SERBIAN JOURNAL OF ELECTRICAL ENGINEERING
Vol. 7, No. 2, November 2010, 269-289

UDK: 004.896:621.311.15

Static Security Based Available Transfer Capability (ATC) Computation for Real-Time Power Markets

Chintham Venkaiah[1,a)], Dulla Mallesham Vinod Kumar[1,b)]

Abstract: In power system deregulation, the Independent System Operator (ISO) has the responsibility to control the power transactions and avoid overloading of the transmission lines beyond their thermal limits. To achieve this, the ISO has to update in real-time periodically Available Transfer Capability (ATC) index for enabling market participants to reserve the transmission service. In this paper Static Security based ATC has been computed for real-time applications using three artificial intelligent methods viz.: i) Back Propagation Algorithm (BPA); ii) Radial Basis Function (RBF) Neural network; and iii) Adaptive Neuro Fuzzy Inference System (ANFIS). These three different intelligent methods are tested on IEEE 24-bus Reliability Test System (RTS) and 75-bus practical System for the base case and critical line outage cases for different transactions. The results are compared with the conventional full AC Load Flow method for different transactions.

Keywords: Available transfer capability, Intelligent techniques, Power system deregulation, Real-time power markets, Security analysis.

Introduction

The Available Transfer Capability (ATC) of a transmission network is the unutilized transfer capability of a transmission network for the transfer of power for further commercial activity, over and above already committed usage [1]. Power transactions between a specific seller bus/area and a buyer bus/area can be committed only when sufficient ATC is available. Thus such transfer capability can be used for reserving transmission services, scheduling firm and non-firm transactions and for arranging emergency transfers between seller bus/area and buyer bus/areas of an interconnected power system network.

Christie et. al. [2] reported that the US Federal Energy Regulatory Commission (FERC) began the federal deregulation process by requiring“open access” to transmission services, so that all companies owning generation would have equal opportunity to locate and obtain transmission services between their generation sites and their customers. The ATC values for the next hour and for each hour into the future would be placed on a website known as the Open

[1]Department of Electrical Engineering, National Institute of Technology, Warangal – 506004 Andhra Pradesh, India; E-mail: [a)]ch.venkaiah@gmail.com; [b)]vinodkumar.dm@gmail.com

Access Same-time Information System (OASIS), to be operated by Independent System Operator (ISO). Anyone wishing to send a power transaction on the ISO’s transmission system would access OASIS web pages and use the ATC information available there to determine if the transmission system could accommodate the transaction, and to reserve the necessary transmission service. Thus the ATC must be computed fast and accurately. Hamoud [3] described a method based on ATC concept for assessing the feasibility of simultaneous bilateral transaction and it utilized the Ontario Hydro’s Probabilistic Composite System Evaluation Program (PROCOSE) which employs DC Load flow to perform the analysis. Hamoud further [4] proposed a simple, efficient and practical method employing PROCOSE for determining the ATC between any two locations in the system and the ATC’s for selected transmission paths between them. Marija et. al. [5] discussed some theoretical aspects of ATC and the problems associated with its evaluation under open access. Viktor et. al. [6] included ATC in Optimal Transaction Management (OTM) method for remedial transactions curtailment and this method is found well suited for market-related analysis. Jayashree et. al. [7] proposed a unified optimization model and algorithm for assessing ATC and carrying out Congestion management using Unified Power Flow Controller (UPFC) in a deregulated Power Systems handling both pool and bilateral transactions. This method used DC Load flow model and repeated Linear Programming routine. The dc load-flow-based methods [2-7] are a bit faster than their ac counterparts but model only real power flow (in Mega Watts) in the lines rather than MVA, and assume the network to be loss free.

Ejebe et. al. [8] presented a detailed formulation and implementation of a fast program for ATC calculation based on the linear incremental power flow approximation. Fradi et. al. [9] presented a method to calculate energy transaction allocation factors for allocation of any nonlinear transmission system quantity to the active transactions placed on a transmission system. Ashwani and Srivastava [10] proposed a methodology based on AC Power Transfer Distribution Factors (ACPTDF) to allocate the active power loading in transmission lines. The methods based on power transfer /outage distribution factors [8-10] can cater to only the scenarios that are too close to the base case from which the factors are derived.

Jain et. al. [11] presented an approach based on RBF neural network to rank contingencies expected to cause steady state bus voltage violations. Ejebe et. al. [12] implemented a methodology developed for ranking transmission line outages and generator outages according to the severity of their effects on bus voltage or line flows. Wu [13] proposed a novel algorithm for contingency ATC computation and a sensitivity analysis for system uncertainties.

Luo et. al. [14] proposed a neural network solution methodology for the problem of real power transfer capability calculation. The Quick prop algorithm is utilized to train the neural network for estimating the transfer capability and the inputs to neural network are generator status, line status and load status. The artificial neural network (ANN) method [14] requires a large input vector so that it has to oversimplify determination of ATC by limiting it to a special case of power transfer to a single area from all of the remaining areas. So this method is unable to track down the bus-to-bus transactions, which is the true spirit of deregulation. The Adaptive Neuro Fuzzy method has a limitation with the universal index as all the line outage cases are considered for two categories leading to inaccurate ATC values in most of the line outage cases. Khairuddin et. al. [15] proposed a novel method with the full details for determining ATC in a large power system from only three input variables through fuzzy modeling. Khairuddin et. al. [16] introduced the concept of variable slack bus and the source bus is assigned to slack bus for ATC computation. The ATC is computed based on the highest possible increment of sink bus load above the base case. Here, ANN techniques have not been integrated with fuzzy systems for fast ATC computation.

In this paper to overcome the above limitations, to reduce the computational burden and to execute ATC in real time different Artificial Intelligence (AI) techniques viz., Back Propagation Algorithm (BPA), Radial Basis Function (RBF) Neural Network and Adaptive Neuro Fuzzy Inference System (ANFIS) have been utilized and compared with the AC Load flow based ATC. These methods are tested on standard IEEE 24-bus [17] Reliability Test System (RTS) and 75-bus [18] practical system, for base case and critical line outage cases, for different transactions.

In recent years, hybrid fuzzy neural networks have attracted considerable attention for their useful applications in such fields as control, pattern recognition, image processing, forecasting etc. In all these applications, there are different fuzzy neural network architectures proposed for different purposes and fields. The integrated system will possess the advantages of both neural networks (e.g. learning abilities, optimization abilities, and connectionist structures) and fuzzy systems (e.g humanlike IF-THEN rules thinking and ease of incorporating expert knowledge). In this way, one can bring the low-level learning and computational power of neural networks into fuzzy systems and also high level, humanlike IF-THEN rule thinking and reasoning of fuzzy systems into neural networks. Thus, on the neural side, more and more transparency is pursued and obtained either by pre-structuring a neural network to improve its performances or by a possible interpretation of the weight matrix following the learning stage. On the fuzzy side, the development of methods allowing automatic tuning of the parameters that characterize the fuzzy system can largely draw inspiration from similar methods used in the connectionist community. Thus, neural networks can improve their transparency, making them closer to fuzzy systems, while fuzzy systems can self adapt, making them closer to neural networks. Integrated systems can learn and adapt new associations, new patterns and new functional dependencies.

