Power Systems Disturbance Classification Using Modular Neural Networks with Multilayers Experts.

R. M. Magalhães, C. K. S. Santos, J. D. Melo, M. F. de Medeiros, A. D. Dória Neto

***Abstract*— This work evaluates the effectiveness of a new hybrid algorithm in the training of modular neural networks with multilayers specialists. The algorithm is applied on the classification of electric disturbances previously characterized and pre-processed, making possible the comparison between its performance with multilayers perceptrons networks. It presents the algorithm's development and inclusion of delta-bar-delta rule, such as the data acquisition methodology and its pre-processing.**

Key words: Eletrical Disturbance, Artificial Neural Networks, Modular Neural Experts, Hybrid Algorithm, Wavelet Transform.

I. Introduction

The artificial neural network has been utilized to solve a large number of engennering problems, including functions aproximation [8], control [9] as well as pattern classification [3]. The network type, its architecture and its training algorithm are chosen and evaluated according to dimension and complexity of the problems.

In the recent past years, the literature [10] presented some develoment directions in machine learning which have a highlight in this research scene, including the committee machines and also methods for scaling up supervised learning algorithms. During this period, ensembles studies, which deal with aggregation among many learning machines on the resolution of others complex problems, had a good effort with empiric data, formalization and new methods [11], [12], [13], [14]. Following the same idea, new training methods, such as hierarchy and parallel processing, have been developed and evaluated in several aplications [8], [15], [17], [18], [19].

The power quality suplied by an eletric system is one of the greatest point of interest for power supply companies and eletric energy consumers. Therefore, an efficient analisys of disturbances that imply in respect to power quality indices, in reduction of efficiency, is essential for all the agents involved [2]. The literature presents distinct approches in the acquisition, characterization and classification of disturbs present in high voltage transmission networks. Among another recent contribuitions, it could be included the application of Souza et al [4] and Ortiz et al [5] using multilayer perceptrons in disturbance classification. The discrete wavelet transform for characterization of voltage or current signals made by Machado et al [6] and, more recently, the detailed analysis of the influence of eletrical signal preprocessing in neural network classifier [1].

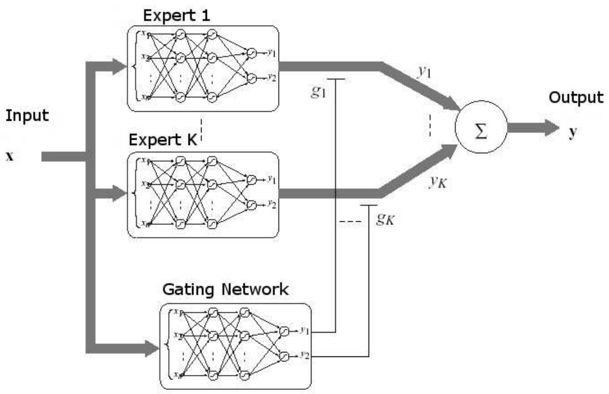

Fig. 1. Extended modular neural net diagram.

Following this develoment perspective, this work proposes a learning algorithm for extended modular neural network. The motivation is the unsatisfactory results in patterns classification and function approximation using Jacobs and Jordan architecture [16]. In such cases modular neural experts are composed by a single linear neurons layer and the training is done through associative gaussian mixture model.

The architecture proposed in this work uses multilayer preceptron experts, as shown in Figure 1. The network has K experts, L layers of q neurons with linear or nonlinear activation function and bias as well as a gate network with similar architecture.

The description of the data to be classified is shown in section II. In section III the developed algorithm to train the modular network with multilayer experts is presented. Results and comparison with methods found in the literature are given in section IV. Finally, conclusions are presented in section V.

II. Evaluated Disturbances

This work uses the same methodology and data as Medeiros et al [1] used in his work research about the influence of the signals preprocessing before the intelligent classifier. Next, a brief description of the process, processing and characterization of the data in four different classes is done.

