

Counterpropagation Neural Network for Solving Power Flow Problem

Jayendra Krishna, Laxmi Srivastava

Abstract—Power flow (PF) study, which is performed to determine the power system static states (voltage magnitudes and voltage angles) at each bus to find the steady state operating condition of a system, is very important and is the most frequently carried out study by power utilities for power system planning, operation and control. In this paper, a counterpropagation neural network (CPNN) is proposed to solve power flow problem under different loading/contingency conditions for computing bus voltage magnitudes and angles of the power system. The counterpropagation network uses a different mapping strategy namely counterpropagation and provides a practical approach for implementing a pattern mapping task, since learning is fast in this network. The composition of the input variables for the proposed neural network has been selected to emulate the solution process of a conventional power flow program. The effectiveness of the proposed CPNN based approach for solving power flow is demonstrated by computation of bus voltage magnitudes and voltage angles for different loading conditions and single line-outage contingencies in IEEE 14-bus system.

Keywords—Admittance matrix, counterpropagation neural network, line outage contingency, power flow

I. INTRODUCTION

POWER flow or load flow analysis is performed to determine the steady state operating condition of a power system, by solving the static load flow equations (SLFE) for a given network. The main objective of power flow (PF) studies is to determine the bus voltage magnitude with its angle at all the buses, real and reactive power flows (line flows) in different lines and the transmission losses occurring in a power system. Power flow study is the most frequently carried out study performed by power utilities and it is required to be performed at almost all the stages of power system planning, optimization, operation and control.

Fast security assessment is of paramount importance in a modern power system to provide reliable and secure electricity supply to its consumers. To perform the

contingency screening, which is one of the most CPU time-consuming tasks for on-line security assessment, the computation of the operating state in every few minutes is required simulating the occurrence of several contingencies and different loading conditions [1].

During last four decades, almost all the known methods of numerical analysis for solving a set of non-linear algebraic equations have been applied in solving power flow problems [2], [3]. The desirable features to compare the different PF methods can be the speed of solution, memory storage requirement, accuracy of solution and the reliability of convergence depending on a given situation. Though, robustness or reliability of convergence of the method is required for all types of application, the speed of solution is more important for on-line applications compared to the off-line studies.

For contingency selection, fast non-iterative approximate power flow methods such as DC power flow method, linearised AC power flow or decoupled power flow or fast decoupled power flow methods are used, which provides results having high inaccuracies. Full AC power flow methods are accurate but become unacceptable for on-line implementation due to high computational time requirements.

With the advent of artificial intelligence, in recent years, expert systems, pattern recognition, decision tree, neural networks and fuzzy logic methodologies have been applied to the security assessment problem [4]-[10], and other power system problems [11]-[13]. Amongst these approaches, the applications of artificial neural networks (ANNs) have shown great promise in power system engineering due to their ability to synthesize complex mappings accurately and rapidly. Most of the published work in this area utilizes multi-layer perceptron (MLP) model based on back propagation (BP) algorithm, which usually suffers from local minima and over-fitting problems [7], [10], [14]. Its ability to generalise a pattern depends on the learning rate and the number of units in hidden layer. In reference [15], a neural network load flow using an ANN-based minimisation model is proposed. A separate MLP model based on Levenberg-Marquardt second order training method has been used for computation for bus voltage magnitude and for angle at each bus of power system in [16]. As the number of neural networks required to solve power flow problem are large, it may not be applicable to a practical power system having huge number of buses.

A feed-forward counterpropagation neural network (CPNN) is proposed in this paper, which uses a different

Manuscript received September 25, 2005. This work was supported in part by University Grant Commission (UGC), New Delhi, India, under Research grant no. F 14-16/2003 dated 27/03/03.

Jayendra Krishna was with the Electrical Engineering Department, Madhav Institute of Technology & Science, Gwalior, India. He is now with Delhi Metro, India.