This paper is organized as follows. Section 2 details the problem formulation of ATC computation for real-time power markets. Section 3 gives an insight into Artificial Intelligent methods (BPA, RBF and ANFIS) application for ATC computation. Section 4 illustrates legibly the effectiveness of utilization of intelligent methods on standard IEEE 24-bus system and Practical Indian 75-bus system. The conclusions are presented in Section 5.

2 Problem Formulation

The ATC problem for real-time application has been attempted in two different ways i) Neural Network approach and ii) Adaptive Nero Fuzzy approach. For a given source-sink pair, tracing the least "indirect path" using line impedance data, identifies the neighboring bus. The one having the least impedance among all the possible indirect paths is chosen. If there are a number of buses on the chosen indirect path between a source and a sink then the bus immediately after the source is labeled as the neighboring bus. A new universal index (γ) has been proposed to represent a given operating condition of a power system taking into account demands at all the buses except the sink and neighboring bus. At the sink bus load is to be increased until it violates the thermal limit and the neighboring bus is a generator bus. Hence the loads are considered at these two buses.

The Universal index (γ) is defined as

$$\gamma = \sum_{\substack{i=1, \\ i \neq N_s, N_n}}^{N} P_{di} \, , \tag{1}$$

where P_{di} is demand (MW) at bus i, N is the total number of buses, N_s and N_n are sink and neighboring bus and $A_{\max}$ is the thermal load ability (MVA) of the line having the highest limit in the system.

The Performance Index (*PI*) for the purpose of contingency screening [11, 12] to identify the critical lines is defined as

$$PI^v = \sum_{i=1}^{N} \frac{\alpha_i}{2} \left[\frac{\Delta v_i}{\Delta v_i^{\lim}} \right]^2 , \tag{2}$$

where $\Delta v_i = v_i - v_i^{sp}$ and $\Delta v_i^{\lim} = \left(v_i^{\max} - v_i^{\min} \right) / 2$, v_i is post outage voltage magnitude at bus i; α_i is user defined constant (Generally taken as 1), and v_i^{sp} is specified voltage magnitude at bus *i*.

2.1 Input variables

The inputs to the neural network play a vital role to extract the features. Therefore to compute ATC between a given pair of source-sink buses in a large system, only three inputs are considered to a neural network for base case. These are sink bus injection (P_s), the neighboring bus injection (P_n) and the universal index (γ) for the base case. Binary inputs are used to represent critical line outages in addition to the base case inputs. The sink and neighboring bus injections are the differences between respective local generation and demand in MW.

2.1.1 Neural network approach

Apart from three inputs viz. the sink bus injection (P_s), the neighboring bus injection (P_n) and the universal index (γ), the critical line outages are represented by binary inputs that represent for each critical line outage condition. For example, two input binary variables can represent four conditions:

0 0 – normal operating condition (Base case);

1 0 – critical line-2 outage;

0 1 – critical line-1 outage;

1 1 – critical line-3 outage.

Similarly to represent number of line outages (NL) we need only maximum of $\log_2(NL+1)$ inputs. Moreover by considering only critical line outages the number of inputs will be decreased.

2.1.2 Adaptive Neuro Fuzzy Inference System (ANFIS) approach

In Adaptive Neuro Fuzzy Inference System approach in addition to the three base case inputs, Category Index (C) is used to represent various critical line outages. Thus total inputs considered here are the sink bus injection (P_s), the neighboring bus injection (P_n), the universal index (γ) and the Category Index (C). The total number of inputs to the Adaptive Neuro Fuzzy Inference system including critical line outages is reduced to four. Compared to the neural network critical line outage representations, only one input category index (C) is required to represent critical line outages in the Adaptive Neuro Fuzzy Inference System viz.

C = 1 for normal operating condition (Base case);

C = 3 for critical line-2 outage;

C = 2 for critical line-1 outage;

C = 4 for critical line-3 outage.

As the number of inputs to the ANFIS are reduced to four compared to Neural Network approach (inputs are five considering critical line outages), the computational time will be reduced drastically.

3 Artificial Intelligent (AI) Models

3.1 Back Propagation Algorithm (BPA)

A schematic diagram of the topology of BPA is shown in Fig. 1. This network consists of a set of n input neurons, m output neurons and one hidden layer of k intermediate neurons. Data flows into the network through the input layer, passes through the hidden layer and finally flows out of the network through output layer. The network thus has a simple interpretation as a form of input-output model, with network weights as free parameters. Such networks [19] can model functions of almost any arbitrary complexity, with the number of layer and number of neurons in each layer, determining the function complexity.

In Fig. 1 the input signal X_i ($i = 1, \ldots, n$) are multiplied by the weights W_{ij}; then operated on by the activation function $f(x)$ to produce the b_j of the hidden layer. Similar operations can be made on outputs of the network. Here

$$b_j = f\left(\sum_{i=1}^{n} X_i W_{ij}\right), \tag{3}$$

where ‘ f ‘ is a transfer function of activation function, which can take the form of non-linear function. For the non linear sigmoid function

$$f(x) = \left(1 + \mathrm{e}^{-x}\right)^{-1}. \tag{4}$$

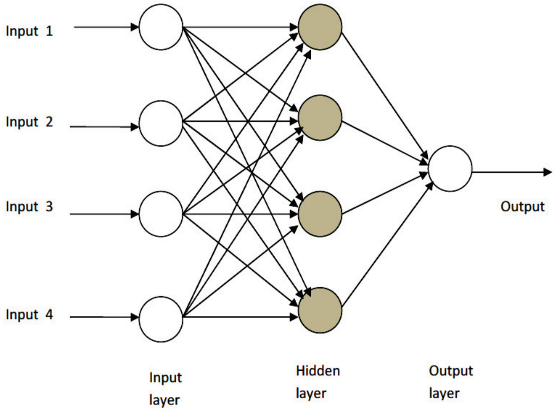

Fig. 1 – *Topology of a three layered* MLP.

Training is a procedure used to minimize the difference between outputs of Multi-layer Perceptron (MLP) and the desired values by adjusting the weights of the network. Sets of input vectors are presented to the network until training is completed. Once the network is trained the new input data is presented to the network to determine the output.

3.2 Radial Basis Function (RBF) Neural Network

A potential advantage of Radial Basis Function Network (RBF) is its ability to augment new training data without the need for retraining. RBF has only one nonlinear hidden layer and linear output layer. During training, all of the input variables are fed to hidden layer directly without any weight and only the weights between hidden and output layers have to be modified using error signal. Thus, it requires less training time in comparison to BPA model.

