A FUNCTIONAL LINK NETWORK BASED ADAPTIVE POWER SYSTEM STABILIZER

A Thesis

Submitted to the College of Graduate Studies and Research

In Partial Fulfillment of the Requirements

For the Degree of Master of Science

In the Department of Electrical and Computer Engineering

University of Saskatchewan

Saskatoon, Saskatchewan, Canada

By

Saradha Srinivasan

© Copyright Saradha Srinivasan June 2011. All rights reserved.

PERMISSION TO USE

I agree that the Library, University of Saskatchewan, may make this thesis freely available for inspection. I further agree that permission for copying of this thesis for scholarly purposes may be granted to the professors who supervised the thesis work recorded herein or, in their absence, by the Head of the Department of Electrical and Computer Engineering or the Dean of the College of Engineering. It is understood that due recognition will be given to me and to the University of Saskatchewan in any use of the material in this thesis. Copying or publication or any other use of this thesis for financial gain without approval by the University of Saskatchewan and my written permission is prohibited.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Electrical and Computer Engineering

57 Campus Drive

Saskatoon, Saskatchewan.

Canada, S7N 5A9.

ABSTRACT

An on-line identifier using Functional Link Network (FLN) and Pole-shift (PS) controller for power system stabilizer (PSS) application are presented in this thesis. To have the satisfactory performance of the PSS controller, over a wide range of operating conditions, it is desirable to adapt PSS parameters in real time. Artificial Neural Networks (ANNs) transform the inputs in a low-dimensional space to high-dimensional nonlinear hidden unit space and they have the ability to model the nonlinear characteristics of the power system. The ability of ANNs to learn makes them more suitable for use in adaptive control techniques.

On-line identification obtains a mathematical model at each sampling period to track the dynamic behavior of the plant. The ANN identifier consisting of a Functional link Network (FLN) is used for identifying the model parameters. A FLN model eliminates the need of hidden layer while retaining the nonlinear mapping capability of the neural network by using enhanced inputs. This network may be conveniently used for function approximation with faster convergence rate and lesser computational load.

The most commonly used Pole Assignment (PA) algorithm for adaptive control purposes assign the pole locations to fixed locations within the unit circle in the z-plane. It may not be optimum for different operating conditions. In this thesis, PS type of adaptive control algorithm is used. This algorithm, instead of assigning the closed-loop poles to fixed locations within the unit circle in the z-plane, this algorithm assumes that the pole characteristic polynomial of the closed-loop system has the same form as the pole characteristic of the open-loop system and shifts the open-loop poles radially towards the centre of the unit circle in the z-plane by a shifting factor α according to some rules. In this control algorithm, no coefficients need to be tuned manually, so manual parameter tuning (which is a drawback in conventional power system stabilizer) is minimized. The PS control algorithm uses the on-line updated ARMA parameters to calculate the new closed-loop poles of the system that are always inside the unit circle in the z-plane.

Simulation studies on a single-machine infinite bus and on a multi-machine power system for various operating condition changes, verify the effectiveness of the combined model of FLN identifier and PS control in damping the local and multi-mode oscillations occurring in the system. Simulation studies prove that the APSSs have significant benefits over conventional PSSs: performance improvement and no requirement for parameter tuning.

ACKNOWLEDGEMENTS

I would like to thank Dr. Ramakrishna Gokaraju, for giving me the exciting opportunity to carry out this research at the University of Saskatchewan and directing and supervising my work throughout this period of my career.

The author greatly appreciates the help provided by Mr. Anand Hariharan and Ms. Urvi Malhotra with the simulation study of the proposed control algorithm. The author's final acknowledgement and sincere appreciation go to her husband, daughter, parents and in-laws for their constant support and encouragement.

DEDICATION

This work is dedicated to my husband “Srinivasan Sethuraman”, daughter “Shweta Srinivasan”, beloved parents “Vijayalakshmi Lakshmanan” and “Lakshmanan” and my in-laws “Vijayalakshmi Sethuraman” and “Sethuraman”.

CONTENTS

ABSTRACT ……ii

ACKNOWLEDGEMENTS ……...iii

CONTENTS ……..v

LIST OF FIGURES ……vii

CHAPTER 1 1

INTRODUCTION..... 1

1.1 Power System Stability and Control..... 1

1.2 Power System Damping Controllers..... 2

1.3 Power System Stabilizers 4

1.4 Conventional Power System Stabilizers 4

1.5 Adaptive Power System Stabilizers..... 5

1.5.1 On-line Parameter Identifier..... 7

1.5.2 Controller 8

1.5.3 Control Strategy 8

1.6 Objective..... 9

1.7 Thesis Organization..... 10

CHAPTER 2 11

NEURAL NETWORKS AND SYSTEM IDENTIFICATION..... 11

2.1 Introduction 11

2.2 Artificial Neural Networks 11

2.3 System Identification Using ANN 12

2.3.1 Nonlinear Autoregressive eXogenous Inputs (NARX) Model..... 14

2.3.2 Input-Output model 14

2.3.3 Autoregressive Moving Average (ARMA) model..... 15

2.3.3.1 Auto Regressive (AR) Model 15

2.3.3.2 Moving Average (MA) Model 16

2.3.3.3 ARMA model 16

2.3.4 Nonlinear Autoregressive Moving Average (NARMAX) model 16

2.4 Network Architectures and Learning Algorithm 17

2.4.1 Multi-layered Perceptron (MLP) 17

2.4.2 Recurrent Neural Networks 19

2.4.3 Higher Order Neural Networks 20

2.4.3.1 Radial Basis Function Networks 21

2.4.3.2 Functional Link Networks 22

2.5 Problem Formulation 24

2.6 Functional Link Network Identifier 25

2.6.1 Enhanced Inputs to the FLN Identifier 26

2.6.2 Learning Algorithm for FLN Identifier 26

2.6.3 Linearization of FLN-Identifier 27

2.7 Summary 29

CHAPTER 3 30

POLE-SHIFT LINEAR FEEDBACK CONTROL 30

3.1 Introduction 30

3.2 PS-Control 30

3.2.1 Background 30

3.2.2 Adaptive Pole-Shifting Controller 31

3.2.3 Taylor Series Expansion of Control Signal u(t) in Terms of α_t 33

3.2.4 System Output Prediction, $\hat{y}$ (t+1) 34

3.2.5 Performance Index and Constraints 34

3.2.6 Properties of the PS-Algorithm 35

3.2.6.1 Pole-Shift factor, α 35

3.2.6.2 Optimization function 36

3.2.6.3 Steady-state characteristics 36

3.2.6.4 Dynamic characteristics 37

3.2.6.5 Closed-loop stability 37

3.2.7 Third Order ARMA Model - Control Strategy 37

3.2.8 Linear Discrete System Example 39

3.3 Summary 41

CHAPTER 4 42

APSS USING FLN-IDENTIFIER AND POLE-SHIFT CONTROLLER 42

4.1 Introduction 42

4.2 On-Line Identification Using FLN Identifier 42

4.2.1 Proposed FLN Structure 43

4.2.2 System Model 43

4.2.3 Selection of the order of ARMA Model 43

4.2.4 Control Strategy 45

4.2.4.1 Constraints 46

4.2.4.2 Optimization 47

4.3 Simulation Studies on a Single Machine Infinite Bus Power System 48

4.3.1 Normal Load 48

4.3.2 Light Load 51

4.3.3 Voltage Reference Change 53

4.3.4 Leading Power Factor 54

4.3.5 Fault Test 55

4.3.6 Single Machine Infinite Bus Power System Performance – Discussion 56

4.4 FLN-Identifier and PS-Control – A Multi-Machine Power System Case Study 57

4.4.1 Multi-Machine Power System Model 57

4.4.2 Multi-Machine Studies – Simulation Results 58

4.4.2.1 PSS on one unit 58

4.4.2.2 PSS on three units 59

4.4.2.3 Self-coordination ability of APSS 60

4.4.2.4 Three-phase to ground fault test 61

4.5 Summary 62

CHAPTER 5 63

CONCLUSIONS 63

5.1 Summary 63

5.2 Strengths of the Research Work 63

5.3 Limitations of the Research Work 65

5.4 Suggestions for Future Work 66

REFERENCES 67

Appendix A 70

Back propagation algorithm 70

APPENDIX B 72

SINGLE-MACHINE POWER SYSTEM 72

APPENDIX C 74

MULTI-MACHINE POWER SYSTEM 74

LIST OF FIGURES

Fig 1.1 Model reference adaptive control 6
Fig 1.2 Self-tuning adaptive control 7
Fig 2.1 NARX solution for the system identification 15
Fig 2.2 Multi- layer perceptron 18
Fig 2.3 Recurrent Neural Network using Local Feedback Loops 20
Fig 2.4 Radial-basis function network 22
Fig 2.5 Schematic of functional-link network 23
Fig 2.6 FLN Identifier 26
Fig 2.7 A & B parameters 28
Fig. 3.1Closed loop system block diagram configuration 32
Fig 3.2 Linear discrete time system response to step changes in reference 39
Fig 3.3 Adaptive optimal pole-shifting process for the discrete system 40
Fig 4.1 System model used in APSS Studies 44
Fig 4.2 Power angle variation to a 0.05 pu step increase in torque and return to initial conditions under normal load with PSS 49
Fig 4.3 Speed variation to a 0.05 pu step increase in torque and return to initial conditions under normal load with PSS 49
Fig 4.4 Control signals for a 0.05 pu step increase in torque and return to initial conditions under normal load with PSS 50
Fig 4.5 Terminal voltage response to a 0.05 pu step increase in torque and return to initial conditions under normal load with PSS 50
Fig 4.6 Power angle response to a 0.15 pu change in torque and return to initial conditions under light load with PSS 51
Fig 4.7 Speed variation to a 0.15 pu change in torque and return to initial conditions under light load with PSS 51
Fig 4.8 Control signals for a 0.15 pu change in torque and return to initial conditions under light load with PSS 52
Fig 4.9 Terminal voltage response to a 0.15 pu change in torque and return to initial conditions under light load with PSS 52

Fig 4.10 Power angle response to a 0.02 pu step increase in voltage and return to initial conditions 53
Fig 4.11 V_t response to a 0.02 pu step increase in voltage and return to initial conditions 53
Fig 4.12 Control signals for a 0.02 pu step increase in voltage and return to initial conditions ... 54
Fig 4.13 Power angle response to a 0.2 pu step increase in torque under leading power factor conditions 54
Fig 4.14 Speed variation to a 0.2 pu step increase in torque under leading power factor conditions 55
Fig 4.15 Power angle response to a three-phase to ground fault at the middle of one transmission line 55
Fig 4.16 Control signals for a three-phase to ground fault at the middle of one transmission line 56
Fig 4.17 A five machine power system configuration 58
Fig 4.18 System response with APSS installed on generator G_3 59
Fig 4.19 System response with PSSs installed on G_1, G_2 and G_3 60
Fig 4.20 System response with APSS installed on generators G_1 and G_3 and CPSS on G_2, G_4 and G_5 61
Fig 4.21 System response to a three phase to ground fault with PSSs installed on G_1, G_2 and G_3 62
Fig B.1. AVR and exciter model Type ST1A, IEEE Standard P421.5/D15 72
Fig B.2 CPSS model Type PSS1A, IEEE Standard P421.5/D15 73

CHAPTER 1

INTRODUCTION

1.1 Power System Stability and Control

Electric power systems are complex interconnected systems spread over vast geographical areas. The characteristics of the power systems vary with varying loads. They are also subject to many kinds of disturbances. Maintaining synchronism between the various parts of a power system becomes increasingly difficult, as the systems and interconnections between systems continue to grow.

Power system stability is the ability of an electric power system, for a given initial operating condition, to regain a state of operating equilibrium after being subjected to a physical disturbance, with most system variables bounded so that practically the entire system remains intact[1].

The power system is a highly nonlinear system that operates in a constantly changing environment; loads, generator outputs and key operating parameters change continually. When subjected to a disturbance, the stability of the system depends on the initial operating condition as well as the nature of the disturbance.

Power systems are subjected to a wide range of disturbances, small and large. Small disturbances in the form of load changes occur continually; the system must be able to adjust to the changing conditions and operate satisfactorily. A large disturbance may lead to structural changes due to the isolation of the faulted elements.

For stability study, the torque developed by any particular means can be broken down into synchronizing torque that is in phase with machine rotor angle and damping torque that is in phase with machine rotor speed. Sufficient and necessary conditions for the system to be stable, is that both synchronizing and damping torques have to be positive. Positive synchronizing torque assures restoring of the rotor angle of the machine following an arbitrary small displacement of this angle. Positive damping is necessary to damp out any oscillation due to any perturbation[2].

The objective of all stability studies is to determine whether or not the rotors of the machines being disturbed return to constant speed operation. Various kinds of unstable characteristics and

complicated system dynamic and transient behavior occur because of the interaction between electrical and mechanical parts in an individual element[3, 4]. Stability studies are usually classified into three types depending upon the nature and order of magnitude of the disturbance.

1) Steady-state Stability – When a power system is able to maintain synchronism after it is subjected to small and gradual change in load, it is said to be steady-state stable. Solution to the steady-state problem is to examine the stability of the system under incremental variations about an equilibrium point.
2) Transient Stability – This determines if the system will remain in synchronism following major disturbances such as transmission system faults, sudden load changes, loss of generating units or line switching.
3) Dynamic Stability – This refers to the long time response of the power system to small disturbances. In dynamic stability studies, the excitation system and turbine-governing system are represented along with synchronous machine models which provide for flux-linkage variation in the machine air-gap. This differs from the transient stability because no major shock or impact is considered. The system will lose synchronism if the damping in the system is not strong enough. In contrast to transient stability, dynamic stability tends to be a property of the state of the system.

The power system stability problem can be solved either in the time domain or in the frequency domain. Generally, dynamic stability is treated as an extension of transient stability and is thus solved in the time domain.

1.2 Power System Damping Controllers

Damping of power system oscillations plays an important role not only in increasing the transmission capability but also for stabilization of power system conditions after critical faults, particularly in weakly coupled systems. The main advantage of artificial damping is to make net damping positive.

In conventional synchronous machines damping of speed oscillations is achieved by appropriate control of the amplitude of the rotor MMF by forcing the field current. It is established that machine damping due to forced changes in the amplitude of the rotor MMF is a function of [5]:

- The magnitude of the component of field current in phase with the change in rotor speed.

- The power transfer, that is, the power angle or rotor angle, δ

Several stabilizing signals have been tried from which the following signals have been found to have a significant effect in increasing the damping [6]:

- Change in rotor speed
- Synchronous machine electrical power
- Accelerating power and/or
- The power angle or rotor angle

A good power system must have both steady-state and transient stability. If steady-state stability exists, damping may be required to ensure transient stability by eliminating excessive overshoot. Damping alone may not be sufficient for control of systems which easily lose steady-state equilibrium. In this case, dynamic braking will help to maintain synchronism when generation and loads tend to diverge by slowing the accelerating group until steady-state equilibrium is restored [7]. The fast control of terminal voltage can help to limit the first swing but without adequate damping, stability would be lost on subsequent swings [8].

Flexible AC Transmission System (FACTS) are used at transmission level to provide damping of system oscillations. These devices are used to control power flow along transmission lines and improve power system stability. STATCOM is one of the parallel FACTS devices that are used for voltage regulation. It is also used to improve power system stability by injecting or absorbing reactive power [9].

The Static Synchronous Series Compensator (SSSC) is an important FACTS device which can allow rapid and continuous changes in the transmission line impedance so that the active power flow along the compensated transmission line can be maintained within a specified range under a range of operating conditions. Studies show that SSSC can be very effective in maintaining the transient and oscillatory stability of a power system by providing extra damping to power flow oscillations [10].

FACTS are very fast acting type power electronic devices, but the major problem in applying to the power system is the cost associated with FACTS. A STATCOM, SSC or UPSC cost in the range of 10 to 50 million dollars. FACTS came into existence around early 1990s, but so far only one installation is present in the North American system. But, PSSs are inexpensive technologies and they have been used for number of years to provide additional supplementary controllers for generator excitation system and they cost a fraction of the cost of FACTS (15 to 20,000 dollars).

That makes the PSSs a more economical option for damping oscillations both at the generation level (local mode oscillations) and at the transmission line level (inter-area oscillations) and to bring back stability to the system. The PSS are explained in the next section.

1.3 Power System Stabilizers

To provide extra damping for the system and to improve the dynamic performance, a supplementary control signal in the excitation system and/or the governor system of a generating unit can be used [11]. The objective of power system stabilizers is to provide a supplementary feedback signal in the excitation system of a generator to produce a positive damping [12]. Power system stabilizer (PSS) adds damping to the generator rotor oscillations by controlling its excitation using auxiliary stabilizing signals. These oscillations of concern typically occur in the frequency range of 0.2 to 2.0 Hz. Inadequate damping of the oscillations may limit the ability to transmit power. To provide damping, the stabilizer must produce a component of electrical torque in phase with the rotor speed deviations.

In multi-machine power systems, improving the damping of one generator by using PSS may be insufficient to improve the damping of the other generators. Post fault condition may differ from the pre fault conditions and poorly damped swings may result after severe fault. So, adding PSS to all generators is necessary for damping of both local and inter-area oscillations [13].

The transfer function of the stabilizer must compensate for the gain and phase characteristics of the excitation system, the generator and the power system, which collectively determine the transfer function from the stabilizer output to the component of electrical torque which can be modulated via excitation control. This transfer function is strongly influenced by voltage regulator gain, generator power level and AC system strength.

1.4 Conventional Power System Stabilizers

CPSS are linear controllers with fixed parameters. Normally, the design of CPSSs is based on a linearized machine model corresponding to a nominal operating condition. They provide good dynamic performance for that specific operation condition, but often do not provide satisfactory results over a wide range of operating conditions and system parameters.

CPSS uses a lead/lag compensation network to compensate for the phase shift caused by the low frequency oscillation of the system. By appropriately tuning the parameters of a lead/lag compensation network, it is possible to make a system have desired damping characteristics.

Since power systems are nonlinear systems, their configuration and parameters change with time. The linearized system models used to design the CPSSs are valid only at the operating point that is used to linearize the system. As the fixed parameter controller, CPSS cannot provide optimal performance under wide operating conditions.

A control that "adapts" to changing system characteristics will have the ability to improve power system performance. The idea has led to the research and development of adaptive power system stabilizers (APSSs). In recent years, new approaches have been proposed for PSS design, such as fuzzy logic control, adaptive control and neuro-control [14]. Many of these approaches, however, lack one or more of the three basic and important features that a PSS should have, i.e. simplicity of structure, low computation time and adaptability. Besides, multi-modal nature of oscillations and mutual interactions among generators should be considered in the PSS design.

1.5 Adaptive Power System Stabilizers

Controllers based on adaptive control techniques cope with changing characteristics of the controlled plant by adjusting their parameters online [15]. Whenever an adaptive controller detects changes in system operating conditions, it responds by determining a new set of control parameters.

The adaptive control theory provides a possible way to solve many of the problems associated with the CPSS. Two distinct approaches – direct adaptive control and indirect adaptive control can be used to control a plant adaptively [16].

1. Direct Adaptive Control – In this method, the output of the reference model is compared with the adjustable system. The value of mismatch between the controlled coordinates (y_p) of the system and the model (y_r) is used by the adaptation mechanism to perform parameter adjustment.

$$\| y_r(t) - y_p(t) \| = \varepsilon$$

where ε is called the mismatch error. This kind of adaptive control is often referred to as the model reference adaptive control (Fig 1.1).

The performance of this algorithm depends on the choice of a suitable reference model and the derivation of an appropriate learning mechanism.

2. Indirect Adaptive Control – In this type of control, the controller parameters are adjusted as a function of the parameters of the dynamic model of the controlled plant obtained online. When the characteristics of the actual plant change, the parameters of the model are modified by an identification algorithm that tracks the actual plant, and the controller parameters change correspondingly. This kind of adaptive control is often referred to as the self-tuning adaptive control (Fig 1.2).

