

REAL-TIME IMPLEMENTATION OF A NEURO-AVR FOR SYNCHRONOUS GENERATOR

M. M. Salem**

A. M. Zaki**

O. P. Malik*

*The University of Calgary, Canada

**Electronics Research Institute, Cairo, Egypt
masalem32@hotmail.com

Abstract: A simple neuro-controller for a synchronous generator is presented in this paper. The controller performs the function of the terminal voltage control. By representing the proposed neuro-controller in s-domain, its parameters to ensure system stability can be obtained analytically. Results of simulation studies on a non-linear seventh order generator model with the neuro-controller using calculated parameters are given. Real-time implementation and experimental verification of the neuro-controller as an automatic voltage regulator for a physical model of a single-machine infinite-bus power system is described. Results of experimental studies demonstrate the effectiveness of this simple neuro-controller.
Copyright © 2002 IFAC

Keywords: Neural Control, power System control.

1. INTRODUCTION

The artificial neural network (ANN) technology has matured enough to be applied successfully in many control fields. However, its success will eventually depend on its ability to remove a major obstacle, i.e. the lack of a firm theory. There is no general theory available to assist the developer to design neural networks (El-Sharkawi and Niebur, 1996). Because of the absence of a model, there is no complete theoretical basis to relate the ANN parameters to the characteristic of a system being controlled.

As a first step towards a solution for many problems that face practical applications of neural network in control field, an s-domain model of a simple neuro-controller is developed in this paper. Using this model, stability analysis of the proposed neuro-controller is presented.

Training of the proposed neuro-controller is on-line by the back propagation (BP) algorithm using a modified error function. By representing the neuro-controller learning equations in the s-domain, the controller parameters can be determined analytically. Using the calculated parameters, applications of the neuro-controller as a field excitation controller for a synchronous generator are illustrated by simulation studies and experimental verifications.

2. NEURO-CONTROLLER STRUCTURE

The overall control system with the proposed neuro-controller consisting of one neuron is shown in Fig.1.

The neuro-controller uses a linear hard limit activation function and a modified error feedback function.

2.1 Training Process

The neuro-controller uses a simple procedure to update its weight on-line. There is no need for any off-line training. There is no need for parameter identification or reference model. The neuro-controller is trained directly, in an on-line mode, from the system output and there is no need to determine the states of the system. The neuro-controller uses the sampled values of the system output to compute the error using the modified error function. This error is back propagated through the single neuron to update its weight. Then the output of the neuro-controller is computed which is equal to the neuron weight.

The neuro-controller output can be derived as:

$$u(t)=W(t) \quad (1)$$

$$W(t)=W(t-1)+h*WCT(t) \quad (2)$$

WCT is the neuron weight correction term based on the modified error function. $W(t)$, $u(t)$, h are neuron weight, neuron output and the learning rate respectively.

3. MODEL IN s-DOMAIN

Based on equations (1) and (2), the neuro-controller model in s-domain can be derived as presented in the appendix. The derived model is shown in Fig. 2.

4. APPLICATION TO A SINGLE-MACHINE INFINITE-BUS POWER SYSTEM

4.1 Simplified Linear Model

First a simplified linear model of a synchronous machine is used. This model, as shown in Fig. 3, describes the relation between the generator field voltage and terminal voltage in open loop (DeMello and Concordia, 1969). The neuro-controller as an automatic voltage regulator (AVR) with the simplified machine model is shown in Fig. 4.

The proposed modified feedback function in this case is:

$$f(s) = 1 + k_v s \quad (3)$$

The closed loop transfer function is:

$$\frac{V_t(s)}{V_{ref}(s)} = \frac{h_1 / T'_{do}}{s^2 + \left(\frac{1 + h_1 k_v}{T'_{do}}\right)s + h_1 / T'_{do}} \quad (4)$$

This equation represents a second order system. Assuming that $k_v = 0$ (unity feedback), the system response to a 0.03pu step change in reference input for two values of h_1 is shown in Fig. 5 ($h_1 = 2$ and $h_1 = 10$).

