp values.

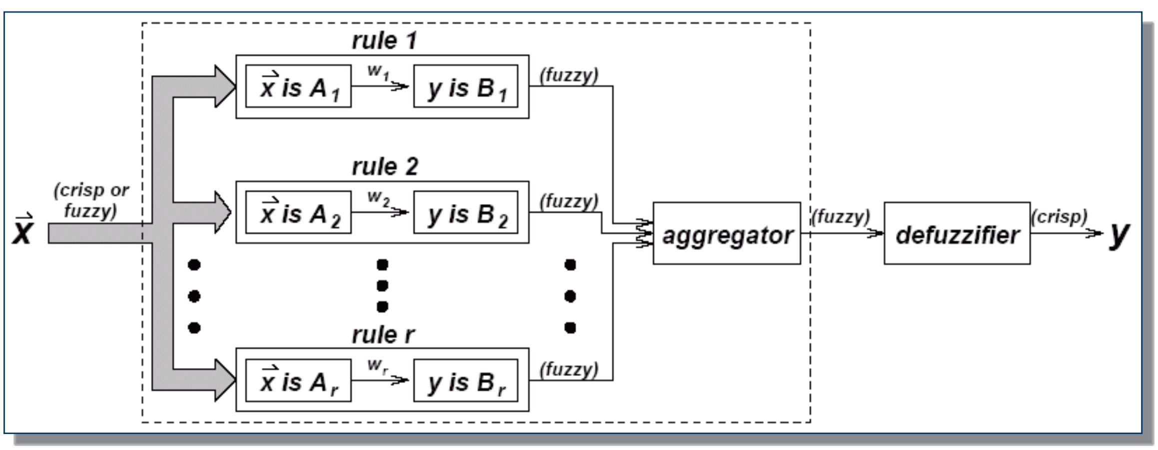


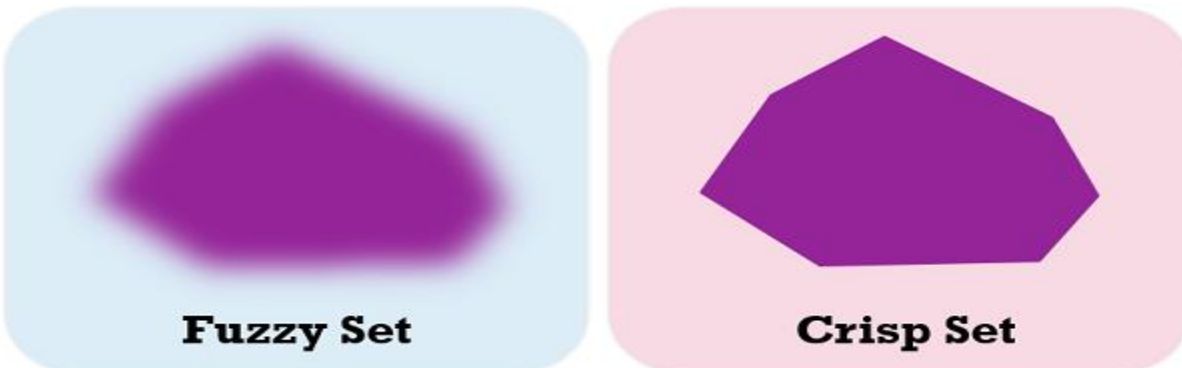
Figure 4.6: Block diagram for a fuzzy inference system

FUZZY LOGIC APPLICATION IN LOAD DISTRIBUTION

There are numerous papers which talk about use of fuzzy logic in load balancing. These papers address the issue of uncertainty in the state of the distributed system and how it can be resolved using fuzzy logic. Stankovic applied the Bayesian decision theory to job scheduling to manage uncertainty in the overall system state. A fuzzy expert system for load balancing has been described by Kumar et al. Each node of a distributed computing system has an expert system that plays the role of a distributed decision maker. The fuzzy expert system reflects the impression in state information and makes scheduling decision based on a fuzzy logic. In it has been assumed that the performance of any distributed computing system cannot be improved beyond a limit which is determined by the degree of uncertainty in global state. Here each distributed node dynamically adjusts its thresholds denoting the amount of consistency relaxation depending on the degree of uncertainty in the system state. A fuzzy based consistency model provides a mechanism that allows each node to make flexible scheduling and state update decisions based on its threshold.Dierkes has compared his own fuzzy based load balancing algorithm with two standard algorithms: Join Shortest Queue and prefer fastest. For his own algorithm also he has taken different scenarios with different number of rules and fuzzy sets. By using Mamdam's fuzzy inference in his decision algorithm, Dierkes was able to map attributes such as queue length onto a fuzzy set with three terms (small medium and large) over specific ranges. In another paper , Dierkes based load balancing decision making on a set of possible actions, a set of goals with priorities, and an effect matrix. The effect matrix showed the effect of the actions on the goals (whether the action either distracts or supports a goal). The application of fuzzy sets taken from the matrix significantly improved system performance, with a bias towards achieving the goals given weight in the effect matrix. Three practical approaches for load balancing in distributed object computing systems have been studied in . This paper considers JavaSpecs based, request redirection based and fuzzy decision based approaches and it has been found that fuzzy decision based algorithm outperforms the other two considerably. Another fuzzy logic based algorithm has been implemented in which effectively reduces the amount of communication messages and also helps in giving high throughputs, low response time and short turnaround times. Fuzzy logic has been used for scheduling of periodic tasks in soft real time multiprocessor systems in . Load balancing in Web server cluster has been proposed in where the actual status of the real server has been modeled using fuzzy inference system and adaptive Neuro-fuzzy inference system. It has been shown that by using these approaches the scheduling overhead has been reduced and availability of these servers has been increased.

Center of gravity defuzzification method has been performed using the discretisation method, and two new methods namely slope based method and the modified transformation function method in . It has been found that the new methods give very good accuracy at low computational cost as compared to the discretisation method.
Fuzzy set and crisp sets

Difference Between Fuzzy Set and Crisp Set



Fuzzy set and crisp set are the part of the distinct set theories, where the fuzzy set implements infinite-valued logic while crisp set employs bi-valued logic. Previously, expert system principles were formulated premised on Boolean logic where crisp sets are used. But then scientists argued that human thinking does not always follow crisp “yes”/”no” logic, and it could be vague, qualitative, uncertain, imprecise or fuzzy in nature. This gave commencement to the development of the fuzzy set theory to imitate human thinking.For an element in a universe, that comprise fuzzy sets can have a progressive transition among several degrees of membership. While in crisp sets the transition for an element in the universe between membership and non-membership in a given set is sudden and well defined.

Fuzzy set theory and operations

Before illustrating the mechanisms which make fuzzy logic machines work, it is important to realize what fuzzy logic actually is. Fuzzy logic is a superset of conventional(Boolean) logic that has been extended to handle the concept of partial truth- truth values between "completely true" and "completely false". As its name suggests, it is the logic underlying modes of reasoning which are approximate rather than exact. The importance of fuzzy logic derives from the fact that most modes of human reasoning and especially common sense reasoning are approximate in nature.

The essential characteristics of fuzzy logic as founded by Zader Lotfi are as follows.

- In fuzzy logic, exact reasoning is viewed as a limiting case of approximate reasoning.
- In fuzzy logic everything is a matter of degree.
- Any logical system can be fuzzified
- In fuzzy logic, knowledge is interpreted as a collection of elastic or, equivalently , fuzzy constraint on a collection of variables
- Inference is viewed as a process of propagation of elastic constraints.

The third statement hence, define Boolean logic as a subset of Fuzzy logic.

Fuzzy Sets

Fuzzy Set Theory was formalised by Professor Lofti Zadeh at the University of California in 1965. What Zadeh proposed is very much a paradigm shift that first gained acceptance in the Far East and its successful application has ensured its adoption around the world.

A paradigm is a set of rules and regulations which defines boundaries and tells us what to do to be successful in solving problems within these boundaries. For example the use of transistors instead of vacuum tubes is a paradigm shift - likewise the development of Fuzzy Set Theory from conventional bivalent set theory is a paradigm shift.

Bivalent Set Theory can be somewhat limiting if we wish to describe a 'humanistic' problem mathematically. For example, Fig 1 below illustrates bivalent sets to characterise the temperature of a room.

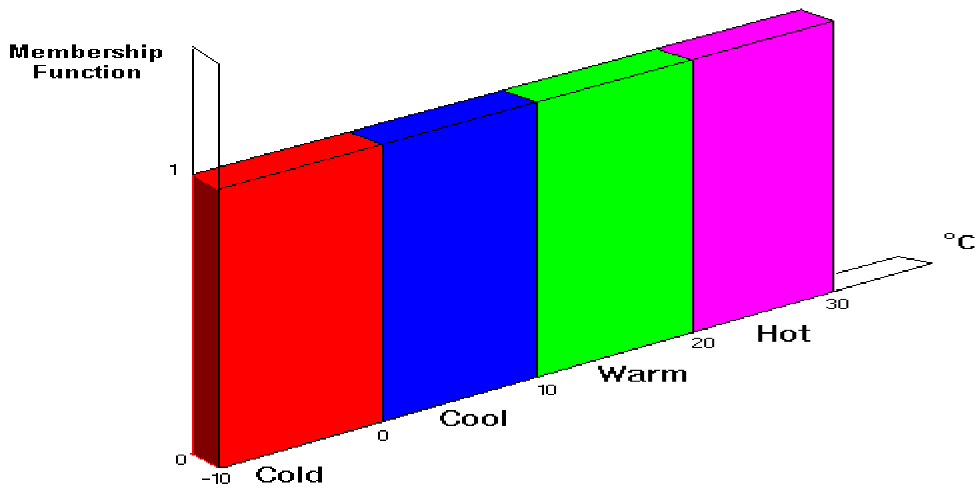


Fig. 1 : Bivalent Sets to Characterize the Temp. of a room.

The most obvious limiting feature of bivalent sets that can be seen clearly from the diagram is that they are mutually exclusive - it is not possible to have membership of more than one set (opinion would widely vary as to whether 50 degrees Fahrenheit is 'cold' or 'cool' hence the expert knowledge we need to define our system is mathematically at odds with the humanistic world). Clearly, it is not accurate to define a transiton from a quantity such as 'warm' to 'hot' by the application of one degree Fahrenheit of heat. In the real world a smooth (unnoticeable) drift from warm to hot would occur.

This natural phenomenon can be described more accurately by Fuzzy Set Theory. Fig.2 below shows how fuzzy sets quantifying the same information can describe this natural drift.

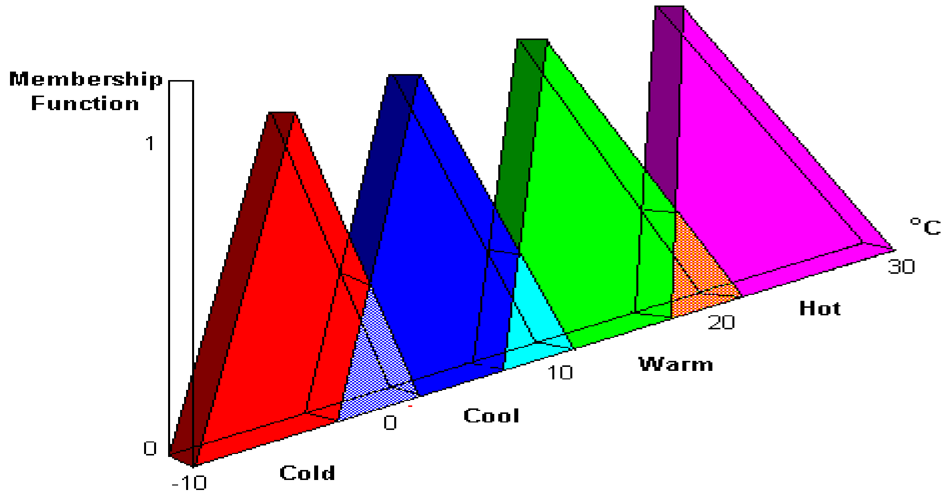


Fig. 2 - Fuzzy Sets to characterize the Temp. of a room.

