01 Jan 2005

MLP/RBF Neural-Networks-Based Online Global Model Identification of Synchronous Generator

Jung-Wook Park

Ganesh K. Venayagamoorthy
Missouri University of Science and Technology

Ronald G. Harley

Recommended Citation

J. Park et al., "MLP/RBF Neural-Networks-Based Online Global Model Identification of Synchronous Generator," *IEEE Transactions on Industrial Electronics*, Institute of Electrical and Electronics Engineers (IEEE), Jan 2005.
The definitive version is available at https://doi.org/10.1109/TIE.2005.858703

MLP/RBF Neural-Networks-Based Online Global Model Identification of Synchronous Generator

Jung-Wook Park, *Member, IEEE,* Ganesh Kumar Venayagamoorthy, *Senior Member, IEEE,* and Ronald G. Harley, *Fellow, IEEE*

***Abstract*—This paper compares the performances of a multilayer perceptron neural network (MLPN) and a radial basis function neural network (RBFN) for online identification of the nonlinear dynamics of a synchronous generator in a power system. The computational requirement to process the data during the online training, local convergence, and online global convergence properties are investigated by time-domain simulations. The performances of the identifiers as a global model, which are trained at different stable operating conditions, are compared using the actual signals as well as the deviation signals for the inputs of the identifiers. Such an online-trained identifier with fixed optimal weights after the global convergence test is needed to provide information about the plant to a neurocontroller. The use of the fixed weights is to provide against a sensor failure in which case the training of the identifiers would be automatically stopped, and their weights frozen, but the control action, which uses the identifier, would be able to continue.**

***Index Terms*—Global model, multilayer perceptron neural network (MLPN), nonlinear dynamic system, online identification, radial basis function neural network (RBFN), synchronous generator.**

I. Introduction

A synchronous generator in a power system is a nonlinear fast-acting multiple-input multiple-output (MIMO) device [1], [2]. Due to its wide operating range, complex dynamics, nonlinearity, and the changing system configuration, the entire system cannot be accurately represented by a fixed model, which is then used for the design of conventional linear system/controllers [3]. Both the model and the controller have to change and adapt in order to achieve the best performance by the controller as system conditions change. Artificial neural networks (ANNs) offer an alternative. They are able to adaptively model or identify such a nonstationary nonlinear MIMO process/plant online while the process is changing, and thereby yield information that can be used by another ANN to control the process.

This paper focuses on the identifier only. It makes a new contribution by comparing the multilayer perceptron neural network (MLPN) and a radial basis function neural network (RBFN) for the online identification of a synchronous generator in an electric power system. It evaluates these two identifier networks first when they use deviations of the measured signals from their respective set points, and then when they use the actual measured signals. The paper also gives a more detailed explanation of why the MLPN (rather than the RBFN) was used as the model network (identifier) for the design of optimal neurocontrollers based on the adaptive critic designs (ACDs), which was reported in the authors' recent technical papers [3]–[5].

In addition to the computational requirements to process the data, the local and global convergence properties of the online identifiers are evaluated after they have undergone continuous online training for a period of time. Two training methods by which the identifiers' weights are trained using the actual signals as well as the deviation of signals from their set points over a wide range of stable operating conditions [such an identifier is called a global model identifier (GMI)] are presented (the more detailed explanation for the term "global model" is given in Section III-B).

Manuscript received January 13, 2004; revised March 12, 2005. Abstract published on the Internet September 26, 2005.

J.-W. Park is with the School of Electrical and Electronic Engineering, Yonsei University, Seoul 120-749, Korea (e-mail: jungpark@yonsei.ac.kr).

G. K. Venayagamoorthy is with the Department of Electrical and Computer Engineering, University of Missouri, Rolla, MO 65409 USA (e-mail: ganeshv@ece.umr.edu).

R. G. Harley is with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0250 USA (e-mail: rharley@ece.gatech.edu).

Digital Object Identifier 10.1109/TIE.2005.858703

II. Adaptive Neural Network Identifiers for Synchronous Generator

A. Plant Modeling

In Fig. 1, the synchronous generator, turbine, exciter, and transmission system are connected to a source of fixed voltage magnitude and frequency (called an infinite bus) and form the plant (dashed block in Fig. 1) that has to be identified. Although, in practice, a power network is more complicated, this simplified system in Fig. 1 is chosen to illustrate the concepts described in this paper. The nonlinear seventh-order dynamic model for the generator (**G** in Fig. 1) is described by $d-q$ axis equations, with the machine current, speed, and rotor angle as the state variables, and has $d-q$ damper windings [1], [2]. These mathematical models are given in (1)–(3). For voltage $d-q$ equations

$$\begin{aligned} v_d &= R_a i_d + \frac{d\lambda_d}{dt} - \omega\lambda_q \\ v_q &= R_a i_q + \frac{d\lambda_q}{dt} + \omega\lambda_d \\ v_{fd} &= R_f i_f + \frac{d\lambda_f}{dt} \\ v_{kd} &= 0 = R_{kd} i_{kd} + \frac{d\lambda_{kd}}{dt} \\ v_{kq} &= 0 = R_{kq} i_{kq} + \frac{d\lambda_{kq}}{dt} \end{aligned} \tag{1}$$

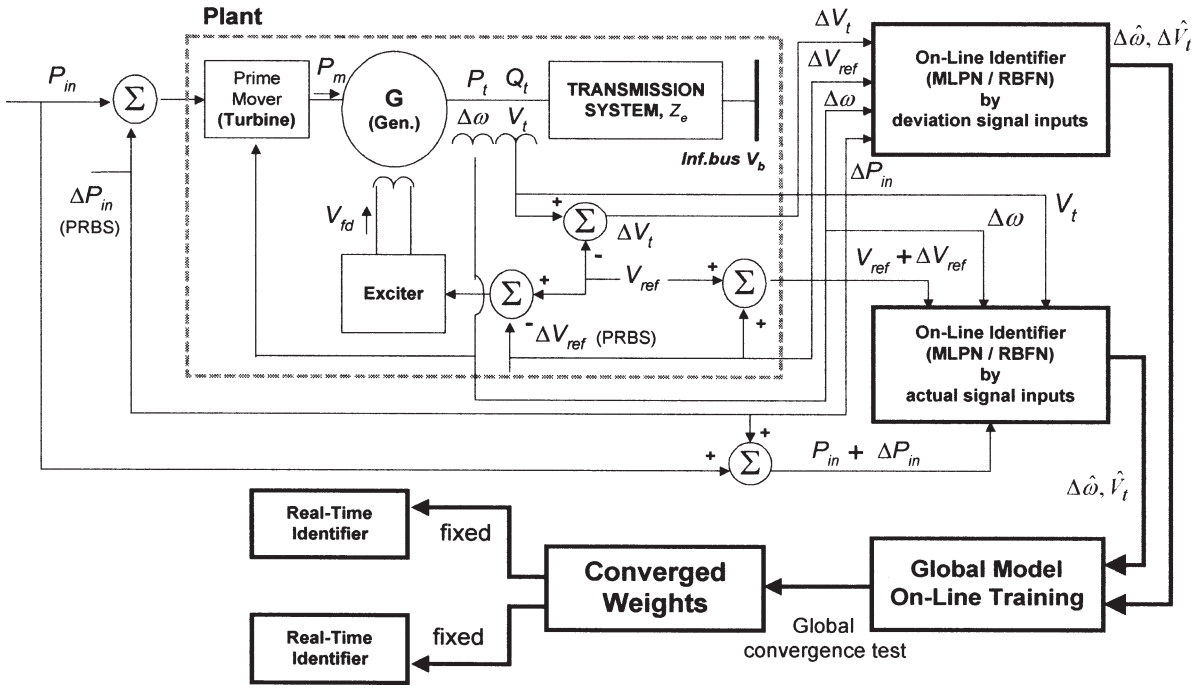

Fig. 1. Plant model used for online identification: synchronous generator connected to an infinite bus, which represents all the other generators by a voltage source V_b.

