Soft Magnetic Encoder Architecture Employing RBF Neural Networks

[1]N. Sangeetha, [2]Dr. Rajashekar J. S
[1]Student, [2]Prof &Head of the department
[1]Instrumentation Technology,
[1]Dayananda Sagar College of Engineering, Bengaluru, India

Abstract: This paper proposes an alternative design of magnetic encoder for analog angle measurement based on Radial Basis Function (RBF) neural network. In order to realize analog angle output, multiple linear Hall-effect sensors and a cylindrical magnet are used. RBF neural network is used to approximate the multi-dimensional nonlinear function between the vector of sensor values and the angle output. For successful approximation of the function between inputs and output, the parameters of the RBF network needs to be determined by training. Training data consisting of vector of sensor values and the corresponding angle value at various angular positions of the rotatory shaft needs to provided as input to the RBF neural network trainer. The RBF neural network with trained parameters can be used to output the angle value for the given sensor values as input. The trained RBF neural network operation is implemented using 8-bit microcontroller. With this design and implementation, the angle value at various positions of the rotatory shaft is obtained with sufficient level of accuracy. This design of the magnetic encoder also allows some flexibility in terms of placement of the sensors and magnet.

***IndexTerms* - Rotary encoder, magnetic encoder, multisensor data fusion, Hall effect sensor, ANN, RBF neural networks, analog angular measurement.**

I. Introduction

In various industrial scenario, continuous angle output is a necessary requirement. Industrial equipment such as CNC machines, 3D printers, robots etc. use electrical motor for precise positioning. The positioning accuracy depends on the accuracy of the movement and control of the rotation of motor. For controlling the motion, precise analog angle measurement is needed.

Magnetic encoders detect the magnetic field and convert it into electrical signal. They are immune to dust, dirt, humidity etc. and hence they are robust and immune to the environmental conditions. They also possess excellent shock resistance. Hence magnetic encoders are a popular choice in industry for detection of angular position of rotary shaft.

This project proposes an alternative design of absolute magnetic encoder for analog angle measurement based on RBFNN. In order to realize analog angle output, multiple linear Hall-effect sensors and a cylindrical magnet are used.

Linear Hall-effect sensor is used to measure the magnetic field density around it. It generates electrical signal in response to the presence of magnetic field around it. Its output signal voltage varies linearly with magnetic flux density.

The cylindrical magnet is attached to the rotary shaft and sensors are kept stationary around the rotary shaft. As the shaft rotates, the magnet also rotates and the distance between sensors and magnet changes. The angular position of the shaft can be determined based on sensor values. In order to uniquely identify the position of the magnet, minimum 3 linear Hall-effect sensors are needed.

The placement of multiple sensors and magnet needs to be done so as to ensure unique sensor values for each angular position of the magnet. Once such configuration is found then the task reduces to mapping the multi-dimensional function between sensor values and angle output. This paper proposes RBFNN for approximating the multi-dimensional function between sensor values and angle output.

Artificial Neural Networks (ANN) are computing systems inspired by biological neural networks. ANNs learn by examples and they don't generally need task specific programming. An ANN consists of connected units called artificial neurons, which are simplified version of biological neuron. A neuron consists of one or several inputs and outputs. A neuron applies some activation function on the linear combination of its inputs. A neuron is said to "fire" when linear combination of its inputs exceeds some threshold.

A single neuron can act as a linear classifier and is called perceptron. Perceptron does a linear classification based on a simple learning algorithm that keeps on adjusting the weights iteratively for each example. Perceptron learns by example. This is a major breakthrough that lead to possibility of learning by example for artificial machines.

A single perceptron is not of much practical importance since it cannot solve complex problems with nonlinear classification boundaries. Then focus shifted on ANN architectures with network of neurons. Multi layered ANN came into picture. Sometimes these are also termed as Multi layered Perceptron's (MLPs). But the training methodology was not known for long time for layered ANNs. Backpropagation algorithm for training the multilayer ANN was discovered and it lead to major breakthrough. Since then the usage of ANN grew to large extent and it opened the doors to possibility of solving practical problems such as digit recognition, image recognition, feature recognition, driver less vehicles etc.

