1-1-1998

Simulation Studies with a Continuously Online Trained Artificial Neural Network Controller for a Micro-Turbogenerator

Ganesh K. Venayagamoorthy
Missouri University of Science and Technology

Ronald G. Harley

Recommended Citation

G. K. Venayagamoorthy and R. G. Harley, "Simulation Studies with a Continuously Online Trained Artificial Neural Network Controller for a Micro-Turbogenerator," *Proceedings of the International Conference on Simulation, 1998*, Institute of Electrical and Electronics Engineers (IEEE), Jan 1998.
The definitive version is available at https://doi.org/10.1049/cp:19980671

SIMULATION STUDIES WITH A CONTINUOUSLY ONLINE TRAINED ARTIFICIAL NEURAL NETWORK CONTROLLER FOR A MICRO-TURBOGENERATOR

G K Venayagamoorthy R G Harley

University of Natal, South Africa

ABSTRACT

This paper reports on the simulation studies carried out using MATLAB/SIMULINK and the practical implementation of a Continuously Online Trained (COT) Artificial Neural Network (ANN) controller to identify the continuous changing complex nonlinear dynamics of the power system, and another COT ANN to control a micro-turbogenerator which consists of a turbine simulator and a micro-alternator connected to an infinite bus through a short transmission line in a laboratory environment. This neural network controller augments/replaces the traditional automatic voltage regulator (AVR) and the turbine governor of the generator. Simulation and practical results are presented to show that this COT ANN identifier/controller has the potential to allow turbogenerators to operate more closely to their steady state stability limits.

INTRODUCTION

Offline trained Artificial Neural Networks (ANNs) have been investigated by Zhang et al (1) and Kobayashi et al (2) to provide damping signals (such as power system stabilisers) to turbogenerators. ANNs are basically intelligent nonlinear controllers which improve their steady state stability limits and nevertheless ensure a successful "ride through" a severe transient disturbances such as three phase faults. This allows greater usage of existing power plant which has obvious economic advantages.

Shepstone and Harley (3), improved on earlier work of others by proposing the use of a ***Continuously Online Trained*** (COT) Artificial Neural Network (ANN) as an adaptive turbogenerator controller to augment and perhaps even replace not only the automatic voltage regulator (as previous researchers have suggested), but also the ***turbine governor***. The conventional automatic voltage regulator (AVR) and turbine governor are usually designed to control the nonlinear turbogenerator in some optimal fashion, around a fixed operating point; therefore this performance is degraded at any other operating point, but the COT ANN controller designed (3), overcomes this problem. The work reported in (3), was in turn extended by Venayagamoorthy and Harley (4), who simulated the physical implementation of a COT ANN controller on a micro-turbogenerator.

This paper reports on the feasibility of using ANNs for controlling a 3 kW Mawdsley micro-turbogenerator available in the Micro-Machine Research Laboratory in the Department of Electrical Engineering, University of Natal, Durban with emphasis on the simulation issues using MATLAB/SIMULINK.

SIMULINK as a software tool has been designed for solving nonlinear differential equations in either state space or block diagram form. SIMULINK links into MATLAB and enables the SIMULINK programs to be called from MATLAB using, often, a single command. The results of the SIMULINK programs can be automatically passed back to MATLAB for further analysis. Simulations in SIMULINK run three to ten times faster than similar programs in MATLAB because SIMULINK programs are compiled. Due to these advantages offered by MATLAB/SIMULINK, this simulation environment was chosen for these studies.

The COT ANN architecture employed in this investigation consists of two separate artificial neural networks. One is used to identify the fast changing dynamics of the micro-turbogenerator and the other one for its control. Both ANNs are derived and modelled using SIMULINK. The second ANN replaces the conventional AVR and governor. In this paper the COT ANN is compared with the conventional controller setup under various transient operating conditions to show that ANNs have the potential to act as intelligent controllers for turbogenerators. Simulation results of the turbogenerator equipped with either the conventional or the COT ANN controller are presented in this paper. Currently, the COT ANN is under implementation in the Machine Research Laboratory in the Department of Electrical Engineering, University of Natal, Durban. Some laboratory measured results are represented in this paper to validate the SIMULINK models derived.

