Modified Cascade-Correlation Algorithm for Improved Transformer Tap Changing Operation

M. Fakhrul Islam, Joarder Kamruzzaman and Guojun Lu

Gippsland School of Computing & Information Technology
Monash University, Churchill 3842, Australia
{Fakhrul.Islam, Joarder.Kamruzzaman, Guojun.Lu }@infotech.monash.edu.au

Abstract-- The previous researches have shown that artificial neural network (ANN) can be used for on-load tap changer control of parallel transformers in power distribution system [1],[2],[3]. In those research reports the proposed ANN for application in this control were developed using various algorithms for obtaining good performance. However, further improvement of ANN based transformer tap changer operation is always desirable. A thumb rule for obtaining good generalization is to use the smallest network that solves the problem [4]. In this paper we show that a small size of ANN is obtainable for improved transformer tap changer operation by modifying the Cascade-Correlation (CC) algorithm. Experimental results demonstrate that significant improvement in performance is achieved using modified cascade-correlation algorithm instead of the standard cascade-correlation. The comparison of the ANN performances of the algorithms in this application is analyzed and the results are presented.

Index Terms—Neural Network Application, Cascade correlation, Bayesian regularization, Transformers tap changers, Voltage control.

I. Introduction

The existing tap changer control such as the master/follower, the power factor, the negative reactance, and the circulating current, can operate the tape changer only in closed primary bus system. The newly proposed artificial neural network control of tap changer operation allows the parallel transformers to operate both in closed primary bus system [2] and also when the primary busbar is connected across the power network [3]. The ANN takes voltage level, bus coupler circuit breaker status, circulation current, real and reactive power of transformer, real and reactive power of total load as the inputs and can map operational decision at the output which in turn relays to the operating mechanism of the tap changer control. The input variables to the ANN are continuous except the circulating current and bus coupler status which are basically used as supervisory controls. Since some of the variables are continuous, it is necessary to calculate a set of input data that would represent all the possible variations of the power system and tap positions of the parallel transformers. We have used sampling values of the continuously changing variables in preparation of the data sets.

For effective implementation of ANN in tap changer operation, it must possess high generalization ability. The generalization characteristic of ANN depends on the architecture, learning algorithm and quality of the training data. The training data should cover the problem domain and carry the features that are effective in solving the problem. Learning algorithm plays a vital role in achieving generalization ability of ANN. A single learning algorithm can not assure the best result for all types of problems. Choice of learning algorithm to achieve the best performance is somewhat problem specific and the choice can be made from a variety of existing algorithms or needs to be modified to suit the particular application. Alternately, it has to be newly developed.

Although the backpropagation has many reports of successful implementation on various complex problems, the standard backpropagation with the steepest gradient descent learning algorithm is found too slow in the training phase [5],[6]. It has undergone many modifications to improve the learning speed e.g. Bayesian regularization [7], and scaled conjugate gradient [8]. Previous experiments on tap changer operation problem show that Backpropagation with Bayesian regularization (BR) provides better performance than the scaled conjugate gradient and standard backpropagation [3]. These learning algorithms use fixed architecture which has several limitations (discussed in section II).

The aim of this work is to further improve the performance of tap changer operation using ANN presented in previous studies [1],[2],[3]. Constructive learning algorithms that automatically build network architecture of required size have been reported to perform better than that of fixed architecture [9]. We modified the cascade correlation algorithm proposed by Fahlman *et al.* [10] to construct a network for tap changer operation which showed improvement over fixed architecture studied previously.

II. Fixed Vs constructive ANN architecure

Good generalizing capacity of the ANN demands the optimal architecture. If the network is very small it can not learn well and in the case of a large architecture it memorizes the training data and exhibits the poor generalization [11]. The standard back propagation, Bayesian regularization and scaled conjugate algorithm train the ANN on a pre-selected fixed architecture. In order to find the correct size of the architecture a large number of networks are to be trained and the one which provides the best results is selected. This trial and error process is practically cumbersome.