(Corresponding author e-mail: j-krishna@hotmail.com).

Laxmi Srivastava is with the Electrical Engineering Department, Madhav Institute of Technology & Science, Gwalior, India. (e-mail: srivastaval@hotmail.com).

mapping strategy namely counterpropagation. The CPNN provides a practical approach for implementing a pattern mapping task, since learning is fast in this network [17], [19].

The effectiveness of the proposed CPNN based approach is demonstrated by computation of bus voltage magnitudes and angles following different single line-outage contingency at different loading conditions on IEEE 14-bus system [20].

II. METHODOLOGY

Fig.1 shows the architecture of the proposed counterpropagation neural network. The composition of the input variables for the proposed neural network has been selected to emulate the solution process of a conventional power flow program.

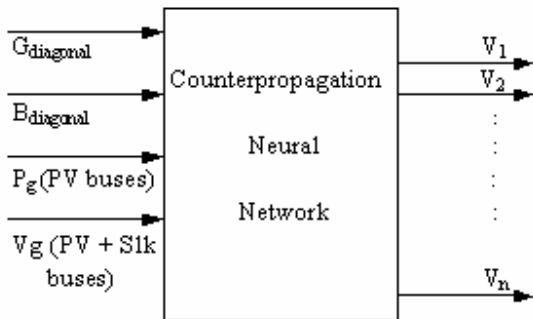


Fig. 1 Proposed CPNN Architecture

The input consists of the electric network parameters represented by the diagonal elements of the bus conductance and susceptance matrix, voltage magnitudes V_g of generation and slack buses, the active power generations P_g of PV buses. In order to speed up the neural network training, the conductance and susceptance are normalised between 0.1 and 0.9. For this CPNN based power flow model, the system loads, active and reactive power components are represented like constant admittance and they are included into the diagonal of the bus admittance matrix $[Y]=[G]+j[B]$, where $[G]$ and $[B]$ are the bus conductance and susceptance matrices respectively.

A. Power Flow Problem

The objective of power flow study is to determine the voltage and its angle at each bus, real and reactive power flow in each line and line losses in the power system for specified bus or terminal conditions. The power flow studies are conducted for the purpose of planning (viz. short, medium and long range planning), operation and control. For the purpose of power flow studies, it is assumed that the three-phase power system is balanced and also mutual coupling between elements is neglected. Variable associated with each bus of the power system include four quantities viz. voltage magnitude V_i , its phase angle δ_i , real power P_i and reactive power Q_i total $4n$ variable for n buses system. At every bus

two variables are specified, the remaining two can be found by solving the $2n$ power flow equations. Out of these four quantities only two are generally specified at a few bus and depending upon which two are specified, we have three categories of buses, namely Swing Bus or Reference Bus, Generator Bus or PV Bus and Load Bus or PQ Bus.

From the nodal current equations, the total current entering the i^{th} bus of m bus system is given by

$$I_i = Y_{i1}V_1 + Y_{i2}V_2 + \dots + Y_{im}V_m = \sum_{k=1}^m Y_{ik}V_k \quad (1)$$

where Y_{ik} is the admittance of the line between buses i and k and V_k is the voltage at bus k .

$$\text{In polar coordinates } V_i = V_i \angle \delta_i = V_i e^{j\delta_i}$$

$$V_k = V_k \angle \delta_k = V_k e^{j\delta_k}$$

$$Y_{ik} = Y_{ik} \angle \theta_{ik} = |Y_{ik}| e^{-j\theta_{ik}}$$

Here δ is the angle of the bus voltage and θ_{ik} is bus admittance angle. At i^{th} bus, complex conjugate power will be

$$S_i^* = P_i - jQ_i = V_i^* I_i \quad (2)$$

$$= V_i^* \sum_{k=1}^m (Y_{ik} V_k) \quad (3)$$

$$\text{or } P_i - jQ_i = \sum_{k=1}^m [V_i |Y_{ik}| V_k e^{-j(\theta_{ik} + \delta_i - \delta_k)}] \quad (4)$$

