

Comparison of MLP and RBF Neural Networks Using Deviation Signals for Indirect Adaptive Control of a Synchronous Generator

Jung-Wook Park, *Student Member, IEEE*, and Ronald G. Harley, *Fellow, IEEE*

School of Electrical and Computer Engineering
Georgia Institute of Technology
GA 30332-0250, U.S.A.

(e-mail: jwpark@ece.gatech.edu and ron.harley@ece.gatech.edu)

Ganesh. K. Venayagamoorthy, *Member, IEEE*

Department of Electrical and Computer Engineering
University of Missouri-Rolla
MO 65409-0249, U.S.A.

(e-mail: ganeshv@ece.UMR.EDU)

Abstract – This paper compares the performances of a multilayer perceptron neurocontroller (MLPNC) and a radial basis function neurocontroller (RBFNC) for backpropagation through time based indirect adaptive control of the synchronous generator. Also, the neurocontrollers are compared with the conventional controller for small as well as large disturbances to the power system.

I. INTRODUCTION

A synchronous generator in a power system is a nonlinear, fast acting, multiple-input multiple-output (MIMO) device [1]-[2]. Conventional linear controllers (CONVC) for the synchronous generator consist of the automatic voltage regulator (AVR) and speed governor, and they are designed to control, in some optimal fashion, the generator around one particular operating point; and at any other point the generator's performance is degraded. Adaptive controllers for synchronous generator are usually designed using linear models and traditional techniques of identification, analysis, and synthesis to achieve the desired performance. However, due to a synchronous generator's wide operating range, its complex dynamics [3]-[5], its transient performance, its nonlinearities, and a changing system configuration, it cannot be accurately modelled as a linear device.

Artificial neural networks (NNs) offer an alternative as nonlinear adaptive controllers. For indirect control of the nonlinear dynamic plant based on the model reference adaptive system (MRAS), the NNs are able to adaptively model or identify a non-stationary, nonlinear, and MIMO process/plant on-line while the process is changing, and thereby yield information that can be used by another neural network to control the process [6]-[7].

Researchers have until now used two different types of neural networks, namely, a multilayer perceptron network (MLPN) [3]-[5], [8]-[11] or a radial basis function network (RBFN) [12]-[16], both in single and multi-machine power system studies. Proponents of each type of neural network (NN) have claimed advantages for their choice of NN, without comparing the performance of the other type for the same study. The applications of NNs in the power industry

will expand, and at this stage there is no authoritative fair comparison between the MLPN and the RBFN.

This paper makes a new contribution by directly comparing the MLPN and RBFN for use in the indirect adaptive control of a synchronous generator plant. Factors considered in this paper include the performance of system damping and transient stability for different kinds of faults applied to the plant with continually on-line training. The computational complexity and time, which are important issues in real-time operation, are also compared.

Only *deviations* of signals from their set points are used (as proposed by [3]-[5], [8]-[11], [13], [16]) as the inputs of the NNs and not their *actual* values.

The process/plant modeling and the MLPN/RBFN are described in Section II. The *backpropagation through time* (BPTT) [17] with the single truncation depth based indirect adaptive control scheme is presented in Section III. Several simulation results are presented with case studies in Section IV. Finally, the conclusions are given in Section V.

II. DESCRIPTION OF PLANT AND MLPN/RBFN

A. Plant modeling

The synchronous generator, turbine, exciter and transmission system connected to an infinite bus in Fig. 1 form the plant (dotted block in Fig. 1) that has to be controlled. Nonlinear equations are used to describe the dynamics of the plant in order to generate the data for the NN controllers and identifiers. On a physical plant, this data would be measured. The generator (**G**) with its damper windings is described by the seventh order d-q axis set of equations with the generator current, speed, and rotor angle as the state variables. In the plant, P_t and Q_t are the real and reactive power at the generator terminal, respectively, Z_e is the transmission line impedance, P_m is the mechanical input power to the generator, V_{fd} is the exciter field voltage, V_b is the infinite bus voltage, $\Delta\omega$ is the speed deviation, ΔV_t is the terminal voltage deviation, V_t is the terminal voltage, ΔV_{ref} is the reference voltage deviation, V_{ref} is the reference voltage,

$\hat{f}: \mathbf{X} \in \mathbf{R}^n \rightarrow \mathbf{Y} \in \mathbf{R}^m$ is as follows.