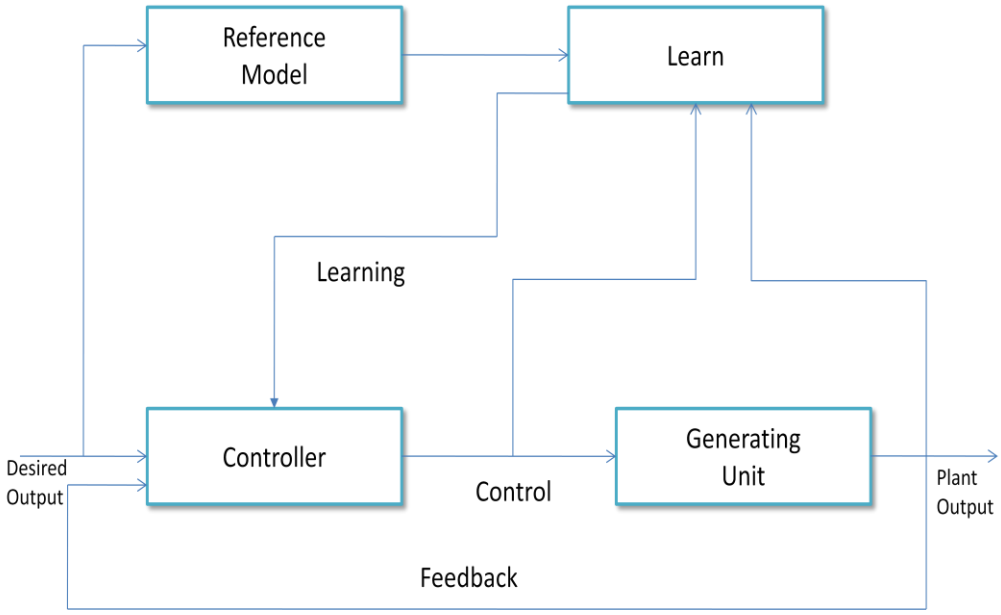


Fig 1.1 Model reference adaptive control

One of the most effective indirect adaptive control techniques is the Self tuning adaptive control. In self tuning adaptive control, the controller parameters are adjusted as a function of the parameters of the dynamic model of the controlled plant obtained. An identification algorithm that tracks the actual plant modifies the controller parameters, when the characteristics of the actual plant change. Because of the flexibility, auto-tuning properties and ease of implementation using PLCs, self-tuning adaptive technique have gained wide spread acceptability. The structure of the self-tuning APSS is shown in Fig 1.2.

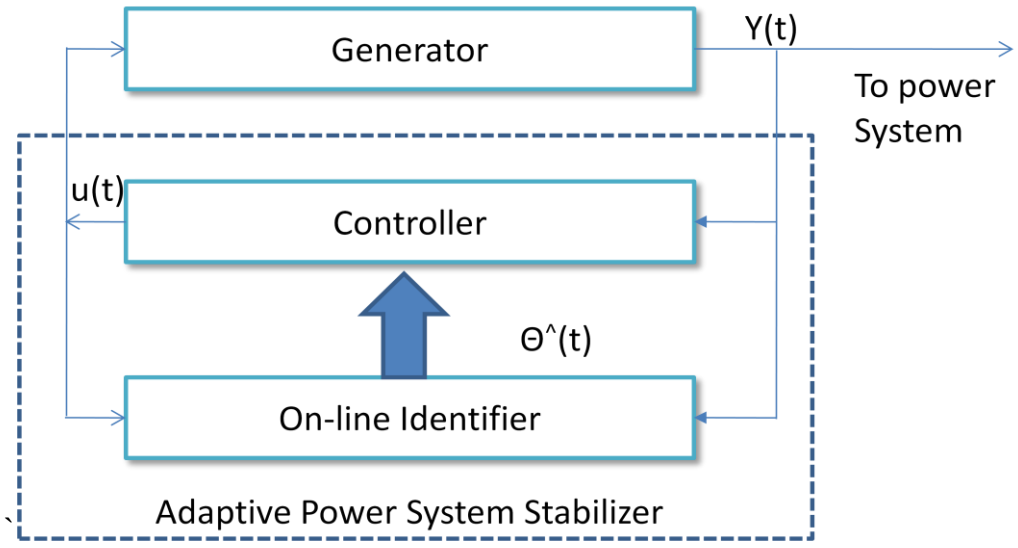


Fig 1.2 Self-tuning adaptive control

The self-tuning adaptive power system stabilizer involves two parts: an on-line parameter identifier and a controller. At each sampling period, a mathematical model is obtained by an on-line identification method to track the dynamic behavior of the plant. The control strategy calculates the control signal based on the on-line identified parameters.

1.5.1 On-line Parameter Identifier

This part is the essence of the APSS which gives the PSS the ability to adapt. The choice of the identification model (i.e. its parameterization) and the method of adjusting its parameters based on the identification error constitute the two principal parts of the identification problem [17]. Input and output of the generating unit are sampled at each sampling instant, and an updated mathematical model is obtained by some on-line identification method to represent the dynamic behavior of the generating unit at that instant of time. For a time varying stochastic system, such as a power system, its dynamic behavior varies from time to time. It is expected that the mathematical model obtained each sampling period can track changes in the controlled system with the on-line identifier. The extent to which the identified model fits the dynamics of the actual generating unit determines the failure or success of the APSS.

For the self-tuning APSS development, the recursive least squares (RLS) method is commonly used. RLS method is one of the simplest of the on-line identification methods; it requires less computation time and has reasonably good numerical stability and fast convergence

property when the system conditions do not change drastically. But the operating condition may change drastically very quickly; often the RLS identification algorithm will not be able to track the changes. For handling the time-varying parameters of the power system, nonlinear type of identification procedures are desirable. Nonlinear algorithms have excellent approximation properties, provide integrity and good fault tolerance behavior and they produce impressive results for poorly defined systems.

A FLN identifier is proposed in this thesis. In a functional-link network, functional expansion of the input increases the dimension of the input pattern. Thus, identification of complex nonlinear dynamic systems and the creation of nonlinear decision boundaries in the multidimensional input space become easier. Also, FLN identification has the advantages of clear structure, simple training principle, analytic mathematical formula etc. A forgetting factor is used to improve the tracking ability of the on-line identifier to the changing power system.

A third order auto-regressive moving average (ARMA) model is used to describe the power system. ARMA model is a linear model. A proper set of parameters of the model corresponds to a certain operating condition of the power plant. The ARMA model has to be updated on-line to track the change of the operating condition. The ARMA model acts as a link between identifier and controller. The PS Control uses the ARMA parameters obtained on-line from the linearization of the identifier to compute the control signal.

1.5.2 Controller

Based on an identified model of the system, the adaptive control is computed by an algorithm which shifts the closed-loop poles of the system to some optimal locations inside the unit circle in the z-domain to minimize a given performance criterion. With the self optimization property, outside intervention in the controller design procedure is minimized, thus simplifying the tuning procedure during commissioning.

Successful application of the APSS can be achieved with the on-line identifier and the control part working together.

1.5.3 Control Strategy

Minimum variance (MV) control algorithms optimize the system output response directly. They are fast, but are not easily susceptible to a stability analysis of the closed-loop system.

Pole Assignment (PA) strategy concentrates on the closed-loop stability rather than the time domain responses. In this algorithm, according to the desired response and stability margin of the

controlled system, the closed-loop poles are assigned to specific locations within the unit circle in the *z*-domain. The disadvantage of this control technique is that it is difficult to select the closed loop poles that meet both stability and time domain response criteria.

Pole Shift control algorithm takes the advantages of both MV and PA algorithm. In PS-Control algorithm, the pole characteristic polynomial of the closed-loop system is assumed to have the same form as the pole characteristic polynomial of the open-loop system, and in closed-loop the open-loop poles are shifted radially towards the center of the unit circle in the z-plane by a shifting factor α. This algorithm self-searches the optimal value of the pole-shift factor α according to the performance index minimization used in the MV control principle. The closed loop poles are restricted to be within the unit circle in the *z*-domain during the optimization.

1.6 Objective

The primary objective of the thesis is to develop a combined functional link neural network based identifier and pole shift controller for APSS application. The objective of this work includes the following aspects:

1. ANNs are capable of complex nonlinear system identification by tuning neuron's weights. Functional link network (FLN) is used for system identification in this thesis. FLN is a single layer structure in which nonlinearity is introduced by enhancing the input pattern with functional expansion. The mapping between the input nodes and enhancement nodes in the FLN is fixed. So, only those weights that contribute to the output are updated using the learning algorithm of the FLN. This significantly reduces the amount of learning in the FLN.
2. A third-order discrete auto-regressive moving average (ARMA) model is used to describe the power system. At every sampling instant, the input-output mapping of the ANN is linearized to get the ARMA parameters. The Pole-Shift (PS) controller uses the on-line updated regression coefficients to calculate the closed-loop poles of the system. To achieve the desired performance, the unstable poles are moved inside the unit circle in the z-plane and the control is calculated.
3. Simulation studies on a single machine and multi-machine power system to verify the effectiveness of the FLN identifier and PS control are performed.

1.7 Thesis Organization

The contents of this thesis are arranged as follows:

- Chapter 1 contains an overview of the objective and motivation of the research.
- Chapter 2 discusses different neural network architectures and the advantages of Functional link neural networks. The reason for selecting FLN for identification is also given in Chapter 2.
- Chapter 3 presents the basic concepts of the pole shifting control strategy. Details of the control algorithm and its advantages over the existing methods are given in this Chapter. A simulation example to demonstrate the proposed control algorithm applied to a linear discrete system is also given in this Chapter.
- Chapter 4 presents simulation studies on a single-machine and a multi-machine power system using FLN-Identifier and PS control. A comparison between CPSS and APSS with FLN Identifier and PS control is also given in this chapter.
- Conclusion and further research in the area of ANN based APSSs that can be explored are summarized in Chapter 5.

CHAPTER 2

NEURAL NETWORKS AND SYSTEM IDENTIFICATION

2.1 Introduction

Artificial Neural Networks (ANNs) transform the inputs in a low-dimensional space to high-dimensional nonlinear hidden unit space and hence are more likely to model the nonlinear characteristics of the power system. Many different ANNs and neural network based control architectures have been developed since eighties. ANNs such as Multi-layer Perceptrons (MLPs), Radial Basis Functions (RBFs) and B-Splines have been developed.

The nonlinear functional mapping properties of neural networks are central to their use in identification and control. Different type of network architectures, their advantages and disadvantages and the rationale for using Functional Link Networks (FLNs) for system identification, enhanced inputs that are used in the FLN, a brief description of Autoregressive Moving Average (ARMA) model and Nonlinear Autoregressive Moving Average with Exogenous Inputs (NARMAX) methodology and an overview of obtaining ARMA parameters from the FLN network using Taylor series expansion are presented in this chapter.

2.2 Artificial Neural Networks

Artificial Neural networks (ANNs) are a powerful tool for many complex applications including functional approximation, nonlinear system identification and control, pattern recognition, classification and optimization because of their nonlinear signal processing and learning capability [18]. The ANN's are capable of generating complex mapping between the input and the output space and form arbitrarily complex nonlinear decision boundaries.

Neural networks have the capability of interpolation over the entire range for which they have been trained. They provide the capability of adaptability not possessed by fixed parameter devices designed and tuned for one operating condition. ANN requires efficient training algorithms and sufficient accurate data for training [19].

A neural network consists of many simple computational elements or nodes arranged in layers and operating in parallel. Strength of connection between the nodes is defined by the weights of the network. The weights are adapted during use to yield good performance. Network architectures, node characteristics and learning rules define the neural networks [20]. Advantages

of neural network include their capability to capture nonlinear relationships between input-output patterns, high parallelism, learning ability and adaptability.

2.3 System Identification Using ANN

System identification is the experimental approach to the modeling of a process or a plant of unknown parameters. It involves experimental planning, the selection of a model structure, parameter estimation and model validation. The procedure of identification, in practice, is iterative in nature that we have to go back and forth between these steps until a satisfactory model is built.

Most of the ANN-based system identification techniques are based on multilayer feed-forward networks such as multilayer perceptron (MLP) trained with backpropagation (BP). These networks are robust and effective in modeling and control of complex dynamic plants [21].

MLP is highly nonlinear in parameters; the mean square error is very complicated. It has a large number of global minima that may lie at infinity for some problems. The error surfaces also generally contain many local minima and may have flat areas where the gradients almost vanish. When the weights fall into these flat regions, learning becomes extremely slow.

Recurrent neural networks (RNNs) are neural networks with one or more feedback loops. The feedback can be local or global (see Section 2.4.2). They can be used as associative memories and also as input-output mapping networks. RNN has atleast one feedback loop when compared to feed-forward neural network. RNN allows signal to flow in both forward and backward directions, giving the network a dynamic memory useful to mimic dynamic systems. But, training these networks becomes difficult due to the feedback connections.

Radial Basis Function (RBF) network is a two layer processing structure. The hidden layer consists of an array of nodes. Each node contains a parameter vector called a centre. The node calculates the Euclidean distance between the center and the network input vector, and passes the result through a nonlinear function. The RBF networks can learn functions with local variations and discontinuities effectively and also possess universal approximation capability. In these networks, choosing an appropriate set of RBF centers for effective learning still remains a problem.

Nonlinear learning can be avoided by initially performing some nonlinear functional transform or expansion of the network inputs and combine the resulting terms linearly. The FLN

structure has good nonlinear approximation ability and learning of the weights is a linear problem. FLN is different from RBF because the centers and widths are free parameters in RBF. Only when the centers and widths are all fixed can a RBF network be regarded as a FLN.

FLN is truly linear in the parameters. Approximation theory not only states that a sufficient FLN with the correct weights can accurately implement an arbitrary continuous function but also ensures that these parameters can always be learnt in the least squares sense. This second property is an advantage of using the extended model set concept or the FLN to model nonlinear systems [22]. The global feedback in FLN model has the potential of reducing the memory requirement significantly. So FLN has been suggested in this thesis for system identification.

FLNN architecture uses a single layer feed forward neural network and uses non-linear functions to expand the input vector. The main advantages of FLNN are:

1. Low computational cost, while maintaining the approximation performance of the MLP network.
2. FLNN has a simpler architecture since they do not have hidden layer and is computationally efficient.
3. FLNNs have faster convergence rate than MLPs.

Functional Link Neural Network (FLNN) has been suggested in this thesis to bridge the gap between the single layer neural networks (linear networks) and the complex, computationally intensive multi layer neural networks. MLPs offer input-output mapping only and linearization of MLP is not possible. The major difference between MLP and FLNN is that FLNN has only input and output layers and the hidden layers are replaced by the nonlinear mappings. The task performed by the hidden layers in MLP is performed by functional expansion in FLNN. An FLN is much simpler than a MLP, since it does not require complete weight updates like the MLP network.

Most APSSs are model-based. The widely used models in APSSs are auto regression moving average (ARMA) and nonlinear auto regression moving average with exogenous inputs (NARMAX) models. Different ANN models like Nonlinear Autoregressive eXogenous Inputs (NARX) model, Input-Output model, ARMA and NARMAX models are discussed in detail from Section 2.3.1 to Section 2.3.4.

2.3.1 *Nonlinear Autoregressive eXogenous Inputs (NARX) Model*

An important class of discrete-time nonlinear systems is the Nonlinear Autoregressive with eXogenous Inputs (NARX) model. The following equation represents a NARX model [23]:

$$y(t) = f(u(t-n_u), \dots, u(t-1), u(t), y(t-n_y), \dots, y(t-1)) \quad (2.1)$$

where *u(t)* and *y(t)* represent input and output of the network at time *t*, n_u and n_y are the input and output order, and function *f* is a nonlinear function. When the function *f* can be approximated by a multilayer perceptron, the resulting system is called a NARX network.

2.3.2 *Input-Output model*

Suppose the unknown model is only accessible through its output. Let the system be a single input, single output kind. Let *y(n)* denote the output of the system due to the input *u(n)* for varying discrete-time *n*. Then, the identification model takes the form:

$$\hat{y}(n+1) = \varphi(y(n), \dots, y(n-q+1), u(n), \dots, u(n-q+1)) \quad (2.2)$$

where q is the order of the unknown system. Eqn. (2.2) is also a representation of the NARX model. At time n+1, the q past values of the output are all available. The model output $\hat{y}(n+1)$ represents an estimate of the actual output $y(n+1)$. The estimate $\hat{y}(n+1)$ is subtracted from $y(n+1)$ to produce the error signal

$$e(n+1) = y(n+1) - \hat{y}(n+1) \quad (2.3)$$

where $y(n+1)$ plays the role of desired response. The error $e(n+1)$ is used to adjust the synaptic weights of the neural network so as to minimize the error in some statistical sense. NARX solution for the system identification problem is shown in Fig 2.1. The identification model shown in Fig 2.1 is of a series-parallel form because the actual output of the system is fed back to the input of the model.

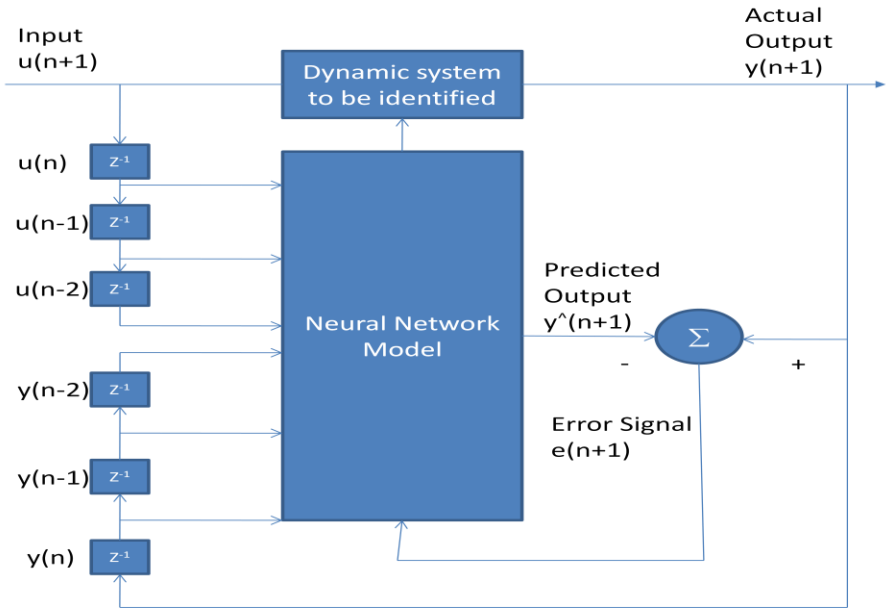


Fig 2.1 NARX solution for the system identification

2.3.3 Autoregressive Moving Average (ARMA) model

ARMA model is a combination of an autoregressive (AR) model and a moving average (MA) model [24]. The order of the ARMA model in discrete time t is described by two integers (g, h), that are the orders of the AR and MA parts respectively. The general expression for an ARMA process y(t) is:

$$y(t) = \sum_{i=1}^{g} a(i).\ y(t-i) + \sum_{i=0}^{h} b(i)\ .\ x(t-i) \qquad (2.4)$$

Where

g is the order of the AR-part of the ARMA model

$a_1, a_2, \dots, a_g$ are the coefficients of the AR part of the model

h is the order of the MA part of the ARMA model

$b_1, b_2, \dots, b_h$ are the coefficients of the MA part of the model

x(t) are elements of the white noise

2.3.3.1 Auto Regressive (AR) Model

Let $\{X_t, t = 0,\ \pm1, \pm2,\ \dots\}$ is stationary time series with zero mean. The generalized form of AR model can be described as follows

$$\emptyset(z^{-1})X_t = \varepsilon_t \qquad (2.5)$$

Where $\emptyset(z^{-1}) = 1 - \emptyset_1 z^{-1} - \emptyset_2((z^{-1})^2) - \cdots - \emptyset_r((z^{-1})^r)$. z^{-1} denotes the backward shift operator, ε_t is stationary white noise with zero mean, r is the order of AR model and $\emptyset_j (j = 1, \dots, r)$ are parameters of the AR model. And AR model can also be described as follows:

$$X_t - \emptyset_1 X_{t-1} - \emptyset_2 X_{t-2} - \cdots - \emptyset_r X_{t-r} = \varepsilon_t \quad (2.6)$$

2.3.3.2 Moving Average (MA) Model

Let $\{X_t, t = 0, \pm 1, \pm 2, \dots\}$ is stationary time series with zero mean. The generalized form of AR model can be described as follows

$$X_t = \theta(z^{-1})\varepsilon_t \quad (2.7)$$

Where $\theta(z^{-1}) = 1 - \theta_1 z^{-1} - \theta_2((z^{-1})^2) - \cdots - \theta_q((z^{-1})^q)$. q is the order of MA model and B and ε_t are the same as AR model and $\theta_j (j = 1, \dots, q)$ are the parameters of the MA model. MA model can also be described as follows:

$$X_t = \varepsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \cdots - \theta_s \epsilon_{t-q} \quad (2.8)$$

2.3.3.3 ARMA model

Let $\{X_t, t = 0, \pm 1, \pm 2, \dots\}$ is stationary time series with zero mean. The generalized form of ARMA model can be described as follows

$$\emptyset(z^{-1})X_t = \theta(z^{-1})\varepsilon_t \quad (2.9)$$

Where $\emptyset(z^{-1}) = 1 - \emptyset_1 z^{-1} - \emptyset_2((z^{-1})^2) - \cdots - \emptyset_r((z^{-1})^r)$ and $\theta(z^{-1}) = 1 - \theta_1 z^{-1} - \theta_2((z^{-1})^2) - \cdots - \theta_q((z^{-1})^q)$. r, q are the orders of the ARMA model. $\emptyset_j (j = 1, \dots, r)$ and $\theta_j (j = 1, \dots, q)$ are parameters of the ARMA model. When n=0, ARMA(m,n) model is AR(m) model. When m=0, ARMA(m,n) model is MA(n) model. The ARMA model can also be described as follows

$$X_t - \emptyset_1 X_{t-1} - \emptyset_2 X_{t-2} - \cdots - \emptyset_r X_{t-r} = \varepsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \cdots - \theta_s \epsilon_{t-q} \quad (2.10)$$

2.3.4 Nonlinear Autoregressive Moving Average (NARMAX) model

NARMAX models are an extension of ARMA models where the sampled response of a system is modeled as the weighted sum of previous input and response values plus nonlinear combinations of these input and response samples [25].