As shown in Fig. 5, the overall system behaves as a stable second order system. But as h_1 increases, the system is more oscillatory. This is an expected result even for a complicated structure neural network (Salem, et al, 2000a).

As mentioned in (Salem, et al, 2000b), while using this function ($k_v = 0$) by the BP algorithm to train a neuro-controller on-line, the controller has no information about the system output movement towards its target value.

To improve the performance of a neuro-controller which is trained on-line by the BP algorithm, a modified function was introduced in (Salem, et al, 2000a). The effect of the modified function on the neuro-controller performance based on analytical studies is described below.

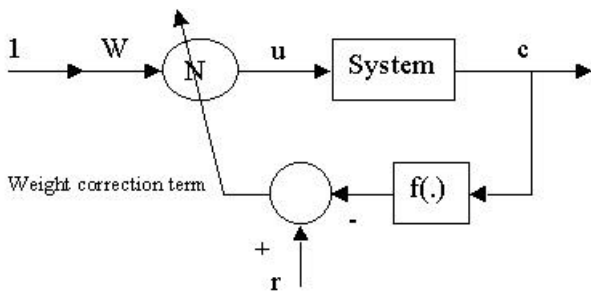


Fig.1 Overall control system with simple neuro-controller

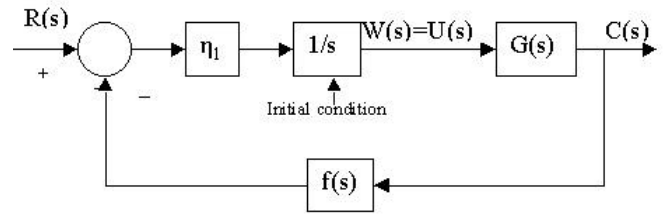


Fig. 2 Neuro-controller model in s-domain

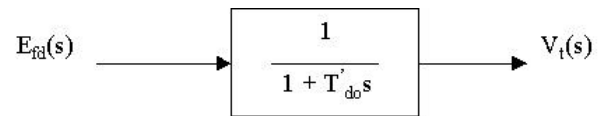


Fig. 3 Simplified linear model of machine on open circuit

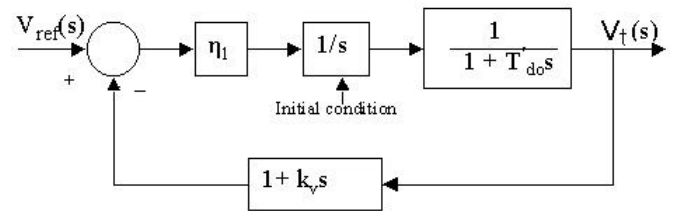


Fig. 4 Neuro-controller as an AVR

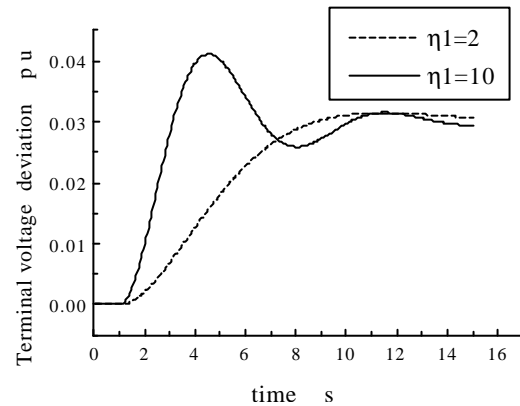


Fig. 5 Neuro-controller performance using unity feedback function

Assuming that the neuro-controller uses the proposed feedback function $f(s)$, equation (3), in its training, a critically damped response to a step change in reference input can be obtained for (Ogata, 1990):

$$k_v = \frac{2\sqrt{h_1 T'_{do}} - 1}{h_1} \quad (5)$$

For $h_1 = 500$, $T'_{do} = 5.67$, then $k_v = 0.21$ for a critically damped response. This critically damped response is shown in Fig. 6 for a 0.03pu step change in reference input. Also, other values for h_1 and k_v for a critically damped response can be obtained. These values are:

$h_1 = 100$, $k_v = 0.47$, $h_1 = 10$, $k_v = 1.4$

The system response corresponding to these values is also shown in Fig. 6. It is clear from this figure that as $h1$ increases the response is better. So, the proposed values for $h1$ and k_v are: $h1 = 500$, $k_v = 0.21$.

It is well known that the simplified linear model of the synchronous generator is different from no-load to load condition. The block diagram shown in Fig. 7 can approximately represent this simplified linear model under load condition (Anderson and Fouad, 1977). Using approximate values for k_3 and k_6 ($k_3=0.3$, $k_6=0.5$ (Anderson and Fouad, 1977)), system response to a 0.03pu step change in reference input is shown in Fig.8. . In this case the neuro- controller uses the proposed values for $h1$ and k_v .

It can be seen that even though the response in this case is slightly under damped, it is still close to the critically damped response as shown in Fig. 8. Then it may be concluded that the neuro controller performance is almost the same in the two cases, which is an expected result for an adaptive controller. Considering that the neuro-controller is an adaptive controller, one can depend on its parameters ($h1$ and k_v) obtained based on a simplified linear model to use them in a highly non-linear model. Application of this neuro-controller with its proposed parameters ($h1 = 500$, $k_v = 0.21$) to control a synchronous generator represented by a non-linear model is presented in the following studies.

4.2 Synchronous Generator Non-Linear Model

A nonlinear seventh-order model is used to simulate the dynamic behavior of the generating unit connected to a constant voltage bus through two parallel transmission lines (Shamsollahi and Malik, 1997).

A computer program is used to simulate in time domain the generator non-linear model with the neuro-controller represented by the s-domain model. The time step in this program is 1ms.

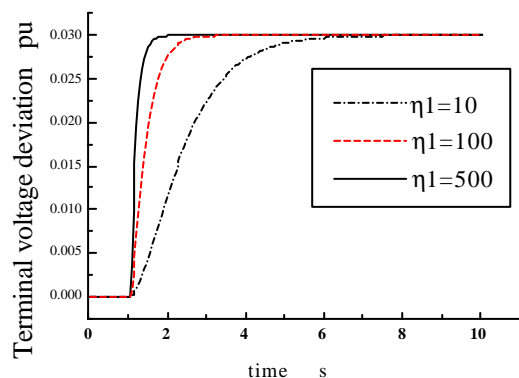


Fig. 6 Neuro-controller performance using modified error function

With the generating unit operating at 0.7pu power and 0.85 power factor lag, a study was conducted to show how the modified function parameters affect the system performance.

In case of $k_v = 0$, the system response to a 0.03pu step change in reference voltage is shown in Fig. 9 for different values of $h1$.

As it is clear from this figure, this controller is not suitable to control the generator terminal voltage.

In case of $k_v = 0.21$, system response to a 0.03pu step increase in reference voltage at 1s followed by a 0.1pu step increase in reference torque at 5s is shown in Fig. 10 for different values of $h1$.

It is seen from Fig. 10(a) that the terminal voltage response is excellent and it is less sensitive to the variations of $h1$. It also demonstrates that the parameters obtained based on a simplified linear model can be used for the non-linear model as well.