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

where \mathbf{X} is the input vector, \mathbf{C}_j is the j^{th} center of RBF unit in the hidden layer, h is the number of RBF units, b_i and v_{ji} are the bias term and the weight between the hidden and output layers, respectively, and y_i is the i^{th} output in the m -dimensional space. Once the centers of RBF units are established, the width of the i^{th} center in the hidden layer is calculated by (3).

$$\beta_i = \left[\frac{1}{h} \sum_{j=1}^h \sum_{k=1}^n (\|c_{ki} - c_{kj}\|) \right]^{1/2} \quad (3)$$

where c_{ki} and c_{kj} are the k^{th} value of the center of i^{th} and j^{th} RBF units. In (2) and (3), $\|\cdot\|$ represents the euclidean norm.

There are four different ways for input-output mapping using the RBFN, depending on how the input data is fed to the network [19].

- 1) Batch mode clustering of centers and batch mode gradient decent for linear weights.
- 2) Batch mode clustering of centers and pattern mode gradient decent for linear weights.
- 3) Pattern mode clustering of centers and pattern mode gradient decent for linear weights.
- 4) Pattern mode clustering of centers and batch mode gradient decent for linear weights.

Considering highly extensive computational complexity during on-line training, the batch mode k -means clustering algorithm for centers is firstly calculated off-line for the centers of the RBF unit. The pattern mode least-mean-square (LMS) algorithm is then calculated on-line to update the neural network's weights. By trial and error, twelve and six neurons in the hidden layer for the RBFNI and RBFNC respectively, are optimally chosen for this study.

III. A TRUNCATED BPTT BASED INDIRECT ADAPTIVE CONTROL

The structure of the overall system for the indirect adaptive control of the plant using the neurocontroller (RBFNC/MLPNC) and neuroidentifier (RBFNI/MLPNI) is shown in Fig. 4. By using the output of an optimal predictor as the model reference rather than zero value for the desired deviation outputs, the problem of the neurocontroller possibly generating excessive control signals, can be overcome. The design and description of the optimal predictor are explained

in [8]. Also, the scaling factor vector \mathbf{K} as proposed by [3], [10], [14], [16] is used for the control signal to prevent the non-minimum phase system from becoming unstable.

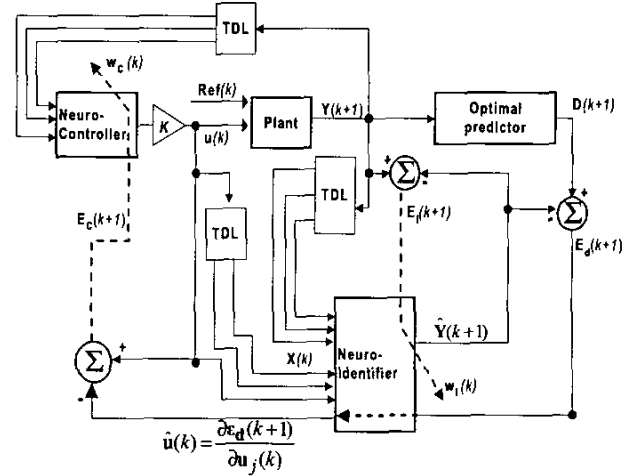


Fig. 4. The structure of the overall system for the indirect adaptive control using a neuroidentifier and neurocontroller

A. Control strategy

The control scheme in Fig. 4 is based on the backpropagation through time (BPTT) with the single truncation depth. BPTT accounts for the link between *present* actions and *future* consequences.

BPTT is computed through the neuroidentifier instead of the plant to derive estimates of the dynamic derivatives of the instantaneous total least square error energy $\epsilon_d(k+1)$ between the identifier's output and optimal predictor's output at time $k+1$, with respect to the input vector $\mathbf{u}(k)$ at time k . BPTT thus gives a direct method to compare the performances of the RBFN and MLPN because the one-step behind estimated control signal $\hat{\mathbf{u}}(k) = \partial \epsilon_d(k+1) / \partial \mathbf{u}_j(k)$ requires calculating the different network Jacobians (Eqns. (8) and (9)) for the two NNs during the real-time control of plant.