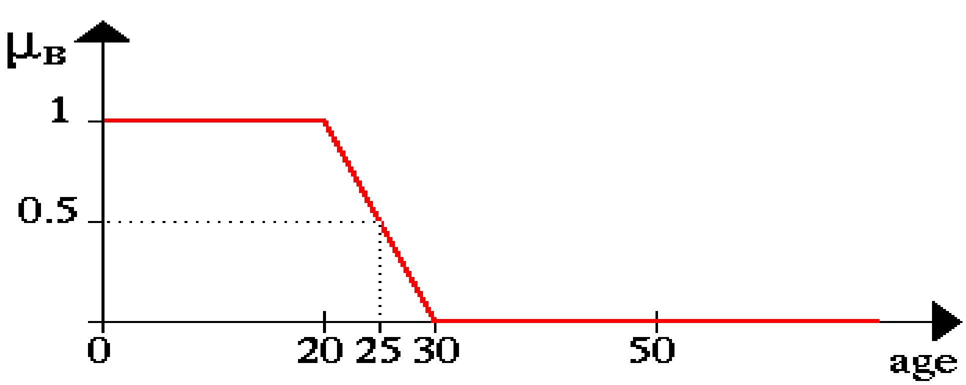
The whole concept can be illustrated with this example. Let's talk about people and "youthness". In this case the set S (the universe of discourse) is the set of people. A fuzzy subset YOUNG is also defined, which answers the question "to what degree is person x young?" To each person in the universe of discourse, we have to assign a degree of membership in the fuzzy subset YOUNG. The easiest way to do this is with a membership function based on the person's age.

young(x) = { 1, if age(x) <= 20,

(30-age(x))/10, if 20 < age(x) <= 30,

0, if age(x) > 30 }

A graph of this looks like:



Given this definition, here are some example values:

Person	Age	degree of youth
Johan	10	1.00
Edwin	21	0.90
Parthiban	25	0.50
Arosha	26	0.40
Chin Wei	28	0.20
Rajkumar	83	0.00

So given this definition, we'd say that the degree of truth of the statement "Parthiban is YOUNG" is 0.50.

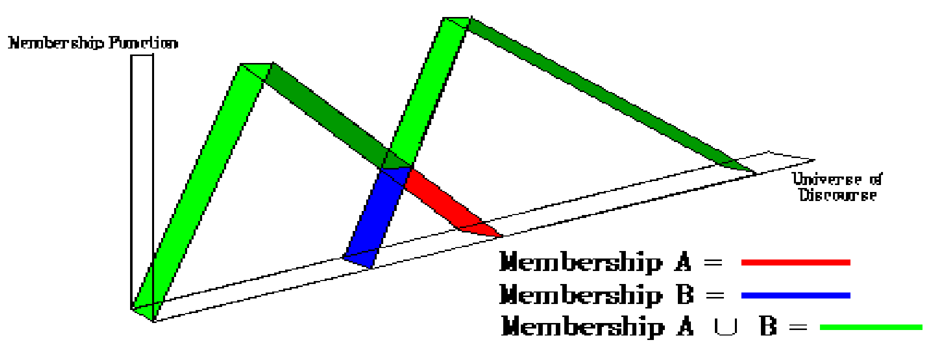
Note: Membership functions almost never have as simple a shape as age(x). They will at least tend to be triangles pointing up, and they can be much more complex than that. Furthermore, membership functions so far is discussed as if they always are based on a single criterion, but this isn't always the case, although it is the most common case. One could, for example, want to have the membership function for YOUNG depend on both a person's age and their height (Arosha's short for his age). This is perfectly legitimate, and occasionally used in practice. It's referred to as a two-dimensional membership function. It's also possible to have even more criteria, or to have the membership function depend on elements from two completely different universes of discourse.

Fuzzy Set Operations.

Union

The membership function of the Union of two fuzzy sets A and B with membership functions μ_A and μ_B respectively is defined as the maximum of the two individual membership functions. This is called the *maximum* criterion.

$$\mu_{A \cup B} = \max(\mu_A, \mu_B)$$

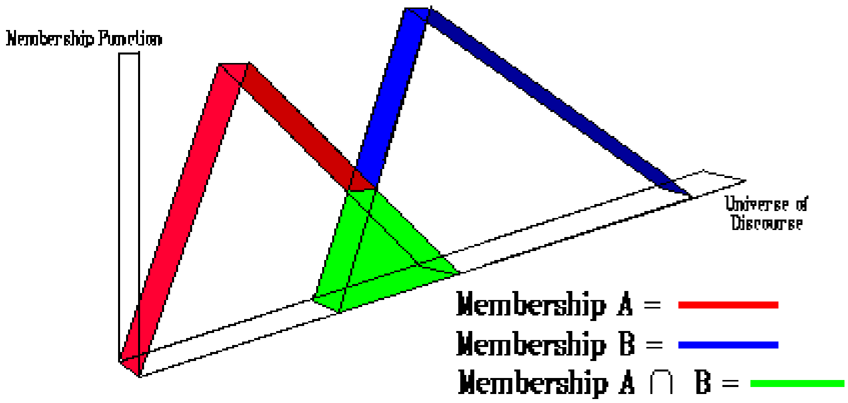


The Union operation in Fuzzy set theory is the equivalent of the **OR** operation in Boolean algebra.

Intersection

The membership function of the Intersection of two fuzzy sets A and B with membership functions μ_A and μ_B respectively is defined as the minimum of the two individual membership functions. This is called the *minimum* criterion.

$$\mu_{A \cap B} = \min(\mu_A, \mu_B)$$

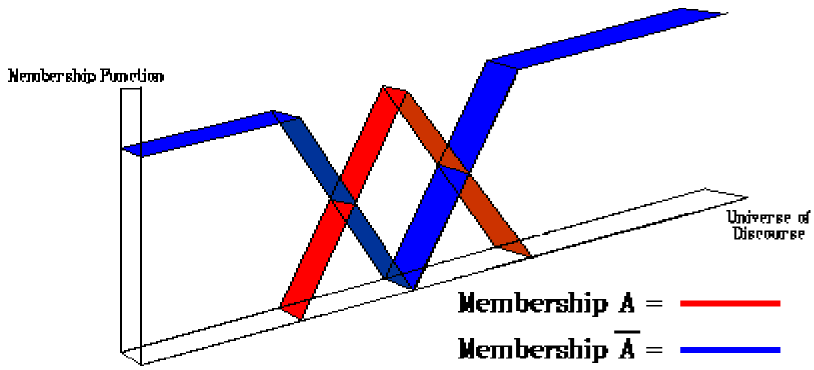


The Intersection operation in Fuzzy set theory is the equivalent of the **AND** operation in Boolean algebra.

Complement

The membership function of the Complement of a Fuzzy set A with membership function μ_A is defined as the negation of the specified membership function. This is caleed the *negation* criterion.

$$\mu_{\overline{A}} = 1 - \mu_A$$



The Complement operation in Fuzzy set theory is the equivalent of the **NOT** operation in Boolean algebra.

The following rules which are common in classical set theory also apply to Fuzzy set theory.

De Morgans law

$$\overline{(A \cap B)} = \overline{A} \cap \overline{B} \,,\; \overline{(A \cup B)} = \overline{A} \cap \overline{B}$$

Associativity

$$(A \cap B) \cap C = A \cap (B \cap C)$$

$$(A \cup B) \cup C = A \cup (B \cup C)$$

Commutativity

$$A \cap B = B \cap A \,,\; A \cup B = B \cup A$$

Distributivity

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

Glossary

Universe of Discourse

The Universe of Discourse is the range of all possible values for an input to a fuzzy system.

- Fuzzy Set

  A Fuzzy Set is any set that allows its members to have different grades of membership (membership function) in the interval [0,1].

- Support

  The Support of a fuzzy set F is the crisp set of all points in the Universe of Discourse U such that the membership function of F is non-zero.

- Crossover point

  The Crossover point of a fuzzy set is the element in U at which its membership function is 0.5.

- Fuzzy Singleton

  A Fuzzy singleton is a fuzzy set whose support is a single point in U with a membership function of one.

Properties of Fuzzy sets:

Properties of Fuzzy Sets

Fuzzy sets are defined as sets that contain elements having varying degrees of membership values. Given A and B are two fuzzy sets, here are the main properties of those fuzzy sets:

Commutativity :-

- $(A \cup B) = (B \cup A)$
- $(A \cap B) = (B \cap A)$

Associativity :-

- $(A \cup B) \cup C = A \cup (B \cup C)$
- $(A \cap B) \cap C = A \cap (B \cap C)$

Distributivity :-

- $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
- $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$

Idempotent :-

- $A \cup A = A$
- $A \cap A = A$

Identity :-

- $A \cup \Phi = A \Rightarrow A \cup X = X$
- $A \cap \Phi = \Phi \Rightarrow A \cap X = A$

Note: (1) Universal Set 'X' has elements with unity membership value. (2) Null Set 'Φ' has all elements with zero membership value.

Transitivity :-

- If A ⊆ B, B ⊆ C, then A ⊆ C

Involution :-

- $(A^c)^c = A$

De morgan Property :-

- $(A \cup B)^c = A^c \cap B^c$
- $(A \cap B)^c = A^c \cup B^c$

Note: $A \cup A^c \neq X$; $A \cap A^c \neq \Phi$

Since fuzzy sets can overlap "law of excluded middle" and "law of contradiction" does not hold good.

Fuzzy and Crisp Relations:

Crisp and Fuzzy Relations

A **crisp relation** represents the presence or absence of association, interaction, or interconnectedness between the elements of two or more sets. This concept can be generalized to allow for various degrees or strengths of relation or interaction between elements. Degrees of association can be represented by membership grades in a **fuzzy relation** in the same way as degrees of set membership are represented in the fuzzy set. In fact, just as the crisp set can be viewed as a restricted case of the more general fuzzy set concept, the crisp relation can be considered to be a restricted case of the fuzzy relations.

Fuzzy to Crisp Conversion:

Applications:How is fuzzy used in washing machine? Inbuilt sensors monitor the washing process and make corrections to produce the best washing results. The fuzzy logic checks for the extent of dirt and grease, the amount of soap and water to add, direction of spin, and so on. The machine rebalances washing load to ensure correct spinning.

What is fuzzy inference system? Fuzzy Inference System is the key unit of a fuzzy logic system having decision making as its primary work. It uses the "IF…THEN" rules along with connectors "OR" or "AND" for drawing essential decision rules.

Characteristics of Fuzzy Inference System

Following are some characteristics of FIS −

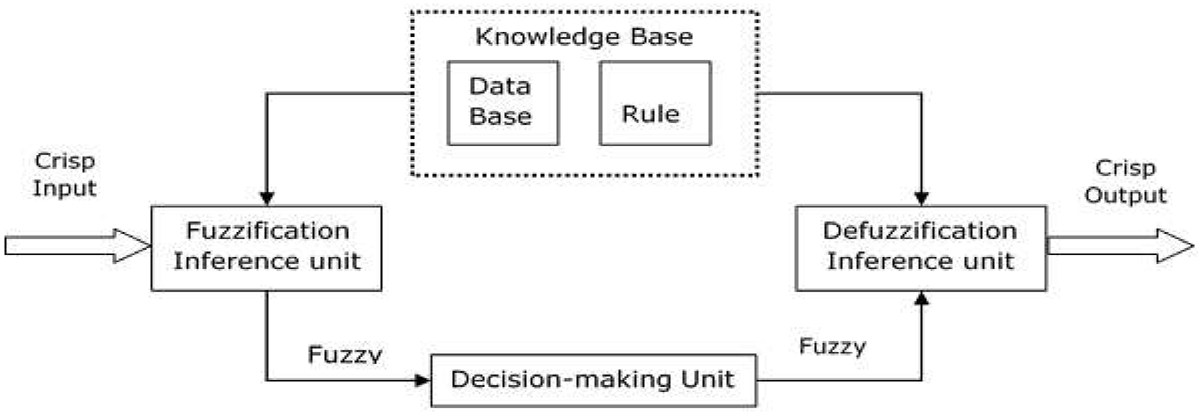
- The output from FIS is always a fuzzy set irrespective of its input which can be fuzzy or crisp.
- It is necessary to have fuzzy output when it is used as a controller.

- A defuzzification unit would be there with FIS to convert fuzzy variables into crisp variables.

Functional Blocks of FIS

The following five functional blocks will help you understand the construction of FIS −

- Rule Base − It contains fuzzy IF-THEN rules.
- Database − It defines the membership functions of fuzzy sets used in fuzzy rules.
- Decision-making Unit − It performs operation on rules.
- Fuzzification Interface Unit − It converts the crisp quantities into fuzzy quantities.
- Defuzzification Interface Unit − It converts the fuzzy quantities into crisp quantities. Following is a block diagram of fuzzy interference system.



Working of FIS

The working of the FIS consists of the following steps −

- A fuzzification unit supports the application of numerous fuzzification methods, and converts the crisp input into fuzzy input.
- A knowledge base - collection of rule base and database is formed upon the conversion of crisp input into fuzzy input.
- The defuzzification unit fuzzy input is finally converted into crisp output.