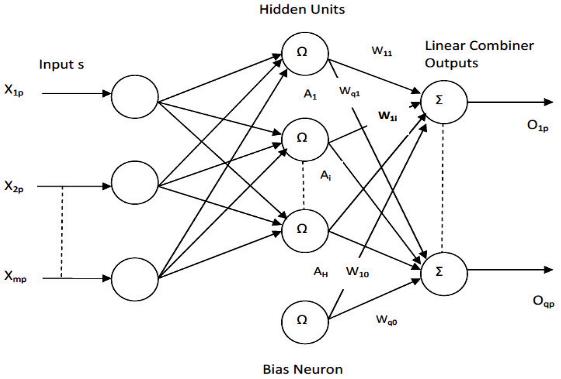

Fig. 2 – *Radial Basis Function Network model.*

The RBF Neural Network is shown in Fig. 2. The RBF network [11] hidden layer has non-linear Gaussian function, which is defined by a center position and a width parameter. The width of the RBF unit controls the rate of decrease of function. The output of the i^{th} unit $a_i\,(x_p)$ in the hidden layer is given by

$$a_i\left(x_p\right) = \exp\left(-\sum_{j=1}^{r}\frac{\left|x_{jp}-\overline{x}_{ji}\right|^2}{\Psi_i^2}\right), \qquad (5)$$

where $\overline{x}_{ji}$ is centre of ith RBF unit for input variable j, Ψ_i is width of i^{th} RBF unit, x_{jp} is j^{th} variable of input pattern p and r is dimension of input vector.

The connection between the hidden units and the output units are weighted sums. The output value O_{qp} of the q^{th} output node for p^{th} incoming pattern is given as

$$O_{qp} = \sum_{i=1}^{H} w_{qi} a_i \left(X_p \right) + w_{qo}\,, \tag{6}$$

where w_{qi} is weight between i th RBF unit and q th output node, w_{qo} is biasing term at q th output node and H is number of hidden layer (RBF) nodes.

The parameters of the RBF units are determined in three steps of the training activity. First, the unit centers are determined by some form of clustering algorithm. Then the widths are determined by a nearest neighbor method. Finally, weights connecting the RBF units and the output units are calculated using delta rule.

3.3 Adaptive Neuro Fuzzy Inference System (ANFIS*)*

The fuzzy logic has two main advantages. The way fuzzy logic tackles the dimensionality of a problem is computationally more efficient than that by other artificial intelligence (AI) techniques (such as ANN, expert system, etc.). Another advantage is that fuzzy logic can capture uncertainties inherent in an incomplete or reduced set of data. It is noteworthy that rigorous mathematics intensive conventional methods have none of these two advantages.

3.3.1 Fuzzification of Inputs

Each of the inputs is converted from a single crisp value into a maximum of two fuzzy values using the widely used triangular functions that may overlap with one another as shown in Fig. 3. The x-axis in Fig. 3 represents the crisp values of i th input (I_i) while the y-axis shows "membership grade" (μ_i) that may vary from 0.0 to 1.0. Each triangle has a fuzzy attribute that can be coded by a linguistic variable (e.g., "low") or a number implying level of fuzziness (e.g., 1). However, for the sake of mathematical representation, a number is used. The total number of such attributes or triangles for i th input is denoted by m_i. The x coordinates of three vertices of each triangle are respectively a_{ij}, c_{ij} and b_{ij}, when $j = 1, 2, \ldots, m_i$. Equation (7) shows crisp (I_i) to fuzzy (I_i^f) conversion for i th input [15]

$$\begin{aligned} & I_i^f = \{1\}, I_i \le c_{i1}; I_i^f = \{m_i\}, I_i > c_{im_i}\,, \\ & I_i^f = \left\{ (1,2), (2,3), \ldots \left(m_{i-1,} m_i \right) \right\}, c_{i1} < I_i \le c_{im_i}\,, \end{aligned} \tag{7}$$

where $i = 1, 2, 3, 4$ (i.e., for ATC determination), I_1, I_2, I_3 and I_4 are P_s, P_n, γ and C, respectively.

The membership grade (μ_i) corresponding to each fuzzy value of given crisp input can be obtained using (8):

$$\begin{aligned}\left(\mu_i\right)_j &= \frac{I_i - a_{ij}}{c_{ij} - a_{ij}}; a_{ij} \le I_i \le c_{ij}, j \in I_i^f,\\ \left(\mu_i\right)_j &= \frac{I_i - b_{ij}}{b_{ij} - c_{ij}}; c_{ij} \le I_i \le b_{ij}, j \in I_i^f,\end{aligned} \tag{8}$$

where j implies the numbers picked up by the i th input's fuzzy value (I_i^f) as in (7).

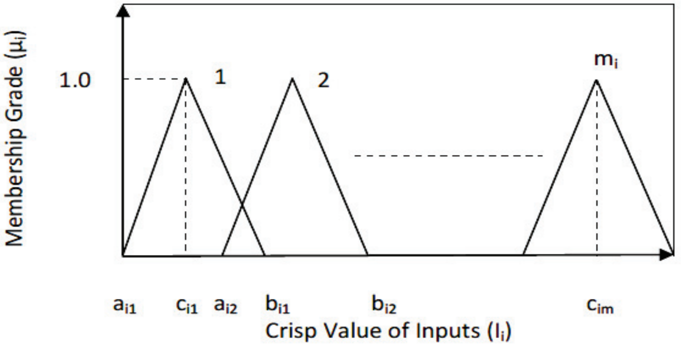

Fig. 3 – *Triangular membership function for* i^{th} *input.*

3.3.2 Inference on ATC

The rule-base relating ATC to the inputs for a large system is developed using Sugeno fuzzy model. A set of first-order polynomial equations is used to infer a crisp value of ATC from crisp values of four inputs. It should be noted that a given set of crisp values for the four inputs will not fire all of the $\prod_{i=1}^{4} m_i$ rules rather q number of rules when $1 \le q \le 2^4$ (i.e., one to sixteen rules). This is because, as shown in (7), each input's crisp value has a maximum of two fuzzy values. The required overall crisp value ATC is obtained as in (9) that uses weighted average of the individual crisp outputs from each of the fired rules, that is ATC_o

$$\text{AT}'\text{C}' = \frac{\sum_{\text{o}\in q}\left(\mu_\text{o}\text{ATC}_\text{o}\right)}{\sum_{\text{o}\in q}\mu_\text{o}}, \tag{9}$$

where "o" implies each of the fired q rules, and μ_o is as in (10):

$$\mu_o = \prod_{i=1}^{4} \mu_i \text{ ,} \tag{10}$$

where μ_1, μ_2, μ_3, μ_4 are the membership grades calculated using (8) respectively, for the four input fuzzy values (i.e. I_1^f, I_2^f, I_3^f and I_4^f).