The error vector $\mathbf{E}_d(k+1)$ between $\hat{\mathbf{u}}(k)$ and $\mathbf{u}(k)$ is used to adapt the weights in the neurocontroller, thus driving the plant towards the optimal predictor output.

The mathematical analysis to calculate the $\hat{\mathbf{u}}(k)$ by BPTT with the single truncation depth, including the Jacobians for the two NN's, is described in detail below.

$$\epsilon_d(k+1) = \frac{1}{2} \sum_j [\mathbf{E}_{d,j}(k+1)]^2 = \frac{1}{2} \sum_j [\mathbf{D}_j(k+1) - \hat{\mathbf{Y}}_j(k+1)]^2 \quad (4)$$

$$\hat{\mathbf{u}}(k) = \frac{\partial \epsilon_d(k+1)}{\partial \mathbf{u}_j(k)} = \frac{\partial \epsilon_d(k+1)}{\partial \mathbf{g}_{l,j}(k)} \frac{\partial \mathbf{g}_{l,j}(k)}{\partial \mathbf{u}_j(k)} \quad (5)$$

where

$$\begin{aligned}
\frac{\partial \varepsilon_{\mathbf{d}}(k+1)}{\partial \mathbf{g}_{l,j}(k)} &= \frac{\partial \varepsilon_{\mathbf{d}}(k+1)}{\partial \mathbf{Y}_{l,j}(k)} \frac{\partial \mathbf{Y}_{l,j}(k)}{\partial \mathbf{g}_{l,j}(k)} = \frac{\partial \varepsilon_{\mathbf{d}}(k+1)}{\partial \mathbf{Y}_{l,j}(k)} \Phi'_{l,j}(\mathbf{g}_{l,j}(k)) \\
&= \Phi'_{l,j}(\mathbf{g}_{l,j}(k)) \frac{\partial \varepsilon_{\mathbf{d}}(k+1)}{\partial \mathbf{g}_{L,j}(k)} \frac{\partial \mathbf{g}_{L,j}(k+1)}{\partial \mathbf{Y}_{l,j}(k)} \\
&= \Phi'_{l,j}(\mathbf{g}_{l,j}(k)) \sum_{j=1}^{m_l} \delta_{L,j}(k) \cdot w_{L,j}(k)
\end{aligned} \quad (6)$$

where L and l denote the output layer and hidden layer, respectively, j is the index neuron number of the layer, m_l is the number of neurons in the hidden layer, $\mathbf{g}(k)$ is the regression vector as the activity of the neuron, w is weight, and $\delta(k)$ is defined as the local gradient function. The term $\delta_{L,j}(k)$ in (6) is calculated as the follows.

$$\begin{aligned}
\delta_{L,j}(k) &= -\frac{\partial \varepsilon_{\mathbf{d}}(k+1)}{\partial \mathbf{g}_{L,j}(k)} \\
&= -\frac{\partial \varepsilon_{\mathbf{d}}(k+1)}{\partial \mathbf{E}_{\mathbf{d},j}(k+1)} \frac{\partial \mathbf{E}_{\mathbf{d},j}(k+1)}{\partial \hat{\mathbf{Y}}_{L,j}(k+1)} \frac{\partial \hat{\mathbf{Y}}_{L,j}(k+1)}{\partial \mathbf{g}_{L,j}(k)} \\
&= \mathbf{E}_{\mathbf{d},L,j}(k+1) \cdot \frac{\partial \hat{\mathbf{Y}}_{L,j}(k+1)}{\partial \mathbf{g}_{L,j}(k)} \\
&= \mathbf{E}_{\mathbf{d},L,j}(k+1) \cdot \Phi'_{L,j}(\mathbf{g}_{L,j}(k))
\end{aligned} \quad (7)$$

For the last term in (5), $\partial \mathbf{g}_{l,j}(k) / \partial \mathbf{u}_j(k)$ is $w_{l,j}(k)$ for the MLPN and 1 for the RBFN, respectively, because the regression vector in the hidden layer is formed by the inner products between input vector and weights for the MLPN, and for the RBFN, the input vector is passed directly to the hidden layer without weights.

