# OPTIMIZED INTERCONNECTIONS IN PROBABILISTIC SELF-ORGANIZING LEARNING

J.A.Starzyk, Mingwei Ding, Haibo He
School of Electrical Engineering and Computer Science
Ohio University
Athens, OH, 45701, USA

**ABSTRACT**
This paper focuses on self-organization of a multi-layered feed forward artificial neural network structure. Both the selection of interconnections among neurons and their optimum weights are studied. In this learning structure, the neurons are sparsely connected and dynamically adjust their connectivity structure. Only the feed-forward propagation is used and each neuron dynamically adjusts its threshold based on the incoming data. By analogy to the signal weighting, this paper derived how to set the optimal interconnection weights for neuron's inputs. The binary input weight selection, suitable for hardware implementation, is discussed. Comparison between the binary and optimal weighting scheme is presented. Simulation examples for financial data analysis and power quality disturbance classification problems show the effectiveness of the proposed scheme.

**KEY WORDS**
Probabilistic Neural Network, Self-organizing Learning, Optimal Weight, Input Selection Strategy, Dual Neural Network, Financial Data Analysis, Power Quality Classification

## 1. Introduction

The probabilistic neural networks (PNNs), introduced in [1], have attracted an increasing attention in the area of machining learning. It was reported that the PNNs have the advantage of being able to learn additional information from the new data and do not require access to the original data [2]. Recent study further demonstrated that the PNNs also are able to accommodate new classes that may be introduced with new data [3]. The above advantages of PNNs made them a popular choice for solving pattern recognition problems as reported in recent literature. Paper [4] proposed a network structure determination algorithm for pattern classification. The proposed iterative algorithm contains two parts where the first part is used to identify an appropriate smoothing parameter, and the second part is used to determine a suitable pattern of the layer neurons using an orthogonal forward regression algorithm. In paper [5], a novel temporal updating approach was developed for PNN classifiers that can be used to track temporal changes in a sequence of images. This is done by utilizing the temporal contextual information and adjusting the PNN to adapt to such changes. Simulation results from satellite cloud imagery data show the effectiveness of the proposed scheme. Paper [6] proposes a new constructive probabilistic neural network (CPNN) for freeway incident detection and concluded that the CPNN is an efficient adaptive classifier for incident detection problems in changing site traffic environment. Some other reported applications of PNN include analog fault detection and classification problems [7] and the estimation of the risk of mortality after cardiac surgery [8].

In this paper, we propose a new online, probability based, dynamically reconfigurable (i.e. during runtime), data-driven, self-organized learning array. In this learning array, the processing units (neurons) are sparsely connected and dynamically adjust their connectivity structure. Each neuron can implement various arithmetic and logic functions. The neurons can self-reconfigure and use local interconnect for maximum performance. Only the feed-forward propagation is considered where, based on the incoming data, each neuron dynamically adjusts its threshold and connectivity structure in an array of evolvable signal processing blocks. Detailed discussion of the optimal weight selection for inputs of each neuron and the selection of the best connectivity structure is presented. Example applications for this kind of self-organizing learning are given.

This paper is organized as follows: Section 2 discusses the structure of the probability based self-organizing learning array. Section 3 discuses the optimal weight and fixed weight input selection. Comparison of these two schemes is presented. Section 4 gives the simulation results. Two application problems, named financial data analysis and power quality disturbances classification are investigated to show the effectiveness of the proposed method. Finally, a conclusion ion is given in Section 5.

## 2. Self-organizing neural network structure

Self-organization is important in artificial neural networks (ANNs) and machine learning. A self-organizing learning algorithm (SOLAR) that combines neural networks and information theory was presented in

[9-11]. This entropy-based learning algorithm was simulated on benchmark data and proved to be superior over many existing machine learning algorithms [10], while dealing with noisy or incomplete data. Based on this algorithm, we proposed SOLAR system which is different from classical ANNs in the way it is organized and how it learns. SOLAR self-organizes its hardware to perform classification and recognition tasks. SOLAR has a fixed array of elemental processing units acting as single neurons, and programmable interconnections between them. Initially, SOLAR neurons are randomly connected to previously generated neurons. They learn adaptively using both primary inputs and inputs from other neurons. Controlled by signals from other neurons, they perform basic transformations of their input signals. A neuron parameters and connections are dynamically reconfigured as a result of training, and effectively, SOLAR's structure self-organizes establishing its final wiring and neurons' functionality.