The model widely used in APSSs is an ARMA model. ARMA model is a linear model. A proper set of parameters of the model corresponds to a certain operating condition of the power

plant. This model has to be updated online to track the change of the operating condition. The identifier used in this thesis is a third order ARMA model.

Different types of neural network models namely Multilayer Perceptron (MLP), Recurrent Neural Networks (RNN), Radial Basis Function Network (RBF) and Functional Link Networks (FLN) are described in the next section.

2.4 Network Architectures and Learning Algorithm

ANNs have a number of advantages [26]:

- An ANN has the capability of synthesizing complex and transparent mappings which are very difficult to be expressed in mathematical form.
- Once an ANN is trained, it can provide the ability to solve the mapping problems much faster than conventional methods because of the parallel mechanism.
- ANNs are robust. When the input data is not complete or has some noise, the ANN can still get the correct results.
- ANNs can adjust to the new environment because they can be trained on-line by its error performance.
- They require only less memory.

2.4.1 Multi-layered Perceptron (MLP)

A very popular model of ANN is the multi-layer perceptron shown in Fig. 2.2. This type of ANN has an input layer, an output layer and one or more hidden layers. The layers are interconnected by the pretrained weights. The input layer receives the input vector from the outside as the input of the ANN, and directly passes the signals to the nodes in the next layer. Each hidden layer and output layer consists of a number of elements which are called neurons. Each neuron gets the input from the lower layer. Neurons in hidden layers send their outputs to the neurons in the upper layer through weights, and the outputs of the neurons in output layer are the outputs of the ANN.

Two kinds of signals are identified in this network:

1. *Function Signals:* A function signal is an input signal that comes in at the input end of the network, propagates forward through the network, and emerges at the output end of the network as an output signal.
2. *Error Signals:* An error signal originates at an output neuron of the network, and propagates backward (layer by layer) through the network.

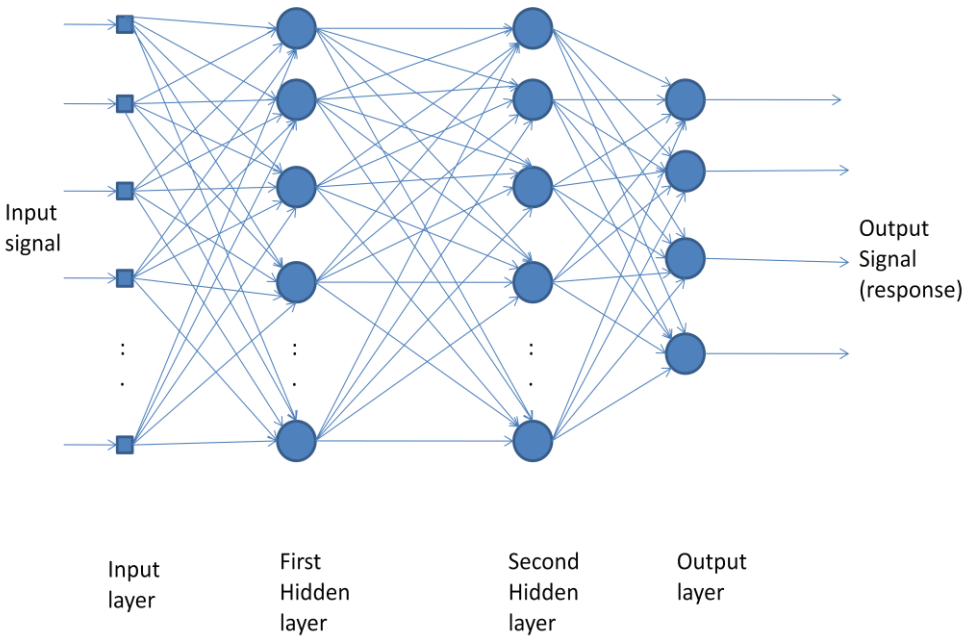


Fig 2.2 Multi- layer perceptron

The output neurons constitute the output layers of the network. The first hidden layer is fed from the input layer made up of sensory units (source nodes); the resulting outputs of the first hidden layer are in turn applied to the next hidden layer.

Consider an *ith* neuron in the *kth* layer. The combining, v_i^k, and activation, x_i^k, terms for the neuron are:

$$v_{ik} = \sum_{j=1}^{n_k-1} w_{ij}^k x_j^{k-1}(t) + b_i^k \tag{2.11}$$

$$x_i^k(t) = F(v_i^k(t)) \tag{2.12}$$

where w_{ij}^k and b_i^k are the connection weights and threshold, respectively, and F(.) is the activation function. The activation function of the hidden nodes is chosen as the following *tan hyperbolic* function

$$F\big(v(t)\big) = \frac{1-\exp(-2v(t))}{1+\exp(-2v(t))} \tag{2.13}$$

The most widely used algorithm used for training a MLP is the back-propagation algorithm. Back-propagation (BP) algorithm is explained in the Appendix A.

Despite the universal approximation capability of MLP networks, their use is limited because they cannot be represented in the form of a function. RBF and FLNs overcome this disadvantage and will be discussed later in Sections 2.4.3.1 and 2.4.3.2.

2.4.2 Recurrent Neural Networks

RNNs are another type of NN used for system identification. Recurrent network architectures incorporate a static multilayer perceptron or parts thereof and they exploit the nonlinear mapping capability of the multilayer perceptron. Recurrent neural network is shown in Fig 2.3. Many recurrent networks can be represented by the state-space model, where the state is defined by the output of the hidden layer fed back to the input layer via a set of unit delays. Following are the properties of the RNN: A recurrent network is said to be controllable if an initial state is steerable to any desired state within a finite number of time steps. The recurrent network is said to be observable if the state of the network can be determined from a finite set of input/output measurements.

There are two modes of training a static multilayer perceptron: batch mode and sequential mode. The sensitivity of the network is computed for the entire training set before adjusting the free parameters of the network in the batch mode training. On the other hand, parameter adjustments are made after the presentation of each pattern in the training set, in the sequential mode of training. There are two modes of training a recurrent network:

a. Epochwise training – For a given epoch, the recurrent network starts running from some initial state until it reaches a new state, at which point the training is stopped and the network is reset to an initial state for the next epoch. The initial state of the next epoch need not be the same as the previous initial state. But, the initial state of the new epoch has to be different from the state that was reached by the network at the end of the previous epoch.
b. Continuous training – This method of training is suitable for situations where there are no reset states available and/or on-line training is required. During continuous training, the network learns while signal processing is being performed by the network. The learning process never stops.

Back-propagation through time and real-time recurrent learning algorithm are two different training algorithms for the recurrent networks. Both the algorithms are based on the method of gradient descent, whereby the instantaneous value of a cost function is minimized with respect to the synaptic weights of the network. Both the algorithms are simple to implement but converge slowly. The back-propagation algorithm is a well established algorithm. The details of this algorithm are given in Appendix A.

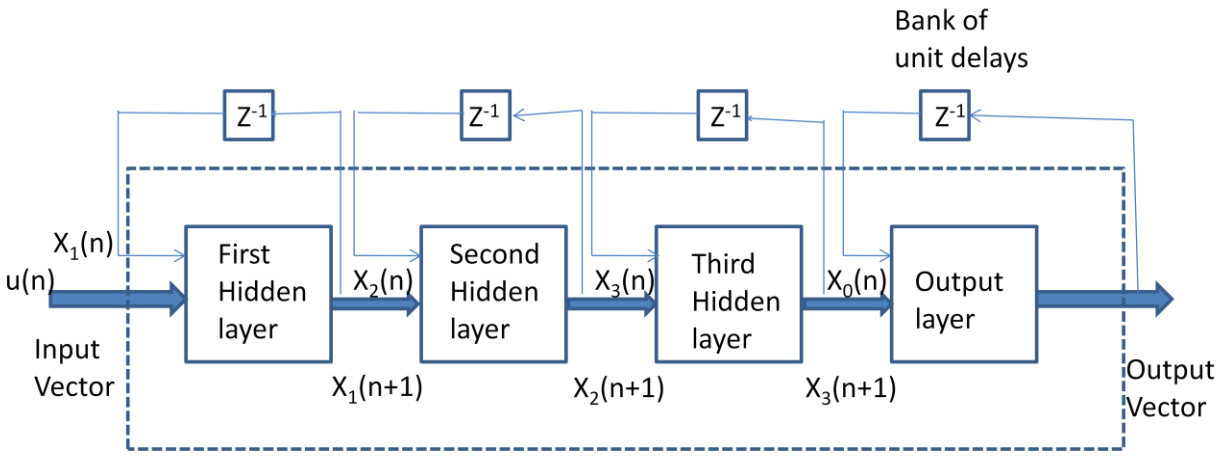


Fig 2.3 Recurrent Neural Network using Local Feedback Loops

2.4.3 Higher Order Neural Networks

Conventional ANN models are incapable of handling discontinuities in the input training data. Artificial neural networks act as "black boxes", and thus are unable to provide explanations for their behavior. Higher order neural networks (HONNs) on the other hand, provide information concerning the basis of the data and hence can be considered as 'open box' models. Furthermore, HONN models are also capable of simulating higher frequency and higher order nonlinear data, compared with those derived from ANN-based models. This is the motivation for developing the identification algorithm with a higher order network in this thesis.

HONNs expand the capabilities of standard feed-forward neural networks by including input nodes that provide the network with a more complete understanding of the input patterns and their relations. Basically, the inputs are transformed so that the network does not have to learn some basic mathematical functions. The inclusion of these functions enhances the network's understanding of a given problem and has been shown to accelerate training on some applications [27].

A major advantage of HONNs is that only one layer of trainable weights is needed to achieve nonlinear separability, unlike the typical MLP or feed-forward networks. They are simple in their architecture and require fewer numbers of weights to learn the underlying equation when compared to ordinary feed forward networks, in order to deliver the same input-output mapping. Consequently, they can learn faster. Also, higher order terms in HONNs can increase the information capacity of neural networks. Larger capacity means that the same function or problem can be solved by network that has fewer units.

The simpler characteristic of HONNs, having a single layer of trainable weights, can offer a large saving of hardware in the implementation. HONNs have certain unique characteristics; stronger approximation property, faster convergence rate, greater storage capacity and higher fault tolerance than lower order neural networks. The networks have been considered as good candidates, due to their design flexibility for given geometric transforms, robustness to noisy inputs, inherent fast training ability and nonlinearly separable. Some of the important HONNs are discussed in Section 2.4.3.1 and 2.4.3.2.

2.4.3.1 Radial Basis Function Networks

The construction of a radial-basis function (RBF) network, involves three layers with entirely different roles. The input layer is made up of source nodes that connect the network to its environment. The second layer, the hidden layer in the network, applies a nonlinear transformation from the input space to the hidden space; in most applications the hidden space is of high dimensionality. The output layer is linear, supplying the response of the network to the activation pattern applied to the input layer.

The input layer does not process the information; it only distributes the input variables to the hidden layer. Each neuron on the hidden layer represents a radial function and the number of radial functions depends on the problem to be solved. A radial basis function network is shown in Fig 2.4.

The mostly used radial basis function is the symmetrical Gaussian function. They are characterized by two parameters: the centroid represented by c_j and the width represented by σ_j. The output from the j^{th} Gaussian kernel for an input vector x_i can be estimated by the following equation:

$$\phi_j(x) = \exp\left(-\left(\frac{\|x_i - c_j\|^2}{{\sigma_j}^2}\right)\right) \tag{2.14}$$

Where $x_i = \{x_{i1}, x_{i2}, \ldots\ldots, x_{id}\}$ is the d-dimensional input vector, c_j is the centroid vector, and σ_j is the width of the radius, which determines the portion of the input space, where the j^{th} kernel will have a non significant zero response.

MLP and RBF networks differ from each other in several important aspects:

1. RBF network has a single hidden layer, but MLPs may have one or more hidden layers.
2. The hidden layer of RBF network is nonlinear, whereas the output layer is linear. But, the hidden and output layers of MLP are usually all nonlinear.

3. MLPs construct global approximations to nonlinear input-output mapping. The RBF networks using exponentially decaying localized nonlinearities (e.g Gaussian functions) construct local approximations to nonlinear input-output mappings. For the approximation of a nonlinear input-output mapping, MLP may require a smaller number of parameters than the RBF network for the same degree of accuracy.

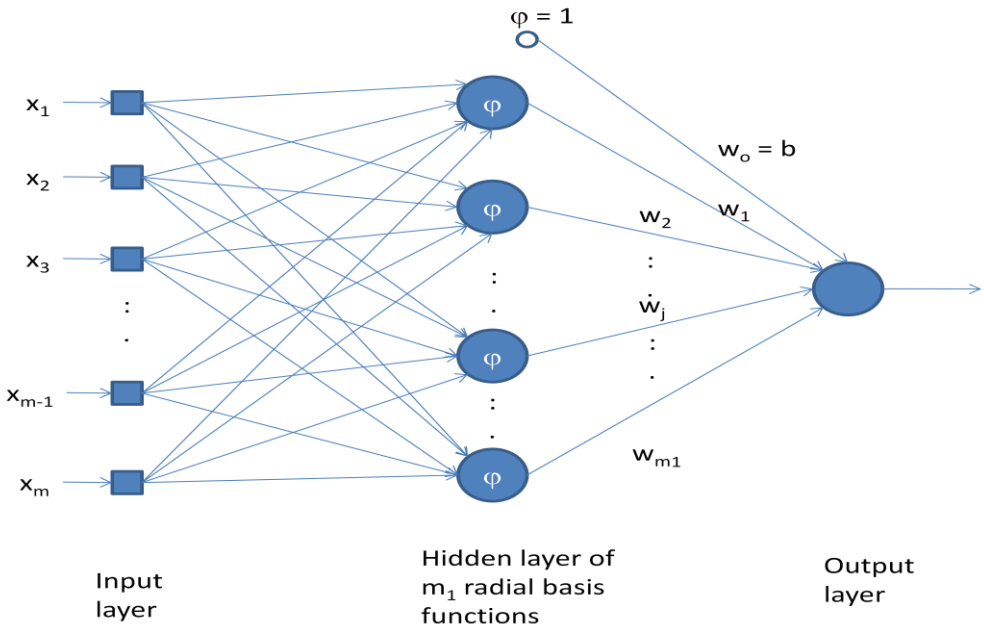


Fig 2.4 Radial-basis function network

RBF networks can be used to identify the time-varying parameters of the power system. But, it has been proven that the RBF network becomes unnecessarily large by arbitrarily choosing some data points as centers and trying to reduce the mean squared error.

2.4.3.2 Functional Link Networks

A multilayer perceptron with a suitable architecture is capable of approximating virtually any function of interest. But, finding such a network is not easy [28]. Problems such as local minima trapping, saturation, weight interference, initial weight dependence and overfitting, make neural network training difficult. Moreover, most neural learning methods, being based on gradient descent, cannot search the non-differentiable landscape of multilayer architecture. The main idea behind a FLN is the use of links for effecting information processing transformations. The essential idea behind FLN is the generation of an enhanced pattern to be used in place of the actual pattern [29].

An easy way to avoid the architectural problems of MLPs is by removing the hidden layers. The removing process can be executed without giving up nonlinearity, provided that the input layer is endowed with additional higher order units, also known as sigma-pi units. This is the idea behind higher order networks (HONs). FLNs are basically a subset of HONs [30]. FLNs are HONs without hidden units. Despite their linear nature, FLNs can capture non-linear input-output relationships, provided that they are fed with an adequate set of polynomial inputs. The general architecture of the FLN is shown in Fig 2.5.

In contrast to the linear weighting of the input pattern produced by the linear links of artificial neural network, the functional link acts on an element of a pattern or on the entire pattern itself by generating a set of linearly independent functions, then evaluating these functions with the pattern as an argument. Thus class separability is possible in the enhanced feature space.

FLNs are simple in their architectures and require less number of weights to learn the underlying approximating polynomials. This potentially reduces the number of required training parameters. As a result, they can learn faster, since each iteration of the training procedure takes less time. This makes them suitable for complex problem solving where the ability to retrain or adapt to new data in real time is critical.

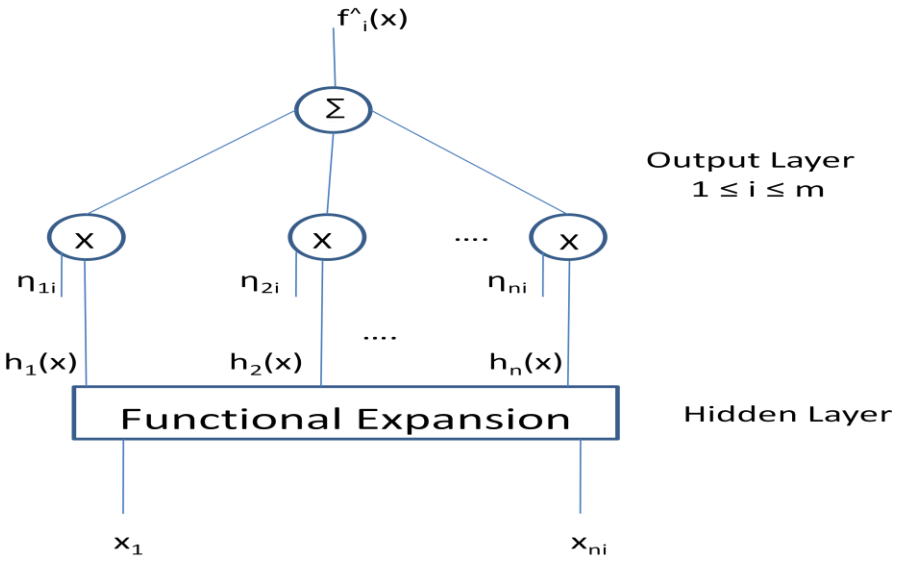


Fig 2.5 Schematic of functional-link network

Functional links not only increases learning rates, but also simplifies the learning algorithms. There are two functional link network models, the functional expansion model and the tensor (or

outerproduct) model. In the functional expansion model, the functional link acts on each node singly. In the tensor or outerproduct model each component of the input pattern multiplies the entire input pattern vector. Rapid quadratic optimization in the learning of weights and simplification in hardware and computational procedures are the significant advantages of using a flat net architecture [31].

