Multilayer Perceptrons Neural Network Automatic Voltage Regulator With Applicability And Improvement In Power System Transient Stability

Aslam P. Memon, A. Sattar Memon, Asif Ali Akhund, Riaz H. Memon

Abstract: In electrical power system network, the excitation system contributes in an effective voltage control and enhancement of the system stability. It must be able to respond quickly to a disturbance enhancing the transient stability and the small signal stability.

This work presents an artificial neural network (ANN) based automatic voltage regulator (AVR) controller for the excitation voltage system of synchronous machine in order to investigate the applicability and to improve the transient response. Multilayer Perceptrons (MLP) is the most popular type of Feedforward neural networks (FFNN) architecture of ANN and has proved many suceesful applications in power system and power system control and stability.

The linearized model of SM with single machine connected to infinite bus (SMIB) and AVR xcitation system is developped in Matlab/Simulink. The performance of proposed MLP neural networks is tested, verified and compared with conventional PID Proportional Integral Derivative (PID) AVR controller of synchronous generator. From simulation results, it is found that the MLP ANN AVR controller demonstrates not only the promising applicability but also better response by removing oscillations very quickly improving transient stability of power system.

Keywords: Synchronous Machine, Transient Stabilty, Automatic Voltage Regulator, Matlab/Simulink, Multilayer Perceptrons.

Aslam P. Memon is working as Assistant Professor in Department of Electrical Engineering, Quaid-e-Awam U.E.S.T, Nawabshah, Sindh, Pakistan. Email: **aslam@quest.edu.pk**

Abdul Sattar Memon is working as Assistant Professor in Department of Electrical Engineering, Quaid-e-Awam U.E.S.T, Nawabshah, Sindh, Pakistan Email: **a_sattarmemon@yahoo.com**

Asif A. Akhund is working as Lecturer in Department of Electrical Engineering, S.C.E.T, Rahim Yar Khan, Punjab, Pakistan. Email: **engr.asifakhund@gmail.com**

Riaz H. Memon is working as Lecturer in Department of Electrical Engineering, Quaid-e-Awam U.E.S.T, Nawabshah, Sindh, Pakistan. Email: **hussainriaz157@gmail.com**

I. INTRODUCTION

Three principal control systems directly affect a synchronous generator: the boiler, governor and exciter. Excitation system is considered as the source of field current for the excitation of synchronous generator and includes exciter and AVR (Automatic Voltage Regulator), and excitation control system is the feedback control system, which includes the synchronous machine, and its excitation. The objective of the control strategy is to generate and deliver power in an interconnected system as economically and reliably as possible while maintaining the voltage and frequency within permissible limits. [1-2].

In an interconnected power system, LFC (Load Frequency Control) and AVR equipment are installed for each generator. The controllers are set for a particular operating condition and take care of small changes in load demand to maintain the frequency and voltage magnitude within the specified limits. Small changes in real power are mainly dependent on changes in rotor angle (δ) and, thus, the frequency [3-5]. The reactive power is mainly dependent on the voltage magnitude (i.e., on the generator excitation). Changes in real power affects mainly the system frequency, while reactive power is less sensitive to changes in frequency and is mainly dependent on changes in voltage magnitude. Thus, real and reactive powers are controlled separately. The LFC loop controls the real power and frequency and the AVR loop regulates the reactive power and voltage magnitude [4-7].

The excitation controllers used previously described in literature are fixed gain controllers of which the gain settings are determined based on a particular operating condition [8-13]. Therefore, the gain settings of these controllers are always a compromise between the best values for light and heavy load conditions and, in some cases, a set of gain settings which are suitable for one operating condition may be completely unsatisfactory for another loading condition. In order to have the best controller gains over a wide range of loading conditions and to achieve better performance when the generator is subjected to different operating or loading conditions, we need the controllers which should have self learning and adaptation capabilities to cope with changes and uncertainties in the system; based only on the closed-loop real time input/output data and a qualitative knowledge of the system behavior; require no complicated controller design procedures so that anyone can use it easily and finally require neither off-line identification nor precise knowledge of system dynamics. The field of artificial neural networks (ANNs) has become enormously fashionable area of research in recent years and ANNs have found numerous successful applications in almost every field of engineering, and science. Application of ANNs to power system problems has been a research area with growing interest [14]. ANNs have been successfully applied to control many applications such as in D.C motor speed control, induction motor speed control; servomotor control and turbogenerator control [8-13].