SOLAR uses two types of neurons – feature neurons and merging neurons. Feature neuron perform transformation of the feature signals and output transformed features as well as probability based voting results, while merging neurons merge voting results into final classification decision.

This section focuses on the self-organization of a multi-layered feed forward network structure used in SOLAR. After a nonlinear transformation of the feature signals, individual neurons' correct recognition rates are estimated based on the training sample probabilities. These probabilities are used in the merging neurons to arrive at the classification decision. Merging neurons perform classification based on voting results from feature neurons or outputs of other merging neurons.

Denote the $j$ th merging neuron at $i$ th layer as $n_{ij}$. To optimize learning network performance, each merging neuron will actively search for the most useful way to connect to the feature neurons and other merging neurons outputs. Due to local and sparse connectivity structure of SOLAR array, a neuron $n_{ij}$ can only receive its inputs from neurons included in its *selection set* $R_{ij}$. Each neuron $n_{ij}$ will select $C_{ij}$ neurons from $R_{ij}$ as its actual *input set*, where $C_{ij} \subset R_{ij}$. When neuron $n_{ij}$ receives all the information from its input set, a voting scheme is necessary to make correct classification based on this information. For two neighboring neurons in the same layer $n_{ij}$ and $n_{ij+1}$, their selection sets $R_{ij}$ and $R_{ij+1}$ will overlap. In this way, the neurons in $i$ th layer will get different information from previous layer with some degree of redundancy. The size of neuron's *selection set* and the amount of their overlap depend on the network size.

Two strategies can be followed for the initial input set selection.

1. Random selection: Randomly select a connection from a neuron $\hat{n}$, where $\hat{n} \in R_{ij}$ and $\hat{n} \notin C_{ij}$, and check if $P(\hat{n}) > \min_{n_x \in C_{ij}} (P(n_x))$ (where $P(\hat{n})$ is the recognition rate of the neuron $\hat{n}$, $P(n_x)$ is the recognition rate of $n_x$). If yes, $\hat{n}$ is selected to substitute neuron $n_x$ with the minimum $P(n_x)$, otherwise keep $C_{ij}$ unchanged.

2. Greedy selection: Always choose connections from $C_{ij}$ neurons which have the highest recognition rate within $R_{ij}$.

These strategies result in various learning rates and classification qualities of the SOLAR array.

In general, the neurons with low correct recognition rates don't contribute significantly to the final classification results and the connection to these neurons may be broken without effective change in the classification quality. This is illustrated in Fig. 1. Here, a merging neuron is connected to 40 neurons from previous layers, each with different correct classification probabilities (doted line). By combining inputs from these neurons using random selection (dashed line), and in descending order (solid line) the merging neuron improves its recognition performance. As we can see, combining inputs in descending order improves the classification results sooner than in the random order, and may lead to elimination of unnecessary interconnections between neurons. Next we will discuss how to combine the neural inputs using an optimum weights and binary weights.

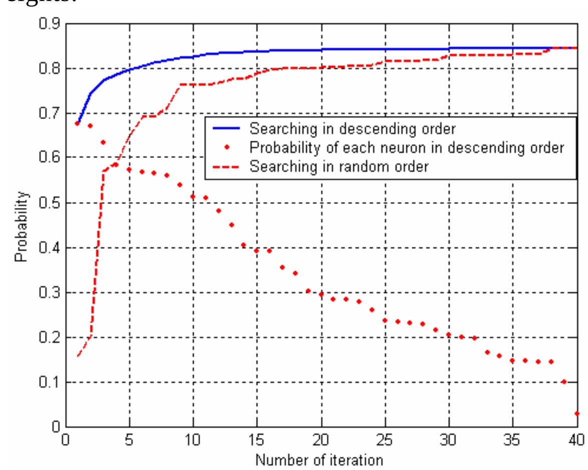


Fig. 1 Correct classification probability increases by merging other neurons inputs.

