Hybrid Neuro-Genetic Based Controller of Power System

Dr. Akram Fajer Bati & Nabeel E. Adam

Received on :27/12/2005
Accepted on :4/5/2006

Abstract

Evolutionary algorithms, Genetic algorithms in particular, are known to be robust and have been increasing popularity in the field of numerical optimization. Neural networks and genetic algorithms demonstrate powerful problem solving ability. They are based on quite simple principles, but take advantage of their mathematical nature: non-linear iteration. Neural networks with back-propagation learning showed results by searching for various kinds of functions. However, the choice of the basic global performance index (parameter weights) often already determines the success of the training process. The study presents a hybrid controller system; has been optimized by genetic algorithm optimization tool. GA based optimization scheme for simultaneous coordination of multiple power system damping controllers. Local measurements will be considered as input signals to the damping controller. The proposed algorithm will be applied to tuning controller of a single machine infinite bus power system . All simulations will be carried out using MATLAB based package for nonlinear simulations of power systems Controllers will be designed using MATLAB neural network functions and genetic algorithms optimization tool.

الخلاصة

الخوارزميات التطورية و بصورة خاصة الخوارزميات الجينية معروفة بانها ذات متانة عالية.الشبكات العصبية والخوارزميات الجينية تعطي قابلية التعلم بالشبكات العصبية باستخدام ظاهرة الانتشار العكسي لها مساوى بالاستفادة من متانة الخوارزميات الجينية في ايجاد الاحل الامثل للمسألة متغلبا على مساويء استخدام طرق التعلم في الشبكات العصبية. يبين البحث كيفية تعلم مسيطر هجين لنظام قدرة متكون من ماكنة واحدة مرتبطة الى قضيب لا نهائي بايجاد الارتباطات المثلى للمسيطر. تم استبدال مسيطرات النظام للتحكم بالفولتية ومتحكم التوربين بمسيطر واحد . تمت الدراسة باستخدام MATLAB package لتعليم الشبكة العصبية المكافئة للنظام وادوات الخورزميات الجينية لايجاد اوزان المسيطر.

1-Introduction

Turbo-generators supply most of the electrical energy production and therefore are the major components in electric power systems and their performance is directly related to security and stability of system . The tendency of power system to develop restoring forces equal to or greater than the disturbing forces in order to maintain the state of equilibrium is known as stability. The stability problem is concerned with the behaviour of the synchronous machine and are generally divided into two main categories, steady state stability and transient stability. An extension of the steady-state stability is known as the dynamic stability, which is concerned with small disturbances lasting for a long time including the action of automatic control devices [1].

Conventional automatic voltage regulators (AVRs) and turbine governors are designed to control, in some optimal fashion, around one operating point, at any other point the generator performance is degraded [2]. The Single Machine Infinite Bus (SMIB) system is used by the designers with their non-linear equations linearized at one operating point for the design, evaluating and modification of the dynamic performance of the particular controller. The non linear equations of the SMIB are then linearized at one operating point and used for the design of traditional controllers. Artificial Neural Networks (ANNs) are good at identifying, then controlling a non-linear system, and are suitable for multi-variable applications, as they can identify the interactions between the inputs and outputs. As such, an accurate model of the system is no longer needed [3]. It has been shown that a multi-layer feed-forward neural network using deviation signals as its inputs can identify the complex and non-linear dynamics of a SMIB system with sufficient accuracy in designing a generic controller, to give optimal dynamic system response irrespective of the generator load and system configurations. Numerous publications have reported on the designs of ANN controllers for turbo-generators, and have presented both simulation and experimental results showing that ANNs have the potential to supplement and even replace traditional controllers. Detailed studies were carried out previously on the real time implementation of a Continually Online Trained (COT) ANN identifier/controller for a SMIB system and simulation results were validated by actual measurements on a laboratory system [4]. In recent years, researchers are increasingly interested in the use of Genetic Algorithms (GAs) as means to control various classes of systems. Genetic algorithms are robust search techniques based on the principles of evolution. GAs being evolutionary method, have gained recognition as a general problem solving technique in many applications, including function optimization, image processing, classification and machine learning, training of neural networks, and system control. The association of GA with ANN was found highly appropriate applying to plant system control optimization study. Their advantages then be made full use of the ability of ANN to learn complex nonlinear mapping, and that of GA to find the global optimum in a bounded parametric search space [5]. GAs are stochastic, parallel search algorithms based on the mechanics of natural selection and the process of evolution [6,7]. The application of GAs to control engineering can be broadly classified into two main areas: off-line design and analysis and on-line adaptation and tuning.