In constructive approach, e.g. cascade-correlation algorithm [10], training and architecture of an ANN are developed simultaneously where the architecture grows from the small size to its required size. Since the architecture development and training of an ANN are simultaneously carried out in cascade-correlation algorithm, it has become highly desirable to avoid the exhaustive approaches of finding correct structure in trial error method.

Another proposed method that minimizes the architecture to some extend is known as pruning algorithm [12] which is related with the active participation of weight connections. When the weight connections do not contribute to error minimization, it is removed to reduce the architecture. In the pruning process, initially the training is started with a network larger than necessary and gradually reduces the network size. This requires extremely larger training time. In the contrary, cascade-correlation network starts with a minimal architecture and gradually grows the required size accruing to less computational time. Recent studies on many problems show that cascade-correlation learning network is more suitable for classification problem instead of regression tasks [13], [14]. In this study we considered cascade-correlation algorithm with a view of achieving better performance because of its ability to grow network architecture automatically to the required size and its reportedly good generalization capacity in some studies.

III. Cascade-correlation learning Algorithm

Cascade-correlation network initially starts by training a single output layer using the original input data set. If there exists a weight space $\hat{\mathbf{w}}$ that can classify the input data using a single layer of weights the problem is easily solved. But it is impossible to achieve such solution if the problem is not linearly separable. In that case hidden units are added one at a time unless a weight space is found that can classify the training data. The hidden units are generated by training a candidate unit as follows. For the first hidden unit the candidate unit is connected to the input units and trained to maximize the correlation between its output and the residual error of the output layer over the training data set. Once trained the weights connecting the input to candidate unit is frozen and the candidate unit is then connected to the output layer as a hidden unit and output layer (i.e., all weights connecting to the output units from the inputs as well as newly installed hidden unit) is trained again to minimize the residual error. From the second hidden unit the candidate unit is connected to the input units and all the previously generated hidden units and is trained in the same way using the recent residual error. In this process the hidden layers form a cascade-structure of the network as shown in Fig. 1. A detail description is found in [10].

The output layer in the standard cascade correlation is trained to minimize the sum of the squared error over the training data sets as follows.

$$E = \frac{1}{2}\sum_p \sum_k \left(t_{p,k} - y_{p,k}\right)^2 \quad (1)$$

where $t_{p,k}$ and $y_{p,k}$ are the target and actual outputs of output neuron *'k'* for pattern *'p'*.

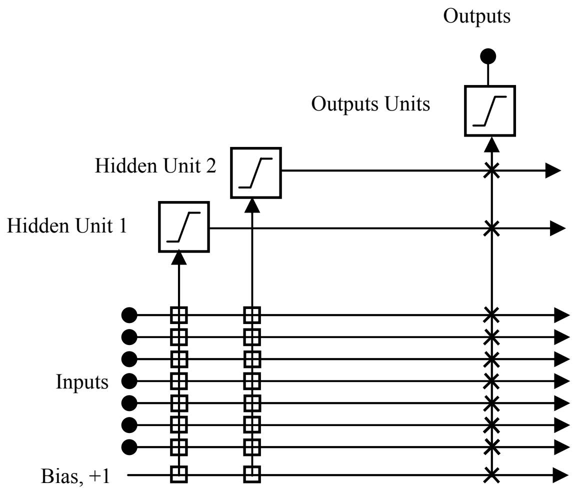

Fig. 1. The cascade-correlation architecture with two hidden units added. The vertical lines represent all incoming activations. Connections represented by boxes are frozen and those represented by 'x' are repeatedly training.

The weight update to minimize sum of squared error is carried out using gradient descent technique as in (2)

$$\mathbf{w}(t) = -\eta \nabla E\big|_{\mathbf{w}=\mathbf{w}(t)} + \alpha \Delta \mathbf{w}(t-1) \quad (2)$$

where E is the residual error for connection weights **w***(t)* in the output layer, $\Delta\mathbf{w}$ *(t)* is the weight change in *t*-th iteration, η and α are the learning rate and a momentum factor respectively.