The real power at i^{th} bus will be

$$P_i = \text{Real} \sum_{k=1}^m [V_i |Y_{ik}| V_k e^{-j(\theta_{ik} + \delta_i - \delta_k)}] \quad (5)$$

$$\text{or } P_i = \sum_{k=1}^m [V_i |Y_{ik}| V_k \cos(\theta_{ik} + \delta_i - \delta_k)]$$

$$P_i = \sum_{k=1}^m V_i \cdot V_k [G_{ik} \cos(\delta_i - \delta_k) + B_{ik} \sin(\delta_i - \delta_k)] \quad (6)$$

Similarly, the reactive power at i^{th} bus will be

$$Q_i = \text{Imaginary} \sum_{k=1}^m [V_i |Y_{ik}| V_k e^{-j(\theta_{ik} + \delta_i - \delta_k)}] \quad (7)$$

$$\text{or } Q_i = \sum_{k=1}^m [V_i |Y_{ik}| V_k \sin(\theta_{ik} + \delta_i - \delta_k)]$$

$$Q_i = \sum_{k=1}^m V_i \cdot V_k [G_{ik} \sin(\delta_i - \delta_k) - B_{ik} \cos(\delta_i - \delta_k)] \quad (8)$$

Equations (6) and (8) are known as Static Power Flow Equation (SPFE).

The power flow equations used in this method for computation of voltage corrections are given as,

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & N \\ J & L \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \frac{\Delta V}{V} \end{bmatrix} \quad (9)$$

$$\text{where } H_{ik} = \frac{\partial P_i}{\partial \delta_k}; \quad N_m = \frac{\partial P_i}{\partial V_k} V_k$$

$$J_{km} = \frac{\partial Q_i}{\partial \delta_k}; \quad L_{km} = \frac{\partial Q_i}{\partial V_k} V_k$$

where H, N, J and L are the sub-matrices of the Jacobian. Eq (9) may be written as

$$\begin{bmatrix} \Delta \delta \\ \Delta V \\ V \end{bmatrix} = \begin{bmatrix} H & N \\ J & L \end{bmatrix}^{-1} \begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} \quad (10)$$

The solution of Eq (10) provides the correction vector i.e. $\Delta \delta$'s for all the PV and PQ type buses and ΔV 's for all the PQ type buses, which are used to update the earlier estimates of δ 's and V's. This iterative process is continued till the mismatch vector i.e. ΔP 's for all the PV and PQ type buses and ΔQ 's for all the PQ buses become less than a pre-assigned tolerance value ϵ . As can be observed from Eqs (9) and (10), during each iteration, Jacobian elements are to be calculated and its inverse is also required. Due to this fact, the Newton-Raphson method requires more time per iteration. However, this method provides accurate results and is the most reliable AC power flow method. To get accuracy in power flow solution, the NR power flow program has been developed in this paper and run to generate several training / testing patterns.

B. Counterpropagation Neural Network

The counterpropagation neural network is a hybrid of Kohonen clustering network and a Grossberg outstar. The CPNN model (Fig. 2) involves both supervised and unsupervised learning.

The Kohonen network implements the winner-take-all (competitive) strategy for the weights from the units in the input layer to the units in the hidden layer, and the Grossberg outstar maps the winning neuron into the desired output. The unsupervised and supervised training are applied to train the CPNN model. Since the PF problem demands a solution with high precision, the neural networks have to be trained considering a very small stopping criterion.

A large number of load patterns are generated randomly by perturbing the load at all the buses in wide range, voltage magnitude at PV and slack buses and real power generation at PV buses and transformer tap setting. Single line outages are considered as contingencies. Newton-Raphson (NR) power flow program is used to generate training / testing patterns for different load scenarios and for all the single-line outage contingencies.