In off-line applications the GA can be employed as a search and optimization engine, for example to select suitable control laws for a known plant to satisfy given performance criteria or to search for optimal parameter settings for a particular controller structure. In on-line adaptation GAs may be used as a learning mechanism to identify characteristics of unknown or non-stationary systems or for adaptive controller tuning for known or unknown plants. This study focuses on the use of Genetic Algorithms as an optimization tool for connection weights of neural controller to a power system.

2- Modeling of Power System

2-a Synchronous generator model

The synchronous generator connected to an infinite bus is a multivariable non linear dynamic system, described by the well known set of equations [8]. For the analysis and design of control systems for synchronous machine, a simplified linearized third order model, which is called also Heffron-Phillips is most popular[9].The linearization constants k1,k2…..,k6 depend on the parameters ,and the operating condition of a synchronous generator connected to an infinite bus through transmission lines.

2-b Exciter model

The most important component other than the synchronous machine in the power system is the excitation system. The most basic form[10] of expressing the exciter model can be represented by a gain *Ke* and a single time constant *Te* ,the excitation system – transfer function is :-

$$\frac{V_f}{Vr} = \frac{K_e}{1 + sT_e} \quad \text{................(1)}$$

where V_r =the output voltage of regulator

$V_{f=}$ field voltage

2-c Turbine and governing system

There are different types of turbine having different characteristics. The source of of mechanical power can be hydraulics turbine, steam turbine and others. The simplest form of model for non-reheat steam turbine can be approximated by using a single time constant T_t . The governing system time constant T_g .

$$\frac{\Delta p_m}{\Delta p_v} = \frac{1}{1 + sT_t} \quad \text{.................. .(2)}$$

$$\frac{\Delta P_v}{\Delta P_g} = \frac{1}{1 + sT_g} \quad \text{...............…...(3)}$$

where $\Delta P_{m=}$the changes in mechanical power input

$\Delta P_{v=}$the changes in steam valve position

ΔP_g=the changes in speed governor output

The parameters, operating conditions of the synchronous generator and the time constants of the exciter, turbine and governor are given in appendix.The overall power system is modeled in MATLAB SIMULINK package as shown in fig (1) ,and the open-loop response for the voltage and active power deviations are shown in fig. (8).

3-Artificial Neural Networks(ANNs)

Neural networks are composed of simple elements operating in parallel. These elements are inspired by biological nervous systems. As in nature, the network function is determined largely by the connections between elements. We can train a neural network to perform a particular function by adjusting the values of the connections (weights) between elements. Commonly neural networks are adjusted, or trained, so that a particular input leads to a specific target output. Such a situation is shown below. There, the network is adjusted, based on a comparison of the output and the target, until the network output matches the target. Typically many such input/target pairs are used, in this *supervised learning*, to train a network. figure (2) shows block diagram for ANN learning.

Neural networks have been trained to perform complex functions in various fields of application including pattern recognition, identification, classification, and control systems. Back propagation training using LEVENGERG algorithm is used to reduce the Mean Square Error(MSE) between the actual values of the plant and estimated values of the neural network identifier. The neural network architecture consists of two separate ANNs, namely one for the identifier and the other for the controller. The neural network is used for dynamic modeling, and the controller is used to replace the automatic voltage regulator and the governor used in conventional controllers.

3-a ANN identifier architecture

The neuro-identifier is developed using the series-parallel Nonlinear Auto Regressive Moving Average (NARMA) model [11].This model output $y^\wedge$ at time ($k+1$) depends on both past n values of output and m past values of input .The neuro-identifier output equation takes the form given by :-

$$Y(k+1)=\begin{bmatrix} y(k, y(k-1).....,y(k-m+1) \\ u(k),u(k-1),....,u(k-m+1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots(4)$$