4 Simulation Results

4.1 ATC for bilateral transactions on IEEE 24-bus RTS

The IEEE 24-bus RTS [17] has been used to compare the performance of proposed Neural Networks & ANFIS methods with that of full AC load flow-based ATC determination. The pair of buses 23 (source) and 16 (sink) is considered for illustrating the determination of ATC. The path 23-13-11-14-16 has been identified as the one having the least impedance path among all of the indirect paths that connect 16 to 23. This has led to selection of bus 13 as the neighbor to this source–sink.

4.1.1 Generation of patterns

The Training and Testing patterns are generated using load-flow, treating bus 23 as slack, 16 and 13 both as PV (i.e., bus with specified real power and voltage) buses. The other bus types were retained as what those should be in a normal load flow. The load at sink bus (No. 16) was incremented in steps of 10 MW to repeat the load flow until thermal limit is exceeded in any line of the test system. The maximum possible increment achieved above base-case load at the sink bus was the ATC for the corresponding case.

4.1.2 Training

Training sets provided to the neural network are representative of the whole state space of concern so that the trained system has the ability of generalization. Training patterns for the IEEE 24-bus RTS are composed of: Load levels of 50%, 75%, and 100% of base case while all lines in operation with different Sink bus injection. Contingency ranking is done on this system. It is found that the lines 7, 18 and 37 are the first three critical lines for the IEEE 24-bus RTS. Single Line outage of these lines at 50%, 75%, and 100% of base load with different Sink bus injection are considered for the pattern generation. Total 240 patterns are generated randomly, Out of which 180 patterns are used for the training and the remaining novel 60 patterns which are not the part of training pattern are used for the testing considering base case as well as the critical outage cases. There are 180 training patterns in total covering the base case and three critical line outage cases are considered.

4.1.3 Testing

The trained neural network and ANFIS was tested using 60 patterns, which are composed of 30 load variation cases and 30 critical line outage cases with different sink bus injections. None of these 60 patterns were used in the training of the neural network.

4.2 ATC for bilateral transactions on 75-bus practical system

The 75-bus practical system [18] has been used to compare the performance of proposed Neural Networks & ANFIS methods with that of full AC load flow-based ATC determination. The pair of buses 14 (source) and 5 (sink) is considered for illustrating the determination of ATC. As there is no direct path between the source bus and sink bus one of the effective generator buses connected to the indirect path between buses 14 and 5 is taken as the neighboring bus. So generator bus 6 is taken as the neighboring bus.

4.2.1 Generation of patterns

The load at sink bus (No. 5) was varied in steps of 5 MW to repeat the load flow until thermal limit is exceeded in any line of the system. The maximum possible increment achieved above base case load at the sink bus was the ATC for the corresponding case.

4.2.2 Training

Training sets provided to the neural network are representative of the whole state space of concern so that the trained system has the ability of generalization. Training patterns for the 75-bus system are composed of: Load levels of 25%, 50% and 75% of base case while all lines in operation with different Sink bus injection. Contingency ranking is done on this system. It is found that the lines 25, 22, 19 are the first three critical lines for the 75-bus system. Single Line outage of these lines at 25%, 50% and 75% of base load with different Sink bus injection are considered for the pattern generation. Total 300 patterns are generated randomly, Out of which 210 patterns are used for the training and the remaining novel 90 patterns which are not the part of training pattern are used for the testing considering base case as well as the critical outage cases. There are 210 training patterns in total covering the base case and three critical line outage cases are considered.

4.2.3 Testing

The trained neural network and ANFIS was tested using 90 patterns, which are composed of different loading cases and different line contingency cases with different sink bus injections. None of these 90 patterns were used in the training of the neural network.

4.3 Back Propagation Algorithm (BPA) for IEEE 24-bus RTS

4.3.1 Input layer

The input layer consists of five neurons to give inputs Sink bus injection (P_s), Neighboring bus injection (P_n) and Universal Index (γ) and 2 binary inputs are selected to represent four cases as below.

0 0 – for Base case;

1 0 – for critical Line-18 outage;

0 1 – for critical Line-7 outage;

1 1 – for critical Line-37 outage.

4.3.2 Output layer

The output layer has only one neuron whose output is the ATC from bus 23 to bus 16.

4.3.3 Hidden layer

The neural network with one hidden layer with 9 neurons has been considered by hit and trial, which has provided minimum error. Fig. 4 shows graphically the BPA based ATC as compared to exact values of ATC as determined from AC load flow based calculation [16] for IEEE 24-bus RTS.

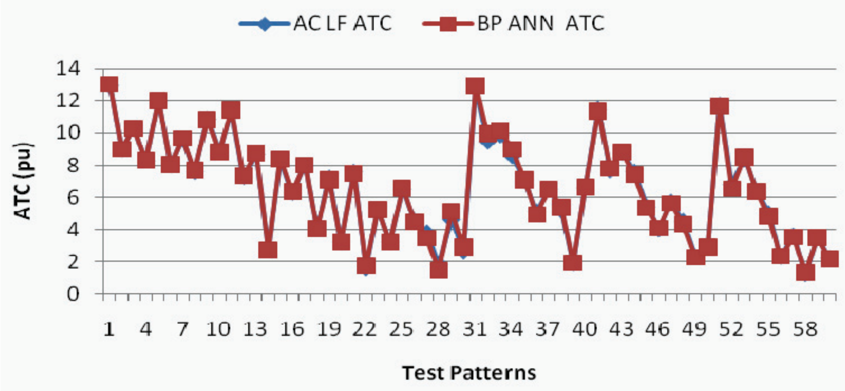

Fig. 4 – IEEE 24-*bus* RTS *comparison of* BPA *Neural Network* ATC *and* AC LF *based* ATC.

4.4. Back Propagation Algorithm (BPA) for 75-bus practical system

4.4.1 Input Layer

The input layer consists of five neurons to give inputs Sink bus injection (P_s), Neighboring bus injection (P_n) and Universal Index (γ) and 2 binary inputs are selected to represent four cases as below.

0 0 – for Base case;

1 0 – for critical Line-22 outage;

0 1 – for critical Line-25 outage;

1 1 – for critical Line-19 outage.

4.4.2 Output Layer

The output layer has only one neuron whose output is the ATC from bus 14 to bus 5.

4.4.3 Hidden Layer

The neural network with one hidden layer with 9 neurons has been considered by hit and trial, which has provided minimum error. Fig. 5 shows graphically the BPA based ATC as compared to exact values of ATC as determined from AC load flow based calculation [16] for 75-bus practical system.