The demand of above required controller can easily be fulfilled and implemented by ANNs which possess many properties like ANNs: learn by experience rather than by modelling or programming, architectures are distributed, inherently parallel and potentially real time, have the ability to generalize, do not require a prior understanding of the process or phenomenon being studied, can form arbitrary continuous nonlinear mappings and are robust to noisy data.

II. ARTIFICIAL NEURAL NETWORKS (ANNs)

Artificial neural networks have great potential since they are built on a firm mathematical foundation that includes versatile and well-understood mathematical tools [6-7, 15]. The ability of ANNs to model complex relationships makes them superior to conventional controller system. Conventional controllers require a good knowledge about mathematical model of the controlled system, which may not be available. Most ANN controllers on the other hand do not need such requirements and can handle complex systems efficiently. They learn to map input-output relationships by training process. The ANNs are trained to identify a process either off-line or on-line during the real time operation of the system [16-18].

Artificial neural networks are a system loosely modeled on the human brain and it is a network of interconnected elements. These elements were inspired from the studies of biological nervous systems. In other words, neural networks are an attempt at creating machines that work in a similar way to the human brain by building these machines using components that behave like biological neurons. ANNs can easily handle complicated problems and can identify and learn correlated patterns between sets of input data and corresponding target values. After training, these networks can be used to predict the outcome from new input data. Being universal function approximators, they are capable of approximating any continuous nonlinear functions to arbitrary accuracy [18]. The other advantages inbuilt in NNs are their robustness, parallel architecture and fault tolerant aptitude [19]. In literature ANN architectures are roughly divided into categories namely: feedback neural networks, cellular neural networks and most popular feedforward multilayer neural networks which are further sub divided into two classes (i) the multilayer perceptron (MLP) and (ii) radial basis function (RBF) neural networks. This work is exclusively related and focused on MLP NN architectures with its applicability [6-7, 17-20].

A. Multilayer Perceptron Neural Network (MLPNN) and its Approximation Capabilities

An MLP network is the best-known type of feedforward neural networks, and it has found applications in almost every field of engineering and science. An MLP network consists of an input layer, one or more hidden layers and an output layer. The number of hidden layers and the number of neurons in each layer are not fixed. Each layer may have a different number of neurons, depending on the application. Generally, an MPL network has a different number of neurons and different synaptic weights (and biases) for different layers. All neurons in a hidden layer have a sigmoid non-linearity such as a logistic function [6, 17-19]:

$$y_i = \frac{1}{1+\exp(-u_i)} \tag{1}$$

Or a hyperbolic tangent functions:

$$y_i = a\tanh(u_i) \tag{2}$$

Where u_i is the net internal activity level of neuron i, y_i is the output of the same neuron and a, b are constants. Generally, an MLP network learns faster with hyperbolic tangent function than the logistic function [19-20]. The important point to emphasize here is that non-linearity is smooth (i.e. differentiable everywhere). The output layer neurons may have the same activation function as the hidden layer neurons. However, many applications use a linear function of the output layer neurons. An MLP network is trained by the back-propagation algorithm. This algorithm adjusts the weights and biases of an MLP network so as to minimize the sum of squared errors of the network. This is done by continuously adjusting the values of the network weights and biases in the direction of steepest descent with respect to error. This procedure is called a steepest descent procedure.

A number of researchers [18-25] have proved mathematically that a single hidden layer MLP network is capable to approximate any continuous multivariable function to any desired degree of accuracy, provided that sufficiently many hidden layer neurons are available. In [23] author also proved another important result relating to the approximation capability of MLP networks employing sigmoidal hidden layer activations. They showed that these networks not only approximate an unknown function but also its derivative. In fact, Hornik, er al.[23] also showed that these networks can approximate functions that are not differentiable function. Light [24] extended Cybenko's results to continuous function and showed that integer weights and biases are sufficient for accurate approximation. The above results are very promising and provide great comfort to researchers in reinforcing their beliefs about the capabilities of MLP networks.

III. MODEL FOR THE MLPNN APPLICATIONS

The system considered in this paper is synchronous generator connected to an infinite bus through a transmission line having resistance R_e and inductance L_e (or a reactance X_e).
The system introduced in this paper is a single machine connected to an infinite bus (SMIB) as shown in Fig. 1. In order to develop the simulation model for this paper, we consider schematic diagram of governor and AVR excitation of synchronous generator as shown in Fig. 2. In the first step we linearize the system with proper assumptions and later on we make transfer function model of "the complete simulation linearized model of synchronous generator with LFC and PID excitation controller system" as shown in Fig. 3 [1-7, 17].