International Conference on SIMULATION, 30 September - 2 October 1998, Conference Publication No.457, © IEE, 1998

SIMULATION OF A PHYSICAL LABORATORY MODEL OF A POWER SYSTEM

The physical power system model setup in the laboratory as shown in Figure 1 consists of a micro-alternator driven by a DC motor whose speed - torque characteristics are modified (to behave like a turbine) by a turbine simulator, and a single short transmission line which links the micro-alternator to an infinite bus. A 3 kW, 220 V, three phase micro-alternator is employed to model the generating unit. This micro-alternator is designed to have per-unit parameters which are typical of those normally expected of 30-1000 MW generators. The machine parameters are given in Table 1. A separately excited 5.6 kW DC motor is used to drive the micro-alternator as a prime mover.

The micro-alternator in Figure 1 was modelled by using the general state variable equation of a synchronous machine given in eq.(1) where *F(x)* is the non-linear terms.

$$\dot{x} = Ax + Bu + F(x) \tag{1}$$

Equation (1) is developed from the two axis dq-equations with the machine currents, speed and rotor angle taken to be the state variables, with one damper winding on each axis to give a seventh order model expressed in eq.(2):

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \\ \dot{i}_d \\ \dot{i}_f \\ \dot{i}_{kd} \\ \dot{i}_q \\ \dot{i}_{kq} \end{bmatrix} = \omega_0 \begin{bmatrix} 0 & \frac{1}{\omega_0} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{K}{J\omega_0} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -Y_{1d}R_a & -Y_{4d}R_f & -Y_{5d}R_{kd} & -Y_{1d}(X_{mq}+X_a) & -Y_{1d}X_{mq} \\ 0 & 0 & -Y_{4d}R_a & -Y_{2d}R_f & -Y_{6d}R_{kd} & -Y_{4d}(X_{mq}+X_a) & -Y_{4d}X_{mq} \\ 0 & 0 & -Y_{5d}R_a & -Y_{6d}R_f & -Y_{3d}R_{kd} & -Y_{5d}(X_{mq}+X_a) & -Y_{5d}X_{mq} \\ 0 & 0 & Y_{1q}(X_{md}+X_a) & Y_{1q}X_{md} & Y_{1q}X_{md} & -Y_{1q}R_a & -Y_{3q}R_{kq} \\ 0 & 0 & Y_{3q}(X_{md}+X_a) & Y_{3q}X_{md} & Y_{3q}X_{md} & -Y_{3q}R_a & -Y_{2q}R_{kq} \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \\ i_d \\ i_f \\ i_{kd} \\ i_q \\ i_{kq} \end{bmatrix}$$

$$+\dot{\delta} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{M_e}{J\dot{\delta}} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Y_{1d}(X_{mq}+X_a) & Y_{1d}X_{mq} \\ 0 & 0 & 0 & 0 & 0 & Y_{4d}(X_{mq}+X_a) & Y_{4d}X_{mq} \\ 0 & 0 & 0 & 0 & 0 & Y_{5d}(X_{mq}+X_a) & Y_{5d}X_{mq} \\ 0 & 0 & -Y_{1q}(X_{md}+X_a) & -Y_{1d}X_{md} & -Y_{1q}X_{md} & 0 & 0 \\ 0 & 0 & -Y_{3q}(X_{md}+X_a) & -Y_{3q}X_{md} & -Y_{3q}X_{md} & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ i_d \\ i_f \\ i_{kd} \\ i_q \\ i_{kq} \end{bmatrix}$$

$$+\omega_0 \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{J\omega_0} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Y_{1d} & Y_{4d} & 0 & 0 & 0 \\ 0 & 0 & Y_{4d} & Y_{2d} & 0 & 0 & 0 \\ 0 & 0 & Y_{5d} & Y_{6d} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Y_{1q} & 0 \\ 0 & 0 & 0 & 0 & 0 & Y_{3q} & 0 \end{bmatrix} \begin{bmatrix} 0 \\ M_t \\ u_d \\ u_f \\ 0 \\ u_q \\ 0 \end{bmatrix} \tag{2}$$

where:

i - currents in the different coils,
δ - rotor angle,
$\dot{\delta}$ - speed deviation,
Y - admittance of coils
R - resistance of coils
X - impedance of coils
J - inertia of the machine
K - stiffness of the machine shaft,
ω_0 - nominal speed in one second,
u - voltages in the different coils,
M_t - mechanical torque.

The short transmission line is modelled using the following equations in the state space form.

$$u_d = U_m \sin\delta - R_e i_d - X_e i_q - L_e \dot{i}_d \tag{3}$$

$$u_q = U_m \cos\delta - R_e i_q + X_e i_d - L_e \dot{i}_q \tag{4}$$

where u_d and u_q are voltage components at the machine terminals, U_m is the voltage at the infinite bus and R_e, L_e, X_e are transmission line parameters.