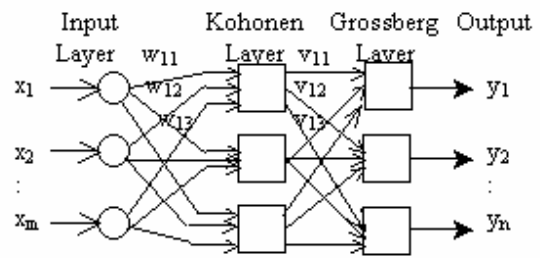


Fig. 2 Feed-forward Counterpropagation Network

CPNN is trained in a two-phase process. In the first phase, the Kohonen layer neuron weights are adjusted to match the input. The second training phase helps to adjust the Grossberg weights in order to fit the desired neuron output. To reduce the training time, the input vectors are normalized by dividing each component of an input vector by that vector's length and also the initial randomized weights for Kohonen' layer are normalized.

Two counterpropagation neural networks are developed in this work, one (CPNN1) for computation of bus voltage magnitudes at all the PQ type buses, while the other (CPNN2) for computation of bus voltage angles at PV type and PQ type buses. After training, the knowledge about the voltage magnitudes at all the PQ buses and voltage angle at different PV and PQ buses for various contingencies under different system operating conditions (training patterns) are stored in the structured memory by the trained CPNNs.

C. Solution Algorithm

The solution algorithm for power flow problem using CPNN is as follows:

- (i) A large number of load patterns are generated randomly by perturbing the load at all the buses, real power generation at the generator buses, voltage magnitudes at PV & slack buses and transformer tap settings.
- (ii) AC power flow (NR) programs are run for all the load patterns and also for contingency cases to calculate bus voltage magnitudes at all the PQ type buses and voltage angle at all the PV and PQ type buses except the slack bus.
- (iii) The diagonal elements of the bus conductance and susceptance matrix (active and reactive loads added to it), voltage magnitudes at PV and slack buses and real power generations at PV buses are selected as input features.
- (iv) All the input vectors and the initial randomized weights for Kohonen' layer are normalized before applying to the counterpropagation network.
- (v) Train the Kohonen clustering network by applying it to the CPNN competitive layer. Set iteration count $C=1$.
- (vi) Determine the unit (neuron) that wins the competition by determining the unit k whose vector w_k is closest to the given input.

(vii) Update the winning unit's weight vector as

$$w_k(C+1) = w_k(C) + \eta(x_i - w_k(C))$$

(viii) Repeat Steps (v) through (vii) until all input vectors are applied.

(ix) Increase the iteration count by one ($C = C+1$) and repeat Steps (v) through (viii) until all input vectors are grouped properly by applying the training vectors several times.

(x) After training the kohonen's layer, apply a normalized input vector x_i to the input layer and the corresponding desired output vector y_i to the output layer.

(xi) Determine the winning neuron k in the competitive layer.

(xii) Update the weights on the connections from the winning competitive unit to the output units

$$v_k(C+1) = v_k(C) + \eta(x_i - v_k(C))$$

(xiii) Repeat Steps (x) through (xiii) until all the input-output pairs in the training data are mapped satisfactorily

III. TEST RESULT

The IEEE-14 bus system, which is composed of 14 buses and 20 lines, has been used to test the proposed methodology. The data for IEEE-14-bus system were taken from [20] with buses renumbered to make bus-1 as slack bus having pre-specified voltage as $1.06\angle 0^\circ$ p.u., buses 2-5 as *PV* buses and buses 6-14 as load (*PQ*) buses. One CPNN model (CPNN1) was trained to provide bus voltage magnitude at all the *PQ* buses, while the other neural network (CPNN2) was trained to compute the bus voltage angles at all the *PV* and *PQ* type buses.