Where *y(k)*and *u(k)* represent the output and input of the plant at time *k* respectively. The ANN identifier structure is fixed as three layer feed-forward neural network with twelve inputs, a single hidden layer with fourteen neurons with tan sigmoid activation function and two outputs. The inputs are the actual deviation in the input to the exciter, the actual deviation inputs to the turbine, the actual deviation in terminal voltage and the actual deviation in electrical power. The four inputs are time delayed by 10 ms and together with eight previously delayed values form the twelve inputs for the model. The outputs of the ANN are the estimated terminal voltage deviation and the estimate electrical power deviation of the turbo-generator. In figure (3), *Vref* is the exciter input voltage and *Pin* is the turbine input power. Note that no feedback control loops are used since the Multi Layer Perceptron (MLP) is evaluated on the plant without controllers. The input power deviation (□*Pin*) and exciter input voltage deviation *(*□*Vref*) are generated as small pseudo-random binary sequence signals (PRBSs) as proposed by to perturb the plant in order to measure its response power deviation *(*□*Pe*□□) and terminal voltage deviation (□*Vt*), thus a two-input two-output system[12]. Figure (4) shows the PRBSs, outputs of the plant, outputs of identifier, and error performance of ANN .

3-b ANN controller architecture

The second ANN forms the controller and is a three-layer feed-forward neural network with eight inputs, a single hidden layer with twelve neurons, tan sigmoid activation function and two outputs with linear activation function. There are 134 weights between neurons and biases. The input signals are the change in reference voltage, the change in input turbine power, the outputs of the system, which are the change in electrical output power (active power) and the change in terminal voltage. Each output is time delayed and the outputs of the plant depend on previous values of outputs. The global optimal values of the controller weights to make the outputs tracking the inputs of the plant are obtained using genetic algorithms.

4-Genetic Algorithms

Genetic algorithms (GAs) are stochastic search methods that mimic the metaphor of natural biological evolution. GAs model natural processes, such as selection, recombination (crossover), mutation. Figure (5) shows the structure of a simple evolutionary algorithm. GAs work on populations of individuals instead of single solutions. In this way the search is performed in a parallel manner [13].
GAs are probabilistic search techniques loosely based on the Darwinian principle of evolution and natural selection [14]. For maximization (or minimization) of a function $f(x)$ for $x \in [a; b]$, the argument x is represented as a binary string called a chromosome Scaling in x may be necessary so that the range $[a; b]$ is covered. A population is a set of chromosomes representing values of x that are candidates for the desired x that gives the maximum or minimum $f(x)$. Each chromosome has a fitness that is a numerical value which must be minimized.

5- Genetic operators

i- Selection

As with all genetic algorithm operators, there exist a number of selection operators in the literature. However, the standard two methods are described below. The basic operation of the selector is to choose individuals which will constitute a number of individuals which will be available for crossover and mutation and therefore progression to the next generation.

a-Tournament Selection

Two individuals are selected at random from the population, and whichever has the highest fitness is chosen to enter the mating pool.

b-Roulette Wheel Selection

Individuals are chosen at random from the population for entrance to the mating pool. However, the probability of being selected is allocated proportionally according to the fitness of the individual. A very fit individual therefore has more "numbers" on the roulette wheel and therefore has a higher chance of selection.

c-Elitism

An important part of the selection process is that every individual, no matter how unfit, has a chance of progression into the next generation. Generally speaking, fit individuals are required, but diversity must also be maintained in the population and selection is part of the mechanism of ensuring this. Many algorithms make use of the concept of elitism however, where a certain proportion of the very fit solutions are used to make up the next generation unchanged. There are a variety of arguments for and against this process. However, in this study elitist strategy is used.
Once selected, an individual is combined with another, their genetic material is

combined and the subsequent new individuals are made. This process is known as crossover.

ii-Crossover

Crossover takes two or more individuals (collectively known as parents) and creates two or more new individuals (known as children). This process combines the genetic information of the already fit parent individuals in an effort to increase the fitness of the child solutions. As before, there are many crossover operators in the literature for example single point, two points, uniform crossover…..etc.

iii- Mutation

The mutation operator changes chromosome values in the child solutions at random. This random mutation ensures that the algorithm can overcome local minima and find globally good solutions for the problem. It is chiefly this random behaviour which makes the algorithm non-deterministic and therefore unlikely to find the same solution in subsequent algorithm runs.