The SIMULINK model of the state space equation in eq.(2) and the transmission model are shown in Figure 2. All the linear terms in eq.(2) and the required initial conditions are placed in the state space block in Figure 2 and the nonlinear terms in eq.(2), which are mostly products of states, are dealt with separately and are shown feeding into a multiplexer or into a summation block. The matrices and the initial conditions required in this SIMULINK block diagram are calculated and passed to the block by a MATLAB program.

The traditional AVR and exciter combination of Figure 3 are modelled in state space using SIMULINK as a single second order device with limits on its output voltage levels.

The exciter saturation factor S_e is given in the equation below.

$$S_e = 0.6093 \exp(0.2165 V_{fd}) \tag{5}$$

T_{v1}, T_{v2}, T_{v3} and T_{v4} are the time constants of the PID voltage regulator compensator; T_{v5} is the input filter time constant; T_e is the exciter time constant; K_{av} is the AVR gain; V_{fdm} is the exciter ceiling; and, V_{ma} and V_{mi} are the AVR maximum and minimum ceilings.

The turbine (simulator) and governor combination of Figure 4 are also modelled in state space using SIMULINK as a fourth order device so that reheating between the high pressure and intermediate pressure stages may be included in the model. The output of the turbine simulator is limited between zero and 120%.

In Figure 4, P_{ref} is the turbine input power set point value, P_m is the turbine output power, and pδ is the speed deviation.

The gain and time constants used are as follows:
- Governor gain, (K_g)
- Phase advance compensation, (T_{g1})
- Phase advance compensation, (T_{g2})
- Maximum turbine output, (P_{max})
- Servo time constant, (T_{g3})
- Entrained steam delay, (T_{g4})
- Steam reheat time constant, (T_{g5})
- p.u. shaft output ahead of reheater, (F)

To model the turbogenerator the following three SIMULINK subroutines shown in Figure 5, are combined to form the complete turbogenerator:

(a) A subroutine simulating a seventh order nonlinear model of a synchronous generator and a model of a short transmission line.
(b) A subroutine that implements a second order exciter and automatic voltage regulator.
(c) A subroutine that implements a fourth order turbine and governor.

The SIMULINK program has been run from a MATLAB program that does the following:

(a) reads a MATLAB file containing the required synchronous machine's parameters and data;
(b) using the machine data calculates the synchronous machine's reactances and susceptances;
(c) creates the necessary matrices and calculates matrix gain;
(d) calculates the transmission line impedance matrix;
(e) calculates the synchronous machine's initial conditions for use in the state space block in Figure 2;
(f) calculates the AVR's and governor's initial conditions;
(g) runs the SIMULINK simulation and plot the output results using MATLAB commands.

A separate MATLAB file has written to calculate the initial steady state values of the machine terminal voltage, field voltage and rotor angle for an operating point defined by the real power, P and the power factor, pf at machine terminals for a defined infinite bus voltage.

ANN CONTROLLER ARCHITECTURE

It is difficult to use either a simple ANN reference model for adaptive control of turbogenerators because of existing complex non-linearities in the turbogenerators, nor an ANN inverse model due to the high gain loops around the turbogenerator. The ANN controller adapted in this paper is the one proposed by (3) and has a two stage architecture as shown in Figure 6, namely an identifier stage and controller stage.

COT ANN Turbogenerator Identifier

The Identifier ANN (IANN) in Figure 6 is of the feedforward type and has three layers consisting of an input layer with twelve inputs, a single hidden layer with fourteen neurons and an output layer with two outputs. The inputs to the IANN are the deviation in the *actual* input to the turbine simulator, the deviation in the *actual* input to the exciter, the *actual* value of output voltage deviation and *actual* speed deviation of the generator. These four ANN inputs are delayed by the sample period of 20 ms and together with eight previously delayed values form twelve inputs altogether to the IANN. The IANN outputs are the *estimated* terminal voltage deviation and *estimated* speed deviation of the generator.

The number of neurons in the hidden layer of the IANN is determined empirically. The IANN weights are set to small random values and the conventional backpropagation algorithm is used to update these weights of the IANN. The differences between the respective outputs of the turbogenerator model and the outputs of IANN generate the error signals for the updating of weights in the IANN. A reasonable learning rate is determined by training this neural network and setting the learning rate parameter so that a compromise between the training time and the accuracy of the network is achieved.

The IANN is implemented using SIMULINK blocks as shown in Figure 7. The top half of Figure 7 models the forward path through an ANN with twelve inputs, one hidden layer consisting of fourteen neurons, and two outputs. The bottom half of Figure 7 models the backpropagation algorithm and updates the interconnection weights after each complete forward pass plus backpropagation pass.