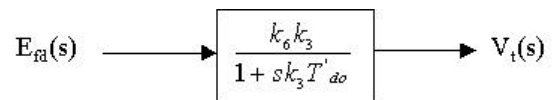


Fig. 7 Machine simplified linear model on load

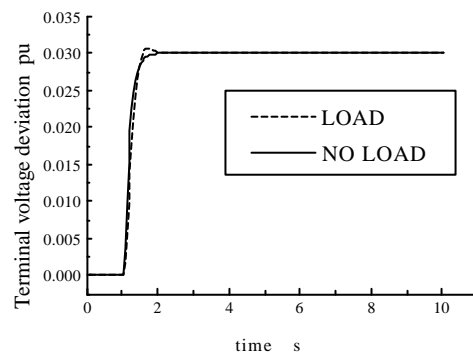


Fig. 8 Neuro-controller performance under load and no load conditions

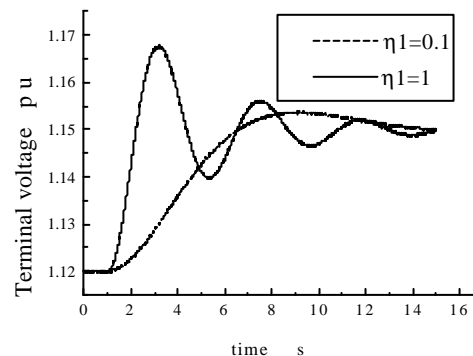


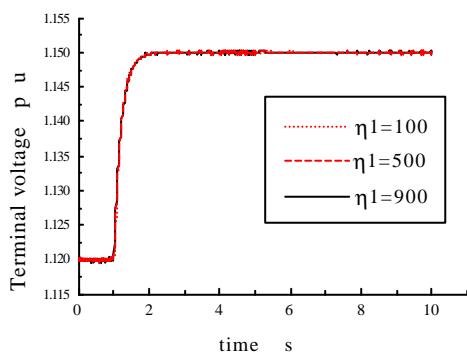
Fig. 9 Neuro-controller performance using unity feedback function

With the function $f(s) = 1+k_v/s$, the neuro-controller is designed to control the generator terminal voltage only. It can be seen from Fig. 10(b) that the system still needs a supplementary signal to enhance stability. As proposed in (Salem, et al, 2000b), including an additional term in $f(s)$ based on the generator speed deviation can enhance system stability.

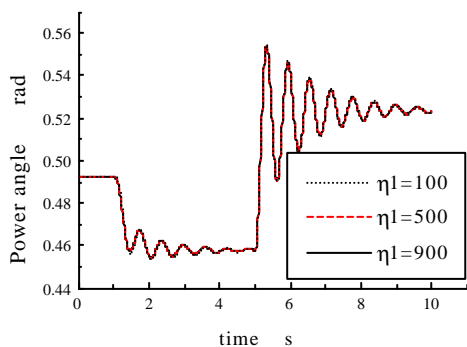
5. REAL-TIME IMPLEMENTATIONS

5.1 Power System Physical Model

Schematic diagram of the physical model of a single-machine infinite-bus power system available in the Power System Research Laboratory at the University of Calgary is shown in Fig. 11. It consists of a 3-phase 3kVA, 220V synchronous micro-alternator driven by a 220V, 30A dc motor. The generator is connected to a constant voltage bus through two parallel transmission lines. The lumped element physical model of the transmission line simulates the performance of a 500kV, 300km long double circuit transmission line, which consists of six Π sections. A system of 3 three-phase Circuit Breakers controlled by a ROM based logic circuit is used. Of the three sets of breakers, two sets are used at the ends of the transmission line and the third is used to apply a three-phase to ground short circuit at any distance. The contact open and close timings can be programmed.



(a) Terminal voltage response



(b) Power angle response

Fig. 10 Neuro-controller performance as an AVR

A Time Constant Regulator (TCR) is used to change the effective field time constant of the generator in order to emulate a large generating unit. Three phase ac voltages at the generator terminals are stepped down, rectified and filtered with a cut-off frequency of 8Hz to obtain the dc terminal voltage (V_t) feedback signal. The neuro-controller is implemented on a DSP board based on the TMS320C30 DSP chip. The terminal voltage (V_t) signal is used in the error function to train the neuro-controller on-line and compute the required field control signal, which is fed to the TCR.