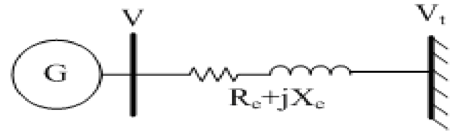


Fig. 01. SMIB system

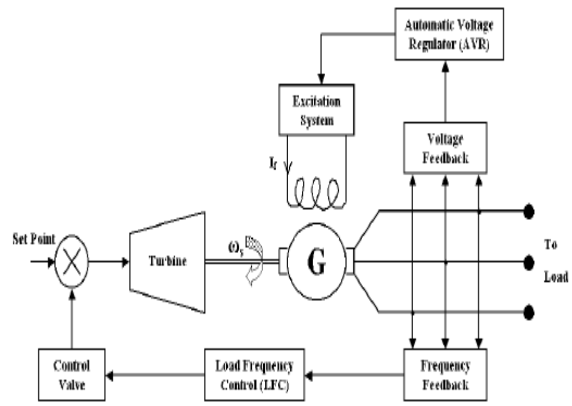


Fig. 02. Block diagram which is used for linear model with governor and AVR system.

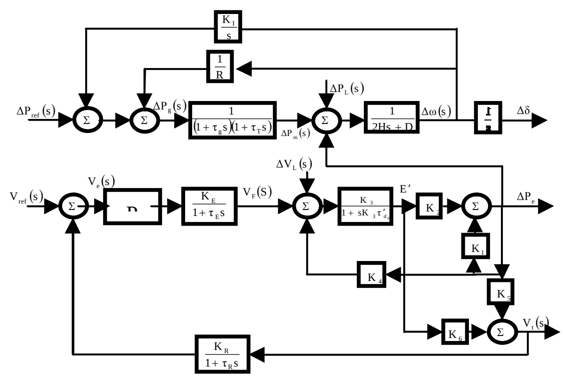


Fig. 03. Linearized model of governor and AVR excitation of synchronous generator with PID AVR excitation controller system

We select this model of Fig. 3, for our research work and we should know the parameters required for this model to run. They are gains of various controllers, the coupling coefficients or constants of linearized model (K_1-K_6), speed regulation, set of synchronous generator time constants, input and output of the model.

The gain values of various components as well as other constants required in this simulation model are typically values gathered from papers, articles and books [26-28, 1-7, 17]. Corresponding outputs in terms of the terminal voltage V_t and frequency deviation step responses $\Delta\omega$ are generated and the impacts on different parameters are observed. The variables in this simulation are $K_1, K_2, K_3, K_4, K_5, K_6$ and rest of the controller gains and time constants are fixed unless otherwise stated. All the values are described in per unit. K_3 is not mentioned in this data because it is impedance factor, which is a constant and is included with generator gain. Simulink, the selection of solver and time duration for simulation also is an important factor because it affects the accuracy and efficiency of the model [5-7, 17, 30]. In this simulation ode 45(Dormand-Prince) will be utilized because it was found to produce a more efficient response when compared to the other solvers available in Simulink. The period of simulation is set as 30 seconds so as to verify that there are no further oscillations [6-7, 17]. The numerical values of various components as well as other constants required developing simulation model [4-7, 17].

For turbine and governing systems: $K_g = 1.0, \tau_g = 0.2, K_T = 1.0, \tau_T = 0.5, R = 0.05$, SMIB (Synchronous generator and constants of linear model):

$D = 0.8, H = 5, K_1 = 1.5, \quad K_2 = 0.2, \quad K_4 = 1.4, K_5 = -0.1, K_6 = 0.5, K_G = 0.8, \tau_G = 1.4$

Values for excitation model: $K_E = 1.0, \tau_E = 0.4, K_A = 10, \tau_A = 0.1, K_R = 1.0, \tau_R = 0.05$