The operation of the simulation is as follows:

(a) MATLAB is used to generate pseudorandom power deviation ΔP_{ref} and exciter input deviation ΔV_{field} signals for training of the IANN and these are saved;
(b) the pseudorandom signals are fed into the SIMULINK model of the turbogenerator. The resulting output speed deviation δ' and output terminal voltage deviation ΔV_t values are saved;
(c) the saved values in (a) and (b) above are then time delayed by one, by two and by three time periods as required. The time delayed signals are input to the IANN;

(d) the IANN calculates the estimated speed deviation δ' and terminal voltage deviation ΔV_t for the subsequent time period;
(e) the output of the IANN in (d) above is compared with the output of the turbogenerator for the subsequent time period. The difference between the outputs are used to form the error signals;
(f) the error signals from (e) above and the backpropagation algorithm are used to update the weights of the IANN at each sample time, in this case 20ms;
(g) steps (a) to (f) are repeated until the outputs of the IANN and of the turbogenerator model correspond sufficiently close.

During the investigations an offline training period of 20 s to 30 s was used which resulted in an error for speed deviation and terminal deviation of less than 0.006 rad/s and 1.35 $\times 10^{-3}$ pu respectively Longer training periods did not result in a substantial reduction in these errors. The tracking capabilities of the IANN were tried out by terminating the backpropagation training after 25 s, but continuing with the simulations of the turbogenerator model and the IANN for a further 5 s. Figures 8 and 9 show that the IANN can identify the complex nonlinear dynamics of the turbogenerator after only four seconds of training. Figure 10 shows that the IANN can also track, albeit with reduced accuracy, outputs even when the training is terminated.

COT ANN Turbogenerator Controller

The COT Neural Network Controller (NNC) in Figure 1 is a three layer network with six inputs, ten hidden neurons and two outputs. The inputs are the turbogenerator's *actual* speed and *actual* terminal voltage deviations. Each of these inputs is time delayed for one, two and three sample periods. The outputs of the NNC form the inputs to the turbogenerator's exciter and the turbine simulator. The number of neurons and learning rate are determined empirically as for the IANN. The NNC is implemented using SIMULINK blocks as for the IANN.

The SIMULINK model of the turbogenerator with the neural network controller online is shown in Figure 11.

The NNC in Figure 6 operates with online learning, however, it is necessary to train the NNC initially before online control operation is undertaken. Once the controller undertakes online operation the following basic steps are used thereafter:

(a) for a set of input signals, sample the output of the turbogenerator and the IANN (see Figure 6). Use the differences between these two outputs and the backpropagation algorithm to update the weights in the IANN and fix these weights;
(b) for the same input signals as in step (a), again sample the output of the IANN and compare the output of the identifier with the output of the desired response predictor for the turbogenerator. Use the difference between these two signals to form the error and backpropagate this error signal through the ANN to the output of the NNC;
(c) compare the output of the NNC with the backpropagated signal and with difference between these two signals form the error signal, and with the backpropagation algorithm, update the weights in the NNC. Apply the output of the NNC (obtained with the updated weights) to the exciter and turbine simulator of the turbogenerator to achieve the desired control;
(d) Repeat steps (a) to (d).

An advantage of this controller architecture is that the signals used are deviations from the setpoints and therefore when the turbogenerator is operating at the desired operating point there will be zero inputs to the NNC and zero outputs. This means that online learning takes place only when deviations from setpoints occur, and therefore ensuring minimum controller drift.

SIMULATION RESULTS

The dynamic and transient operation of the ANN regulator was compared with operation of the conventional controller (AVR and turbine governor) under two different conditions: a three phase short circuit on the infinite bus, and ± 5% step changes in the terminal voltage setpoint. Each of these was investigated for the turbogenerator driven at different power factors and transmission line configurations.

Samples of results are displayed in Figures 12 to 14. Figure 12 shows the performance of the NNC for ± 5% step changes in the terminal voltage with the turbogenerator operating at 1 pu power and 0.85 lagging power factor (in all the result graphs conventional controller is shown with solid lines while the neural network with dashed lines). Figure 13 shows a turbogenerator operating under the same conditions and experiencing a 50 ms three phase short circuit on the infinite bus. Figure 14 shows a turbogenerator experiencing a 50 ms three phase short circuit on the infinite bus with double the transmission line impedance used in Figure 13. In each of these tests ANN regulator has a performance at least comparable to that of a conventional controller and in each test the NNC has similar response times but with better damping.

