GOKARAJU RANGARAJU
INSTITUTE OF ENGINEERING AND TECHNOLOGY

Department of Electrical & Electronics Engineering

Soft Computing Techniques

Course

By R. Anil Kumar

INDEX

Department of Electrical & Electronics Engineering

Vision of the Institute

To be among the best of the institutions for engineers and technologists with attitudes, skills and knowledge and to become an epicenter of creative solutions.

Mission of the Institute

To achieve and impart quality education with an emphasis on practical skills and social relevance

Vision of the Department

To impart technical knowledge and skills required to succeed in life, career and help society to achieve self sufficiency.

Mission of the Department

1. To become an internationally leading department for higher learning.
2. To build upon the culture and values of universal science and contemporary education.
3. To be a center of research and education generating knowledge and technologies which lay groundwork in shaping the future in the fields of electrical and electronics engineering.
4. To develop partnership with industrial, R&D and government agencies and actively participate in conferences, technical and community activities.

Department of Electrical & Electronics Engineering

Programme Educational Objectives (B.Tech. – EEE)

This programme is meant to prepare our students to professionally thrive and to lead.

During their progression:

Graduates will be able to

PEO 1: Have a successful technical or professional careers, including supportive and leadership roles on multidisciplinary teams.

PEO 2: Acquire, use and develop skills as required for effective professional practices.

PEO 3: Able to attain holistic education that is an essential prerequisite for being a responsible member of society.

PEO 4: Engage in life-long learning, to remain abreast in their profession and be leaders in our technologically vibrant society.

Programme Outcomes (B.Tech. – EEE)

At the end of the Programme, a graduate will have the ability to

PO 1: Apply knowledge of mathematics, science, and engineering.

PO 2: Design and conduct experiments, as well as to analyze and interpret data.

PO 3: Design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.

PO 4: Function on multi-disciplinary teams.

PO 5: Identify, formulates, and solves engineering problems.

PO 6: Understanding of professional and ethical responsibility.

PO 7: Communicate effectively.

PO 8: Broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.

PO 9: Recognition of the need for, and an ability to engage in life-long learning.

PO 10: Knowledge of contemporary issues.

PO 11: Utilize experimental, statistical and computational methods and tools necessary for engineering practice.

PO 12: Demonstrate an ability to design electrical and electronic circuits, power electronics, power systems; electrical machines analyze and interpret data and also an ability to design digital and analog systems and programming them.

PEOs & POs Mapping

Programme Educational Objectives (PEOs)	Programme Outcomes (POs)											
	1	2	3	4	5	6	7	8	9	10	11	12
1	M	M	-	-	H	-	-	H	H	-	H	H
2	-	-	M	M	H	H	H	-	-	-	-	H
3	-	-	-	-	H	H	M	M	M	M	H	H
4	-	-	-	M	M	H	M	H	H	-	M	H

* H: Strongly Correlating (3); M: Moderately Correlating (2)& L: Weakly Correlating (1)

Department of Electrical & Electronics Engineering

GRIET/DAA/1H/G/18-19 05 May 2018

ACADEMIC CALENDAR
Academic Year 2018-19

III & IV B.TECH – FIRST SEMESTER

S. No.	EVENT	PERIOD	DURATION
1	1st Spell of Instructions	02-07-2018 to 01-09-2018	9 Weeks
2	1st Mid-term Examinations	03-09-2018 to 05-09-2018	3 Days
3	2nd Spell of Instructions	06-09-2018 to 24-10-2018	7 Weeks
4	2nd Mid-term Examinations	25-10-2018 to 27-10-2018	3 Days
5	Preparation	29-10-2018 to 06-11-2018	1 Week 3 Days
6	**End Semester Examinations (Theory/ Practicals) Regular/Supplementary**	**08-11-2018 to 08-12-2018**	**4 Weeks 3 Days**
7	Commencement of Second Semester, A.Y 2018-19	10-12-2018	

III & IV B.TECH – SECOND SEMESTER

S. No.	EVENT	PERIOD	DURATION
1	1st Spell of Instruction	10-12-2018 to 02-02-2019	8 Weeks
2	1st Mid-term Examinations	04-02-2019 to 06-02-2019	3 Days
3	2nd Spell of Instruction	07-02-2019 to 06-04-2019	**8 Weeks 3 Days**
4	2nd Mid-term Examinations	08-04-2019 to 10-04-2019	3 Days
5	Preparation	11-04-2019 to 17-04-2019	1 Week
6	**End Semester Examinations (Theory/ Practicals) Regular**	18-04-2019 to 08-05-2019	3 Weeks
7	**Supplementary and Summer Vacation**	**09-05-2019 to 22-06-2019**	**6 Weeks 3 Days**
8	Commencement of First Semester, A.Y 2019-20	24-06-2019	

Department of Electrical & Electronics Engineering

2018-19 (I Sem) Subject Allocation sheet

II YEAR (GR17)	Section-A	Section-B
Special Functions and Complex Variable	Dr GS	Dr GS
Electromagnetic Fields	SN	SN
Network Theory	MS	MS
DC Machines and Transformers	Dr BPB	Dr BPB
Computer Organization	PRK	PRK
DC Machines Lab	MP/DSR	PRK/DSR
Electrical Networks Lab	YSV/GBR	YSV/GBR
Electrical Simulation Lab	GSR/PS	GSR/PS
Environmental Science		
III YEAR (GR15)	Section-A	Section-B
Power Transmission System	VVRR/MP	VVRR/MP
Microcontrollers	PK	PK
Power Electronics	Dr TSK	DKK
Electrical Measurements& Instrumentation (PE-1)	UVL	UVL
Solar & Wind Energy Systems (OE-1)	PSVD/Dr JP	PSVD/Dr JP
Sensors/Measurements& Instrumentation Lab	PSVD/PS	UVL/PS
Power Electronics Lab	PPK/MRE	SN/MRE
Microcontrollers Lab	RAK/DKK	PK/DKK
IV YEAR(GR15)	Section-A	Section-B
Power Semiconductor Drives	YSV	Dr DGP
Power System Operation & Control	Dr JSD	Dr JSD
High Voltage DC Transmission Systems	MRE	Dr SVJK
Electrical Distribution Systems (PE-3)	VVSM	
High Voltage Engineering (PE-3)	VUR	
Soft Computing Techniques (OE-3)	RAK	RAK
DSP based Electrical Lab	AVK/DKK	AVK/DKK
Power Systems Simulation Lab	VVSM / GSR	VVSM / GSR
Power Electronic Drives Lab	MP/GBR	MP/GBR

Department of Electrical & Electronics Engineering

I/I BEE(AICTE)	A/B	C/D/E
BEE	ML	
BEE	KS	
BEE	MK	
BEE	MVK	
BEE	MNSR	
Civil II/I (GR15)	A	B
ET	PPK	PPK
M.Tech (PE)(AICTE)	A	
Electric Drives System	Dr DGP	
Power Electronic Converters	Dr TSK	
Power Quality	AVK	
Electric and Hybrid Vehicles	Dr BPB	
Electrical Drives Laboratory	AVK/GBR	
Power Electronics Lab	SN/MS	
M.Tech (PS)(AICTE)	A	
Power System Analysis	Dr JSD	
Power System Dynamics	Dr SVJK	
Power Quality	AVK	
Electric and Hybrid Vehicles	Dr BPB	
Power System Steady State Analysis Lab	VVSM/VVRR	
Power System Dynamics Lab	Dr SVJK/YSV	

GOKARAJU RANGARAJU
INSTITUTE OF ENGINEERING AND TECHNOLOGY

Department of Electrical & Electronics Engineering

T/PRIN/06/G/01/18-19 Jul/2018

B Tech (EEE) – IV/I Semester Section A

DAY/ HOUR	9:00-10:00	10:00-11:30	11:30-1:00	12:30-01:00	01:00-02:30	02:30-04:00
MONDAY				Break		
TUESDAY		SCT Sec A				
WEDNESDAY			SCT Sec A			
THURSDAY						
FRIDAY						SCT Sec A
SATURDAY						

GOKARAJU RANGARAJU
INSTITUTE OF ENGINEERING AND TECHNOLOGY

Department of Electrical & Electronics Engineering

T/PRIN/06/G/01/18-19 Jul/2018

B Tech (EEE) – IV/I Semester Section B

DAY/ HOUR	9:00-10:00	10:00-11:30	11:30-12:00	12:30-01:00	01:00-02:30	02:30-04:00
MONDAY		SCT Sec B		Break		
TUESDAY					SCT Sec B	
WEDNESDAY						SCT Sec B
THURSDAY						
FRIDAY						
SATURDAY						

Department of Electrical & Electronics Engineering

T/PRIN/06/G/01/18-19 Jul/2018

B Tech (EEE) - I Semester Section A and B

DAY/ HOUR	9:00- 10:00	10:00-11:30	11:30-1:00	12:30-01:00	01:00-02:30	02:30-04:00
MONDAY		SCT Sec B		Break		
TUESDAY		SCT Sec A			SCT Sec B	
WEDNESDAY			SCT Sec A			SCT Sec B
THURSDAY						
FRIDAY						SCT Sec A
SATURDAY						

Work Load:

S. No	Name	Working Hours
1	R. Anil Kumar (A Section)	6 Hours
2	R. Anil Kumar (B Section)	6 Hours

Department of Electrical & Electronics Engineering

Course Code: GR15A4162 L T P C

IV Year I Sem 2 1 0 3

SOFT COMPUTING TECHNIQUES

UNIT I Neural Networks-I(Introduction & Architecture): Neuron, Nerve structure and synapse, Artificial Neuron and its model, activation functions, Neural network architecture: single layer and multilayer feed forward networks, recurrent networks. Various learning techniques; perception and convergence rule, Auto-associative and hetero-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-I (Introduction): Basic concepts of fuzzy logic, Fuzzy sets and Crisp sets, Fuzzy set theory and operations, Properties of fuzzy sets, Fuzzy and Crisp relations, 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, Fuzzifications & Defuzzification's, Fuzzy Controller, Industrial applications.

UNIT V Genetic Algorithm(GA): Basic concepts, working principle, procedures of GA, flow chart of GA, Genetic representations, (encoding) Initialization and selection, Genetic operators, Mutation, Generational Cycle, applications.

Department of Electrical & Electronics Engineering

Session Plan

S. No	Unit No.	Date	Topic
1	1	03.07.2018	Introduction to Biological Neuron and Artificial Neuron
2	1	04.07.2018	Neuron Structure and Synapse, comparision with ANN
3	1	06.07.2018	Types of Activation Functions
4	1	10.07.2018	Classification Activiation Functions in detail
5	1	11.07.2018	Neural network architecture: Single Layer feedforward N/W
6	1	13.07.2018	Neural network architecture: Multi Layer feedforward N/W
7	1	17.07.2018	Neural network architecture: Recurrent Networks
8	1	18.07.2018	Learning Techniques classification
9	1	20.07.2018	Different types of Learning Rules
10	1	24.07.2018	Perceptron Convergence Rule
11	1	25.07.2018	Introduction to Associative Memories
12	1	27.07.2018	Auto Associative Memory introduction with diagram
13	1	31.07.2018	Hetero Associative Memory
14	2	01.08.2018	Introduction to Perceptron Model and its solution
15	2	03.08.2018	Single Layer Perceptron Model
16	2	07.08.2018	Multilayer Perceptron Model
17	2	08.08.2018	Introduction to Back Propagation Neural Network
18	2	10.08.2018	Back Propagation Learning Algorithm
19	2	14.08.2018	Effect of Learning Rule Coefficient in Back Propagation NN
20	2	17.08.2018	Various types effecting parameters in BPNN
21	2	21.08.2018	List the applications of Neural Networks
22	3	22.08.2018	Introduction to Fuzzy Sets
23	3	24.08.2018	Difference between Fuzzy sets and Classical Sets
24	3	28.08.2018	Properties of Classical Sets and Fuzzy Sets
25	3	29.08.2018	Fuzzy Set Theory Operations
26	3	31.08.2018	Relations of Fuzzy and Crisp/Classical Sets
27	3	07.09.2018	Fuzzy to Crisp Conversion
28	3	11.09.2018	Fuzzy to Crisp Conversion contd..
29	4	12.09.2018	Introduction to Membership Functions in Fuzzy Sets
30	4	14.09.2018	Inference in Fuzzy Logic System
31	4	18.09.2018	If-Then Rules used in Fuzzy Logic
32	4	19.09.2018	Introduction to Fuzzy Implications
33	4	21.09.2018	Introduction to Fuzzy Algorithms
34	4	25.09.2018	Methods of Fuzzification used
35	4	26.09.2018	Methods of Defuzzification

36	4	28.09.2018	Fuzzy Logic Controller Block diagram operation
37	4	03.10.2018	Industrial Applications using Fuzzy Logic Controllers
38	4	05.10.2018	Introduction to Genetic Algorithms
39	5	9.10.2018	Basic Concepts in GA
40	5	10.10.2018	Working principle of Genetic Algorithm with block diagram
41	5	12.10.2018	Procedure steps for using Genetic Algorithm
42	5	16.10.2018	Flow Chart of GA
43	5	16.10.2018	Genetic Algorithm representations
44	5	17.10.2018	Encoding methods and Selection procedure in GA
45	5	19.10.2018	Introduction to Genetic Algorithm operators
46	5	23.10.2018	Mutation and Generational Cycle of GA
47	5	24.10.2018	Applications of GA

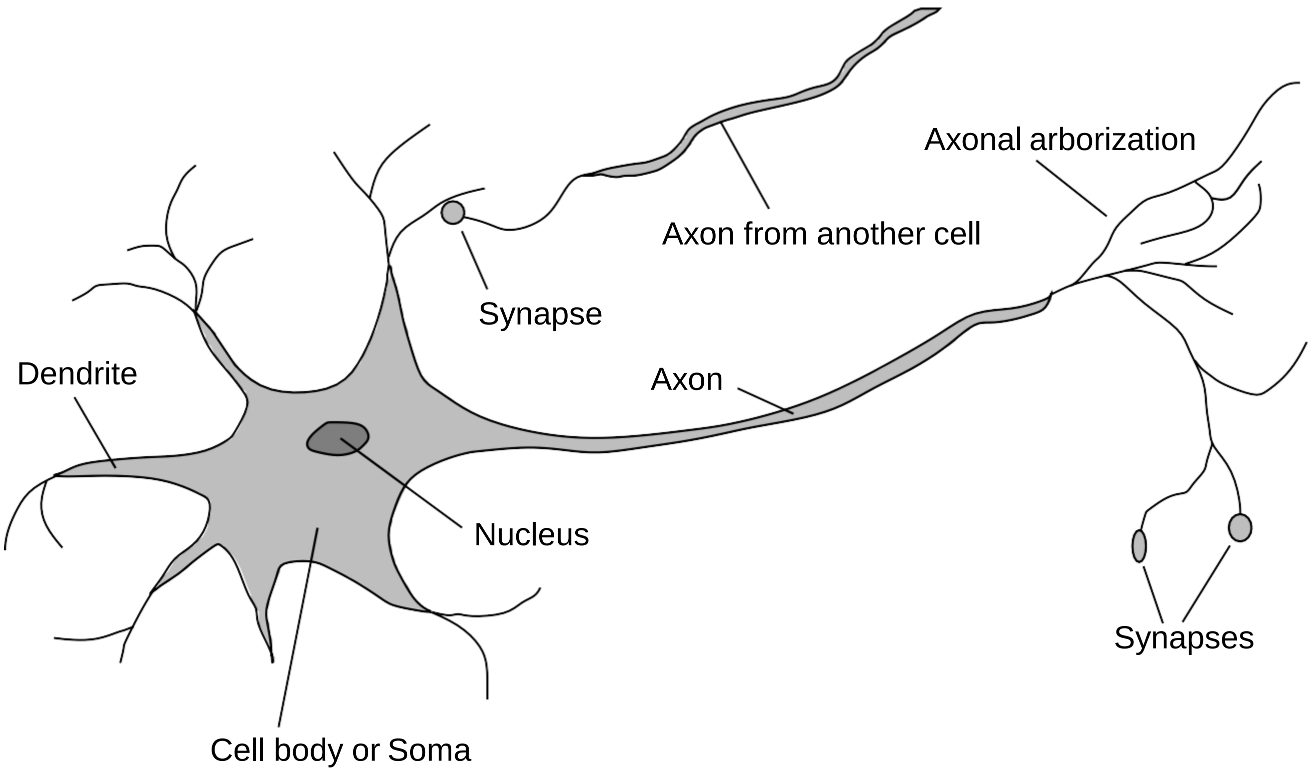
Neural networks

Outline

- Brains
- Neural networks
- Perceptrons
- Multilayer perceptrons
- Applications of neural networks