5.2 Hardware Implementation

The neuro-controller is developed on a DSP board supplied by SPECTRUM Signal Processing Inc. It contains a Texas Instruments TMS320C30 DSP chip. The chip is a 32-bit floating-point device with a speed of 16.7 million instructions per second. Its performance is further enhanced through its large on-chip memories, concurrent DMA controller, two external interface ports. Two 200kHz, 16-bit analog I/O channels on board, coupled with direct access to all serial and parallel I/O channels of DSP chip, provide the exterior input-output functions. The 32-bit on-chip timer is programmed by software to a resolution of 120 ns. The board is mounted inside a PC, which is equipped with corresponding development and debugging tools.

The terminal voltage V_t feedback signal is fed to DSP board through the A/D channel. This input signal goes through a filter, which limits the noise and provides anti-aliasing protection. The filtered signal is then stored in a buffer. The DSP chip reads the buffer and computes the control signal V_c . The computed V_c is fed to the D/A channel that filters the signal for smoothing before sending it out. The output signal goes through an amplifier circuit to provide the required field control signal to the TCR.

5.3 Software Implementation

The software consists of two modules, the PC module and the DSP module. The PC module is a C program running on an 80386 microcomputer. The main function of the PC module is to download the DSP program into the DSP board, initialize the communication between the PC and DSP programs, and store DSP input/output data in a file for further analysis. The DSP module is a C program running on the DSP board. This program contains a main function and an interrupt routine. Using software initial value for the reference voltage, the feedback of the terminal voltage and the controller output, the main function trains the neuro-controller for a number of iterations to obtain a stable initial value for the controller output.

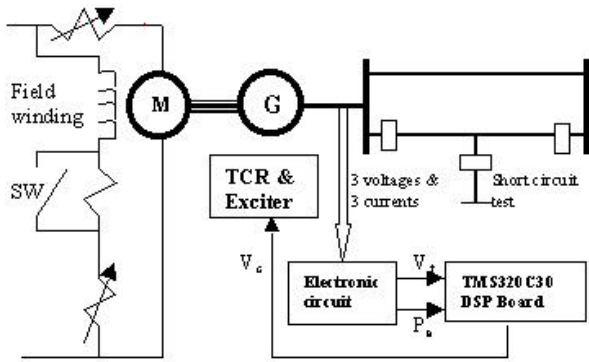


Fig. 11 System model

to 1ms (the sampling period). It then enables the interrupt flag to make the DSP ready to receive the timer interrupt signal. The timer produces repetitive interrupt signals according to the count in its register. This interrupt signal serves in two ways. First, it initiates A/D and D/A conversion, and second, it directs the DSP to execute the interrupt routine.

The interrupt routine reads V_t through the A/D and calculates the error function, updates the neuro-controller weight, and calculates the control output signal V_c . This signal goes through an amplifier to the TCR.

5.4 Experimental Results

The performance of the neuro-AVR (NAVR) was investigated by a number of experimental tests for a variety of operating conditions and disturbances. Results of a representative set of these tests are presented in the following sections. In these tests, the sampling rate is 1ms with a learning rate (q) of 500. Performance of the neuro-controller has been compared with a commercial AVR. This commercial AVR is implemented on a Programmable Logic Controller (PLC) to control the terminal voltage of the generating unit. It is programmed using a function block programming language called FULPA. Three phase voltages and currents at the generator terminals are stepped down to form six input signals to the AVR. The PLC-based AVR computes the required field control signal which is fed to the TCR.

5.5 Voltage Reference Step Change

With the generating unit operating at 0.34 pu power, 0.9 pf lead and terminal voltage of 1.04 pu, a 0.05 step increase in voltage reference is applied at 10s. At time 20s, the change in input reference voltage is removed and the system returns to its original operating condition. As shown in Fig. 12, the system response with the NAVR, which is trained on-line, is very good. Also, the NAVR has small effect on the active power while changing the terminal voltage. It is also clear from this figure that the NAVR control signal is stable and changes smoothly with a dramatic effect on the system performance. To show the

performance clearly for better comparison, the NAVR and commercial AVR plots are intentionally off-set in time.