6- Design of neuro-genetic controller

The choice of suitable performance index is extremely important for the design of power system controller. The controller that collects data about the behaviors of the plant for a certain interval of time is propagated through the neurons of controller to give a proper action to the plant. The outputs of the identifier are compared with the reference inputs. The performance index or cost function (fitness) is minimized. For two inputs, two outputs system the fitnessis:-

$$\text{MSE}=\frac{1}{N}\sum_{i=1}^{N}\ (\Delta P_m-\Delta P_e)^2+(\Delta V_{ref.}-\Delta V_t)^2$$

where N is number of samples. The connection weights of a controller are represent as string of variables [x1,x2,…..,xi] where i represent the number of variables. Genetic algorithms optimization tools in Matlab package is used to minimize the objective function (fitness) which, is function of the variables (weights) of the controller as shown in figure (6).

GAs are probabilistic search techniques loosely based on the Darwinian principle of evolution and natural selection [14]. For maximization (or minimization) of a function $f(x)$ for $x \in [a; b]$, the argument x is represented as a binary string called a chromosome Scaling in x may be necessary so that the range $[a; b]$ is covered. A population is a set of chromosomes representing values of x that are candidates for the desired x that gives the maximum or minimum $f(x)$. Each chromosome has a fitness that is a numerical value which must be minimized. A computer program has been written in Matlab to specify the objective function to be used with genetic algorithm tool G.A.T.The operators are given in table 1

Table 1 GA operators

Population size	20
no. of individuals	134
crossover rate	0.8
mutation rate	0.01
elite	2

7- Discussion

The variables of objective function which each one represent a controller is calculated and and the genetic algorithms optimize the controller. Figure (7), show values of the best individuals and best fitness changing during the generations. The open-loop response for the system, which represent the dynamic behaviors is shown in figure (8).The parameters of the controller are obtained such that the outputs track the inputs, these variables are introduced into a program to obtain the closed-loop response. There is a significant improvement in the behaviors of the system like peak overshoot, rise time and steady state error.

Conclusions

The traditional approach to building system controllers requires a prior model of the power system. The quality of the model, that is, loss of precision from linearization and/or uncertainties in the system's parameters negatively influences the quality of the resulting control. The methods of soft computing such as neural networks possess non-linear mapping capabilities, do not require an analytical model and can deal with uncertainties in the system's parameters. Combined with the evolutionary learning (such as genetic algorithms) these methods are capable of producing global-optimal controllers for a given control task. For example, genetic algorithms have been used to produce parameters of an optimized system controller such as weights of a neural network controller cascaded with the identified system to damp the dynamic performance and enhance the active power and voltage deviations of a power system due to step changes in inputs.

References

[1] H. Saadat,Power System Analysis,Mc-Graw-HillInc.,pp.460,1999.

[2] Adkins B.,Harley R.G., " The general theory of alternating current machines " ,Champon and Hall,London,1975,ISBN0-412-15560-5.

[3] Hunt K.G.,Sbarbaro D,Zbikowski R,Gawthrop P.J., "Neural Networks for Control Systems,-a survey",Automatica, Vol.28,No.6,pp 1083-111,1992.

[4] Venayagamoorthy GK,Harey R.G. "Experimental Studies with a Continually Online Artificial Neural Network Controller for turbogenerator", IEEE pp 2158-2163,1999.

[5] Asiedu Y., R.W. Besant and P. Gu, "HVAC duct system design using genetic algorithms", HVAC&R Research, Vol.6, No.2, pp.149-173,2000.

[6] D. E. Goldberg, Genetic Algorithms in Search,Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.

[7] J. Holland, Adaptation In Natural and Artificial Systems, The University of Michigan Press, Ann Arbour, 1975.

[8] P.M. Anderson and A.A.Fouad, " Power System Control and Stability",The lowa state university,press,Amess,Loua,1977.

[9] K.T.Law,D.J.Hill,and N.R. Godfrey' "Robust Controller Structure for Coordinated Power System Voltage Regulator and Stabilizer Design"IEEE Trans.on control system technology,No.3,pp.220-232,1994.

[10] P. Kundur, "Power System Stability and Control", McGraw-Hill Inc,1994.

[11] K. S. Narendra and K. Parthasarathy, " Identification and Control of Dynamical

Systems using Neural Networks",IEEE Trans., On Neural Networks, vol.1, Mar. 1990, pp 4-27

[12] G.K. Venayagamoorthy and R.G. Harley, "Implementation of an adaptive neural network identifier for effective control of turbo-generators", Electric Power Engineering, IEEE,PowerTech. Conference,pp. 134-139.1999

[13] H. Pohlheim , " Genetic and Evolutionary Algorithm Toolbox for use with Matlab" version 3.5a (July 2004) .