Also, $\Phi'_{L,j}(\mathbf{g}_{L,j}(k))$ in (7) is 1 for both NNs because the linear function is used as the activation function in the output layer.

The network Jacobians in the hidden layer, $\Phi'_{l,j}(\mathbf{g}_{l,j}(k))$, in (6) for the MLPN and RBFN are given in (8) and (9), respectively.

$$\Phi'(\mathbf{g}_j) = \frac{\partial}{\partial \mathbf{g}_j} \left(\frac{1}{1 + \exp(\mathbf{g}_j)} \right) / \partial \mathbf{g}_j = \Phi(\mathbf{g}_j)(1 - \Phi(\mathbf{g}_j)) \quad (8)$$

$$\Phi'(\mathbf{g}_j) = \frac{\partial}{\partial \mathbf{g}_j} \left(-\exp \left\{ -\frac{\|\mathbf{g}_j - \mathbf{C}\|^2}{\beta^2} \right\} \right) / \partial \mathbf{g}_j = \left(2 \sum_{p=1}^{m_l} \frac{C_p - \mathbf{g}_j}{\beta_p^2} \right) \Phi(\mathbf{g}_j) \quad (9)$$

B. On-line training process

The input vector, $\mathbf{u}(k)$ and the output vector, $\mathbf{Y}(k+1)$ in Fig. 4 are $\mathbf{u}(k) = [\Delta P_{in}(k), \Delta V_{ref}(k)]$ and $\mathbf{Y}(k) = [\Delta \omega(k), \Delta V(k)]$ for on-line training with deviation signals. The sampling frequency to generate $\mathbf{u}(k)$ is 25 Hz, while the main supply frequency for the plant is 50 Hz. The neuroidentifier's output, $\hat{\mathbf{Y}}(k+1) = \hat{\mathbf{f}}(\mathbf{x}(k))$, where $\mathbf{x}(k) = [\mathbf{Y}(k) \ \mathbf{u}(k) \ \mathbf{Y}(k-1) \ \mathbf{u}(k-1) \ \mathbf{Y}(k-2) \ \mathbf{u}(k-2)]^T$. The error vectors, $\mathbf{E}_I(k+1)$ and $\mathbf{E}_C(k+1)$ are used to update the neuroidentifier and neurocontroller's weights during on-line training, respectively.

The neuroidentifier and neurocontroller are trained on-line in two phases; the first phase is called *pre-control phase* in which their weights are previously adjusted by injecting small pseudo-random binary signals (PRBSs) to the turbine and exciter, and control signals from the neurocontrollers are not applied to the plant. The second phase is called the *post-control phase* in which the neuroidentifier and neurocontroller are trained on-line, and the synchronous generator is being controlled by the output of neurocontroller.

IV. CASE STUDIES BY TIME-DOMAIN SIMULATION

This section makes two comparisons. Firstly, it compares the computational complexity and time required by the neuroidentifiers to process the data set in real-time. Secondly, the performances of the neurocontrollers (MLPNC/RBFNC) trained with deviation signals are compared with CONVC for the improvement of system damping and transient stability. Two different types of disturbances, namely a three phase short circuit at the infinite bus and a $\pm 5\%$ step change in the reference voltage of exciter are considered. The CONVC has been tuned by the method explained in [8].

A. The computational complexity and elapsed time

During 0.2 s of training time, the computational complexity required for the neuroidentifier (MLPNI/RBFNI) to process the data set is measured by calculating the number of floating-point operations (FLOPS). For the RBFN, the computational complexity depends on the number of RBF unit centers, which contain information about the entire operating range. In Table I, the RBFNs with 10, 12, 15 and 21 centers are denoted RBFN10, RBFN12, RBFN15, and RBFN21, respectively, for convenience of presentation. Simulation results show that the larger numbers of RBF unit centers do not necessarily yield better performance. Table I clearly shows that the RBFNs requires less FLOPS and less time than the MLPN.

The absolute value of time used in a practical implementation by dedicated microprocessor hardware will be much less than the values shown in Table I.