The total number of inputs is 29, including diagonal values of G and B, real and reactive loads, real bus power generation at bus no. 2, bus voltage magnitudes at 4 *PV* and the slack buses. For training and testing of CPNN, 25 load scenarios were generated by perturbing the load at all the buses in the range of 50% to 150%, *PV* bus voltage magnitude between 0.9 to 1.10, real power generation between 80% to 120%, transformer tap setting between 0.9 to 1.10. Single-line outages were considered as contingencies. Newton-Raphson (NR) power flow program was used to generate training / testing patterns for 25 load scenarios and for all the single-line outage contingencies.

The NR method converged for different loading conditions and for 19 line outage cases i.e. for 500 cases. Out of 500 generated patterns, 400 patterns corresponding to 20 load scenarios were arbitrarily selected and used for training of the CPNN, while 100 patterns corresponding to 5 load scenarios were used for testing the performance of the trained counter propagation neural networks.

Two CPNNs were developed, one for computation of bus voltage magnitudes at all the 9 *PQ* type buses, while the other for computation of bus voltage angle at 4 *PV* type buses and 9 *PQ* type buses (total 13). The number of hidden neurons (nodes) could be decided using some trial and error method.

The optimum structures of the neural networks were found

to be 29-223-9 for CPNN1 and 29-257-13 for CPNN2. Though the number of nodes in the Kohonen's layer seems to be large it will not affect the training/ testing time, as for any input there will be only one winning neuron and only that will participate in the training or testing process. The large number of neuron ascertains accuracy of CPNN during testing phase. The trained CPNNs were tested for 100 unknown (testing) patterns and were found to give accurate and fast computation of bus voltage magnitudes and voltage angles. The test results for one load scenario for outage of line no.18 (which has maximum testing error) are shown in Table I and Table II for voltage magnitude computation at 9 *PQ* buses and voltage angle computation at all the 13 buses except slack bus respectively.

TABLE I
 COMPARISON OF NR AND CPNN1 (29-223-9) OUTPUTS

| Voltage Mag (pu) | NR Method | CPNN Model | Absolute Error | % Error |
|------------------|-----------|------------|----------------|---------|
| V_6 | 0.9907 | 0.9886 | 0.0021 | 0.212 |
| V_7 | 1.0275 | 1.0251 | 0.0024 | 0.234 |
| V_8 | 0.982 | 0.9841 | 0.0021 | 0.214 |
| V_9 | 0.9903 | 0.9940 | 0.0037 | 0.374 |
| V_{10} | 0.9935 | 0.9933 | 0.0002 | 0.020 |
| V_{11} | 1.0241 | 1.0262 | 0.0021 | 0.205 |
| V_{12} | 1.0447 | 1.0444 | 0.0003 | 0.029 |
| V_{13} | 1.0333 | 1.0332 | 0.0001 | 0.010 |
| V_{14} | 0.9478 | 0.9585 | 0.0107 | 1.129 |

TABLE II
 COMPARISON OF NR AND CPNN2 (29-257-13) OUTPUTS

| Bus Angle (Deg.) | NR Method | CPNN Model | Abs. Error | % Error |
|------------------|-----------|------------|------------|---------|
| δ_2 | -5.1986 | -5.2442 | 0.0456 | 0.877 |
| δ_3 | -15.7247 | -15.7113 | 0.0134 | 0.085 |
| δ_4 | -13.9797 | -13.9962 | 0.0165 | 0.118 |
| δ_5 | -12.1803 | -12.2345 | 0.0542 | 0.445 |
| δ_6 | -9.1608 | -9.2524 | 0.0916 | 0.999 |
| δ_7 | -12.1749 | -12.2302 | 0.0553 | 0.454 |
| δ_8 | -7.0813 | -7.1070 | 0.0257 | 0.363 |
| δ_9 | -13.7942 | -13.8456 | 0.0514 | 0.373 |
| δ_{10} | -14.0071 | -14.0377 | 0.0306 | 0.218 |
| δ_{11} | -18.8335 | -18.6292 | 0.2043 | 1.085 |
| δ_{12} | -14.6863 | -14.6858 | 0.0005 | 0.003 |
| δ_{13} | -24.0712 | -24.1285 | 0.0573 | 0.238 |
| δ_{14} | -30.3637 | -30.2264 | 0.1373 | 0.452 |