The problem is divided into four main steps: capturing the signal, pre-processing, definition and classification of descriptors. The first step, which comes to obtaining the electrical signals has been carried out through the oscillography network of São Francisco Hydro Eletric Company (CHESF) and also from the simulation via Transient Alternative Program (ATP). This network consists of 370 oscillographs

The authors are in the Intelligent Systems Laboratory, Federal University of Rio Grande do Norte, Natal, RN 59078-900, BRAZIL (phone: +55 84 3215 3771; (email: {rafael,crisluci,jdmelo,firmino,adriao}@dca.ufrn.br).

operation with a sampling rate ranging between 20 and 256 samples/cycle. The signals used in this study were collected in voltage lines levels of 69, 230 and 500 kV, with a rate of 128 samples/cycle during 14 cycles [3].

The pre-processing stage is to suggest descriptors that characterize the signal variations when deviation from a certain standard is present. For this purpose, wavelet transform is used [6], for the extraction of important information from corrupted signal, enabling the identification of the electrical signal disturbance in question. This is done by a four steps algorithm [2]:

1) The decomposition of the corrupted signal with different resolution levels, resulting in various wavelet coefficients;
2) Calculation of the energy concetrated in each decomposition level [7];
3) Repetition of steps 1 and 2, just for the sinusoidal component of the referece signal, which is obtained from the first cycle of the analized signal,
4) Comparison between the energies concentrated in each level of the disturbance signal and the reference signal (Steps 2 and 3), by the percentage difference between the distributions of energy in each one, using the following relation

$$\Delta_j(\%) \;=\; \left[\frac{Edis_j - Eref_j}{max(Eref)}\right] \times 100 \qquad (1)$$

where $Edis_j$ is the energy content concentrated in level j for the disturbance signal, $Eref_j$ the energy concentrated in level j for the reference signal, and $max(Eref)$ is the maximum amount of energy from the reference signal.

Medeiros et al [2] shows, from applications, that is necessary to reduce the phase delay between the reference signal and the signal with disturbances so the classification stage has a better result. This can be done using a reference signal built from the registered signal.

The third step, which deals with the descriptors definition, is performed from the decomposition of signals from the previous step. The data to be classified are defined from 10 descriptors as presented in Table I.

After the descriptors were computed, four disturbances classes are defined according to Table II.

The final step is the classification. It is performed by the application of classifiers based on artificial neural networks. This work, uses the hybrid algorithm developed for modular networks architectures with multilayer perceptron. A comparison is done with the results obtained in this system with other multilayers perceptrons trained with the resilient backpropagation (RPROP).

III. Hybrid Algorithm

For the training of the modified modular network new algorithm was developed, adapted according to Jacobs and Jordan[16] algorithms,for the model of gaussian mixing associative, and also the erro backpropagation algorithm, by including the calculation of the descent gradients. Before we discuss it, some conventions should be given.

TABLE I

Patterns Descriptors

	Descriptors
1	Maximum absolute percentage difference of energies between the different wavelet decomposition levels of the signals with and without disturbance.
2	Level where occurred the maximum percentage difference
3	Second maximum percentage difference
4	Level where occurred the second maximum percentage difference
5	Third maximum percentage difference
6	Level where occurred the third maximum percentage difference
7	Fourth maximum percentage difference
8	Level where occurred the fourth maximum percentage difference
9	Fifth maximum percentage difference
10	Level where occurred the fifth maximum percentage difference

TABLE II

Disturbance class definition

	Classes
1	Voltage sag
2	Voltage swell
3	Harmonics
4	Transients

The modular network proposed has MLP experts (indexed by i, $i = 1,...,k$) with L^{Esp_i} layers (indexed by l, $l = 1,...,L^{Esp_i}$) with $q^{(l)Esp_i}$ neurons in each layer (indexed by j, $j = 1,...,q^{(l)Esp_i}$). And also a MLP gating network with L^{Pas} layers (indexed by ($l = 1,...,L^{Pas}$) with $q^{(l)Pas}$ neurons in each layer (indexed by j; $j = 1,...,q^{(l)Pas}$). The neurons activation fuctions in all networks can be linear or non-linear.