TABLE I
THE NUMBER OF FLOATING POINT OPERATIONS AND ELAPSED TIME
REQUIRED DURING 0.2 s

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

B. Three phase short circuit test to represent a large impulse type disturbance

First, the plant is operating in the steady state condition shown in Table II. At $t=0.3$ s, a temporary three phase short circuit is applied to an infinite bus for 100 ms from $t=0.3$ s to 0.4 s.

TABLE II
THE STEADY STATE OPERATING CONDITION

P_r (pu)	1	V_{fd} (pu)	0.0020
Q_r (pu)	0.2343	V_b (pu)	1
P_{in} (pu)	1.0020	V_i (pu)	1.034
δ ($^\circ$)	74.8799	Z_c (pu)	$0.02+j0.4$

The results of three phase short circuit test, comparing the performance of the RBFNC, MLPNC, and CONVC, are shown in Figs. 5, 6, and 7.

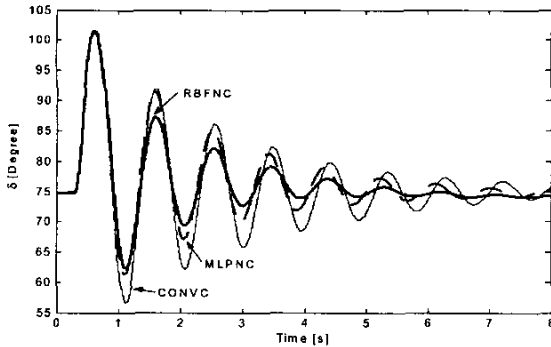


Fig. 5. Three phase short circuit test: Rotor angle (δ)

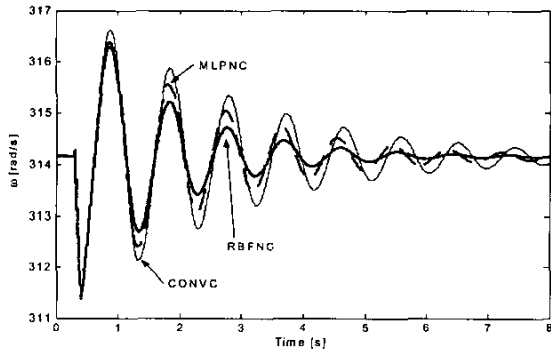


Fig. 6. Three phase short circuit test: Speed (ω)

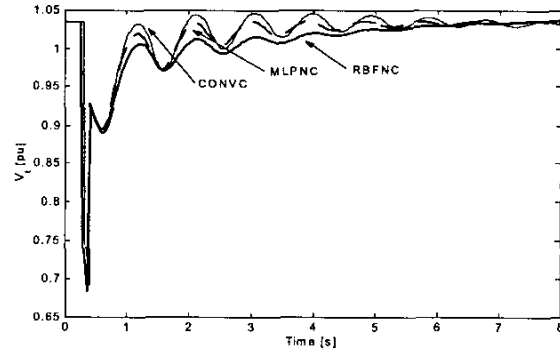


Fig. 7. Three phase short circuit test: Terminal voltage (V_i)

These results show that not only do the neurocontrollers damp out the oscillations for the rotor angle (δ), the speed (ω) and terminal voltage (V_i) more effectively than the CONVC, but also that the RBFNC provides a better performance than the MLPNC.

C. $\pm 5\%$ Step changes in the reference voltage of exciter

Like the three phase short circuit test, the plant is operating in the steady state condition shown in Table II. At $t=1$ s, a step increase in the reference voltage of the exciter is applied, resulting in a 5% step increase from the nominal value of the terminal voltage.

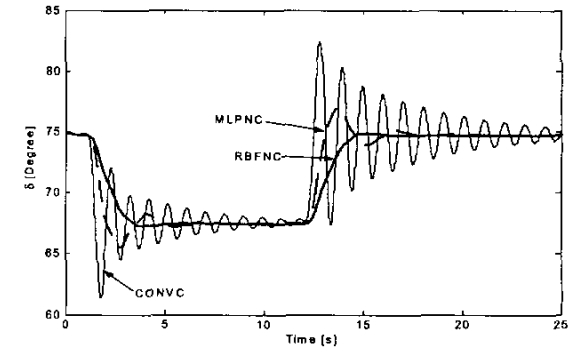


Fig. 8. Step changes in reference voltage of exciter: Rotor angle (δ)