Brains

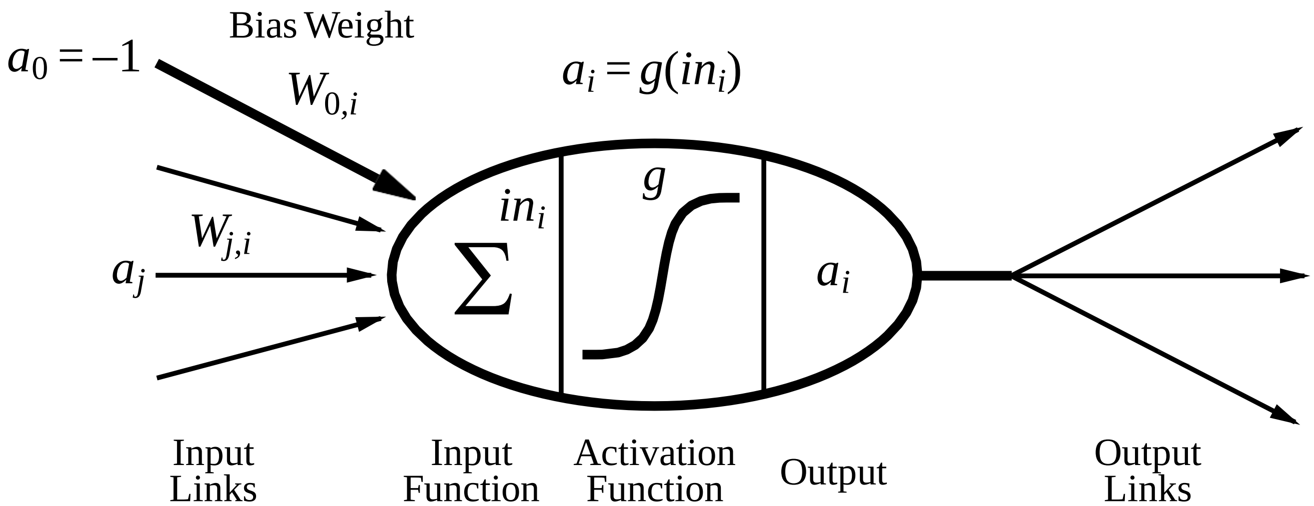
10^{11} neurons of > 20 types, 10^{14} synapses, 1ms–10ms cycle time
Signals are noisy "spike trains" of electrical potential



McCulloch–Pitts "unit"

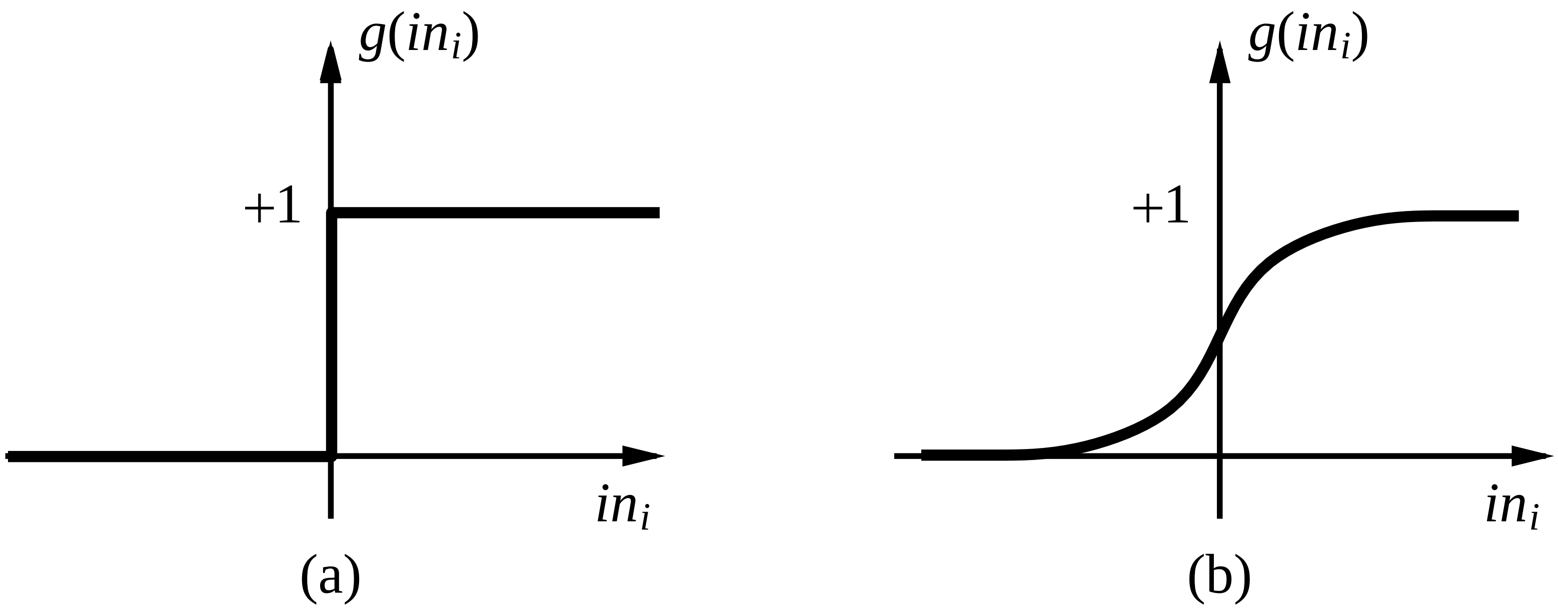
Output is a "squashed" linear function of the inputs:

$$a_i \leftarrow g(in_i) = g\left(\Sigma_j W_{j,i} a_j\right)$$



A gross oversimplification of real neurons, but its purpose is to develop understanding of what networks of simple units can do

Activation functions

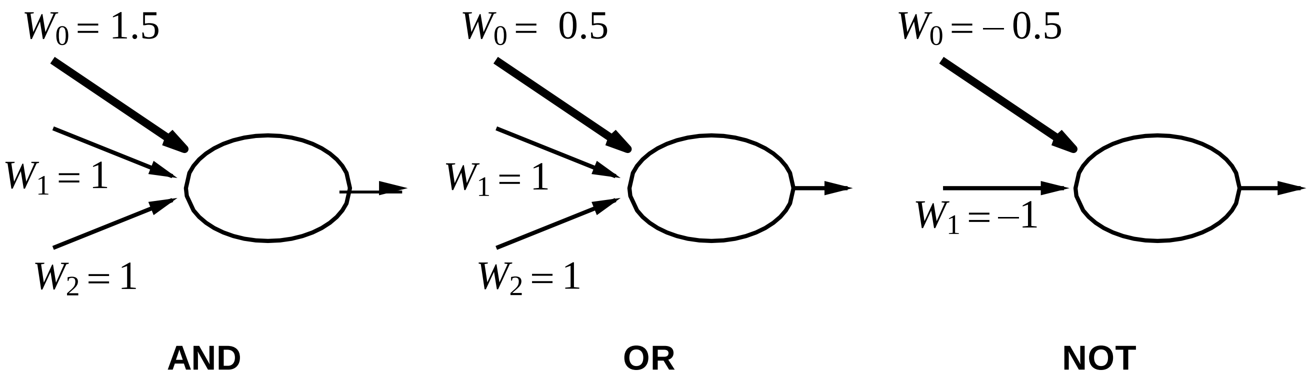


(a) is a step function or threshold function

(b) is a sigmoid function $1/(1 + e^{-x})$

Changing the bias weight $W_{0,i}$ moves the threshold location

Implementing logical functions



McCulloch and Pitts: every Boolean function can be implemented

Network structures

Feed-forward networks:

– single-layer perceptrons

– multi-layer perceptrons

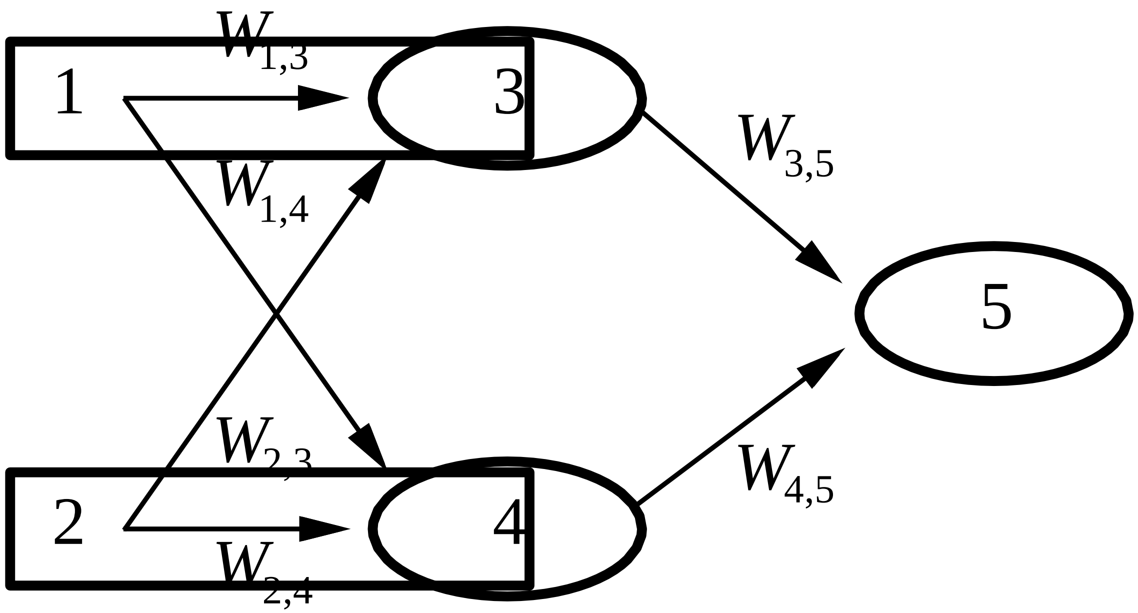
Feed-forward networks implement functions, have no internal state

Recurrent networks:

– Hopfield networks have symmetric weights ($W_{i,j} = W_{j,i}$)
$g(x) = \text{sign}(x)$, $a_i = \pm 1$;

– recurrent neural nets have directed cycles with delays
⇒ have internal state (like flip-flops), can oscillate etc.

Feed-forward example

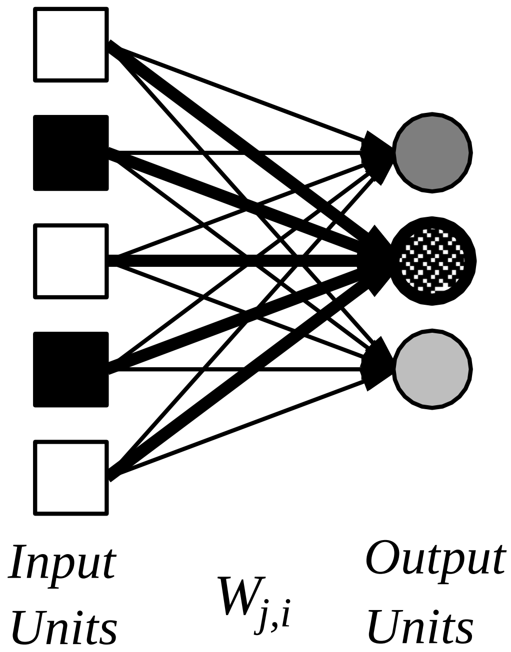


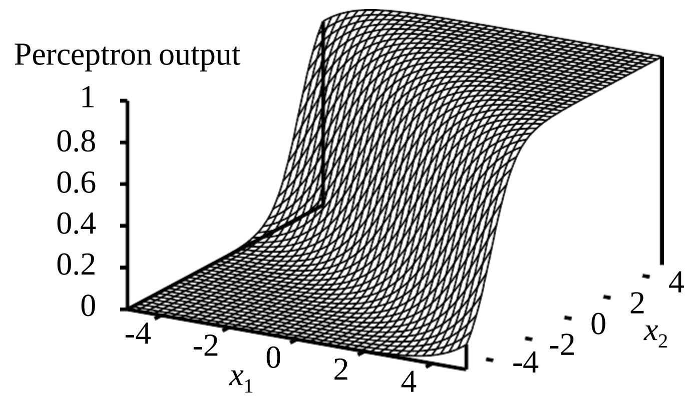
Feed-forward network = a parameterized family of nonlinear functions:

$$a_5 = g(W_{3,5} \cdot a_3 + W_{4,5} \cdot a_4)$$
$$= g(W_{3,5} \cdot g(W_{1,3} \cdot a_1 + W_{2,3} \cdot a_2) + W_{4,5} \cdot g(W_{1,4} \cdot a_1 + W_{2,4} \cdot a_2))$$

Adjusting weights changes the function: do learning this way!