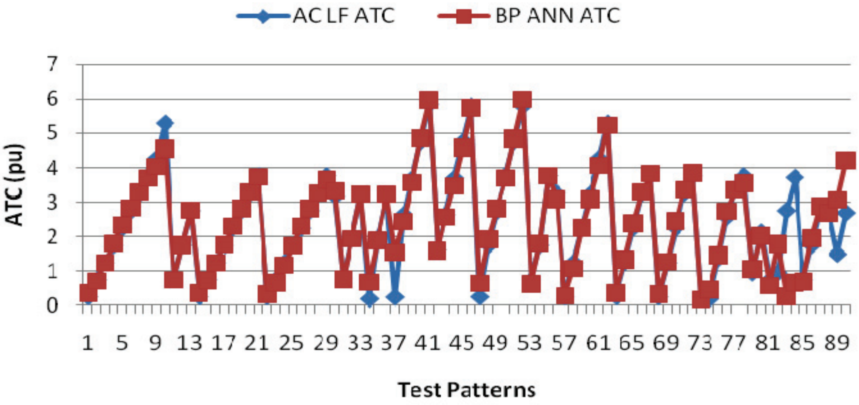

Fig. 5 – 75-*bus practical system comparison of* BPA *Neural Network* ATC *and* AC LF *based* ATC.

4.5 Radial Basis Function Neural Network (RBFN) for IEEE 24-bus RTS & 75-bus practical system

To demonstrate the effectiveness of the proposed RBF model, it has been trained and tested with the patterns generated as discussed in Sections 4.1 and 4.2. The RBF model used here has same 5 neurons in the input layer, 1 neuron in the output layer as utilized for BPA. The number of hidden neurons selected as 75 with Gaussian density function. Euclidean distance-based clustering [11] technique has been employed in this paper to select the number of hidden (RBF) units and unit centers. The normalized input and output data are used for training of the RBF neural network. The optimal learning is achieved at the global minimum of testing error. It was observed that the training in this case was faster and also its performance was better as compared to the BPA model.

The training of RBF neural network requires less computation time as compared to the BPA model, since only the second layer weights have to be calculated using error signal. The training of RBF network has been made still faster by applying adaptive learning rate and momentum.

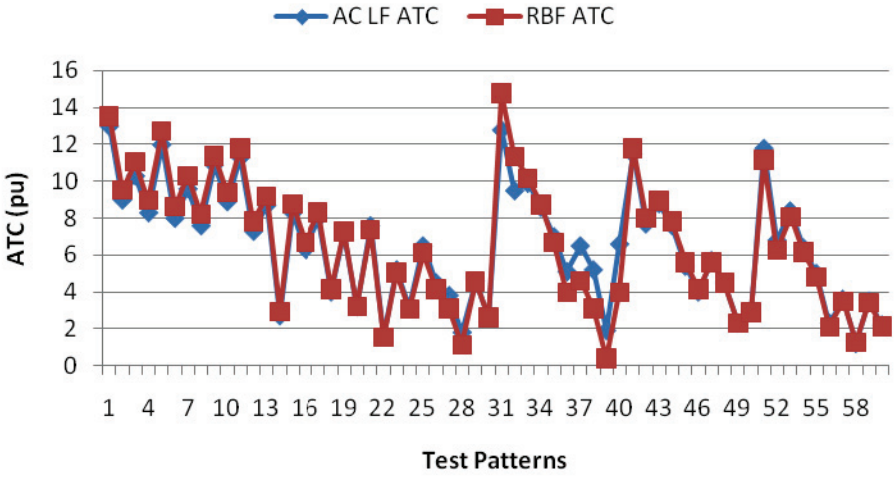

Fig. 6 – IEEE 24-*bus* RTS *comparison of* RBF ATC *and* AC LF *based* ATC.

Figs. 6 and 7 shows graphically the RBF neural network estimates for ATC as compared to exact values of ATC as determined from AC load flow method, for the IEEE 24-bus RTS and 75-bus practical system respectively.

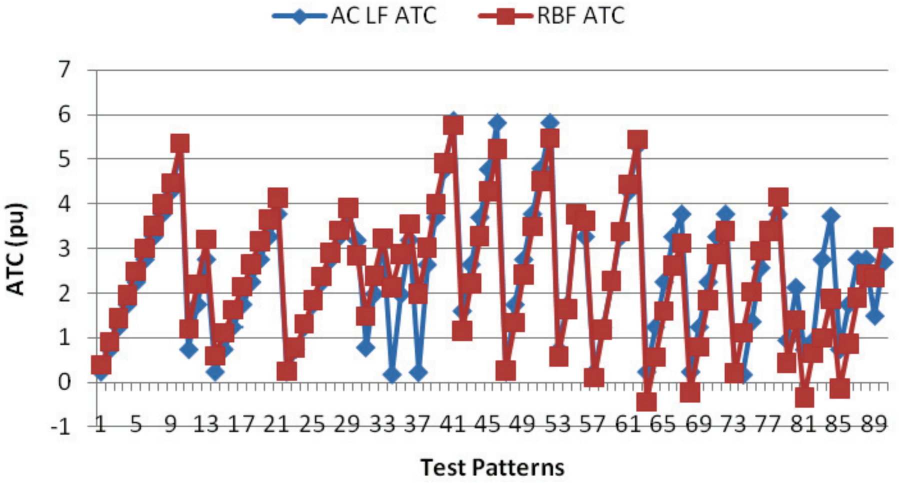

Fig. 7 – 75-*bus practical system comparison of* RBF ATC *and* AC LF *based* ATC.

4.6 Adaptive Neuro Fuzzy Inference System (ANFIS) for IEEE 24-bus RTS & 75-bus practical system

ATC between a given pair of source-sink buses in a large system is determined using the same inputs as given in BPA and RBF methods, except instead of taking binary input variables for critical line outage conditions, a single variable is taken and it is given a separate integer value to distinct each outage case. The inputs thus become Sink bus injection (P_s), neighboring bus injection (P_n), Universal Index (γ) and category Index(C).

The C value has been specified for the IEEE 24-bus RTS is as follows:

C=1 for Base case;

C=3 for critical line-18 outage;

C=2 for critical line-7 outage;

C=4 for critical line-37 outage.

The C value has been specified for the 75-bus practical system is as follows:

C=1 for Base case;

C=3 for critical line-22 outage;

C=2 for critical line-25 outage;

C=4 for critical line-19 outage.

These four inputs are fuzzified and ATC has been calculated. The numbers of fuzzy sets (attributes) chosen are respectively 3, 5, 3 and 4 for P_s, P_n, γ and C. The linguistic attributes corresponding to three levels are low, medium, and high respectively. Since the neighboring bus may also have generation in excess of its local load, its membership levels are five implying negative high, negative low, zero, positive low, and positive high, respectively. For training by ANFIS, the MATLAB Fuzzy Toolbox [20] was used. Fig. 8 shows graphically the ANFIS estimates of the ATC as compared to exact values as determined from AC load flow based calculation for IEEE 24-bus Reliability Test system.