[14] C.R. Reeves and J.W. Rowe. Genetic Algorithms Principles and Perspective.A Guide to GA Theory. Kluwer, Boston, 1997.

Appendix

Synchronous generator: [8]

Rated MVA $S=160\ MVA$

Rated kV $15\ kV$

$X_d=0.245,\ X_d'=0.185,\ X_d''=0.64,\ X_q=0.385,$

$X_q'=0.185,\ X_q''=0.64, r_a=1.64, X_l=0.0031,$ *[pu]*

$T'_{d0}=5.9, T''_{d0}=0.033, T'_{q0}=0.54, T''_{q0}=0.076,$ *Al time constants are in sec.*

Exciter: *gain* $k_e=1$ $T_e=1sec.$

Governor & Turbine gains $k_g=1, k_t=1$

$T_g=0.2,\ T_t=0.5$ *sec.*

$K_1=1.5, K_2=0.2,\ K_3=0.8,\ K_4=1.4,\ K_5=-0.1, K_6=0.5$

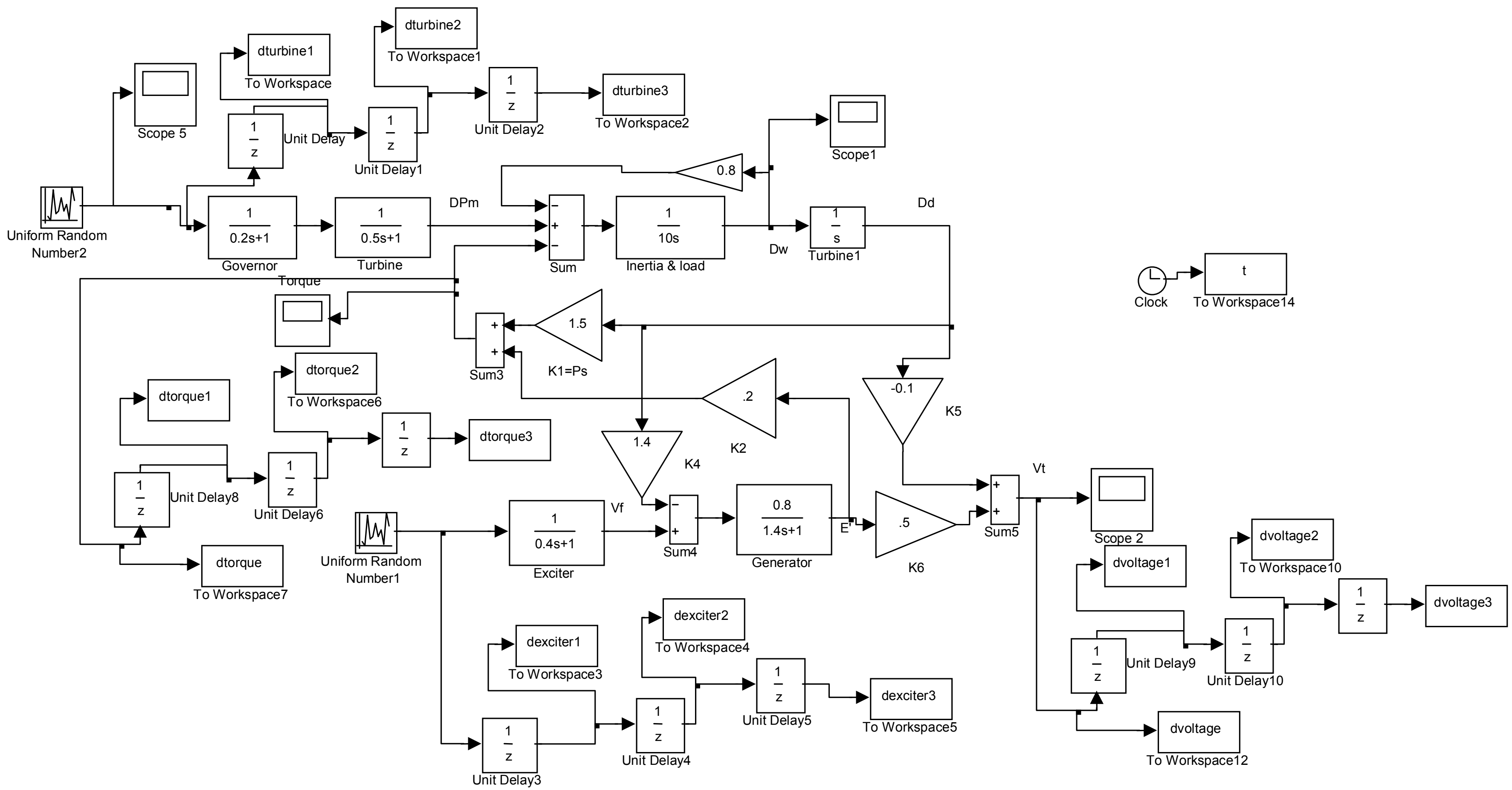

Fig. 1. Simulink Model of the plant

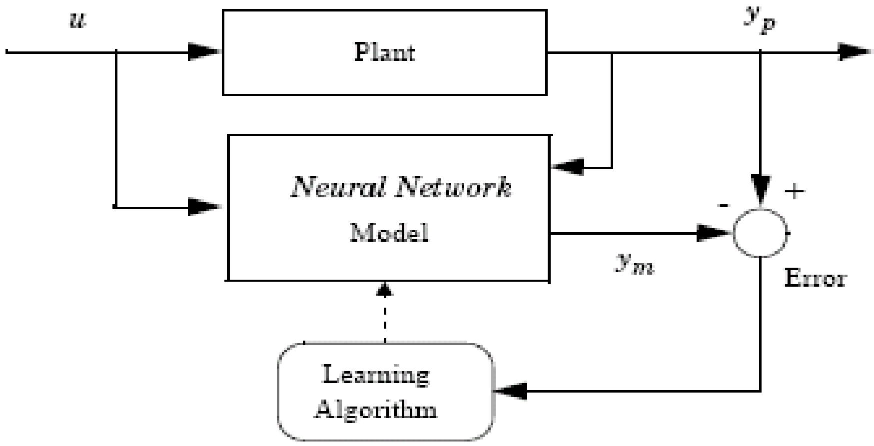

Fig. 2. ANN and Plant Block Diagram for learning

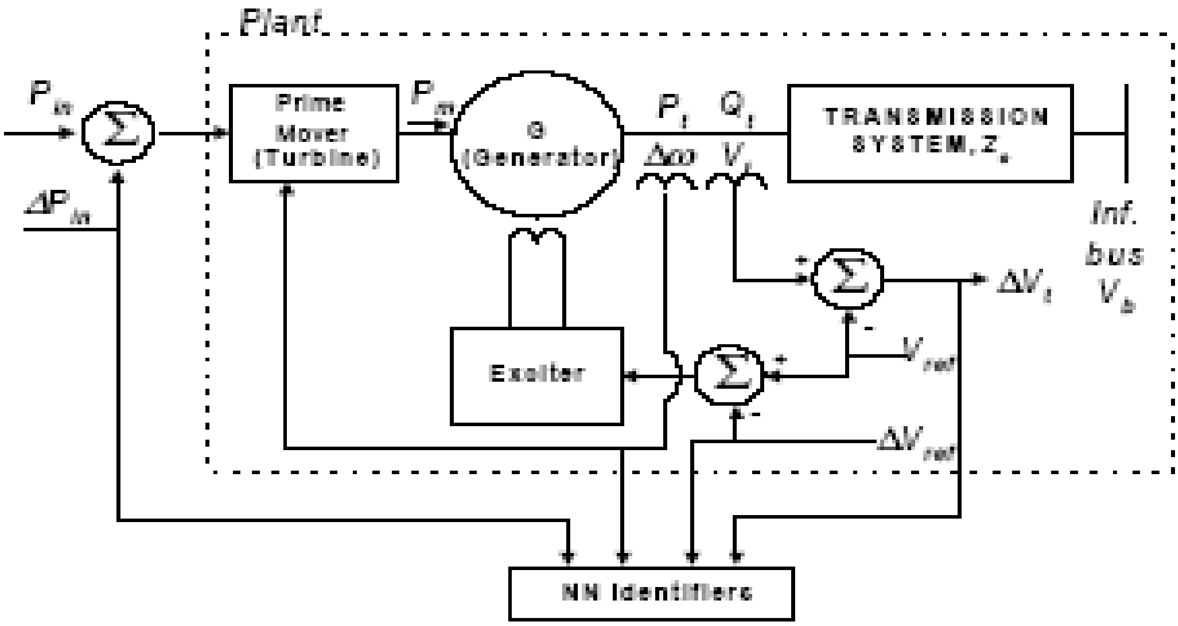

Fig. 3. Plant model used for identification of synchronous Generator connected to infinite bus