Fuzzy logic is the form of
Traditional set theory is also known as crisp set theory
The truth values of traditional set theory is ___________and tht of fuzzy set is_______
Fuzzy logic is extension of crisp set with an extension of handling the concept of partial truth.
the room temperature is hot. Here the hot(use of linguistic variable is used) can be represented by ____
the values of the set membership is represented by ________________
_________ were the first to utilize fuzzy logic practically on high speed trains in sendei
Choose the fuzzy operations from the following__________.
Fuzy logic is more linguistic in nature ,called______ that can be applied to fuzzy set theory
Fuzzy logic is usually represented as
Like relational databases there does exist fuzzy relational databases
__________ is/are the way/s to represent uncertainity
________ are the algorithms that learn from more complex environments(hence eco) to generalize,approximate a
The height h(A) of a fuzzy set A is defined as h(A)=supA(X) where X belongs to A. then the fuzzy set A is called norr
what is fuzzy approximation thoerem(FAT) ?
Consider a fuzzy set old as defined below old= {(20,0),(30,0.2),(40,0.4),(50,0.6),(60,0.8),(70,1),(80,1)}. Then the alp
Equilibrium of a fuzzy complement c is a solution of the equation
________ is / are the way/s to represent uncertainity
Involutive property of the standard fuzzy complement c, for each a?[,] is ______________
How is fuzzy logic is different from conventionl control methods?
If a and b are two fuzzy set with membership functions ? A(x) = {0.6,0.5,0.1,0.7,0.8}? B(x)={0.9,0.2,0.6,0.8,0.5} the
Given U={1,2,3,4,5,6,7} A={(3,0.7),(5,1),(6,0.8)} then A will be:(where ~ ? Complement)
Who is the founder of fuzzy logix?
What is the sequence of steps taken in designing a fuzzy logic machine?
Fuzzy logic is _____________
Crisp relation on sets are subsets of___________ of the given sets
If A=1,2,3 and B = 1,2,6,7 what is aUb
If a=4,5,6,7 what is the cardinality of set a
Identify the power set of a in count, if a= 1,5,7
Universal qualifier and extensial qualifier are used in
Union of two fuzzy set A and b are represented by
The synergestic integration of two or more technology is called as
An element either belongs to or doent belongs to a set are termed as
Neural networks are trained with known samples of an problem to acquire_______ about it
Guiding principle of the soft computing is to ___________
Poor choice of learning coefficient in back propagation network can result in a _______________-
An artificial network neural back propagation is used totrain data and infer results for journal ball bearing Is called
Associative memories are a class of neural network architechture which is stored associate patterns in some form
Recognition of characters and fabric defect identification and applications of ________
For affective decision making _______ techniques are used to render the fuzzy outputs of a system in crisp system
A graph g is an ordered two tuple (V,E) consisting of a set V(vertix) and E(edges) with a direction called
In propositional logic,the associative property of (P^Q)^R=?
The idempotence of propositional logic of PvP=?
________ theory is an effective tool to tackle the problem of uncertainity
A kind of computing capable of producing accurate results even if the supplied data is noisy
Neural network is a ___________ model of biological neuron system which is _______ madae up of highly interco

The stability-plasticity propety holds good for_______
Summation of weighted inputs in artificial neural network is called as ___________.
Convergence refers to ________.
When a target output is not given to an algorithim but algorithim learns on it adaption called
Average weight of brain is about
"Survival of the fittest" theory is idealy suited for
Crisp relation onsets are _________of cartesian product of the given sets
________ operators are mutation and cross over
Which algorithim is used for solving temporal probablistic reasoning
How does the state of process is described in HMM
Where does the additional variable are added in HMM
Which algorithim works by first running standard forward pass to compute
Which suggest the existence of efficient reversive algorithim for online smoothing
Which reveals an improvemnet in online smoothing

two valued logic

TRUE

either 0 or 1,between 0&1

TRUE

fuzzy set
discrete set
japanese
union
hedges
IF-THEN-ELSE rules
true
fuzzy logic
fuzzy relations db
h(A)=0
a fuzzy system can model any continuous system
{(40,0.3)}
c(a)-a=1
fuzzy logic
c(c(a))=c(a)
IF and THEN approach
{0.9,0.5,0.6,0.8,0.8}
{(4,0.7),(2,1),(1,0.8)}
aristole
fuzzification-> rule evaluation-> defuzzification
used to respond to questions in a humanlike way
complement
2,3,
infifnity

3

propositional logic
µaUb(x)=min(µa(x),µb(x))
hard computing technology
fuzzy set
data
exploit the intolerance for imprecision,uncertainity and partial truth
faster convergence
augumented network
recall
two input percetron
generalized modus ponens
null graph
p^Q^R
P
crisp set
hard computing
simplified and massively paralllel

neural network and fuzzy system
sum of product
dispersion of multiple values
hebbian learning
100g
neural networks
bi-sets
genetic
hill climbing search
literal
temporal model
smoothing
matrix
matrix formulation

crist set logic
FALSE
between 0& 1,either 0 or 1
FALSE
crisp set
degre of truth
chinese
intersection
lingual variable
IF-THEN rules
FALSE
probability
ecorithms
h(A)<0
the conversion of fuzzy logic to probability
{50,60,70,80}
c(a)-a=2
probability
c(c(a))=1
FOR approach
{0.6,0.2,0.1,0.7,0.5}
{(4,0.3),(5,0),(6,0.2)}
buddha
defuzzification->fuzzification->rule valuation
a new programming language used to program animation
union
1,2,5,7
1
9

predicate logic
µaUb(x)=max(µa(x),µb(x))
soft computing technology
crisp set
knowledge
exploit the intolerance for imprecision,uncertainity and truth
failure convergence
neuronet
show
back propagation networks
generalized modus tolens
graph
Pv(QvR)
p^P
null set
soft computing
complex and massively parallel architechture

neural networks and genetic system
product of sum
arriving at exact solution
unsupervised learning
1500g
neural&fuzzy system
subsets
artificial
hitten morko model
single random variable
reality model
modified smoothing
constant space
set formulation

many valued logic
cant be said
between 0&1, between 0&1
cant be said
fuzzy & crispy set
probabilities
indians
complement
fuzzy variable
both
conditional databases
entropy
fuzzy set
h(A)=1
a continuous system can model an fuzy system
{(20,0.1)(30,0.2)}
c(a)=2a
entropy
c(c(a))=0
WHILE approach
{0.1,0.5,0.4,0.2,0.2}
{(1,1),(2,1),(3,0.3),(4,1),(6,0.2),(7,1)}
zadeh lofti
fuzzification->rule evaluation
the result of fuzzy thinking
cartesian product
1,2,3,6,7
4
8
crisp logic
µaUb(x)=1-(µa(x),µb(x))
hybrid technology
super set
experience
exploit the intolerance for imprecision,certainity and partial truth
fatal error
conscious networks
relate
fuzzy system
defuzzification
bi-directional graph
(P^R)v(Q^R)
~(P~P)
fuzzy set
hybrid computing
simplified and massively parallel distributed processing system

only for traditional programming
signum function
changing target out
supervised learning
1.2kg
genetic algorithim
both
both
depth _first search
discrete variable
probability model
HMM
constant time
HMM

binary set logic
not determined
either 0&1,either 0or 1
not determined
none of the above
both degree of truth and probabilities
none
all of the mentioned
none of the mentioned
none of the above
logical databases
all of the mentioned
none of the mentioned
h(A)<1
fuzzy patches covering a series of fuzzy rules
{(20,0),(30,0)(40,1),(50,1),(60,1),(70,1),(80,1)}
c(a)-a=0
all of the mentioned
c(c(a))=a
DO approach
{0.1,0.5,0.4,0.2,0.3}
{(3,0.3),(6,0.2)}
bart kosko
rule evaluation -> fuzzification-> defuzzification
a term that indicates logical values greater than one
none of the above
1,2
none of these

2

none of these
both a and c
none
null set
technical input
exploit the tolerance for imprecision,uncertainity and partial truth
none of the above
none of the above
none of the above
genetic algorithm
inferncing
di-graph
none of the above
~P
none of the above
genetic computing
complex and massively parallel distributed processing system

none of the above
threshold function
training the algorithim to learn
both b&c
none of these
both a&c
none
none
breath_first search
possible states of the world
all of the above
none
none
none

many valued logic
TRUE
either 0 or 1,between 0&1
TRUE
fuzzy set
degre of truth
japanese
all of the mentioned
hedges
IF-THEN rules
true
all of the mentioned
fuzzy set
h(A)=1
a fuzzy system can model any continuous system
{(20,0),(30,0)(40,1),(50,1),(60,1),(70,1),(80,1)}
c(a)-a=0
all of the mentioned
c(c(a))=a
IF and THEN approach
{0.1,0.5,0.4,0.2,0.2}
{(1,1),(2,1),(3,0.3),(4,1),(6,0.2),(7,1)}
zadeh lofti
fuzzification-> rule evaluation-> defuzzification
used to respond to questions in a humanlike way
cartesian product
1,2,3,6,7
4
8
predicate logic
$\mu_{aUb}(x)=max(\mu_a(x),\mu_b(x))$
hybrid technology
crisp set
knowledge
exploit the tolerance for imprecision,uncertainity and partial truth
failure convergence
neuronet
recall
back propagation networks
defuzzification
di-graph
Pv(QvR)
P
fuzzy set
soft computing
complex and massively parallel architechture

none of the above
sum of product
training the algorithim to learn
unsupervised learning
1500g
genetic algorithim
subsets
genetic
hitten morko model
possible states of the world
temporal model
smoothing
constant space
matrix formulation

Unit IV: Fuzzy Logic II(Fuzzy membership, Rules) Membership functions, interference in fuzzy logic, fuzzy if-then rules, Fuzzy implications and Fuzzy algorithms, Fuzzyfications & Defuzzifications, Fuzzy controller, Industrial applications

Fuzzy sets: Fuzzy sets support a flexible sense of membership of elements to a set. While in crisp set theory an element either belongs to or does not belong to a set, in fuzzy set theory many degree ofmembership(between 0 and 1) are allowed. Thus, a membership function is associated with a fuzzy set (A Complement) such that the function maps every element of the universe of discourse X(or the reference set) to the interval[0,1]. Formally, the mapping is written as Mhu(x):X->[0,1]

A fuzzy set is defined as follows: If X is a universe of discourse and x is a particular element of X, then a fuzzy set A defined on X may be written as a collection of ordered pairs.

A ={(x,mhu(x)),x belongs to X} where each pair(x,mhu(x)) is called as singleton.In crisp sets mhu(x) isdropped.

An alternative definition which indicates a fuzzy set as a union of all mhu(x)/x singleton is given by

$A= \sum mhu_{A\ comp}(X_i)/X_i)$ in the discrete case and

$A=\int mhu_{A\ comp}(x)/x$ in the continuous case

Here the summation and integration sign indicates the union of all $mhu_{A\ comp}(x)/x$ singletons

Example

Let X={g1,g2,g3,g4,g5} be the reference set of students. Let $_{Acomp}$ be the fuzzy set of "smart" students, where "smart" is a fuzzy linguistic term

A comp = {(g1,0.4)(g2,0.5)(g3,1)(g4,0.9)(g5,0.8)}

Here, A comp indicates that the smartness of g1 is 0.4, g2 is 0.5 and so on when graded over a scale of 0-1.Though fuzzy sets model vagueness, it needs to be realized that the definition of sets have one kind of fuzzy set while referring to the height of a building and another kind of fuzzy set while referring to the height of humans.