In order to generate a hidden unit, each candidate unit is trained by maximizing the correlation 'C' between its output and the latest residual error at the output layer. The correlation 'C' is defined as

$$\mathrm{C}=\sum_k \left| \sum_p (y_p - \bar{y})(E_{p,k} - \bar{E}_k) \right| \quad (3)$$

where '*k*' is the network output at which the error is measured and '*p*' is the training data. The quantities $\bar{y}$ and $\bar{E}_k$ are the values of y and E_k averaged over all training data. The maximization of correlation 'C' is carried out by gradient ascent governed by (4) and (5).

$$\delta_p = \sum_k \sigma_k (E_{p,k} - \bar{E}_k) f'_p \quad (4)$$

$$\frac{\partial C}{\partial w_i} = \sum_p \delta_p I_{i,p} \quad (5)$$

where σ_k is the sign of the correlation between the candidate unit value and the residual error at output 'k', f'_p is the derivative of the activation function of the candidate unit with respect to the sum of its inputs for pattern 'p', $I_{i,p}$ is the input the candidate unit receives from unit 'i' for pattern 'p', and w_i is the weight connection. The weight of the candidate unit is updated using (6) and (7).

$$\Delta w_i(t) = \lambda \frac{\partial C}{\partial w_i(t)} \tag{6}$$

$$w_i(t+1) = w_i(t) + \Delta w_i(t) \tag{7}$$

where λ is a constant.

IV. MODIFIED CASCADE-CORRELATION LEARNING ALGORITHM

Many previous research for recognition of handwritten ZIP code, numeral, alphanumeric characters, show that weight smoothing is useful to improve ANN generalization capability [15],[16],[17]. Weight smoothing is a regularization technique. As stated in Section I, previous study on ANN based tap changer showed that Bayesian regularization demonstrated better generalization performance than other algorithms. This is due to the "smoothness" of weights achieved by this algorithm. In order to utilize connection weight constraints along with architectural advantage, we modified the error function in the output layer of cascade-correlation algorithm by incorporating Bayesian regularization. Therefore, for the output layer, the minimization function E is changed into an objective function F defined as

$$F = \gamma E_D + (1-\gamma) E_W \tag{8}$$

where E_D is the sum of squared errors, $E_w = \|\mathbf{w}\|^2/2$ is the sum of squares of the network parameters, and γ (<1.0) is the performance ratio parameter, the magnitude of which dictates the emphasis of the training. In the Bayesian framework weight space is initially assigned to provide the prior distribution. If D={$\mathbf{x}_m$, $\mathbf{t}_m$} is the data set of the input-target pair, the posterior probability of distribution of the weight $p(\mathbf{w}|\mathrm{D},\gamma)$ can be expressed as

$$p(\mathbf{w}\,|\,\mathrm{D},\gamma) = \frac{p(\mathrm{D}\,|\,\mathbf{w},\gamma)\,p(\mathbf{w}\,|\,\gamma)}{p(\mathrm{D}\,|\,\gamma)} \tag{9}$$

where $p(\mathbf{w}|\gamma)$ is the prior distribution, $p(\mathrm{D}|\mathbf{w},\gamma)$ is the likelihood function and $p(\mathrm{D}|\gamma)$ is a normalization factor, which guarantees that the total probability is 1. The optimal weight, in Bayesian framework, maximizes the posterior probability $p(\mathbf{w}|\mathrm{D},\gamma)$ that is equivalent to minimizing the function in (8). The performance ratio on parameter γ is optimized by applying the Bayes' rule

$$p(\gamma\,|\,\mathrm{D}) = \frac{p(\mathrm{D}\,|\,\gamma)\,p(\gamma)}{p(\mathrm{D})} \tag{10}$$

If we assume a uniform prior density $p(\gamma)$ for the regularization parameter γ, then maximizing the posterior probability is achieved by maximizing the likelihood function $p(\mathrm{D}|\gamma)$. Since all probabilities have a Gaussian form it can be expressed as

$$p(D\,|\,\gamma) = (\pi/\gamma)^{-N/2}[\pi/(1-\gamma)]^{-L/2} Z_F(\gamma) \tag{11}$$

where L is the total number of parameters in the ANN. Supposing that F has a single minimum as a function of $\mathbf{w}$ at $\mathbf{w}^*$ and has the shape of a quadratic function in a small area surrounding that point, Z_F is approximated as

$$Z_F \approx (2\pi)^{L/2} \det{}^{-1/2} H^* \exp(-F(\mathbf{w}^*)) \tag{12}$$

where $H=\gamma\nabla^2 E_D + (1-\gamma)\nabla^2 E_W$ is the Hessian matrix of the objective function. Using (12) into (11), the optimum value of γ at the minimum point can be determined.