K_1 is the change in electrical torque for a small change in rotor angle at constant d axis flux linkage; i.e., the synchronizing torque coefficient. K_2 is the change in electrical torque for small change in the d axis flux linkage at constant rotor angle, τ'_{d0} is the direct axis open circuit time constant of the machine. K_3 is an impedance factor and K_3 = final value of unit step v_F response $= \lim_{t\to\infty} E'_\Delta(t) \big] \delta_\Delta = 0$, K_4 is the demagnetizing effect of a change in the rotor angle (at steady state). K_5 is the change in the terminal voltage V_t for a small change in rotor angle at constant d axis flux linkage, or K_6 is the change in terminal voltage V_t for a small change in the d axis flux linkages at constant rotor angle, or [4-7, 17]

A. Outline and limitations of the Simulation Model

There are some assumptions made prior to the design of the simulation model. They are as follows:

- A single turbine is used and will produce a constant torque with a constant speed maintained during steady state operation (at synchronous speed).
- The output terminals of the generator are connected to infinite busbar that has constant load.
- Only basic and linear models of the power system components (i.e. turbines, feedback sensors, exciter, governor etc) will be used except for the model of synchronous generator.
- The time constants of the synchronous machine used in this paper are assumed to be the optimum time constants extracted based on the values given in [5-6, 29-30].

This paper provides a means of determining the application of MLP neural networks in the excitation system of synchronous generator. The responses of synchronous machine in a power system are observed by computerized simulation. Extensive studies of various component parts are essential to closely simulate a working model.

Some of these studies include [5-6, 29-30]:

- Speed governor control
- Automatic voltage regulator (AVR) control
- Effects of using different types of prime mover (turbine)
- Proportional Integral Derivative (PID) control for excitation system
- Direct-quadrature axis theorem

The use of different AVRs, turbines and governors will not be included in the scope of this thesis so as to probe into the effects of the synchronous machine only.

With proper modelling of the synchronous machine in the power system, we can better understand how the machine reacts under small disturbances and different operating or loading conditions and hence designs a better controller of the synchronous machine [5-6, 29-30]. This research will be the interesting pioneering study of the simulation of the MLP neural network excitation response of synchronous generators, using first order model of synchronous machines and will serve as a basis for simulation of more comprehensive power systems in the future research.

IV. DESIGN OF MULTILAYER PERCEPTRON (MLP) NETWORKS

First layer has weights coming from the input. Each subsequent layer has a weight coming from the previous layer. The last layer is the network output.

The weights and biases are initialized and adapted with a specified learning function. Training is done with specified hyperbolic tangent sigmoid transfer function. Performance is measured according to the specified performance function [6-7, 17].

In our research model we have selected the following aspects

1. First step: Feedforward network architectures are created
2. Second step: the ranges of the PID gains at input and output from the minimum and maximum are recorded.
3. Two layer network structures have been chosen for this work.
4. The hidden layer or first layer requires eight neurons or nodes, and the output layer selects one neuron.
5. In the first layer/hidden layer selects hyperbolic tangent sigmoid transfer function.
6. The second layer needs linear transfer function as the activation functions.
7. Back-propagation algorithm is applied as the learning scheme
8. Learning parameters are: Show =6, it means the result is shown at every 6^{th} iteration, Learning rate (lr) is 0.06, Epochs are 10000, which is the maximum number of iterations, Goal is set at 1e-6.
9. Iteration type loop uses Levenberg-Marquardt algorithm which is the fastest training algorithm for networks of moderate size and possesses very simple and an efficient Matlab implementation, occupying very less memory.
10. Mean squared error (mse) is selected as performance function and network is stored with a command "generate simulation net".

MLP networks with only one hidden layer are sufficient for satisfactory performance, provided an appropriate number of neurons is used in the hidden layer.

A large data set may cause the learning process to be very slow; hence large data set is not essential for successful training. An appropriate subset of data for training can improve the speed of training without having a serious effect on the performance.

With the help of Levenberg-Marquardt algorithm into the standard back-propagation learning algorithm, the training time of an MLP network can be improved significantly.

There is no straightforward rule of choosing an appropriate number of hidden layer neurons for an optimal performance, which is a big disadvantage of MLP networks. This number is chosen by trial and error methods, starting with two or three neurons, and then increasing the number gradually, until satisfactory performance is achieved. This method of MLP is laborious and time-consuming task.

The weights and biases have been initialized and adapted with a specified learning function and training with specified hyperbolic tangent sigmoid transfer function. In the last performance is measured according to the specified performance function.

Steps for the training are as follows:

It should be noted that this process of MLP training consumes 27 seconds.