Tests at other operating points confirmed that the controller is self-learning and performance does not degrade as with the conventional controllers.

PRACTICAL RESULTS

The implementation of the COT ANN controller was carried out in a similar way to that used in the simulation studies. A real-time digital control environment (RTDCE) developed by Wesbster et al (5), is used to implement the COT ANN controller. This RTDCE uses two Inmos T800 transputers and other interface hardware. The input signals to the neural networks were sampled at 70 Hz using the four A/D channels on the RTDCE. The outputs of the NNC (which are control signals)are output through the D/A channels of RTDCE and added to the exciter and the turbine simulator reference signals.

The IANN and NNC have to be trained offline and online to obtain proper connection weights before the NNC can be allowed to undertake online control of the turbogenerator. The offline training was carried out by applying ±7% deviations in the field voltage V_{field} of the exciter. The terminal voltage deviation of the IANN and micro-alternator are shown Figure 15. The practical results verify that an ANN can identify the complex nonlinear dynamics of turbogenerator as predicted by the simulation results.

CONCLUSION

The simulation studies of this work indicate that the two COT ANNs can identify and control the turbogenerator almost as well as a traditional AVR and governor combination, when the network configuration and system operating point conforms to that for which the AVR and governor were tuned. However, when system conditions change, such as different power levels and transmission line configurations, the ANN identifier and controller track these changes and do not give a degraded performance as the traditional AVR and governor do. It has also been verified that ANNs can online identify the continuous changing complex nonlinear dynamics of a power system. The successful performance of the COT ANNs even when the system configuration changes, comes about because the ***online training never stops***.

REFERENCES

1. Zhang Y, Chen GP, Malik OP, Hope GS, 1993, "An artificial neural network based powér system stabiliser", IEEE Trans. on Energy Conversion, Vol 8, No 1, 71-77

2. Kobayashi T, Yokoyanna A, Sekine A, 1994, "Nonlinear adaptive control of synchronous generator using neural network based regulator", International Conference on Intelligent Systems Application to Power Systems, Vol 1, 55-61.

3. Shepstone NM, Harley RG, Jennings G and Rodgerson J, 1996, "An investigation into the feasibility of using neural networks to control turbogenerators", IEEE-Africon Vol 2, 849-852.

4. Venayagamoorthy GK, Harley RG, 1998, "A continuously online trained artificial neural network controller for a microturbogenerator", Southern African Universities Power Engineering Conference, 215-218.

5. Webster MR, Woodward DR, Daina G, Harley RG, Levy DC, 1993, "Evaluation of a transputer based embedded controller for high dynamic control applications", Proc. of the IFAC World Congress, 395-400.

TABLE 1- Micro-alternator constants

$T_{d0}' = 6.69s$	$T_{q0}'' = 0.25s$	$X_d' = 0.205pu$
$T_d' = 0.66s$	$T_q'' = 27ms$	$X_d'' = 0.164pu$
$T_{d0}'' = 33ms$	$T_{kd} = 38ms$	$X_q = 1.98pu$
$T_d'' = 26.4ms$	$X_d = 2.09pu$	$X_q'' = 0.213pu$