Radial Basis Function Neural Network (RBFNN) is a particular type of ANN with one layer of input neurons, one hidden layer consisting of neurons with RBF activation function and one output layer that simply performs linear combination of outputs from hidden layer. RBFNN is applied for tasks such as classification of clusters, function approximation etc. In this paper, RBFNN are used for function approximation.

The RBFNN are so called because the activation function used is a function of radial distance i.e. Euclidian distance between input vector and Centre of the RBF neuron.

RBFNN is used to approximate the multi dimensional non linear function between vector of sensor values and angle output. All the parameters of RBFNN needs to be determined to successfully approximate the multi dimensional function between inputs and output. In order to train the network, data needs to be obtained for the vector of sensor values and angle output at various angular positions of the rotating magnet. This data is obtained physically using the magnetic encoder device designed and developed as suggested in methodology section in this paper.

The trained RBNN is implemented using ATmega32A microcontroller. The forward pass operation of the trained RBFNN is implemented using ATmega32A microcontroller to output the angle value given a vector of sensor values. The magnetic encoder device developed using such microcontroller is used to obtain the actual results of the angle output for various angular positions of the rotating magnet. The results section discusses this elaborately.

II . RESEARCH METHODOLOGY

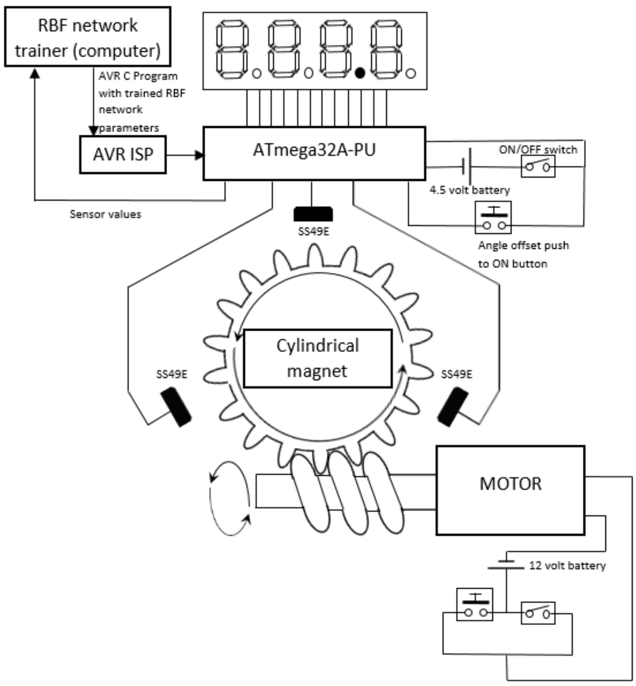

fig 1:magnetic encoder architecture

Above figure shows the magnetic encoder architecture. It consists of multiple linear Hall-effect sensors placed around a rotating cylindrical magnet. The magnet is rotated using 10 RPM motor. In order to reduce further the RPM of the magnet, a worm gear on the motor side and a spur gear on the magnet side is used. The output from the linear Hall-effect sensors varies as the magnet rotates. Minimum 3 sensors are required to uniquely identify the angular position of the magnet. The angular position of the magnet can be determined based on the output signals of the linear Hall-effect sensors. RBF neural network is used to map the multi-dimensional function between sensor inputs and the angle output. The parameters of the RBF network need to be determined so as to output the angle value given the sensor inputs. This determination of the RBF network parameters is done by RBF neural network trainer program in a PC. The data for sensor inputs and corresponding angular position of the magnet at different angle positions from 0 to 360 degrees needs to be sent to PC to determine the parameters of RBF network. The trained RBF neural network can then be used to output the angle value. RBF neural network operation to output angle value is implemented using AVR C language in ATmega32A-PU microcontroller. The angle output value is displayed on 4 digit multiplexed 7 segment display.