## 3. Optimal and binary weight inputs

In this section, we first describe how to select the optimum interconnection weights set by analogy to signal processing. This will be followed by discussion of the binary input weight selection, which is more suitable for hardware implementation.

For an $n$ input system with input signal $s_i$ multiplied by a nonnegative weight $w_i$, we have the combined signal energy

$$\hat{s}^2 = (w_1 s_1 + w_2 s_2 + \ldots + w_n s_n)^2 \qquad (1)$$

Without loss of generality we can assume that all the noise signals $n_0$ are the same and have noise energy equal to one. So the combined signal noise has energy

$$\hat{n}^2 = n_0^2 (w_1^2 + w_2^2 + \ldots + w_n^2) \qquad (2)$$

In addition we normalize weights to have

$$w_1 + w_2 + \ldots + w_n = 1 \qquad (3)$$

Our objective is to find the set of weights that maximizes the combined signal to noise rate or

$$\max_{w_i} (\hat{s}^2 / \hat{n}^2) = F(w_i) \qquad (4)$$

To get the solution for equation (4), we take the gradient of $F(w_i)$

$$\nabla F_{w_i} = \frac{2\hat{s}\frac{\partial \hat{s}}{\partial w_i}\hat{n}^2 - 2\hat{s}^2\hat{n}\frac{\partial \hat{n}}{\partial w_i}}{\hat{n}^4}$$

$$= \frac{2\sum_{k=1}^{n} w_k s_k \cdot s_i \cdot \left(\sum_{k=1}^{n} w_k^2\right) - 2w_i \cdot \left(\sum_{k=1}^{n} w_k s_k\right)^2}{\hat{n}^4} \qquad (5)$$

In order to find the maximum value of $F(w_i)$, we must satisfy

$$\nabla F_{w_i} = 0 \qquad \text{for } i=1,\ldots,n \qquad (6)$$

so we must have

$$s_i \left(\sum_{k=1}^{n} w_k s_k\right)\left(\sum_{m=1}^{n} w_m^2\right) = w_i \left(\sum_{k=1}^{n} w_k s_k\right)^2 \qquad (7)$$

and subsequently

$$s_i \left(\sum_{k=1}^{n} w_k s_k\right) = w_i \frac{\left(\sum_{k=1}^{n} w_k s_k\right)^2}{\sum_{m=1}^{n} w_m^2} = w_i \left(\frac{\hat{s}^2}{\hat{n}^2}\right) \qquad (8)$$

Since (8) is satisfied for various signals $s_i (i = 1,2 \ldots n)$, with the identical terms on both sides of all these equations, we must have

$$\frac{w_i}{s_i} = A = const \qquad (9)$$

*A. Optimal weights*

Suppose that a merging neuron combines outputs of n neurons with known classification probabilities. Our objective is to estimate the resulting output probability for each class. We will use the result (9) to derive the optimal weighting for probabilistic self-organized learning. Without loss of generality we will focus on the two class classification problem. Neurons in the SOLAR network have different classification qualities measured by their recognition rate $p$. In a two class problem this probability is based on the ratio of the number of points from majority class over total number of points. Neuron fires with value representing probability that a sample came from class 1. In such case $p = 0.5$ represents the lowest information meaning that out of the two classes each one is equally likely. On the other hand, $p = 1$ or $p = 0$ represents full information, meaning that we are certain about the class type. By analogy to signal and noise, we can represent this with $|p - 0.5|$ being a signal value and $0.5 - |p - 0.5|$ being a noise value.

Introducing a new variable $\overline{p}$ as the signal value

$$\overline{p} = p - 0.5 \qquad (10)$$

we get the signal-to-noise ratio $\hat{p}$, $(\hat{p} \in (-\infty, \infty))$ as

$$\hat{p} = \frac{\overline{p}}{noise} = \frac{\overline{p}}{0.5 - |\overline{p}|} = \frac{1}{\frac{1}{2\overline{p}} - sign(\overline{p})} \qquad (11)$$

where $\overline{p} \in [-0.5, 0.5]$ and $noise \in [0, 0.5]$.