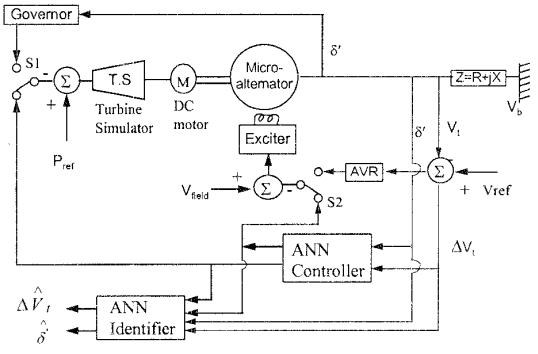

Figure 1: Physical laboratory model of a power system

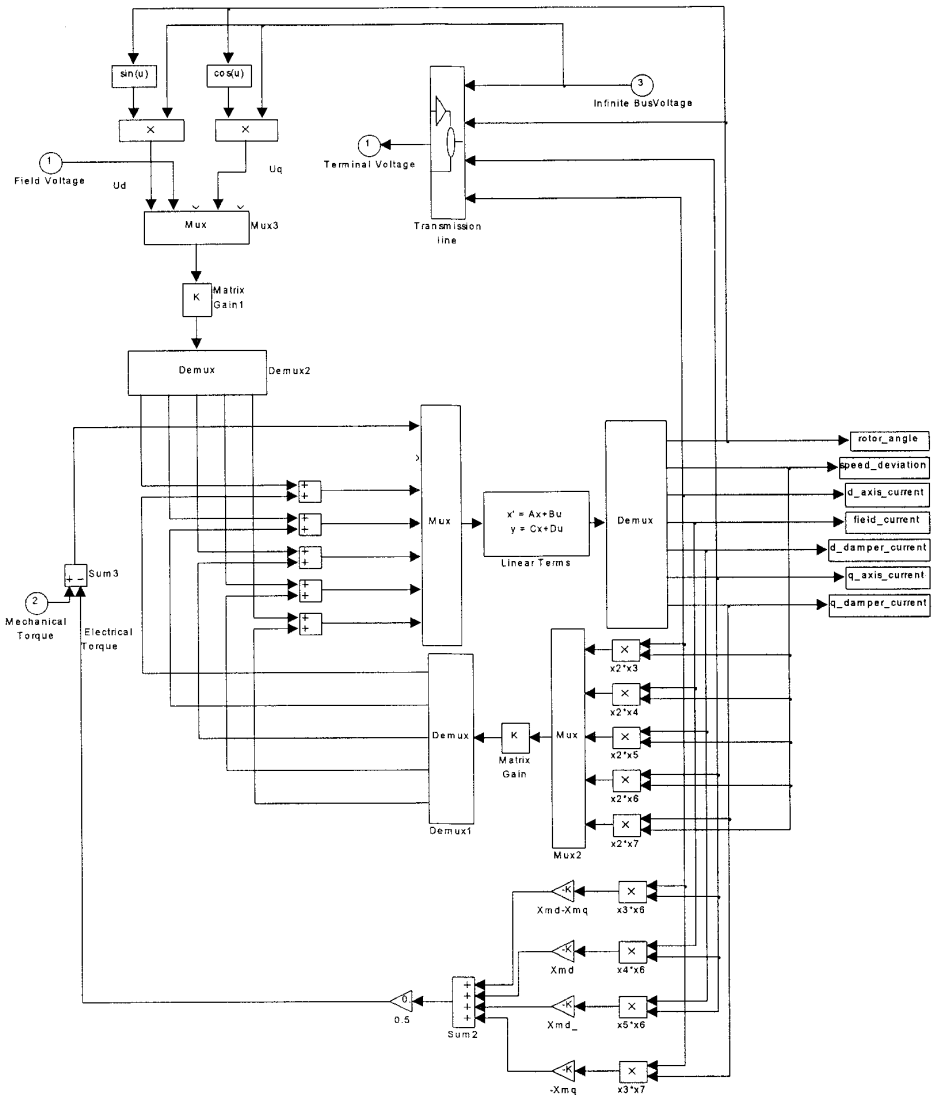

Figure 2: Seventh order synchronous machine model and transmission line

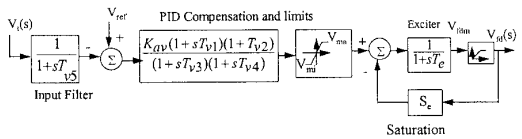

Figure 3: Block diagram of the AVR and excitation system

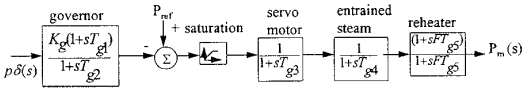

Figure 4: Block diagram of the turbine governor and turbine

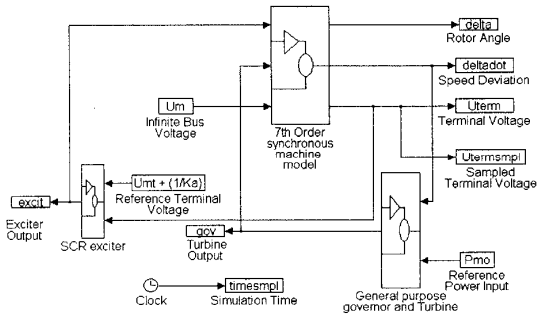

Figure 5: SIMULINK simulation program

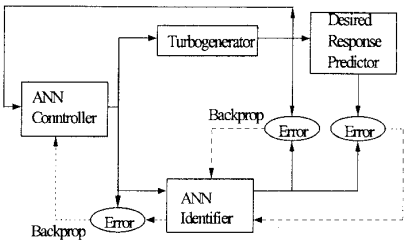