Single-layer perceptrons





Output units all operate separately—no shared weights

Adjusting weights moves the location, orientation, and steepness of cliff

Perceptron learning

Learn by adjusting weights to reduce error on training set

The squared error for an example with input x and true output *y* is

$$E = \frac{1}{2}Err^2 \equiv \frac{1}{2}(y - h_{\mathbf{W}}(\mathbf{x}))^2 ,$$

Perform optimization search by gradient descent:

$$\frac{\partial E}{\partial W_j} = Err \times \frac{\partial Err}{\partial W_j} = Err \times \frac{\partial}{\partial W_j} \left(y - g(\Sigma_{j=0}^{n} W_j x_j) \right)$$
$$= -Err \times g'(in) \times x_j$$

Simple weight update rule:

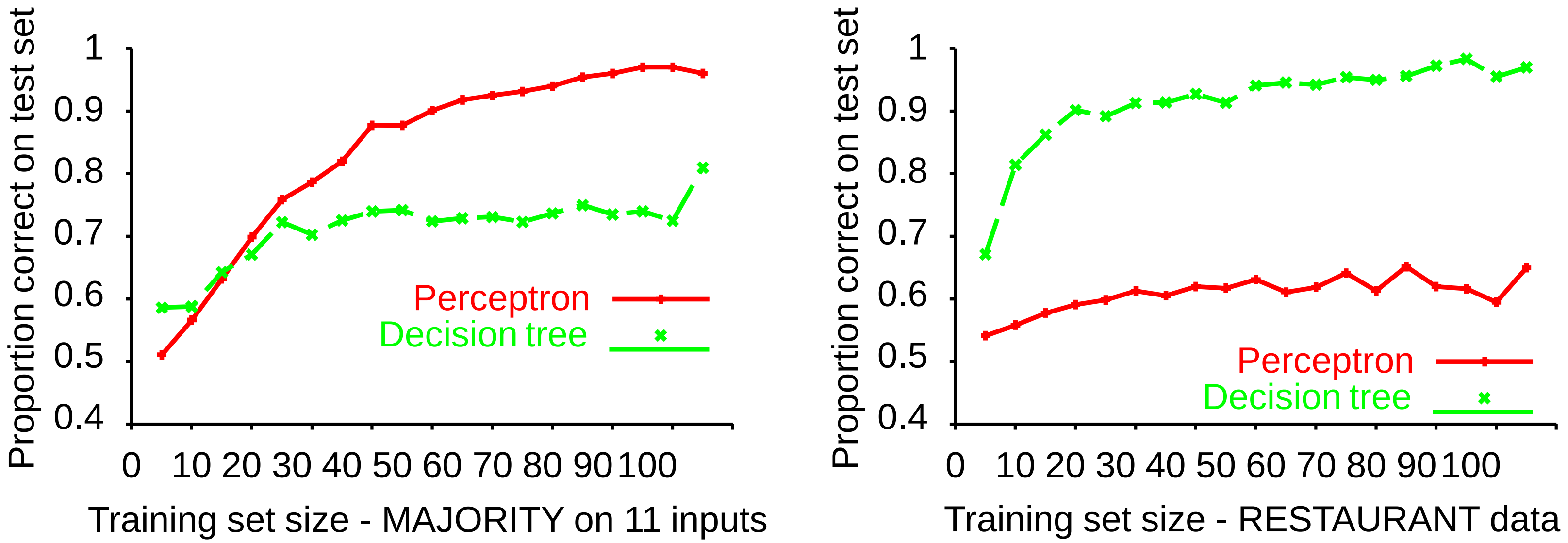
$$W_j \leftarrow W_j + \alpha \times Err \times g'(in) \times x_j$$

E.g., +ve error ⇒ increase network output

⇒ increase weights on +ve inputs, decrease on -ve inputs

Perceptron learning contd.

Perceptron learning rule converges to a consistent function
for any linearly separable data set

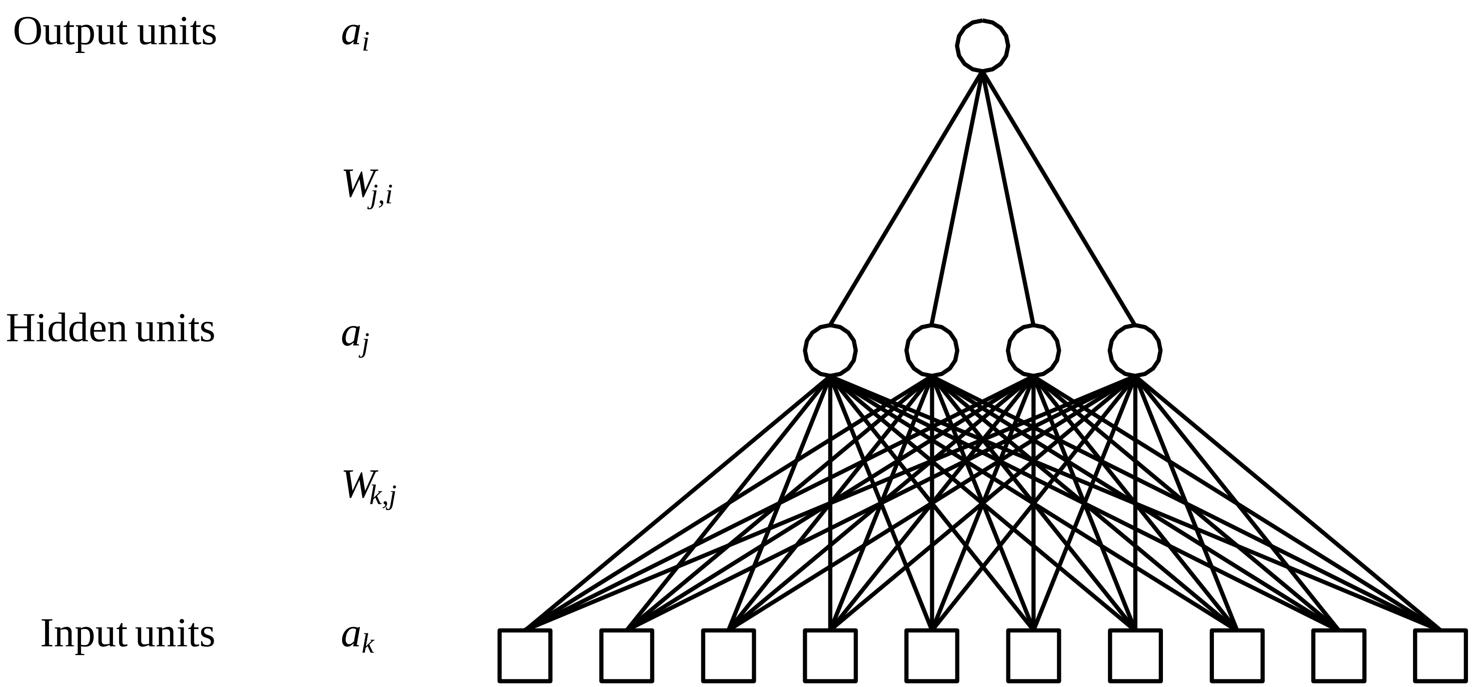


Perceptron learns majority function easily, DTL is hopeless

DTL learns restaurant function easily, perceptron cannot represent it

Multilayer perceptrons

Layers are usually fully connected;
numbers of hidden units typically chosen by hand



Summary

Most brains have lots of neurons; each neuron $\approx$ linear–threshold unit (?)

Perceptrons (one-layer networks) insufficiently expressive

Multi-layer networks are sufficiently expressive; can be trained by gradient descent, i.e., error back-propagation

Many applications: speech, driving, handwriting, fraud detection, etc.

Engineering, cognitive modelling, and neural system modelling subfields have largely diverged

FUZZY LOGIC

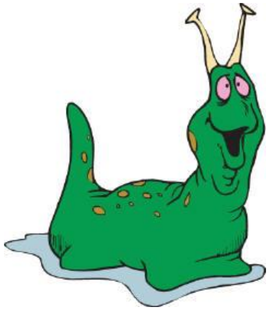
OVERVIEW

- What is Fuzzy Logic?
- Where did it begin?
- Fuzzy Logic vs. Neural Networks
- Fuzzy Logic in Control Systems
- Fuzzy Logic in Other Fields
- Future

WHAT IS FUZZY LOGIC?

- Definition of fuzzy
  - Fuzzy – “not clear, distinct, or precise; blurred”
- Definition of fuzzy logic
  - A form of knowledge representation suitable for notions that cannot be defined precisely, but which depend upon their contexts.

TRADITIONAL REPRESENTATION OF LOGIC



Slow

Fast

Speed = 0

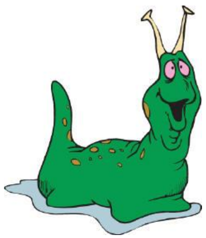
Speed = 1

```
bool speed;
get the speed
if ( speed == 0) {
//  speed is slow
}
else {
//  speed is fast
}
```

FUZZY LOGIC REPRESENTATION

- For every problem must represent in terms of fuzzy sets.

- What are fuzzy sets?



Slowest

[0.0 – 0.25]

Slow

[0.25 – 0.50]

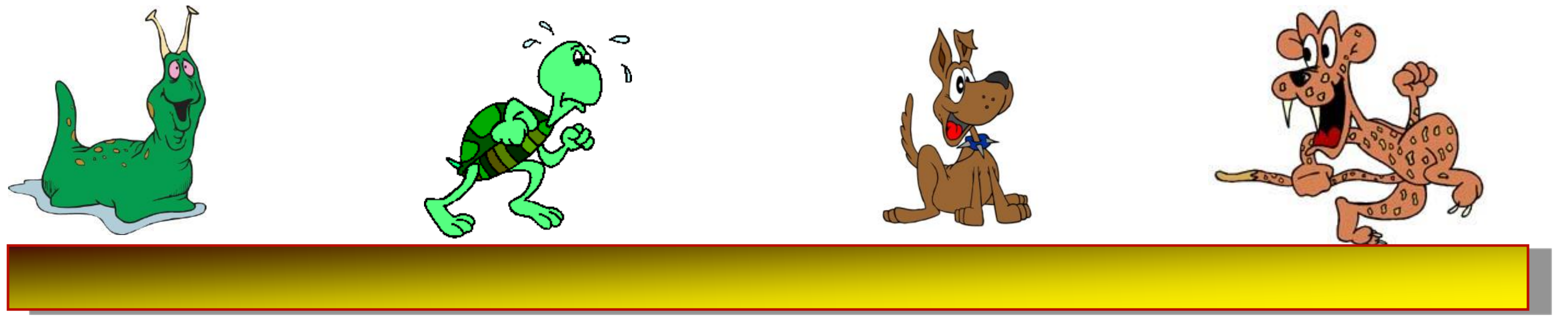
Fast

[0.50 – 0.75]

Fastest

[0.75 – 1.00]

FUZZY LOGIC REPRESENTATION CONT.



Slowest Slow Fast Fastest

```
float speed;
get the speed
if ((speed >= 0.0)&&(speed < 0.25)) {
//  speed is slowest
}
else if ((speed >= 0.25)&&(speed < 0.5))
{
//  speed is slow
}
else if ((speed >= 0.5)&&(speed < 0.75))
{
//  speed is fast
}
else // speed >= 0.75 && speed < 1.0
{
//  speed is fastest
}
```

ORIGINS OF FUZZY LOGIC

- Traces back to Ancient Greece
- Lotfi Asker Zadeh (1965)
  - First to publish ideas of fuzzy logic.
- Professor Toshire Terano (1972)
  - Organized the world's first working group on fuzzy systems.
- F.L. Smidth & Co. (1980)
  - First to market fuzzy expert systems.

FUZZY LOGIC VS. NEURAL NETWORKS

- How does a Neural Network work?
- Both model the human brain.
  - Fuzzy Logic
  - Neural Networks
- Both used to create behavioral systems.

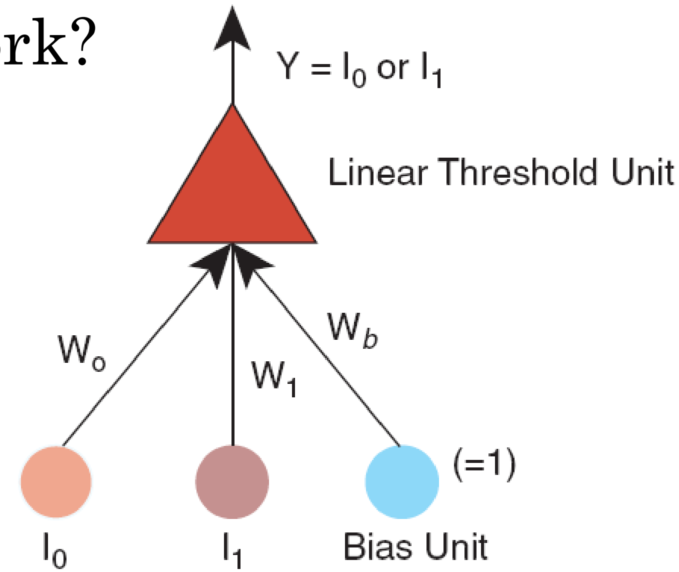


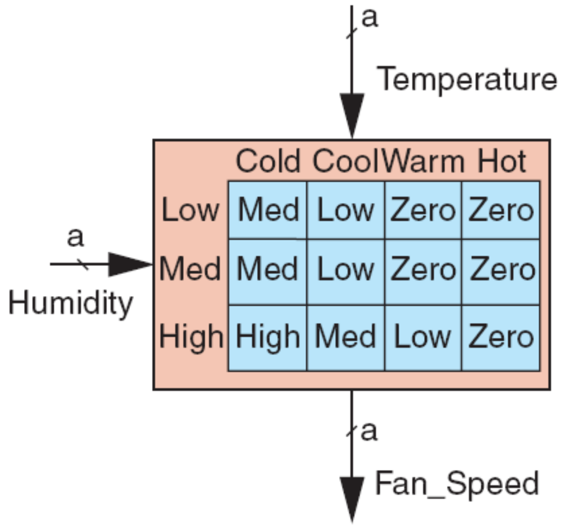
Fig. 2 A simple, single-unit adaptive network

FUZZY LOGIC IN CONTROL SYSTEMS

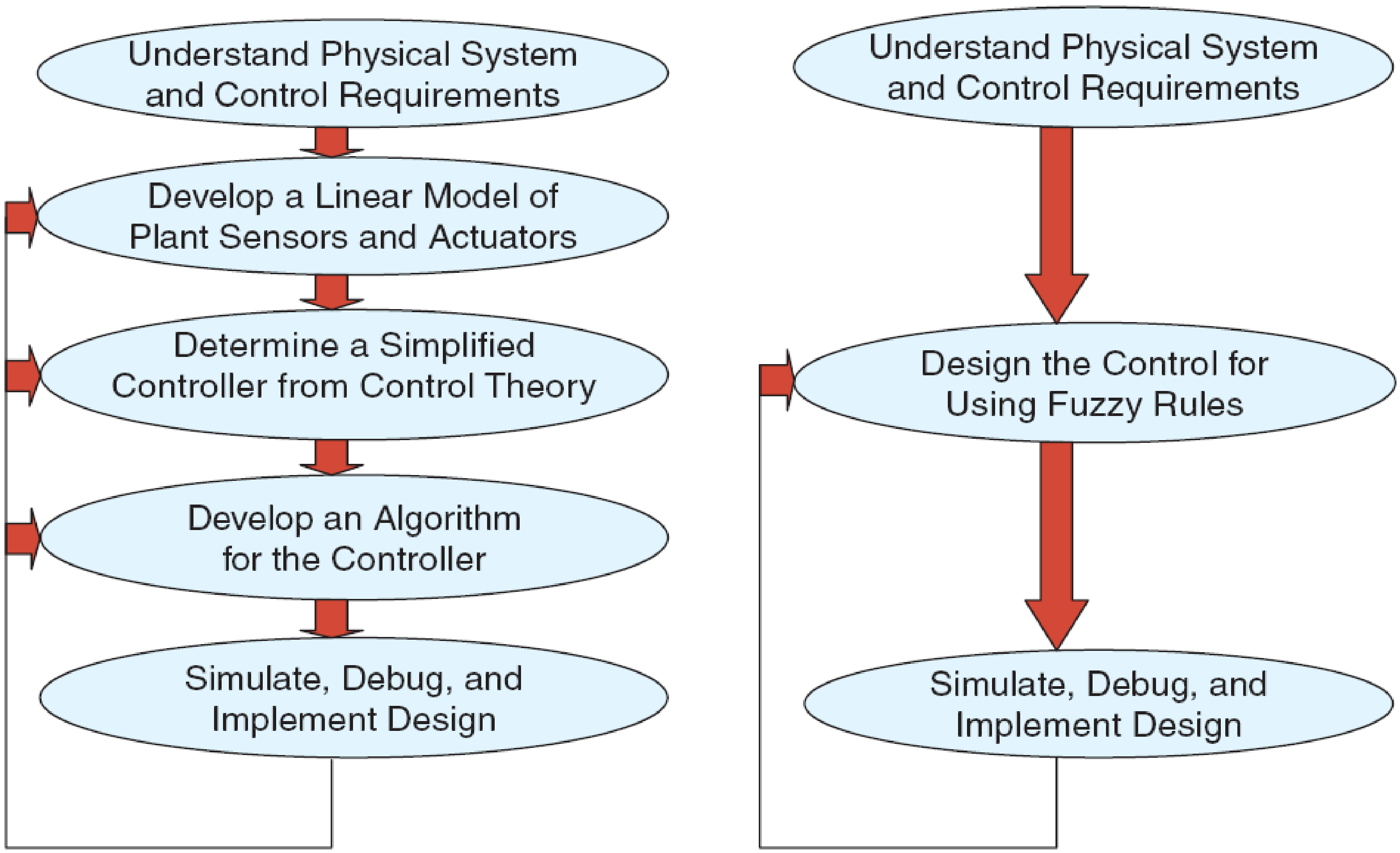
- Fuzzy Logic provides a more efficient and resourceful way to solve Control Systems.
- Some Examples
  - Temperature Controller
  - Anti – Lock Break System (ABS)

TEMPERATURE CONTROLLER

- The problem
  - Change the speed of a heater fan, based off the room temperature and humidity.
- A temperature control system has four settings
  - Cold, Cool, Warm, and Hot
- Humidity can be defined by:
  - Low, Medium, and High
- Using this we can define the fuzzy set.