As FLNs do not have any hidden layer; the architecture becomes simple. Thus, nonlinear modeling can be accomplished, by means of a linear learning rule, such as delta rule. As FLNs involve linear mapping in polynomial space, they can easily map linear and nonlinear terms [40]. FLN may be conveniently used for function approximation and pattern classification with faster convergence rate and lesser computational load than an MLP structure.

In the functional expansion model the functional link acts on each node singly. It might induce the same additional functionalities for each node in the input pattern. The tensor or outer-product model is simply a special case of the functional-expansion model. In this model, each component of the input pattern multiplies the entire input pattern vector. The functional link in this case generates an entire vector from each of the individual components.

In the tensor model, the effect of the nonlinear functional transform is to change the representation of the input pattern, so that instead of being described in terms of a set of components, it is described as

$$\{x_i, x_i x_j\}\ (j \geq i) \tag{2.15}$$

Or as $\{x_i, x_i x_j, x_i x_j x_k\}(j \geq i, k \geq j \geq i)$ and so on.

Once the structure of FLN or the set of model bases is given, the RLS algorithm provides an efficient means for real-time adaptation of the network weights.

2.5 Problem Formulation

Consider a dynamic system which is governed by the following nonlinear relationship:

$$y(t) = f\left(y(t-1), \ldots., y(t-n_y), u(t-1), \ldots., u(t-n_u)\right) + e(t) \tag{2.16}$$

where

$$y(t) = [y_1(t), \ldots., y_m(t)]^T \tag{2.17}$$

$$u(t) = [u_1(t), \ldots., u_r(t)]^T \tag{2.18}$$

$$e(t) = [e_1(t), \ldots., e_m(t)]^T \tag{2.19}$$

are the system output, input and noise vectors respectively; n_y and n_u are the corresponding lags in the output and input; and f(.) is some vector-valued nonlinear function. The aim is to realize or to approximate the underlying dynamics f(.) using neural networks. Introduce the network input vector

$$x(t) = \left[y^T(t-1) \dots y^T\left(t-n_y\right)u^T(t-1) \dots u^T(t-n_u)\right]^T \quad (2.20)$$

with a dimension

$$n_I = m \times n_y + r \times n_u \quad (2.21)$$

The modeling and identification task can then be formulated using the neural network input-output response

$$\hat{y}(t) = \hat{f}(x(t)) \quad (2.22)$$

as the one-step-ahead predictor for y(t).

The system representation (2.2) is a simplified case of the general nonlinear system known as the NARMAX model:

$$y(t) = f\left(y(t-1), \dots., y\left(t-n_y\right), u(t-1), \dots., u(t-n_u), e(t-1), \dots., e(t-n_e)\right) + e(t) \quad (2.23)$$

2.6 Functional Link Network Identifier

Learning algorithm for the FLN identifier, enhanced inputs to the identifier and the A and B parameters are presented in this section.

System identification plays a very important role in the success of the APSS and has received much attention in its development. The tracking property of the identification methods is very important for the APSS to achieve desired dynamic performance. When the power system is subject to a disturbance or its operating point changes, the APSS must sense this change quickly and smoothly adjust the identified model to represent the new state of the power system [32]. The FLN model shown in Fig 2.6 is used to identify the ARMA parmeters. Performing nonlinear functional transform or expansion of the network inputs and combining the resulting terms linearly will avoid nonlinear learning.

The value of the new basis function $h_j(x)$ depends only on the input x and a given functional expansion contains no other free parameters. Once the structure of the FLN is given, the RLS algorithm provides an efficient means for real-time adaptation of the network weights.

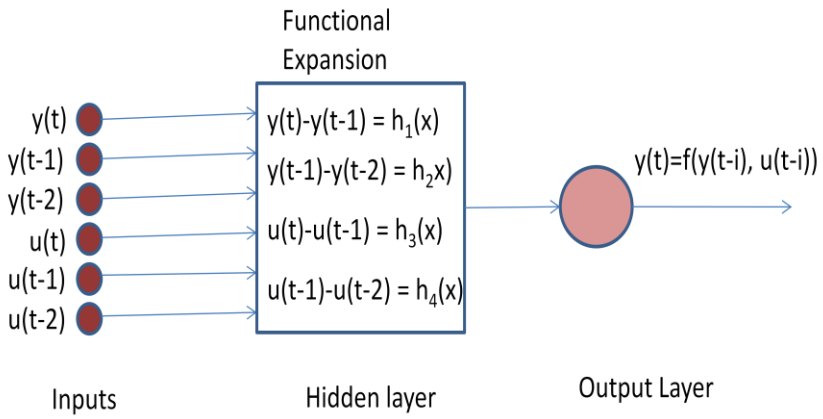


Fig 2.6 FLN Identifier

2.6.1 Enhanced Inputs to the FLN Identifier

$$y(t) = [\mu 1\ \mu 2\ \mu 3\ \mu 4\ \mu 5\ \mu 6\ \mu 7\ \mu 8\ \mu 9\ \mu 10]\begin{bmatrix} x1 \\ x2 \\ x3 \\ x4 \\ x5 \\ x6 \\ (x1-x2) \\ (x2-x3) \\ (x4-x5) \\ (x5-x6) \end{bmatrix} \tag{2.24}$$

Where $x1 = y(t),\ x2 = y(t-1),\ x3 = y(t-2),\ x4 = u(t),\ x5 = u(t-1), x6 = u(t-2)$

2.6.2 Learning Algorithm for FLN Identifier

Nonlinear learning rules have the disadvantage of unacceptably slow learning rate and local minima problems. To avoid nonlinear learning, nonlinear functional transform or expansion of the network inputs can be performed initially and then the resulting terms can be combined linearly. Once the structure of the FLN or the set of model bases is given, RLS algorithm provides an efficient means for real-time adaptation of the network weights.

The weight vector, θ, can be calculated by the following RLS equations:

$$G(t) = \frac{P(t)\,\varphi(t)}{\lambda(t)+\varphi^T(t)P(t)\varphi(t)} \tag{2.25}$$

$$P(t+1) = \frac{1}{\lambda(t)}[P(t) - G^T(t)P(t)\varphi(t)] \tag{2.26}$$

$$\theta(t+1) = \theta(t) + G(t)[y(t) - \theta^T(t)\varphi(t)] \tag{2.27}$$

Where λ(t) is the forgetting factor, P(t) is the covariance matrix and G(t) is the modifying gain vector.

The forgetting factor λ (t) can be chosen as a constant or a variable. The forgetting factor is usually computed according to the rule

$$\lambda(t) = \lambda_0\lambda(t-1) + (1-\lambda_0) \tag{2.28}$$

The identification error can be defined as:

$$e(t) = y(t) - \theta^T\varphi(t) \tag{2.29}$$

The accuracy of the identified model and the change in the environmental conditions are indicated by the absolute value of e (t). It can be used as a criterion to determine the value of the forgetting factor λ(t):

$$\lambda(t) = \min\{\lambda_0\lambda(t-1) + (1-\lambda_0), 1 - \frac{e^2(t)}{\Sigma_0}\} \tag{2.30}$$

Where Σ_0 is a constant. Σ_0 is different for different systems under investigation. For smooth parameter identification, a minimum forgetting factor λ_{min} must be specified. A maximum forgetting factor λ_{max} can also be specified to keep the identifier more sensitive to the system changes under steady state operation.

$$\lambda_{min} < \lambda(t) < \lambda_{max} \tag{2.31}$$

2.6.3 Linearization of FLN-Identifier

FLN network is trained on-line every sampling period making it an adaptive approach. The on-line updating of weights allows the APSS to track the operating conditions of the power system and any changes in the ARMA parameters.

Linear parameters of the standard ARMA model are obtained by linearizing the output of the FLN $y(t) = f(y(t-i), u(t-i))$ using Taylor series expansion and retaining the first-order terms and ignoring the higher order terms at each sampling instant:

$$\Delta y = \frac{\partial y}{\partial y(t-1)}\Delta y(t-1)| + \cdots + \frac{\partial y}{\partial u(t-1)}\Delta u(t-1) + \cdots \tag{2.32}$$

The partial derivative terms, $\left[\partial y/\partial x_i\right]$ of Eqn. 5.14, determine the system parameters θ which can be used in computing the control signal.

$$\theta(1) = a_1 = \frac{\partial y}{\partial x_1} = \mu_1 + \mu_7 \tag{2.33}$$

$$\theta(2) = a_2 = \frac{\partial y}{\partial x_2} = \mu_2 - \mu_7 + \mu_8 \quad (2.34)$$

$$\theta(3) = a_3 = \frac{\partial y}{\partial x_3} = \mu_3 - \mu_8 \quad (2.35)$$

$$\theta(4) = b_1 = \frac{\partial y}{\partial x_4} = \mu_4 + \mu_9 \quad (2.36)$$

$$\theta(5) = b_2 = \frac{\partial y}{\partial x_5} = \mu_5 - \mu_9 + \mu_{10} \quad (2.37)$$

$$\theta(6) = b_3 = \frac{\partial y}{\partial x_6} = \mu_6 - \mu_{10} \quad (2.38)$$

Typical curves for the on-line variation of the ARMA parameters during a three phase to ground fault applied at 1s at the middle of one transmission line are shown in Fig 2.7. The fault is cleared 50ms later by opening the breakers at both ends. The initial operating conditions are 0.5 pu active power delivered to the bus at 0.93 pf lag.

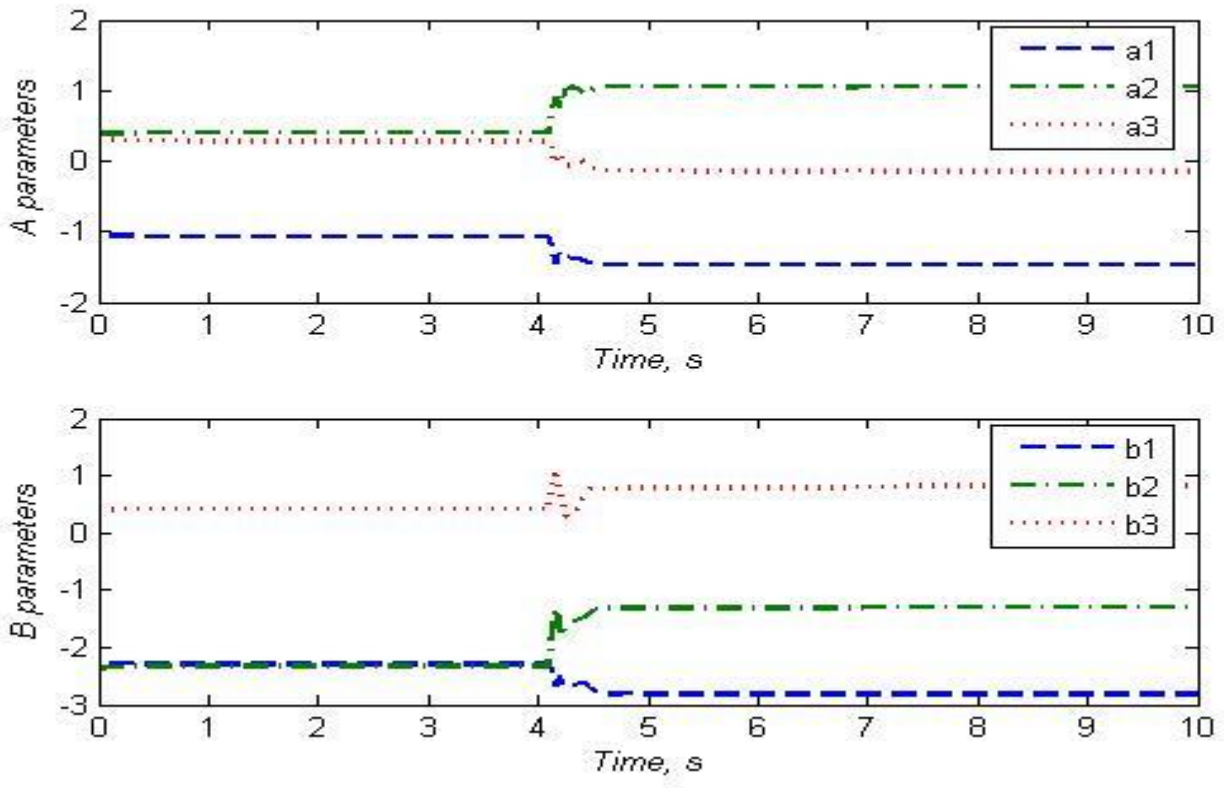


Fig 2.7 A & B parameters

2.7 Summary

System identification using artificial neural networks is explained in Section 2.3. Advantages and disadvantages of using MLP, RNN, RBF and FLN for system identification are also presented. Widely used models in the APSS are discussed from Section 2.3.1 to 2.3.3.

Section 2.4 presented various neural network architectures and their learning algorithms. The characteristics of FLN that make it suitable for system identification are also presented in this section.

Enhanced inputs that are being used in this thesis are discussed in Section 2.6.1. A brief overview about the linearization of FLN-identifier to obtain the A and B parameters of the ARMA model and a graph representing these parameters are given in Section 2.6.3. A detailed discussion about the Pole-Shift controller is presented in Chapter 3.

CHAPTER 3

POLE-SHIFT LINEAR FEEDBACK CONTROL

3.1 Introduction

A general introduction to system identification using neural networks and learning algorithms were given in Chapter 2. The identifier adjusts online the parameters of a model of the system continuously, in response to the changing network and operating conditions of the system. The controller has to calculate a control that ensures optimum damping of active power oscillations, based on the updated system parameters.

Fixed parameter controllers cannot provide effective control over a wide operating range for systems that are nonlinear, time-varying and subject to uncertain noise. An adaptive controller which has the ability to adjust its own parameters, and even the structure, online according to the environment will yield a satisfactory control performance [33].

Pole Assignment methods are generally implemented assuming that a low order discrete model is a close approximation to the power system. But, the power system is a non-linear, high-order continuous system and it is hard for the discrete model to describe the dynamic behavior of the system accurately. Also, a significant amount of computation will be required, if a high-order model is used to represent the system [34].

This Chapter explains an adaptive Pole Shifting Control algorithm, that combines the advantages of Minimum Variance (MV) and Pole-Assignment (PA) control algorithms.

3.2 PS-Control

3.2.1 Background

MV control algorithms [35] have the common feature of optimizing the system output response directly, thus are fast, but are not easily susceptible to a stability analysis of the closed loop system. PA based adaptive control algorithm places emphasis on the stability of the closed-loop system rather than the system output response directly. In this algorithm, only the desired system closed-loop poles are specified, and the update of controller parameters is based on explicit system identification. MV controller shifts all the poles towards the origin, but the PA controller has the freedom to place the poles at other locations. PA algorithm can be applied to the non-minimum phase (NMP) systems easily and safely. With proper choice of the closed-loop

poles it can yield satisfactory dynamic response. This algorithm ensures closed-loop stability as long as the identified parameters converge to their true values and there are no control limits. But, it not always easy to choose suitable closed loop pole locations, especially if the system operates over a wide range.

Minimum variance controllers can only be used for small disturbance cases provided the system is minimum phase (MP). Pole assigned controllers can only be used in the cases where the system is well known, although it can damp out large disturbances efficiently. Linear quadratic and pole-shifting controllers can be most beneficial when the system is not well known, but has stable open-loop poles [36].

A self-tuning controller based on pole assignment has the advantages of overcoming the drawbacks of minimum variance control and of incorporating comparatively simple control calculation algorithm. In addition it always produces a much smoother control action which is more acceptable. Based on pole assignment control, pole-shifting control further simplifies the calculation algorithm while retaining the basic advantages. Instead of considering both the poles and the zeros of the system, PS control considers only the poles of the system and allows the zeros to be configured according to the design algorithm. This simplification is very significant for on-line computer control [37]. The use of a self-searching pole-shifting technique increases the flexibility when applied to varying operating conditions encountered in power systems. It eliminates the necessity of choosing the closed-loop pole locations and still possesses the quality of robustness of 'pole-assignment' and ease of reference signal tracking of 'self-tuning' controller.

3.2.2 Adaptive Pole-Shifting Controller

Consider a system modeled by

$$A(z^{-1})y(t) = B(z^{-1})u(t) + e(t) \tag{3.1}$$

Where y(t), u(t), e(t) are system output, system input and white noise respectively, and $A(z^{-1})$, $B(z^{-1})$ take the form,

$$A(z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_i z^{-i} + \cdots + a_{n_a} z^{-n_a} \tag{3.2}$$

$$B(z^{-1}) = b_1 z^{-1} + \cdots + b_i z^{-i} + \cdots + b_{n_b} z^{-n_b} \tag{3.3}$$

where n_a and n_b are the orders of the polynomials $A(z^{-1})$ and $B(z^{-1})$ respectively.

Assuming the system parameters $\{a_i\}a$, $\{b_i\}$ are known from a real-time parameter identification method. Based on the identified system parameters, procedure for computing the control signal using PS algorithm is described below:

Assume the feedback loop has the form as shown in Fig 3.1,

$$\frac{u(t)}{y(t)} = -\frac{G(z^{-1})}{F(z^{-1})} \tag{3.4}$$

Where

$$F(z^{-1}) = 1 + f_1 z^{-1} + \cdots + f_i z^{-i} + \cdots + f_{n_f} z^{-n_f} \tag{3.5}$$

$$G(z^{-1}) = g_0 + g_1 z^{-1} + \cdots + g_i z^{-i} + \cdots + g_{n_g} z^{-n_g} \tag{3.6}$$

Where

$$n_f = n_b - 1, n_g = n_a - 1 \tag{3.7}$$

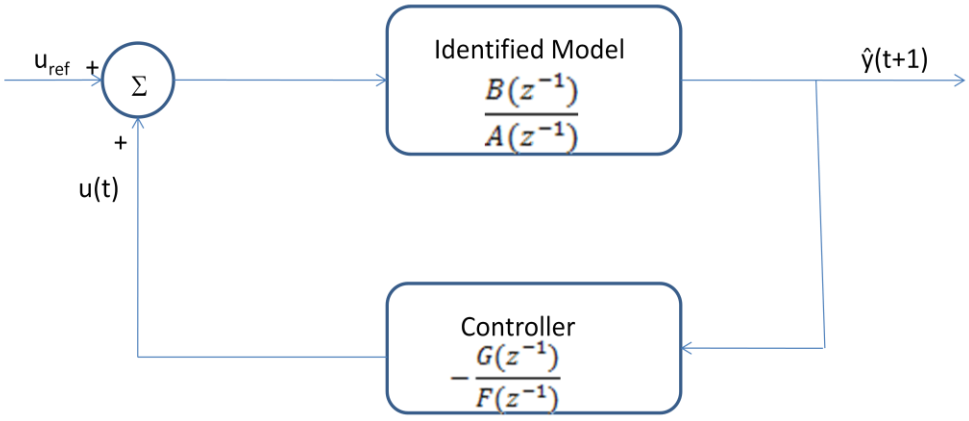


Fig. 3.1Closed loop system block diagram configuration

In the pole-shifting algorithm, the characteristic polynomial of the closed-loop system is assumed to have the same form as the characteristic polynomial of the open-loop system, and the open-loop poles are shifted radially towards the centre of the unit circle in the z-plane by a shifting factor α. Thus the following equation holds:

$$A(z^{-1})F(z^{-1}) + B(z^{-1})G(z^{-1}) = A(\alpha z^{-1}) \tag{3.8}$$

Expanding both sides of Eqn. (3.8) and comparing the coefficients gives

$$\begin{bmatrix} 1 & 0 & . & 0 & b_1 & 0 & . & 0 \\ a_1 & 1 & . & 0 & b_2 & b_1 & . & 0 \\ . & a_1 & . & . & . & b_2 & . & . \\ a_{n_a} & . & . & 1 & b_{n_b} & . & . & b_1 \\ 0 & a_{n_a} & . & a_1 & 0 & b_{n_b} & . & b_2 \\ . & 0 & . & . & . & 0 & . & . \\ . & . & . & . & . & . & . & . \\ 0 & 0 & . & a_{n_a} & 0 & 0 & . & b_{n_b} \end{bmatrix} \begin{bmatrix} f_1 \\ . \\ . \\ f_{n_f} \\ g_0 \\ . \\ . \\ g_{n_g} \end{bmatrix} = \begin{bmatrix} a_1(\alpha - 1) \\ a_2(\alpha^2 - 1) \\ . \\ . \\ a_{n_a}(\alpha^{n_a} - 1) \\ 0 \\ . \\ 0 \end{bmatrix} \tag{3.9}$$

Or

$$Mw(\alpha) = L(\alpha) \tag{3.10}$$

The control limit should be taken into account in the stabilizer design to avoid servo system saturation or equipment damage. If u_{min} and u_{max} are the lower and upper limits respectively, the optimal solution of the pole-shifting factor α should also satisfy the control constraint. Once the pole-shifting factor α is found from the performance index optimization, the control signal can be calculated from Eqns. (3.9 and 3.4).