A. First step

The first step of the algorithm is the calculation of *a priori* probality associated to the i-th layer neuron output of the gating network

$$g_i(n) = \frac{exp(u_i^{(l)}(n))}{\sum_{j=1}^{K} exp(u_j^{(l)}(n))} \qquad (2)$$

where $u_i^{(l)}(n)$ is the i-th output neuron of the l-th layer from the gating network, is giving by

$$u_i^{(l)}(n) = \varphi_{p_i}^{(l)}(v_{p_i}^{(l)}(n)) \qquad (3)$$

where $\varphi_{p_i}^{(l)}$ is the derivative function from the activation potential $v_{p_i}^{(l)}(n)$ from the i-th neuron of the l-th from the gating network

$$v_{p_i}^{(l)}(n) = \sum_{j=0}^{q^{(l)Pas}} a_{ij}^{(l)} u_i^{(l-1)}(n) \qquad (4)$$

where $a_{ij}^{(l)}$ is the weight associated to the j-th neuron input i from the layer l from the gating network.

B. Second step

The second step of the algorithm is responsible to evaluate the the values of *a posteriori* probabilities $h_i(n)$ associated to the output i-th neuron from the output of the gating network, described by

$$h_i(n) = \frac{g_i(n)exp(-\frac{1}{2}\parallel \mathbf{d}(n) - \mathbf{y}_{e_i^{(k)}}(n) \parallel^2)}{\sum_{j=1}^{K} g_j(n)exp(-\frac{1}{2}\parallel \mathbf{d}(n) - \mathbf{y}_{e_i^{(k)}}(n) \parallel^2)} \tag{5}$$

where $\mathbf{d}(n)$ is the desired response and $\mathbf{y}_{e_i^{(k)}}(n)$ is the actual response of the neuron i from the layer l of the k-th expert

$$y_{e_i^{(k)}}^{(l)}(n) = \varphi_{e_i^{(k)}}^{(l)}(v_{e_i^{(k)}}^{(l)}(n)) \tag{6}$$

and $\varphi_{e_i^{(k)}}^{(l)}$ is the activation potential derivative $v_{e_i^{(k)}}^{(l)}$ from the neuron i, layer l, and expert k.

C. Third step

The work contribution is shown on the third step, where the incrementation in synaptic weights of the modular network is done with the multiple layers. The synaptic weights from the networks experts are updated according to the, followin equation

$$w_{e_i^{(k)}j}^{(l)}(n+1) = w_{e_i^{(k)}j}^{(l)}(n) + \eta\delta_{e_i^{(k)}}^{(l)}(n)y_j^{(l-1)}(n) \tag{7}$$

where η is the learning rate, and the gradient $\delta_{e_i^{(k)}}^{(l)}$ for the output layer neurons is obtained by

$$\delta_{e_i^{(k)}}^{(L)}(n) = h_i(n)e_i(n)^{(L)}\varphi'^{(L)}_{e_i^{(k)}}(v_{e_i^{(k)}}^{(L)}(n)) \tag{8}$$

where e_i is the difference between $\mathbf{d}_i$ and $\mathbf{y}_{e^{(k)}}$. The gradient for the neurons of hidden layers is computed as

$$\delta_{e_i^{(k)}}^{(l)}(n) = \varphi'^{(l)}_{e_i^{(k)}}(v_{e_i^{(k)}}^{(l)}(n)) \sum_{m=1}^{q^{(l+1)Esp_i}} \delta_m^{(l+1)}(n)w_{me_i^{(k)}}^{(l+1)}(n) \tag{9}$$

The synaptic weight update of the gating network is done through

$$a_{p_ij}^{(l)}(n+1) = a_{p_ij}^{(l)}(n) + \eta\delta_{p_i}^{(l)}(n)y_j^{(l-1)}(n) \tag{10}$$

where the output layer gradient is obtained by

$$\delta_{p_i}^{(L)}(n) = [h_i(n) - g_i(n)]\varphi'^{(L)}_{p_i}(v_{p_i}^{(L)}(n)) \tag{11}$$

The error is the difference between h_i and g_i. The gradient of hidden layers is calculated by

$$\delta_{p_i}^{(l)}(n) = \varphi'^{(l)}_{p_i}(n)(v^{(l)} \sum_{m=1}^{q^{(l+1)Pas}} \delta_m^{(l+1)}(n)a_{mp_i}^{l+1}(n) \tag{12}$$