The ATC values calculated for different test cases by the three methods are given in **Table 1** for Base case and line outage cases along with the AC Load Flow based ATC values. Out of 60 test patterns the first 30 patterns presented in **Table 1** correspond to normal operating condition and the remaining 30 patterns in **Table 1** correspond to critical line outages with 10 patterns for each line.

Fig. 9 shows the comparisons of ANFIS based ATC and AC LF based ATC results for the 75-bus practical system.

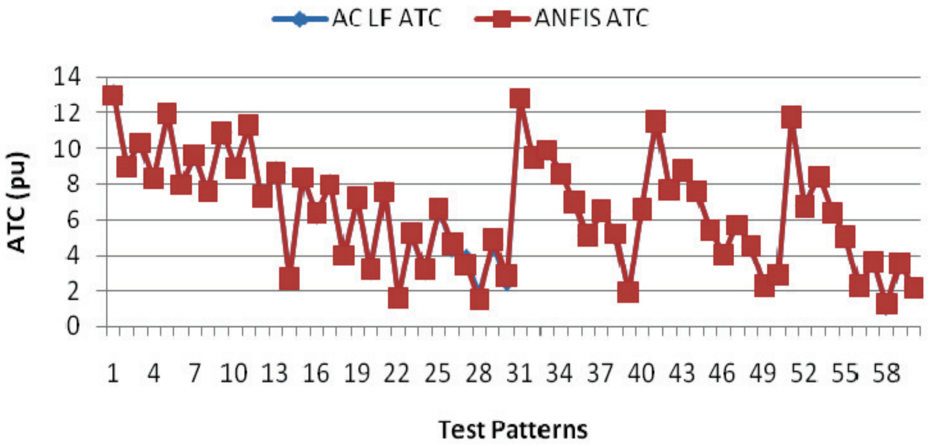

Fig. 8 – IEEE 24-*bus* RTS *comparison of* ANFIS ATC *and* ACLF ATC.

Table 1

ATC *between bus* 23 *and bus* 16 *for* IEEE 24-*bus* RTS (*Base Case* & *Critical Line Outages*).

Test Patterns	**AC LF ATC(pu)**	**BPA ATC(pu)**	**RBF ATC(pu)**	**ANFIS ATC(pu)**	**Test Patterns**	**AC LF ATC(pu)**	**BPA ATC(pu)**	**RBF ATC(pu)**	**ANFIS ATC(pu)**
1	13.00001	13.0530	13.506985	12.9700	31	12.80001	12.9640	14.775820	12.8010
2	09.00000	08.9914	09.518487	08.9700	32	09.50000	09.9520	11.346738	09.5025
3	10.30000	10.2810	11.043727	10.3090	33	09.90000	10.1410	10.166296	09.9001
4	08.30000	08.2917	08.965892	08.3094	34	08.60000	08.9781	08.736770	08.6016
5	12.00001	12.0390	12.720891	11.9750	35	07.00000	07.1053	06.681277	07.0000
6	07.99999	08.0358	08.613604	07.9748	36	05.10000	04.9325	03.966622	05.0999
7	09.60000	09.6559	10.296741	09.6234	37	06.50000	06.4989	04.601788	06.4999
8	07.60000	07.6841	08.222308	07.6237	38	05.20000	05.3608	03.105386	05.2077
9	10.90001	10.8090	11.352566	10.9130	39	01.90000	01.9331	00.414151	01.9096
10	08.90000	08.8253	09.379425	08.9124	40	06.60000	06.6140	03.990614	06.5997
11	11.30001	11.4150	11.786911	11.3150	41	11.50001	11.3400	11.803089	11.5010
12	07.30000	07.3441	07.835326	07.3142	42	07.70000	07.8051	08.008680	07.7019
13	08.70000	08.7402	09.192936	08.6707	43	08.80000	08.8100	08.951054	08.8002
14	02.70000	02.6969	02.946276	02.6687	44	07.60000	07.4440	07.837973	07.6012
15	08.30000	08.3976	08.770634	08.3445	45	05.40000	05.3226	05.604135	05.4016
16	06.30000	06.3528	06.688462	06.3451	46	04.00000	04.0617	04.146664	04.0020
17	07.99999	07.9974	08.302246	07.9698	47	05.70000	05.5940	05.592674	05.6999
18	04.00000	04.0289	04.134586	03.9714	48	04.50000	04.3272	04.477858	04.5004
19	07.20000	07.1004	07.273947	07.2113	49	02.30000	02.2584	02.302838	02.3005
20	03.20000	03.2284	03.222071	03.2188	50	02.90000	02.8876	02.897452	02.8998
21	07.60000	07.4950	07.373643	07.5562	51	11.80001	11.6940	11.165541	11.7680
22	01.60000	01.7300	01.535939	01.5938	52	06.80000	06.5329	06.266925	06.7367
23	05.20000	05.2348	05.062666	05.2498	53	08.40000	08.4842	08.085509	08.4004
24	03.20000	03.1999	03.068753	03.2468	54	06.40000	06.3507	06.172842	06.4003
25	06.50000	06.5503	06.115197	06.6203	55	05.00000	04.8195	04.810942	05.0366
26	04.50000	04.4658	04.138824	04.6191	56	02.30000	02.3278	02.116342	02.3004
27	03.80000	03.4563	03.128177	03.4477	57	03.60000	03.5195	03.495235	03.5999
28	01.80000	01.4665	01.144288	01.5144	58	01.20000	01.2906	01.269242	01.2505
29	04.60000	05.1265	04.550518	04.8551	59	03.50000	03.5012	03.433395	03.5455
30	02.60000	02.9055	02.621739	02.8556	60	02.20000	02.1620	02.135205	02.1744

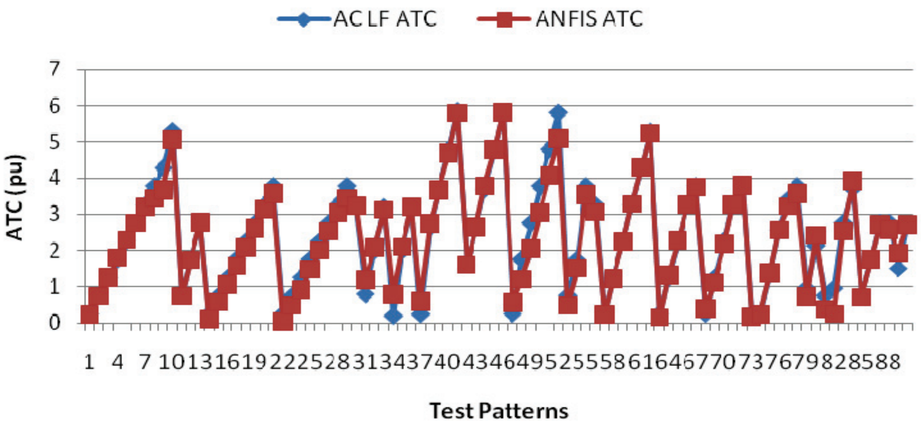

Fig. 9 – 75-*bus practical System comparison of* ANFIS ATC *and* ACLF ATC.