Foresee and Hagan [7] propose to apply Gauss-Newton approximation to Hessian matrix, which can be conveniently implemented if the Lebenberg-Marquart optimization algorithm is used to locate the minimum point. This minimizes the additional computation required for regularization.

V. DATA COLLECTION

In ANN control the nature of tap-changer operation is treated as the solution of a classification problem [1]. A comprehensive data set that can be straightaway applied to train ANN and test its performance is not possible to obtain from the electric supply company. The main reason is that the existing tap changer control methods has restriction to the way the ANN control will allow the transformer parallel operation since they do not comply with that operation. So, the company can not have the data at the critical situations (e.g. transformers when connected across the network and tap positions are not appropriate) of the transformers parallel operation in the substations. However, the data those are available to electric supply company is useful to calculate the necessary data to train ANN and test it for performance evaluation. The detail system analysis, mathematical formulae and procedure for calculation of the data set are presented in [2],[3]. Transformer details and load data variations in the year 2002 of a 220 kV substation in Victoria, Australia have been used to calculate the required data set.

A. Consideration in Pre-processing the Data

The transformers in the above mentioned substation were found carrying lower load in comparison to their capacity. Since the substation has to meet the growing demand of the power network in future we have considered the maximum permissible load of the two transformers equal to their normal capacity ratings for this data generation.

This experiment is carried out using seven variables in the input vector components. These are voltage level, bus coupler circuit breaker status, circulation current, real and reactive power of transformer, real and reactive power of total load. Two of the seven input variables in the data set are used as supervisory information to the ANN. They are coupling circuit breakers and circulating current between the transformers. The radial operation of the transformers occurs if any of the coupling circuit breakers is off and is represented by '0'. On the contrary the transformers parallel operation occurs if all the coupling circuit breakers are closed and is represented by '1'. The circulating current can be expressed as follows:

$$I_{circ} = \begin{cases} 0 & \text{circulating current is within the threshold value} \\ 1 & \text{circulating current exceeds the threshold value} \end{cases}$$

The threshold value is always less than the circulating current that flows when any of the transformers tap positions is different by only one tap apart from its appropriate tap position that provides equal voltage at the secondary bus.

B. Determining the Data Sets

The actual circulating current values and other variables such as real and reactive power sharing of parallel transformers, when transformers taps are connected in appropriate as well as inappropriate positions, are determined by preparing a number of tables considering voltage differences in magnitude and phase angles of 0^0, 2^0, 5^0, 8^0, and 10^0 when primary side of the parallel transformers are connected across power network. For the voltage magnitude and phase angle variation at the primary side of the two parallel transformers, the appropriate tap positions that give the equal voltage at the secondary bus were selected. In this appropriate tap position the circulating current reduces to almost zero for identical transformers. Reactive power and real power components are proportional between the transformers and also between the individual transformers to the total substation power components.

The sharing of real and reactive power components by the individual transformer for inappropriate tap position is calculated as follows: one of the transformers tap position was fixed to appropriate tap position and the other transformer tap-position is moved from the appropriate position to create the differences from 1 tap to maximum 9 taps (the transformer has 10 tap positions). For each of the cases of 1 to 9 tap differences, the power components and circulating currents are determined using the voltages and impedances from the tables. The calculation was repeated for different loads from the range of the power variation of the substation. We use the equations and related equivalent circuits presented in [3] to determine the effective impedances to prepare the tables and calculate the numerical values of variables to generate the data set.

Numerical values of the variables calculated at every stage are grouped to form the input vector. The input vectors are divided into three classes and associated with target values in ANN according to the tap changer operation as '-1' for tap lower; '0' for tap hold; and '1' for tap rise.