As can be observed from Table I and Table II, the maximum absolute error is approx. 1% for bus voltage magnitude and angle computation, which is within acceptable limits. Both the trained CPNNs are able to compute voltage

magnitudes and voltage angles accurately. The results obtained by CPNN1 for voltage magnitude at bus nos. 7 & 11 and by CPNN2 for voltage angle at bus nos. 4 & 13 for all the testing patterns are compared in Fig.3, Fig.4, Fig.5 and Fig.6 respectively.

From these figures, it is clear that the trained counterpropagation neural networks are able to solve power flow problem accurately for unknown load patterns.

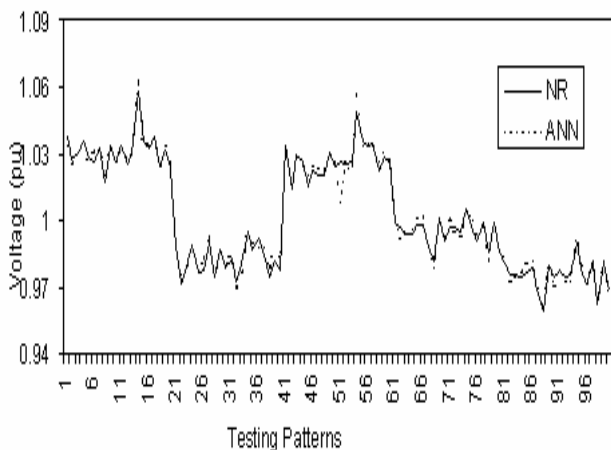


Fig. 3 Voltage magnitude at bus no. 7

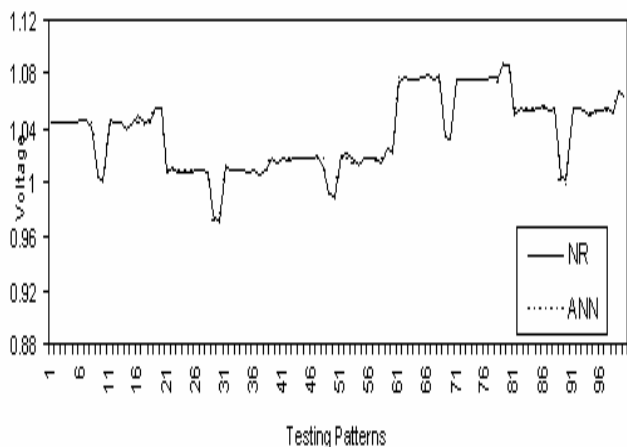


Fig. 4 Voltage magnitude (p.u.) at bus no. 11

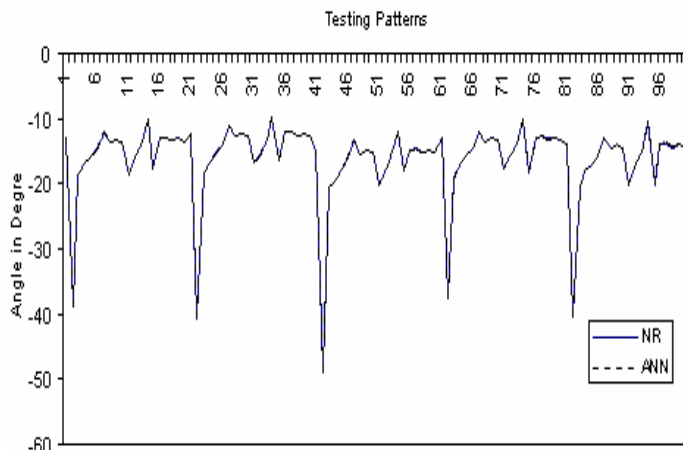


Fig. 5 Voltage angle at bus no. 4

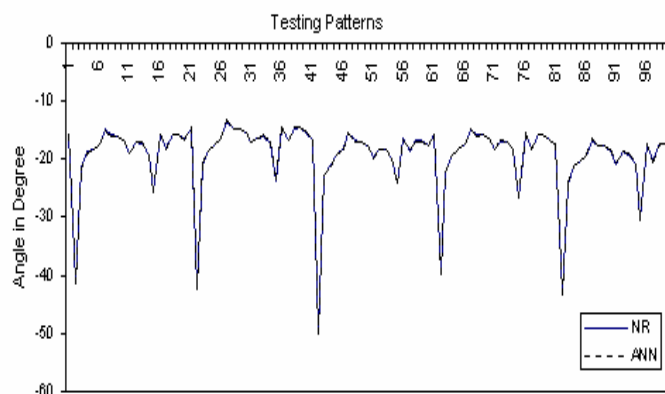


Fig. 6 Voltage angle at bus no. 13

IV. CONCLUSION

Counterpropagation neural networks have been developed to solve power flow problem in an efficient manner. In multi-layer feedforward neural network the training process is slow, and its ability to generalize a pattern-mapping task depends on the learning rate and the number of neurons in the hidden layer. On the other hand training of a counterpropagation neural network is very fast, at the same time the generalization capability of the CPNN allows it to produce a correct output even when it is given an input vector that is partially incomplete or partially incorrect.

Two CPNNs were trained, one for computation of voltage magnitude at all the *PQ* type buses and other for voltage angle at all the *PV* and *PQ* buses. The trained CPNNs were able to compute bus voltages magnitudes and voltage angles accurately for previously unseen patterns having changing load / generation conditions of the power system and for single-line outage contingencies as well.

Full AC power flow takes long time, as it should be run for any change in load/ generations and topology. On the other hand, once the CPNN models are successfully trained they provide accurate values of bus voltage magnitudes at all the PQ buses and voltage angles at all the PV and PQ type buses almost instantaneously. These values of voltage magnitudes and voltage angles can be used to compute line-flows and line losses etc. The counterpropagation neural networks based power flow method can be implemented for on-line security assessment in Energy Management Systems.