Test Patterns: -31-40: critical line-7 outage; 41-50: critical line-18 outage; 51-60: critical line-37 outage.

The ATC values computed for different test cases on 75-bus practical system by the three methods are given in **Table 2** for base case and line outage cases along with the AC load flow based ATC values.

Table 2
ATC *between bus* 14 *and bus* 5 *for Practical Indian* 75-*bus system*
(*Base Case* & *Critical Line Outages*).

Test Patterns	AC LF ATC(pu)	BPA ATC(pu)	RBF ATC(pu)	ANFIS ATC(pu)	Test Patterns	AC LF ATC(pu)	BPA ATC(pu)	RBF ATC(pu)	ANFIS ATC(pu)
1	0.2411	0.33897	0.38740	0.2350	46	5.8148	5.75140	5.22970	5.8212
2	0.7423	0.70198	0.90887	0.7420	47	0.2397	0.62539	0.26216	0.5690
3	1.2447	1.23360	1.43460	1.2500	48	1.7472	1.91570	1.33350	1.2100
4	1.7482	1.79230	1.96090	1.8000	49	2.7577	2.81460	2.41820	2.0600
5	2.2528	2.33060	2.48410	2.3000	50	3.7723	3.69510	3.48770	3.0500
6	2.7584	2.83490	3.00060	2.7600	51	4.7913	4.85030	4.51410	4.0700
7	3.2650	3.29130	3.50660	3.2000	52	5.8153	5.99420	5.47100	5.0900
8	3.7727	3.69000	3.99870	3.4600	53	0.7423	0.61960	0.57455	0.4790
9	4.2816	4.02800	4.47370	3.6800	54	1.7482	1.79890	1.63770	1.5000
10	5.3030	4.53610	5.35950	5.0600	55	3.7727	3.75970	3.76460	3.5400
11	0.7426	0.73239	1.18980	0.7390	56	3.2645	3.08530	3.62050	3.0600
12	1.7483	1.74520	2.20390	1.7300	57	0.2382	0.26147	0.09906	0.2260
13	2.7584	2.76800	3.20480	2.7700	58	1.2422	1.06860	1.17310	1.2300
14	0.2416	0.35301	0.59489	0.1060	59	2.2510	2.25540	2.27360	2.2500
15	0.7427	0.71782	1.10700	0.5830	60	3.2639	3.08690	3.37210	3.2900

Table 2 (continuation)

ATC *between bus* 14 *and bus* 5 *for Practical Indian* 75-*bus system*
(*Base Case* & *Critical Line Outages*).

Test Patterns	AC LF ATC(pu)	BPA ATC(pu)	RBF ATC(pu)	ANFIS ATC(pu)	Test Patterns	AC LF ATC(pu)	BPA ATC(pu)	RBF ATC(pu)	ANFIS ATC(pu)
16	1.2450	1.23040	1.62320	1.0600	61	4.2809	4.05700	4.43930	4.2900
17	1.7484	1.76590	2.13990	1.5700	62	5.3024	5.22910	5.44740	5.2400
18	2.2529	2.29420	2.65350	2.0900	63	0.2416	0.35506	-0.43767	0.1470
19	2.7585	2.80750	3.16030	2.6200	64	1.2451	1.30640	0.56736	1.3300
20	3.2651	3.28990	3.65680	3.1400	65	2.2530	2.38500	1.59390	2.2800
21	3.7729	3.72510	4.13960	3.5800	66	3.2651	3.29430	2.61440	3.2700
22	0.2411	0.30981	0.24812	0.0347	67	3.7728	3.82460	3.11380	3.7500
23	0.7423	0.64276	0.77496	0.4850	68	0.2401	0.30988	-0.22649	0.3990
24	1.2447	1.16170	1.30690	0.9350	69	1.2439	1.27040	0.79634	1.1100
25	1.7482	1.73270	1.84030	1.4700	70	2.2522	2.44490	1.84170	2.1900
26	2.2528	2.29020	2.37130	2.0000	71	3.2647	3.36490	2.88150	3.2800
27	2.7584	2.80560	2.89610	2.5300	72	3.7725	3.85550	3.39050	3.800
28	3.2650	3.26070	3.41110	3.0500	73	0.2344	0.16712	0.19830	0.1660
29	3.7728	3.64760	3.91280	3.4500	74	0.1765	0.45422	1.10420	0.2340
30	3.1842	3.33460	2.84990	3.2400	75	1.3636	1.45440	2.02960	1.3900
31	0.7857	0.74220	1.48500	1.1900	76	2.5709	2.72570	2.94970	2.5700
32	1.9840	1.93640	2.38060	2.0900	77	3.4270	3.35960	3.39990	3.2300
33	3.2013	3.22460	3.23010	3.1200	78	3.7728	3.54230	4.15170	3.5800
34	0.1822	0.67127	2.13010	0.8040	79	0.9462	1.04280	0.42852	0.7300
35	1.9734	1.88960	2.87790	2.1000	80	2.1323	2.01920	1.39140	2.4300
36	3.1928	3.23580	3.55220	3.2000	81	0.7399	0.56256	-0.33536	0.3850
37	0.2313	1.53120	1.99220	0.6000	82	0.9462	1.77970	0.65186	0.2500
38	2.6388	2.43360	3.01420	2.7400	83	2.7570	0.24033	0.99235	2.5300
39	3.6971	3.56120	3.99890	3.6700	84	3.7230	0.64390	1.86470	3.9400
40	4.7707	4.86550	4.92100	4.7000	85	0.7418	0.69017	-0.13852	0.7100
41	5.8619	5.97250	5.75770	5.8100	86	1.7478	1.94250	0.87236	1.7470
42	1.6007	1.58650	1.15090	1.6018	87	2.7580	2.87070	1.89310	2.7100
43	2.6436	2.56550	2.21710	2.6452	88	2.7587	2.68260	2.41150	2.5800
44	3.7011	3.46460	3.27050	3.7920	89	1.4929	3.06300	2.34200	1.9400
45	4.7732	4.58640	4.28340	4.7927	90	2.6941	4.21620	3.25520	2.7100

Test Patterns: -20-45: critical line-25 outage; 46-65: critical line-22 outage; 66-90: critical line-19 outage.

The training and testing times of the intelligent techniques viz. BPA, RBF and ANFIS have been compared in terms of CPU time (in seconds) for computing ATC for both the systems are as shown in **Table 3**.

Table 3
Comparison of CPU *Time* (*in seconds*).

