Using ANN To Predict The Best HUB Location

A. B. Aljunaid and I. AbuElMaaly
Electrical Engineering Department
University of Khartoum
Khartoum, Sudan

A. Sagahyroon
Computer Engineering Department
American University
Sharjah, UAE

Abstract - **The power and usefulness of artificial neural networks have been demonstrated in several applications including signal processing, speech recognition and robotic control. In this paper, we present the results of an attempt to assess the feasibility of using artificial neural networks in predicting the best Hub location in a STAR topology local area network. The idea is to train the network to predict the position that would produce the most economical wiring or cabling plan, that is, the coordinates produced by the trained network are a reflection of the shortest path possible between the nodes in the network given the selected position of the Hub.**

I. Introduction

The star local area network (LAN) topology connects workstations, computers, printers, servers and other elements at one central location called a cross-connect or hub. The creators of a STAR LAN have several design considerations that include the determination of the cabling medium, wiring plan and cost ,determination of the maximum distances for cable runs, as well as determining the best hub location that will achieve the most economical wiring layout possible. In a STAR topology, the purpose of a hub is to regenerate and retime network signals. This is done at the bit level to a large number of hosts (e.g. 4, 8, or even 24) using a process known as concentration.

Artificial Neural Networks (ANN) are powerful tools that use a machine learning approach to numerically solve relationships between inputs and outputs. The use of neural networks has increased substantially over the last several years because of the advances in computing performance and of the increased availability of powerful and flexible neural net software. ANN has been applied with promising results to address a variety of problems in the electrical and computer engineering fields [1-4].

Neural networks attempt to simulate the functioning of the human brain by constructing a highly parallel computational structure. Much like their biological counterparts, neural networks are composed of many processing units known as *neurons* that are interconnected to form a parallel processing system. Most neural networks operate by processing some type of input signal to produce some form of an output response. The processing elements are usually grouped together into a layered structure known as a slab, or layer where each processing element on each layer performs an analog integration of its inputs to determine its activation value [5]. Processing begins with the entire network in a quiescent state. An external stimulus (a set of input signals to be processed by the network) is applied to the input layer, where each signal stimulates one of the processing elements on the input layer. Each processing element on the input layer generates a single output signal, with a magnitude that is a function of the total stimulation received by the unit. Collectively the outputs produced by all of the processing elements on the layer are then passed on as an input pattern to the subsequent layer of processing elements. This process is repeated until the final layer' produces an output for the current input stimuli. The fundamental goal of employing a neural network is to produce some type of predicted solution to a problem. Most network configurations achieve a prediction after the successful completion of two tasks. First, the network must be taught how to solve the particular problem. This step is usually accomplished by providing a series of example problems known as *training* vectors where the correct output response is known in advance. The network use stage then tells the network to solve the problem given a new set of input parameters. The most commonly implemented learning model for training neural networks uses *supervised learning* techniques. In supervised learning, the network is trained on a group of training patterns. For each input vector presented to the network, there is a corresponding output vector. This output vector mimics the function of a *teacher* to provide some validation of the learning process. During the training process, the teacher makes a comparison between the network output vector and the correct target vector. Any variation in the two vectors indicates the error in the network prediction to the given problem. The most widespread implementation of supervised training employs *error backpropagation* techniques [6]. In error backpropagation training, the difference between the network output and the correct output is used to adjust the connection weights. The method sends the error in output values backwards through the network to adjust the weights in a systematic manner. This

E-87

1-4244-0387-1/06/$20.00 ©2006 IEEE

method has successfully been applied to many complex problems.

The primary goal of the work discussed in this paper is to assess the feasibility of using ANN in predicting the best hub (24-ports) location out of 25 possible positions which represent the nodes connected to the STAR LAN as well as the Hub. That is, the ANN should predict the best position that will provide the shortest path length and hence the shortest and most economical cabling requirements. To complete this task, we used a variety of tools. AutoCAD was used to generate the spatial data (i.e. coordinates for 25 possible positions); the Arc/Info 3.5.1 GIS package was used to prepare all possible paths within a LAN. The shortest path within a LAN was computed using Excel utilities. To generate a well-trained neural network model, 355 different patterns were used for training and testing. The neural network package used is NeuroShell2 [7].

II. Methodology

The test data of this research work have been created and designed using AutoCAD activities, where the relative coordinates of terminals and Hub have been digitized. A scaling factor was applied to these data for producing new patterns of LAN's coordinates. . A total of 355 patterns of these data were used to train and test the Neural Network model, while an additional ten patterns were used to evaluate the potentiality of the Artificial Neural Network production. Table I includes a sample of two generated test patterns and the computed best Hub position.