To further test the performance of the NAVR, the operating condition is changed to 0.7pu power, 0.94 power factor lag and 1.07pu terminal voltage. The same disturbance of 0.05pu step change in input reference voltage is applied with the same timing. System responses to this disturbance with NAVR and the commercial AVR are shown in Fig. 13. Although the operating condition is much different, the NAVR still provides very good performance for the generator terminal voltage.

5.6 Three-Phase Short Circuit Test

With the system operating at 0.8pu power and a 0.9 power factor lag, a transient test was conducted to test the performance of the NAVR in response to a disturbance. In this test a three-phase to ground short circuit was applied at one third of one transmission line, and the fault was cleared 100ms later by disconnecting the line. The disconnected line is successfully reconnected after 1s. The same disturbance was applied after 10s in case of NAVR. It can be seen from Fig. 14 that the NAVR can retain the system stability and keep the system operating at a stable

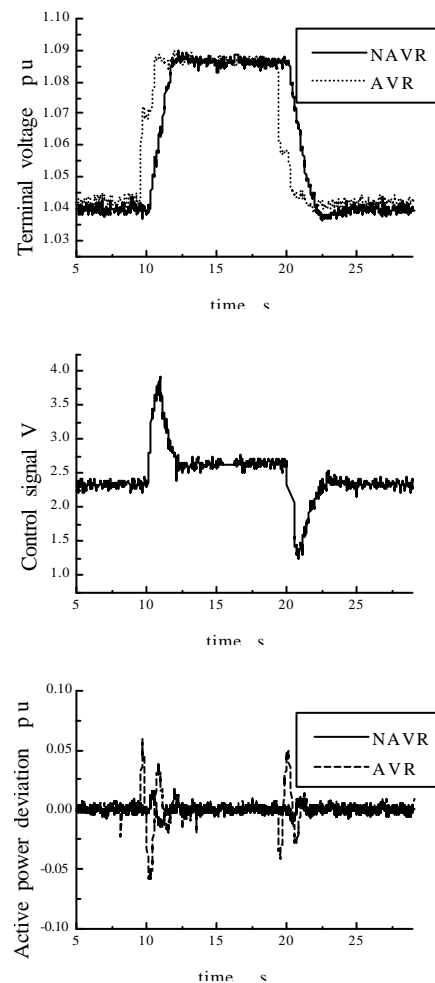


Fig. 12 System response to 0.05pu step disturbance in voltage reference, $P=0.34$, $pf=0.9$ lead.

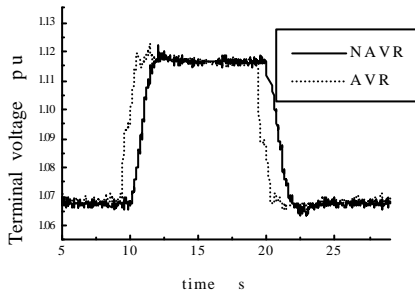


Fig. 13 System response to 0.05pu step disturbance in voltage reference, P=0.7, pf=0.94 lag.

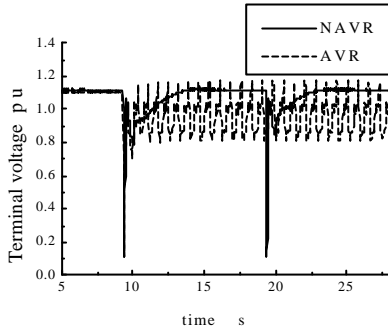


Fig. 14 System response to a three-phase to ground fault P=0.8pu, pf=0.93 lag.