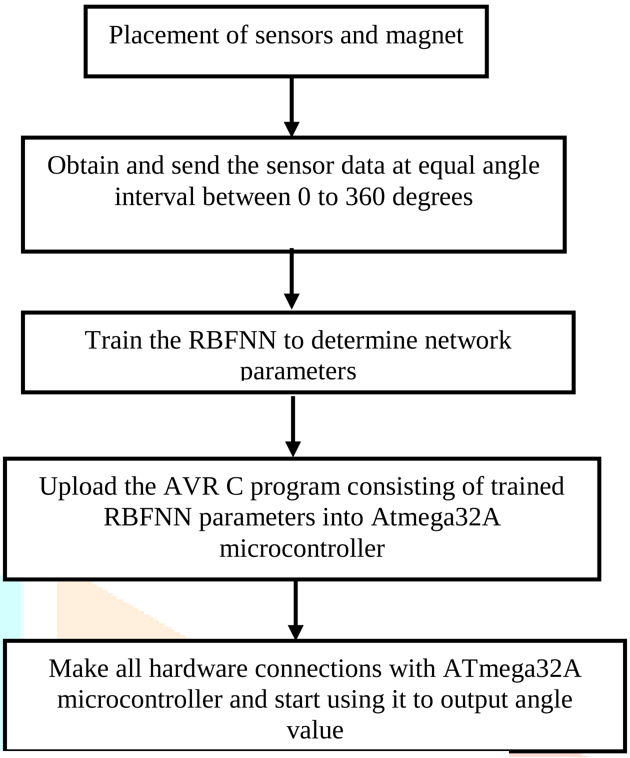

fig 2:Magnetic encoder methodology

The primary aim of proper positioning of sensors and magnet is to ensure unique vector values of multiple sensors at all possible relative positions of sensors and magnet. If the vector of sensor values is not unique at different angular positions, then, mathematically, it becomes difficult to obtain angle output.

Sensor data is collected at equal angle interval for one cycle of rotation. Smaller the angle interval, greater the accuracy. Using this data for vector of sensor values and corresponding angle value, the RBFNN is trained. Following section explains the details of training RBFNN.

2.1 RBF neural network parameters determination

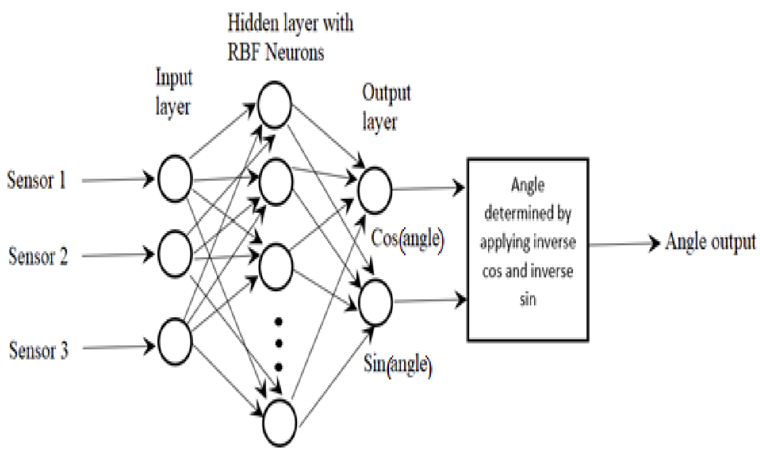

fig 3:magnetic encoder RBF network architecture

Above figure illustrates architecture of RBFNN. Since 3 linear Hall-effect sensors are used, 3 number of input neurons are chosen for the RBF network. All the input neurons are connected fully to the hidden layer neurons. There are no weights associated with connections between input layer neurons and hidden layer neurons.

Two output neurons for outputting cos(angle) and sin(angle) are chosen. The single output neuron for outputting directly the angle value is not done. This is because, with such design, the angle output obtained won't be accurate, there will be a sudden jump from 359 degrees to 0 degrees. The input vector of sensor values is cyclical in nature and hence output also needs to be cyclical in nature. For this reason, two cyclical outputs for cos and sin are chosen. After obtaining the output for cos and sin values from the network, inverse cos and inverse sin needs to be applied to obtain angle value.

The number of hidden layer neurons is a design parameter. The neurons in hidden layer are called RBF neurons. Each RBF neuron has two parameters associated with it: centre, radius

The dimension of centre point of RBF neuron is equal to the number of neurons in input layer, 3 in this case. All the RBF neurons are fully connected with output neurons. All these connections have weight associated with them. The output layer simply does a linear combination of its inputs.