Training data set is formed by taking 50% of each data set representing 0^0, 5^0 and 10^0 voltage phase angle differences. The test data is formed by 50% of the remaining data from each set representing 0^0, 5^0, 10^0 and also by full data sets from 2^0 and 8^0 voltage phase angle differences.

VI. Experimental Results and Discussion

A. Training the Cascade-Correlation Architecture

The training data set is used to train and develop the architecture by adding hidden units of the ANN as shown in Fig. 1. The learning and architecture construction is terminated by a set of criterion such as achievement of minimum error, maximum hidden units, number of epochs etc. We used a pool of candidate units each with a different set of random initial weights instead of a single candidate unit. All units in the pool receive individually the same input signals and maximize correlation with the same residual error for each training data. After the training is terminated, the candidate unit that has the best correlation is installed.

We carried out 90 trials with each of the training algorithms with similar learning parameters and using different initial weights. The predicted values of ANN outputs in all the algorithms are interpreted using the threshold values as follows:

$$\text{ANN output} \begin{cases} > 0.5 & \text{tap rise} \\ < -0.5 & \text{tap lower} \\ \text{otherwise} & \text{tap hold} \end{cases}$$

Our primary interest is to obtain an ANN which provides generalization as higher as possible. In classification problems the principle selection criterion for assessment of the good performance is the misclassification rate [5]. However, it is desirable to achieve the best generalization performance with smaller number of hidden units in order to avoid overtraining.

B. Comparison of Cascade-Correlation Trained ANNs

In this section we analyze and compare the results of the ANNs trained by CC and modified CC (MCC) learning algorithms. The comparisons are in terms of the false response rate (which is same as misclassification) and the size of the architecture. Best performance of the algorithms in the cascade-correlation is then compared with the backpropagation with Bayesian regularization which was found to have better performance in the previous study.

Table 1 shows the convergence of cascade-correlation training algorithms to the solutions of different hidden unit number. It can be observed from Table 1 that CC algorithm, though converged into smaller number of hidden units in few trials, mostly converged at seven hidden units. On the other hand, MCC more often converged at four hidden units. This demonstrates that MCC is more likely to converge to smaller sized network.

Table 2 shows the best, worst, average values and standard deviation (STDEV) of ANN false response rates in percentage for the two algorithms with different number of hidden units.

TABLE I
ANN CONVERGING TO THE SOLUTIONS WITH HIDDEN UNITS (HU)

ALGORITHMS	HU 4	HU 5	HU 6	HU 7
CC Learning	2	3	9	76
MCC Learning	20	14	10	46

TABLE 2
ANN FALSE RESPONSE RATES IN DIFFERENT HIDDEN UNITS

Algorithms	Hidden Units	Best	Worst	Average	STDEV
CC Learning	4	1.1	2.4	1.8	0.009
	5	0.5	1.8	1.3	0.007
	6	0.4	2.2	1.1	0.006
	7	0.3	6.8	2.1	0.013
MCC Learning	4	0.3	0.74	0.41	0.001
	5	0.32	0.91	0.45	0.002
	6	0.34	1.12	0.63	0.003
	7	0.28	2.41	1.03	0.006

The trials those converged at seven hidden units (equal to input size) reached the maximum size of the architecture allowed in the structure development and learning process. Many of the previous research works used the hidden layer neurons less than the neurons in input layer [17],[18]. Some researchers used hidden layer neurons 30% less than the number of neurons in the input layer [19]. In general, it is expected that a suitable smallest architecture with hidden units at least less than that in the input should avoid overfitting and most likely to achieve better generalization. So, we aimed to concentrate to those ANNs that converged with fewer than seven hidden units.

Figure 2 represents the false response (misclassification) rate versus the ANN trials that converged with four to six hidden units in two algorithms for cascade-correlation and for modified cascade-correlation. In CC only two out of 90 trials are found to converge with four hidden units but the false responses are higher than the trials converged with more hidden units. Contrarily, in MCC the numbers of trials that converged with four hidden units is increased to twenty and have lower false response rates than those converged with more hidden units (Fig. 2(b)). Figures 3 and 4 represent the average values and standard deviation of ANN false response rates for different hidden units as converged in CC and MCC training algorithms. It is observed that both the average values and standard deviations of ANN false responses for each set of the hidden units is lower in the case of MCC than CC and the lowest values are always in MCC when converges with four hidden units.