BENEFITS OF USING FUZZY LOGIC



ANTI LOCK BREAK SYSTEM (ABS)

- Nonlinear and dynamic in nature
- Inputs for Intel Fuzzy ABS are derived from
  - Brake
  - 4 WD
  - Feedback
  - Wheel speed
  - Ignition
- Outputs
  - Pulsewidth
  - Error lamp

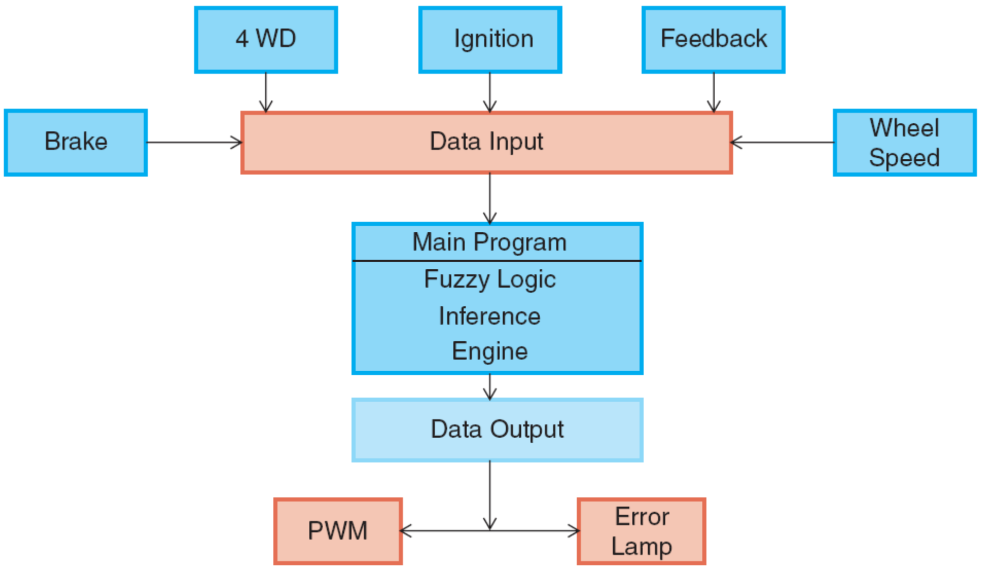


Fig. 6 ABS block diagram

FUZZY LOGIC IN OTHER FIELDS

- Business
- Hybrid Modeling
- Expert Systems

CONCLUSION

- Fuzzy logic provides an alternative way to represent linguistic and subjective attributes of the real world in computing.
- It is able to be applied to control systems and other applications in order to improve the efficiency and simplicity of the design process.

Genetic Algorithms

Introduction

- After scientists became disillusioned with classical and neo-classical attempts at modeling intelligence, they looked in other directions.
- Two prominent fields arose, connectionism (neural networking, parallel processing) and evolutionary computing.
- It is the latter that this essay deals with - genetic algorithms and genetic programming.

What is GA

- A **genetic algorithm** (or **GA**) is a search technique used in computing to find true or approximate solutions to optimization and search problems.
- Genetic algorithms are categorized as global search heuristics.
- Genetic algorithms are a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination).

What is GA

- Genetic algorithms are implemented as a computer simulation in which a population of abstract representations (called chromosomes or the genotype or the genome) of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem evolves toward better solutions.
- Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible.

What is GA

- The evolution usually starts from a population of randomly generated individuals and happens in generations.
- In each generation, the fitness of every individual in the population is evaluated, multiple individuals are selected from the current population (based on their fitness), and modified (recombined and possibly mutated) to form a new population.

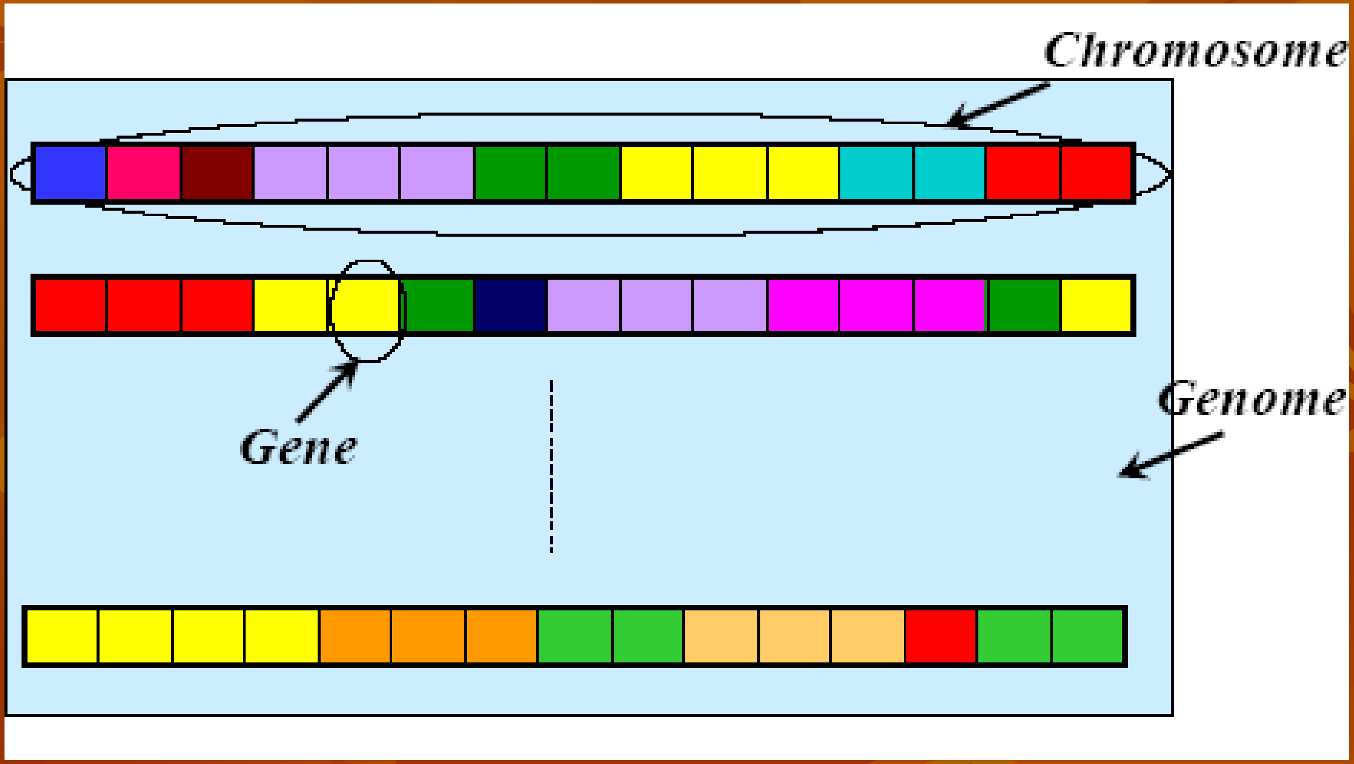
What is GA

- The new population is then used in the next iteration of the algorithm.
- Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.
- If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.

Key terms

- **Individual** - Any possible solution
- **Population** - Group of all *individuals*
- **Search Space** - All possible solutions to the problem
- **Chromosome** - Blueprint for an *individual*
- **Trait** - Possible aspect (*features)* of an *individual*
- **Allele -** Possible settings of trait (black, blond, etc.)
- **Locus -** The position of a *gene* on the *chromosome*
- **Genome -** Collection of all *chromosomes* for an *individual*

Chromosome, Genes and Genomes

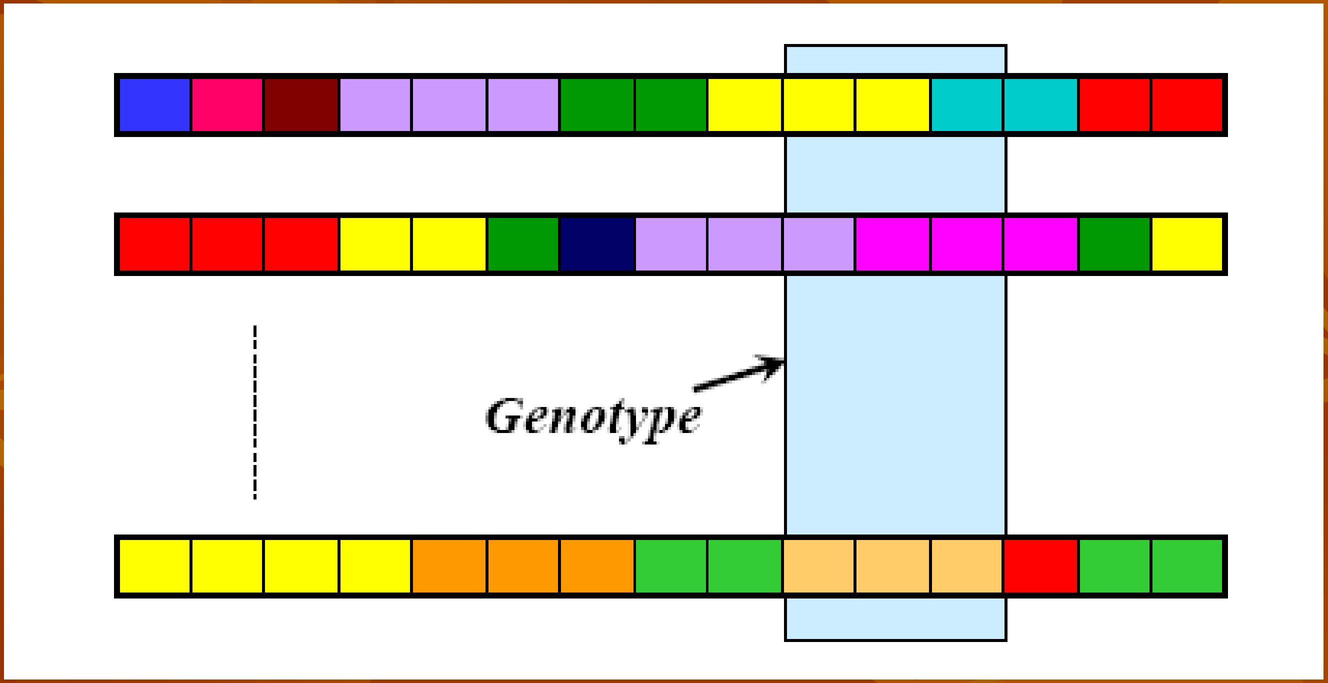


Genotype and Phenotype

- ***Genotype:***
  – Particular set of genes in a genome

- ***Phenotype:***
  – Physical characteristic of the genotype (smart, beautiful, healthy, etc.)

Genotype and Phenotype



GA Requirements

- A typical genetic algorithm requires two things to be defined:
- a genetic representation of the solution domain, and
- a fitness function to evaluate the solution domain.

- A standard representation of the solution is as an array of bits. Arrays of other types and structures can be used in essentially the same way.
- The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size, that facilitates simple crossover operation.
- Variable length representations may also be used, but crossover implementation is more complex in this case.
- Tree-like representations are explored in Genetic programming.

Representation

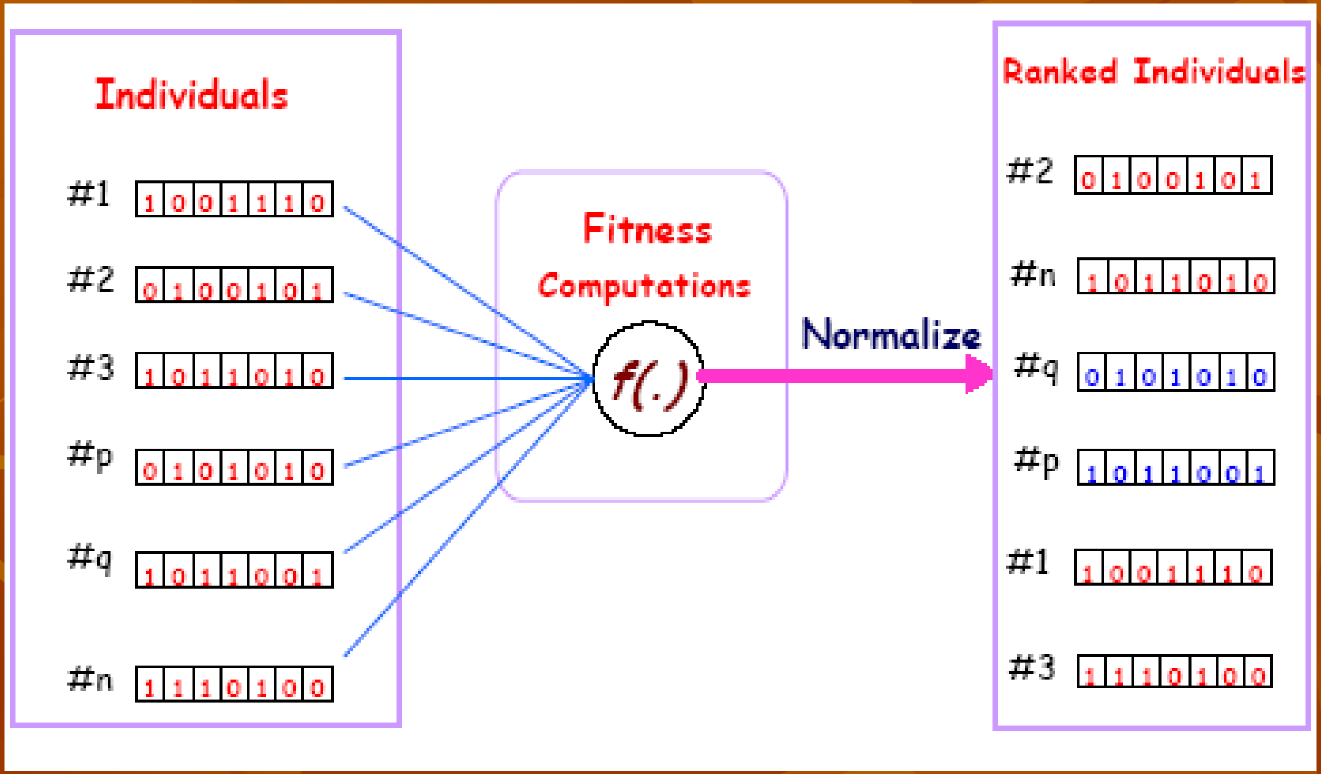
Chromosomes could be:

- Bit strings (0101 ... 1100)
- Real numbers (43.2 -33.1 ... 0.0 89.2)
- Permutations of element (E11 E3 E7 ... E1 E15)
- Lists of rules (R1 R2 R3 ... R22 R23)
- Program elements (genetic programming)
- ... any data structure ...