Membership Function

The membership function values need not always be described by discrete values, often these turn out to be as described by a continuous function. The fuzzy membership function for the fuzzy linguistic term “cool” relating to temperature may turn out to be as in the below figure

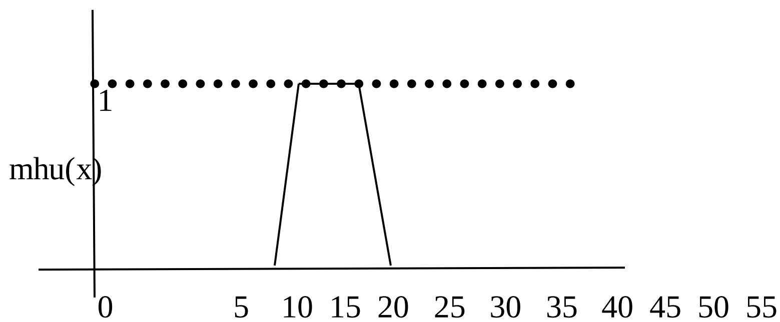
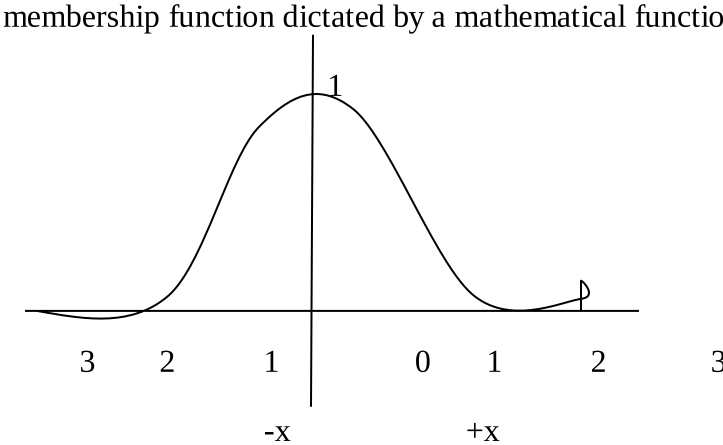


Fig Continuous membership function for “cool”

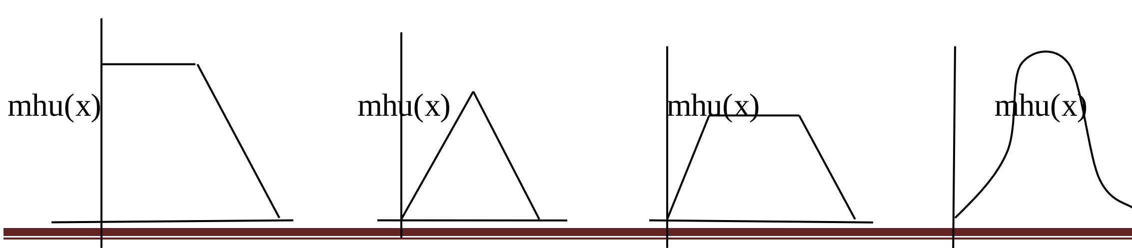
A membership function can also be given mathematically as

$$Mhu_{A\ comp}(x) = 1/(1+x)^2$$

Continuous membership function dictated by a mathematical function



Different shapes of membership functions exist. The shapes could be triangular, trapezoidal, curved or their variations as shown below.



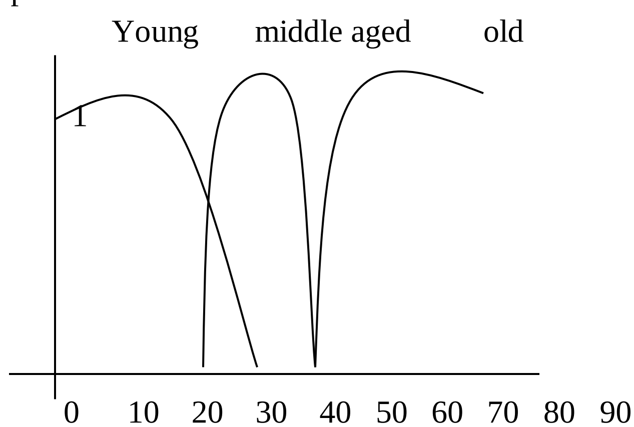
X x x

x

Example: Consider the set of people in the following age groups

0-10	40-50
10-20	50-60
20-30	60-70
30-40	70 and above

The fuzzy sets "young", "middle aged", and "old" are represented by the membership function graphs as illustrated below



Example of Fuzzy sets expressing "Young", "middle-aged", and "old"

Basic Fuzzy set Operations

- Union
- Intersection
- Complement
- Product of two fuzzy sets
- Equality
- Product of two fuzzy sets with a crisp number
- Power of a fuzzy set
- Difference
- Disjunctive sum

Properties of Fuzzy sets

Fuzzy sets follow some of the properties satisfied by crisp sets. Crisp sets can be thought of as special instances of fuzzy sets. Any fuzzy set A comp is a subset of the reference set x. Also, the membership of any element belonging to the null set Phi is 0 and the membership of any element belonging to the reference set is 1.

- Commutativity
- Associativity
- Distributivity
- Idempotence
- Identity
- Transitivity
- Involution
- De Morgan's Law

Point in a nutshell

- Fuzzy set theory is an effective tool to tackle the problem of uncertainty.
- In crisp logic, an event can take on only two values,either a 1 or 0 depending on whether its occurrence is true or false respectively. However, in fuzzy logic the event may take a range of values between 0 and 1.
- Crisp sets are fundamental to the study of fuzzy sets. The basic concepts include universal set, membership, cardinality of a set, family of sets, venn diagrams, null set, singleton set, power set, subset, and super set. The basic operations on crisp sets are union,intersection, complement, and difference.A set of properties are satisfied by crisp sets. Also, the concept of partition and covering result in the tow important rules, namely rule of addition and principle of inclusion and exclusion.
- Fuzzy sets support a flexible sense of membership and is defined to be the pair(x,mhu(x)) where mhu(x) could be discrete or could be described by a continuous function. The membership functions could be triangular, trapezoidal, curved or its variations.
- Fuzzy sets similar to crisp sets satisfy properties such as commutativity, associativity, distributivity, De Morgan's laws and so on.
- Crisp relation on sets are subsets of the Cartesian product of the given sets. A crisp relation associates the tuples by means of a relation. A Cartesian relation could be represented by a relation matrix.

Defuzzification:

In many situations, for a system whose output is fuzzy, it is easier to take a crisp decision if the output is represented as a single scalar quantity. This conversion of a fuzzy set to single crisp value is called defuzzification and is the reverse process of fuzzification.Several methods are available they are centroid method, centre of sums(COS), and means of maxima(MOM).

___________ is a completely characterized by its membership function
________function is a curve that defines how each point input is mapped to value
Membership value or degree of membership between _________
The ______ space is reffered to universe of discouse,a fancy name for simple concept
________membership function is specified with three parameters
________membership function is specified with four parameters
_______and_________membership functions have used in real time implementation
A________membership function is determined completely by its centre and width
________symbol represent the membership function centre
________symbol represent the membership function width
______and______membership function are popular for specifying fuzzy sets
_____is expressed as membership function based on definition of fuzzy intersection
_____ statement are used to formulate conditional statement comprise fuzzy logic
In If-then,If-part of the rule is called as_________
In If-then,then-part of the rule is called as_________
Antecedent is also called as_______
In general,_____forms of rule exist for any linguistic variables
The conditional statement can be expressed in_______ form
A________statement expressed by a human might involve compound rule structures
The process allows converting a numeric value into a fuzzy input is called _____
________rule may be decomposed and reduced to a number of simple canonical rules
There are _______ ways to do fuzzification
_______fuzzification is generally used in implementation where there is no noise
_______is the reverse process of fuzzification
The_____ of a fuzzy set is the process of conversion of a fuzzy quantity into a crisp value
________engineers have many different ways of defuzzifying
_______method is also known as the height method
The_______method is both simple and quick method
The_______ also referred to as centre of area and centroid method
_______method takes the peak value of fuzzy set and weighted sum of peak values
_____method is most used and physically appealing of all defuzzification methods
______ method determines centre of area below the combined membership function
_____method is only valid for symmetrical output membership functions
______ method is also called middle of maxima
______is faster than many defuzzification methods that are presently in use
______process involves the algebraic sum of individual output fuzzy sets
______is the process of formulating nonlinear mapping from input space to output space
The process of______involves all membership functions, fuzzy logic operators, if-then rules
The differences between these three fuzzy inferences, also called_________
There are____types of fuzzy inference, which have widely employed in various applications
The_____type fuzzy modelling was first proposed as first attempt to control a steam engine
_____rule of composition inferred output of each rule is chosen from minimum firing strength
_____rule of composition inferred output of each rule is a fuzzy set scaled down by its firing strength via algebraic product
The Sugeno fuzzy model, also known as the________
The value of the set membership is represented by_______
Fuzzy logic is usually represented as_______
_________fuzzy model was proposed by Takagi,Sugeno and Kang

_________in an effort to develop a systematic approach to generate fuzzy rules from a given input-output data set
In Mamdani's fuzzy model______values are used as input to convert a fuzzy set to a crisp value
_______,fuzzy model, consequent of each fuzzy if-then rule is represented by a
fuzzy set with a monotonic MF
__________is the most common rule of composition

f(x,y) is a first-order polynomial, the resulting system is called________Sugeno fuzzy model
f(x,y) is a constant, it is called as________Sugeno fuzzy model

_______operator is sometimes replaced with weighted sum operator to reduce computation
__________fuzzy model aggregates each rule's output by the method of weighted average

Speed and pressure of steam engine can be expressed with__________conditional statement
________system of rules where at least one rule must satisfied,connected by OR connectives

The ______membership function, each rule are aggregated using the graphical equivalent
A linguistic statement expressed by a human might involve_________ rule structures

Defuzzification of a fuzzy set is the process of conversion of a fuzzy quantity into____value

fuzzy set	logic	fuzzy logic	none of these
fuzzy	membership	boolean	activation
0 and 1	1 and 2	0 and 2	0
input	output	logic	none of these
Triangular	Rectangular	Trapezoidal	parallelized
Rectangular	Trapezoidal	parallelized	Triangular
Triangular,trapezoidal	trapezoidal,Rectangular	trapezoidal,Triangular	none of these
Triangular	Trapezoidal	Gaussian	sigmoid
c	σ	α	λ
σ	c	λ	α
Sigmoid,Gaussian	Triangular,trapezoidal	Gaussian,Generalized bell	Sigmoid,Triangular
An	A_s	A_1	A_2
If-Then	If	Switch	none of these
antecedent	consequent	descedent	none of these
antecedent	consequent	descedent	none of these
premise	descedent	linguistic	inference
three	two	one	four
statistical	mathematical	linguistic	none of these
linguistic	conditional	compound	statistical
fuzzification	Defuzzification	inference	none of these
conditional	compound	statistical	mathematical
one	two	three	four
Singleton	rounding off	max-membership	none of these
inference	Defuzzification	descedent	antecedent
inference	descedent	Defuzzification	antecedent
Fuzzy control	Fuzzy logic	Fuzzy relations	none of these
Mean-max	max-membership	Weighted average	Centre of gravity
Mean-max	max-membership	Weighted average	Centre of gravity
max-membership	Centre of gravity	Mean-max	Weighted average
max-membership	Centre of gravity	Weighted average	Mean-max
Centre of gravity	Weighted average	Mean-max	max-membership
Weighted average	Centre of gravity	Mean-max	max-membership
Weighted average	max-membership	Centre of gravity	Mean-max
Mean-max	max-membership	Centre of gravity	Weighted average
Centre of sums	Mean-max	Weighted average	max-membership
Mean-max	Centre of sums	Weighted average	max-membership
Inference	fuzzy logic	fuzzy set	none of these
fuzzy relation	fuzzy inference	fuzzy logic	fuzzy control
fuzzy models	fuzzy relations	fuzzy logic	none of these
two	three	four	one
Mamdani	Sugeno	Tsukamoto	none of these
max-min	max-product	if-then	none of these
max-min	max-product	if-then	none of these
TSK fuzzy model	fuzzy model	sug fuzzy model	none of these
discrete set	degree of truth	probabilities	both a & b
If-Then rules	if	if-then else rues	both a & b
Sugeno	Mamdani	Tsukamoto	none

Sugeno fuzzy model	Tsukamoto fuzzy model	Mamdani fuzzy inference	none of these
crisp	fuzzy	set	mean
Mamdani	Sugeno	Tsukamoto	none of these
Max-min	max-product	if-then	none of these
zero-order	first-order	second-order	third-order
zero-order	first-order	second-order	third-order
weighted average	weighted product	Weighted mean	none of these
Sugeno	Tsukamoto	Mamdani	both a & b
if-then	linguistic	if-then else	switch
Disjunctive	conjuctive	consequent	none of these
aggregated	truncated	consequent	minimum
compound	statistical	conditional	canonical
fuzzy	set	crisp	none of these

fuzzy set
membership
0 and 1
input
Triangular
Trapezoidal
Triangular,trapezoidal
Gaussian
c
σ
Gaussian,Generalized bell
As
If-Then
antecedent
consequent
premise
three
mathematical
linguistic
fuzzification
compound
two
Singleton
Defuzzification
Defuzzification
Fuzzy control
max-membership
max-membership
Centre of gravity
max-membership
Centre of gravity
Centre of gravity
Weighted average
Mean-max
Centre of sums
Centre of sums
Inference
fuzzy inference
fuzzy models
three
Mamdani
max-min

max-product
TSK fuzzy model
degree of truth
If-Then rules
Sugeno

Sugeno fuzzy model
crisp

Tsukamoto
Max-min

first-order
zero-order

weighted average
Tsukamoto

linguistic
Disjunctive

truncated
compound

crisp

Unit v: Genetic Algorithm (GA)- Basic concepts, working principle, procedures of GA, flowchart of GA, Genetic representations, (encoding) Initialization and selection, genetic operators, mutation, Generational cycle, applications

Introduction: Decision making features occur in all fields of human activities such as scientific and technological and affect every sphere of life. Engineering design which entails sizing, dimensioning, and detailed element planning . For example, an aircraft wing can be made from aluminium or steel and once material shape are chosen, there are many methods of devising the required internal structure. In civil engineering, designing a roof to cover large area devoid of intermediate columns requires optimal designing.