Gaussian function is used as activation function for RBF neurons. The Gaussian activation function is expressed mathematically as follows:

$$\phi = e^{\left(-\beta\frac{(\bar{X}-\bar{C})^2}{r^2}\right)}$$

where $\bar{C}$ is the centre of the RBF neuron
r is radius of RBF neuron
β is parameter for controlling shape of the activation function
$\bar{X}$ is the input vector

Each RBF neuron has its own centre, radius values. $\beta = 1$ is chosen for all the RBF neurons. One set of weights are associated with output neuron for cos(angle), second set of weights are associated with output neuron for sin(angle).

The values for weights associated with both the output neurons needs to be determined by training the RBFNN by supplying the values for 3 sensor inputs and corresponding values for cos(angle) and sin(angle).

The training methodology is as follows:

There are two sets of parameters associated with RBFNN as follows:

1. Centre, radius value associated with each RBF neuron in hidden layer
2. Weight value for connection between each hidden layer neuron and output neuron

The first set of parameters are determined based on the sensor curves. And the weights of the RBFNN are determined directly using following formula:

$$\bar{W} = (A^T A)^{-1} A^T \bar{B}$$

Where A is the matrix of outputs from RBF neurons. Each row in this matrix corresponds to the output of all the RBF neurons for one vector of sensors input.

$$A = \begin{bmatrix} \phi11 & \phi12 & \ldots & \phi1n \\ \phi21 & \phi22 & \ldots & \phi2n \\ \ldots & \ldots & \ldots & \ldots \\ \phi m1 & \phi m2 & \ldots & \phi mn \end{bmatrix}$$

Where ϕij indicates output from jth RBF neuron for the ith vector of input.
m = number of input vectors
n = number of RBF neurons

W is the matrix of weights for connections between RBF neurons and output neurons. Each column in this matrix represents weights for all the connections between one output neuron and all the RBF neurons in hidden layer.

$$W = \begin{bmatrix} w11 & w12 \\ w21 & w22 \\ \ldots & \ldots \\ wn1 & wn2 \end{bmatrix}$$

B is the matrix of output from RBF network. Each row in this matrix corresponds to the output from all the output neurons for one input vector.

$$B = \begin{bmatrix} cos(\theta1) & sin(\theta1) \\ cos(\theta2) & sin(\theta2) \\ \ldots & \ldots \\ cos(\theta m) & sin(\theta m) \end{bmatrix}$$

During training phase, all the elements of the matrix A and B are known and the objective is to determine the weights i.e. elements in W matrix.

After the training is successful, RBFNN can be used to output cos(angle) and sin(angle) for given vector of sensor input. The forward pass operation of the trained RBFNN is used to output given an input vector of sensor values. This can be expressed in matrix form as follows:

A W = B

During usage of RBFNN, W i.e. matrix of weights is known. Matrix A is obtained by applying Gaussian function for the input vector for all the RBF neurons. B is the result i.e. output of the RBFNN forward pass operation.

III. RESULTS AND DISCUSSION

The hardware for absolute magnetic encoder based on RBFNN is developed as mentioned in methodology section. This device is used to obtain various results as mentioned in following sections.