GA Requirements

- The fitness function is defined over the genetic representation and measures the *quality* of the represented solution.
- The fitness function is always problem dependent.
- For instance, in the knapsack problem we want to maximize the total value of objects that we can put in a knapsack of some fixed capacity.
- A representation of a solution might be an array of bits, where each bit represents a different object, and the value of the bit (0 or 1) represents whether or not the object is in the knapsack.
- Not every such representation is valid, as the size of objects may exceed the capacity of the knapsack.
- The *fitness* of the solution is the sum of values of all objects in the knapsack if the representation is valid, or 0 otherwise. In some problems, it is hard or even impossible to define the fitness expression; in these cases, interactive genetic algorithms are used.

A fitness function



Basics of GA

- The most common type of genetic algorithm works like this:
- a population is created with a group of individuals created randomly.
- The individuals in the population are then evaluated.
- The evaluation function is provided by the programmer and gives the individuals a score based on how well they perform at the given task.
- Two individuals are then selected based on their fitness, the higher the fitness, the higher the chance of being selected.
- These individuals then "reproduce" to create one or more offspring, after which the offspring are mutated randomly.
- This continues until a suitable solution has been found or a certain number of generations have passed, depending on the needs of the programmer.

General Algorithm for GA

- **Initialization**
- Initially many individual solutions are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions.
- Traditionally, the population is generated randomly, covering the entire range of possible solutions (the *search space*).
- Occasionally, the solutions may be "seeded" in areas where optimal solutions are likely to be found.

General Algorithm for GA

- **Selection**
- During each successive generation, a proportion of the existing population is selected to breed a new generation.
- Individual solutions are selected through a *fitness-based* process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected.
- Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate only a random sample of the population, as this process may be very time-consuming.
- Most functions are stochastic and designed so that a small proportion of less fit solutions are selected. This helps keep the diversity of the population large, preventing premature convergence on poor solutions. Popular and well-studied selection methods include roulette wheel selection and tournament selection.

General Algorithm for GA

- In roulette wheel selection, individuals are given a probability of being selected that is directly proportionate to their fitness.
- Two individuals are then chosen randomly based on these probabilities and produce offspring.

General Algorithm for GA

Roulette Wheel's Selection Pseudo Code:

```
for all members of population
        sum += fitness of this individual
end for
for all members of population
        probability = sum of probabilities + (fitness / sum)
        sum of probabilities += probability
end for
loop until new population is full
        do this twice
                number = Random between 0 and 1
                for all members of population
                        if number > probability but less than next probability then
                                you have been selected
                end for
        end
        create offspring
end loop
```

General Algorithm for GA

- **Reproduction**
- The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation.
- For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously.
- By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each child, and the process continues until a new population of solutions of appropriate size is generated.

General Algorithm for GA

- These processes ultimately result in the next generation population of chromosomes that is different from the initial generation.
- Generally the average fitness will have increased by this procedure for the population, since only the best organisms from the first generation are selected for breeding, along with a small proportion of less fit solutions, for reasons already mentioned above.

Crossover

- the most common type is single point crossover. In single point crossover, you choose a locus at which you swap the remaining alleles from on parent to the other. This is complex and is best understood visually.
- As you can see, the children take one section of the chromosome from each parent.
- The point at which the chromosome is broken depends on the randomly selected crossover point.
- This particular method is called single point crossover because only one crossover point exists. Sometimes only child 1 or child 2 is created, but oftentimes both offspring are created and put into the new population.
- Crossover does not always occur, however. Sometimes, based on a set probability, no crossover occurs and the parents are copied directly to the new population. The probability of crossover occurring is usually 60% to 70%.

Crossover


Parent 1
1011010 010100110
Parent 2
0011010 110110101
Child 1
1011010 110110101
Child 2
0011010 010100110

Mutation

- After selection and crossover, you now have a new population full of individuals.
- Some are directly copied, and others are produced by crossover.
- In order to ensure that the individuals are not all exactly the same, you allow for a small chance of mutation.
- You loop through all the alleles of all the individuals, and if that allele is selected for mutation, you can either change it by a small amount or replace it with a new value. The probability of mutation is usually between 1 and 2 tenths of a percent.
- Mutation is fairly simple. You just change the selected alleles based on what you feel is necessary and move on. Mutation is, however, vital to ensuring genetic diversity within the population.

Mutation

General Algorithm for GA

- **Termination**
- This generational process is repeated until a termination condition has been reached.
- Common terminating conditions are:
  - A solution is found that satisfies minimum criteria
  - Fixed number of generations reached
  - Allocated budget (computation time/money) reached
  - The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results
  - Manual inspection
  - Any Combinations of the above

GA Pseudo-code

Choose initial population
Evaluate the fitness of each individual in the population
Repeat

Select best-ranking individuals to reproduce

Breed new generation through crossover and mutation (genetic operations) and give birth to offspring

Evaluate the individual fitnesses of the offspring

Replace worst ranked part of population with offspring

Until <terminating condition>

Symbolic AI VS. Genetic Algorithms

- Most symbolic AI systems are very static.
- Most of them can usually only solve one given specific problem, since their architecture was designed for whatever that specific problem was in the first place.
- Thus, if the given problem were somehow to be changed, these systems could have a hard time adapting to them, since the algorithm that would originally arrive to the solution may be either incorrect or less efficient.
- Genetic algorithms (or GA) were created to combat these problems; they are basically algorithms based on natural biological evolution.

Symbolic AI VS. Genetic Algorithms

- The architecture of systems that implement genetic algorithms (or GA) are more able to adapt to a wide range of problems.
- A GA functions by generating a large set of possible solutions to a given problem.
- It then evaluates each of those solutions, and decides on a "fitness level" (you may recall the phrase: "survival of the fittest") for each solution set.
- These solutions then breed new solutions.
- The parent solutions that were more "fit" are more likely to reproduce, while those that were less "fit" are more unlikely to do so.
- In essence, solutions are evolved over time. This way you evolve your search space scope to a point where you can find the solution.
- Genetic algorithms can be incredibly efficient if programmed correctly.

Genetic Programming

- In programming languages such as LISP, the mathematical notation is not written in standard notation, but in prefix notation. Some examples of this:
- + 2 1 : 2 + 1
- * + 2 1 2 : 2 * (2+1)
- * + - 2 1 4 9 : 9 * ((2 - 1) + 4)
- Notice the difference between the left-hand side to the right? Apart from the order being different, no parenthesis! The prefix method makes it a lot easier for programmers and compilers alike, because order precedence is not an issue.
- You can build expression trees out of these strings that then can be easily evaluated, for example, here are the trees for the above three expressions.

Genetic Programming

```
   +                    *                    *
  / \                  / \                  / \
 1   2                +   2                +   9
                     / \                  / \
                    1   2                -   4
                                        / \
                                       2   1
```

Genetic Programming

- You can see how expression evaluation is thus a lot easier.
- What this have to do with GAs? If for example you have numerical data and 'answers', but no expression to conjoin the data with the answers.
- A genetic algorithm can be used to 'evolve' an expression tree to create a very close fit to the data.
- By 'splicing' and 'grafting' the trees and evaluating the resulting expression with the data and testing it to the answers, the fitness function can return how close the expression is.

Genetic Programming

- The limitations of genetic programming lie in the **huge** search space the GAs have to search for - an infinite number of equations.
- Therefore, normally before running a GA to search for an equation, the user tells the program which operators and numerical ranges to search under.
- Uses of genetic programming can lie in stock market prediction, advanced mathematics and military applications .

Evolving Neural Networks

- Evolving the architecture of neural network is slightly more complicated, and there have been several ways of doing it. For small nets, a simple matrix represents which neuron connects which, and then this matrix is, in turn, converted into the necessary 'genes', and various combinations of these are evolved.

Evolving Neural Networks

- Many would think that a learning function could be evolved via genetic programming. Unfortunately, genetic programming combined with neural networks could be *incredibly* slow, thus impractical.
- As with many problems, you have to constrain what you are attempting to create.
- For example, in 1990, David Chalmers attempted to evolve a function as good as the delta rule.
- He did this by creating a general equation based upon the delta rule with 8 unknowns, which the genetic algorithm then evolved.

Other Areas

- Genetic Algorithms can be applied to virtually any problem that has a large search space.
- Al Biles uses genetic algorithms to filter out 'good' and 'bad' riffs for jazz improvisation.
- The military uses GAs to evolve equations to differentiate between different radar returns.
- Stock companies use GA-powered programs to predict the stock market.

Example

- $f(x) = \{MAX(x^2): 0 <= x <= 32 \}$
- Encode Solution: Just use 5 bits (1 or 0).
- Generate initial population.

A	0	1	1	0	1
B	1	1	0	0	0
C	0	1	0	0	0
D	1	0	0	1	1

- Evaluate each solution against objective.

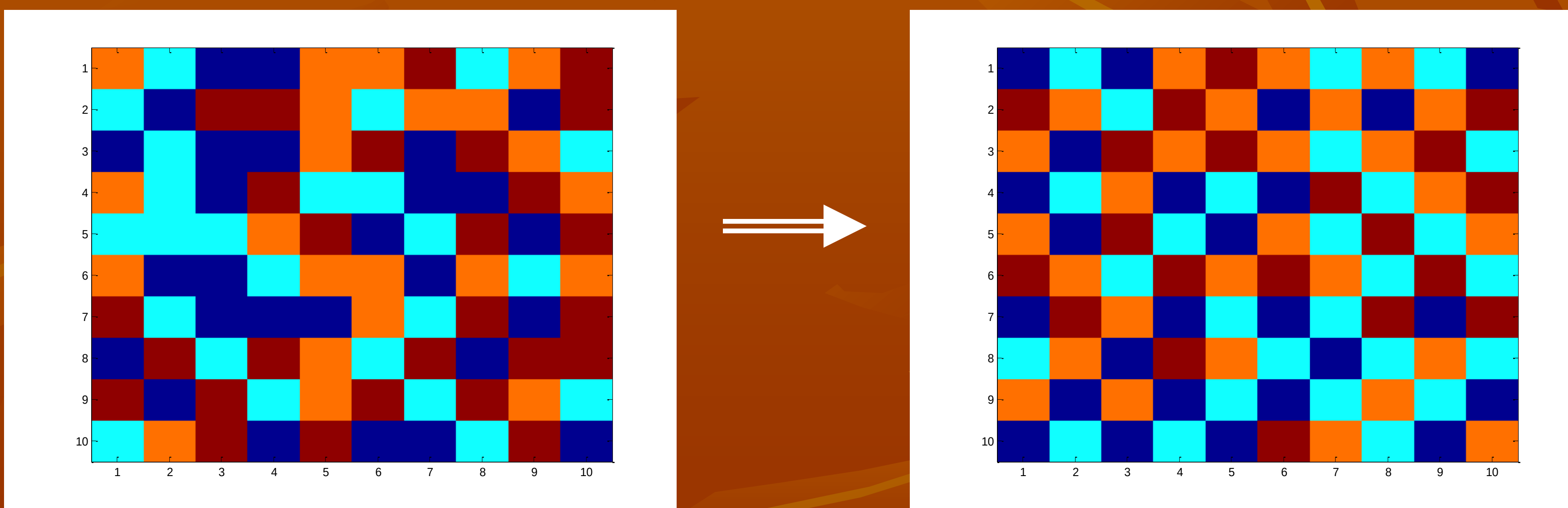
Sol.	**String**	**Fitness**	**% of Total**
A	01101	169	14.4
B	11000	576	49.2
C	01000	64	5.5
D	10011	361	30.9

Example Cont'd

- Create next generation of solutions
  - Probability of "being a parent" depends on the fitness.
- Ways for parents to create next generation
  - Reproduction
    - Use a string again unmodified.
  - Crossover
    - Cut and paste portions of one string to another.
  - Mutation
    - Randomly flip a bit.
  - COMBINATION of all of the above.

Checkboard example

- We are given an ***n*** by ***n*** checkboard in which every field can have a different colour from a set of four colors.
- Goal is to achieve a checkboard in a way that there are no neighbours with the same color (not diagonal)



Checkboard example Cont’d

- Chromosomes represent the way the checkboard is colored.
- Chromosomes are not represented by bitstrings but by **bitmatrices**
- The bits in the bitmatrix can have one of the four values 0, 1, 2 or 3, depending on the color.
- Crossing-over involves matrix manipulation instead of point wise operating.
- Crossing-over can be combining the parential matrices in a horizontal, vertical, triangular or square way.
- Mutation remains bitwise changing bits in either one of the other numbers.

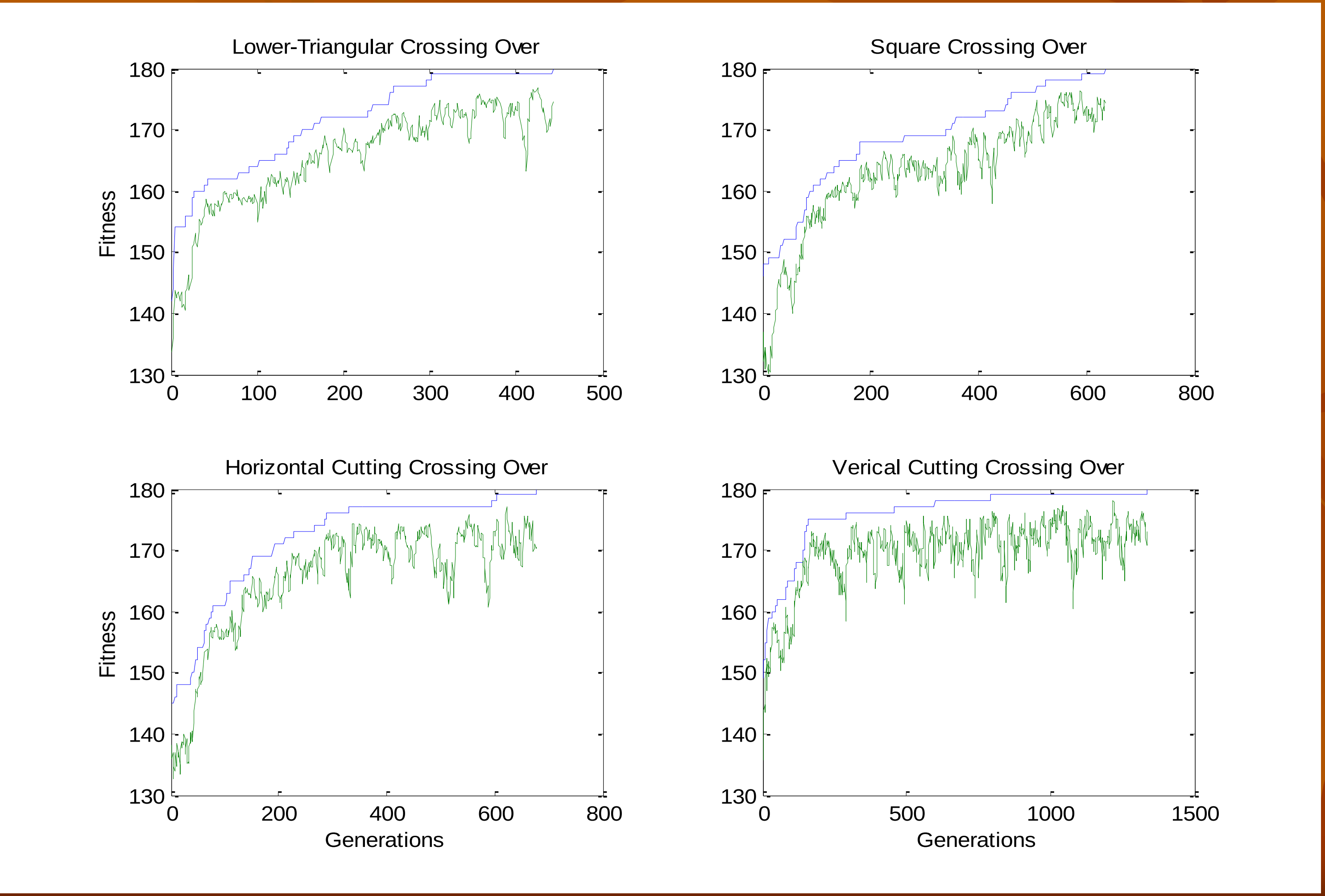
Checkboard example Cont'd

- This problem can be seen as a graph with ***n*** nodes and ***(n-1)*** edges, so the fitness **f(x)** is defined as:

$$\mathbf{f(x) = 2 \cdot (n-1) \cdot n}$$

Checkboard example Cont'd

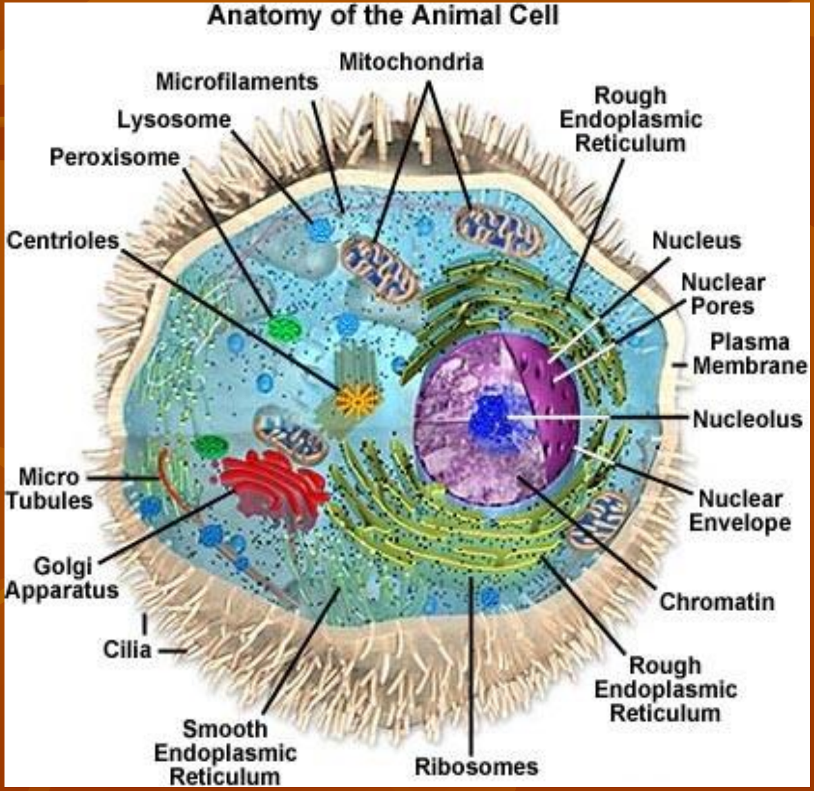
- Fitnesscurves for different cross-over rules:



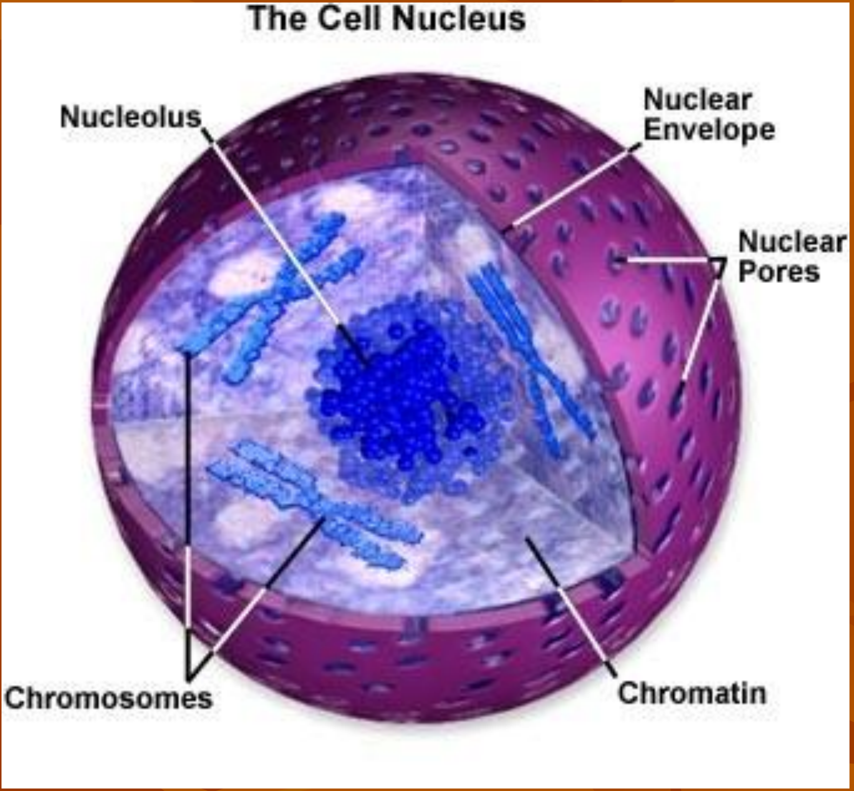
Questions

??

THANK YOU



Anatomy of the Animal Cell
Mitochondria
Microfilaments
Lysosome
Peroxisome
Rough Endoplasmic Reticulum
Centrioles
Nucleus
Nuclear Pores
Plasma Membrane
Nucleolus
Micro Tubules
Nuclear Envelope
Golgi Apparatus
Chromatin
Cilia
Rough Endoplasmic Reticulum
Smooth Endoplasmic Reticulum
Ribosomes



The Cell Nucleus
Nucleolus
Nuclear Envelope
Nuclear Pores
Chromosomes
Chromatin

Department of Electrical & Electronics Engineering

Name of the Course: Soft Computing Techniques

Course Objectives:

At the end of the Course Student must be able to

1. Tell what the types of activation functions are used in Artificial Neural Networks
2. Summarize the Back Propagation Neural Networks and the factors effecting it.
3. List the types of Associative memories.
4. Identify the difference between the Fuzzy Sets and Classical Sets.
5. Compare the types of Defuzzification methods to convert fuzzy values to crisp values.
6. Choose the Fuzzification methods and applications designed with Fuzzy Logic.
7. Explain the importance of Genetic Algorithm and its applications.

Course Outcomes:

At the end of the Course Student will be able to

1. Choose the type of activation function for a selected Artificial Neuron Network Model.
2. Illustrate the learning rules and working single and multi-layer Perceptron Model.
3. Tell the importance of Auto and Hetero Associative Memories.
4. Illustrate the effect of learning coefficient in Back Propagation Neural Network.
5. List the Fuzzy Logic methods and Defuzzification Methods used in converting Fuzzy Set to Classical Set.
6. Discuss the applications of Fuzzy Logic in Industrial Applications.
7. Elaborate the design procedure steps involved in designing application with Genetic Algorithm.

CO – PO MAPPINGS

ASSESSMENT METHODS:

1. Regular attendance
2. Written tests clearly linked to learning objectives
3. Classroom assessment techniques like tutorial sheets and assignments.
4. Seminars.

1. Program Educational Objectives (PEOs) – Vision/Mission Matrix (Relationships are indicated by mark "X")

PEOs	**Mission of department**			
	Higher Learning	Contemporary Education	Technical knowledge	Research
Graduates will have a successful technical or professional careers, including supportive and leadership roles on multidisciplinary teams	X	X	X	X
Graduates will be able to acquire, use and develop skills as required for effective professional practices		X	X	
Graduates will be able to attain holistic education that is an essential prerequisite for being a responsible member of society	X		X	
Graduates will be engaged in life-long learning, to remain abreast in their profession and be leaders in our technologically vibrant society.	X		X	X

2. Course Objectives-Course Outcomes Relationship Matrix

(Relationships are indicated by mark "X)

Course-Outcomes / Course-Objectives	1	2	3	4	5	6	7
1	X	X		X		X	X
2	X			X	X		X
3	X	X	X	X		X	X
4		X	X	X	X	X	
5	X		X		X		X
6	X	X			X	X	
7	X	X	X	X		X	X

3. Course Objectives-Program Outcomes (POs) Relationship Matrix (Relationships are indicated by mark "X")

P-Outcomes / C-Objectives	a	b	c	d	e	f	g	h	i	j	k	l
1	X	X	X		X		X		X			X
2		X		X	X			X			X	X
3	X	X	X	X			X	X	X	X		
4	X	X	X		X				X	X	X	
5	X		X	X			X	X		X	X	X
6		X		X	X		X	X	X		X	X
7	X		X	X	X		X		X	X	X	

4. Course Outcomes-Program Outcomes (POs) Relationship Matrix (Relationships are indicated by mark "X")

P-Outcomes / C-Outcomes	a	b	c	d	e	f	g	h	i	J	k	l
1	H	H	H	M	H		M	H	M		H	H
2		H	M	H			M	M	H	M	M	
3	H		H		M				M	H		M
4	M	H	H	M	H		M	M		M	M	H
5		M					H		H		H	M
6		H	M	M	H			H		M		H
7	H	M		M	M		M	H	M	H	M	M

M- Medium **H-High**

5. Courses (with title & code)-Program Outcomes (POs) Relationship Matrix
(Relationships are indicated by mark "X")

P-Outcomes / Course	a	b	c	d	e	f	g	h	i	j	k	l
Soft Computing Techniques-GR14D4162	X	X	X		X			X	X		X	X

6. Program Educational Objectives (PEOs)-Course Outcomes Relationship Matrix (Relationships are indicated by mark "X")

P-Objectives(PEO) / Course-Outcomes	1	2	3	4
1	X	X		X

2		X		X
3	X	X	X	
4		X	X	X
5	X	X		X
6	X	X	X	
7	X		X	X

7. Assignments & Assessments-Program Outcomes (POs) Relationship Matrix
(Relationships are indicated by mark "X")

Assessments:
1. Internals
2. Assignments
3. Seminars
4. Externals

P-Outcomes / Assessments	a	b	c	d	e	f	g	H	i	j	k	l
1	X			X	X	X	X	X	X	X		
2	X			X	X	X			X			
3	X			X	X	X		X	X	X		X
4	X	X		X			X		X		X	

8. Relationship Matrix (Relationships are indicated by mark "X")

Assessments:
1. Internals
2. Assignments
3. Seminars
4. Externals

P-Objectives (PEOs) / Assessments	1	2	3	4
1		X	X	X
2	X	X		X
3		X		X
4	X	X	X	X

RUBRICS of "Soft Computing Techniques" Course.

OBJECTIVE: Work effectively with others

STUDENT OUTCOME: Ability to function in a multi-disciplinary team

S.No	Student Name	Performance Criteria	Unsatisfactory	Developing	Satisfactor y	Exemplar y	Scor e
			1	2	3	4	
1	Tharun Teja	**Research & Gather Information**	Does not collect any information that relates to the topic.	Collects very little information some relates to the topic	Collects some basic Informatio n most relates to the topic.	Collects a great deal of Informati on all relates to the topic.	5
		Fulfill team role's duty	Does not perform any duties of assigned team role.	Performs very little duties.	Performs nearly all duties.	Performs all duties ofassigne d team role.	5
		Share Equally	Always relies on others to do the work.	Rarely does the assigned work-- often needs reminding.	Usually does the assigned work-- rarely needs reminding.	Always does the assigned work without having to be reminded	5
		Listen to other team mates	Is always talking— never allows anyone else to speak.	Usually doing most of the talking-- rarely allows speak.	Listens, but sometimes talks too much.	Listens and speaks a fair amount.	5

						Average score	5
2.	Nikhil Reddy	**Research & Gather Information**	Does not collect any information that relates to the topic.	Collects very little information --some relates to the topic	Collects some basic informatio n--most relates to the topic	Collects a great deal of informati on--all relates to the topic.	4
		Fulfill team role's duty	Does not perform any duties of assigned team role.	Performs very little duties.	Performs nearly all duties.	Performs all duties of assigned team role.	4
		Share Equally	Always relies on others to do the work.	Rarely does the assigned work--often needs reminding.	Usually does the assigned work--rarely needs reminding.	Always does the assigned work without having to be	4
		Listen to other team mates	Is always talking--never allows anyone else to speak.	Usually doing most of the talking--rarely allows others to speak.	Listens, but sometimes talks too much.	Listens and speaks a fair amount.	4
						Average score	4

3	Ravi Teja	**Research & Gather Information**	Does not collect any information that relates to the topic.	Collects very little information --some relates to the topic	Collects some basic informati o n--most relates to the topic	Collects a great deal of informat i on--all relates to the topic.	3
		Fulfill team role's duty	Does not perform any duties of assigned team role.	Performs very little duties.	Perform s nearly all duties.	Perform s all duties of assigne d team role	3
		Share Equally	Always relies on others to do the work.	Rarely does the assigned work--often needs reminding.	Usually does the assigne d work--rarely needs reminding.	Always does the assigne d work without having to be	3
		Listen to other team mates	Is always talking--never allows anyone else to speak.	Usually doing most of the talking--rarely allows others to speak.	Listens, but sometime s talks too much.	Listens and speaks a fair amount.	3
						Averag e score	3

Assessment process and Relevant Surveys conducted:

9. Constituencies -Program Outcomes (POs) Relationship Matrix (Relationships are indicated mark “X”).

1. **Alumni**
2. **Government employers**
3. **Students**

P-Outcomes / Constituencies	a	b	c	d	e	f	g	h	i	j	k	l
1	X	X	X	X	X	X	X		X	X		X
2	X	X	X	X	X	X	X		X			X
3	X	X			X	X	X	X		X	X	X

Assessment Process and Areas of improvements:

Prepare the following Matrix:

10. The improvements Matrix are summarized below and described in the text that follows.

Hint:

Format:

Proposed change	**Year proposed**	**Year implemented**	**Old version**	**New version**	**Comments**