where

$$\begin{aligned}\lambda_d &= L_d i_d + L_{md} i_f + L_{md} i_{kd}\\ \lambda_f &= L_{md} i_d + L_{ff} i_f + L_{md} i_{kd}\\ \lambda_{kd} &= L_{md} i_d + L_{md} i_f + L_{kkd} i_{kd}\\ \lambda_q &= L_q i_q + L_{mq} i_{kq}\\ \lambda_{kq} &= L_{mq} i_q + L_{kkq} i_{kq}.\end{aligned} \tag{2}$$

The rotor angle, rotor speed, and electromagnetic torque are described by

$$\begin{aligned}\frac{d\delta}{dt} &= \omega - \omega_o\\ \frac{d\omega}{dt} &= \frac{1}{J}(T_s - T_e)\\ T_e &= \left(-\frac{\omega_o}{2}\right)[i_d i_q (L_d - L_q) + i_f i_q L_{md}\\ &\qquad + i_q i_{kd} L_{md} - L_{mq} i_{kq} i_d].\end{aligned} \tag{3}$$

In (1) and (2), $(v_d/v_q, i_d/i_q)$ and $(v_{kd}/v_{kq}, i_{kd}/i_{kq})$ are the $d-q$ axis components of (voltage, current) in the armature and damper windings, λ_d/λ_q and $\lambda_{kd}/\lambda_{kq}$ are the $d-q$ axis components of armature and damper flux linkages, and λ_f and i_f are the field flux linkage and field current. R_a, R_f, and R_{kd}/R_{kq} are resistances of armature, field, and $d-q$ damper windings, respectively. L_d/L_q, L_{kkd}/L_{kkq}, and L_{md}/L_{mq} are the $d-q$-axes components of self-inductance, damper self-inductance, and mutual inductance, respectively. In (3), δ is the rotor angle with reference to the infinite bus, ω_o is the synchronous speed in steady state, T_s and T_e are turbine output shaft torque and generator electromagnetic torque, and J is the polar moment of inertia. By substituting the flux linkage $d-q$ equations in (2) into the voltage equations in (1), and using the equations in (3), the seventh-order set of differential equations can be found [1] and [2] for the dynamic model of the generator $(\mathbf{G})$.

The voltage equations for the transmission system (in Fig. 1) are given in (4), which are derived from the methods of circuit and Park's transformation theory [1], [2]

$$\begin{aligned}v_{td} &= V_{bd}\sin\delta - R_e i_d - L_e\frac{di_d}{dt} + \omega L_e i_q\\ v_{tq} &= V_{bq}\cos\delta - R_e i_q - L_e\frac{di_q}{dt} - \omega L_e i_d\end{aligned} \tag{4}$$

where v_{td} and v_{tq} are the $d-q$-axes components of terminal voltage, V_{bd} and V_{bq} are the $d-q$-axes components of the infinite bus voltage, and R_e and L_e are the resistance and inductance of the transmission line. Furthermore, the general IEEE exciter and turbine models are used for the plant in Fig. 1. Their transfer functions and block diagrams are given in [2]. The exciter is represented by a simple first-order model. The third-order turbine model provides for reheating between the high pressure and intermediate pressure stages; the output of the turbine is limited between 0% and 120%. The nonlinear equations given in (1)–(4), together with the general IEEE exciter and turbine model in [2], are used to describe the dynamics of the plant in order to generate the data by simulation for the ANN identifiers [3]. (In practice, these data would be measured on the physical system).

In the plant, P_t and Q_t are the real and reactive power at the generator terminals, respectively. Z_e is the transmission line

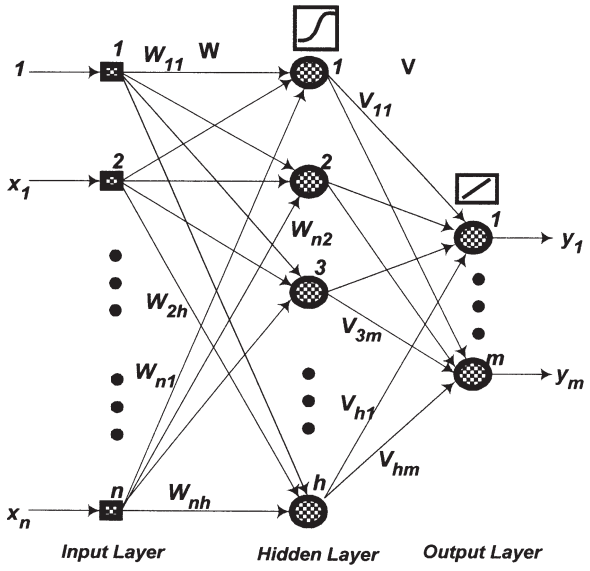

Fig. 2. MLPN.

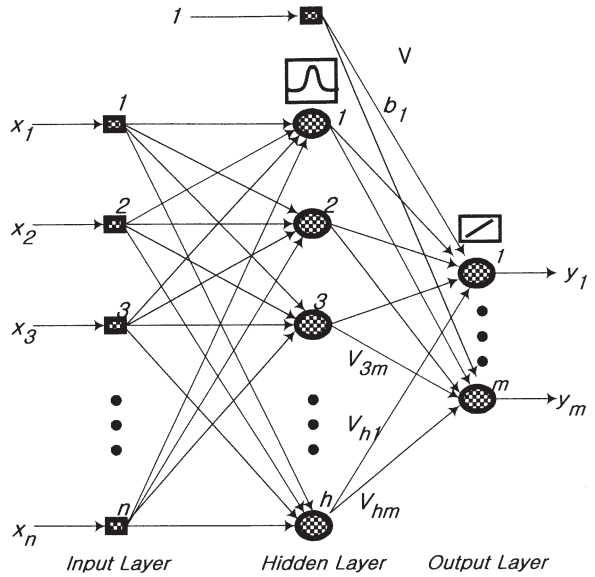

Fig. 3. RBFN.

impedance, P_m is the mechanical input power to the generator, V_{fd} is the generator field voltage, V_b is the infinite bus voltage, $\Delta\omega$ is the generator speed deviation, ΔV_t is the generator terminal voltage deviation, V_t is the terminal voltage, $\Delta V_{\rm ref}$ is the reference voltage deviation, $V_{\rm ref}$ is the terminal voltage reference value, $\Delta P_{\rm in}$ is the turbine input power deviation, and $P_{\rm in}$ is the turbine input power. The $\Delta P_{\rm in}$ and $\Delta V_{\rm ref}$ signals (inputs) in Fig. 1 are small pseudorandom binary signals (PRBSs) [3] applied to perturb the plant in order to measure the speed deviation $(\Delta\omega)$ and terminal voltage deviation (ΔV_t) (outputs) responses. These outputs are important indicators of generator damping performance. The same sequence of PRBSs is injected to both the MLPN and RBFN identifiers in order to ensure a fair comparison. The magnitude of these PRBSs is limited to a maximum of $\pm 10\%$ of their nominal values.