By analogy to combining signals $s_i$ in (1), we may combine $\hat{p}_i$ from different neurons with optimized weights proportional to $|\hat{p}_i|$

$$w_i = \frac{|\hat{p}_i|}{\sum_{k=1}^{n} |\hat{p}_k|} \qquad (12)$$

so, the weighted result $\hat{p}_{out}$ is

$$\hat{p}_{out} = \frac{s}{n} = \frac{\sum_{i=1}^{n} w_i \hat{p}_i}{\sqrt{\sum_{i=1}^{n} w_i^2}} = \frac{\sum_{i=1}^{n} |\hat{p}_i| \hat{p}_i}{\sqrt{\sum_{i=1}^{n} |\hat{p}_i|^2}} \qquad (13)$$

substituting $\hat{p}_{out}$ in to equation (11), and solving it we can get the equivalent $\overline{p_{out}}$

$$\overline{p_{out}} = \frac{1/2}{\frac{1}{\hat{p}_{out}} + sign(\hat{p}_{out})} \qquad (14)$$

from which we can calculate the resulting recognition rate

$$p_{out} = \overline{p_{out}} + 0.5 \qquad (15)$$

$p_{out}$ represents our belief that the classified sample belong to class 1. Table 1 shows six cases of the $p_{out}$ calculation based on the above analysis. Each case contains recognition rates $(p_1, p_2, p_3)$ of three neurons connected to the merging (output) neuron.

Table 1: Calculation of the output recognition rate

| | | | | | | |
|---|---|---|---|---|---|---|
| $p_1$ | 0.1 | 0.6 | 0.6 | 0.1 | 0.1 | 0.2 |
| $p_2$ | 0.1 | 0.6 | 0.6 | 0.9 | 0.2 | 0.2 |
| $p_3$ | 0.6 | 0.6 | 0.8 | 0.4 | 0.1 | 0.2 |
| $\hat{p}_{out}$ | -5.64 | 0.433 | 1.541 | -0.011 | -5.852 | -2.598 |
| $p_{out}$ | 0.0752 | 0.6511 | 0.8032 | 0.4945 | 0.073 | 0.139 |

*B. Binary weights*

The results obtained in (15) are based on the derived optimal weights applied to selected input neurons $C_{i,j}$. However, in practical hardware implementation, a simplified interconnection scheme is always desired. A binary weight does not require multiplication to obtain the combined input signal. A neuron is either wired to a node from its selection set $R_{ij}$ or not. This corresponds to choosing $1/n$ as weights for all the connected inputs ($n$ is the number of connections). Based on (13), we have

$$\frac{\hat{S}^2}{\hat{n}^2} = \frac{\left(\sum_{i\in C_{ij}} \frac{1}{n}\hat{P}_i\right)^2}{\sum_{i\in C_{ij}} \left(\frac{1}{n}\right)^2} = \frac{\left(\sum_{i\in C_{ij}} \hat{P}_i\right)^2}{n} \tag{16}$$

or

$$\frac{\hat{S}}{\hat{n}} = \frac{\left|\sum \hat{P}_i\right|}{\sqrt{n}} \tag{17}$$

We will use (17) to study the effect of adding connections of different signal strength at a neuron's input. Let us denote the stronger connection as $P_{\max}$, and the weaker connection, that is to be added, as $P_{mix}$, assuming that $P_{\max}$ and $P_{mix}$ are both greater then 0.5. Thus the corresponding scaled variables $\hat{P}_{\max}$ and $\hat{P}_{mix}$ are

$$\hat{P}_{\max} = \frac{P_{\max} - 0.5}{1 - P_{\max}},$$

$$\hat{P}_{mix} = \frac{P_{mix} - 0.5}{1 - P_{mix}} \tag{18}$$

We can calculate the recognition rate resulting from combination of the two inputs as $\hat{P}_{comb}$ and then obtain $P_{comb}$, which will give an estimate of the recognition rate if $P_{mix}$ is appended to $P_{\max}$. The effect of combing this new connection with $\hat{p}_{\max}$ is shown in Fig. 2, where $dP = P_{\max} - P_{mix}$, is the difference between the existing stronger connection and the possible new connection, and $P_{gain}$ defined as $P_{comb} - P_{\max}$, indicates the improvement of the final probability due to $P_{mix}$. From Fig. 2 we can see that, the $P_{gain}$ decreases as the $P_{mix}$ becomes smaller.