Figure 6: Artificial neural network architecture

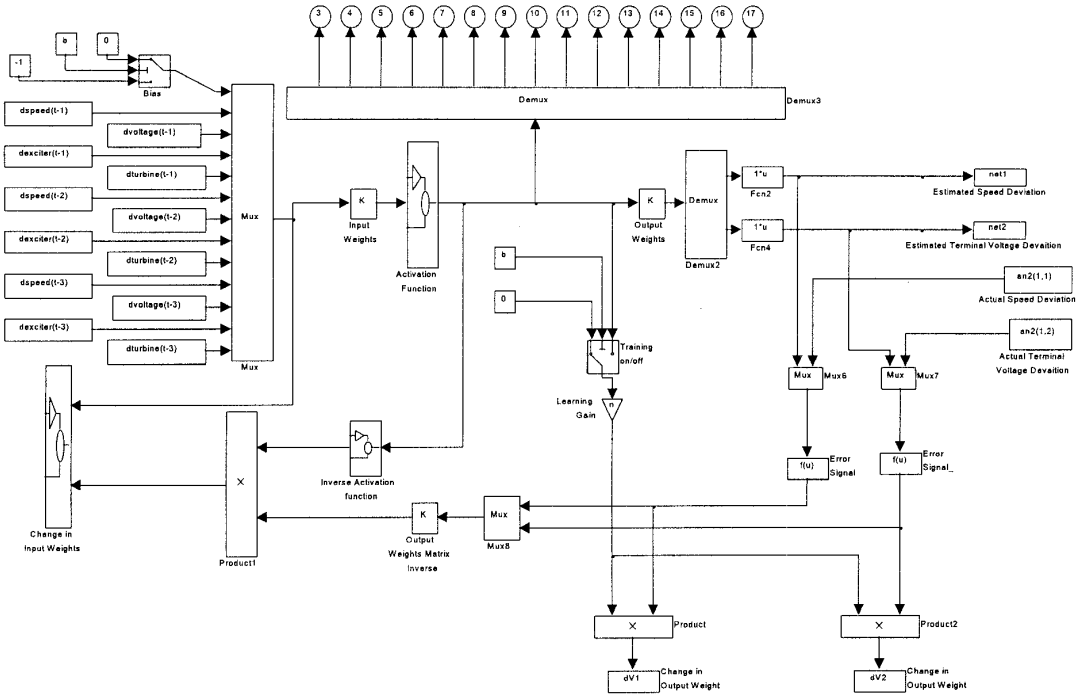

Figure 7: SIMULINK model of IANN

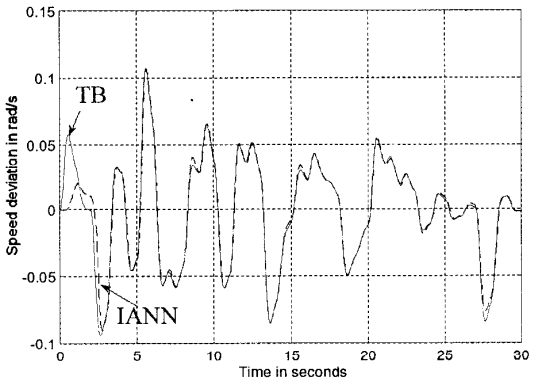

Figure 8: Speed deviation signal δ' of the turbogenerator (TB) and IANN

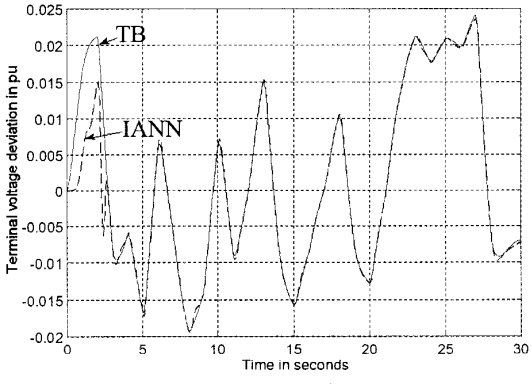

Figure 9: Terminal voltage deviation signal ΔV_t of the turbogenerator (TB) and IANN

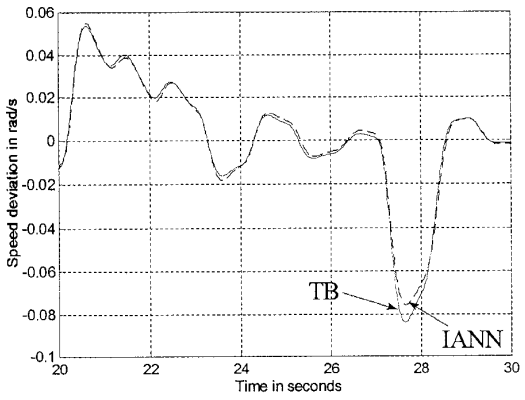

Figure 10: Speed deviation signal δ′ of the turbogenerator (TB) and IANN when the training is terminated