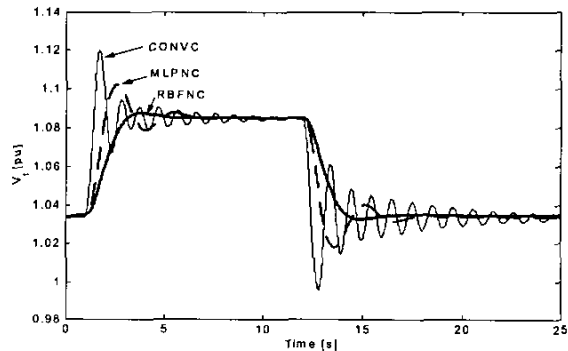


Fig. 9. Step changes in reference voltage of exciter: Terminal voltage (V_i)

At $t=12$ s, the change in reference voltage is removed, and the system returns to the initial steady state condition.

The results in Figs. 8 and 9 show that the neurocontrollers improve the transient system damping compared to the CONVC, but that the RBFNC outperforms the MLPNC, i.e. the overshoot is less, and the desired point is reached slightly faster.

V. CONCLUSIONS

This paper compared the performance of a multilayer perceptron neurocontroller (MLPNC) and a radial basis function neurocontroller (RBFNC) along with the conventional controller (CONVC) to control a synchronous generator connected to a power system. The neurocontrollers use the *deviation* of signals in a continually on-line training mode obtained by the *backpropagation through time* (BPTT) based indirect adaptive control. The results show that the RBFNC improves the system damping and dynamic transient stability more effectively than the MLPNC or CONVC. Also, the RBFNC requires fewer computational complexities and elapsed time to train the network on-line, than the MLPNC.

In general, the BPTT based indirect adaptive control method can be effective when the neurocontrollers are trained on-line with deviation signals, and the RBFNC should be preferred to the MLPNC.

ACKNOWLEDGMENT

Financial support by the National Science Foundation (NSF), USA, is gratefully acknowledged.

VI. REFERENCES

- [1] P.M. Anderson and A.A. Fouad, "Power system control and stability," IEEE Press, New York, 1994, ISBN 0-7803-1029-2.
- [2] B. Adkins and R.G. Harley, "The general theory of alternating current machines," Chapman and Hall, London, 1975, ISBN 0-412-15560-5.
- [3] Payman Shamsollahi and O.P. Malik, "Direct Neural Adaptive Control Applied to Synchronous Generator," *IEEE Trans. on Energy Conversion*, Vol.14, No.4, pp. 1341-1346, December 1999.
- [4] Young-Moon Park, Myeon-Song Choi, and Kwang Y. Lee, "A Neural Network-Based Power System Stabilizer using Power Flow Characteristics," *IEEE Trans. on Energy Conversion*, Vol.11, No.2, pp. 435-441, June 1996.
- [5] T. Kobayashi and A. Yokoyama, "An Adaptive Neuro-Control System of Synchronous Generator for Power System Stabilization," *IEEE Trans. on Energy Conversion*, Vol.11, No.3, pp. 621-630, September 1996.
- [6] S. Haykin, *Neural Networks: A comprehensive Foundation*, Prentice-Hall International, Inc. 1999, ISBN 0-13-908385-5.
- [7] K.S. Narendra, and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Trans. on Neural Networks*, Vol.1, No.1, pp. 4-27, March 1990.
- [8] G.K. Venayagamoorthy and R.G. Harley, "A Continually Online Trained Neurocontroller for Excitation and Turbine Control of a Turbogenerator," *IEEE Trans. on Energy Conversion*, Vol.16, No.3, pp. 261-269, September 2001.
- [9] J. He and O.P. Malik, "An Adaptive Power System Stabilizer Based on Recurrent Neural Networks," *IEEE Trans. on Energy Conversion*, Vol.12, No.4, pp. 413-418, December 1997.