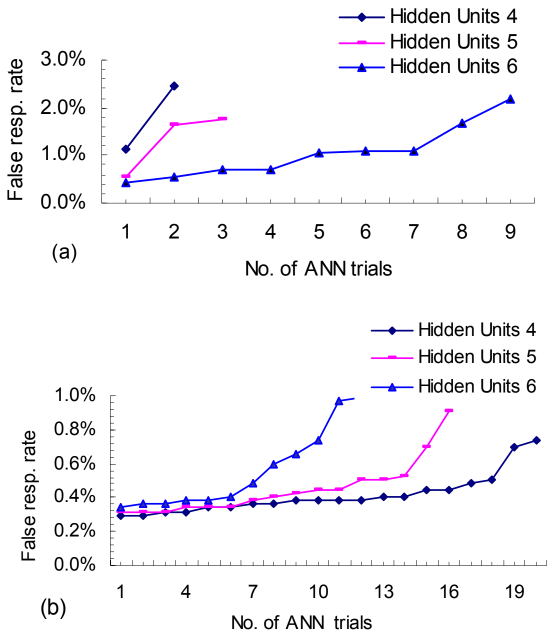

Fig. 2. ANN false response rate vs. trials in ascending order of false response (a) for cascade-correlation, (b) for modified cascade-correlation.

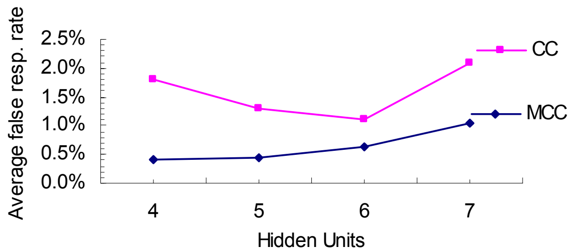

Fig. 3. Average false response rate versus hidden units for CC and MCC trained ANN.

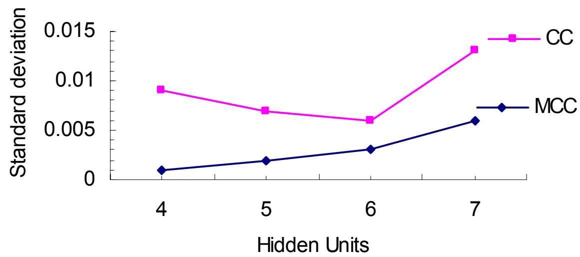

Fig. 4. Standard deviation among the trials versus hidden units for CC and MCC trained ANN.

C. Comparison between Modified CC and Backpropagation with BR

We trained a two-layer ANN architecture consisting four hidden units by backpropagation using BR algorithm for twenty trials. The false responses in MCC trials that converged with four hidden units are then compared with the false responses from BR trained ANN. Figure 5 shows the false response rates of the BR and MCC trained ANN groups in ascending order of their values. The comparison shows that

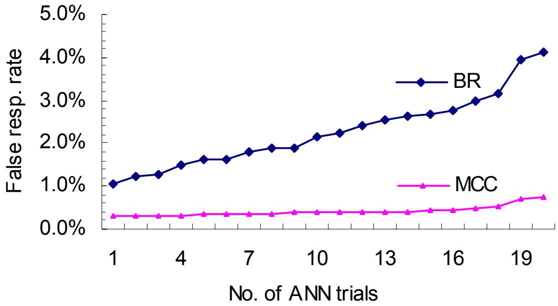

Fig.5. ANN false response rate vs. trials in ascending order of false response with 4 hidden units for BR and MCC.

false response rates from the MCC trials are much better than those obtained using backpropagation with Bayesian regularization. The average false response over all the trials with four hidden units in MCC is 0.41% and that in backpropagation with BR is 2.27%. This indicates that, for tap changer operation, the MCC algorithm performs much better than backpropagation with BR.