B. Multilayer Perceptron Neural Network

In this paper, the MLPN consists of three layers of neurons (input, hidden, and output layers as shown in Fig. 2) interconnected by the input and output weight matrices $\mathbf{W}$ and $\mathbf{V}$, respectively. The weights of the MLPN are obtained using the backpropagation algorithm [6]. The activation function for neurons in the hidden layer is the following sigmoidal function

$$h(x) = \frac{1}{1+\exp(-x)}. \tag{5}$$

During online training, the MLPN starts with small random initial values for its weights, and then computes a one-pass backpropagation algorithm at each time step k, which consists of a forward pass propagating the input vector through the network layer by layer, and a backward pass to update the weights by the gradient descent rule. The output layer neurons are formed by the inner products between the nonlinear regression vector from the hidden layer and the output weight matrix $\mathbf{V}$. The inner weights $(\mathbf{W})$ for the MLPN are updated by (6)

$$\begin{aligned}\Delta \mathbf{W}_l &= \frac{\partial \boldsymbol{J}}{\partial \mathbf{W}_l} = \frac{\partial \boldsymbol{J}}{\partial t_A}\frac{\partial t_A}{\partial p_L}\frac{\partial p_L}{\partial q_L}\frac{\partial q_L}{\partial p_l}\frac{\partial p_l}{\partial q_l}\frac{\partial q_l}{\partial \mathbf{W}_l} \\ &= \left[\{h(q_l)\,(1-h(q_l))\,\boldsymbol{X}\}\sum_{j=1}^{m_l} 1\cdot \mathbf{W}_L\right] \end{aligned} \tag{6}$$

where

- $\boldsymbol{J}(k)$ $= (1/2)\sum_j [\boldsymbol{E}_j(k)]^2$, where $\boldsymbol{E}(k)$ is the error vector between outputs of the plant and MLPN, and k indicates discrete sampling time;
- t_A target value;
- L, l output and hidden layers, respectively;
- m_l number of neurons in the hidden layer;
- p output of the activation function for a neuron;
- q regression vector as the activity of a neuron;
- $\boldsymbol{X}$ input vector of the MLPN.

The function h is the sigmoidal function in (5).

By trial and error, 14 neurons in the hidden layer are optimally chosen for the online identification. These values depend on a tradeoff between convergence speed and accuracy.

C. Radial Basis Function Neural Network

For a fair comparison, the RBFN also consists of three layers (see Fig. 3). However, the input signals are each assigned to a node in the input layer and then passed directly to the hidden layer without weights. The hidden layer nodes are called radial basis function (RBF) units, defined by a parameter vector called the "center" and a scalar called the "width." The Gaussian density function is used in the hidden layer as an activation function. The linear weights v_{ji} between the hidden and output layers are solved or trained by a linear least squares

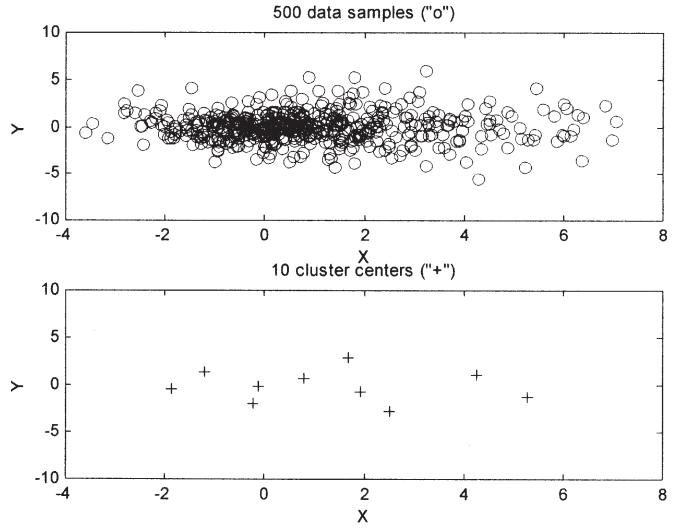

Fig. 4. Demonstration for 2-D clustering centers by the batch mode k-means clustering algorithm.

optimization algorithm [6], [7]. The overall input–output mapping for the RBFN $\hat{f} : \boldsymbol{X} \in \mathbf{R}^n \rightarrow \boldsymbol{Y} \in \mathbf{R}^m$ is

$$y_i = b_i + \sum_{j=1}^{h} v_{ji} \exp\left(-\frac{\|\boldsymbol{X} - \mathbf{C}_j\|^2}{\beta_j^2}\right) \tag{7}$$

where $\boldsymbol{X}$ is the input vector, $\mathbf{C}_j \in \mathbf{R}^n$ is the jth center of the RBF unit in the hidden layer, h is the number of RBF units, b_i and v_{ji} are the bias terms and the weight between hidden and output layers, respectively, and y_i is the ith output in the m-dimensional space. Once the RBF centers are established, the width β_i of the ith center in the hidden layer is calculated as

$$\beta_i = \left[\frac{1}{h}\sum_{j=1}^{h}\sum_{k=1}^{n}(\|\mathbf{c}_{ki} - \mathbf{c}_{kj}\|)\right]^{\frac{1}{2}} \tag{8}$$

where $\mathbf{c}_{ki}$ and $\mathbf{c}_{kj}$ are the kth value of the center of the ith and jth RBF units. In (7) and (8), $\|\cdot\|$ represents the Euclidean norm. There are the following four different ways for input–output mapping using the RBFN, depending on how the data is fed to the network [8]:

1) batch mode clustering of centers and batch mode gradient descent for linear weights;
2) batch mode clustering of centers and pattern mode gradient descent for linear weights;
3) pattern mode clustering of centers and pattern mode gradient descent for linear weights;
4) pattern mode clustering of centers and batch mode gradient descent for linear weights.

The suitability of each of the four methods is discussed below.

1) Centers of RBF Units: The pattern mode clustering of centers is not feasible for the online identification because of excessive memory and computational complexity, since the center vectors and their widths have to be adapted with changing input patterns. Hence, the batch mode k-means clustering algorithm is used for determining the centers off-line, instead

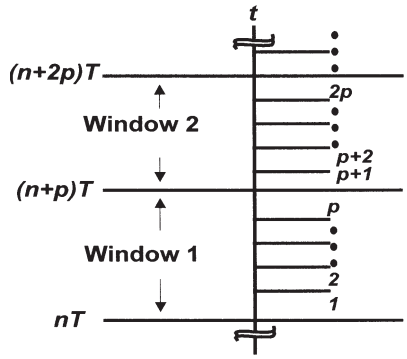

Fig. 5. Memory allocation structure considered for determination of output weights using the RBFN.

TABLE I
GLOBAL CONVERGENCE TEST

Identifiers	Online Training Time	$\Delta\omega$		ΔV_t	
		E_A (%)	MSE	E_A (%)	MSE
MLPN	2800 s	0.2084	1.556×10^{-5}	0.4234	3.783×10^{-5}
RBFN12	1400 s	0.2480	1.100×10^{-5}	0.1189	2.963×10^{-6}

TABLE II
NUMBER OF FLOATING-POINT OPERATIONS AND ELAPSED TIME REQUIRED DURING 0.2 s IN SIMULATION

Identifiers	MLPN	RBFN10	RBFN12	RBFN15	RBFN21
FLOPS	22,920	14,180	15,700	17,980	22,540
Elapsed time	0.6510 s	0.5337 s	0.6210 s	0.6410 s	0.6510 s

of the recursive k-means algorithm [7], [9]. The batch mode k-means clustering algorithm can be described as follows.

Step 1) A large amount of data (referred to as the data set) is generated by simulating the plant (in Fig. 1) behavior at many different operating conditions (see Fig. 6) over the targeted operating range for feature extraction. At each operating condition, 5000 samples are stored for subsequent training of the neural network.

Step 2) Initialize the center of each cluster to represent a different randomly selected data set in step 1).

Step 3) Assign each data set to the cluster nearest to it. This is done by calculating the Euclidean distances between training patterns and centers.

Step 4) When all the data sets are assigned, the mean (average) position is calculated for each cluster center.

Step 5) Repeat steps 3) and 4) until the cluster center changes become acceptably small according to the previously chosen stopping criterion.

Fig. 4 illustrates how the batch mode k-means clustering algorithm forms the ten cluster centers, and each center (indicated by "+") represents the locally supported grouping data set from the 500 data samples (indicated by "o") in a two-dimensional (2-D) space.