Network architecture: Feedforward network architectures have been selected

Range of the input and initialization of the network parameters: The ranges of the PID incoming signal and output signal outgoing from PID are recorded

Structure of the network: we choose one hidden layer network

Numbers of the nodes or the neurons in the hidden layer: Only 10 neurons/nodes are selected

Numbers of the neurons in the output layer: Only 01 neuron is for output layer

Activation function in the layers: Hyperbolic tangent sigmoid and linear transfer functions in hidden and output layers are chosen

Basic learning scheme: Back-propagation algorithm

Learning parameters are: Show =6, it means the result is shown at every 6th iteration, Learning rate (lr) is 0.06, Epochs are 10000, which is the maximum number of iterations, Goal is set at 1e-6.

Training of the network: Levenberg-Marquardt (LM) algorithm has been selected because it is the fastest version of the back-propagation algorithm.

Performance function: Mean squared error (mse) is selected as the performance function.

Table 01 shows design parameters of MLP architecture.

V. SIMULATIONS RESULTS

The simulations results in figures 04-07 discuss the responses of MLP AVR (terminal voltage Vt) and MLP LFC (frequency/speed deviation $\Delta\omega$) controllers with conventional controller in order to see the enhancement in stability of power system. MATLAB 7.13, Simulink 7.8 and Neural Network Toolbox 7.0.2 (R2011b) version have been utilized for these simulation results.

A. Terminal voltage and frequency/speed deviation responses:

As frequency takes more time than voltage hence time settling for frequency deviations and terminal voltage can be fixed at 25 and 0.4 seconds till the responses passes their oscillations and become stable. Terminal voltage (Vt), frequency/speed deviation ($\Delta\omega$) and excitation voltage responses are shown in Figures. 04, 05, 06 and the collective responses of all the controllers in 07 respectively.

Figure 04 from left and top to bottom shows the Vt responses (a) without controller (b) with conventional PID AVR controller (c) with MLP AVR and (d) with all the controllers (without controller, with PID AVR and MLP AVR) collectively. The collective responses of (e) are shown in large at right of figure 04. These all results clearly indicate that MLP is showing very good response with improvement in stability very easily and efficiently.

Figure 05 from left and top to bottom shows the frequency/speed deviation responses (a) without controller (b) with conventional PID LFC controller (c) with MLP LFC and (d) with all the controllers (without controller, with PID LFC and MLP LFC) collectively. The collective responses of (e) are shown in large at right of figure 05.

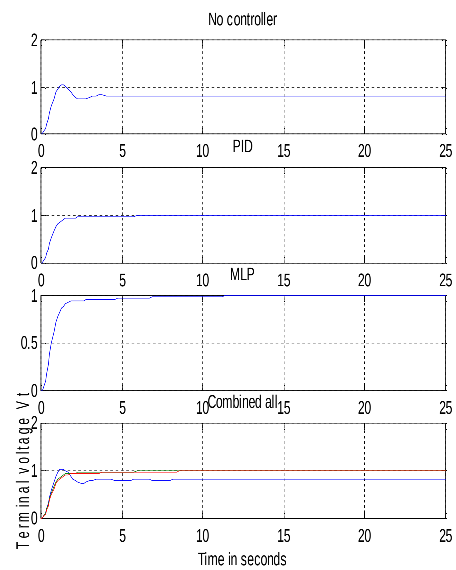
Figure 06 from left and top to bottom shows the excitation voltage responses and figure 07 demonstrates all the responses (Terminal voltages, Frequency deviations and Excitation voltages) with all controllers (Without controller, With PID, and With MLP).

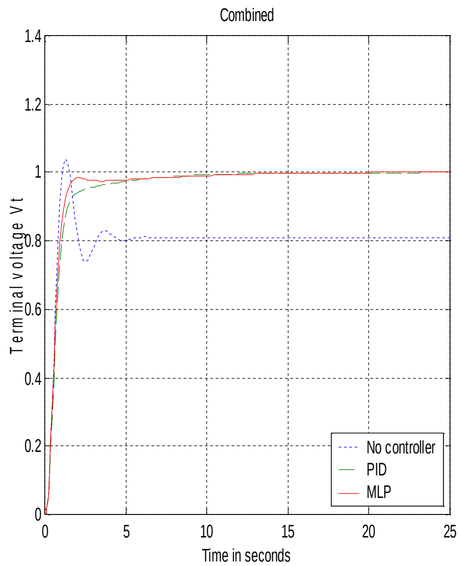
When MLP AVR and MLP LFC responses are compared, they prove good improvement in transient, small signal and dynamic stability. Both MLP AVR and MLP LFC controllers have their own excellent responses separately having no coupling between them which is verified from their results.