Soft computing Techniques

Session Plan

S. No	Unit No.	Date	Topic
1	1	03.07.2018	Introduction to Biological Neuron and Artificial Neuron
2	1	04.07.2018	Neuron Structure and Synapse, comparision with ANN
3	1	06.07.2018	Types of Activation Functions
4	1	10.07.2018	Classification Activiation Functions in detail
5	1	11.07.2018	Neural network architecture: Single Layer feedforward N/W
6	1	13.07.2018	Neural network architecture: Multi Layer feedforward N/W
7	1	17.07.2018	Neural network architecture: Recurrent Networks
8	1	18.07.2018	Learning Techniques classification
9	1	20.07.2018	Different types of Learning Rules
10	1	24.07.2018	Perceptron Convergence Rule
11	1	25.07.2018	Introduction to Associative Memories
12	1	27.07.2018	Auto Associative Memory introduction with diagram
13	1	31.07.2018	Hetero Associative Memory
14	2	01.08.2018	Introduction to Perceptron Model and its solution
15	2	03.08.2018	Single Layer Perceptron Model
16	2	07.08.2018	Multilayer Perceptron Model
17	2	08.08.2018	Introduction to Back Propagation Neural Network
18	2	10.08.2018	Back Propagation Learning Algorithm
19	2	14.08.2018	Effect of Learning Rule Coefficient in Back Propagation NN
20	2	17.08.2018	Various types effecting parameters in BPNN
21	2	21.08.2018	List the applications of Neural Networks
22	3	22.08.2018	Introduction to Fuzzy Sets
23	3	24.08.2018	Difference between Fuzzy sets and Classical Sets
24	3	28.08.2018	Properties of Classical Sets and Fuzzy Sets
25	3	29.08.2018	Fuzzy Set Theory Operations
26	3	31.08.2018	Relations of Fuzzy and Crisp/Classical Sets
27	3	07.09.2018	Fuzzy to Crisp Conversion
28	3	11.09.2018	Fuzzy to Crisp Conversion contd..
29	4	12.09.2018	Introduction to Membership Functions in Fuzzy Sets
30	4	14.09.2018	Inference in Fuzzy Logic System
31	4	18.09.2018	If-Then Rules used in Fuzzy Logic
32	4	19.09.2018	Introduction to Fuzzy Implications
33	4	21.09.2018	Introduction to Fuzzy Algorithms
34	4	25.09.2018	Methods of Fuzzification used
35	4	26.09.2018	Methods of Defuzzification
36	4	28.09.2018	Fuzzy Logic Controller Block diagram operation
37	4	03.10.2018	Industrial Applications using Fuzzy Logic Controllers
38	4	05.10.2018	Introduction to Genetic Algorithms
39	5	9.10.2018	Basic Concepts in GA
40	5	10.10.2018	Working principle of Genetic Algorithm with block diagram
41	5	12.10.2018	Procedure steps for using Genetic Algorithm
42	5	16.10.2018	Flow Chart of GA

43	5	16.10.2018	Genetic Algorithm representations
44	5	17.10.2018	Encoding methods and Selection procedure in GA
45	5	19.10.2018	Introduction to Genetic Algorithm operators
46	5	23.10.2018	Mutation and Generational Cycle of GA
47	5	24.10.2018	Applications of GA

Soft computing Techniques

Session Plan

S. No	Unit No.	Date	Topic
1	1	02.07.2018	Introduction to Biological Neuron and Artificial Neuron
2	1	03.07.2018	Neuron Structure and Synapse, comparision with ANN
3	1	04.07.2018	Types of Activation Functions
4	1	09.07.2018	Classification Activiation Functions in detail
5	1	10.07.2018	Neural network architecture: Single Layer feedforward N/W
6	1	11.07.2018	Neural network architecture: Multi Layer feedforward N/W
7	1	16.07.2018	Neural network architecture: Recurrent Networks
8	1	17.07.2018	Learning Techniques classification
9	1	18.07.2018	Different types of Learning Rules
10	1	23.07.2018	Perceptron Convergence Rule
11	1	24.07.2018	Introduction to Associative Memories
12	1	25.07.2018	Auto Associative Memory introduction with diagram
13	1	30.07.2018	Hetero Associative Memory
14	2	31.07.2018	Introduction to Perceptron Model and its solution
15	2	01.08.2018	Single Layer Perceptron Model
16	2	06.08.2018	Multilayer Perceptron Model
17	2	07.08.2018	Introduction to Back Propagation Neural Network
18	2	08.08.2018	Back Propagation Learning Algorithm
19	2	13.08.2018	Effect of Learning Rule Coefficient in Back Propagation NN
20	2	14.08.2018	Various types effecting parameters in BPNN
21	2	20.08.2018	List the applications of Neural Networks
22	3	21.08.2018	Introduction to Fuzzy Sets
23	3	27.08.2018	Difference between Fuzzy sets and Classical Sets
24	3	28.08.2018	Properties of Classical Sets and Fuzzy Sets
25	3	29.08.2018	Fuzzy Set Theory Operations and Relations
26	4	10.09.2018	Introduction to Membership Functions in Fuzzy Sets
27	4	11.09.2018	Inference in Fuzzy Logic System
28	4	12.09.2018	If-Then Rules used in Fuzzy Logic
29	4	17.09.2018	Introduction to Fuzzy Implications
30	4	18.09.2018	Introduction to Fuzzy Algorithms
31	4	19.09.2018	Methods of Fuzzification used
32	4	24.09.2018	Methods of Defuzzification
33	4	25.09.2018	Fuzzy Logic Controller Block diagram operation
34	4	26.09.2018	Industrial Applications using Fuzzy Logic Controllers
35	4	01.10.2018	Introduction to Genetic Algorithms
36	5	03.10.2018	Basic Concepts in GA
37	5	08.10.2018	Working principle of Genetic Algorithm with block diagram
38	5	09.10.2018	Procedure steps for using Genetic Algorithm
39	5	10.10.2018	Flow Chart of GA
40	5	15.10.2018	Genetic Algorithm representations
41	5	16.10.2018	Encoding methods and Selection procedure in GA
42	5	17.10.2018	Introduction to Genetic Algorithm operators

43	5	22.10.2018	Mutation and Generational Cycle of GA
44	5	23.10.2018	Applications of GA

Assignment 1

1. Describe working biological neuron
2. Classify the working operation difference between Biological and Artificial Neuron with a neat diagram
3. Differentiate between Auto Associative and Hetero Associative Memory.
4. Classify the types of Learning Methods
5. Describe any two factors affecting the Back-Propagation Training

Assignment 2

1. Differentiate between types of learning rules
2. Describe the Fuzzy Logic Controller in detail.
3. Illustrate the Fuzzy inference procedures involved in designing Fuzzy Logic Controllers
4. Describe any two parameters that are used for selection in training Back-Propagation Neural Network.
5. What are the important aspects of using Genetic Algorithms

Assignment 3

1. Describe mutation operation used in genetic modelling
2. Consider A = {(x1,0.2), (x2,0.7), (x3,0.4)} and B = {(y1,0.5),(y2,0.6)} are two fuzzy sets defined in the universe of discourse X = {x1,x2,x3} and Y = {y1,y2} respectively. Find cartesian product of A and B.
3. Elaborate any one application based on Fuzzy Logic Controller in detail.
4. Describe any two methods of Defuzzification.
5. Describe max-min composition in Fuzzy Logic Control.

Assignment 4

1. Illustrate the Fuzzy inference procedures involved in designing Fuzzy Logic Controllers
2. Consider A = {(x1,0.2), (x2,0.7), (x3,0.4)} and B = {(y1,0.5),(y2,0.6)} are two fuzzy sets defined in the universe of discourse X = {x1,x2,x3} and Y = {y1,y2} respectively. Find cartesian product of A and B.
3. Elaborate any one application based on Fuzzy Logic Controller in detail.
4. Describe any two methods of Defuzzification.
5. Describe max-min composition in Fuzzy Logic Control.

Assignment 5

1. Describe any two parameters that are used for selection in training Back-Propagation Neural Network.
2. What are the important aspects of using Genetic Algorithms?
3. Describe mutation operation used in genetic modelling
4. Explain the selection procedure for Genetic algorithm
5. Describe any one application of Genetic Algorithm in detail

Tutorial 1

1. List the factors affecting the Back-Propagation Training
2. Describe the properties of classical sets.
3. Define Cardinality of Classical Set and Fuzzy Set with an example
4. What is the use of Fuzzification and Defuzzification?
5. Describe the Fuzzy Logic Controller in detail.
6. Mention the Fuzzy inference procedures involved in designing Fuzzy Logic Controllers
7. What are the important aspects of using Genetic Algorithms
8. List the three simple Genetic Algorithm Operators which are largely used.
9. Consider A = {(x1,0.2), (x2,0.7), (x3,0.4)} and B = {(y1,0.5),(y2,0.6)} are two fuzzy sets defined in the universe of discourse X = {x1,x2,x3} and Y = {y1,y2} respectively. Find cartesian product of A and B.
10. Describe the properties of Fuzzy Sets.
11. Describe any two methods of Defuzzification
12. Elaborate any one application of Fuzzy Logic Controller
13. Describe the operation of Mutation in Genetic Algorithm in detail.
14. Write the differences between Fuzzy Sets and Classical Sets
15. What are the steps involved in designing a Fuzzy Logic Controller?
16. Describe any one application of Genetic Algorithms

Tutorial 2

1. Describe any two factors factors affecting the Back-Propagation Training
2. Describe the Fuzzy Logic Controller in detail.
3. Illustrate the Fuzzy inference procedures involved in designing Fuzzy Logic Controllers
4. Describe any two parameters that are used for selection in training Back-Propagation Neural Network.
5. What are the important aspects of using Genetic Algorithms
6. Describe mutation operation used in genetic modelling
7. Consider A = {(x1,0.2), (x2,0.7), (x3,0.4)} and B = {(y1,0.5),(y2,0.6)} are two fuzzy sets defined in the universe of discourse X = {x1,x2,x3} and Y = {y1,y2} respectively. Find cartesian product of A and B.
8. Elaborate any one application based on Fuzzy Logic Controller in detail.
9. Describe any two methods of Defuzzification.
10. Describe max-min composition in Fuzzy Logic Control.

REVISED Bloom's Taxonomy Action Verbs

Definitions	I. Remembering	II. Understanding	III. Applying	IV. Analyzing	V. Evaluating	VI. Creating
Bloom's Definition	Exhibit memory of previously learned material by recalling facts, terms, basic concepts, and answers.	Demonstrate understanding of facts and ideas by organizing, comparing, translating, interpreting, giving descriptions, and stating main ideas.	Solve problems to new situations by applying acquired knowledge, facts, techniques and rules in a different way.	Examine and break information into parts by identifying motives or causes. Make inferences and find evidence to support generalizations.	Present and defend opinions by making judgments about information, validity of ideas, or quality of work based on a set of criteria.	Compile information together in a different way by combining elements in a new pattern or proposing alternative solutions.
Verbs	• Choose • Define • Find • How • Label • List • Match • Name • Omit • Recall • Relate • Select • Show • Spell • Tell • What • When • Where • Which • Who • Why	• Classify • Compare • Contrast • Demonstrate • Explain • Extend • Illustrate • Infer • Interpret • Outline • Relate • Rephrase • Show • Summarize • Translate	• Apply • Build • Choose • Construct • Develop • Experiment with • Identify • Interview • Make use of • Model • Organize • Plan • Select • Solve • Utilize	• Analyze • Assume • Categorize • Classify • Compare • Conclusion • Contrast • Discover • Dissect • Distinguish • Divide • Examine • Function • Inference • Inspect • List • Motive • Relationships • Simplify • Survey • Take part in • Test for • Theme	• Agree • Appraise • Assess • Award • Choose • Compare • Conclude • Criteria • Criticize • Decide • Deduct • Defend • Determine • Disprove • Estimate • Evaluate • Explain • Importance • Influence • Interpret • Judge • Justify • Mark • Measure • Opinion • Perceive • Prioritize • Prove • Rate • Recommend • Rule on • Select • Support • Value	• Adapt • Build • Change • Choose • Combine • Compile • Compose • Construct • Create • Delete • Design • Develop • Discuss • Elaborate • Estimate • Formulate • Happen • Imagine • Improve • Invent • Make up • Maximize • Minimize • Modify • Original • Originate • Plan • Predict • Propose • Solution • Solve • Suppose • Test • Theory

Anderson, L. W., & Krathwohl, D. R. (2001). A taxonomy for learning, teaching, and assessing, Abridged Edition. Boston, MA: Allyn and Bacon.

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)
Department of Electrical and Electronics Engineering

Academic Year: **2018-19**
Year: **IV**
Semester: **I**

MID Exam – I (Objective)
SOFT COMPUTING TECHNIQUES
Code: GR15A4162

Date: 27/09/2018
Marks:5M

Name: **Roll No:**

Answer All the Questions:

1. A neuron is composed of a nucleus – a cell body known as [ ]

(a) Axon (b) Dendrite (c) Synapse (d) Soma

2. In a linear step signal activation function, the output is equal to "1" if and only if [ ]

(a) output is less than Threshold (b) output is greater than Threshold

(c) output is equal to Threshold (d) Both (c) and (d)

3. Hebbian learning rule is _______________ learning method.as [ ]

(a) Supervised (b) Unsupervised (c) Reinforced (d) Both (b) and (c)

4. For which type of Non Linear Activation function, the following function is obtained
Output $O = sgn\ [I]\ \{I – Input\}$ [ ]

(a) Linear (b) Hard Limiter (c) Piecewise Linear (d) Bipolar Signmoidal

5. Autocorrelators obtain their connection matrix by ______ a pattern's element with every other pattern's element. [ ]

(a) Multiplying (b) Adding (c) Subtracting (d) Dividing

6. In input layer of multilayer perceptron _______________ transfer function is used and for hidden and output layer perceptron's _________________________ transfer function is used.

7. The Error in Back Propagation Neural Network is given as __________________________.

8. Multilayer feedforward networks with non-linear activation functions ___________ _______ surface above the Q-dimensional weight space.

9. If the learning coefficient is ___________________ , the weight vector will overshoot from its ideal position and oscillate.

10. Competitive learning is an example for _______________________ learning method.

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)
Department of Electrical and Electronics Engineering

Academic Year: **2018-19**
Year: **IV**
Semester: **I**

MID Exam – I (Subjective)
Soft Computing Techniques
Code: GR15A4162

Date: **27.09.2018**
Duration: **90 min**
Max Marks: **15**

Note: Answer any three questions. All questions carry equal marks.

1. Discuss the types of activation function used in ANN? [**CO 1**]
2. Classify the types of associative memories with neat diagram? [**CO 3**]
3. Describe the method of steepest descent learning? [**CO 2**]
4. Describe the operation of multi layer perception model? [**CO 2**]

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)
Department of Electrical and Electronics Engineering

Academic Year: **2018-19**
Year: **IV**
Semester: **I**

MID Exam – II (Objective)
SOFT COMPUTING TECHNIQUES
Code: GR15A4162

Date: 27/10/2018
Marks:5M

Name: **Roll No:**

Answer All the Questions:

1. If the learning coefficient is ______________, no learning takes place [ ]

   (a) Greater than 1 (b) Infinite (c) Zero (d) One

2. The truth values of traditional set theory is _________ and that of Fuzzy Set is____. [ ]

   a) Either 0 or 1, between 0 & 1 b) Either 0 or 1, Either 0 & 1

   c) Between 0 or 1, between 0 & 1 d) Between 0 or 1, Either 0 & 1

3. The values of the set membership is represented by [ ]

   a) Discrete Set b) Degree of truth c) Probabilities d) Both (c) and (d)

4. Fuzzy logic is usually represented as [ ]

   a) IF-THEN-ELSE rules b) IF-THEN rules

   c) Both (a) and (b) d) None of the mentioned

5. Which of the following can be founded in Genetic Algorithms?

   a) Mutation b) Cross over c) reproduction d) All the above

6. Soft Computing techniques deals with____________________, __________________ and __________________.

7. One of the Fuzzy Inferring Procedure is ___________________________.

8. Cardinality of a fuzzy set defined as _______________________________.

9. Centroid method is also known as _____________________.

10. If a = 1010 0110 then using "shift right operator" a >>2 = ____________________________.

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)
Department of Electrical and Electronics Engineering

Academic Year: **2018-19**
Year: **IV**
Semester: **I**

MID Exam – II (Subjective)
Soft Computing Techniques
Code: GR15A4162

Date: **27.10.2018**
Duration: **90 min**
Max Marks: **15**

Note: Answer any three questions. All questions carry equal marks.

