International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 01 | Jan-2016 www.irjet.net p-ISSN: 2395-0072

NEURAL NETWORK IMPLEMENTATION CONTROL MOBILE ROBOT

S. Parameshwara[1], Manjunath A. C.[2], Vishnu Bhat Yalakki[2], Madhu S.[2], Amaresh Hiremath[2]

[1] Assistant Professor, Department of Electronics and Communication Engineering, The National Institute of Engineering, Mysuru, India-570008

[2] Students, Department of Electronics and Communication Engineering, The National Institute of Engineering, Mysuru, India-570008

Abstract - ***This paper presents a design of a low cost autonomous vehicle based on artificial neural network for navigation in unknown environments. The vehicle is equipped with IR Transmitter and IR Receiver which is used to find the obstacle in the path of the vehicle. The presence of the obstacle at different positions in the environment is provided to the network through the corresponding input pattern. The network then accepts this input pattern. The network then accepts this input pattern and generates corresponding output pattern. Motor Driver is used to drive the motors, all interfaced to an Atmega 2560 microcontroller. The neural network running inside the microcontroller is a multilayer feed forward network with back – propagation training algorithms. The network is trained offline with sigmoid as activation function for neurons. The input pattern used to train the network forms only a subset of the possible inputs. Results have shown that the robot is able to react to inputs which are different from those used for training the network appropriately. Also it is found that up to six neurons can be implemented in hidden layer with this technique.***

Key Words: *Artificial neural network, Low cost, Atmega 2560, Neurons*

1. INTRODUCTION

Neural networks can be artificial or biological neural networks.

1.1 Biological neural network

The human brain provides proof of the existence of massive neural networks that can succeed at those cognitive, perceptual and control tasks in which humans are successful. The brain is capable of computationally demanding perceptual acts (e.g. recognition of face, speech) and control activities (e.g. body movements and body functions). The advantage of the brain is its effective use of massive parallelism, the highly parallel computing structure, and the imprecise information – processing capability.

1.2 Artificial neural networks

Artificial neural networks (ANNs) are computational models inspired by an animal's central nervous system (in particular the brain) which is capable of machine learning as well as pattern recognition Artificial neural network are generally presented as systems of interconnected "neurons" which can compute values from inputs. There is no signal formal definition of what an artificial neural network is. However, a class of statistical models may commonly be called "Neural" if they process the following characteristics.

- Consists of sets of adaptive weights, i.e. numerical parameter that are turned by a learning algorithms
- They are capable of approximate ting non – linear function of their inputs. The adaptive weights are conceptually connecting strengths between neurons, which are activated during training and prediction.

In modern implementations of artificial neural networks, the approach inspired by biology has been largely abandoned for a more practical approach based on statistics and signal processing. In some of these systems, neural networks or part of neural networks (like artificial neurons) form components in larger systems that combine adaptive and non – adaptive element.

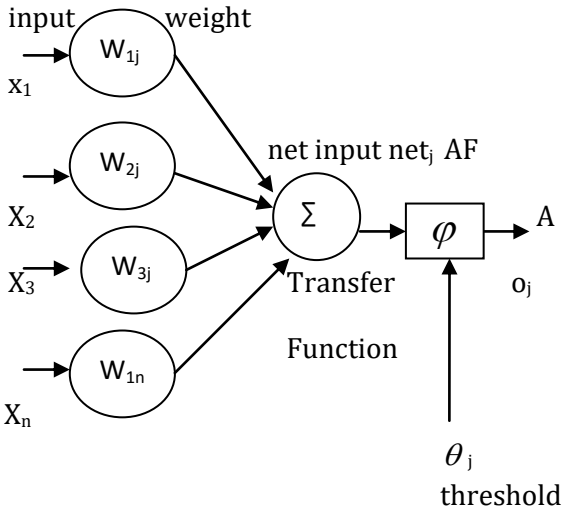
Artificial neural networks have been developed as generalizations of mathematical models of biological nervous systems. A first wave of interest in neural network emerged after the introduction of simplified neurons by McCulloch and Pitts. ANN's can be used to solve real world problems using machine intelligence.

The basic processing elements of neural network are called artificial neurons, or simply neurons or nodes.

1.3 Why Neuron Model?

A typical artificial neuron and the modelling of a multilayered neural network are illustrated in the fig. The signal flow from inputs (x1,…….,xn) is considered to be unidirectional, which are indicated by arrows, as is a neuron's output signal flow (O) The neuron output signal is given by the following relationship:

$$O_j = \theta \left(\sum_{i=1}^{n} w_{ij} x_i \right)$$



AF- Activation function

A – Activation

Fig -1: Artificial neuron

2. BACK PROPAGATION ALGORITHMS

The simple perception is just able to handle linearly separable or linearly independent problems. By taking the partial derivative of the error of the network with respect to each weight, we will learn a little about the direction the error of the network is moving. In fact, if we take the negative of this derivation and then proceed to add it to the weight, the error will decrease until it reaches local minima. This makes sense because if the derivative is positive, this tells us that the error is increasing when the weight is increasing. The obvious thing to do then is to add and a negative value to the weight and vice versa if the derivative is negative derivatives calculated in the layer down steam), this algorithm has been called the back propagation algorithm.