The aim is to make objective function a maximum or minimum. It is required to find an element X0 in A if it exists such that

$$F(X_0) <= F(X) \text{ for minimization}$$

$$F(X) <= F(X_0) \text{ for maximization}$$

The following question may arise in determining optimal solution

- Does an optimal solution exist?
- Is it unique?
- What is the procedure?
- How the sensitive the optimal solution is?
- How the solution behaves for small changes in parameters?

Since 1940 several optimization problems have not been fully solved by classical procedures including:

1. Linear programming
2. Transportation
3. Assignment
4. Nonlinear programming
5. Dynamic programming
6. Inventory
7. Queuing
8. Replacement
9. Scheduling

The classification of optimization techniques is shown in below figure 1. We have been following traditional search technique for solving non-linear equations. The second figure 2 shows the classes of both traditional and non-traditional search techniques . Any engineering problem will have large number of solutions out of which some are feasible and some are feasible. The designer’s task is to get the best solution out of the feasible solution. The complete set of feasible solutions constitute feasible design space and the progress towards the optimal design involves some kind of search within the space(combinatorial optimization). The search is of two kinds namely deterministic and stochastic.

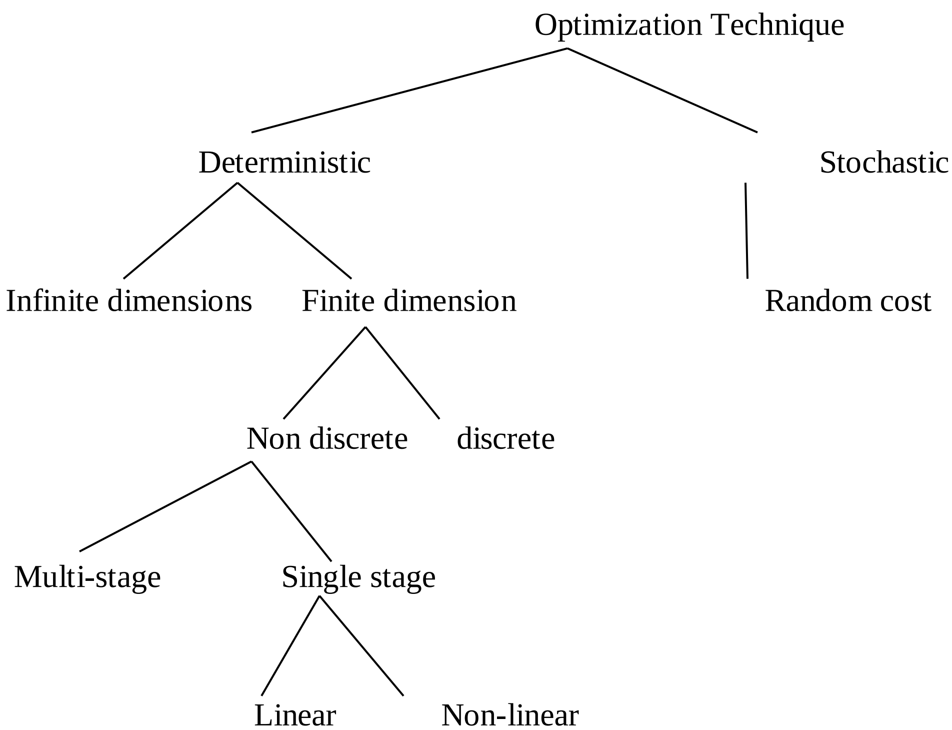


Figure 1 – Classification of Optimization Technique

In the case of deterministic search , algorithm methods such as steepest gradient methods are employed, whereas in stochastic approach random variables are introduced. Whether the search is deterministic or stochastic it is possible to improve the reliability of results. Reliability refers to getting result near optimum. A transition rule must be used to improve reliability. Algorithms vary according to the transition rule used to improve the result. Non-traditional search and optimization methods have become popular in engineering optimization problems in recent days. These algorithms include:

- Simulated annealing
- Ant colony optimization
- Random cost

- Evolution strategy
- Genetic algorithms
- Cellular automata

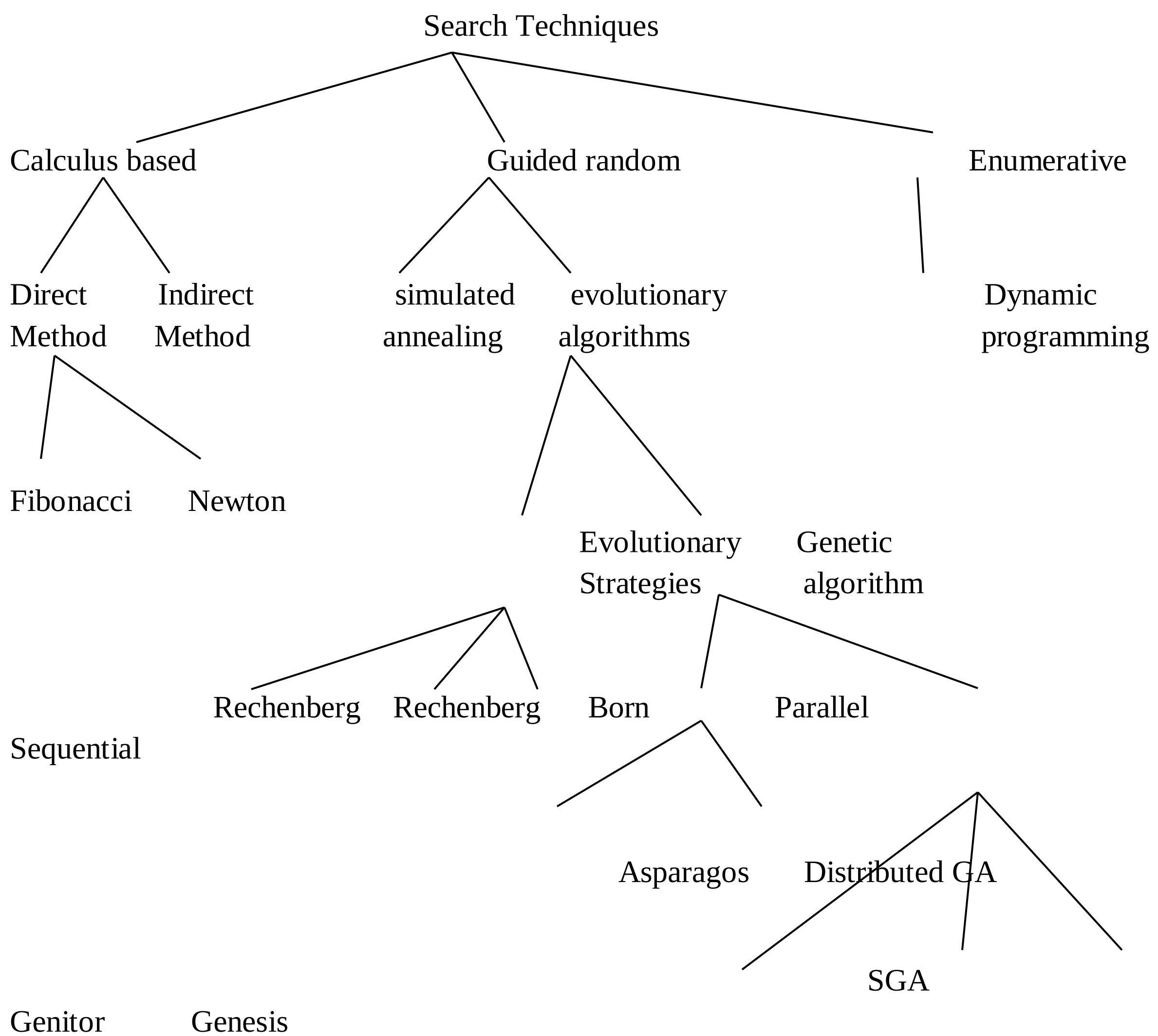


Figure 2 – Classes of search techniques

Simulated annealing mimics the cooling phenomenon of molten metals to constitute a search procedure. Genetic algorithm and evolutionary strategies mimic the principle of natural genetics and natural selection to construct search and optimization procedures. The collective behavior that emerges from group of social insects such as ants, bees, wasps, and termites has been dubbed as swarm intelligence. The foraging of ants has led to a novel algorithm called Ant colony optimization for re-routing network traffic in busy telecommunication systems. Random cost method is a stochastic algorithm which moves as enthusiastically uphill as down-hill.

BASIC CONCEPTS

Genetic algorithms are good at taking larger, potentially huge, search spaces and navigating them looking for optimal combinations of things and solutions which we might not find in a life time.

Genetic algorithms need design space to be converted into genetic space. Genetic algorithms work with a coding of variables. The advantage of working with a coding of variable space is that coding discretizes the search space even though the function may be continuous. A more striking difference between genetic algorithms and most of the traditional optimization methods is that GA uses a population of points at one time in contrast to the single point approach by traditional optimization methods. This means that GA processes a number of designs at the same time. To improve the search direction in traditional optimization methods, transition rules are used and they are deterministic in nature but GA uses randomized operators. Random operators improve the search space in and adaptive manner. The important aspect of GA are:

1. Definition of objective function
2. Definition and implementation of genetic representation
3. Definition and implantation of genetic operators

Once these three have been defined, the GA should work fairly well. We shall be able to perform by different variations, improve the performance, find multiple optima(species if they exist) or parallelize the algorithm.

Biological background:

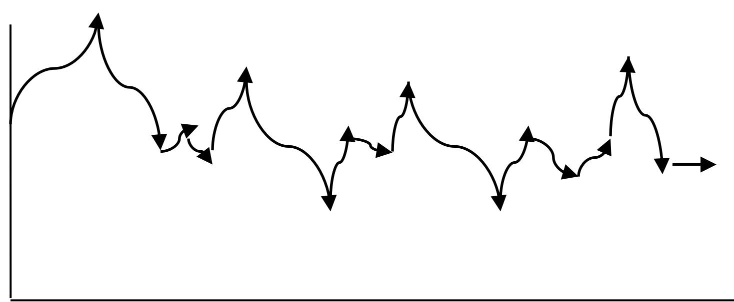
All living organisms consist of cells. In each cell, there is a set of chromosomes which are strings of DNA and serve as a model for the whole organisms. A chromosome consists of genes on blocks of DNA as shown below in the diagram. Each gene encodes a particular pattern. Each gene encodes a trait (e.g) colorof eyes.possible setting of traits(bluish brown eyes) are called alleles. Complete set of genetic material is called genome and a particular set of genes in genome is called genotype. The genotype is based on organism's phenotype (development after birth), its physical and mental characteristics such as eye color, intelligence and so on.

Creation of offsprings

During the creation of offspring, recombination occurs (due to cross over) and in that process genes from parents form whole new chromosome in some way. The new created offspring can then be mutated. Mutation is the modified form of DNA. These changes are mainly caused by errors in copying genes from parents. The fitness of an organism is measured by means of success of organism in life.

Search space

If we are solving some problem, we work forward some solution which is the best among others. The space for all possible feasible solutions is called search space. Each solution can be marked by its value of the fitness of the problem. 'looking for a solution' means looking for extrema (either maxima or minima) in search space. The search space can be known by the time of solving a problem and we generate other points as the process of finding the solution continues as shown in below fig.



Examples of search space

The problem is that, search space is complicated and one does not know where to look for the solution or where to start from and this is where genetic algorithm is useful. GA's are inspired by Darwinian theory of survival of the fittest. Algorithm is started with a set of solution (represented by chromosomes) called populations. Solutions for one population are taken and used to form a new population. This is motivated by a hope that new population will be better than the old one. Solutions which are selected to form new population (offspring), are selected according to their fitness. The more suitable they are, the more chances they have to reproduce. This is repeated until some conditions(number of populations) for improvement of best solution are satisfied.

Working principle

To illustrate the working principle of GA we first consider unconstrained optimization problem.Then we will see how GA can be used to solve a constrained optimization problem. Let us consider the following maximization problem.

Maximize $f(X)$

$X_i(L) <= X_i <= X_i(U)$ for $i=1,2,3\ldots..N$

If we want to minimize f(X), for f(X) >0, then we write the objective functions as

Minimize 1/1+f(X)

If f(X)<0 instead of minimizing f(X), maximize{-f(X)]. Hence, both maximization and minimization can be handled by GA.