2) Output Weights: After the vector of cluster centers and the width of each center are calculated off-line, the values of the weights, v_{ji} in (7) can subsequently be updated during online

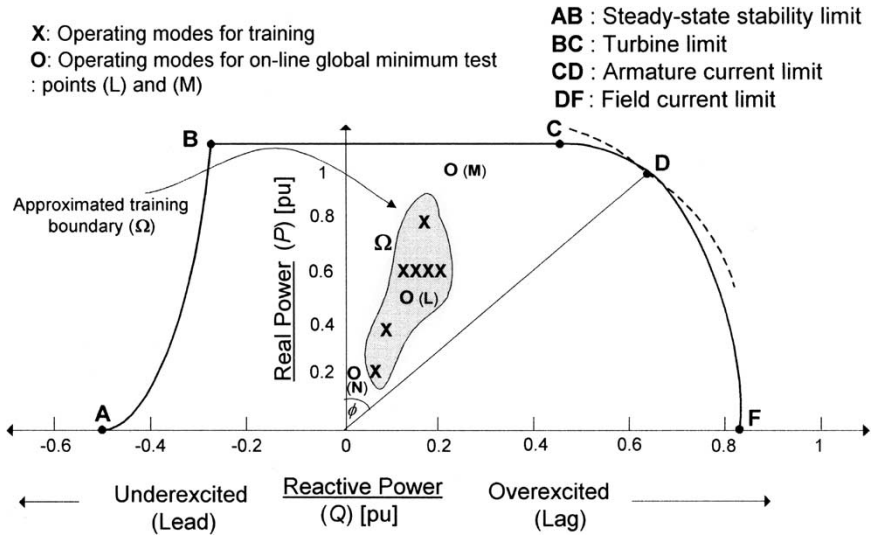

Fig. 6. Synchronous generator capability limits in the $P-Q$ plane.

training by using the linear least square algorithm based on the pattern mode, or the pseudoinverse technique based on the batch mode in the orthogonal least squares (OLS) algorithm. The pseudoinverse technique takes less time for learning and will be more effective for one-step ahead prediction performance than the pattern mode least square algorithm. However, the pseudoinverse technique has a drawback for the generalization or global minimum convergence without further weight updates, as explained below.

Consider the memory allocation structure given in Fig. 5. Window 1 contains the system data from time step nT to $(n+p)T$, and window 2 contains data from $(n+p)T$ to $(n+2p)T$. When the pseudoinverse technique is used, the RBFN identifier processes all the data in each window as a single set such that training for one-step ahead prediction is fast. From Fig. 5, the drawback of the pseudoinverse technique is that it does not have a learning mechanism for generalization, because the weights calculated in each window represent the system information for only that particular window. In other words, information in one window for weights is lost in the other windows. Moreover, the use of data in all windows is not possible for online identification because of memory limitations. Consequently, the batch mode k-means clustering algorithm for the centers of RBF units is computed off-line, and the pattern mode gradient descent algorithm for output weights is calculated online.

III. Global Model Online MLPN/RBFN Identifiers: Simulation Studies

In this paper, the GMIs are trained using the actual signals as well as the deviation signals at a number of different stable operating points before testing their usefulness. The computational requirement required to process data and local/global convergence properties are used to evaluate the convergence performance of the two MLPN- and RBFN-based online identifiers. After having trained online for a period of time, the training error should have converged to a value so small that if training were to stop due to loss of input signals, and the weights frozen, then the neural networks would continue to identify the plant correctly while the operating condition remains fixed; this means that the ANNs have reached local convergence. Furthermore, the training of ANNs is said to have reached global convergence when, after changing the operating conditions as well as freezing the weights, the network's response is still reasonably acceptable [10] in terms of some predefined convergence error metric. At this point, the ANN-based identifiers are evaluated as to how well they have converged by measuring the average absolute convergence error (E_A) and the mean square convergence error (MSE) in (9)

$$\begin{aligned} e(k) &= y(k) - \hat{y}(k) \\ E_A &= \frac{1}{N}\sum_{k=1}^{N} |e(k)| \\ \text{MSE} &= \frac{1}{N}\sum_{k=1}^{N} e(k)^2 \end{aligned} \tag{9}$$

where N is the number of training data samples and $e(k)$ is the error variable between the output (y) from the plant and estimated output $(\hat{y})$ from the identifiers at each sample k. As an illustration, the simulation results during training (carried out for the test of Section III-B) are given in Table I.

A. Number of Floating-Point Operations

During 200 ms of online training time (with the actual sampling time of 20 ms), the computational requirement of the identifiers (MLPN/RBFN) to process the data set in simulation is measured by calculating the number of FLoating point Operations Per Second (FLOPS) and the elapsed computer time needed for the calculation. For the RBFN, the number of FLOPS depends on the number of RBF unit centers, which contain information about the targeted entire operating range. In Table II, the RBFNs with 10, 12, 15, and 21 centers are denoted RBFN10, RBFN12, RBFN15, and RBFN21, respectively, for convenience of presentation. These results clearly show that the RBFNs require less FLOPS and less time than

TABLE III
OPERATING CONDITIONS DATA SET FOR GLOBAL MODEL

Operating conditions	P_t (pu)	Q_t (pu)	P_{in} (pu)	δ (rad/s)	Z_e (pu)
E	0.2	0.0853	0.2001	0.3877	0.02+j0.4
F	0.4	0.0983	0.4003	0.7167	0.02+j0.4
G	0.6	0.1270	0.6007	0.9678	0.02+j0.4
H	0.8	0.1720	0.8013	1.1578	0.02+j0.4
I	0.6	0.1273	0.6007	1.0270	0.025+j0.5
J	0.6	0.1211	0.6007	1.0312	0.03+j0.5
K	0.6	0.1339	0.6007	1.0823	0.03+j0.6
L	0.5	0.1106	0.5005	0.8513	0.02+j0.4
M	1.0	0.2340	1.002	1.3069	0.02+j0.4
N	0.2	0.0084	0.2001	0.5250	0.025+j0.75

the MLPN. Further simulations, which are not included in Table II (in order to save space), show that a further increase in the number of RBF centers does not necessarily yield better performance with online training.

The absolute value of time used in the practical implementation by dedicated digital signal processor hardware will be much less than the values shown in Table II. By trial and error, the online identification performance by the RBFN12 showed the best results in terms of mean square error compared to RBFNs with 10, 15, and 21 RBF centers. Therefore, the RBFN12 is now chosen for a further comparison with the MLPN, which has 14 neurons in the hidden layer, in the following evaluations.

B. Global Model Identifiers Trained by Deviation Signal Inputs

For the deviation signal inputs based online identifiers, the following associated vectors with the variables of the nonlinear autoregressive moving average with exogenous inputs (NARMAX) model [3], [6], [10] are used:

1) reference vector into the plant, $Ref(k) = [P_{\rm in}(k), V_{\rm ref}(k)]$;
2) input vector to the plant or identifiers, $\boldsymbol{U}(k) = [\Delta P_{\rm in}(k), \Delta V_{\rm ref}(k)]$;
3) output vector of the plant, $\boldsymbol{Y}(k) = [\Delta\omega(k), \Delta V_t(k)]$;
4) input vector to the identifiers, $\boldsymbol{X}(k) = [\boldsymbol{Y}(k-1)\,\boldsymbol{U}(k-1)\,\boldsymbol{Y}(k-2)\,\boldsymbol{U}(k-2)\,\boldsymbol{Y}(k-3)\,\boldsymbol{U}(k-3)]^{\rm T}$;
5) output vector of the identifiers, $\hat{\boldsymbol{Y}}(k) = [\Delta\hat{\omega}(k), \Delta\hat{V}_t(k)]$.