VII. Conclusions

In this paper, we studied the performances of ANN trained by standard cascade-correlation algorithm for tap changer operation and proposed a modified cascade-correlation (MCC) algorithm to obtain a better solution. The performances of the proposed MCC algorithm trained networks are compared with the results from the standard cascade-correlation and backpropagation with BR trained networks. Backpropagation with BR trained network was previously found as the best among many other algorithms of fixed architecture topology. In this comparison the network trained by the proposed algorithm shows better generalization than all others. This concludes that the proposed MCC algorithm is the effective design option in building neural networks for the tap changer operation.

VIII. Reference

[1] F. Islam, B. Nath, and J. Kamruzzaman, "Reactive Load Control of Parallel Transformer Operations using Neural Networks," *Proc. of 15th Int. Conf. on Industrial and Engineering applications of Artificial Intelligence and Expert Systems, IEA/AIE 2002*, pp. 824-830, 2002.

[2] F. Islam, J. Kamruzzaman, and G. Lu, "Artificial Neural Network for Parallel Transformers Tap-Changer Control for Closed Primary Bus Connection," *Proc. of Australasian Universities Power Engineering Conf. (AUPEC'2003)*, pp. 51-56, 2003.

[3] F. Islam, J. Kamruzzaman, and G. Lu, "Transformer Tap-Changer Control Using Artificial Neural Network for Cross Network Connection," *Proc. of the Third IASTED Int. Conf. on Power and Energy Systems (EuroPES 2003)*, pp. 666-671, 2003.

[4] R. Reed, "Pruning Algorithm-A Survey," *IEEE Transactions on Neural Networks*, vol. 5, pp. 740-746, 1993.

[5] Christopher M. Bishop, *Neural Networks for Pattern Recognition*. New York: Oxford University Press Inc., 1995.

[6] M. Lehtokangas, "Modeling with constructive backpropagation," *Neural Networks*, vol. 12, pp. 707-716, 1999.

[7] F. D. Foresee and M. T. Hagan, "Gauss-Newton Approximation to Bayesian Learning," *Proc. of the Int. Joint Conf. on Neural Networks*, pp. 1930-1935, 1997.

[8] M. Mooler, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Networks*, vol. 6, pp. 525-533, 1993.

[9] M. Lehtokangas, "Modified Cascade-Correlation Learning for Classification," *IEEE Transactions on Neural Networks*, vol. 11, pp. 795-798, 2000.

[10] S. E. Fahlman and C. Lebiere, "The cascade-Correlation Learning Architecture," *Advances in Neural Information Processing Systems 2*, pp. 524-532, 1990.

[11] S. German, E. Bienenstock, and R. Doursat, "Neural network and the bias /variance delimma," *Neural Computation*, vol. 4, pp. 1-58, 1992.

[12] G. Castellano and A. M. Fanelli, "An Iterative Pruning Algorithm for Feedforward Neural Network," *IEEE Transactions on Neural Networks*, vol. 8, pp. 519-531, 1997.

[13] M. Lehtokangas, "Modified Cascade-Correlation Learning for Classification," *IEEE Transactions on Neural Networks*, vol. 11, pp. 795-798, May 2000.

[14] J. Hwang, S. You, S. Lay, and I. Jou, "The cascade-correlation learning: A projection pursuit learning perspective," *IEEE Transactions on Neural Networks*, vol. 7, pp. 278-289, 1996.

[15] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, pp. 541-551, 1989.

[16] K. Fukushima, "A Neural Network for visual pattern recognition," *IEEE computer*, vol. 21, pp. 65-75, 1988.

[17] J. S. N. Jean and J. Wang, "Weight Smoothing to Improve Network Generalization," *IEEE Transactions on Neural Networks*, vol. 5, pp. 752-763, 1994.

[18] S. E. Fahlman, "Faster-Learning Variations on Back-propagation: An Empirical Study," *Proc. of the 1988 Connectionist Models Summer School*, 1988.

[19] D. Popovic, D. Kukolj, and F. Kulic, "Monitoring and assessment of voltage stability margins using artificial neural networks with a reduced input set," *IEE Generation Transmission and Distribution*, vol. 145, pp. 355-362, 1998.