3.2.3 Taylor Series Expansion of Control Signal u(t) in Terms of α_t

It is obvious from Eqns. (3.4 and 3.10) that the control at time t, u(t) is a function of the pole shifting factor α at that time. From now on the notation u(t, α_t) is used to indicate the relation. For this algorithm, u(t, α_t) can be expressed in a Taylor series in terms of the factor α_t. The general Taylor series expansion of u(t, α_t) at any point α_0 is

$$u(t, \alpha_t) = u(t, \alpha_0) + \sum_{j=1}^{\infty} \frac{1}{j!} \left[\frac{(\partial^{(j)} u(t, \alpha_t))}{(\partial {\alpha_t}^{(j)})} \right]_{\alpha_t = \alpha_0} (\alpha_t - \alpha_0)^j \tag{3.11}$$

From Eqn. (3.4),

$$F(z^{-1})u(t, \alpha_t) = -G(z^{-1})y(t) \tag{3.12}$$

Expanding Eqn. (3.12) and using Eqn. (3.10), control, u(t, α_t), can be expressed as

$$u(t, \alpha_t) = X^T(t)w(\alpha_t) = X^T(t)M^{-1}L(\alpha_t) \tag{3.13}$$

Where

$$X(t) = [-u(t-1) \ldots . -u(t-n_f) - y(t) - y(t-1) \ldots . -y(t-n_g)]^T \tag{3.14}$$

is the measurement variable vector.

The *j*th order differential of u(t, α_t) with respect to the pole shifting factor α_t becomes

$$\left[\frac{\partial^{(k)} u(t, \alpha_t)}{\partial {\alpha_t}^{(k)}} \right]_{\alpha_t = \alpha_0} = X^T M^{-1} \left[\frac{\partial^{(k)} L(\alpha_t)}{\partial {\alpha_t}^{(k)}} \right]_{\alpha_t = \alpha_0}$$

$$= X^T(t)M^{-1}L^{(k)}(\alpha_0) \quad (3.15)$$

From Eqn. (3.10) and for simplicity, letting α_0=0, $L^{(k)}(0)$ becomes

$$L^{(k)}(0) = [0 \dots 0\ j!\,a_j\ 0 \dots 0]^T\ k \le n_a \quad (3.16)$$

where $j!a_j$ is the jth term, and

$$L^{(k)}(0) = [0 \dots .0]^T\ k > n_a \quad (3.17)$$

Defining the *j*th order sensitivity constant, s_k, as

$$s_k = \frac{1}{j!}\left[\frac{\partial^{(j)}u(t,\alpha_t)}{\partial {\alpha_t}^{(j)}}\right]_{\alpha_t=\alpha_0}$$

$$= \frac{1}{j!}\ X^T(t)M^{-1}L^{(k)}(0)$$

$$= p_k a_k\ k\ \le\ n_a \quad (3.18)$$

where p_k is the *k*th term of the row vector $X^T(t)M^{-1}$.

Eqn. (3.11) can be written in a simple form as

$$u(t,\alpha_t) = u(t,0) + \sum_{j=1}^{n_a} s_k {\alpha_t}^j \quad (3.19)$$

3.2.4 System Output Prediction, ŷ (t+1)

At time t, a predicted value ŷ(t+1) for the system output y(t+1) at time (t+1) can be got if it is assumed that the control u(t, α_t) at time t is known.

The explicit form of ŷ (t+1) is

$$\hat{y}(t+1) = X^T(t)\beta + b_1 u(t,\alpha_t) + e(t+1) \quad (3.20)$$

Where

$$\beta = [-b_2 - b_3 \dots - b_{n_b}\ a_1\ a_2 \dots a_{n_a}]^T \quad (3.21)$$

is an identified parameter vector.

By substituting Eqn. (3.19) into Eqn. (3.20), the system output prediction value ŷ (t+1) can be expressed as a function of the pole shifting factor α_t, as

$$\hat{y}(t+1) = X^T(t)\beta + b_1\left[u(t,0) + \sum_{k=1}^{n_a} s_k {\alpha_t}^k\right] +\ e(t+1) \quad (3.22)$$

3.2.5 Performance Index and Constraints

Taking the idea of minimum variance control, the performance index is chosen as:

$$\min_{\alpha_t} J(t+1,\alpha_t) = E[\hat{y}(t+1) - \hat{y}_r(t+1)]^2 \quad (3.23)$$

where $y_r(t+1)$is the system output reference.

Substituting Eqn. (3.22) into Eqn. (3.23) and considering the independence between the white noise and other variables, the minimization of J(t+1, α_t) is equivalent to the minimization problem of a quadratic function $\hat{J}$(t+1, α_t)

$$\min_{\alpha_t} \hat{J}(t+1, \alpha_t) = [X^T(t)\beta + b_1\left[u(t,0) + \sum_{k=1}^{n_a} s_k {\alpha_t}^k\right] - y_r(t+1)]^2 \quad (3.24)$$

In order to get a practical stabilizer, when minimizing $\hat{J}$(t+1, α_t) with respect to the pole-shifting factor α_t, it should be noted that α_t will be subject to the following constraints.

1. The controller must keep the closed-loop system stable. It implies that all roots of the closed-loop characteristic polynomial A(α_t z^{-1}) must lie within the unit circle in the z-domain. Supposing λ is the absolute value of the largest characteristic root of A(z^{-1}), then α_t λ is the absolute value of the largest characteristic root of A(α_t z^{-1}). To ensure the stability of the closed-loop system, α_t ought to satisfy the following inequality (stability constraint):

   $$\frac{-1}{\lambda} < \alpha_t < \frac{1}{\lambda} \quad (3.25)$$

2. The control limit should be taken into account in the controller design.. If u_{min} and u_{max} are the lower and upper limits respectively, the optimal solution of the pole-shifting factor α_t should also satisfy the following inequality (control constraint):

   $$u_{min} \leq u(t,0) + \sum_{k=1}^{n_a} s_k {\alpha_t}^k \leq u_{max} \quad (3.26)$$

Eqns. (3.24 - 3.26) constitute the self-optimizing pole shifting control algorithm.

3.2.6 Properties of the PS-Algorithm

3.2.6.1 Pole-Shift factor, α

The system will be more stable if the closed-loop poles are nearer to the center of the unit circle in the z-plane. The pole shift factor α will help achieve this goal.

In the proposed algorithm, the varying range of the pole-shifting factor α is $\left(\frac{-1}{\lambda}, \frac{1}{\lambda}\right)$. For different conditions, it acts in the following way:

1. If λ > 1, the open-loop system is unstable. PS control strategy first behaves as a PA controller, places the largest closed-loop poles within the unit circle to assure closed-loop stability and then optimizes its performance.

2. If λ < 1, the open-loop system is stable. The range of $\left(\frac{-1}{\lambda}, \frac{1}{\lambda}\right)$ is larger than (0, 1). Thus, it provides a more feasible area for performance optimization and it will result in a better performance.

3.2.6.2 Optimization function

The proposed control algorithm turns out to be an optimization problem with inequality constraints $\min_{\alpha_t} \hat{J}(t+1, \alpha_t) = E[X^T(t)\beta - y_r(t+1) + b_1 u(t, \alpha_t)]^2$. Any optimization method (such as modified Lagrange multiplier method, Kuhn-Tucker method etc) can be applied to this problem. But for a real-time adaptive control algorithm, a simpler method will be the better choice. Handling the control constraints in the optimization of the performance index $\hat{J}(t+1, \alpha_t)$ offers a difficult problem. For this, the following strategy is used.

Consider that the open-loop system is stable and the optimal value of the pole-shifting factor α_{opt} is obtained by optimizing $\hat{J}(t+1, \alpha_t)$ with the stability constraint only (obviously $\frac{-1}{\lambda} < \alpha_{opt} < 1.0 \; or \; 1.0 < \alpha_{opt} < \frac{1}{\lambda}$). If the control signal $u(t, \alpha_t)$ obtained by optimizing $\hat{J}(t+1, \alpha_t)$ with the stability constraint outside the control limits, then

i. the control limit (u_{min} or u_{max}) is the only control signal which optimizes $\hat{J}(t+1, \alpha_t)$ under the control constraint and

ii. there must exist a pole shifting factor α_t which satisfies the stability constraint for this control signal

The control $u(t, \alpha_t)$ is a continuous, single-valued function in terms of pole-shifting factor α_t. With the mean-value theorem of continuous functions, α_{limit} must satisfy $\frac{-1}{\lambda} < \alpha_{opt} < \alpha_{limit} < 1.0 \; or \; 1.0 < \alpha_{limit} < \alpha_{opt} < \frac{1}{\lambda}$ which indicates that α_{limit} is within the range of $\left(\frac{-1}{\lambda}, \frac{1}{\lambda}\right)$ and thus assures the closed-loop stability of the controlled system.

The above strategy simplifies the optimization scheme and reduces the optimization time. It indicates that if the control signal has to reach its limit, the best control signal for this situation is the control limit itself and the closed-loop system will not lose its stability under this condition.

3.2.6.3 Steady-state characteristics

When the system is operating under steady state conditions, it is desirable for the controlled system to have enough stability margins to endure disturbances. The proposed algorithm easily

fulfills this steady state stability margin setting through the steady state pole shifting factor. With this property, the proposed algorithm eliminates the necessity of choosing the closed-loop pole locations suitable for both dynamic and steady state conditions and still possesses the quality of robustness of a PA algorithm.

3.2.6.4 Dynamic characteristics

Under dynamic conditions caused by disturbances or changes in the operating conditions, the system output is required to be settled to its normal value or track the reference as quickly as possible. This requires a fast response from the controller. Once it senses a change in the operating conditions, the proposed control algorithm will shift the poles to appropriate locations within the unit circle immediately to yield an optimal control signal corresponding to the change.

3.2.6.5 Closed-loop stability

The stability of the closed-loop system has the highest priority in this algorithm. This algorithm assures the stability of the closed-loop system, as long as the identified parameters converge to their true values. If the identified parameters do not converge to their true values, the stability constraint in Eqn. (3.10) can be modified so that the controller acts in a more cautious manner and satisfies a certain security coefficient,

$$-\frac{1}{\lambda}(1-\sigma) < \alpha < \frac{1}{\lambda}(1-\sigma) \tag{3.27}$$

where σ is called the security coefficient and its value can be set at any desired value between 0% and 100%.

3.2.7 Third Order ARMA Model - Control Strategy

The generating unit is represented by a third order discrete ARMA [38] model of the form:

$$A(z^{-1})y(t) = B(z^{-1})u(t) + e(t) \tag{3.28}$$

where $A(z^{-1})$ and $B(z^{-1})$ are polynomials in the backward shift operator z^{-1} and are defined as

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} \tag{3.29}$$

$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} \tag{3.30}$$

and $y(t), u(t)$ and $e(t)$ are the system output, system input and white noise respectively.

Rewriting Eqn. (3.28) in the form suitable for identification:

$$y(t) = \theta^T \Psi(t) + e(t) \tag{3.31}$$

where

$$\theta(t) = [a_1 \;\; a_2 \;\; a_3 \;\; b_1 \;\; b_2 \;\; b_3]^T \tag{3.32}$$

is the parameter vector and

$$\Psi(t) = [-y(t-1) - y(t-2) - y(t-3)\ u(t-1)\ u(t-2)\ u(t-3)]^T \quad (3.33)$$

is the measurement variable vector.

Once the system model parameters (Eqn. (3.32)) are identified by the FLN identifier (as described in Section 2.6.3), the control signal can be calculated based on the ARMA model (Eqn. (3.28)) . Assuming that the feedback loop has the form,

$$\frac{u(t)}{y(t)} = -\frac{G(z^{-1})}{F(z^{-1})} \quad (3.34)$$

where

$$F(z^{-1}) = 1 + f_1 z^{-1} + f_2 z^{-2} \quad (3.35)$$

$$G(z^{-1}) = g_0 + g_1 z^{-1} + g_2 z^{-2} \quad (3.36)$$

The characteristic polynomial of the closed-loop system is assumed to have the same form as that of the open-loop system in the PS control algorithm. Also, in the closed-loop, the open-loop poles are shifted radially towards the center of the unit circle in the z-plane by a shifting factor α. This gives the following equation:

$$A(z^{-1})F(z^{-1}) + B(z^{-1})G(z^{-1}) = T(z^{-1}) = A(\alpha z^{-1}) \quad (3.37)$$

Expanding both sides of Eqn. (3.37) and comparing the coefficients gives:

$$\begin{bmatrix} 1 & 0 & b_1 & 0 & 0 \\ a_1 & 1 & b_2 & b_1 & 0 \\ a_2 & a_1 & b_3 & b_2 & b_1 \\ a_3 & a_2 & 0 & b_3 & b_2 \\ 0 & a_3 & 0 & 0 & b_3 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ g_0 \\ g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} a_1(\alpha - 1) \\ a_2(\alpha^2 - 1) \\ a_3(\alpha^3 - 1) \\ 0 \\ 0 \end{bmatrix} \quad (3.38)$$

or in matrix form,

$$M.w(\alpha) = L(\alpha) \quad (3.39)$$

If the pole-shift factor α is fixed, the PS control algorithm becomes a special case of Pole Assignment (PA) control algorithm. The rule determining the pole-shifting factor is very important. α is modified on-line according to the operating conditions of the controlled system for optimum performance.

From Eqns. (3.34) and (3.39), the control signal *u(t)* can be expressed as a function of the pole-shifting factor α as:

$$u(t) = X^T(t)w(\alpha) = X^T(t)M^{-1}L(\alpha) \quad (3.40)$$

where

$$X(T) = \left[-u(t-1) - u(t-2) - y(t) - y(t-1) - y(t-2)\right]^T \quad (3.41)$$

u(t) can be expanded into a Taylor series with a simple form [39]:

$$u(t) = u(t,0) + \sum_{i=1}^{3} p_i a_i \alpha^i \quad (3.42)$$

where p_i is the ith term of the row vector $X^T M^{-1}$ and

$$u(t,0) = X^T(t) M^{-1} L(0) \quad (3.43)$$

At time t, the predicted value $\hat{y}(t+1)$ for the system output y(t+1) at time (t+1) can be found if it is assumed that the control u(t) at time t is known. The explicit form of $\hat{y}(t+1)$ is:

$$\hat{y}(t+1) = X^T(t)\beta + b_1 u(t) + e(t+1) \quad (3.44)$$

where

$$\beta = [-b_2 \ -b_3 \ a_1 \ a_2 \ a_3]^T \quad (3.45)$$

is an identified parameter vector.

3.2.8 Linear Discrete System Example

The proposed control algorithm is first applied to a known linear discrete system:

$$y(t) - 0.95\, y(t-1) + 0.09\, y(t-2) + 1.377\, y(t-3) = 0.5\, u(t-1) + 0.1\, u(t-2) + 0.25\, u(t-3) + e(t) \quad (3.46)$$

The error term $e(t)$ accounts for a 2% white noise added to the system. The open loop poles of the plant are at -0.85 and $0.9 \pm j\,0.9$. The control signal is calculated using Eqn. (3.10) in each sampling period such that the cost function given in Eqn. (3.23) is minimized.

The system response to the square wave reference waveform of magnitude ±1 is plotted in Fig 3.2. The reference signal is tracked effectively by the controlled plant.

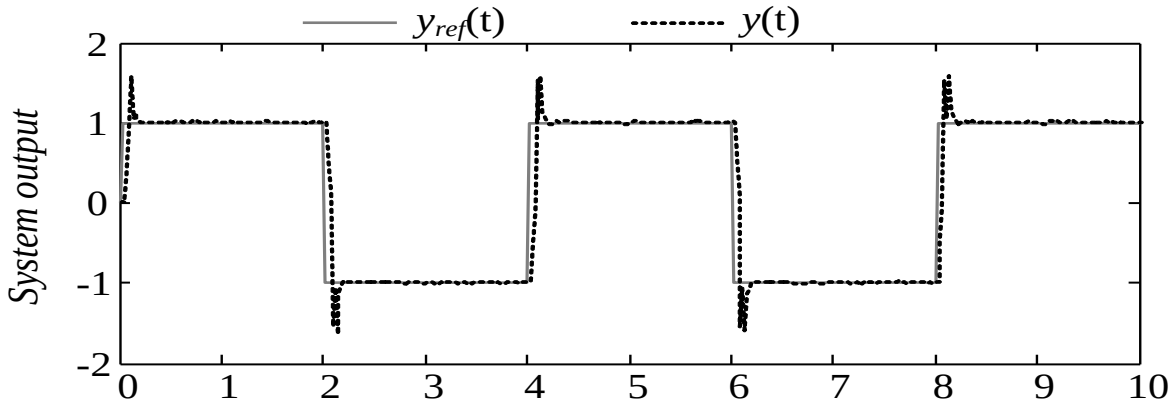
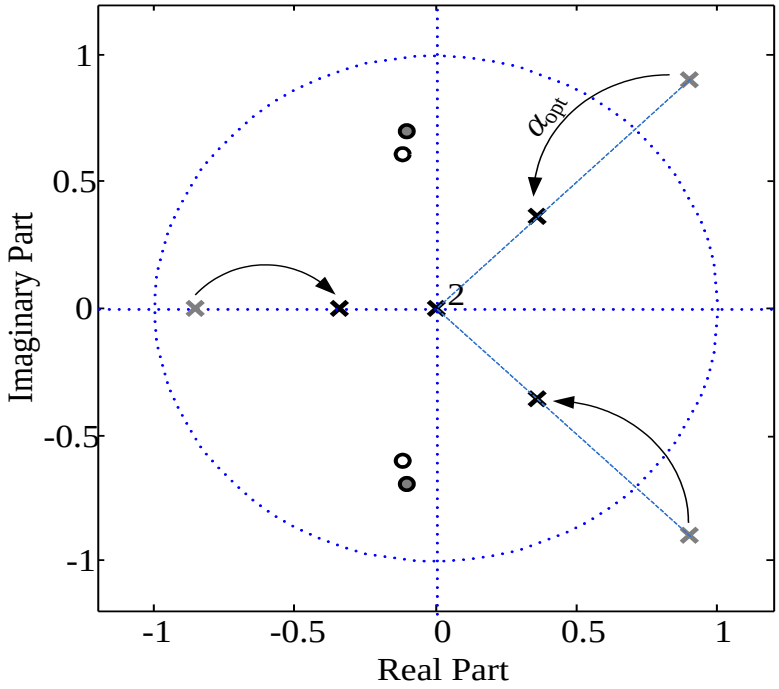
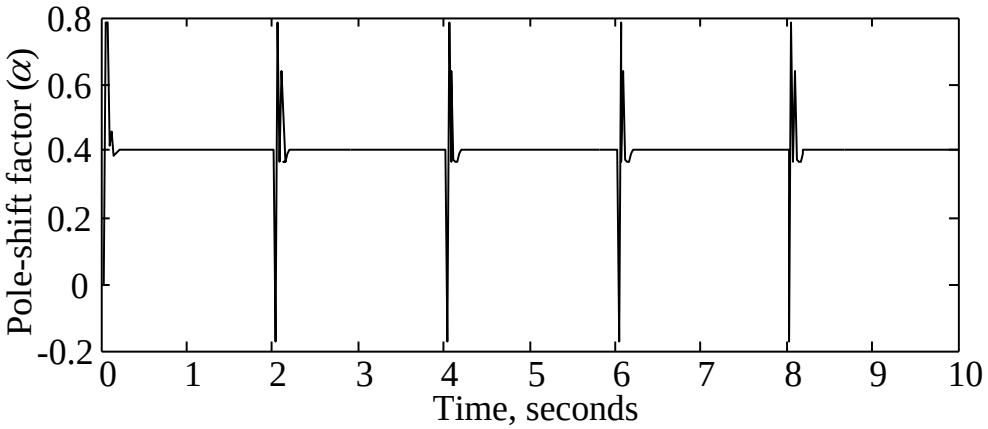


Fig 3.2 Linear discrete time system response to step changes in reference

The open and closed loop poles/zeros of the system are plotted in Fig 3.3. The closed loop poles and zeros are plotted in black color and the open loop poles and zeros are plotted in gray color. Fig 3.3 shows that it is an unstable plant as two of the open loop poles are outside the unit circle. The system is stabilized by the pole-shift controller by moving the unstable poles inside the unit circle. The controller also minimizes the next time-step output prediction error. The optimal pole-shift factor is plotted in Fig 3.3b.