Thus the error backpropagate from the gating network to its hidden layers. To increase the algorithm's efficiency the *momentum* term and *delta-bar-delta* rule was implemented as

$$\Delta w_{ji}(n) = \alpha\Delta w_{ji}(n-1) + \eta\delta_i(n)y_i(n) \tag{13}$$

where α is the *momentum* term and is usually a positive value. The *delta-bar-delta* works in function of the error comportament and change the synaptic weight update by

$$\Delta\eta_{ji}(n+1) = \begin{cases} \kappa & \text{if } S_{ji}(n-1)D_{ji}(n) > 0 \\ -\beta\eta_{ji}(n) & \text{if } S_{ji}(n-1)D_{ji}(n) < 0 \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

where D_{ji} and S_{ji} are defined by

$$D_{ji} = \frac{\partial E(n)}{\partial w_{ji}(n)} \tag{15}$$

and

$$S_{ji}(n) = (1-\xi)D_{ji}(n-1) + \xi S_{ji}(n-1) \tag{16}$$

where ξ is positive constant. The D_{ji} quantity is a current partial derivative error value w_{ji} synaptic weight. The $S_{ji}(n)$ parameter is sum of actual and early partial derivative error value w_{ji} synaptic weight.

IV. RESULTS

For the classification step two sets of data were used: the first one for training and the second one for validation. The training set consists of 800 normalized patterns characterized by 10 descriptors according to Table I and 800 patterns formed by 4 output values according to Table II. (With data obtained directly from real oscillograph records and also from simulated data by the ATP software.)

For validation set 344 input patterns were used with data obtained from the oscillographs. For example, a pattern belonging to class 1 has as a vector output column $[1\ 0\ 0\ 0]^T$.

For the development of modular network architectures is necessery consider the architectures developed by Medeiros et al [1]. Accordingly, it is proposed the equivalence between the number of free parameters and the number of nonlinearity present (neurons) in the network. Considering the best result obtained from his work, the architecture has 10 input values, 10 neurons in the first hidden layer, 40 neurons in the second hidden layer and 4 neurons in the output layer, with an architecture [10→10:40:4]. It is equivalent to have an architecture of 660 free parameters, and 54 neurons. The Table III shows the multilayer perceptrons architecture and modular networks created.

The modular networks have been implemented using C++ language utilising the GNU gcc compiler version 4.1.2, on the GNU/Linux platform. The four modular architectures presented in Table III were trained with different learning parameters. The algorithm learning rate (η) was evaluated in a range of values between 0.001 - 0.40, with distint rates for

TABLE III

ARCHITECTURE COMPATIBILITY

Multi-layers architectures (MLP) [1]				
Net	-	MLP-1	MLP-2	MLP-3
Arch.	-	10→10:20:4	10→10:40:4	10→10:60:4
Params	-	380	660	940
Neurons	-	34	54	74
Multi-layer modular arch. (MLMOD)				
Net	MOD-0	MOD-1	MOD-2	MOD-3
N. experts	3	3	3	3
Exp arch.	10→3:4	10→5:4	10→10:4	10→15:4
Gat arch.	10→5:4	10→5:4	10→10:4	10→15:4
Paramss	196	280	560	840
Neurons	30	36	56	76

experts networks and gating networks. The number of epoch in training has been estimated between 500 and 1000.

Table IV shows a comparison between the sucess percentage in the classification of the validation data set. It is being presented the best results in both architectures. The accuracy values for modular networks were obtained from the average of 10 training executions in each architecture. In this case, the validation data set efficiency of the MOD-0 network was 100

TABLE IV

PERCENTUAL SUCCESS RATE COMPARISON FOR VALIDATION DATA SET

Multi-layers architecture (MLP) [1]				
Net	-	MLP-1	MLP-2	MLP-3
Success rate	-	90,69%	99,70%	84,59%
Multi-layer modular arch. (MLMOD)				
Net	MOD-0	MOD-1	MOD-2	MOD-3
Success rate	98,46%	99,48%	100%	100%

From Tables IV and III it can be seen that the modular networks with the proposed algorithm reach a high amount of accuracy (approx. 100% of accuracy) results not achieved with a MLP network. As expected, a reduction of free parameters of different architectures for modular networks, entails a reduction in learning ability. This is striking in the case of networks MOD-1 and MOD-0, when the reduction of parameters was from 280 to 196 (30%) in neurons from 36 to 30 (approximately 16.7%), an average reduction of accuracy in 1.02 %. In these cases, analyzing the data of training and the confusion matrix, it is observed that the difficulty of learning comes from class 3 (transient).