Test System	BPA	RBF	ANFIS
Training			
IEEE 24-bus RTS	15.483	13.235	10.210
75-bus Practical System	16.950	14.262	10.853
Testing			
IEEE 24-bus RTS	0.0264	0.0258	0.0192
75-bus Practical System	0.0288	0.0252	0.0186

It is found from **Table 3** that all the proposed intelligent techniques took very less time to compute ATC. The simulation was carried out in Pentium® 4 CPU, 3.00 GHz, 496 MB of RAM Personal Computer.

5 Conclusion

In this paper to make use ATC calculations in real time, Artificial Intelligent methods viz.:

i) Back Propagation Algorithm,

ii) Radial Basis Function Neural Networks, and

iii) Adaptive Neuro Fuzzy Inference System

are utilized and compared with the Full AC Load Flow method. To compute ATC between source and sink three inputs are considered i) Sink bus injection (P_s), ii) Neighboring bus injection (P_n) and iii) Universal index (γ). Whereas for the critical line outage cases apart from these three inputs two more additional inputs are considered for the Back Propagation Algorithm (BPA) and Radial Basis Function Neural network (RBF) whereas only one additional input is considered for the Adaptive Neuro Fuzzy Inference System (ANFIS) to identify a particular critical line outage. The proposed method has been tested on IEEE 24-bus Reliability Test System and 75-bus practical System.

The mean absolute error for base case and critical line outage case utilizing BPA neural network were found to be 0.09478 pu and 0.1182 pu respectively for IEEE 24-bus RTS and the corresponding values for 75-bus practical system are 0.08918 pu and 0.25563 pu respectively. For the Radial Basis Function (RBF) Neural network, the mean absolute error for base case and critical line outage case were found to be 0.3959 pu and 0.58798 pu respectively for IEEE 24-bus RTS and the corresponding values for 75-bus practical system are 0.29678 pu and 0.46106 pu respectively. Whereas for the Adaptive Neuro Fuzzy Inference System (ANFIS), the mean absolute error for base case and critical line outage case were found to be 0.0667 pu and 0.009527 pu respectively for IEEE 24-bus RTS and the corresponding values for 75-bus practical system are 0.12267 pu and 0.18739 pu respectively.

The CPU time requirement of the ANFIS method is independent of the system size and also it requires only four inputs irrespective of size of the system. The number of rules and parameters related to fuzzy model are independent of the system size. Hence the Adaptive Neuro Fuzzy Inference System (ANFIS) method can be used on larger systems for real-time power markets.

6 References

[1] Available Transfer Capability Definitions and Determination, North American Electric Reliability Council, June 1996, *http://www.nerc.com*.

[2] R.D. Christie, B.F. Wollenberg, I. Wangesteen: Transmission Management in the Deregulated Environment, Proceedings of the IEEE, Vol. 88, No. 2, Feb. 2000, pp.170 – 195.

[3] G. Hamoud: Feasibility Assessment of Simultaneous Bilateral Transactions in a Deregulated Environment, IEEE Transactions on Power Systems, Vol. 15, No. 1, Fe. 2000, pp. 22 – 26.

[4] G. Hamoud: Assessment of Available Transfer Capability of Transmission Systems, IEEE Transactions on Power Systems, Vol. 15, No. 1, Feb. 2000, pp. 27 – 32.

[5] M.D. Ilic, Y.T. Yoon, A. Zobian: Available Transmission Capacity (ATC) and its Value under Open Access, IEEE Transactions on Power Systems, Vol. 12, No. 2, May 1997, pp. 636 – 645.

[6] V. Maksimovic, I. Skokljev: Remedial Transactions Curtailment via Optimization, Serbian Journal of Electrical Engineering, Vol. 6, No. 1, May 2009, pp. 149 – 166.

[7] R. Jayashree, M.A. Khan: A Unified Optimization Approach for the Enhancement of Available Transfer Capability and Congestion Management using Unified Power Flow Controller, Serbian Journal of Electrical Engineering, Vol. 5, No. 2, Nov. 2008, pp. 305 – 324.

[8] G.C. Ejebe, J.G. Waight, M. Santos-Nieto, W.F. Tinney: Fast Calculation of Linear Available Transfer Capability, IEEE Transactions on Power Systems, Vol. 15, No. 3, Aug. 2000, pp. 1112 – 1116.

[9] A. Fradi, S. Brignone, B.F. Wollenberg: Calculation of Energy Transaction Allocation Factors, IEEE Transactions on Power Systems, Vol. 16, No. 2, May 2001, pp. 266 – 272.

[10] A. Kumar, S.C. Srivastava: AC Power Transfer Distribution Factors for Allocating Power Transactions in a Deregulated Market, IEEE Power Engineering Review, Vol. 22, No. 7, July 2002, pp. 42 – 43.

[11] T. Jain, L. Srivastava, S.N. Singh: Fast Voltage Contingency Screening using Radial Basis Function Neural Network, IEEE Transactions on Power Systems, Vol. 18, No. 4, Nov. 2003, pp. 1359 – 1366.

[12] G.C. Ejebe, B.F. Wollenberg: Automatic Contingency Selection, IEEE Transactions on Power Apparatus and Systems, Vol. PAS-98, No. 1, Jan/Feb. 1979, pp. 97 – 109.

[13] Y.K. Wu: A Novel Algorithm for ATC Calculations and Applications in Deregulated Electricity Markets, International Journal of Electrical Power and Energy Systems, Vol. 29, No. 10, Dec. 2007, pp. 810 – 821.

[14] X. Luo, A.D. Patton, C. Singh: Real Power Transfer Capability Calculations using Multi-layer Feed-forward Neural Networks, IEEE Transactions on Power Systems, Vol. 15, No. 2, May 2000, pp. 903 – 908.

[15] A.B. Khairuddin, S.S. Ahmed, M.W. Mustafa, A.M. Zin, H. Ahmed: A Novel Method for ATC Computations in a Large Scale Power System, IEEE Transactions on Power Systems, Vol. 19, No. 2, May 2004, pp. 1150 – 1158.

[16] A.B. Khairuddin, S.S. Ahmed: Slack-load Bus Pair Technique using Full AC Load Flow Algorithm for On-line Determination of ATC, 2nd World Engineering Congress, July 2002, pp. 25 – 28.

[17] The IEEE Reliability Test System, A Report Prepared by the Reliability Test System Task Force of the Application of Probability Methods Subcommittee, IEEE Transactions on Power Systems, Vol. 14, No. 3, Aug. 1999, pp. 1010 – 1020.

[18] I.J. Raglend, N.P. Padhy: Solutions to Practical Unit Commitment Problems with Operational Power Flow and Environmental Constraints, IEEE Power Engineering Society General Meeting, Montreal, Canada, June 2006.

[19] L. Fausett: Fundamentals of Neural Networks, Prentice-Hall, 1994.

[20] MATLAB Fuzzy Logic Toolbox User's Guide Version 2, 2001.