The $P-Q$ plane enclosing all the safe operating modes within the specified limits of field current, stator current, prime mover power, and steady-state stability [11] is shown in Fig. 6, showing the approximate training boundary region (Ω) in the first quadrant [overexcited generator operation, power factor (pf) lagging] of the $P-Q$ plane for the global model online identification. A total of seven operating conditions E to K within Ω are used for training and appear in Table III. The eighth operating condition L [point O (L) in Fig. 6] is used for the online global convergence test (after the weights of the identifiers have been frozen) at a point in the region of the

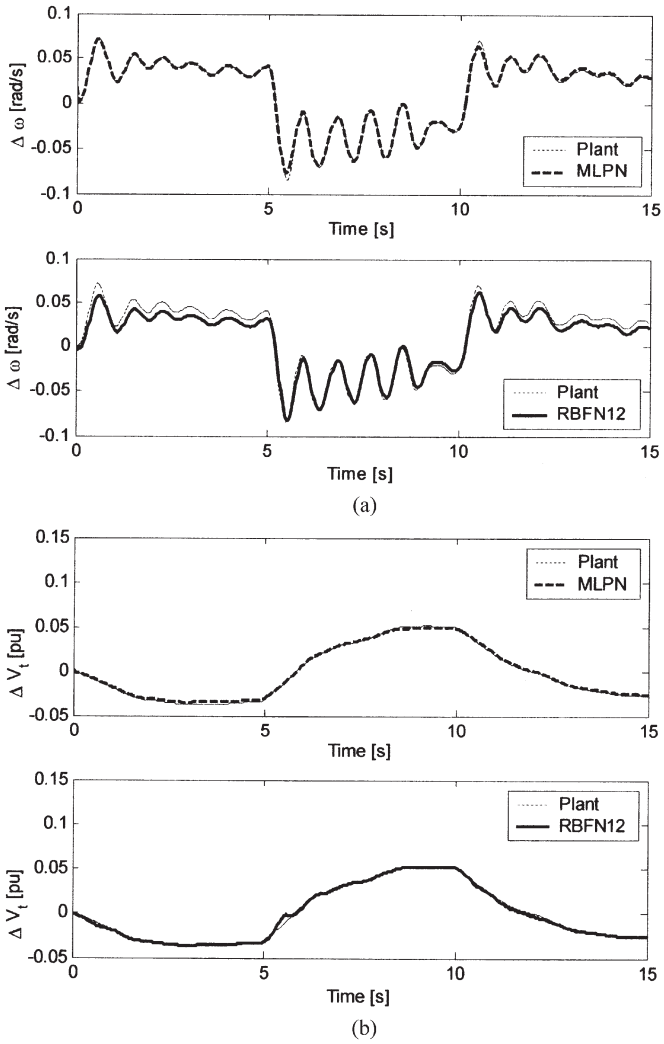

Fig. 7. Deviation signal inputs based online global convergence test. (a) Speed deviation ($\Delta\omega$). (b) Terminal voltage deviation (ΔV_t).

training set. online training starts with random weights and continues for 3000 s for the two identifiers (MLPN and RBFN12) at operating conditions changing from E to K (Table III); each operating condition is randomly selected every 10 s, but the shuffled sequence is the same for the two identifiers. Thereafter, the weights are frozen and the two identifiers are tested for global convergence at the operating condition E in the training set (this is local convergence test) from $t = 0$ s to $t = 5$ s, and at the different operating condition L [point O (L) in Fig. 6] (this is global convergence test) from $t = 5$ s to $t = 10$ s, and again at the initial operating point E from $t = 10$ s to t = 15 s.

The result in Fig. 7 shows that the MLPN identifier has better local convergence performance than RBFN12. Both identifiers show the good global convergence performance.

The PRBS signals are now removed, the identifiers' weights are still fixed, and a 100-ms three-phase short circuit is applied at the infinite bus in Fig. 1 at $t = 0.3$ s, in order to prove that the identifiers are correctly tracking even with large changes in variables. The operating point at which this short circuit is applied is the condition O (M) in Fig. 6, which is not within the set used for the training of the two identifiers (MLPN and RBFN12) and is at approximately 25% more active power than

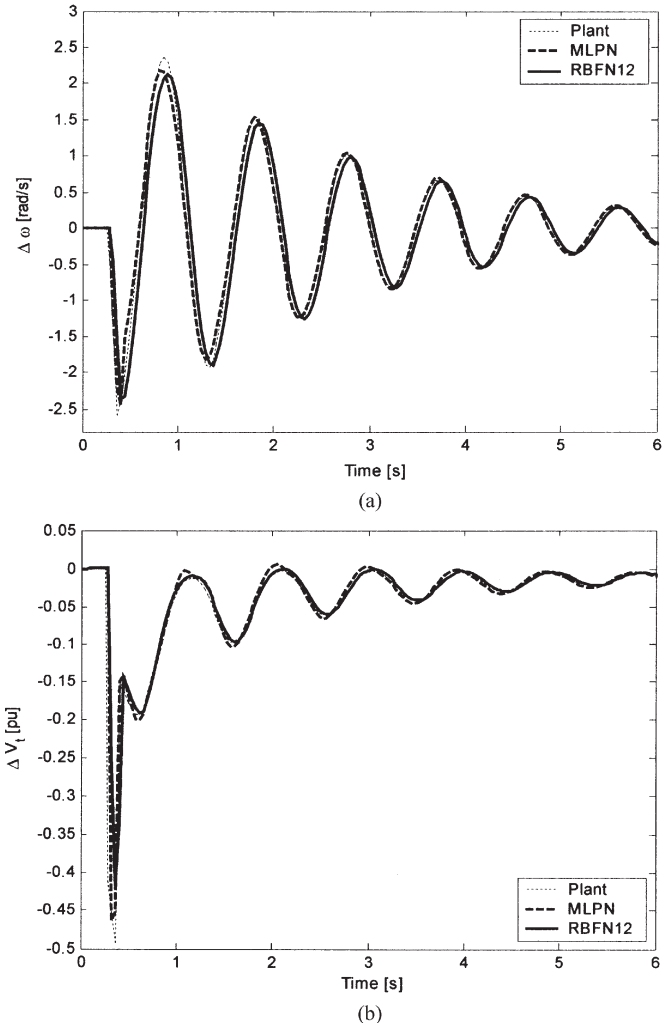

Fig. 8. Online global convergence test by a 100-ms three-phase short circuit. (a) Speed deviation $(\Delta\omega)$. (b) Terminal voltage deviation (ΔV_t).

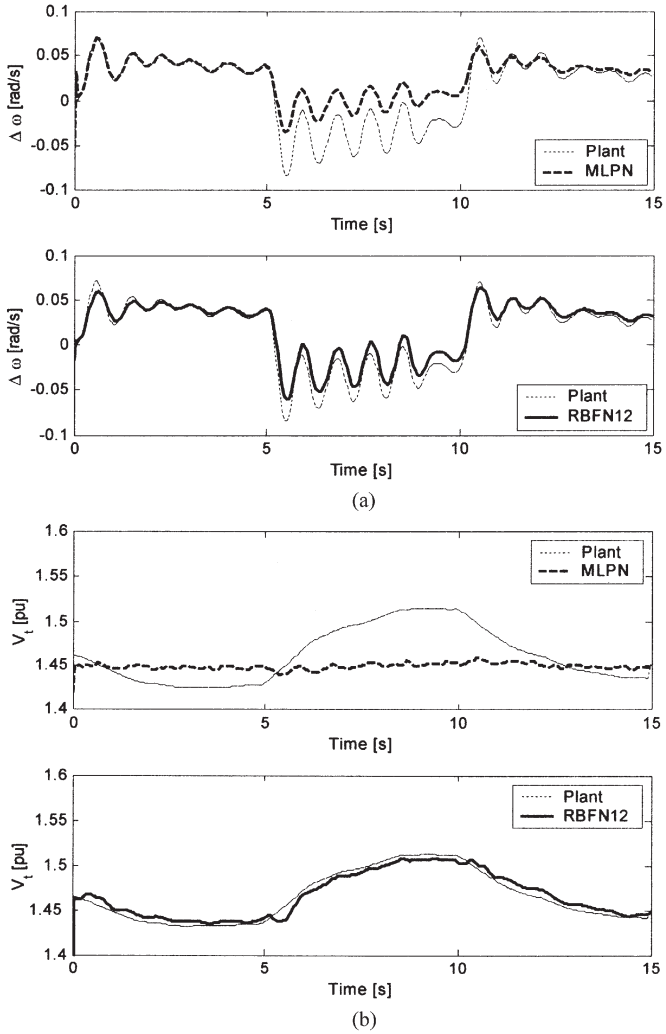

Fig. 9. Actual-signal-inputs-based online global convergence test. (a) Speed deviation $(\Delta\omega)$. (b) Terminal voltage (V_t).

the closest training condition H in Table III. The result in Fig. 8 shows that both identifiers trained by deviation signals give acceptable tracking results even during such large changes, and the MLPN identifier has a slightly better global convergence performance than the RBFN12 identifier.

