International Journal of Sciences & Applied Research

www.ijsar.in

Comparative Analysis of implementation of Logic AND Gate using Hebb and Perceptron Neural Network Algorithms

Neha Bhardwaj Bhullar*

AP (ECE), PEC University of Technology, Chandigarh, India.

Correspondence Address: *Neha Bhardwaj Bhullar, AP (ECE), PEC University of Technology, Chandigarh, India.

Abstract

Logic Gates are the basic building blocks for designing and integrating a chip.

The intent of the paper is to:

1. Mathematically realize the basic AND gate using available Hebb and Perceptron Neural Net algorithms.
2. To compare the convergence, feasibility of both algorithms.
3. To check if the realized gate training inputs are linearly separable or not.
4. To check if training algorithms support both binary and bipolar input data or not.

Keywords: ANN, Hebb Net, Perceptron, bias

Introduction

The main constituent of human brain is neuron. Human brain constitutes about billions of neurons with trillion interconnections. Neuron is the basic constituent of any Artificial Neural Network (ANN). ANN is dynamic in the way that it exploits the data in a way similar to human brain [1].The basic Logic gates are implemented using available neural net algorithms. The advantages of neural nets are Adaptive Learning, fault tolerance, generalization ability etc.

The neural nets are faster (application specific), wherein the cycle time for a single execution steps is in the range of nanoseconds.

Digital Logic Gates are the basics of any circuit. The basic idea is that if the basic gates can be realized using training algorithms of various neural nets, complex applications can also be solved by training a specific net for the same eg pattern classification, pattern association etc.

The instigators of cybernetics are exploiting neurology and current knowledge of human brains to develop fast modern computing machines[5].

Various logic gates and digital circuits based on them can be realised using these neural net algorithms[9],[10]. There are two ways to realize any logic circuit by using neural networks:

1. By doing mathematical calculations and employing formula of the neural net algorithm used and continuing the epochs till convergence is achieved.
2. By doing simulations based on MATLAB/SIMULINK/LAB VIEW.

In this paper, AND Gate is mathematically realized using Hebb and Perceptron Algorithms and their performance is compared afterwards[5].

The different kinds of neural nets deployed in this paper are:

1. Hebb Net
2. Perceptron

The basic truth table for AND Gate is:

Table 1: Truth table for AND Gate.

X_1	X_2	T(target o/p)
0	0	0
0	1	0
1	0	0
1	1	1

Realization using Hebb Net

It was developed by Donald Hebb in 1949.It states that if two neurons are fired simultaneously, then the strength of the connection between them will be increased.[2]

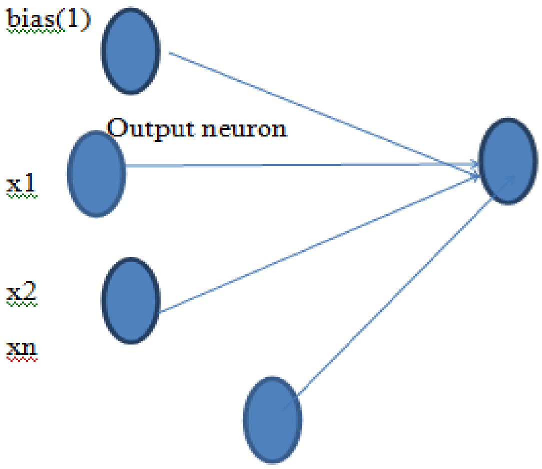


Figure 1: Hebb Net Architecture.

The training algorithm used adjusts the weights of input and output layer and the new weights are formulated using the formula:

W(new)=w(old) + XY

B(new)=B(old)+Y

Where:X=input neuron and Y =output neuron

The epochs are repeated till the weight converges or calculated output is equal to target output (mentioned in table 1)

On implementing AND Gate mathematically using Hebbnet, convergence is achieved after one epoch.The final weights obtained are:

W_1(new)=2 W_2(new)=2 B(new)=-2

A separating line can be drawn and the graph obtained is:

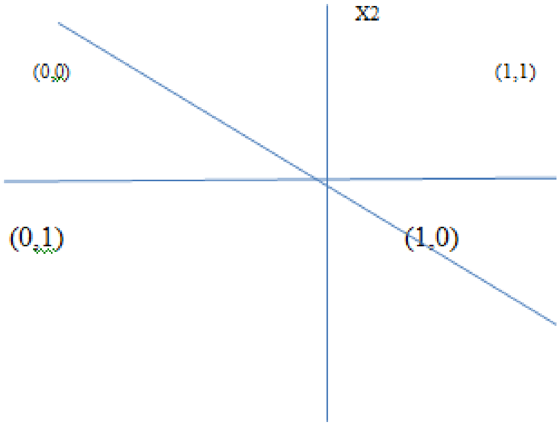
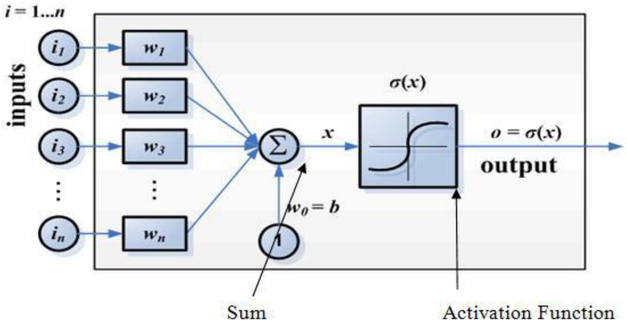


Figure 2: Figure depicting Linear Separability of AND Gate using Hebb Net.

The same training algorithm, when applied to EX-OR Gate was unable to draw a decision boundary, hence EX-OR gate cannot be implemented using Hebb Net.

Perceptron Network



It was developed by Rosenblatt and Minsky and Papert. It is an algorithm of supervised learning and can be used for pattern classification. [3],[4]

	Attainment of weight convergence	Linear Separability	Types of inputs	Initial values of weights and bias	Learning Parameter(α)
HebbNet	After one epoch(appx)	Yes	Preferred bipolar inputs	Zero	Not used
Perceptron	After one epoch	Yes	Both binary and bipolar inputs	Zero	Small random value between 0and 1(1 in this case)

As shown in the diagram above a typical perceptron can have many inputs and these inputs have all individual weights. The perceptron weights can either amplify or de-amplify the original input signal. For example, if the input is 1 and the input's weight is 0.2 the input will be decreased to 0.2.

The weights and bias in this training algorithm is updated using mathematical formulae:

If target values (t) are not equal to calculated output values(Y)

W(new)=w(old) + αxy

B(new)=B(old)+αt

Else: W(new)=w(old)

B(new)=B(old)

Training process is continued till weight convergence is achieved.

After completion of first epoch, the new weights are calculated to be:

W_1(new)=1 W_2(new)=1 B(new)= -1

A decision boundary for this can also be drawn like HebbNet. Perceptron algorithm can be used for both binary and bipolar patterns.

Conclusion

This article implemented the logic gate-AND using Hebb and Perceptron algorithm. Both the realizations were compared in terms of linear separability, weight convergence etc.

Future work

In future, more complicated logic circuits based on these gates can be realized and also the performance can be compared.

References

[1] Jacobson L, “Introduction to Artificial Neural Network”. www.theprojectpot.com/tutorials, 2013.

[2] Hebb, D.O. “The Organization of Behavior. New York: Wiley & Sons”, 1949.

[3] Rosenblatt, Frank (1957), “The Perceptron--a perceiving and recognizing automaton” Report 85-460-1, Cornell Aeronautical Laboratory.

[4] Bishop, Christopher M,“ Pattern Recognition and Machine Learning” Springer 2006.

[5] Alan P. O. et al, “Adaptive Boolean Logic Using Ferroelectrics Capacitors as Basic Units of Artificial Neurons” www.electronic-tutorials.ws.

[6] Martin T. H., Howard B. D. and Mark H. B, “Neural Network Design,”University of Colorado Bookstore (2006).

[7] S. Soundarya et al, “Design and performance analysis of AND Gate using synaptic inputs forNeural Network Application” IJIRST, 2015.

[8] Ele, Sylvester I et al, “Artificial Neuron Network Implementation of Boolean Logic Gates by Perceptron and Threshold Element as Neuron Output Function”,

International Journal of Science and Research, May 2013.

[9] Bishop, “Pattern recognition and machine learning.” Springer, October 2007.

[10] David C.T. Tai et al, “Logic Gate Analysis with Feed Forward Neural Network” WHAMPOA - An Interdisciplinary Journal 51(2006) 235-251.