6. CONCLUSIONS

A neuro-controller with a simple structure for a synchronous generator is presented in this paper. The neuro-controller is trained on-line based on a modified function. The neuro-controller consists of one neuron, one weight, hard limit activation function, and a constant input. Based on this simple structure, the neuro-controller is represented in s-domain. Having the neuro-controller in s-domain, its stability analysis with a simplified generator linear model is presented. The neuro-controller parameters are obtained analytically to ensure system stability. The neuro-controller parameters, which are calculated based on a simplified linear model, can be used for a non-linear model. The neuro-controller is used to function as an AVR for a single-machine infinite-bus power system. Results show that the neuro-controller acts as an adaptive controller. The neuro-controller is implemented in a real-time digital control environment.

7. REFERENCES

- Anderson, P. M. and A. A. Fouad, (1977). *Power System Control and Stability*, Iowa State University Press. Ames, Iowa.
- DeMello, F. P. and Concordia, C. (1969). Concepts of Synchronous Machine Stability as Affected by Excitation Control. *IEEE Trans., Power Apparatus and Systems*, Vol. PAS-88, No. 4, April, pp. 316-329.
- El-Sharkawi, M. A. and Niebur, D. (Editors), (1996). *A Tutorial Course on Artificial Neural Networks with Applications to Power Systems*. IEEE Tutorial

Course, Chapters 7 and 12, IEEE Power Engineering Society Special Publ., no. 96TP112-0.

- Ogata, K., (1990). *Modern Control Engineering*. Second Edition.
- Salem, M. M., A. M. Zaki, O. A. Mahgoub, E. Abu El-Zahab, and O. P. Malik, (2000). Generating Unit Excitation Neuro-Controller. *Proceedings, IFAC Symposium on Power Plants and Power Systems Control*, 25-28 April 2000, Brussels, Belgium, pp. 97-102.
- Salem, M. M., A. M. Zaki, O. A. Mahgoub, E. Abu El-Zahab, and O. P. Malik, (2000). On-Line Trained Neuro-Controller with a Modified Error Function. *Proceedings, Canadian Conference on Electrical and Computer Engineering*, May 5-7, 2000, Halifax, pp. 83-87.
- Salem, M. M., A. M. Zaki, O. A. Mahgoub, E. Abu El-Zahab, and O. P. Malik, (2000). Experimental Verification of a Generating Unit Excitation Neuro-Controller. *Proceedings, IEEE Power Engineering Society, Winter Meeting 2000*, Jan. 22-27, 2000, Singapore.
- Shamsollahi, P. and O. P. Malik, (1997). An Adaptive Power System Stabilizer Using On-Line Trained Neural Networks. *IEEE Trans., Energy Conversion*, Vol. 12, No. 4, December 1997, pp. 382-387.

8. APPENDIX

Based on eqns. (1) and (2), the neuro-controller model in s-domain can be obtained. In time domain, (2) can be written as:

$$W(t) - W(t - \Delta t) = \mathbf{h} * WCT(t) \quad (6)$$

Dividing (6) by Δt :

$$\frac{W(t) - W(t - \Delta t)}{\Delta t} = \frac{\mathbf{h}}{\Delta t} error(t) \quad (7)$$

Using the differential form, (7) can be written as:

$$\frac{dW(t)}{dt} = \mathbf{h}_1 * WCT(t) \quad (8)$$

where $\mathbf{h}_1 = \frac{\mathbf{h}}{\Delta t}$

Representing (8) in s-domain:

$$sW(s) = \mathbf{h}_1 * WCT(s) \quad (9)$$

From (1) and (9):

$$U(s) = \frac{\mathbf{h}_1 * WCT(s)}{s} \quad (10)$$

Equation (10) can be represented as shown in Fig. 16

The general form of the weight correction term is:

$$WCT(s) = R(s) - C(s)f(s) \quad (11)$$

R(s): reference input C(s): system output
f(s): feedback function

Complete model of the proposed neuro-controller in s-domain is shown in Fig. 2, where G(s) represents the controlled system.