This training procedure for the MLPN has been used as part of a neurocontroller successfully tested on practical generators in a multimachine power system [4], [5]. On a physical system, the training of the identifier could occur in either of the following two ways.

1) Data recorded from the physical system over a period of time could be used off-line for the training of identifier, which would start with random weights. This has actually been implemented on a MLPN identifier in [4] and [5].
2) The identifier could be trained online as the physical system moves from one operating condition to the next over periods of minutes or even hours. In both the off-line and online training cases, the identifier will not be allowed to interact with the neurocontroller until the identifier training has converged.

In the simulation results for this paper, it was convenient to change operating conditions at much shorter intervals of time, in order to speed up the simulation process. After all, the purpose of this investigation is to compare the performances of two different types of feedforward neural networks as identifiers only.

C. Global Model Identifiers Trained by Actual Signal Inputs

For the online identification of nonlinear dynamics of the synchronous generator using the MLPN and RBFN, the deviation signals are usually used, because not only do they provide better sensitivity for controller in practice, but also it is easy to amplify a small (deviation) signal than the full (actual) signal. However, the use of deviation signal inputs for the real-time identification might be difficult in a practical power system due to the problems of accurately measuring small values. Furthermore, the deviation signals in practice might be severely distorted with signal noises [12]. For these reasons, the actual signal inputs, which have much larger magnitudes, are here investigated as an alternative to the deviation signals. For the

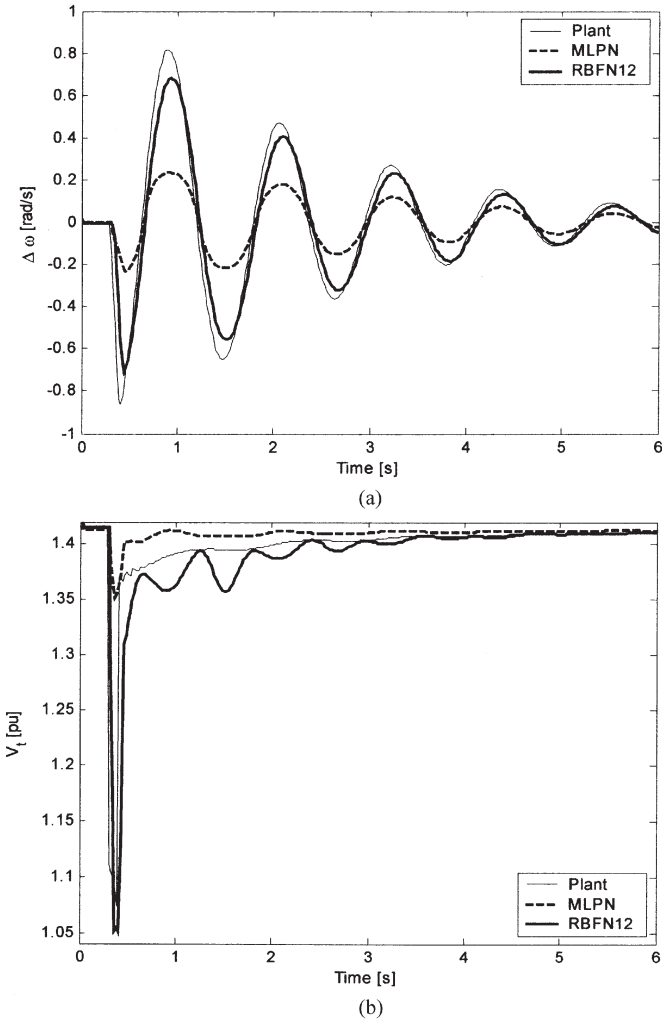

Fig. 10. Actual-signal-inputs-based online global convergence test by a 100-ms three-phase short circuit. (a) Speed deviation ($\Delta\omega$). (b) Terminal voltage (V_t).

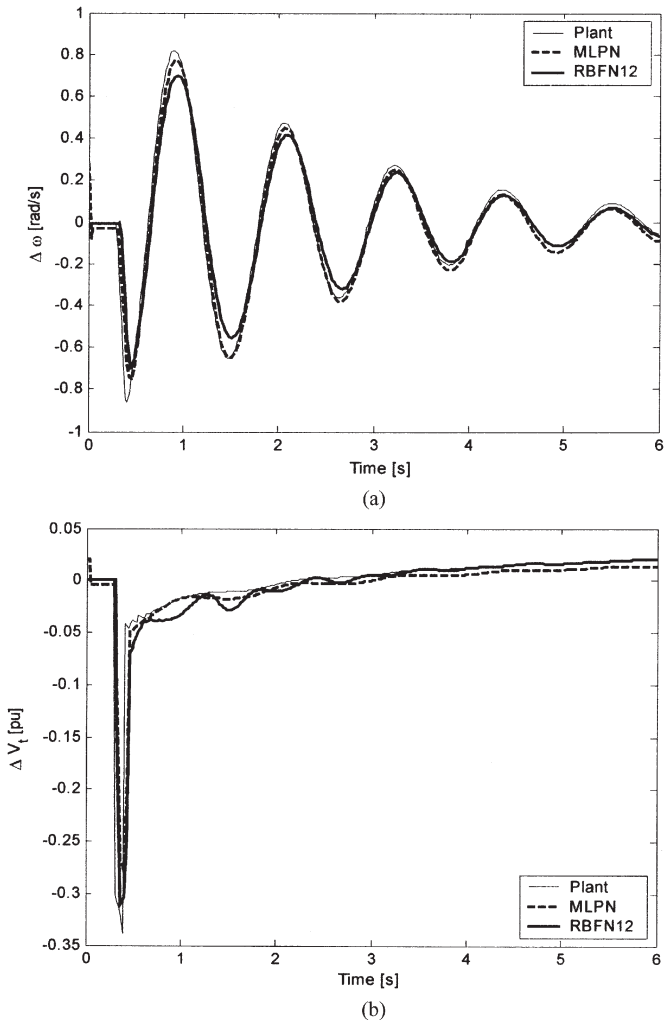

Fig. 11. Sensor failure test by a 100-ms three-phase short circuit. (a) Speed deviation ($\Delta\omega$). (b) Terminal voltage deviation (ΔV_t).

actual signal inputs based online identifiers, the inputs and outputs of identifiers are as follows:

1) input vector to the identifiers, $\boldsymbol{U}(k) = [P_{\rm in}(k) + \Delta P_{\rm in}(k), V_{\rm ref}(k) + \Delta V_{\rm ref}(k)]$;
2) output vector of the plant, $\boldsymbol{Y}(k) = [\Delta\omega(k), V_t(k)]$;
3) output vector of the identifiers, $\hat{\boldsymbol{Y}}(k) = [\Delta\hat{\omega}(k), \hat{V}_t(k)]$.

After online training for 3000 s for both identifiers (MLPN and RBFN12) with actual signal inputs, under the same randomly selected operating conditions as the previous test, the weights of the two identifiers are then frozen. Thereafter, the identifiers are tested at the operating condition E in Fig. 6 from $t = 0$ s to $t = 5$ s, and at the operating condition L from the $t = 5$ s to $t = 10$ s, and again at the operating condition E from $t = 10$ s to $t = 15$ s. The results in Fig. 9 show that the MLPN identifier trained by actual signals did not reach global convergence even though it was trained as a global model over multiple operating points. On the contrary, the RBFN12 identifier still shows to some degree acceptable identification capability. A similar behavior is also observed in the results of Fig. 10 for a large perturbation by applying a 100-ms three-phase short circuit to an infinite bus at the operating condition O (N) in Fig. 6 (see also the operating condition N in Table III).