(a) Pole-zero plot for open-loop (*gray*) and closed-loop system (*black*)



(b) Pole-shift factor α

Fig 3.3 Adaptive optimal pole-shifting process for the discrete system

3.3 Summary

PS-Control algorithm, which combines the advantages of MV algorithm and PA algorithm, is presented in Section 3.2. The FLN-Identifier discussed in Chapter 2 is combined with the linear feedback control system (PS-Controller) for adaptive control of power systems using APSS. In Section 3.2.7, a linear discrete system example is given to show the effectiveness of the proposed control algorithm. Advantages of using FLN identification for APSS application is the main point of attention in this thesis and the simulation studies and the results are discussed in Chapter 4.

CHAPTER 4

APSS USING FLN-IDENTIFIER AND POLE-SHIFT CONTROLLER

4.1 Introduction

In this dissertation, a FLN identifier and PS Control technique have been used respectively for identifying the system characteristics and for the control computations.

The nonlinear functional mapping properties of neural networks are central to their use in identification of nonlinear systems. The approximation capabilities of neural networks are very useful in the modeling of power systems [40]. In FLN, the hidden layer is removed without giving up non-linearity by providing the input layer with expanded inputs that are constructed as the functions of original attributes. This overcomes the drawbacks associated with the multilayer network employing back propagation algorithm. Also, the multilayer perceptron cannot be represented in the form of a function whereas, the FLN networks have a functional form. This facilitates their use in PS control system. Some power system applications such as state estimation uses the FLN [41].

At each sampling instant, the input and output of the FLN are linearized using Taylor Series expansion to find the ARMA parameters on-line. The learning algorithm further adjusts the weights so as to minimize the mean-squared error between the outputs of the plant, i.e desired output and the identifier. This process is repeated every sampling instant making the training on-line, which in turn results in an adaptive approach to identify the power system.

4.2 On-Line Identification Using FLN Identifier

At any given time, power system operation can be categorized to be in one of four states:

- Normal state in which all controlled quantities are within acceptable limits.
- Alert state, in which some variables are out of limits, but not to the point that the system stability is threatened.
- Emergency state, in which system stability is threatened, requiring immediate action to keep the system viable.

- Restorative state, in which stabilized subsystems are reconnected in a systematic way to maintain system integrity.

Fixed-parameter controllers designed for a specific operating point cannot maintain the same quality of performance at other operating points. This difficulty can be overcome using adaptive control techniques, which will improve the dynamic performance of the power system.

The objectives in this section are:

- To verify the performance of APSS consisting of FLN-Identifier and PS-Control.
- To make a comparative analysis of the CPSS and APSS response.

4.2.1 Proposed FLN Structure

The generating unit is represented by a third order discrete ARMA model of the form:

$$A(z^{-1})y(t) = B(z^{-1})u(t) + d(t) \quad (4.1)$$

where $A(z^{-1})$ and $B(z^{-1})$ are polynomials and are defined as

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} \quad (4.2)$$

$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} \quad (4.3)$$

and variables y(t), u(t) and d(t) are the system output, system input and white noise respectively. Rewriting Eqn. (6.1) in the form suitable for identification:

$$y(t) = \theta^T(t)\psi(t) + d(t) \quad (4.4)$$

where

$$\theta(t) = [a_1\ a_2\ a_3\ b_1\ b_2\ b_3]^T \quad (4.5)$$

is the parameter vector (regression coefficients) and

$$\psi(t) = [-y(t-1) - y(t-2) - y(t-3)u(t-1)u(t-2)u(t-3)]^T \quad (4.6)$$

is the measurement variable vector.

4.2.2 System Model

A synchronous generator connected to a constant voltage bus through two transmission lines (Fig 4.1) is used for the APSS studies.

4.2.3 Selection of the order of ARMA Model

The generating unit is identified by a third order discrete ARMA model. The system dynamics determine the selection of the order of the model.

A third-order ARMA model is a sufficient representation for modeling the low frequency oscillations. The dominant low-frequency inputs to the PSS represented by a pair of complex

poles and a single pole representing the system response without oscillation can be represented by a third order system.

Since a second order system can only represent either oscillatory or non-oscillatory part of the system response, this system will not be a practical option. A fourth order system can be identified either as two pairs of complex poles, or one pair of complex poles and two real poles or four real poles. But, two pairs of complex poles can only represent the oscillatory system response and four real poles can represent only the non-oscillatory system response. If we choose, two pairs of complex poles and two real poles, the two real poles seem unnecessary to represent the non-oscillatory part. So, the fourth order system is usually not a good choice.

Since the idea of APSS is to give good estimate of the control signal in the shortest interval, the third-order system is preferred in the studies as it needs minimum number of on-line parameter estimates (Eqn. 4.5).

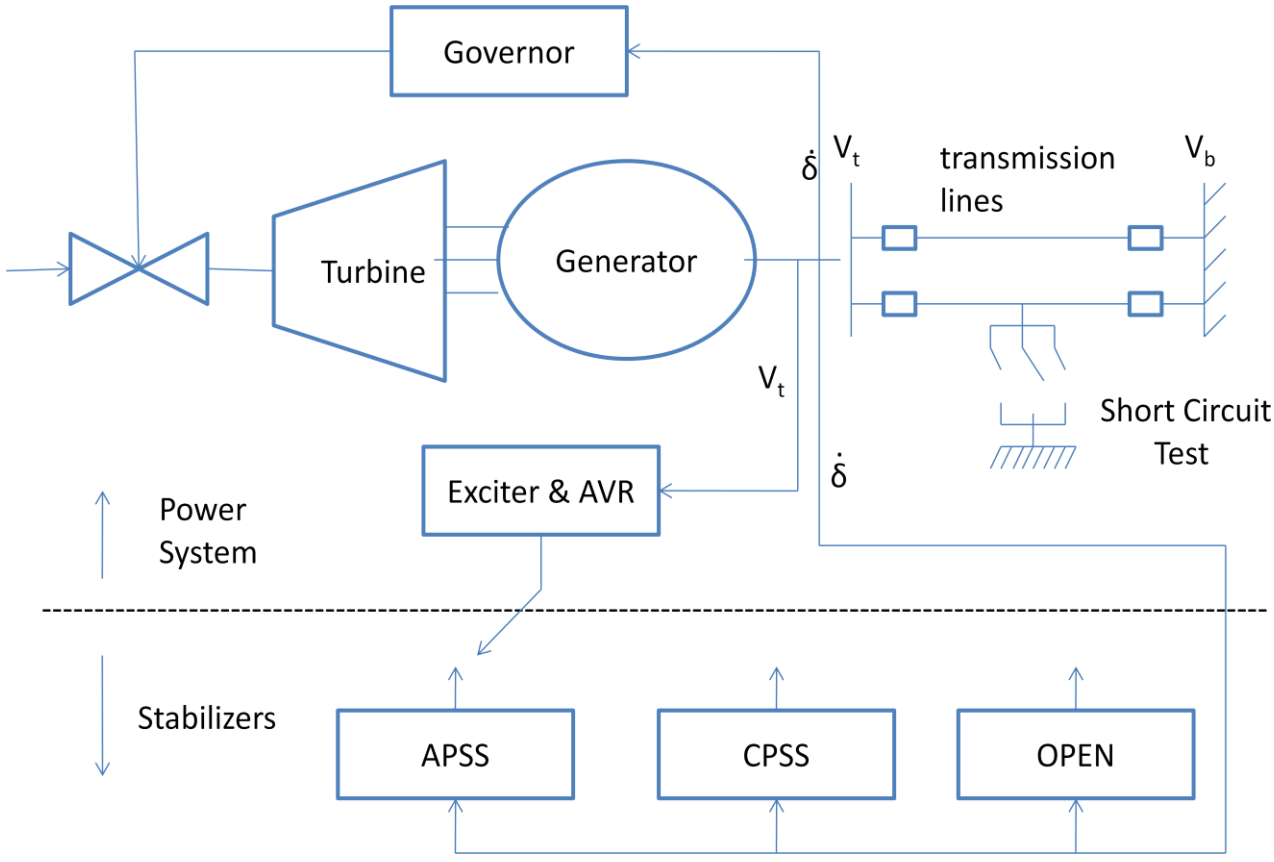


Fig 4.1 System model used in APSS Studies

4.2.4 Control Strategy

The process of obtaining ARMA parameters using FLN identifier is an important point; because it is possible to apply linear analysis control methods such as PS control technique to obtain the control signal. Once the system model parameters (Eqn. (4.5)) are identified using the FLN identifier, the control signal can be calculated based on the ARMA model defined by Eqn. (4.1). The PS equations using a third-order model are given below:

The feedback loop is assumed to have the form,

$$\frac{u(t)}{y(t)} = -\frac{G(z^{-1})}{F(z^{-1})} \tag{4.7}$$

Where

$$F(z^{-1}) = 1 + f_1 z^{-1} + f_2 z^{-2} \tag{4.8}$$

$$G(z^{-1}) = g_0 + g_1 z^{-1} + g_2 z^{-2} \tag{4.9}$$

The characteristic polynomial of the closed-loop system is assumed to have the same form as that of the open-loop system in the PS control algorithm. Also, in the closed-loop, the open-loop poles are shifted radially towards the center of the unit circle in the z-plane by a shifting factor α. This implies the following equation:

$$A(z^{-1})F(z^{-1}) + B(z^{-1})G(z^{-1}) = T(z^{-1}) = A(\alpha z^{-1}) \tag{4.10}$$

Expanding both sides of Eqn. (4.10) and comparing the coefficients gives:

$$\begin{bmatrix} 1 & 0 & b_1 & 0 & 0 \\ a_1 & 1 & b_2 & b_1 & 0 \\ a_2 & a_1 & b_3 & b_2 & b_1 \\ a_3 & a_2 & 0 & b_3 & b_2 \\ 0 & a_3 & 0 & 0 & b_3 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ g_0 \\ g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} a_1(\alpha - 1) \\ a_2(\alpha^2 - 1) \\ a_3(\alpha^3 - 1) \\ 0 \\ 0 \end{bmatrix} \tag{4.11}$$

Or in matrix form

$$M.w(\alpha) = L(\alpha) \tag{4.12}$$

PS control algorithm becomes a special case of the PA control algorithm if the pole-shift factor α is fixed. The rule determining the pole-shifting factor is very important. For optimum performance α is modified on-line according to the operating conditions of the controlled system.

From Eqns. (4.7) and (4.12), the control *u(t)* can be expressed as a function of the pole-shifting factor α as:

$$u(t) = X^T(t)w(\alpha) = X^T(t)M^{-1}L(\alpha) \tag{4.13}$$

Where

$$X(t) = \left[-u(t-1) - u(t-2) - y(t) - y(t-1) - y(t-2)\right]^T \quad (4.14)$$

u(t) can be expanded into a Taylor series:

$$u(t) = u(t,0) + \sum_{j=1}^{3} r_j a_j \alpha^j \quad (4.15)$$

where r_j is the j^{th} term of the row vector $X^T(t)M^{-1}$ and

$$u(t,0) = X^T(t)M^{-1}L(0) \quad (4.16)$$

The predicted value $\hat{y}(t+1)$ for the system output $y(t+1)$ at time $(t+1)$ at time t can be found if it is assumed that the control *u(t)* at time *t* is known. The explicit form of $\hat{y}(t+1)$ is:

$$\hat{y}(t+1) = X^T\beta + b_1 u(t) + e(t+1) \quad (4.17)$$

where

$$\beta = [-b_2 \;\; -b_3 \;\; a_1 \;\; a_2 \;\; a_3]^T \quad (4.18)$$

is an identified parameter vector.

Using the idea of MV control, the performance index is chosen as:

$$\min_\alpha J(t+1,\alpha) = E[\hat{y}(t+1) - y_r(t+1)]^2 \quad (4.19)$$

where $\hat{y}(t+1)$ is the system output reference.

From Eqns. (4.15) and (4.17), considering the independence between the white noise and other variables, the minimization of $J(t+1,\alpha)$ is equivalent to the minimization problem of a quadratic function $\hat{J}(t+1,\alpha)$:

$$\min_\alpha \hat{J}(t+1,\alpha) = \min_\alpha \left\{X^T(t)\beta + b_1\left[u(t,0) + \sum_{j=1}^{3} r_j a_j \alpha^j\right] - y_r(t+1)\right\}^2 \quad (4.20)$$

4.2.4.1 Constraints

When minimizing $\hat{J}(t+1,\alpha)$ with respect to α, α will be subject to the following constraints:

1. The stabilizer must keep the closed-loop system stable. So, all roots of the closed-loop characteristic polynomial $A(\alpha z^{-1})$ must lie within the unit circle in the z-domain. If λ is the absolute value of the largest characteristic root of $A(z^{-1})$, then α . λ is the absolute value of the largest characteristic root of $A(\alpha z^{-1})$. To ensure the stability of the closed-loop system, α ought to satisfy the following stability constraint:

   $$\frac{1}{\lambda} < \alpha < \frac{1}{\lambda} \quad (4.21)$$

2. In the APSS design, to avoid servo system saturation or equipment damage, the control limit should be taken into account. If u_{min} and u_{max} are the lower and upper limits respectively, the optimal solution of the pole-shifting factor α should also satisfy the following control constraint:

$$u_{min} \leq u(t,0) + \sum_{j=1}^{3} p_j a_j \alpha^j \leq u_{max} \quad (4.22)$$

Eqns. (4.20), (4.21) and (4.22) constitute the control strategy for the proposed APSS. The control signal can be calculated using Eqn. (4.15) directly, once the pole-shifting factor α is found from the performance index optimization.

4.2.4.2 Optimization

A simpler optimization method is a better option to solve the minimization problem given in Eqn. (4.20). The solution steps are given below:

1. Let

$$q_0 = u(t) \quad \text{From Eq. (4.15)}$$
$$q_1 = \frac{\partial J}{\partial u}$$
$$q_2 = \frac{\partial u}{\partial \alpha} \quad (4.23)$$
$$q_3 = \frac{\partial q_2}{\partial \alpha}$$
$$k_0 = u(t,0) \quad \text{From Eq. (4.15)}$$
$$k_i = p_j a_j, \quad 1 \leq j \leq 3 \quad \text{From Eq. (4.15)}$$

2. Then,

$$q_0 = k_0 + (k_1 + (k_2 + k_3 * \alpha) * \alpha) * \alpha$$
$$q_1 = b_1 * (\beta + b_1 * f_0)$$
$$q_2 = k_1 + (2.0 * k_2 + 3.0 * k_3 * \alpha) * \alpha \quad (4.24)$$
$$q_3 = 2.0 * k_2 + 6.0 * k_3 * \alpha$$

3. Newton's method is used to find α. Note

$$J' = 0 : J' = q_2 q_3 = 0 \quad (4.25)$$

The following equation is iterated until a pre-specified tolerance (10^{-7}) is reached,

$$\alpha_{new} = \alpha_{old} - \frac{\frac{\partial J}{\partial u}}{\frac{\partial J'}{\partial u}} \quad (4.26)$$

or in terms of $q_1 - q_3$,

$$\alpha_{new} = \alpha_{old} - \frac{q_2 q_1 + \eta(\alpha_{old} - \alpha_0)}{b_1^2 q_2^2 + q_1 q_3 + \eta} \quad (4.27)$$

where α_0 is pre-selected (0.7) at start of iteration and η is the gradient term.

This method simplifies the optimization procedure and reduces the optimization time. The control constraints are included in the optimization procedure as follows:

If the control signal u(t) obtained by optimizing $\hat{J}(t+1, \alpha)$ with only the stability constraint, is outside the control limits, then

- the control limit (u_{min} or u_{max}) is the only control signal which optimizes $\hat{J}(t+1, \alpha)$ under the control constraint and
- there must exist a pole-shifting factor α which satisfies the stability constraint for this control signal.

4.3 Simulation Studies on a Single Machine Infinite Bus Power System

To simulate the dynamic behavior of the single-machine infinite bus power system, a non-linear seventh order model is used. A sampling rate of 20 Hz is chosen for the simulations. The response of the FLN-Identifier after training is compared to that of the actual system output for various disturbances under different operating conditions as described in the following sections. The AVR/Exciter and CPSS refer to IEEE Standard 421.5, Type ST1A AVR and Exciter Model and an IEEE Standard 421.5, PSS1A Type CPSS [50] (Appendix B.2 and B.4).

4.3.1 Normal Load

The generator is operating at 0.97 pu, 0.97 pf lag (normal load). A 0.05 pu step increase in torque reference is applied at 1.0 s.

The power angle (δ) variation of the open-loop, with CPSS and with APSS is shown in Fig 4.2. Speed variation is shown in Fig. 4.3. The control signals for APSS and CPSS and terminal voltage response are given in Fig 4.4 and Fig 4.5.