In order to present the modular network behavior, graphics were generated from the output provided by experts networks, the gating network and the global output modular network. Figure 2 presents a global response from the modular network MOD-0, after a 500 epochs traning, in the classification of 344 patterns from validation data set, and the points from 1 to 86 in the horizontal axis, belonging to class 1, 87 to 172 (class 2), 173 to 258 (class 3) and 259 to 344 (class 4). The curves represent the extent of each individual output from the network for the input data.

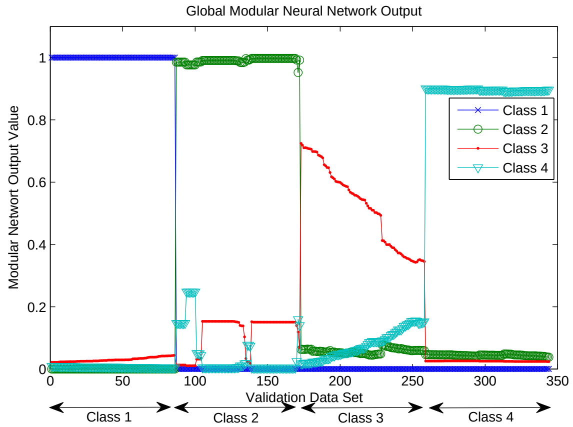

Fig. 2. Global output modular neural network for validation data set.

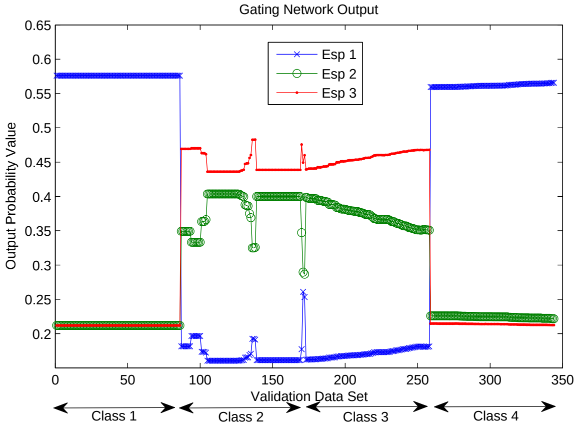

Fig. 3. Gating Network output for validation data set.

According to the analysis above, the classes are well identified. And in particular, range of points between 230 and 250, there is a close choice between the classes 2 and 3, however, in the worst situation since the value assigned to class 3 in the region is 220.73% higher than the value assigned to the class 2, thus ensuring a stability of the selection.

The Figure 3 shows the gating network output for 344 patterns from the validation data set. The output is represented by 3 curves, each one assigned to a function of probability in the selection of a particular expert. Thus, the first output assigns a value of weighting in the output provided by expert 1 in combination of global output. Similarly the output 2 and 3 take the networks experts 2 and 3, respectively. For patterns of class 1 and 4 the gating network provides greater value to expert network 1 and, for patterns from classes 2 and 3, the gating network provides greater values to expert networks 2 and 3, respectively. Therefor, it can be conclud that there was a specialization of components of the modular network (expert networks) in certain sub-data of the data used to train and validate the network.

Figures 4, 5 and 6 present the output provided by network experts 1, 2 and 3 respectively, due to validation data set.