The MLPN trained online with the actual signals did not retain past learned information in that the weights trained by actual signals, which are much less sensitive than deviation signals, did not learn the network's response globally. On the contrary, the RBFN has to some degree reached a global minimum without regard to the type of inputs, because the weights associated with an optimally selected neuron only affect a specified region of the multidimensional input space.

D. Robustness for Sensor Failure

Industrial processes to be controlled must be protected for the case of any sensor failures especially when the identifier provides the basis for a full-state observer, which tracks the corresponding states of the real system in conjunction with online measurement. When the ANN-based identifier is used as an observer in a practical power system, the fixed weights (which have been trained with the deviation signals as a global model) are not sensitive to a sensor failure of input signals,

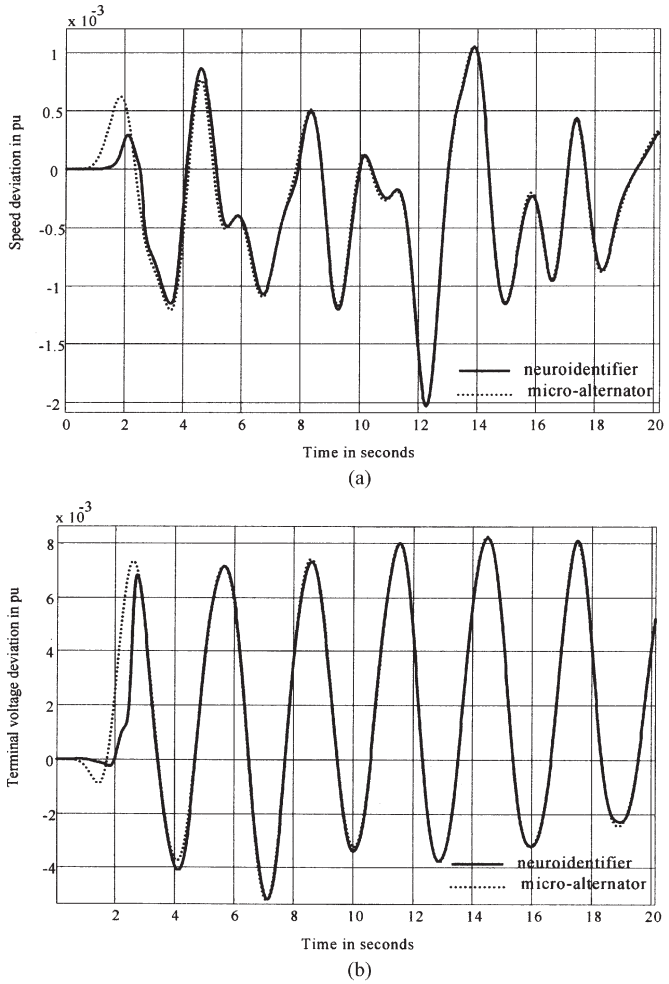

Fig. 12. Experimental results with forced training at $P_{\rm ref} = 0.1$ at unity pf. (a) Speed deviation $(\Delta\omega)$. (b) Terminal voltage deviation (ΔV_t).

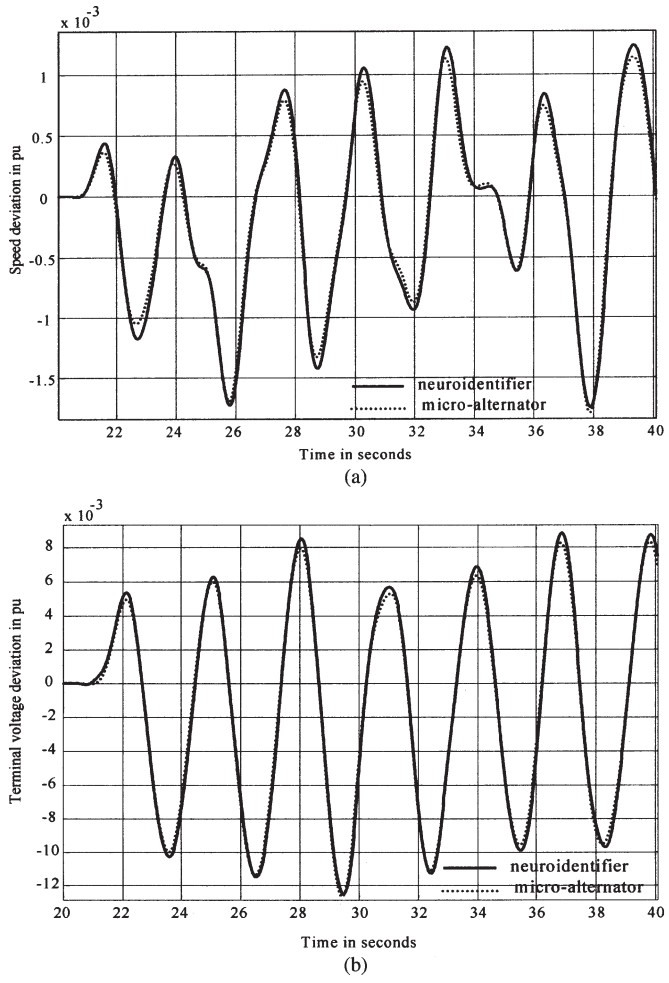

Fig. 13. Experimental results with PRBSs and training terminated at 20 s with $P_{\rm ref} = 0.2$ at unity pf. (a) Speed deviation $(\Delta\omega)$. (b) Terminal voltage deviation (ΔV_t).

which, therefore, make the predictive ability of the identifiers more robust.

To illustrate the robustness of the identifiers for a sensor failure, it is assumed that the sensor array for the input signals $[\Delta P_{\rm in}(k-1)\ \Delta P_{\rm in}(k-2)\ \Delta P_{\rm in}(k-3)]$ (for the GMIs trained by deviation signal inputs in Section III-B) has failed, and, therefore, these signals are not used. As in the previous section, the same 100-ms three-phase short circuit is now applied at the infinite bus at the operating condition N in Table III. The corresponding results in Fig. 11 clearly show that the two identifiers still track the system responses to some degree even without one of the measured input signals during a sensor failure situation.

IV. Global Model Identifiers: Experimental Results

From the simulation studies carried out in Section III, it can be concluded that the two identifiers as global models have each reached global convergence when they are trained with the deviations of measured signals as identifier inputs. Especially, if only deviation signals are used, the MLPN is strongly recommended in view of real-time hardware implementation issues, because the MLPN avoids the extensive computational efforts required in off-line training such as the determination of the RBF centers. Therefore, the usefulness in practice of the MLPN GMI trained with deviation signals is now verified by practical implementation. A constant exciter voltage $V_{\rm ref}$ and a turbine power signal $P_{\rm in}$ are applied to the microalternator [5] for a particular operating condition. The training of the MLPN identifier is carried out by injecting PRBSs $\Delta V_{\rm ref}$ and $\Delta P_{\rm in}$ into the exciter and the turbine, respectively. Fig. 12 shows the terminal voltage deviation and speed deviation, respectively, as a result of training with PRBSs at an operating point: $P_{\rm ref} = 0.1$ per unit (p.u.) at unity pf. These results show that the MLPN identifier takes about 3 s to track the plant dynamics.

Fig. 13 shows the speed deviation and terminal voltage deviation with the PRBSs applied to the exciter and turbine, at a new operating point: $P_{\rm ref} = 0.2$ p.u. at unity pf. In these figures, the training is terminated at $t = 20$ s in order to test the generalization capability of the MLPN identifier at this operating point. It is obvious from the results that the MLPN is able to track the speed and terminal voltage deviations with sufficient accuracy even when online training is stopped. This proves that the MLPN identifier has achieved an adaptive global minimum after 20 s. Similar results were obtained at other operating points and conditions.