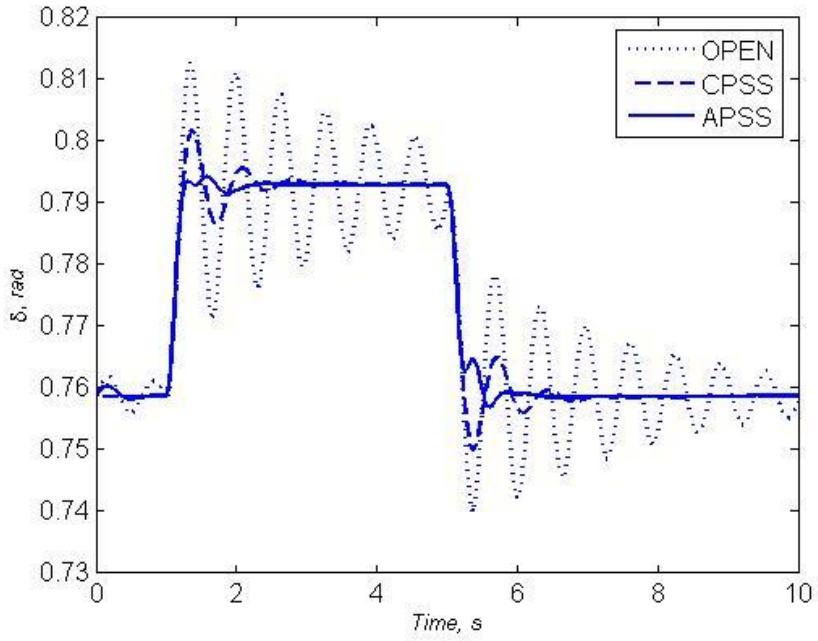


Fig 4.2 Power angle variation to a 0.05 pu step increase in torque and return to initial conditions under normal load with PSS

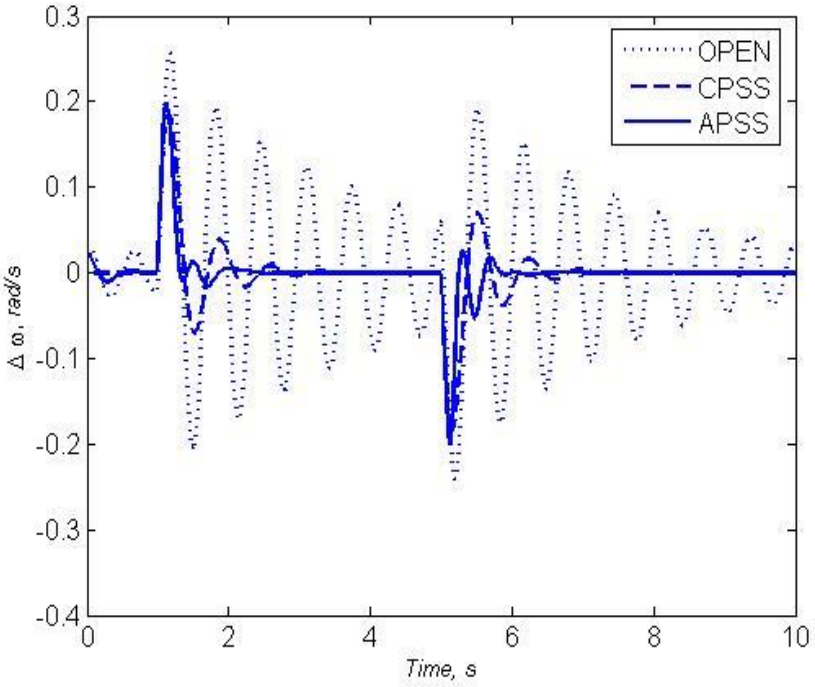


Fig 4.3 Speed variation to a 0.05 pu step increase in torque and return to initial conditions under normal load with PSS

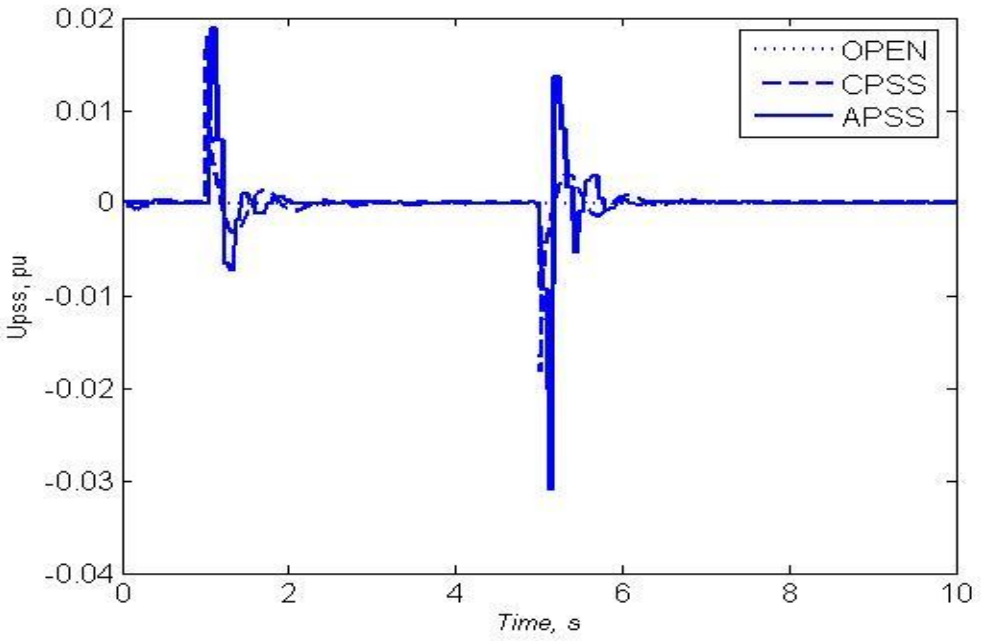


Fig 4.4 Control signals for a 0.05 pu step increase in torque and return to initial conditions under normal load with PSS

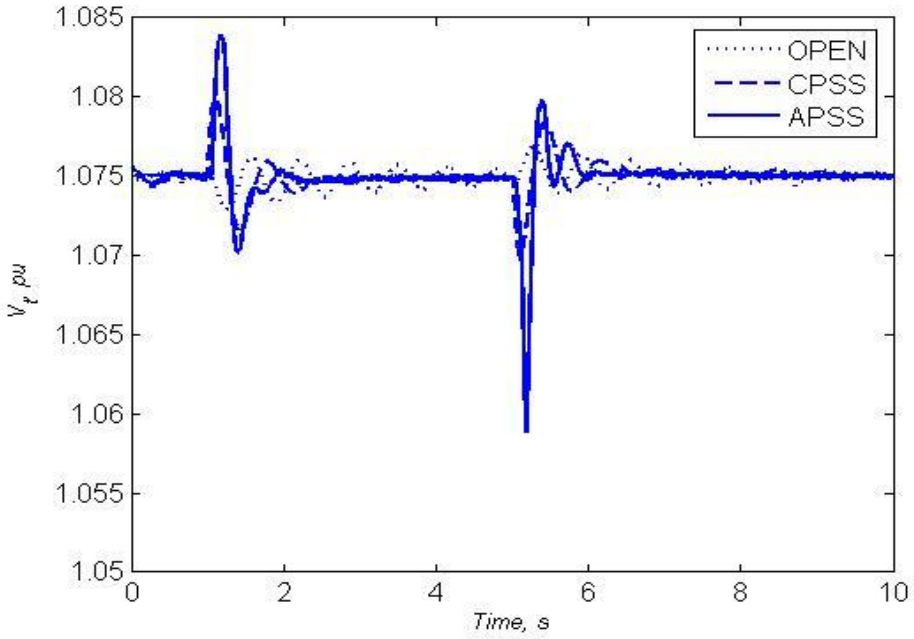


Fig 4.5 Terminal voltage response to a 0.05 pu step increase in torque and return to initial conditions under normal load with PSS

4.3.2 Light Load

The generator is operating at 0.5 pu power at 0.9378 pf lagging. A 0.15 step increase in input torque reference is applied at 1.0 s. The power angle (δ) response of the open-loop, with CPSS and with APSS is shown in Fig 4.6. Speed variation, control signals u(t) for both APSS and CPSS and terminal voltage v(t) response are shown in Fig 4.7, 4.8 and 4.9.

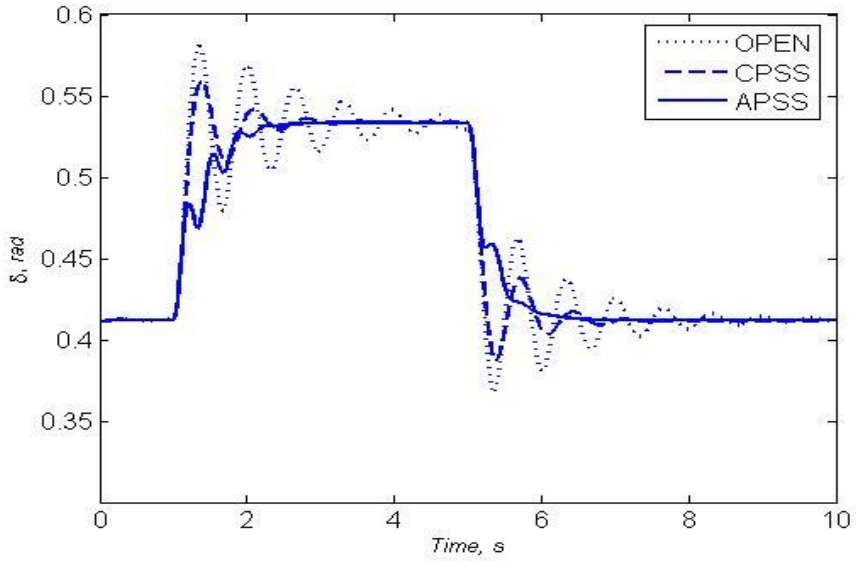


Fig 4.6 Power angle response to a 0.15 pu change in torque and return to initial conditions under light load with PSS

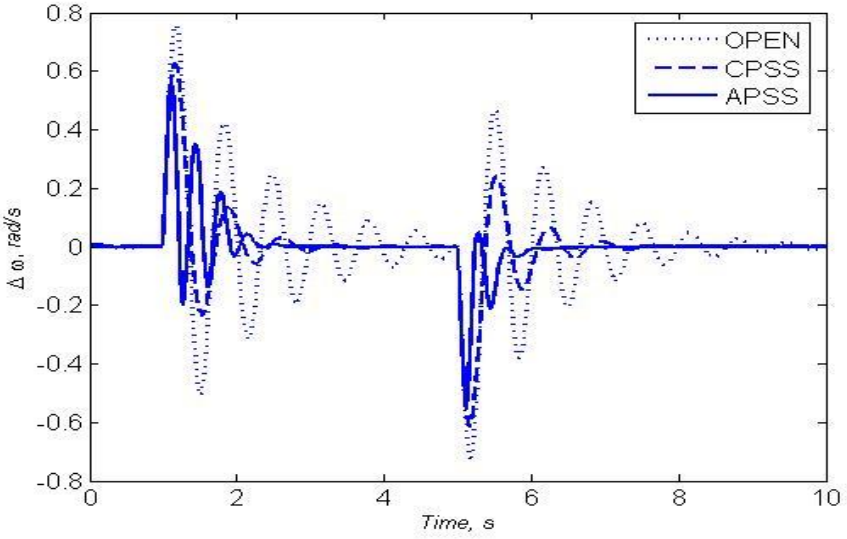


Fig 4.7 Speed variation to a 0.15 pu change in torque and return to initial conditions under light load with PSS

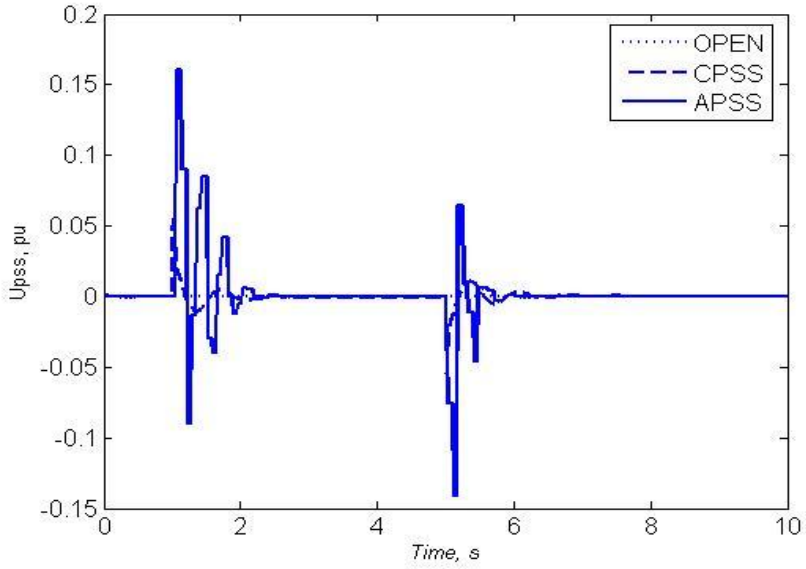


Fig 4.8 Control signals for a 0.15 pu change in torque and return to initial conditions under light load with PSS

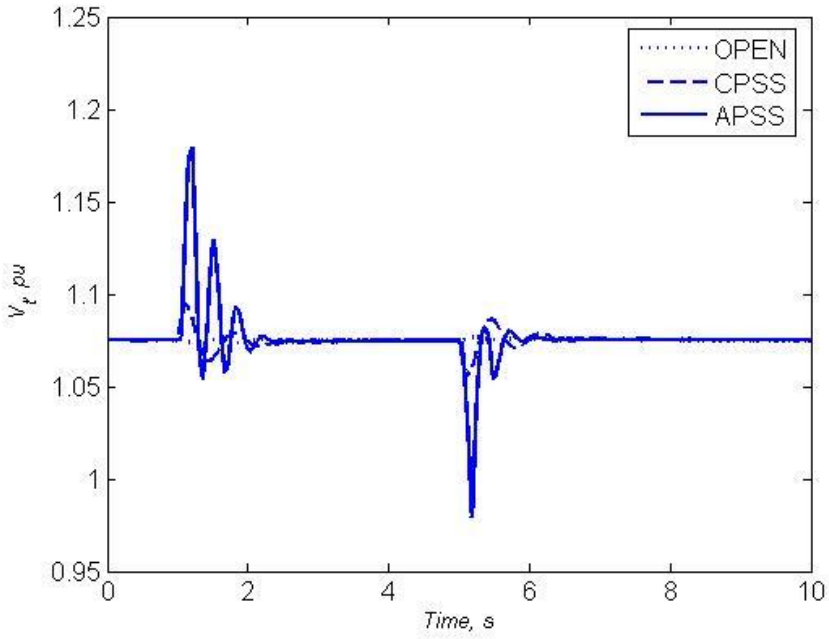


Fig 4.9 Terminal voltage response to a 0.15 pu change in torque and return to initial conditions under light load with PSS

4.3.3 Voltage Reference Change

The generator is operating at 0.6 pu power and 0.95 pf lag and 1.05 pu terminal voltage. A 0.02 pu increase in voltage reference is applied at 1.0 s. The power angle (δ) response is shown in Fig 4.10. The terminal voltage response for open-loop, CPSS and APSS is shown in Fig 4.11. Control signals u(t) for APSS and CPSS are shown in Fig 4.12.

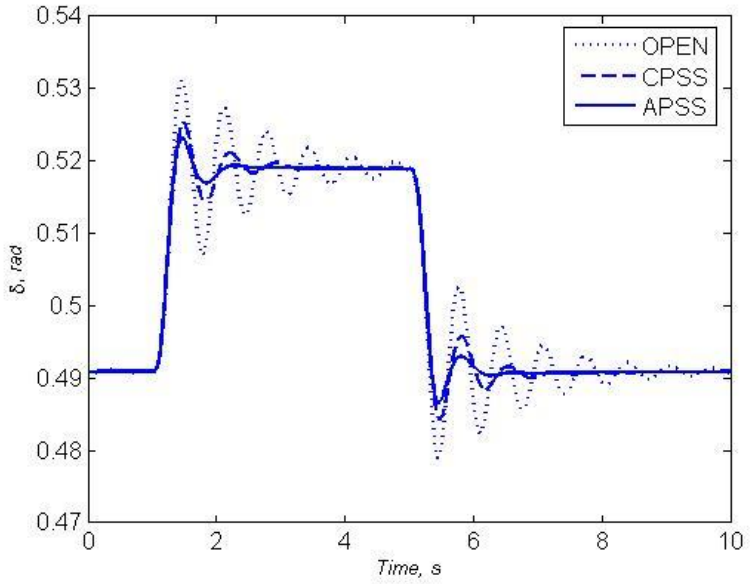


Fig 4.10 Power angle response to a 0.02 pu step increase in voltage and return to initial conditions

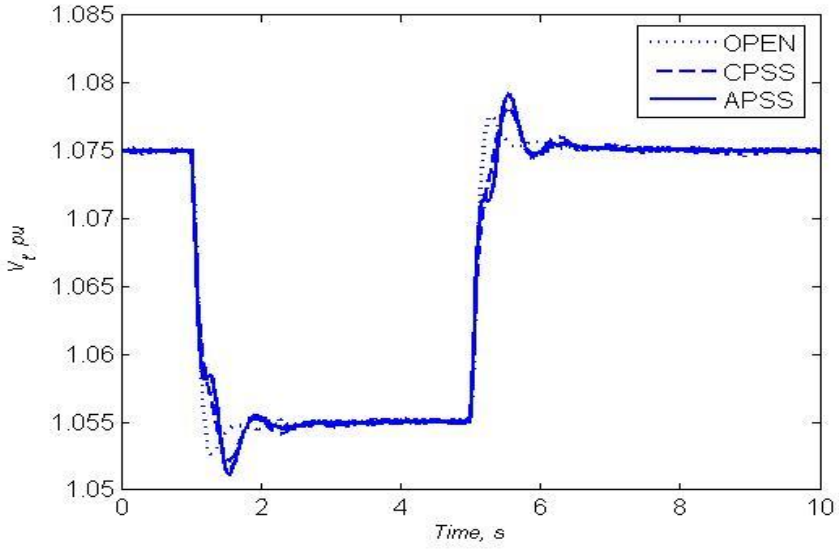


Fig 4.11 V_t response to a 0.02 pu step increase in voltage and return to initial conditions

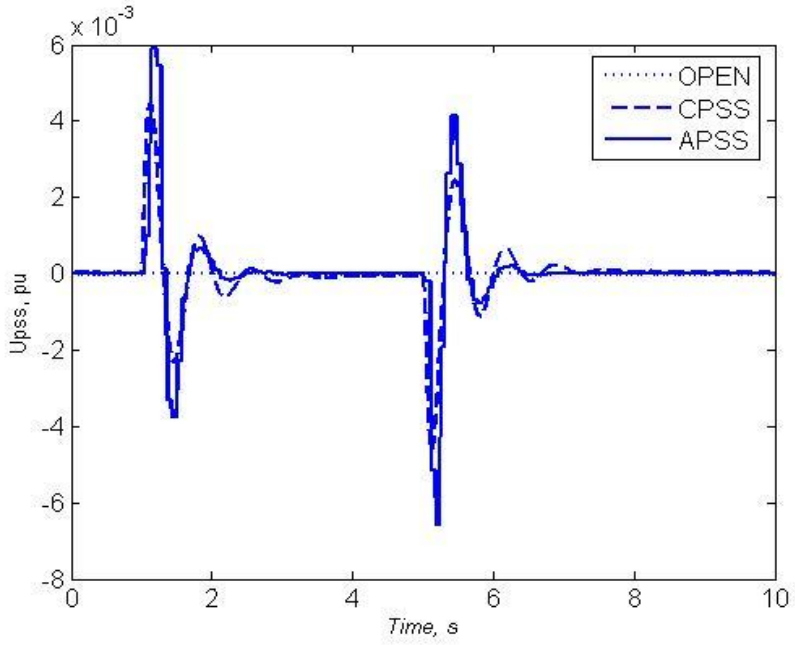


Fig 4.12 Control signals for a 0.02 pu step increase in voltage and return to initial conditions

4.3.4 Leading Power Factor

A 0.2 pu step increase in torque reference was applied at 1 s when the generator is operating at a power of 0.7 pu with 0.96 pf lead. The results are given in Fig 4.13 and 4.14.