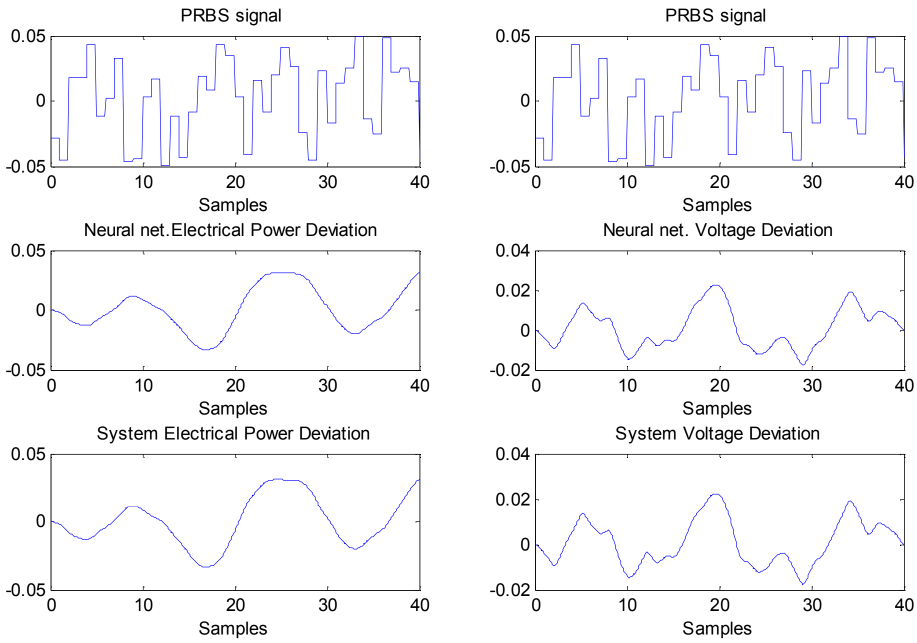

Fig. 4a Plant and Neural Net. Identifier outputs

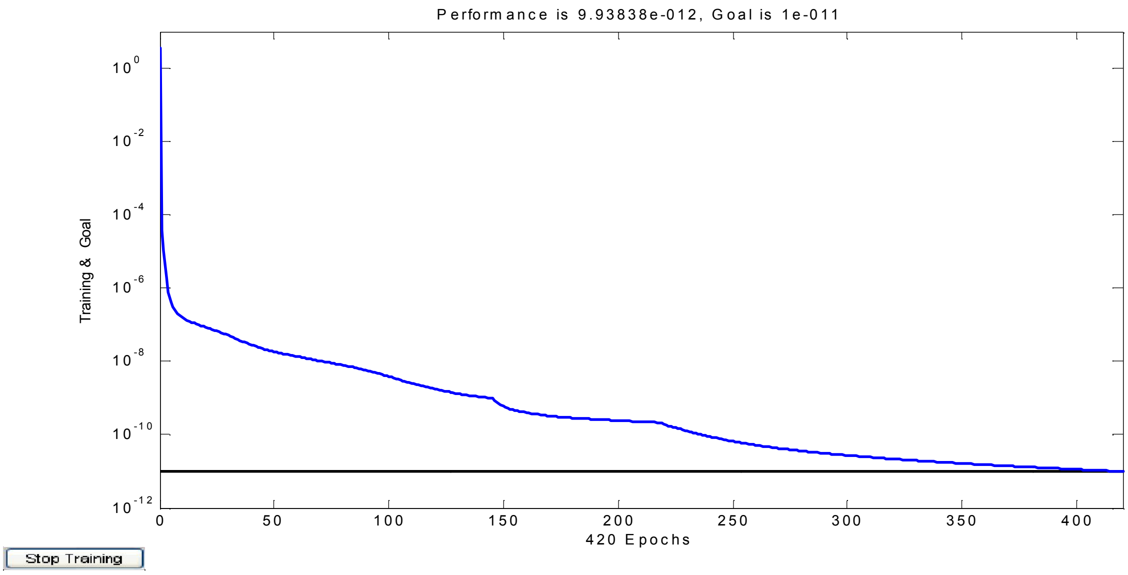

Fig. 4b Error performance of ANN training network