V. Conclusion

This paper compared the performance of a multilayer perceptron neural network (MLPN) and a radial basis function neural network (RBFN), using the actual signals as well as the deviation signals in a continually online training mode, to identify the nonlinear dynamics of a synchronous generator connected to an infinite bus. The online identifiers were each trained at several different stable-operating conditions as global models. In each case, after a period of training, the weights were frozen, and the identifiers were then tested at a different operating condition, either included in the training set or outside the training set. The weights were frozen in order to simulate the condition when the training of the identifiers would be automatically stopped, and their weights frozen, in case a measurement from a sensor was lost, but the control action that uses the identifier would continue until remedial action could be taken. These tests are important to evaluate the confidence levels in the neural network identifiers/models [for example, the model network used in the adaptive-critic-design (ACD)-based optimal neurocontrollers [3]–[5]] developed. From the case studies, the following conclusions can be drawn:

1) The MLPN identifier as a global model successfully converges to a global minimum if it is trained using deviation signals. In this case, the MLPN could have the better local convergence and online global convergence properties to identity a synchronous generator over the RBFN.
2) The MLPN identifier trained by the actual signal fails to identify a synchronous generator even with a global model. In contrast, the RBFN identifier as a global model gives reasonably good results irrespective of whether it was trained using deviation or actual signals.

Generally, the RBFN requires less training time to converge and fewer computational complexities to train the identifiers. However, when the MLPN is trained by using the global model with deviation signal inputs, it has the faster convergence speed, strong local convergence, and online global convergence properties to identity the synchronous generator.

The final recommendation is that the RBFN should be used, for either deviation or actual signal inputs to an identifier, taking into account the importance of having a global model with only online training. If only deviation signals are used, the MLPN is strongly recommended. The use of MLPN is also preferred for the real-time hardware implementation, because the efforts to determine the RBF centers can be avoided.

References

[1] B. Adkins and R. G. Harley, *The General Theory of Alternating Current Machines*. London, U.K.: Chapman & Hall, 1975.
[2] P. M. Anderson and A. A. Fouad, *Power System Control and Stability*. New York: IEEE Press, 1994.
[3] J. W. Park, R. G. Harley, and G. K. Venayagamoorthy, "Adaptive-critic-based optimal neurocontrol for synchronous generators in a power system using MLP/RBF neural networks," *IEEE Trans. Ind. Appl.*, vol. 39, no. 5, pp. 1529–1540, Sep./Oct. 2003.
[4] G. K. Venayagamoorthy, R. G. Harley, and D. C. Wunsch, "Dual heuristic programming excitation neurocontrol for generators in a multimachine power system," *IEEE Trans. Ind. Appl.*, vol. 39, no. 2, pp. 382–394, Mar./Apr. 2003.
[5] ——, "Implementation of adaptive critic based neurocontrollers for turbo generators in a multimachine power system," *IEEE Trans. Neural Netw.*, vol. 14, no. 5, pp. 1047–1064, Sep. 2003.
[6] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ: Prentice-Hall, 1999.
[7] M. A. Abido and Y. L. Abdel-Magid, "On-line identification of synchronous machines using radial basis function neural networks," *IEEE Trans. Power Syst.*, vol. 12, no. 4, pp. 1500–1506, Nov. 1997.
[8] Z. Uykan, C. Guzelis, M. E. Celebi, and H. N. Koivo, "Analysis of input–output clustering for determining centers of RBFN," *IEEE Trans. Neural Netw.*, vol. 11, no. 4, pp. 851–858, Jul. 2000.
[9] S. Chen, S. A. Billings, and P. M. Grant, "Recursive hybrid algorithm for non-linear system identification using radial basis function neural networks," *Int. J. Control*, vol. 55, no. 5, pp. 1051–1070, Mar. 1992.
[10] M. Nørgaard, O. Ravn, N. K. Poulsen, and L. K. Hansen, *Neural Networks for Modelling and Control of Dynamic Systems*. London, U.K.: Springer-Verlag, 2000.
[11] P. Kundur, *Power System Stability and Control*, EPRI Editors. New York: McGraw-Hill, 1993.
[12] B. S. Rigby, N. S. Chonco, and R. G. Harley, "Analysis of a power oscillation damping scheme using a voltage-source inverter," *IEEE Trans. Ind. Appl.*, vol. 38, no. 4, pp. 1105–1113, Jul./Aug. 2002.

Jung-Wook Park (S'00–M'03) was born in Seoul, Korea. He received the B.S. degree (*summa cum laude*) from the Department of Electrical Engineering, Yonsei University, Seoul, Korea, in 1999, and the M.S.E.C.E. and Ph.D. degrees from the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, in 2000 and 2003, respectively.

He is currently an Assistant Professor in the School of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea. He was a Postdoctoral Research Associate in the Department of Electrical and Computer Engineering, University of Wisconsin, Madison, during 2003–2004, and a Senior Research Engineer with LG Electronics Inc., Korea, during 2004–2005. His current research interests are in power system dynamics, flexible ac transmission system (FACTS) devices, inverter systems, hybrid systems, optimization control algorithms, and application of artificial neural networks.

Dr. Park was the recipient of the IEEE Industry Applications Society (IAS) Second Prize Paper Award in 2003. He was also a Technical Co-Chair of the Power Systems and Power Electronic Circuits Committee, IEEE International Symposium on Circuits and Systems (ISCAS), held in Kobe, Japan, in May 2005.

Ganesh Kumar Venayagamoorthy (S'98–M'99–SM'02) received the B.Eng. (Honors) degree with first class honors in electrical and electronics engineering from the Abubakar Tafawa Balewa University, Bauchi, Nigeria, and the M.Sc.Eng. and Ph.D. degrees in electrical engineering from the University of Natal, Durban, South Africa, in 1994, 1999, and 2002, respectively.

He was a Senior Lecturer at the Durban Institute of Technology, KwaZulu-Natal, South Africa, prior to joining the University of Missouri, Rolla, as an Assistant Professor in the Department of Electrical and Computer Engineering in May 2002. His research interests are in computational intelligence, power systems, evolving hardware, and signal processing.

Dr. Venayagamoorthy is a Senior Member of the South African Institute of Electrical Engineers and a Member of the INNS and the American Society of Engineering Education. He is a 2004 NSF CAREER Award recipient, the 2004 IEEE St. Louis Section Outstanding Young Engineer, the 2003 International Neural Network Society (INNS) Young Investigator Award recipient, a 2001 recipient of the IEEE Computational Intelligence Society (CIS) summer research scholarship, and the recipient of five prize papers from the IEEE Industry Applications Society (IAS) and IEEE Computational Intelligence Society. He is an Associate Editor of the IEEE Transactions on Neural Networks. He is currently the IEEE St. Louis CIS and IAS Chapter Chair, the Chair of the Task Force on Intelligent Control Systems and the Secretary of the Intelligent Systems Subcommittee of the IEEE Power Engineering Society.

Ronald G. Harley (M'77–SM'86–F'92) received the M.Sc.Eng. degree (*cum laude*) in electrical engineering from the University of Pretoria, Pretoria, South Africa, in 1965, and the Ph.D. degree from London University, London, U.K. in 1969.

In 1971, he was appointed to the Chair of Electrical Machines and Power Systems at the University of Natal, Durban, South Africa, where he was a Professor of Electrical Engineering for many years, and where he also became the Department Head and Dean of Engineering. He is currently the Duke Power Company Distinguished Professor at the Georgia Institute of Technology, Atlanta. His research interests include dynamic behavior and condition monitoring of electric machines, motor drives, power systems and their components, and controlling them by the use of power electronics and intelligent control algorithms.

Dr. Harley is a Fellow of the Institution of Electrical Engineers, U.K., and the South African Institute of Electrical Engineers. He is also a Fellow of the Royal Society in South Africa and a Founder Member of the Academy of Science in South Africa formed in 1994. During 2000 and 2001, he was one of the IEEE Industry Applications Society's six Distinguished Lecturers. He was the Vice-President of Operations of the IEEE Power Electronics Society for 2003–2004. Currently, he is the Chair of the Distinguished Lecturers and Regional Speakers Program of the IEEE Industry Applications Society and Chair of the Atlanta Chapter of the IEEE Power Engineering Society.