Table 01: Shows design parameters of MLP architecture of neural networks

Architecture	Number of neurons	
	1[st] Layer	2[nd] Layer
MLP	10	01

Transfer functions		Training time	Algorithm for training
1[st] Layer	2[nd] Layer		
Hyperbolic tangent	Linear	27 sec	LMA





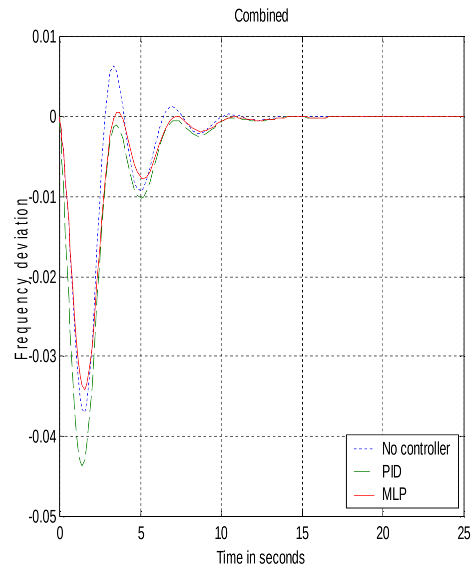
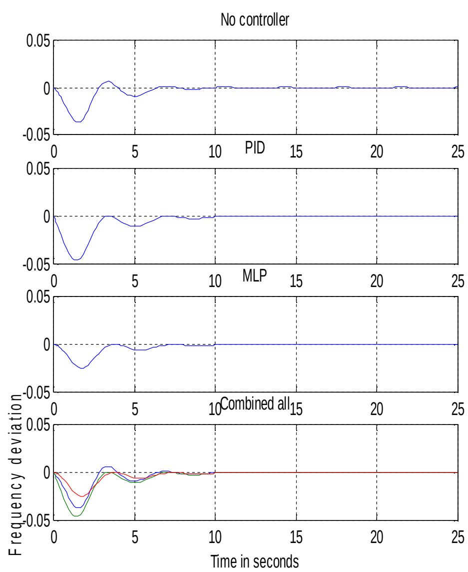
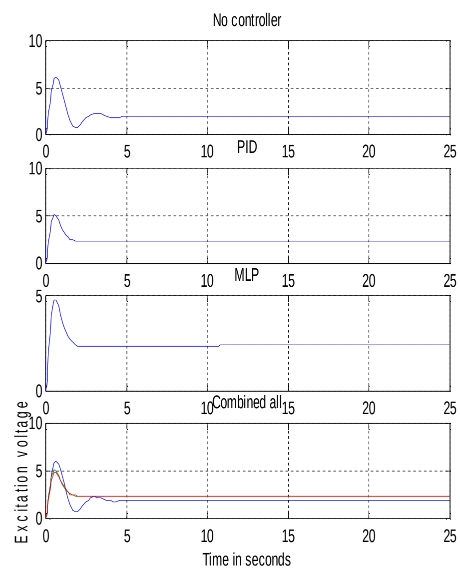


Fig. 04: Terminal voltages responses: Without AVR, PID-AVR, MLP-AVR controllers and collective terminal voltages responses (left, top to bottom) and in large respo

Figure 05: Frequency responses: Without controller, PID-LFC, MLP-LFC controllers and collective frequency responses (left, top to bottom) and in large response with collective controllers (right).





nse with collective controllers (right).

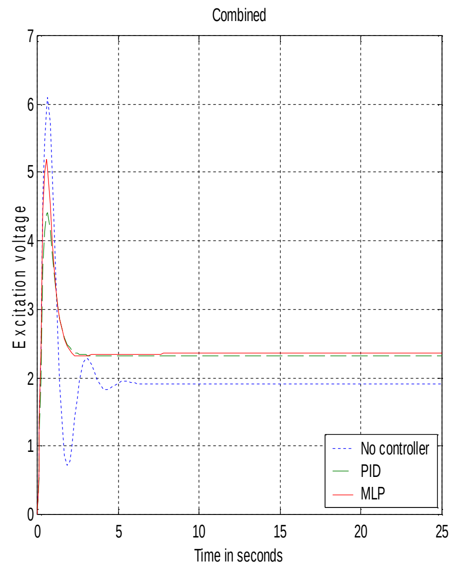


Figure 06: Excitation voltages responses of: Without controller, PID, MLP controllers and collective excitation voltages responses (left, top to bottom) and in large response with collective controllers (right).