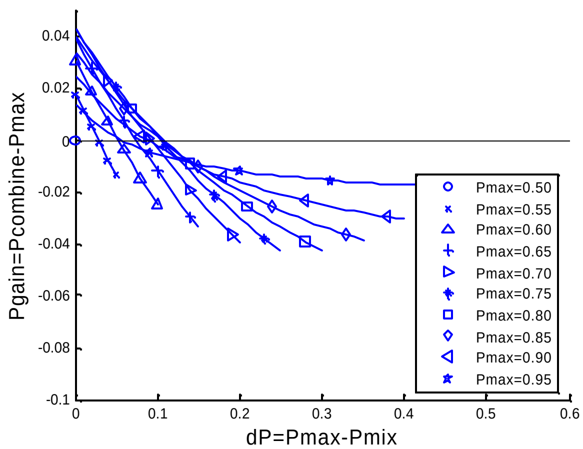


Fig. 2 $P_{gain}$ vs. $dP$ in a binary weighted connection

Based on Fig. 2, we can also obtain the minimum acceptable $P_{mix}$ in order to have a positive $P_{gain}$ as shown in Fig.3. This can be used as a criterion to decide which new connections are acceptable and which are not.

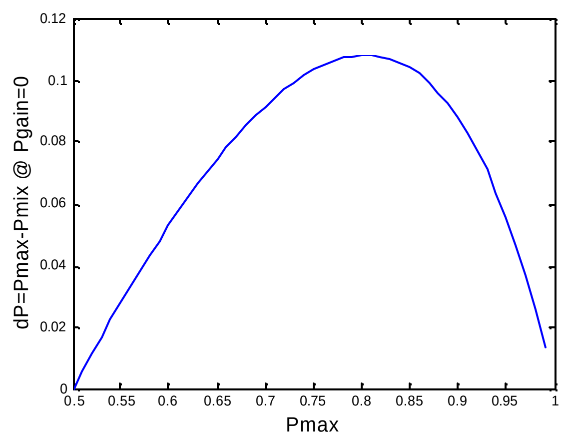


Fig. 3 $dP$ vs. $P_{\max}$ at $P_{gain} = 0$ in a binary weighted connection

As can be seen from Fig. 3, only neurons of a similar recognition rates can be merged to improve the classification performance.

## 4. Simulation results

In this section we illustrate the method with two practical application examples to show the effectiveness of the proposed method. The first one considers financial data analysis and classifies companies based on their future financial performance, and the second example involves power quality disturbance classification.

*Case I: Prediction of financial performance*

Financial analysis based on the proposed method was performed using the Research Insight [12] financial database derived from publicly traded US companies and closed-end funds trading on the NYSE, AMEX, NASDAQ, OTC and Canadian stock exchanges. Financial data reported by these companies over the most recent 20 years include many indicators based on income, balance sheet, and cash flow statements. The training and testing data structure was based on 192 features extracted from data base for 3-year periods. During training two classes of data are defined based on the predicted financial performance measured by the change in the stock price. Companies which would perform below median (stock price increase was less than the median increase) are classified as class 1 and those that perform above the median are classified as class 2. Testing set uses the classification rules developed in the training set and applies it to the next year data. Thus prediction of the financial performance is tested and verified.

Each training data set is based on 3 years period of company's financial information. Testing set is based on different 3-years period of financial information. Fig. 4 shows possible time overlap between training and testing data sets.

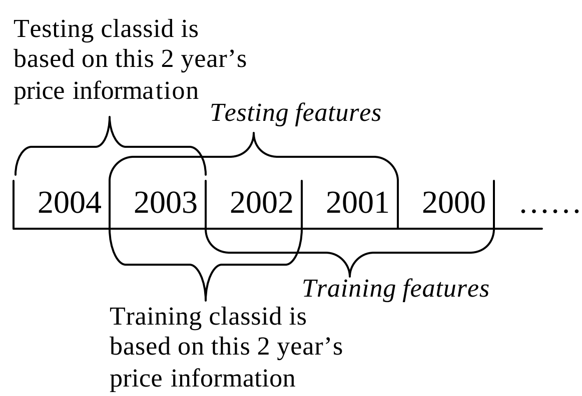


Fig. 4 Training and testing data set

This time overlap by no means indicates that we overlapped training and test samples. Rather it is a way we prepare the data information that includes recent history of the companies financial data. Different time domain slices effectively represent different (nonoverlapping) data so no overlap exists between training and testing data.