I the same problem is solved by multiple regression analysis, given k independent variables or regressing the dependent variables2(k+1)-1 including the intercept which are given in table.

Subsets for regression analysis

Variable	Subsets
3	7
3	15
-	-
9	1023
-	-
19	10,48,578

What is encoding in GA?
Binary Encoding. Binary encoding is the most common one, mainly because the first research of GA used this type of encoding and because of its relative simplicity. In binary encoding, every chromosome is a string of bits - 0 or 1.

What is chromosome in genetic algorithm?
In genetic algorithms, a chromosome (also sometimes called a genotype) is a set of parameters which define a proposed solution to the problem that the genetic algorithm is trying to solve. The set of all solutions is known as the population.

What is cross-over in GA?
Crossover (genetic algorithm) From Wikipedia, the free encyclopedia. In genetic algorithms and evolutionary computation, crossover, also called recombination, is a genetic operator used to combine the genetic information of two parents to generate new offspring.

What is selection in GA?
election is the stage of a genetic algorithm in which individual genomes are chosen from a population for later breeding (using the crossover operator).

What is mutation in GA?
Mutation is a genetic operator used to maintain genetic diversity from one generation of a population of genetic algorithm chromosomes to the next. It is analogous to biological mutation. Mutation alters one or more gene values in a chromosome from its initial state.

What is meant by genetic algorithm?
The genetic algorithm is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution. The genetic algorithm repeatedly modifies a population of individual solutions.

Why do we use genetic algorithm?
Genetic Algorithms and What They Can Do For You. A genetic algorithm solves optimization problems by creating a population or group of possible solutions to the problem. After the genetic algorithm mates fit individuals and mutates some, the population undergoes a generation change.

Are Genetic Algortihms machine learning?
In many respects, genetic algorithms are on the dumb and uncontrolled end ofmachine learning methods. ... Genetic algorithms are important in machine learning for three reasons. First, they act on discrete spaces, where gradient-based methods cannot be used.

Who invented genetic algorithm?
Genetic algorithms in particular became popular through the work of John Holland in the early 1970s, and particularly his book **Adaptation in Natural and Artificial Systems** (1975). His work originated with studies of cellular automata, conducted by Holland and his students at the University of Michigan.

What is genetic algorithm in soft computing?
A genetic algorithm is a heuristic search method used in artificial intelligence andcomputing. It is used for finding optimized solutions to search problems based on the theory of natural selection and evolutionary biology. Genetic algorithms are excellent for searching through large and complex data sets.

What is heuristic method?
In computer science, artificial intelligence, and mathematical optimization, a**heuristic** (from Greek εὑρίσκω "I find, discover") is a technique designed for solving a problem more quickly when classic **methods** are too slow, or for finding an approximate solution when classic **methods** fail to find any exact solution. The term **heuristic** is used for **algorithms** which

find solutions among all possible ones ,but they do not guarantee th best will be found, therefore they may be considered as approximately and not accurate **algorithms**.

Encoding

Encoding of chromosomes is one of the problems, when you are starting to solve problem with GA. Encoding very depends on the problem.

Here, introduced some encoding techniques, which have been already used as successful.

Binary Encoding

Binary encoding is the most common, mainly because first works about GA used this type of encoding.

In **binary encoding** every chromosome is a string of **bits**, **0** or **1**.

Chromosome A	101100101100101011100101
Chromosome B	111111100000110000011111

Example of chromosomes with binary encoding

Binary encoding gives many possible chromosomes even with a small number of alleles. On the other hand, this encoding is often not natural for many problems and sometimes corrections must be made after crossover and/or mutation.

Example of Problem: Knapsack problem
The problem: There are things with given value and size. The knapsack has given capacity. Select things to maximize the value of things in knapsack, but do not extend knapsack capacity.
Encoding: Each bit says, if the corresponding thing is in knapsack.

Permutation Encoding

Permutation encoding can be used in ordering problems, such as travelling salesman problem or task ordering problem.

In **permutation encoding**, every chromosome is a string of numbers, which represents number in a **sequence**.

Chromosome A	1 5 3 2 6 4 7 9 8
Chromosome B	8 5 6 7 2 3 1 4 9

Example of chromosomes with permutation encoding

Permutation encoding is only useful for ordering problems. Even for this problems for some types of crossover and mutation corrections must be made to leave the chromosome consistent (i.e. have real sequence in it).

Example of Problem: Travelling salesman problem (TSP)
The problem: There are cities and given distances between them.Travelling salesman has to visit all of them, but he does not to travel very much. Find a sequence of cities to minimize travelled distance.
Encoding: Chromosome says order of cities, in which salesman will visit them.

Value Encoding

Direct value encoding can be used in problems, where some complicated value, such as real numbers, are used. Use of binary encoding for this type of problems would be very difficult.

In **value encoding**, every chromosome is a string of some values. Values can be anything connected to problem, form numbers, real numbers or chars to some complicated objects.

Chromosome A	1.2324 5.3243 0.4556 2.3293 2.4545
Chromosome B	ABDJEIFJDHDIERJFDLDFLFEGT
Chromosome C	(back), (back), (right), (forward), (left)

Example of chromosomes with value encoding

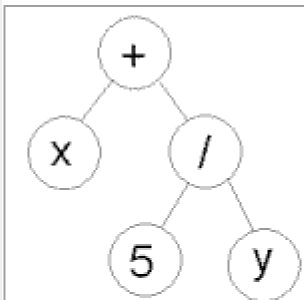
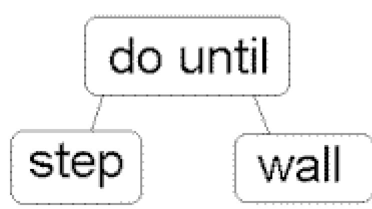
Value encoding is very good for some special problems. On the other hand, for this encoding is often necessary to develop some new crossover and mutation specific for the problem.

Example of Problem: Finding weights for neural network
The problem: There is some neural network with given architecture. Find weights for inputs of neurons to train the network for wanted output.
Encoding: Real values in chromosomes represent corresponding weights for inputs.

Tree Encoding

Tree encoding is used mainly for evolving programs or expressions, for **genetic programming**.

In **tree encoding** every chromosome is a tree of some objects, such as functions or commands in programming language.

Chromosome A	Chromosome B
	
(+ x (/ 5 y))	(do_until step wall)

Example of chromosomes with tree encoding

Tree encoding is good for evolving programs. Programing language LISP is often used to this, because programs in it are represented in this form and can be easily parsed as a tree, so the crossover and mutation can be done relatively easily.

Example of Problem: Finding a function from given values
The problem: Some input and output values are given. Task is to find a function, which will give the best (closest to wanted) output to all inputs.
Encoding: Chromosome are functions represented in a tree.

Crossover and Mutation

Introduction

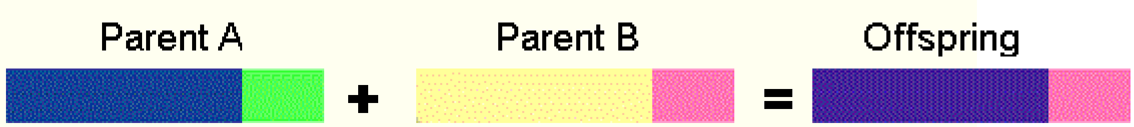
Crossover and mutation are two basic operators of GA. Performance of GA very depends on them. Type and implementation of operators depends on encoding and also on a problem.

There are many ways how to do crossover and mutation. In this chapter are only some examples and suggestions how to do it for several encoding.

Binary Encoding

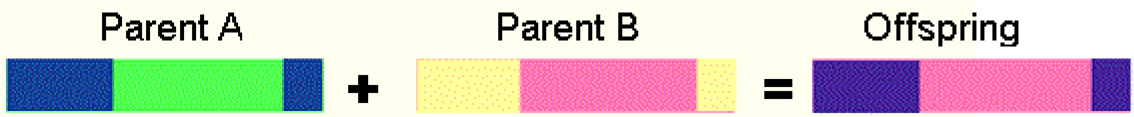
Crossover

Single point crossover - one crossover point is selected, binary string from beginning of chromosome to the crossover point is copied from one parent, the rest is copied from the second parent



11001011+11011**111** = **11001111**

Two point crossover - two crossover point are selected, binary string from beginning of chromosome to the first crossover point is copied from one parent, the part from the first to the second crossover point is copied from the second parent and the rest is copied from the first parent



110010**11** + 11**0111**11 = **11011111**

Uniform crossover - bits are randomly copied from the first or from the second parent

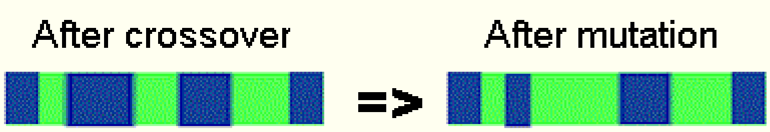
1**10**010**11** + **11**0**111**01 = 11011111

Arithmetic crossover - some arithmetic operation is performed to make a new offspring

11001011 + 11011111 = 11001001 (AND)

Mutation

Bit inversion - selected bits are inverted



1**1**001001 => 1**0**001001

Permutation encoding

Crossover

Single point crossover - one crossover point is selected, till this point the permutation is copied from the first parent, then the second parent is scanned and if the number is not yet in the

offspring it is added
Note: there are more ways how to produce the rest after crossover point

(**1 2 3 4 5** 6 7 8 9) + (**4 5 3 6 8** 9 7 2 1) = (**1 2 3 4 5** 6 8 9 7)

Crossover

All crossovers from **binary encoding** can be used

Mutation

Adding a small number (for real value encoding) - to selected values is added (or subtracted) a small number

(1.29 5.68 **2.86 4.11** 5.55) => (1.29 5.68 **2.73 4.22** 5.55)

Mutation

Order changing - two numbers are selected and exchanged

(1 **2** 3 4 5 6 **8** 9 7) => (1 **8** 3 4 5 6 **2** 9 7)

Value encoding

Crossover

All crossovers from **binary encoding** can be used

Mutation

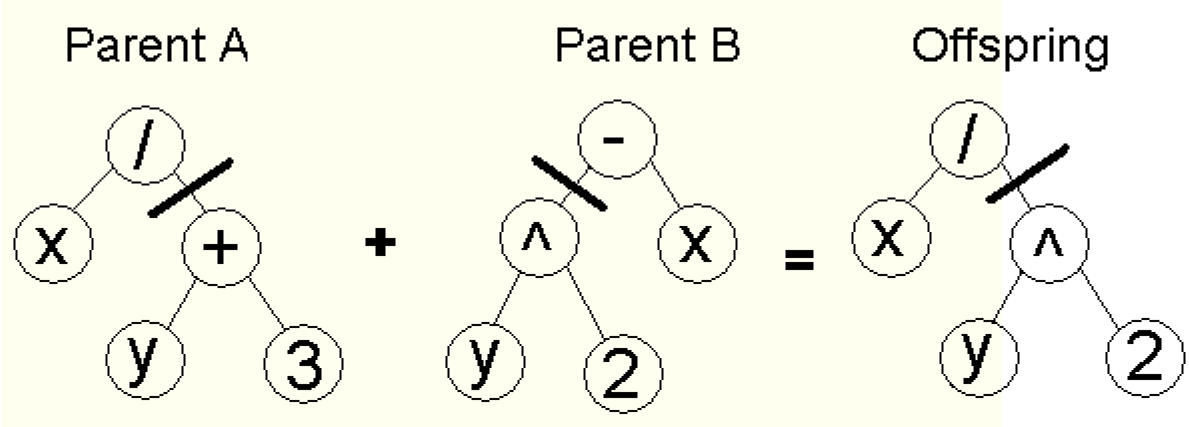
Adding a small number (for real value encoding) - to selected values is added (or subtracted) a small number

(1.29 5.68 **2.86 4.11** 5.55) => (1.29 5.68 **2.73 4.22** 5.55)

Tree encoding

Crossover

Tree crossover - in both parent one crossover point is selected, parents are divided in that point and exchange part below crossover point to produce new offspring



Mutation

Changing operator, number - selected nodes are changed

Travelling Salesman Problem

About the Problem

Travelling salesman problem (TSP) has been already mentioned in one of the previous chapters. To repeat it, there are cities and given distances between them.Travelling salesman has to visit all of them, but he does not to travel very much. Task is to find a sequence of cities to minimize travelled distance. In other words, find a minimal Hamiltonian tour in a complete graph of ***N*** nodes.