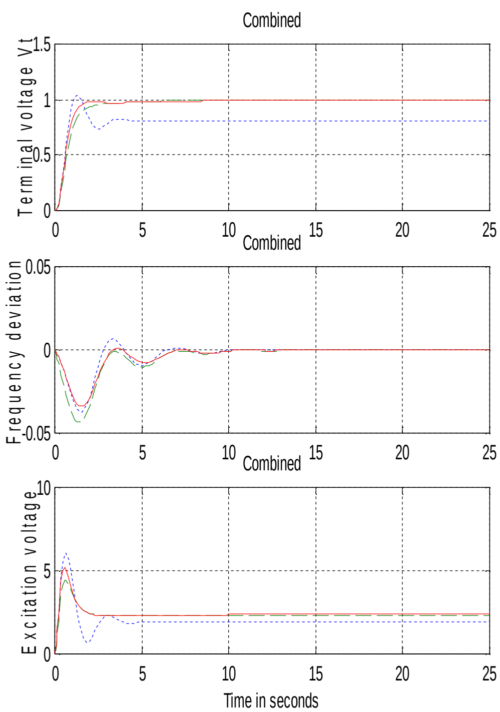


Fig. 07: The collective/combined responses (Terminal voltages, Frequency deviations and Excitation voltages) with all controllers (Without controller, With PID, and With MLP) are shown.

REFERENCES

[1] Kundur, P. (1994), Power System Stability and Control, 4th edition, McGraw-Hill Inc.

[2] Schleif, F.R.; Hunkins, H.D.; Martin, G.E.; and Hattan, E.E. "Excitation Control to Improve Power Line Stability", IEEE Trans. PAS, Vol. 87, pp. 1426-1434, 1968.

[3] deMello, F.P.; and Concordia, C., "Concepts of Synchronous Machine Stability as Affected by Excitation Control," IEEE Trans. PAS, Vol. 88, pp. 316-329, 1969.

[4] Anderson, P.M.; and Fouad, A. A., "Power System Control and Stability", Iowa State University Press, Iowa, U.S.A, 1977.

[5] Saddat, H. (1999), Power System Analysis, 1st edition, McGraw-Hill Inc.

[6] Memon, A. P., "Artificial Neural Network Applications in Electrical Alternator Excitation Systems", M. Phil, Thesis, Mehran University of Engineering and Technology, Jamshoro, Sindh, Pakistan, 2002.

[7] Aslam. P. Memon, "Neural Network Excitation Control System for Transient Stability Analysis of Power System" published in TENCON 2009 - 2009 IEEE Region 10 Conference, published January 2010.

[8] Adil H. Ahmad, Lina J. Rashad, "Excitation and Governing Control of a Power Generation Based Intelligent System" Eng. & Tech. Journal, Vol. 28, No. 5, pp. 871-889, 2010.

[9] Yuan-Yih Hsu and Chung-Yu Hsu, "Design of a Proportional-Integral Power System Stabilizer", IEEE Trans. on Power Systems, Vol. PWRS-1. No. 2, pp.46-53, May 1986.

[10] Yuan-Yih Hsu and K. L. Liou, "Design of Self-tuning PID Power System Stabilizers for Synchronous Generators", IEEE Trans. EC. Vol. 2, pp.343-348, 1987

[11] Ahmed, R. K.; Rattanb and Khalifa, I. (1995) "Real-Time Tracking Control of s DC Motor Using a Neural networks", IEEE 1995-NAECON, Vol. 2, May 1995, pp. 593-600.

[12] Burtton, B. R.; Harley and Habetler, T. (1996), "Identification and Control of Induction Motor Using Artificial Neural Network with Random Weight Change training", IEEE AFRICAN, Vol. 2, Sept. 1996, pp. 833-836.

[13] Strefezza, M.; and Dote, Y. (1993), "Radial Basis Neural Network Adaptive Controller for Servomotor", IEEE, Trans. on Power Systems, Vol. 2, May 1993, pp. 413-417.

[14] Neibur, D. (1995), "Artificial Neural Networks for Power Systems", Report by TF 38.06.06, Electra No. 159, April 1995.