REFERENCES

- [1] S. Weerasooriya, M. El-Sharkawi, M. Damborg, R.J. Marks, II, "Towards static security assessment of a large scale power system using neural networks," *IEEE Proc.-C*, vol. 139, 1992, pp. 64-70.
- [2] B. Stott, "Review of load flow calculation methods," *IEEE Proc.*, Vol. 62, 1974, pp. 916 - 929.
- [3] R. Van Amerongen, "A general-purpose version of the fast decoupled load flow," *IEEE Trans. on Power Systems*, Vol. 4, No. 2, 1990, pp. 760-770.
- [4] T.S. Dillon, "Artificial Neural Network applications in power systems and their relationship to symbolic methods," *Electric Power & Energy Systems*, vol.13, no.2, 1991, pp. 66-72.
- [5] P.K. Kalra, Alok Srivastava and D.K. Chaturvedi, "Possible applications of neural nets to power system operation and control," *Electric Power System Research*, Vol. 25, 1992, pp. 83-90.
- [6] S. Vidya Sagar, Vankayala and D. Rao Natakki, "Artificial neural network and their application to power system – a bibliographical survey," *Electric Power System Research*, Vol. 28, 1993, pp. 67-69.
- [7] L. Srivastava, S.N. Singh and J. Sharma, "ANN applications in power systems: An overview and key issues," *Int. Conf. on computer applications in Elect. Engg., Recent Advances (CERA97)*, U.O.R., Roorkee (India), 1997, pp. 397-403.
- [8] Nima Amjadi, "Dynamic voltage security assessment by a neural network based method," *Electric Power Systems Research*, Vol. 66, no. 3, 2003, pp. 215-226.
- [9] M. Pandit, L. Srivastava and J. Sharma, "Cascade Fuzzy Neural Network Based Voltage Contingency Screening and Ranking," *Electric Power Systems Research*, Vol. 67, 2003, pp. 143-152.
- [10] T. Jain, L. Srivastava, and S.N. Singh, "Fast voltage contingency screening using Radial Basis Function neural network," *IEEE Transactions on Power Systems*, Vol 18, No. 4, 2003, pp. 1359-1366.
- [11] D. Thukaram, H.P Khincha, and H.P Vijaynarasimha, "Artificial neural network and support vector machine approach for locating faults in radial distribution systems," *IEEE Transactions on Power Delivery*, Vol 20, No. 2, 2005, pp. 710-721.
- [12] Won Kim Gwang, and K.Y. Lee, "Coordination control of ULTC transformer and STATCOM based on an artificial neural network," *IEEE Transactions on Power Systems*, Vol 20, No. 2, 2005, pp. 580-586.
- [13] Niu Dong-Xiao, Wanq Qiang, and Li Jin-Chao, "Short term load forecasting model using support vector machine based on artificial neural network," *Proceedings of 2005 Int. Conference on Machine Learning and Cybernetics, 2005*, Vol. 7, 18-21 Aug. 2005, pp. 4260-4265.
- [14] N. Kumar, R. Wangneo, P.K. Kalra, S.C. Srivastava, "Application of Artificial Neural Network to load flow," *Proc. TENCON'91, IEE Region 10 Int. Conf. On EC3-Energy, Computer, Communication and Control System*, Vol.1, 28-30 Aug. 1991, pp 199-203.
- [15] T.T. Nguyen, "Neural Network load-flow," *IEEE Proceedings - Gener. Transm. Distrib.*, Vol. 142, No. 1, 1995, pp. 51 - 58.
- [16] Leonardo Paucar, and Marcos J. Rider, "Artificial Neural Networks for solving power flow problem in electric power systems," *Electric power system Research*, Vol. 62, 2002, pp.139-144.
- [17] K.L. Lo, L.J. Peng, J.F. Macqueen, A.O. Ekwue and D.T.Y. Cheng, "Fast real power contingency ranking using counterpropagation neural network," *IEEE Trans. on Power Systems*, Vol. 13, No. 4, 1998, pp. 1259-1264.
- [18] J. Krishna, L. Srivastava, M. Pandit and S.N. Singh, "On-line Contingency Analysis Using Counterpropagation Neural Network," *Proceedings National Power System conference (NPSC 2004)* held at I.I.T. Chennai in December 2004, pp. 333-337.
- [19] Li Min Fu, *Neural Networks in Computer Intelligence* McGraw-Hill International Editions, 1994.
- [20] L.L. Freris, and A.M. Sasson, "Investigation of the load-flow problem," *Proc. IEE*, Vol. 115, No.10, October 1968, pp. 1459 - 1470.

J. Krishna received his M.E. degree in Electrical Engineering from Madhav Institute of Technology & Science Gwalior (India) in 2004. He worked as Project fellow in the department of Electrical Engineering, M.I.T.S., Gwalior (India) and at present working at Delhi Metro, India. His areas of interest are Power System operation and control and ANN application to Power Systems.

L. Srivastava obtained her M. Tech. degree in Electrical Engineering, from the Indian Institute of Technology, Kanpur, India in 1990 and her Ph.D. degree from University of Roorkee (Presently IIT), Roorkee, India in 1998. She is working as a professor in the Department of Electrical Engineering, M.I.T.S. Gwalior, India. She is currently involved in research in power system optimization and control, security analysis, ANN and fuzzy logic application to power system.