Implementation

Population of 16 chromosomes is used. For encoding these chromosome permutation encoding is used - in chapter about encoding you can find, how to encode permutation of cities for TSP. TSP is solved on complete graph (i.e. each node is connected to each other) with euclidian distances. Note that after adding and deleting city it is necessary to create new chromosomes and restart whole genetic algorithm.

You can select crossover and mutation type. I will describe what they mean.

Crossover

- One point - part of the first parent is copied and the rest is taken in the same order as in the second parent
- Two point - two parts of the first parent are copied and the rest between is taken in the same order as in the second parent
- None - no crossover, offspring is exact copy of parents

Mutation

- Normal random - a few cities are chosen and exchanged
- Random, only improving - a few cities are randomly chosen and exchanged only if they improve solution (increase fitness)
- Systematic, only improving - cities are systematically chosen and exchanged only if they improve solution (increase fitness)
- Random improving - the same as "random, only improving", but before this is "normal random" mutation performed
- Systematic improving - the same as "systematic, only improving", but before this is "normal random" mutation performed
- None - no mutation

Parameters of GA

This chapter should give you some basic recommendations if you have decided to implement your genetic algorithm. These recommendations are very general. Probably you will want to experiment with your own GA for specific problem, because today there is no general theory which would describe parameters of GA for *any* problem.

Recommendations are often results of some empiric studies of GAs, which were often performed only on binary encoding.

- **Crossover rate**
  Crossover rate generally should be high, about **80%-95%**. (However some results show that for some problems crossover rate about 60% is the best.)
- **Mutation rate**
  On the other side, mutation rate should be very low. Best rates reported are about **0.5%-1%**.
- **Population size**
  It may be surprising, that very big population size usually does not improve performance of GA (in meaning of speed of finding solution). Good population size is about **20-30**, however sometimes sizes 50-100 are reported as best. Some research also shows, that best population size depends on encoding, on **size of encoded string**. It means, if you have chromosome with 32 bits, the population should be say 32, but surely two times more than the best population size for chromosome with 16 bits.
- **Selection**
  Basic **roulette wheel selection** can be used, but sometimes rank selection can be better. Check chapter about selection for advantages and disadvantages. There are also some

more sophisticated method, which changes parameters of selection during run of GA. Basically they behaves like simulated annealing. But surely **elitism** should be used (if you do not use other method for saving the best found solution). You can also try steady state selection.

- **Encoding**
  Encoding **depends on the problem** and also on the size of instance of the problem. Check chapter about encoding for some suggestions or look to other resources.
- **Crossover and mutation type**
  Operators depend on encoding and on the problem.

Applications of GA

Genetic algorithms has been used for difficult problems (such as NP-hard problems), for machine learning and also for evolving simple programs. They have been also used for some art, for evolving pictures and music.

Advantage of GAs is in their parallelism. GA is travelling in a search space with more individuals (and with genotype rather than phenotype) so they are less likely to get stuck in a local extreme like some other methods.

They are also easy to implement. Once you have some GA, you just have to write new chromosome (just one object) to solve another problem. With the same encoding you just change the fitness function and it is all.On the other hand, choosing encoding and fitness function can be difficult.

Disadvantage of GAs is in their computational time. They can be slower than some other methods. But with todays computers it is not so big problem.

To get an idea about problems solved by GA, here is a short list of some applications:

- Nonlinear dynamical systems - predicting, data analysis
- Designing neural networks, both architecture and weights
- Robot trajectory
- Evolving LISP programs (genetic programming)
- Strategy planning
- Finding shape of protein molecules
- TSP and sequence scheduling
- Functions for creating images

References:

[1] Neural networks, fuzzy logic, and Genetic algorithms Synthesis and applications

[2]http://www.obitko.com/tutorials/genetic-algorithms/recommendations.php

Questions
Benefits of GA are ____________________.
_________ is a process in which a given bit pattern is transformed into another bit pattern by means of logical bit
Reproduction operator is also known as __________.
GA is used when ?
This method simulates the process of slow cooling of molten metal to achieve the minimum function value in a mi
_________ means the genes from the already discovered good individuals are exploited while promising the new
Issues for GA practitioners are
In tournament selection, if the _________ is too low, convergence rate will be slow and it will take a long time to f
__________ first ranks the population and considers every chromosome receives fitness from ranking
The essential sites for recombination are known as
The replication rate remains same for all the phages irrespective of what sequence is there in a phage is the given
According to _______, " Genetic algorithm are rich-rich in applications across a large and growing number of disci
There are some phages which don't preferentially transduce some special regions of phage genome,these phages
GA techniques are ideally suited for
In rank selection, the worst and the best will have fitness N (number of chromosomes in population)
The bigger part of chromosome should survive to next generation called _______
First the best chromosomes or few best chromosomes are copied to new population called __________
In genetic algorithm, every ________ is compared with a gene in biological term.
Which of the statements don't hold true for the forensics and the amplifications carried out
The genetic relatedness between organisms can be identified by studying the band patterns when different PCR p
A simple GA largely uses three basic operator which are __________.
A single random cross-site is selected which is common for both parent-1 and parent-2 for cross-over called _____
What is problem associated with historical DNA samples
Two random sites are chosen and the contents bracketted by these sites are exchanged between two mated pare
In a ____________ operator, each bit from either parent is selected with a probability of 0.5 and then interchange
How can specify of primer annealing be increased?
There are basically two types of communication laboratory and external.if PCR product is found to be contained b
Which can be used as a precaution in order to minimize contamination?
During amplification,there are chances of having a product of mixture of different sequences .there are various wa
If the template DNA belongs to several individual rather than single one this type of hetrogeneity is known as
A string from the population is selected and the bits between two random sites are inverted as new population
Any two or three bits at random in order are selected and the previous bits are duplicated are called _________.
Genes between two cross-sites are deleted and regenerated randomly called ________.
In cross-over operation, if the bits of parents are segregated and then crossed over to produce offspring called ___
___________ is the probability of mutation which is used to calculate number of bits to be muted
___________ features occur in all fields of human activities such as scientific and technological and affect every s
Any engineering problem will have a large number of solutions out of which some are feasible and some are _____
In _______________, algorithm methods such as steepest gradient methods are employed.
It is possible to improve the reliability of the results where reliability means getting the results ________.
Genetic algorithms and evolutionary strategies mimic the principle of _________ to construct search and optimiza
____________ mimics the cooling phenomenon of molten metals to constitute a search procedure.
The collective behaviour that emerges from a group of social insects such as ants, bees, wasps and termites called
____________ is a stochastic algorithm which moves as enthusiastically uphill and down-hill.
The idea of evolutiary computing was introduced by ___________________.
Apart from structural engineering problems ____________ is applied in many other fields like biology, pattern rec
____________ needs design space to be converted in to genetic space.

Genetic operators are ____________.
Genetic algorithms are inspired by _________ of the survival of the fittest.
Non traditional search and optimization methods have become popular in engineering problems in recent past are
The difference between genetic algorithms and traditional optimization is that uses a ________ of points at one ti
GA'S uses ____________ to improve the search direction.
During the creation of _________, recombination occurs and in that process genes from parents from a whole nev
Mutation refers to ________ modification.
The __________ for all possible feasible solutions for a problem is called search space.
The _______ of an organism is measured by means of success of organuism in life.
solutions which are selected to form new population based on fitness function are called __________.
GA'S uses ____________ to improve the search direction.
GA'S are suitable for solving ______________ problems.
There are cities and given distances between them, it is required to visit all cities only once this problem is called _
According to genetic theory, the ________ one should survive and create new offspring.

opt1
multi-objective optimization, good for noisy environment
masking
selection operator
alternate solution are too slow or overly complicated
Rank selection
selection pressure
termination criterion
machine learning
Rank selection
chi sites
TRUE
gold berg
transducing phages
travelling sales man problem
1
Rank selection
rank selection
chromosome
In the case of forensics,conventional methods such as southern blotting are used very effectively
restriction fragment length polymorphism (RFLP)
reproduction, cross-over and mutation
Single site cross over
they are less in amount thus amplification is difficult
sinle site cross-over
Single site cross over
use of short primers
TRUE
careful use and design of pipettes
direct sequencing can't be used in the case if the template DNA is heterozygnous at the locus
heterzygosity
inversion
inversion
segregation
segregation
mutation rate
objective function
infeasible solution
deterministics search
optimum
tion procedures.
simulated annealing
swarm intelligence
random cost
Rechenberg
Genetic algorithms
Genetic algorithms

cross-over and mutation
darwinian theory
random cost, genetic algorithm, cellular automata
search space
random operators
offspring
DNA
space
fitness
offspring
space
maximization
Travelling salesman problem
best

opt2
termination condition
shift operation
mutation operator
need an exploratory tool to examine new approaches
roulette-wheel selection
population diversity
population size
selective pressure
Tournament selection
gam sites
FALSE
rechen berg
specialized transducing phages
job scheduling
0
Tournament selection
Tournament selection
4 bits
In cases of bone fragments which contain less than 300 nucleotides conventional methods cant be applied as they
amplified fragment length polymorphism (AFLP)
inheritance, migration and mating
two point cross over
because the samples are very old,there can be contamination
two- point cross-over
two point cross over
raising temperature
FALSE
placing the pre-PCRand post-PCR stages in same rooms
sequencing can be used if the template DNA is hetrozygous at the locus
product heterogenityl
deletion
mass inversion
deletion and regeneration
deletion
mutation order
decision making
optimal solution
probablistic search
efficient
natural selection
evolutionary algorithms
artificial intelligence
linear
Gogos schleuter
fuzzy logic
neural network algorithms

offspring and search space
Rechenberg theory
random cost, linear and non-linear
genetic space
boolean operators
population
genome
recombination
search space
seed chromosome
boolean operators
minimization
dynamic programming
worst

opt3
performance and scalability
cross over
cross-over operator
benefits of GA technology meet key problem requirements
tournament selection
convergence rates
all the above
population diversity
roulette wheel selection
red sites
not determined
roulette
generalized transducing phages
routing problem
N
elitism
elitism
substring
the poor condition of DNA also makes the PCR amplification difficult
random amplification of polymorphic DNA(RAPD)
deletion, dominance and segregation
multi point cross-over
they degrade during repeated cooling and heating cycles
multi-point cross over
uniform cross over
adjusting the concentration of sodium ions
cant be said
extracting the DNA along with urface layers
if cloning is done before sequencing then its detected via using only a single clone for sequnecing
populationhetrogenity
duplication
deletion and duplication
cross over and inversion
inversion
flipping 0 into 1

combinatorial optimization
linear search
poor result
artificial selection
dynamic programming
hybrid intelligence
deterministic
tanese
neural networks
evolutionary algorithms

population and generation
asparagos theory
fibonacci and newton search
population
fuzzy operators
mutation
chromosome
traits
traits
genome
fuzzy operators
uncertain
transportation
average

opt4
selection and deletion policies
none of the above
none of the above
all the above
boltzman selection
none of the above
none of the above
reproduction
Elitism
rec sites
cant be said
none of the above
transforming phages
engineering design problems
none of the above
steady-state selection
none of the above
bit
microsatellites composed of simplify varying repeates of CA sequennces is used
polymorphism
sharing, mating and migration
none of the above
as the samples are old,the standard sequences for comparision is not present
uniform cross-over
matric cross over
using polymerase wirh proof reading activity
not determined
use of primers carefully is not very important
in the case of several recombinants are used it cant go undetected
template hetrogenity
segregation
segregation
none of the above
regeneration
none of the above
scheduling
none of the above
non-linear search
nearest result
hybrid selection
none of the above
none of the above
none of the above
none of the above
none of the above
none of the above

none of the above
genitor theory
random cost, fuzzy logic and back propagation networks
population
none of the above
cross-over
none of the above
none of the above
none of the above
none of the above
none of the above
none of the above
none of the above
none of the above

answer
multi-objective optimization, good for noisy environment
masking
selection operator
all the above
boltzman selection
population diversity
all the above
selective pressure
rank selection
chi sites
FALSE
gold berg
generalized transducing phages
engineering design problems
N
steady-state selection
elitism
bit
In the case of forensics,conventional methods such as southern blotting are used very effectively
random amplification of polymorphic DNA(RAPD)
reproduction, cross-over and mutation
single site cross over
because the samples are very old,there can be contamination
two point cross-over
uniform cross over
raising temperature
FALSE
careful use and design of pipettes
sequencing can be used if the template DNA is hetrozygous at the locus
populationhetrogenity
inversion
deletion and duplication
deletion and regeneration
segregation
mutation rate
decision making
infeasible solution
deterministic search
optimum
natural selection
simulated anealing
swarm intelligence
random cost
rechenberg
genetic algorithms
genetic algorithms

cross-over and mutation
darwinian theory
random cost, genetic algorithm, cellular automata
population
random operators
offspring
DNA
space
fitness
offspring
space
maximization
Travelling
best