Since not all the features are reported for every company in the database, we must do the missing data recovery. We use the block-iterative missing data recovery approach based on the method presented in [10]. Since the financial data sets are extremely large (over 10261 companies with 192 features for each company), we first use the dimensionality reduction based on nonlinear PCA analysis using the SeDuMi toolbox [13]. In this way, we can significantly reduce the data dimensionality from 192 to 13~15 before data is applied to SOLAR for classification. The classifier was developed based on data slice from 1998-2000 and applied to predict performance in 2001, 2002 and 2003. Classification result above 50% means that using our classification method we can obtain a better than market performance. As we can see from Table 2, the proposed method can provide good classification results in this very difficult classification problem in the financial area.

Table 2: Average correct classification results for different years with the proposed method

| | Test year | | |
|---|---|---|---|
| | 2001 | 2002 | 2003 |
| Performance | 0.5846 | 0.5962 | 0.5577 |

*Case II: Power quality disturbance classification problem*

Power quality disturbance classification is a challenging and difficult issue in the power engineering community [14-15]. In this part, we show that the combination of the proposed optimized interconnections scheme with the self organizing learning array system can provide an accurate classification result for the power quality issue.

Wavelet multiresolution analysis (MRA) is used to construct the feature vector for SOLAR application. A 7 class power quality disturbance classification problem is considered here. For each type of disturbance class, 200 cases with different parameters based on the signal model presented in paper [16] were generated for training and another 200 cases were generated for testing.

We will compare the classification result obtained by the proposed method with the recent result reported in [16]. For each method, a $7\times7$ confusion matrix $C$ is constructed (see Table 3) to show the analysis results. The diagonal elements represent the correctly classified power quality types. The off-diagonal elements represent the misclassification. As we can see in Table 3, the approach proposed in this paper has better classification accuracy compared to the existing literature report.

## 5. Conclusion

In this paper, we proposed a novel type of probabilistic self-organized learning network, focusing on its interconnection structure. Effect of different parameters combinations for network performance were explored and tested. Theoretical analysis of optimal weighting and fixed weighting schemes were given. It is shown that a dynamically self-organizing neural network is able to locate the position of useful connections from the input set by actively searching for neurons with better classification rates. Experiments showed that the input selection strategy had a significant effect on the final performance. Application of the proposed method to financial data analysis problems and power quality classification problems demonstrated the effectiveness of this approach.

Table 3: Classification results for testing data set
(Db4 wavelet, 10 levels decomposition, 200 cases for each class)

| Method | Analysis Result | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Classification result as reported in [16]: Inductive Inference Approach | | C1 | C2 | C3 | C4 | C5 | C6 | C7 |
| | C1 | 200 | 0 | 0 | 0 | 0 | 0 | 0 |
| | C2 | 0 | 194 | 0 | 0 | 0 | 0 | 0 |
| | C3 | 0 | 0 | 153 | 0 | 11 | 36 | 0 |
| | C4 | 0 | 0 | 0 | 200 | 0 | 0 | 0 |
| | C5 | 0 | 0 | 1 | 0 | 180 | 19 | 0 |
| | C6 | 0 | 0 | 42 | 0 | 15 | 143 | 0 |
| | C7 | 0 | 4 | 0 | 0 | 0 | 0 | 196 |
| | Overall accuracy | ***90.4%*** | | | | | | |
| Method proposed in this paper: SOLAR with optimized interconnection scheme | | C1 | C2 | C3 | C4 | C5 | C6 | C7 |
| | C1 | 200 | 0 | 0 | 0 | 0 | 0 | 0 |
| | C2 | 0 | 200 | 0 | 0 | 0 | 0 | 2 |
| | C3 | 1 | 0 | 174 | 0 | 24 | 1 | 0 |
| | C4 | 0 | 0 | 0 | 200 | 0 | 0 | 4 |
| | C5 | 15 | 0 | 16 | 0 | 161 | 8 | 0 |
| | C6 | 0 | 0 | 2 | 1 | 2 | 194 | 1 |
| | C7 | 0 | 0 | 0 | 0 | 0 | 0 | 200 |
| | Overall accuracy | ***94.93%*** | | | | | | |