3.1 Actual sensor curves

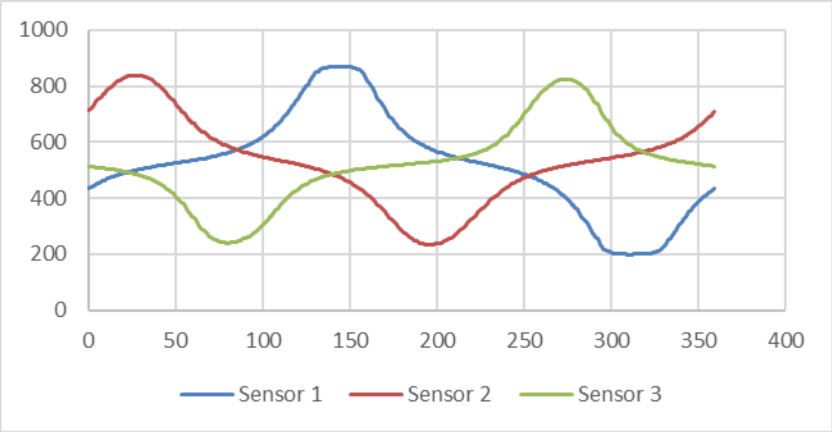

fig 4:sensor output curves

Above figure shows actual sensor values for one cyclic rotation of the rotary shaft. It is evident from above figure that there are no overlapping saturation regions of sensors. Also, the vector of sensor values at all positions in a cycle are unique. This ensures possibility of obtaining angle output value with high accuracy.

3.2 Number of RBF neurons

Error! Reference source not found. shows the curve for accuracy vs number of RBF neurons. Accuracy increases with more number of RBF neurons but after 36 number of RBF neurons, there is not much improvement in accuracy. Hence the number of RBF neurons are chosen as 36 for the implementation of magnetic encoder.

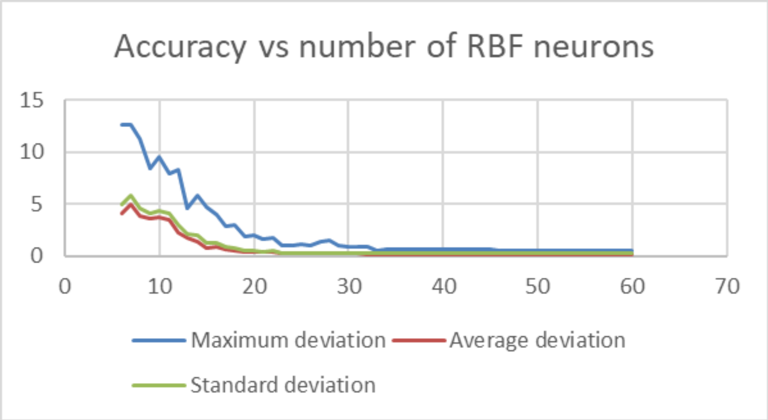

fig 5:number of neurons versus accuracy

3.3 Center of RBF neurons

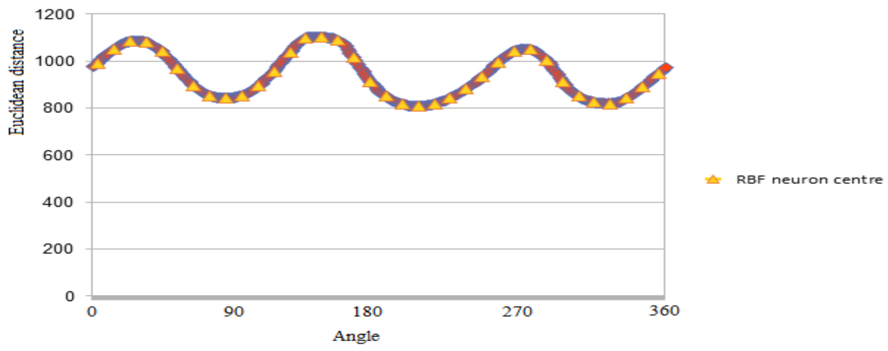

fig 6:center of RBF neurons

Center of RBF neurons are chosen at equal angle interval of near about 10 degrees. Above figure shows the Euclidean distance of vector for sensor values for one cycle. The small triangles shows the positions of 36 RBF neurons.