1. Describe any two parameters that are used for selection in training Back-Propagation Neural Network. [**CO 4**]
2. Illustrate the Fuzzy Inference procedures involved in designing the Fuzzy Logic Controllers. [**CO 5**]
3. Elaborate any one application based on Fuzzy Logic Controller in detail. [**CO 6**]
4. Describe Mutation operator used in Genetic Modelling. [**CO 7**]

Course Attainment Analysis (Soft Computing Techniques)

B. Tech EEE 2018-2019 (A Section)

GR15A4162-Soft Computing Technologies (MID 1)

Date: 27 Sep 2018

Roll Number	1	2	3	4
	CO 1	CO 3	CO 2	CO 2
15241A0201		5	5	3
15241A0202		5	5	
15241A0203		5	4	
15241A0204	5	5	5	4
15241A0205	5	5		5
15241A0206		5	2	4
15241A0207	5	5	3	5
15241A0208	5	5	5	4
15241A0209		5	5	2
15241A0211	5	5		5
15241A0212		5	5	5
15241A0213		5	5	5
15241A0214		5	5	4
15241A0215		5	5	
15241A0216		5	5	
15241A0217		5	5	5
15241A0218		5	5	
15241A0219		5	5	
15241A0220		5	5	5
15241A0221		5	4	
15241A0222	4	5	5	
15241A0224			3	
15241A0225		5	5	5
15241A0226	4	5	5	5
15241A0227		5	5	5
15241A0229		5	5	5
15241A0230		5	4	
15241A0231		5	5	
15241A0232	1	5	5	
15241A0233		5	5	2
15241A0234		5	4	4
15241A0235	5	5		5
15241A0236	5	5		5
15241A0237	3	5	5	
15241A0238		5	5	
15241A0239		5	5	2
15241A0240		5	5	2
15241A0241		5	4	4
15241A0242		5	5	5
15241A0244		5		2
15241A0245		5	3	3

15241A0246		5	5	3
15241A0247		5	5	5
15241A0248		5	5	5
15241A0249		5	5	
15241A0250		5	4	
15241A0251		5	2	
15241A0253		5	5	
15241A0254			1	2
15241A0255	5	5	5	
15241A0256		5	5	3
15241A0257	5	5		5
15241A0258		5	5	
15241A0259		5	5	5
15241A0260		5	5	5
16245A0201		3		
16245A0202		5	3	3
16245A0203		5	5	5
16245A0204		5	5	
16245A0205		5	4	
16245A0206		5	4	
16245A0207	5	5		4
16245A0208		5	4	
16245A0209		5	4	3
16245A0210		5	4	
16245A0211		5	5	2
16245A0212	1	5	5	5
Grand Total	**63**	**323**	**266**	**165**
NSA	**15.0**	**65.0**	**59.0**	**41.0**
Attempt %=(NSA/Total no of students)*100	**22.4**	**97.0**	**88.1**	**61.2**
Average (attainment)= Total/NSA	**4.2**	**5.0**	**4.5**	**4.0**
Attainment%= (Avg/max. Marks for question)*100	**84.00**	**99.38**	**90.17**	**80.49**

CO 1	**84.00**
CO 3	**99.38**
CO 2	**85.33**

Course Attainment Analysis (Soft Computing Techniques)

B. Tech EEE 2018-2019 (B Section)

GR15A4162-Soft Computing Technologies (MID 1)

Date: 27 Sep 2018

Roll Number	1	2	3	4
	CO 1	CO 3	CO 2	CO 2
15241A0261		5	5	
15241A0263	4		5	3
15241A0264		5	5	4
15241A0265		5	5	3
15241A0266		5	5	5
15241A0267		5	5	4
15241A0268		5	5	4
15241A0269	3	3		
15241A0270		4	5	
15241A0271		5	3	
15241A0272	5	5		
15241A0273	2	5	5	
15241A0274	5	5	5	
15241A0275	5	5	5	
15241A0276		5	5	
15241A0277		5	5	
15241A0278	2	3		2
15241A0279		5	5	
15241A0280		5	4	2
15241A0281	3	5	5	
15241A0282	5	5		5
15241A0283	2			
15241A0284	5	5	5	4
15241A0285	3	5	5	
15241A0286		5	2	3
15241A0287		5	3	3
15241A0288	2	4	3	
15241A0289	5			5
15241A0290	5		5	
15241A0291	5		5	
15241A0292	5	5		5
15241A0293	4	5		5
15241A0294	5	3	5	
15241A0295	5	5	5	
15241A0296		5	5	
15241A0297		5	5	5
15241A0298		5	5	5
15241A0299		4	4	3
15241A02A1	5	5	4	
15241A02A2		5	5	4
15241A02A3		5	5	5

15241A02A5		5	5	5
15241A02A6		5	5	3
15241A02A8	5	4		
15241A02A9		5	5	5
15241A02B0		5	4	4
15241A02B1		5	5	
15241A02B2		5	4	
15241A02B3		5	5	5
15241A02B4	2	5	5	
15241A02B5		5	5	5
15241A02B6		5	5	5
15241A02B7		5	5	5
15241A02B8	5	5	4	
15241A02B9	5	5	5	
15241A02C0		5	4	5
16245A0213		5	5	4
16245A0214	5	4	5	4
16245A0215		4	3	4
16245A0217	5	5		5
16245A0218		5		
16245A0219		5	5	5
16245A0220		5	5	5
16245A0221		5	4	3
16245A0222		5	3	
16245A0223		5	2	4
16245A0224	2	5		3
Grand Total	**114**	**298**	**251**	**158**
NSA	**28.0**	**62.0**	**55.0**	**38.0**
Attempt %=(NSA/Total no of students)*100	**41.8**	**92.5**	**82.1**	**56.7**
Average (attainment)= Total/NSA	**4.1**	**4.8**	**4.6**	**4.2**
Attainment%= (Avg/max. Marks for question)*100	**81.43**	**96.13**	**91.27**	**83.16**

CO 1	**81.43**
CO 3	**96.13**
CO 2	**87.22**

Course Attainment Analysis (Soft Computing Techniques)

B. Tech EEE 2018-2019 (A Section)

GR15A4162-Soft Computing Technologies (MID 2)

Date: 31 Oct 2018

Roll Number	1	2	3	4
	CO 4	CO 5	CO 6	CO 7
15241A0201		5	5	3
15241A0202		5	5	
15241A0203		5	4	
15241A0204	5	5	5	4
15241A0205	5	5		5
15241A0206		5	2	4
15241A0207	5	5	3	5
15241A0208	5	5	5	4
15241A0209		5	5	2
15241A0211	5	5		5
15241A0212		5	5	5
15241A0213		5	5	5
15241A0214		5	5	4
15241A0215		5	5	
15241A0216		5	5	
15241A0217		5	5	5
15241A0218		5	5	
15241A0219		5	5	
15241A0220		5	5	5
15241A0221		5	4	
15241A0222	4	5	5	
15241A0224			3	
15241A0225		5	5	5
15241A0226	4	5	5	5
15241A0227		5	5	5
15241A0229		5	5	5
15241A0230		5	4	
15241A0231		5	5	
15241A0232	1	5	5	
15241A0233		5	5	2
15241A0234		5	4	4
15241A0235	5	5		5
15241A0236	5	5		5
15241A0237	3	5	5	
15241A0238		5	5	
15241A0239		5	5	2
15241A0240		5	5	2
15241A0241		5	4	4
15241A0242		5	5	5
15241A0244		5		2
15241A0245		5	3	3

15241A0246		5	5	3
15241A0247		5	5	5
15241A0248		5	5	5
15241A0249		5	5	
15241A0250		5	4	
15241A0251		5	2	
15241A0253		5	5	
15241A0254			1	2
15241A0255	5	5	5	
15241A0256		5	5	3
15241A0257	5	5		5
15241A0258		5	5	
15241A0259		5	5	5
15241A0260		5	5	5
16245A0201		3		
16245A0202		5	3	3
16245A0203		5	5	5
16245A0204		5	5	
16245A0205		5	4	
16245A0206		5	4	
16245A0207	5	5		4
16245A0208		5	4	
16245A0209		5	4	3
16245A0210		5	4	
16245A0211		5	5	2
16245A0212	1	5	5	5
Grand Total	**63**	**323**	**266**	**165**
NSA	**15.0**	**65.0**	**59.0**	**41.0**
Attempt %=(NSA/Total no of students)*100	**22.4**	**97.0**	**88.1**	**61.2**
Average (attainment)= Total/NSA	**4.2**	**5.0**	**4.5**	**4.0**
Attainment%= (Avg/max. Marks for question)*100	**84.00**	**99.38**	**90.17**	**80.49**

CO 4	**84.00**
CO 5	**99.38**
CO 6	**90.17**
CO 7	**80.49**

Course Attainment Analysis (Soft Computing Techniques)

B. Tech EEE 2018-2019 (B Section)

GR15A4162-Soft Computing Technologies (MID 2)

Date: 31 Oct 2018

Roll Number	1	2	3	4
	CO 4	CO 5	CO 6	CO 7
15241A0261		5	5	
15241A0263	4		5	3
15241A0264		5	5	4
15241A0265		5	5	3
15241A0266		5	5	5
15241A0267		5	5	4
15241A0268		5	5	4
15241A0269	3	3		
15241A0270		4	5	
15241A0271		5	3	
15241A0272	5	5		
15241A0273	2	5	5	
15241A0274	5	5	5	
15241A0275	5	5	5	
15241A0276		5	5	
15241A0277		5	5	
15241A0278	2	3		2
15241A0279		5	5	
15241A0280		5	4	2
15241A0281	3	5	5	
15241A0282	5	5		5
15241A0283	2			
15241A0284	5	5	5	4
15241A0285	3	5	5	
15241A0286		5	2	3
15241A0287		5	3	3
15241A0288	2	4	3	
15241A0289	5			5
15241A0290	5		5	
15241A0291	5		5	
15241A0292	5	5		5
15241A0293	4	5		5
15241A0294	5	3	5	
15241A0295	5	5	5	
15241A0296		5	5	
15241A0297		5	5	5
15241A0298		5	5	5
15241A0299		4	4	3
15241A02A1	5	5	4	
15241A02A2		5	5	4
15241A02A3		5	5	5

15241A02A5		5	5	5
15241A02A6		5	5	3
15241A02A9		5	5	5
15241A02B0		5	4	4
15241A02B1		5	5	
15241A02B2		5	4	
15241A02B3		5	5	5
15241A02B4	2	5	5	
15241A02B5		5	5	5
15241A02B6		5	5	5
15241A02B7		5	5	5
15241A02B8	5	5	4	
15241A02B9	5	5	5	
15241A02C0		5	4	5
16245A0213		5	5	4
16245A0214	5	4	5	4
16245A0215		4	3	4
16245A0217	5	5		5
16245A0218		5		
16245A0219		5	5	5
16245A0220		5	5	5
16245A0221		5	4	3
16245A0222		5	3	
16245A0223		5	2	4
16245A0224	2	5		3
Grand Total	**109**	**294**	**251**	**158**
NSA	**27.0**	**61.0**	**55.0**	**38.0**
Attempt %=(NSA/Total no of students)*100	**40.9**	**92.4**	**83.3**	**57.6**
Average (attainment)= Total/NSA	**4.0**	**4.8**	**4.6**	**4.2**
Attainment%= (Avg/max. Marks for question)*100	**80.74**	**96.39**	**91.27**	**83.16**

CO 4	**80.74**
CO 5	**96.39**
CO 6	**91.27**
CO 7	**83.16**

CODE: GR15A4162 | **GR 15** | **SET - 1**

IV B. Tech I Semester Regular Examinations, Nov/Dec 2018
Soft Computing Techniques
(Electrical and Electronics Engineering)

Time: 3 hours **Max Marks: 70**

PART – A

Answer ALL questions. All questions carry equal marks.

*****

10 * 2 Marks = 20 Marks

1). a Define the working operation of Synapse in the biological neuron **[2]**

b Draw the different architectures of Neural Networks **[2]**

c Define the learning methods involved in studying the algorithm of Back Propagation Network. **[2]**

d List the factors affecting the Back-Propagation Training **[2]**

e Mention the properties of classical sets **[2]**

f Define Cardinality of Classical Set and Fuzzy Set **[2]**

g What is the use of Fuzzification and Defuzzification? **[2]**

h Mention the Fuzzy inference procedures involved in designing Fuzzy Logic Controllers **[2]**

i What are the important aspects of using Genetic Algorithms **[2]**

j List the three simple Genetic Algorithm Operators which are largely used. **[2]**

PART – B

Answer any FIVE questions. All questions carry equal marks.

*****

5 * 10 Marks = 50 Marks

2.	a. Describe the working operation of biological neuron with a neat diagram. b. Classify the types Associative memories in detail	**[10]**
3.	Write the steps involved in training Input Layer, Hidden Layer and Output Layer of the Back Propagation Network	**[10]**
4.	a. Consider A = $\{(x_1,0.2),(x_2,0.7),(x_3,0.4)\}$ and B = $\{(y_1,0.5),(y_2,0.6)\}$ are two fuzzy sets defined in the universe of discourse X = $\{x_1,x_2,x_3\}$ and Y = $\{y_1,y_2\}$ respectively. Find cartesian product of A and B. b. Describe the properties of Fuzzy Sets.	**[10]**
5.	a. Describe any two methods of Defuzzification b. Elaborate any one application of Fuzzy Logic Controller	**[10]**
6.	Describe the operation of Mutation in Genetic Algorithm in detail.	**[10]**
7.	a. What is the use of Activation Function in ANN? And list the different types of	**[10]**

	Activation Function used in designing ANN b. Describe the effect of Learning Coefficient in Back Propagation Network	
8.	Write short notes on a. Difference between Fuzzy Sets and Classical Sets b. Steps involved in designing a Fuzzy Logic Controller c. Applications of Genetic Algorithms	**[3+3+4]**

*****

Gokaraju Rangaraju Institute of Engineering & Technology
(Autonomous)

Summation of Teacher Appraisal by Student
Academic Year 2018-19

Name of the Instructor	R. Anil Kumar
Faculty ID	657
Branch	EEE
Class and Semester/Section	IV / I / B
Academic Year	2018-19
Subject Title	SCT
Total No. of Responses/class strength	28/71

Average rating on a scale of 4 for the responses considered:

S. No	Questions of Feedback	Average
1	How do the teacher explain the subject?	3.3333333333333335
2	The teacher pays attention to	3.25
3	The Language and communication skills of the teacher is	3.3611111111111112
4	Is the session Interactive?	3.3611111111111112
5	Rate your teacher's explanation in clearing the doubts	3.3333333333333335
6	Rate your teachers commitment in completing the syllabus	3.25
7	Rate your teachers punctuality	3.2222222222222223
8	Rate your teachers use of teaching aids	3.1666666666666665
9	Rate your teacher's guidance in other activities like NPTEL, Moodle, Swayam, Projects.	3.25
10	What is your overall opinion about the teacher?	3.2777777777777777

Net Feedback on a scale of 1 to 4: 3.2805555555555559