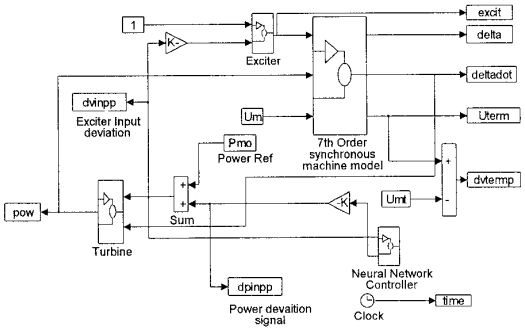

Figure 11: SIMULINK model of turbogenerator with the NNC

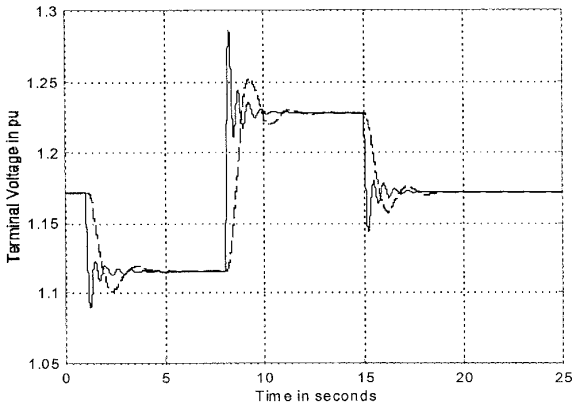

Figure 12: ± 5% step changes in the terminal voltage (P = 1.0 pu, pf = 0,85 lagging)

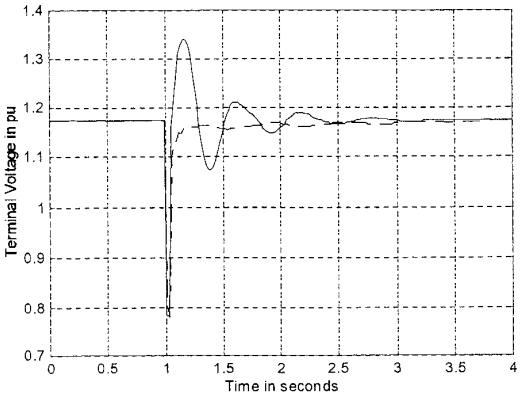

Figure 13: A 50 ms three phase short circuit at the infinite bus (P = 1 pu, pf =0.85 lagging)

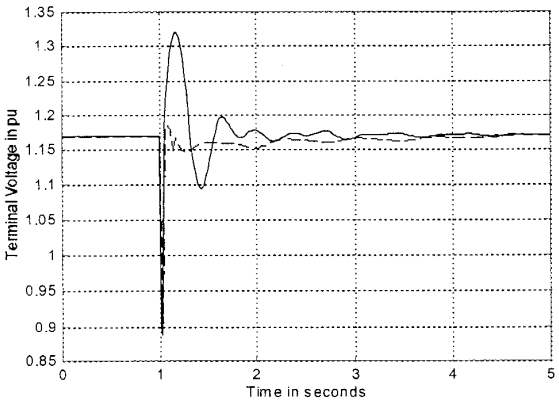

Figure 14: A 50 ms three phase short circuit at the infinite bus (P = 1 pu, pf = 0.85 lagging) with double the transmission line impedance used in Figure 13

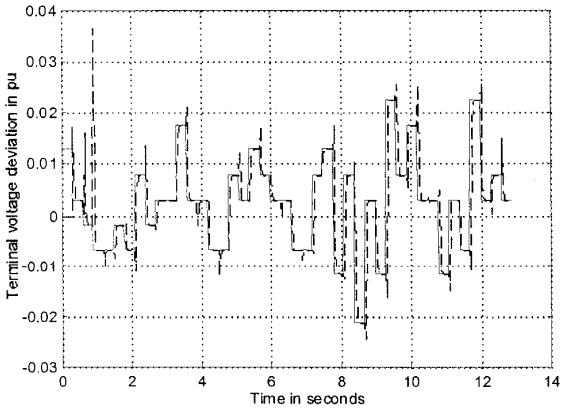

Figure 15: Practical neural network modelling of the dynamics of the turbogenerator