- [10] Young-Moon Park, Seung-Ho Hyun, and Jin-Ho. Lee, "A Synchronous Generator Stabilizer Design Using Neuro Inverse Controller and Error Reduction Network," *IEEE Trans. on Power Systems*, Vol.11, No.4, pp. 1969-1975, November 1996.
- [11] Q.H. Wu, G.W. Irwin, and B.W. Hogg, "A Neural Network Regulator for Turbogenerators," *IEEE Trans. on Neural Networks*, Vol.3, No.1, pp. 95-100, Jan 1992.
- [12] R. Segal, M.L. Kothari, and S. Madhani, "Radial basis function (RBF) network adaptive power system stabilizer," *IEEE Trans. on Power Systems*, Vol.15, No.2, pp. 722-727, May 2000.
- [13] P. K. Dash, S. Mishra, and G. Panda, "A Radial basis function neural network controller for UPFC," *IEEE Trans. on Power Systems*, Vol.15, No.4, pp. 1293-1299, November 2000.
- [14] E. Swidenbank, S. McLoone, D. Flynn, G.W. Irwin, M.D. Brown, and B.W. Hogg, "Neural Network Based Control for Synchronous Generators," *IEEE Trans. on Energy Conversion*, Vol.14, No.4, pp. 1673-1678, December 1999.
- [15] M.A. Abido and Y.L. Abdel-Magid, "On-line identification of synchronous machines using radial basis function neural networks," *IEEE Trans. on Power Systems*, Vol.12, No.4, pp. 1500-1506, November 1997.
- [16] D. Flynn, S. McLoone, G.W. Irwin, M.D. Brown, E. Swidenbank, and B.W. Hogg, "Neural control of turbogenerator systems," *Automatica*, Vol.33, No.11, pp. 1961-1973, November 1997.
- [17] P. J. Werbos, "Backpropagation Through Time: What It Does and How to Do It," *Proc. IEEE*, Vol.78, No.10, pp. 1550-1560, October 1990.
- [18] 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, 1992.
- [19] Z. Uykan, C. Guzelis, M.E. Celebi, and H.N. Koivo, "Analysis of input-output clustering for determining centers of RBFNC," *IEEE Trans. on Neural Networks*, Vol.11, No.4, pp. 851-858, July 2000.

VII. BIOGRAPHIES

Jung-Wook Park (S'00) was born in Seoul, South Korea, on July 18, 1973. He received the B.S. degree (Summa cum laude) from Department of Electrical Engineering, the Yonsei University, Seoul, South Korea, in 1999, and the M.S. degree in Electrical and Computer Engineering from the Georgia Institute of Technology, Atlanta, USA, in 2000, where he is currently pursuing the Ph.D. degree. His current research interests are in power system dynamics, FACTS (Flexible AC Transmission System), electric machines, power electronics, and application of artificial neural networks.

Ronald G. Harley (M'77-SM'86-F'92) was born in South Africa. He obtained a BScEng degree (cum laude) from the University of Pretoria in 1960, and a MScEng degree (cum laude) from the same University in 1965, and PhD from London University in 1969. In 1970 he was appointed to the Chair of Electrical Machines and Power Systems at the University of Natal in Durban, South Africa. He is currently at the Georgia Institute of Technology, Atlanta, USA. He has co-authored some 220 papers in refereed journals and international conferences. Ron is a Fellow of the SAIEE, a Fellow of the IEE, and a Fellow of the IEEE. He is also a Fellow of the Royal Society in South Africa, a Fellow of the University of Natal, and a Founder Member of the Academy of Science in South Africa formed in 1994. He has been elected as a Distinguished Lecturer by the IEEE Industry Applications Society for the years 2000 and 2001. His research interests are in the dynamic and transient behavior of electric machines and power systems, and controlling them by the use of power electronics and intelligent control algorithms.

Ganesh K Venayagamoorthy (S'93-M'98) was born in Jaffna, Sri Lanka. He received a BEng (Honors) degree in Electrical and Electronics Engineering from the Abubakar Tafawa Balewa University, Nigeria, in 1994 and a MScEng degree in Electrical Engineering from the University of Natal, South Africa, in April 1999. He is completing his PhD degree in Electrical Engineering at the University of Natal, South Africa. He was a Research Associate at the Texas Tech University, USA in 1999 and currently at the University of Missouri-Rolla, USA.