3.4 Size of training data

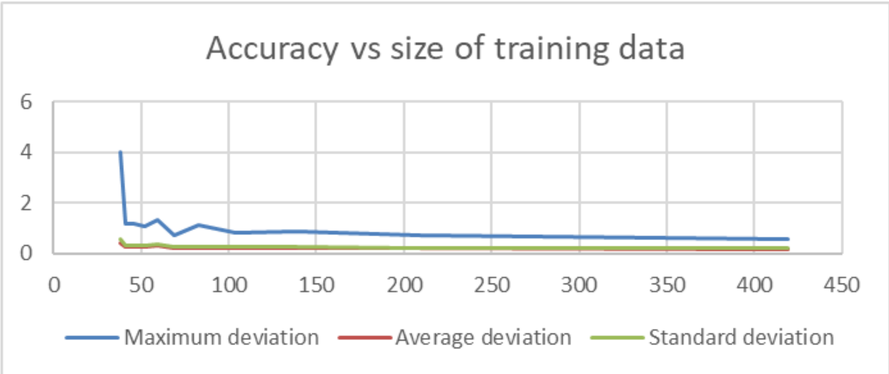

fig 7: accuracy versus training data size

One of the main advantages of RBFNN (or any ANN in general) is that the accuracy of the output increases with more number of training data. Other things remaining same, i.e. the various parameters of the RBFNN, the training methodology also remaining same, if we supply more number of training data, then the accuracy of the output of RBFNN increases. Above curve illustrates this fact.

3.5 Angle output

Following figure shows the curve for expected angle and actual angle output obtained by trained RBF network with above mentioned parameters:

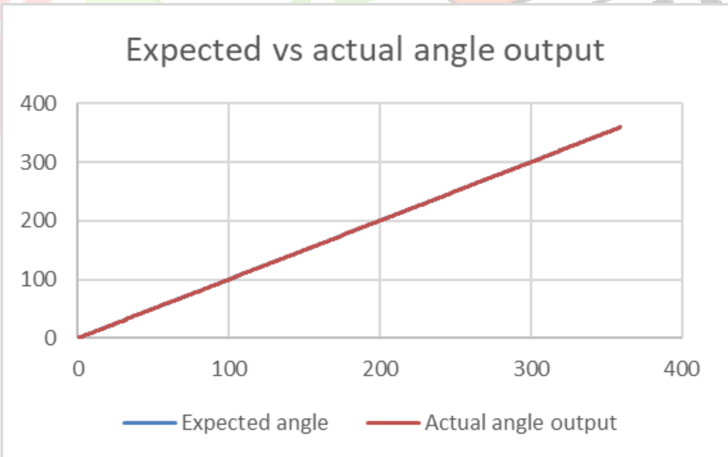

fig 8: expected versus actual angle output

The expected and actual angle output are almost overlapping in above figure. The various measures of error are as follows:

Maximum deviation	=	0.564551
Average deviation	=	0.181372
Standard deviation	=	0.217934

REFERENCES

[1] H. Lee, B. Lee, K. Park, and R. Elmasri, “Fusion Techniques for Reliable Information: A Survey,” International Journal of Digital Content Technology and its Applications, vol. 4, no. 2, pp. 74–88, 2010.

[2] H. T. Le, H. V. Hoang, and J. W. Jeon, “Efficient method for correction and interpolation signal of magnetic encoders,” in Industrial Informatics, 2008. 6th IEEE International Conference on, 2008, pp. 1383–1388.

[3] Arthur Noronha Montanari and Everthon de Souza Oliveira, "A Novel Analog Multisensor Design Based on Fuzzy Logic: A Magnetic Encoder Application", IEEE Sensors Journal, DOI 10.1109/JSEN.2017.2752959.

[4] J.-S. Jang and Chuen-Tsai Sun, “Neuro-fuzzy modeling and control,” Proceedings of the IEEE, vol. 83, no. 3, pp. 378–406, 1995.

[5] Hongqin Xie, Kai Yang, Dadai Lu, “Position control of PM Synchronous motors used for sewing machines based on high accuracy magnetic encoder”.

[6] Jae Wan Park, Ha Xuan Nguyen, Thuong Ngoc-Cong Tran and Jae Wook Jeo, “Improve efficiency multi-turn magnetic encoder that uses Gear system,” 2017, 17th International Conference on Control, Automation and Systems (ICCAS 2017) Oct. 18–21, 2017, Ramada Plaza, Jeju, Korea.

[7] Stephen C. Stubberud, and Kathleen A. Kramer, “Data Association for Multiple Sensor Types Using Fuzzy Logic” IEEE Transactions on Instrumentation and measurements, vol. 55, no. 6, December 2006.

[8] Xiaoguang Wang, Haitao Wang, Hongqin Xie, Dezhang Lou, Kai Yang “Design of Magnetic Encoder Based on Reconstructing and Mapping Looking-up Table ” IEEE Transaction.