2.2 Back propagation training algorithm

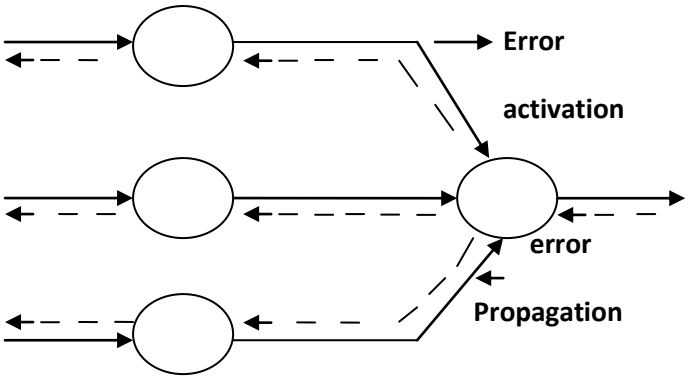


Fig- 2 Back propagation representation

Back propagation algorithms adjust the weights of Neural network in order to minimizes the networks total mean squared error.

3. WORKING AND IMPLEMENTATION

The overall block diagram of the project is as shown in Fig.3

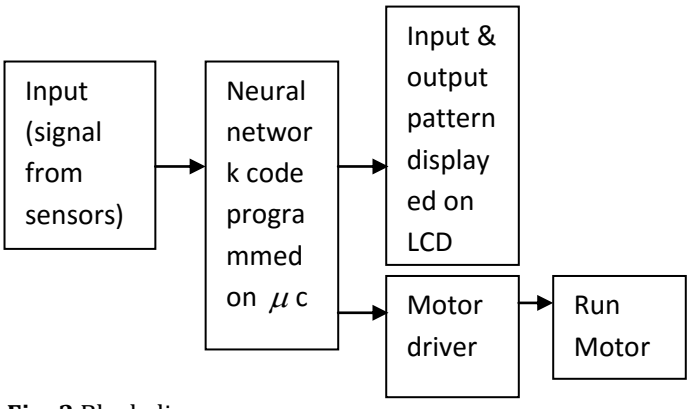


Fig- 3 Block diagram

The robot has to sense its environment in order to move avoiding obstacles. For this purpose the IR proximity sensors are used. These sense the nearby objects within 10 to 13 centimeters of range and provide this information to neural network. The neural network then computers the appropriate motion of the robot and is made to provide the corresponding signals to the motor driver which in turn runs the motor. Also the input and output pattern generated are made to display on the LCD interfaced to the robot. This procedure is illustrated in the block diagram above.

3.2 Model Implementation

The cost of neural networks is given by

$$J(\theta) = \frac{1}{m} \sum_{i=1} m \sum_{k=1} k [-y^i_k \log (h\theta x^i)_k)-(1-y_k{}^i)\log ((1-h\theta x^i)_k)]$$

Where m denotes the number of training examples. Note that the terms corresponding to the bias are not regularized. The gradient of the sigmoid function can be compared as in the remainder of this section. We describe the back propagation algorithm to compute the gradients of the neural network cost function. Once we have computed the gradients, we will be able to train neural network by minimizing the cost function j (θ).

The intuition behind the back propagation algorithm is as follows. Given a training example (xi;yi) , we will first run a ' forward pass' to compute an "error term " δ 1j that measures how much that nodes was "responsible " for any errors in our output. For an output node, we can directly measure the difference between the network's activation and the true target value, and use that to define δ 3j (since layer 3 is the output layer). For the hidden units, we will compute δ 1j based on a weighted average of the error terms of the nodes in layer (1+1).

Minimize a continuous differentiable multivariate function. Starting pointing given by "X" (D by 1), and the function named in the string "f", must return a function value and a vector of partial derivatives. The Polack – Ribiece flavor of conjugate gradients is used to compute search directions, and a line search using quadratic and cubic polynomial approximations and Wolfe- Powell stopping criteria is used together with the slope ratio method for guessing initial step sizes. Additionally a bunch of cheeks are made to make sure that exploration is taking place and that extrapolation will not be unboundedly large. The "length" gives the length of the run: if it is positive, it gives the maximum number of line searchers, if negative its absolute givens the maximum allowed number of function evaluations.

Application of neural networks:

- Sales forecasting
- Industrial process control
- Customer research
- Data validation
- Risk management
- Target marketing

4. CONCLUSIONS

Experimental results have shown the validity of the system in complex environments containing stationary as well as moving obstacles. The present system is designed to avoid obstacles that will come in its way without reduction in its speed. In order to code up with real situations, speed must be available. The neural controller can be further refined by adding speed input to it so that the robot will move faster in the absence of an obstacle and reduce its speed relative to obstacle position provided by sensors. The number of speed levels will directly correspond to the number of regions in sensors information. Also the robot has no knowledge of destination. In order to reach a specified destination, GPS information using a GPS receiver can be fused with IR sensors to find a hurdle free path to destination. The modified trained network can then be implemented using the same microcontroller.

5. FUTURE SCOPE

Robots can be integrated with GPS modules so as we can fix starting point and target point. The robot reaches the target point without colliding to the obstacle present in between the path.

Major development in robot with artificial intelligence includes:

- Neuroscience
- Human robot Interaction
- Function in society

REFERENCES

[1] http://www..cousera.org
[2] http:/www.robotics.org/content-details.cfm/Industrail-Robotics-Featured-Articles/New-Applications-for-mobile-Robots/content_id/3362
[3] http://www.diva-portal.org/smash/get/diva2:9477/FULLTEST01.Pdf
[4] http://www.cs.bham.ac.uk/~jxb/NN/nn.html
[5] http://robotics.hobbizine.com/arduinoann.htmal