The planimetric position of the terminals and Hub of each LAN has been digitized by using AutoCAD drafting utilizes, the digitized data has been exported as drawing exchange format (DXF) which could be imported and handled within GIS environment, where topologies has been constructed and built. The production of the later process is GIS coverage (special data) combined with their corresponding non-spatial database. The Arc/Info 3.5.1 Network model was used to compute the shortest distance between each terminal pair. The process was very difficult and time consuming, where the distance between each terminal and other ones should be computed for the whole Network, in addition to that the whole process should be repeated for other test LANs. This is the main reason stand against recommending the GIS as a solution for extracting the best Hub location of a LAN. Nevertheless GIS was found to be very efficient and helpful tool for preparing test data patterns, which will be processed later within Neural Network environment. Excel was used to compute the total cable length for all possible connection paths within a LAN, based on the results generated by the GIS activity. Both training and test Network patterns have been designed as excel spreadsheet, which could be handled and processed by the Neural package.

A. Building the Neural Network Model

Three Neural Network paradigms have been tested for predicting the best Hub location, which offers the shortest path, and therefore the least cable length of a LAN. These are Back Propagation (BPN), Probabilistic (PNN) and General Regression (GRNN) Neural Network Paradigms. Tables II, III and IV contain the results produced using the three techniques. The GRNN produces the most efficient results, while BPN and PNN paradigms failed to produce sufficient results, therefore the GRNN was selected.

A GRNN model was designed with one input layer, one hidden layer and an output layer. The input and output layers were designed to have 50 and 2 neurons (processing elements) respectively. A total of 70 neurons have been assigned to the hidden layer. The linear method was selected as an activation function for the input layer neurons. However, other activation methods are available, but the selected one is recommended by NeuroShell2 creators. The generitic adaptive calibration method has been selected as a calibration method for testing the Network. The Network was trained until 20 of learning epochs has been processed with no improvement in the smoothing factor greater than 1% of the current smoothing factor.

III. Results

To evaluate the performance of the GRNN network, the trained system was tested using 10 production patterns representing 10 different LANs. Table IV shows the results obtained using the GRNN network. Each pattern contains 25 x, y coordinates. The output is the best (predicted) x, y coordinated for the best Hub location. In Table V, the computed data and the net output data represent the calculated or desired coordinates and the coordinates produced by the trained network. The standard deviations Dx and Dy were computed for both coordinates. The root mean square error of the resultant residuals was computed by calculating the residual values in both X and Y directions. The statistical analysis is used to estimate the accuracy of the predicted values and assess the potentiality of the network.

A comparison of the computed data with the Net data illustrates that the Network was well trained. The statistical analysis resulted in a standard deviation of ± 0.67 implying that , the error in a given position is in the range of ± 0.67 meters. A mean value of 0.33 shows that Network predictions are consistent across all production patterns.

IV. Conclusion

An ANN was used in this work to assist in predicting the optimum Hub position that provides the shortest path length for a STAR LAN that consist of 25 nodes. The Network succeeded in this task within a range of ± 0.67 planmetric error. The presented work produced promising results however further investigation of ANN Network using test data from real world LAN design scenarios is useful. The results can be generalized and the same concept be applied to any problem or application that requires the selection of optimum coordinates or optimum positioning of an object.

References

[1] Shertukde.bH. M, "Efficient call-connection model for multipoint connections in an ATM/B-ISDN network using fuzzy neural networks"International IEEE/IAS Conference on Industrial Automation and Control: Emerging Technologies, 1995

[2] Liang Chen, "Pattern classification by assembling small neural networks: IEEE International Joint Conference on Neural Networks, 2005.

[3] Youshen Xia; Jun Wang, "A general projection neural network for solving optimization and related problems",Proceedings of the International Joint Conference on Neural Networks, 2003.

[4] Joon-Ho Lee, In-Sik Choia and Hyo-Tae Kim: "Natural frequency-based neural network approach to radar target recognition" Signal Processing, IEEE Transactions on Volume 51, Issue 12, Dec. 2003 Page(s):3191 – 3197

[5] D. Scaula, "Building neural networks", John Wiley & Sons, 1999.

[6] Rumelhart, D., Hinton, G., and Williams, R. "Learning Internal Representations by Error Propagation" Parallel Distributed Processing, Cambridge, MA: MIT Press, 1986.