## References:


[1] D. F. Specht, "Probabilistic neural network," *Neural Networks, Vol. 3*, 1990, pp. 109-118;

[2] R. Polikar, L. Udpa, S. S. Udpa, and V. Honavar, "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE Trans. Syst., Man, Cybern. C, vol. 31*, no. 4, Nov. 2001.

[3] Tetsuya Hoya, "On the capability of accommodating new classes within probabilistic neural networks," *IEEE Trans. on Neural Networks, Vol. 14*, No.2 March 2003.

[4] K. Z. Mao, K. –C. Tan, W. Ser, "Probability neural network structure determination for pattern classification, " *IEEE Transactions on Neural Networks, Vol. 11*, No.4, July, 2000. 1009-1016

[5] B. Tian, M. R. A. Sadjadi, T. H. V. Haar, D. Reinke, "Temporal updating scheme for probabilistic neural network with application to satellite cloud classification, " *IEEE Transactions on Neural networks, Vol. 11*, No. 4, July, 2000. pp. 903-920.

[6] X. Jin, D. Srinivasan, R. L. Cheu, "Classification of freeway traffic patterns for incident detection using constructive probabilistic neural networks," *IEEE Transactions on Neural networks, Vol. 12*, No. 5, 2001, pp. 1173-1187

[7] Z. R. Yang, M. Zwolinski, C. D. Chalk, A. C. Williams, "Applying a robust heteroscedastic probabilistic neural network to analog fault detection and classification, " *IEEE Transactions on computer-aided design of integrated circuits and systems, Vol. 19*, No. 1, January, 2000. pp. 142-151

[8] R. K. Orr, "Use of a probabilistic neural network to estimate the risk of mortality after cardiac surgery", *Neural networks for cardiac mortality prediction, Vol. 17*/No. 2, Apr~Jun, 1997, pp. 178-184.

[9] J. A. Starzyk and T-H.Liu, "Design of a self-organizing learning array system," *IEEE Int. Symposium on Circuits and Systems (ISCAS)*, Bangkok, Thailand, May 2003.

[10] J. A. Starzyk, Z. Zhu, "Software simulation of a self-organizing learning array system," *The 6th IASTED Int. Conf. Artificial Intelligence and Soft Computing (ASC 2002)*, Canada.

[11] Janusz Starzyk, Zhen Zhu and Tsun-Ho Liu, "Self-Organizing Learning Array, " *IEEE Transaction on Neural Network*. 2004 (Accepted).

[12] *Standard & Poor's Research Insight SM COMPUSTAT (North America): A Primer for Getting Started*,[Online],Available: http://www2.library.unr.edu/dataworks/compu_primna76.pdf

[13] K. Q. Weinberger, F. Sha, and L. K. Saul "Learning a kernel matrix for nonlinear dimensionality reduction". *Proc. of the Twenty First Int. Conf. on Machine Learning (ICML-04)*, Banff, Canada, (2004).

[14] P. K. Dash, B. K. Panigrahi, D. K. Sahoo, G. Panda, "Power Quality Disturbance Data Compression, Detection, and Classification Using Integrated Spline Wavelet and S-Transform, " *IEEE Transactions on Power Delivery, Vol. 18*, No. 2, 2003, pp. 595-600

[15] G. Panda, P. K. Dash, A. K. Pradhan, S. K. Meher, "Data Compression of Power Quality Events Using the Slantlet Transform, " *IEEE Transactions on Power Delivery, Vol. 17*, No. 2, 2002, pp. 662-667

[16] T. K. A. Galil, M. Kamel, A. M. Youssef, E. F. E. Saadany, M. M. A. Salama, "Power Quality Disturbance Classification Using the Inductive Inference Approach, " *IEEE Transactions On Power Delivery*, Vol.19, No.4, October, 2004