Register__________________

(16CTU603A)

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be University)

(Established Under Section 3 of UGC Act 1956)

Coimbatore – 641 021

B.Sc DEGREE EXAMINATION

FIRST INTERNAL EXAMINATION – DECEMBER2018

SIXTH SEMESTER

COMPUTER TECHNOLOGY

SOFT COMPUTING

Duration: 2 Hours

Date&Session : Maximum: 50 Marks

PartA[20 × 1 = 20 Marks]

(Answer All the Questions)

1. The guiding principle of soft computing is to ______________
   a) exploit the intolerance for imprecision, uncertainty and partial truth
   b) exploit the tolerance for imprecision, uncertainty and truth
   c) exploit the tolerance for imprecision, certainty and partial truth
   d) exploit the tolerance for imprecision, uncertainty and partial truth
2. The synergistic integration of two or more technologies called
   a) Hard computing technology b)Soft computing technology c)Hybrid technology
   b) None of the above
3. An element either belongs to or does not belong to a set are termed as

a)fuzzy setb)crisp setc)super set d)null set

4. Neural networks are trained with known samples of a problem to acquire _________ about it.

a)datab)knowledge c)experience d)technical inputs

5. Choose the right answer for NP-Hard problems

a)matrix addition, subtraction, multiplication and division

b)shortest path determination, greedy method and decision tree classifier

c)Travelling salesman, knapsack problem and konieg'sbridge problem

d)Travelling sales man, knapsack and depth first search

6. A kind of computing capable of producing accurate results even if the supplied data is noisy

a)hard computing b)soft computing c)hybrid computing d)Genetic computing

7. Neural network is a _________ models of biological neuron system, which is _______ made up of highly interconnected neural element.

a)simplified and massively parallel b)complex and massively parallel architecture

c)simplified and massively parallel distributed processing system

d)complex and massively parallel distributed processing system

8. The stability-plasticity property holds good for _________

a)neural network and fuzzy systems b)neural network and genetic systems

c)only for traditional programming d)none of the above

9. Summation of weighted inputs in artificial neural network is called

a) sum-of-product b)product-of-sum c)signum function d)threshold function

10. A graph G is an ordered 2-tuple(V,E) consisting of a set V of vertices and a set E of edges with a direction called ______________.

a)null graph b)graph c)bi-directional graph d)di-graph

11.The synaptic links carrying the weights connect every input neuron to the output neuron but not vice-versa are called ________.

a)back propagation network b)feed forward network

c)hopfield network d)counter propagation network

12.The ____________ is a computational model of the retina and of the eye.

a)neocognitron b)brain-state-in-a-box c)perceptron d)perception

13. Sets of points in two –dimensional space are ____________ if the sets can be separated by a straight line.

a)non-linearly separable b)linearly separable c)both linear and non-linearly separable

d)none of the above

14. Multilayer perceptron have the ability to predict weights for a ______________.

a)linear separable task b)non-linear separable task c)both a & b d)none of these

15. ___________ is an application shows remarkable progress in recognition.

a) character recognition b)speech recognition c)character and speech recognition

d) none of these

16. Convergence refers to

a) dispersion of multiple valuesb) arriving at exact solution

c)changing target output d)training the algorithm to learn

17. When a target output is not given to an algorithm but algorithm learns on its adaptation called________.

a)hebbian learning b)unsupervised learning c)supervised learning d) both b & d

18. The average weight of a brain is about _______.

a) 100 grams b)1500 grams c)1.2 kg d)none of these

19. Learning rate co-efficient determines the size of weight adjustments made at each iteration and hence influences the ________________.

a)rate of convergence b)convergence c)divergence d) both b& c

20. classification of soil is an application of ________________.

a)Single layer feed forward network b)multi-layer feed forward network

c)back propagation network d) both b & c

Part B[3 *2 = 6 Marks]

(Answer All the Questions)

21. State the function of synapse in a neuron.

22. What is activation function?.

23. Give an example for linearly separable task in soft computing.

Part C[3 * 8 = 24 Marks]

(Answer All the Questions)

24.a)Explain the model of Artificial Neural Network.

(OR)

b) Explain the functions of a biological human brain with a neat sketch.

25.a)Discuss about single layer feed forward network.

(OR)

b)Write short notes on i) Supervised learning ii)Reinforced learning

26.a)Write an algorithm for fixed increment perceptron learning for a classification problem with n input features (x1,x2,x3…….xn) and two input classes(0/1).

(OR)

b)Discussthe applications of back propagation neural network.

Register______________

(16CTU603A)

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be university Established Under Section 3 of UGC Act 1956)

Coimbatore-641 021

B.Sc DEGREE EXAMINATION

SECOND INTERNAL EXAMINATION – January 2019

Sixth Semester

Duration: 2 Hours Soft Computing

Date & Session: Maximum:50 Marks

Part A (20*1=20 Marks)

(Answer all the Questions)

1. __________ one of the major classes of neural networks which are imitations of the human brain's ability to associate patterns.
   a)Back propagation networks b)associative memories c)fuzzy learning d)a&b
2. Application of _____________ to real world problems using bipolar coding called as recognition of characters.
   a)associative memory b)single perceptron c)Neuronet d)none of the above
3. The ________ is a set which, with reference to a particular context, contains all possible elements having the same characteristics and from which sets can be formed.
   a)crisp b)universe of discourse c)universal set d) b&c
4. A _____ is a collection of well defined objects.
   a)logic b)set c)complement d)none of the above
5. The complement of a fuzzy set A is a new fuzzy set with a membership function ________.
   a)$\mu_A(x)=1-\mu_A(x)$ b) $\mu_A(x)=\mu_A(x)-1$ c) $\mu_A(x)=1$ d) $\mu_A(x)=0$
6. In fuzzy set theory, the membership of any element belonging to the null set Ø is
   a)1 b)interval 0 to 1 c)0 d)0.5
7. Choose the correct mathematical notation of Cartesian product of two sets A and B by A x B in crisp set theory.
   a)A X B = {(a,b)/a Ɛ A, b Ɛ B} b) A X B = {(b,a)/b Ɛ B, a Ɛ A}
   c) A X B = {(a U b)/a Ɛ A, b Ɛ B} d)none of the above
8. Fuzzy set theory is an effective tool for handling
   a)pattern recognition b)uncertainty c)certainty d)maxima-minima
9. Fuzzy logic survives on _________.
   a)2-state truth table b)multistate truth table c)0 & 1 d)biological neuron
10. In propositional logic, de morgan's law ~(P v Q)=?
    a) (~P v ~Q) b)(~P ^ ~Q) c) P ^ (P v Q) d) none of the above
11. Propositional logic lacks the ability to symbolize __________.
    a)quantification b)involution c)absorption d)none of the above
12. _______________ is classified in to propositional logic and predicate logic
    a)fuzzy logic b)crisp logic c)connective d)quantification
13. The number of elements in a set is called _________.
    a)cardinality b)singleton c)power set d)membership
14. If A={1,2,3,5} what is the cardinality of power set A?
    a)12 b)8 c)0 d)16
15. If (A U B) U C = A U (B U C), choose the name of property in crisp sets
    a)idempotence b)identity c)commutative d)associative

16. Classification of soil is an application of ________________.
a)Single layer feed forward network b)multi-layer feed forward network
c)back propagation network d) both b & c
17. The truth values such as absolutely true, partly true, absolutely false, very true are numerically equivalent to the interval o and 1 are called
a)crisp logic b)propositional logic c)predicate logic d)fuzzy logic
18. Poor choice of learning co-efficient in back propagation networks can result in a
a)faster convergence b)failure in convergence c)fatal error d)none of the above
19. Identify the power set of A in count, if A={1,5,7}
a)3 b)9 c)8 d)2
20. Let A = { 4,5,6,7} what is the cardinality of set A?
a)infinity b)1 c)4 d)none of the above

Part B (3 * 2 = 6 Marks)
(Answer all the Questions)

21.What is meant by associative memory?
22.In crisp logic, if A={2,0,6} and B = {1,3,6} what is A U B?
23. Give an example for a fuzzy set.

Part C (3 * 8 =24 Marks)
(Answer all the Questions)

24. a)Draw the diagram of single layer feed forward network and explain.
(OR)
b)Draw the basic model of MADALINE network.
25. a) Briefly explain about calculation of error in multi-layer perceptron using BPN.
(OR)
b) Explain the application of journal ball bearing problem using neural networks.
26. a) Draw the truth table of fuzzy connectives on fuzzy logic.
(OR)
b) Explain the union, intersection and complement operations on fuzzy sets.

Register_____________

(16CTU603A)

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be university)

(Established Under Section 3 of UGC Act 1956)

Coimbatore-641 021

B.Sc DEGREE EXAMINATION

MODEL EXAMINATION – APRIL 2019

Sixth Semester

Duration: 2 Hours Soft Computing

Date & Session: 12.03.2019 (FN) Maximum:50 Marks

Part A (20*1=20 Marks)

(Answer all the Questions)

1. A kind of computing capable of producing accurate results even if the supplied data is noisy
   a)hard computing b)soft computing c)hybrid computing d)Genetic computing
2. Neural network is a _________ models of biological neuron system, which is _______ made up of highly interconnected neural element.
   a)simplified and massively parallel b)complex and massively parallel architecture
   c)simplified and massively parallel distributed processing system
   d)complex and massively parallel distributed processing system
3. The stability-plasticity property holds good for _________
   a)neural network and fuzzy systems b)neural network and genetic systems
   c)only for traditional programming d)none of the above
4. Summation of weighted inputs in artificial neural network is called
   a) sum-of-product b)product-of-sum c)signum function d)threshold function
5. Convergence refers to
   a) dispersion of multiple values b) arriving at exact solution
   c)changing target output d)training the algorithm to learn
6. When a target output is not given to an algorithm but algorithm learns on its adaptation called ________.
   a)hebbian learning b)unsupervised learning c)supervised learning d) both b & d
7. The average weight of a brain is about _______.
   a) 100 grams b)1500 grams c)1.2 kg d)none of these
8. Learning rate co-efficient determines the size of weight adjustments made at each iteration and hence influences the ________________
   a)rate of convergence b)convergence c)divergence d) both b& c
9. Classification of soil is an application of ________________.
   a)Single layer feed forward network b)multi-layer feed forward network
   c)back propagation network d) both b & c
10. A set with reference to a particular context contains all possible elements having the same characteristics and from which sets can be formed
    a)crisp set b)universal set c)fuzzy set d)none of the above
11. The number of elements contained in a set called
    a)singleton set b)power set c)membership function d)cardinality
12. ________________ theory is an effective tool to tackle the problem of uncertainty.
    a)crisp set b)null set c)fuzzy set d)none of the above
13. Crisp relations on sets are subsets of ___________________ of the given sets.
    a)complement b)union c)Cartesian product d)none of the above

14. The truth values such as absolutely true, partly true, absolutely false, very true are numerically equivalent to the interval o and 1 are called
 a)crisp logic b)propositional logic c)predicate logic d)fuzzy logic
15. Universal quantifier and existential quantifiers are used in
 a)propositional logic b)predicate logic c)crisp logic d)none of the above
16. In propositional logic, the associative property of (P v Q) v R = ?
 a)P ^(Q ^ R) b)P v (Q v R) c)(P ^ R) v (Q ^ R) d) none of the above
17. Mutation means
 a)element of DNA is modified b)a sudden change in genetic set up
 c)changing the offspring d)a &b
18. Genetic algorithms are suitable for
 a)optimization problem b)convergence c)inference d) a& b
19. Symbolic logic helps in description of events in the real world and in addition to
 a)effective tool for inferring from given set of facts
 b)effective tool for deducing information from a given set of facts
 c) a&b
 d)none of the above
20."Survival of the Fittest" theory is ideally suited for
 a)neural networks b)neural and fuzzy systems c)genetic algorithms d) a&d

Part B (3 * 2 = 6 Marks)
(Answer all the Questions)

21. What is soft computing?

22. Define Perceptron.

23.State the basic operations on crisp sets.

Part C (3 * 8 =24 Marks)
(Answer all the Questions)

24. a) Draw the structure of neuron. Explain the functions of interconnected neurons in human brain.

(or)

b) Explain the model of artificial neuron

25.a)Explain the laws of propositional logic in crisp sets.

(or)

b)Write short notes on i)centroid method ii)centre of sums method

26.a)Discuss the basic concept of Genetic Algorithm.

(or)

b)Explain the working principle of genetic algorithm.

Prepared by Dr P.G Sivagaminathan, Department of CS,CA &IT,KAHE