[15] Suykens, J. A. K., "Artificial Neural Networks for Modelling and Control of Non-Linear Systems", 1st edition, Kluwer Academic Publishers, 1996.

[16] Fausett, L., "Fundamentals of Neural Networks, Architectures, Algorithms and Applications", 2nd edition, Prentice Hall, 1994.

[17] Aslam P. Memon., M. Aslam Uqaili, and Zubair Memon "Design of FFNN AVR for Enhancement of Power System Stability Using Matlab/Simulink", Mehran University Research Journal of Engineering and Technology, Vol. 31, No. 03, July, 2012.

[18] Unar, M. A. (1999), "Ship Steering Control Using Feedforward Neural Networks", Ph.D. Thesis, University of Glasgow, Glasgow, Scotland, U.K., 1999.

[19] Unar, M. A., Murry, D. J. (1999), "Automatic Steering of Ships Using Neural Networks", Int. J. Adapt. Control Signal Process, Vol. 13, pp. 203-218, 1999.

[20] Funahashi, K., "On the Approximate Realization of Continuous Mappings by Neural Networks". Neural Networks 2, p.p 359-192, 1989.

[21] Haykin, S., "Neural Networks – A Comprehensive Foundation", Macmillan, 1994.

[22] Funahashi, K., "On the Approximate Realization of Continuous Mappings by Neural Networks". Neural Networks 2, p.p 359-192, 1989.

[23] Hornik, K.,M Stinchcombe, and White, H. "Multilayer Feedforward Networks are Universal Approximatots" Neural Networks 3, p.p. 551-560, 1990.

[24] Light. W.,"Ridge Functions, Sigmoidal Functions and Neural Networks", In(E.W.Cheney, C.K. Chul, and L.I.Schumaker, Eds.), Approximation Theory, Academic Press, Boston, 1992.

[25] Lippmann, R.P.,"An Introduction to Computing With Neural Nets", IEEE ASSP Magazine 4, p.p.4-22, 1987.

[26] ANSI/IEEE Std 122 (1985), IEEE Recommended Practice for Functional and Performance Characteristics of Control Systems for Steam Turbine-Generator Units, The Institute of Electrical and Electronics Engineers, USA, 1985.

[27] ANSI/IEEE Std 421.1, (1986), IEEE Standard Definitions for Excitation Systems for Synchronous Machines, American National Standards Institute and The Institute of Electrical and Electronics Engineers, USA.

[28] IEEE Std 421.4™-2004 (Revision of IEEE Std 421.4-1990), "IEEE Guide for the Preparation of Excitation System Specifications", The Institute of Electrical and Electronics Engineers, USA, NY, 2004.

[29] Walton, A., "A Systematic Analytical Method for the Determination of Parameters of Synchronous Machines from the Results of Frequency Response Tests", Journal of Electrical Engineering-Australia, Vol. 20 No. 1, pp. 35-42, 2000.

[30] Memon, A. P., Shaikh, A. R., Unar, M. A., "A Simple Simulation Technique of Proportional Integral Derivative Excitation Control of Synchronous Generator", Mehran University Research Journal of Engineering and Technology, Vol. 21 No. 1, pp. 39-46, January 2000.

First author

Aslam Pervez Memon was born in Hala, Hyderabad, Sindh Pakistan, on February 11, 1968. He did his B.E. and Master of Philosophy Degree with honors in Electrical Engineering (Electrical Power Engineering) from Mehran University of Engineering and Technology (M.U.E.T) Jamshoro, Sindh, Pakistan.

He is pursuing Ph.D in Electrical Engineering from (M.U.E.T) Jamshoro, Sindh, Pakistan. Currently he is Assistant Professor in Department of Electrical Engineering in Quaid-e-Awam University of Engineering, Science & Technology, (Q.U.E.S.T) Nawabshah, Sindh, Pakistan. His field of interest includes electrical power quality, power electronics, electric drives, power system, artificial neural networks and mathematical modeling and simulation techniques in Matlab, Pspice, and PSCAD. He has approximately 17 years of experience in the research and teaching. He is a member of the PEC, IEEE societies, the National Society of Professional Engineers (NSPE) and an associate member of the International Association of Engineers (IAENG).

He is the currently supervisor and co-supervisor of more than o6 students of master of engineering in Q.U.E.S.T, Nawabshah. He has been the author of more than 15 papers in different journals and conferences in the area of interest.