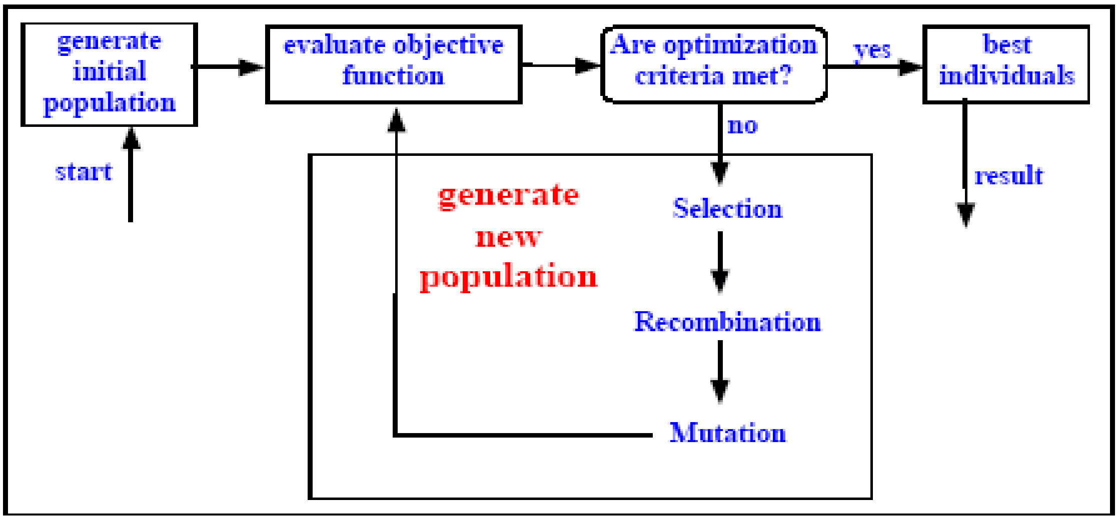

Fig. 5 Structure of single population Genetic Algorithm

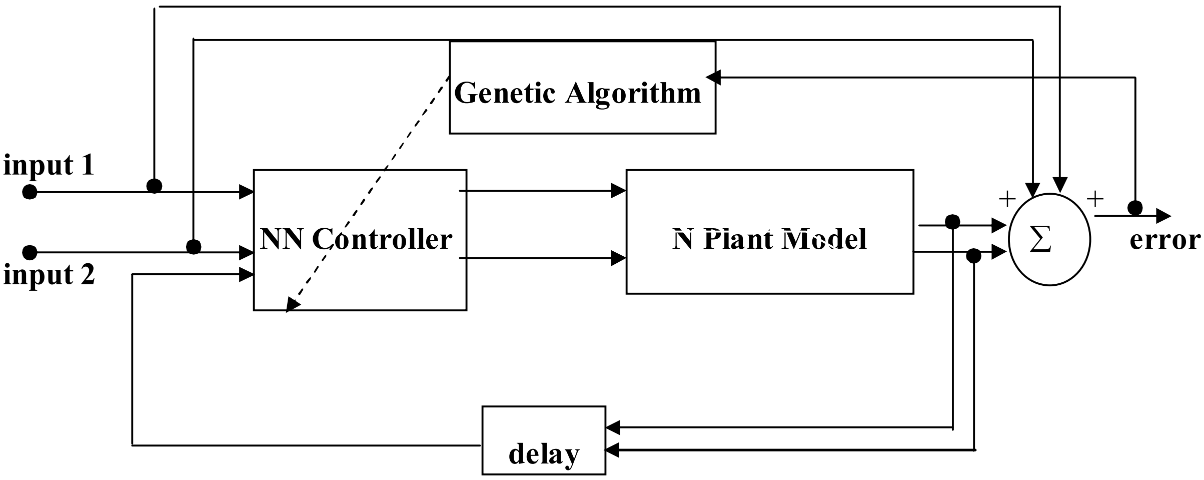

Fig. 6 Neuro-genetic controller of power system plant