[7] http://www.wardsystems.com/

Terminals	LAN1		LAN2		LAN3	
	X	Y	X	Y	X	Y
1	1	11.8	4	23.7	0.5	11.75
2	3	11.8	8	23.7	2	11.75
3	7.5	11.8	20	23.7	3.5	11.75
4	9	11.8	24	23.7	5	11.75
5	10.5	11.8	2.3	20	6.5	11.75
6	12	11.8	4	16.3	11	11.75
7	13.5	11.8	8	16.3	13	11.75
8	0.2	10	20	16.3	0.25	10
9	13.8	10	24	16.3	13.75	10
10	1	8.2	1	7.7	0.5	8.25
11	3	8.2	4	7.7	2	8.25
12	7.5	8.2	7	7.7	3.5	8.25
13	9	8.2	10	7.7	5	8.25
14	10.5	8.2	13	7.7	6.5	8.25
15	12	8.2	22	7.7	11	8.25
16	13.5	8.2	26	7.7	13	8.25
17	2	3.8	0.3	4	2	3.75
18	4	3.8	27.7	4	4	3.75
19	10	3.8	1	0.3	10	3.75
20	12	3.8	4	0.3	12	3.75
21	12.8	2	7	0.3	1.25	2
22	2	0.2	10	0.3	2	0.25
23	4	0.2	13	0.3	4	0.25
24	10	0.2	22	0.3	10	0.25
25	12	0.2	26	0.3	12	0.25
Best Hub Position	7.5	8.2	13	7.7	6.5	8.25

Desired Data		**Net Output**		Dx	Dy	Dp
X	Y	X	Y			
7.5	8.2	8.0924	8.3313	-0.5924	-0.1313	0.606776
13	7.7	19.4601	9.9011	-6.4601	-0.2011	6.824788
6.5	8.25	6.3185	8.4124	-0.1815	-0.1624	0.243549
1	0.25	2.9647	0.2985	-1.9647	-0.0485	1.965299
4	4.9	4.0431	4.4329	-0.0431	-0.4671	0.469084
6	4.7	6.1668	4.6989	-0.1668	-0.0011	0,166804
20.2	13	20.2148	13.0298	-0.0148	-0.0298	0.033273
9.75	9	9.7837	9.1014	-0.0337	-0.1014	0.106853
15	24.8	14.9335	24.8517	-0.0665	-0.0517	0.084233
12.5	8.95	13.2895	10.9813	-0.7895	-2.0313	2.179332
Mean				-0.98171	-0.42893	1.267999
Std.				2.027534	0.907328	2.103077

E- 89

Desired Data		Net Output		Dx	Dy	Dp
X	Y	X	Y			
7.5	8.2	0.40186	0.59814	7.09814	7.60186	10.40057
13	7.7	0.566814	0.433186	12.43319	7.266814	14.40107
6.5	8.25	0.381257	0.618743	6.118743	7.631258	9.781365
1	0.25	0.999674	3.26E-04	0.000326	0.249674	0.249674
4	4.9	0.386276	0.613724	3.613724	4.286276	5.60635
6	4.7	0.4939	0.5061	5.5061	4.1939	6.921411
20.2	13	0.544821	0.455179	19.65518	12.54482	23.31735
9.75	9	0.453445	0.546555	9.296555	8.453445	12.5643
15	24.8	0.314811	0.685189	14.68519	24.11481	28.23436
12.5	8.95	0.518414	0.481586	11.98159	8.468414	14.67217
Mean				9.038873	8.481127	12.61496
Std.				5.780579	6.391231	8.254975

Desired Data		Net Output		Dx	Dy	Dp
X	Y	X	Y			
7.5	8.2	6.502007	8.244781	0.997993	-0.04478	0.998997
13	7.7	13.00016	7.700232	-0.00016	-0.00023	0.00028
6.5	8.25	6.500045	8.249961	-4.5E-05	3.91E-05	5.95E-05
1	0.25	3.018964	0.295609	-2.01896	-0.04561	2.019479
4	4.9	3.672129	4.704621	0.127871	0.195379	0.233504
6	4.7	5.999068	4.701198	0.000932	-0.0012	0.001517
20.2	13	20.0	13	4.96E-06	-4.8E-06	6.88E-06
9.75	9	9.749969	8.999982	3.15E-05	1.81E-05	3.63E-05
15	24.8	15	24.8	0	-1.1E-06	1.14E-06
12.5	8.95	12.5	8.950001	9.54E-07	-7.6E-07	1.22E-06
Mean				-0.08923	0.010361	0.325388
Std.				0.746021	0.0676648	0.672643

Desired Data		Net Output		Dx	Dy	Dp
X	Y	X	Y			
7.5	8.2	6.4978	8.2446	1.0022	-0.0446	1.003192
13	7.7	12.9999	7.7	1E-04	0	1E-04
6.5	8.25	6.5001	8.2499	-1E-04	1E-04	0.000141