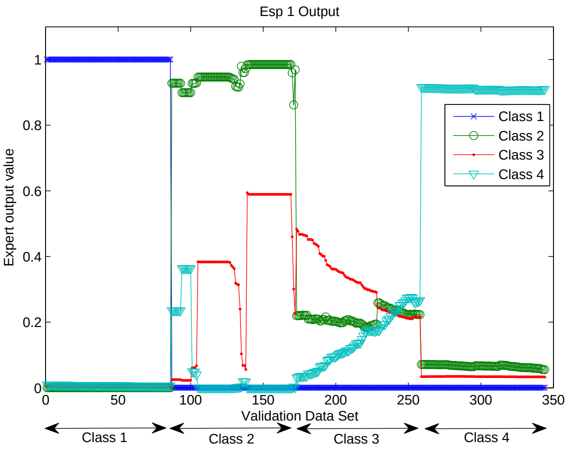

Fig. 4. Expert 1 output for validation data set.

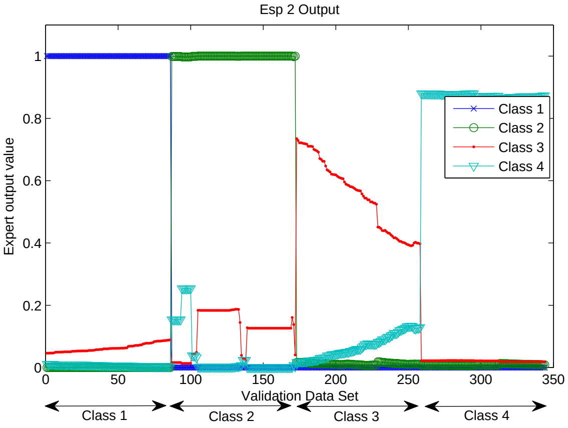

Fig. 5. Expert 2 output for validation data set.

Figure 4 shows the appropriate classification for the elements of class 1 validating therefore, the output of the gating network, when attaches the greatest likelihood of adjustment to the first expert. In Figures 5 and 6 there is a better classification of data for classes 2 and 3, if compared to the classification assigned by expert 1, characterizing again a good choice done by modular network. Finally, for the class 4, the gating network provides greater weighting to the expert network 1, although the others three networks classify properly, the first is the one that presents the greatest value of extent in the selection.

V. Conclusions

The results of the training show that the use of the proposed algorithm for the modular multilayer architectures obtained adequate results. Considering the proposal equivalence between architectures, modular networks reached higher values than simple multilayers perceptron networks (MLP). Reduced architectures were evaluated, but instability occurred, causing the convergence were not achieved in all cases.

As further steps, it is proposed a thorough study on

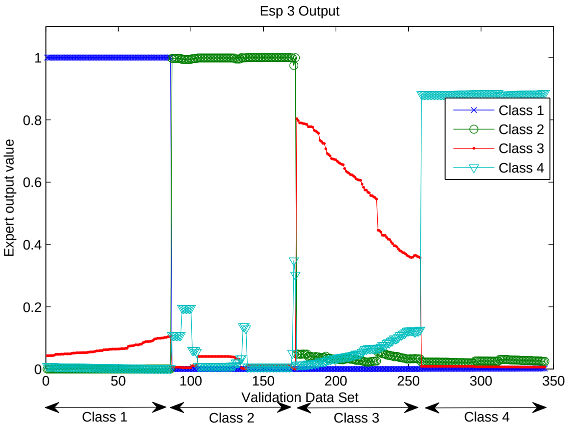

Fig. 6. Expert 3 output for validation data set.

efficiency of the algorithm in modular networks, including an assessment of the processing time and also the addition of elements such as *momentum* term and the delta-bar-delta rule.

Acknowledgment

The authors would like to thank the financial support from the Coordenation of Improvement of Higher Level of Personnel (CAPES), which partly enabled the development of this work and the Company Hydro Valley of San Francisco (CHESF) for technical and financial support.

References

[1] M. F. Medeiros Jr. and C. K. S. Santos and J. T. Oliveira and P. S. M. Pires and J. D. Melo and A. D. D. Neto and J. J. A. L. Leitão, “Influence of Signal Pre-Processing in the Efficiency of Algorithms Based on Neural Networks for Disturbance Classification”,*Computational Intelligence in Image and Signal Processing*, pp 95–100, 2007.

[2] M. F. Medeiros Jr. and J. T. Oliveira and E. G. M. Lacerda and C. K. S. and J. J. A. L. Leitão, “Disturbances Caracterization in Electric Energy Networks Using Second-Generation Wavelet Transform”,*VI INDUSTRY APPLICATION CONFERENCE - VI INDUSCON*, Joinville -SC. VI Induscon, 2004.

[3] M. F. Medeiros Jr. and C. K. S. and J. J. A. L. Leitão and L. R. Soares, “Influência do Pré-Processamento de Sinais na Eficiência de Algoritmos Baseados em Redes Neurais Para Classificação de Distúrbios”,*Simpósio Brasileiro de Sistemas Elétricos*, Campina Grande, 2006.

[4] B. A. Souza and N. S. D. Brito and W. L. A. Neves and R. B. Lima and S. S. B. Silva, “Comparison between backpropagation and RPROP algorithms applied to fault classification in transmission lines”,*Proceedings of The International Joint Conference on Neural Networks & International Conference on Fuzzy Systems* , Budapeste, 2004.

[5] F. Ortiz and A. Ortiz and M. Manana and C. J. Renedo and F. Delgado and L. I. Eguíluz, “Artificial Neural Networks approach to the voltage sag classification”,*International Conference on Renewable Energies and Power Quality'07*, Sevilla, 2007.

[6] R. N. Machado and U. H. Bezerra and E. G. Pelaes, “Classification of signal with voltage disturbance by means of wavelet transform and intelligent computational techiniques.”,*WSEAS Transactions on Power Systems*, v. 1, p. 1538-1532, 2006.

[7] J. W. Resende and C. Penna, “Identification of Power Quality Disturbances using the MATLAB Wavelet Transform Toolbox”,*International Conference on Power Systems Transients (IPST)'01*, Rio de Janeiro, 2001.

[8] R. L. S. Alves, “Competitive Strategies for Multilayer Perceptrons Training using Backpropagation and Parallel Processing”, *Wseas Transactions On Systems*, v. 3, n. 2, p. 352-357, 2004.

[9] C. W. Anderson and Z. Hong, “Reinforcement learning with modular neural networks for control”, *Proceedings of NNACIP‘94, IEEE International Workshop on Neural Networks Applied to Control and Image Processing*, 1994, pp. 90–93.

[10] T. G. Dietterich, “Machine-learning research: four current directions”, *The AI Magazine*, vol. 4, pp 97–136, 1998.

[11] D. Optiz and R. Maclin, “Popular ensemble methods: an empirical study”, *Journal of Artificial Intelligence Reserach*, vol. 11, pp 169–198, 1999.

[12] G. Valentini and F. Masulli, “Ensemble of learning machines”, in *Neural Nets WIRN Vietri-02*, Series Lectures Notes in Computer Sciences, Springer-Verlag, 2002.

[13] L. I. Kuncheva, *Combining pattern classifiers: methods and algorithms.*, John Wiley & Sons, New Jersey, 2004.

[14] C. Dimitrakakis and S. Bengio, “Online adaptive policies for ensemble classifiers.” *Neurocomputin*, vol. 64, pp 211-221, 2005.

[15] P. A. Estévez, H. Paugam-Moisy, D. Puzenat and M. Ugarte, “A scalable parallel algorithm for treining a hierarchical mixture of neural experts”, *Parallel Computing*, vol. 28, pp 861-891, 2002.

[16] R. A. Jacobs and M. I. Jordan, “Adaptive mixtures of local expert”, *Neural Computation*, vol. 3, pp 79–87, 1991.

[17] A. C. M. L. Albuquerque and J. D. de Melo and A. D. Doria Neto, “Part III Evolvable Designs: Evolutionary Computation and Parallel Processing Applied to the Design of Multilayer Perceptrons”. *In: Nedjah, Nadia; (EDS.), de Macedo Mourelle, Luiza. (Org.). Evolvable Machines*. Belim, 2004, v. 161.

[18] I. Wesley-Smith, “A parallel artificial neural network implementation”,*Proceedings of The National Conference On Undergraduate Research*, 2006.

[19] S. Kontár, “Parallel training of neural network for speech recognition”, *Proc. 12th International Conference on Soft Computing MENDEL*, 2006.