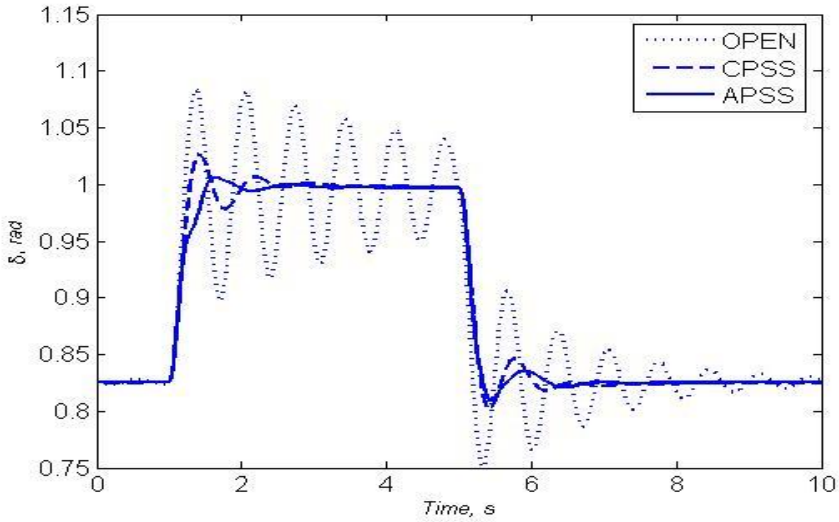


Fig 4.13 Power angle response to a 0.2 pu step increase in torque under leading power factor conditions

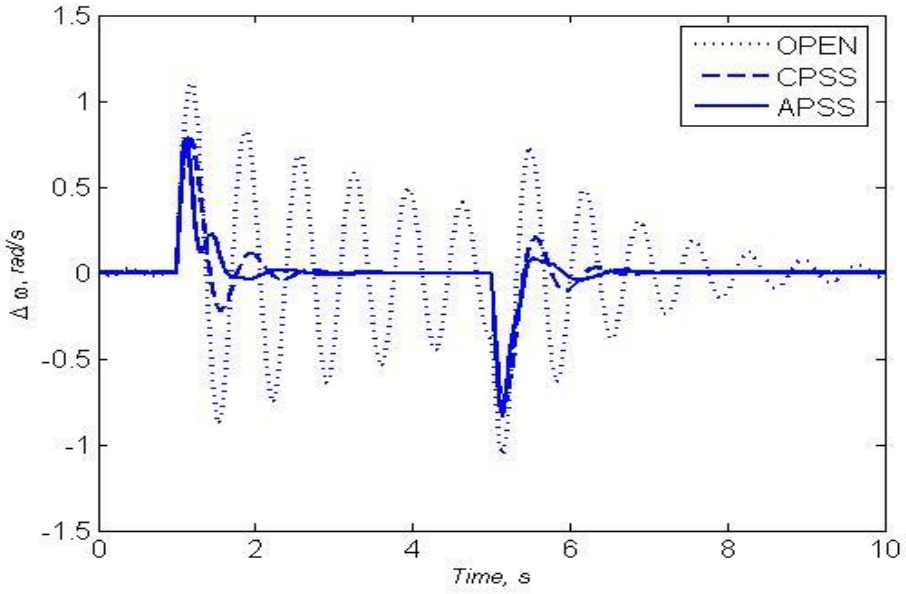


Fig 4.14 Speed variation to a 0.2 pu step increase in torque under leading power factor conditions

4.3.5 Fault Test

The initial operating conditions are 0.5 pu active power delivered to the bus at 0.93 pf lag. A three-phase to ground fault is applied at 1.0 s at the middle of one of the transmission line. Power angle (δ) response is shown in Fig 4.15. Control signals u(t) are shown in Fig 4.16.

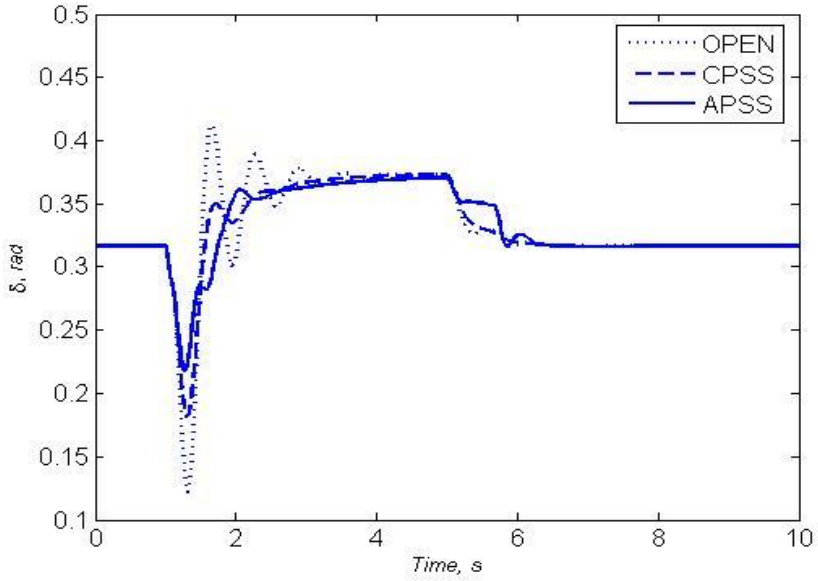


Fig 4.15 Power angle response to a three-phase to ground fault at the middle of one transmission line

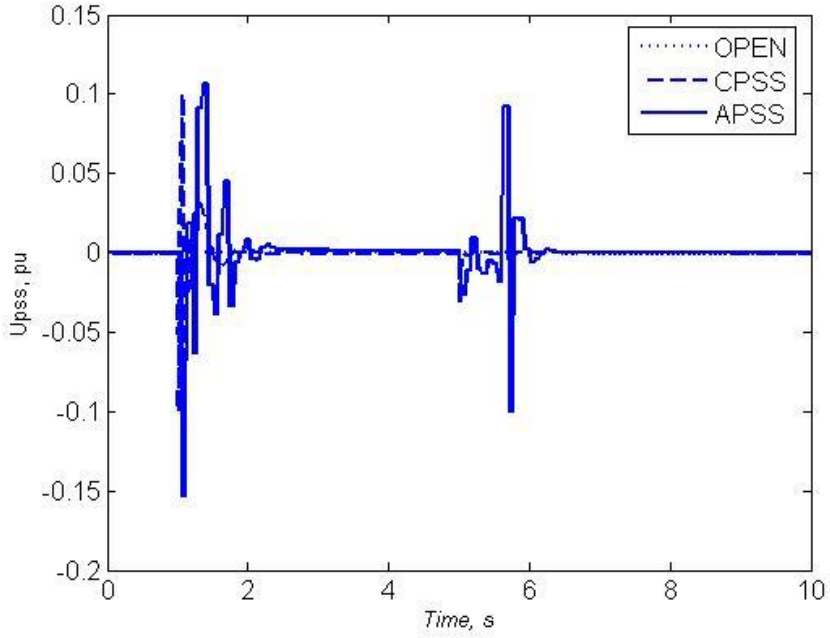


Fig 4.16 Control signals for a three-phase to ground fault at the middle of one transmission line

4.3.6 Single Machine Infinite Bus Power System Performance – Discussion

The results from Figs 4.2 through 4.16 shows that the APSS with FLN identifier has better performance than the CPSS for different operating conditions and disturbances. The APSS provides a better damping in a wide range of operation as compared to the conventional PSS without requiring any parameter tuning.

Advantages of FLN identifier:

- Use of enhanced inputs in the FLN makes it a single layer network model, thus requiring a simpler learning algorithm and faster training.
- As FLNs do not have any hidden layer, the architecture becomes simple. Thus, nonlinear modeling can be accomplished, by means of a linear learning rule. The computational complexity is also reduced.
- It reaches global minima very easily.
- As FLNs involve linear mapping in polynomial space, they can easily map linear and nonlinear terms.
- FLN is truly linear in the parameters. Approximation theory not only says that a sufficient FLN with the correct weights can accurately implement an arbitrary continuous function but also confirms that these parameters can always be learnt at

least in the least squares sense. This second property is an advantage of using the FLN to model nonlinear systems.

4.4 FLN-Identifier and PS-Control – A Multi-Machine Power System Case Study

FLN-identifier and PS controller to damp out multi-mode oscillations in a power system is described in this section. A five machine power system is used in the simulation studies. Behavior of the proposed APSS and its coordination ability with other CPSSs in the system is also explained.

Multi-machine power systems in which the interconnected generating units have different inertia constants and are weakly connected by transmission lines have multi-mode oscillations. These oscillations are generally analyzed in three main oscillation modes, i.e. local, inter-area and inter-machine modes. Depending upon their location in the system, some generators participate in only one mode of oscillation, while others participate in more than one mode [42, 43]. In this section, effectiveness of an APSS to damp out multi-mode oscillations in a multi-machine environment is described.

4.4.1 Multi-Machine Power System Model

A five machine power system without infinite bus, as shown in Fig. 4.17, is used to evaluate the performance of the proposed APSS. Five generating units are connected through a transmission network. Generators G_1, G_2 and G_4 have much larger capacities than G_3 and G_5. All five generators are equipped with governors, exciters and AVRs. The generating units are modeled by five first order differential equations (Appendix C.1). Parameters of all generators, governors, exciters, AVRs, transmission lines, loads and operating conditions are given in Appendix C.2-C.7. Due to different sizes of the generators and system configuration, multi-mode oscillations occur when the system experiences a disturbance.

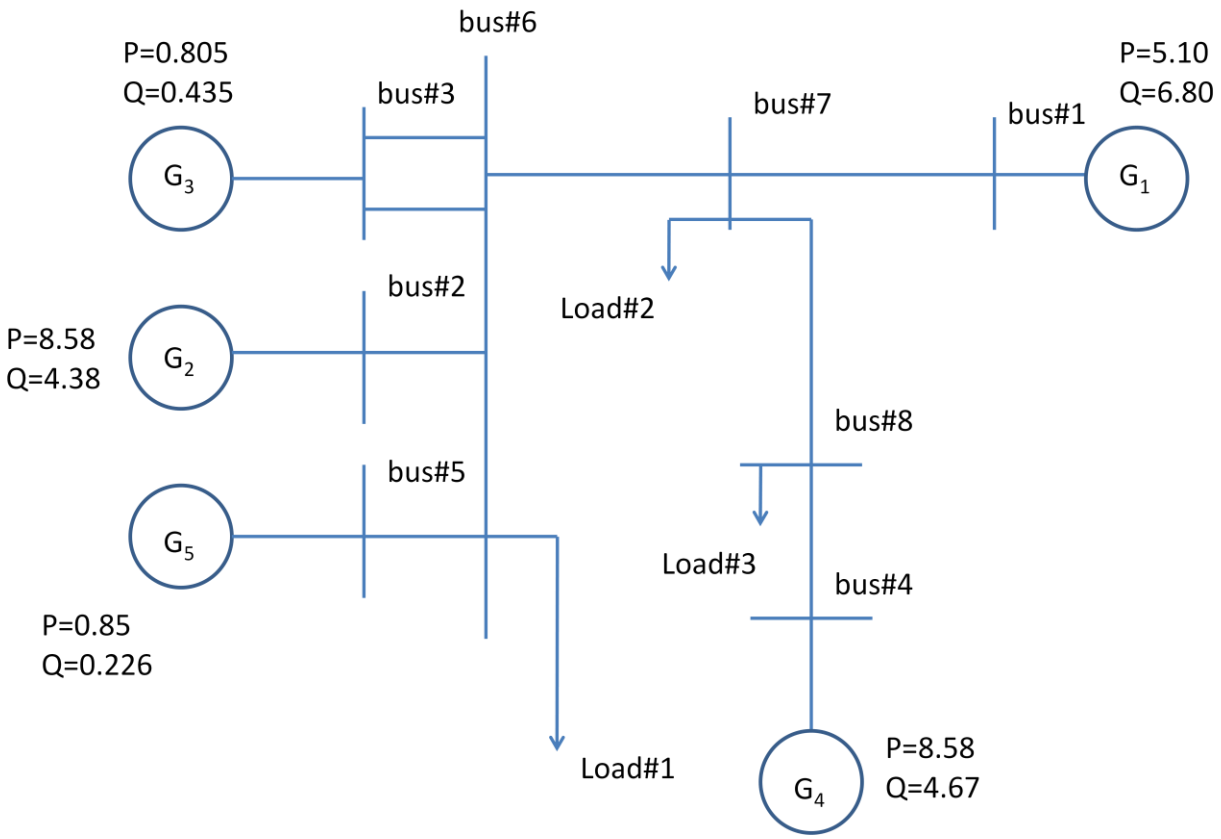


Fig 4.17 A five machine power system configuration

4.4.2 Multi-Machine Studies – Simulation Results

A non-linear fifth-order model is used to simulate the dynamic behavior of the multi-machine power system. The differential equations used to simulate the synchronous generators and the parameters used in simulation studies are given in the Appendix C.1-C.7. The AVR/Exciter and CPSS refer to IEEE Standard 421.5, Type ST1A AVR and Exciter Model and an IEEE Standard 421.5, PSS1A Type CPSS [50] (Appendix B.2 and B.4). The active power deviation, $\Delta P_e(k)$, is sampled at the rate of 20 Hz for parameter identification and control computation.

4.4.2.1 PSS on one unit

A 0.10 pu step decrease in input torque reference of G_3 is applied at 1s at the operating point #1 as given in Appendix C.6. The system returns to its initial condition at 10s. APSS was first installed on G_3 only. The APSS damps out the local mode oscillations effectively as shown in Fig 4.18. Since the rated capacity of G_3 is much less than G_1 and G_2, the APSS has little influence on the inter-area mode oscillations because these oscillations are introduced mainly by these large generators. G_3 does not have enough power to control the inter-area mode

oscillations. For comparison, a CPSS with the following transfer function [44] was installed on G_3:

$$U(s) = K_s \frac{sT_5}{1+sT_5} \frac{1+sT_1}{1+sT_2} \frac{1+sT_3}{1+sT_4} \Delta P(s) \quad (4.28)$$

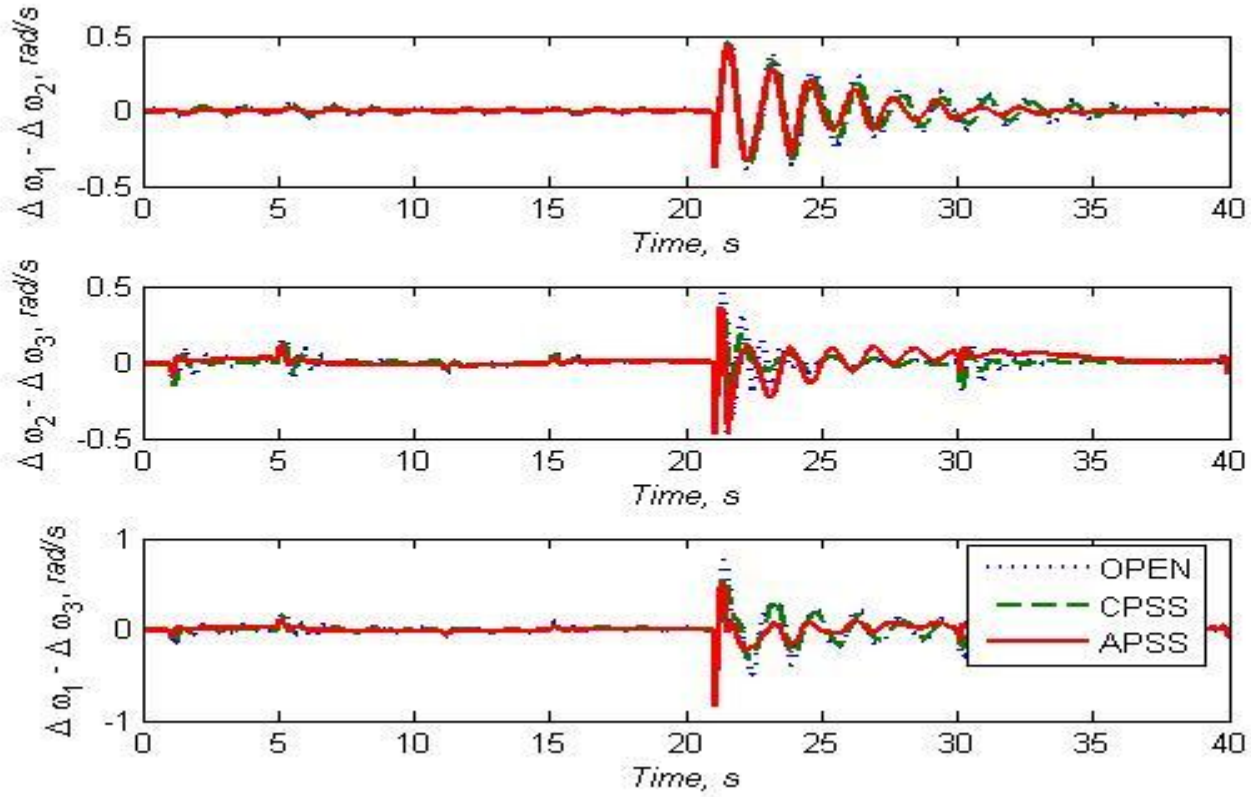


Fig 4.18 System response with APSS installed on generator G_3

4.4.2.2 PSS on three units

Two APSSs are additionally installed on G1 and G2 to damp both the local and the inter-area modes of oscillation. Fig. 4.19 shows that both modes of oscillations are damped out efficiently. Parameters of G1 and G2 have to be re-tuned if CPSSs are to be installed additionally on G1 and G2. Following parameters are used for the CPSS on G1 and G1:

$$K_s = 0.3, T_1 = T_3 = 0.07, T_2 = T_4 = 0.03, T_5 = 0.3 \quad (4.29)$$

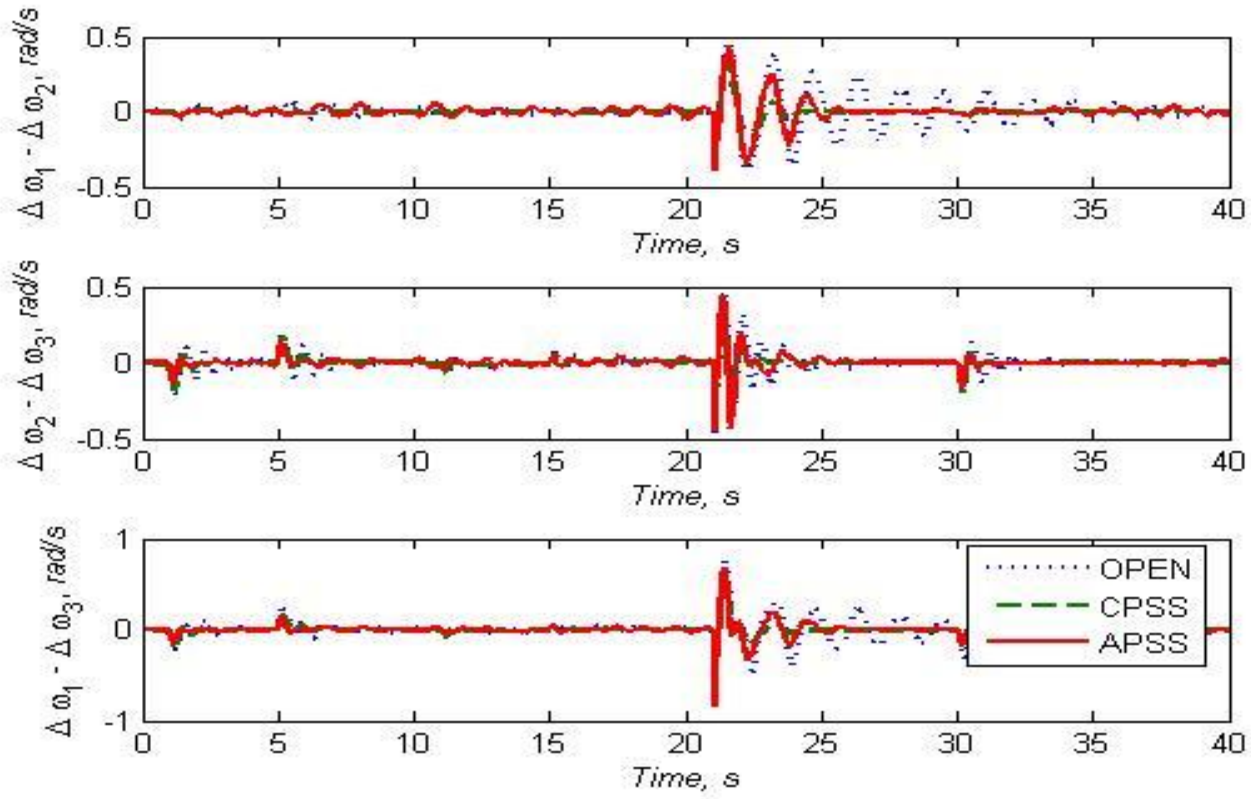


Fig 4.19 System response with PSSs installed on G_1, G_2 and G_3

4.4.2.3 Self-coordination ability of APSS

One of the important features of the APSS is its self-coordinating property. The response for 0.10 pu step decrease in torque reference of G_3 at 1 s is shown in Fig 4.20. The system returns to initial condition at 10 s. The proposed APSS can coordinate itself with existing PSSs in the system automatically due to its on-line learning property. APSS is installed on G_1 and G_3 and CPSS on G_2, G_4 and G_5 to prove this fact. Fig 4.20 shows that APSS can work cooperatively with other CPSSs to damp out the oscillations in the system. The proposed APSS coordinates itself with the other PSSs based on the system behavior at the generator terminals.

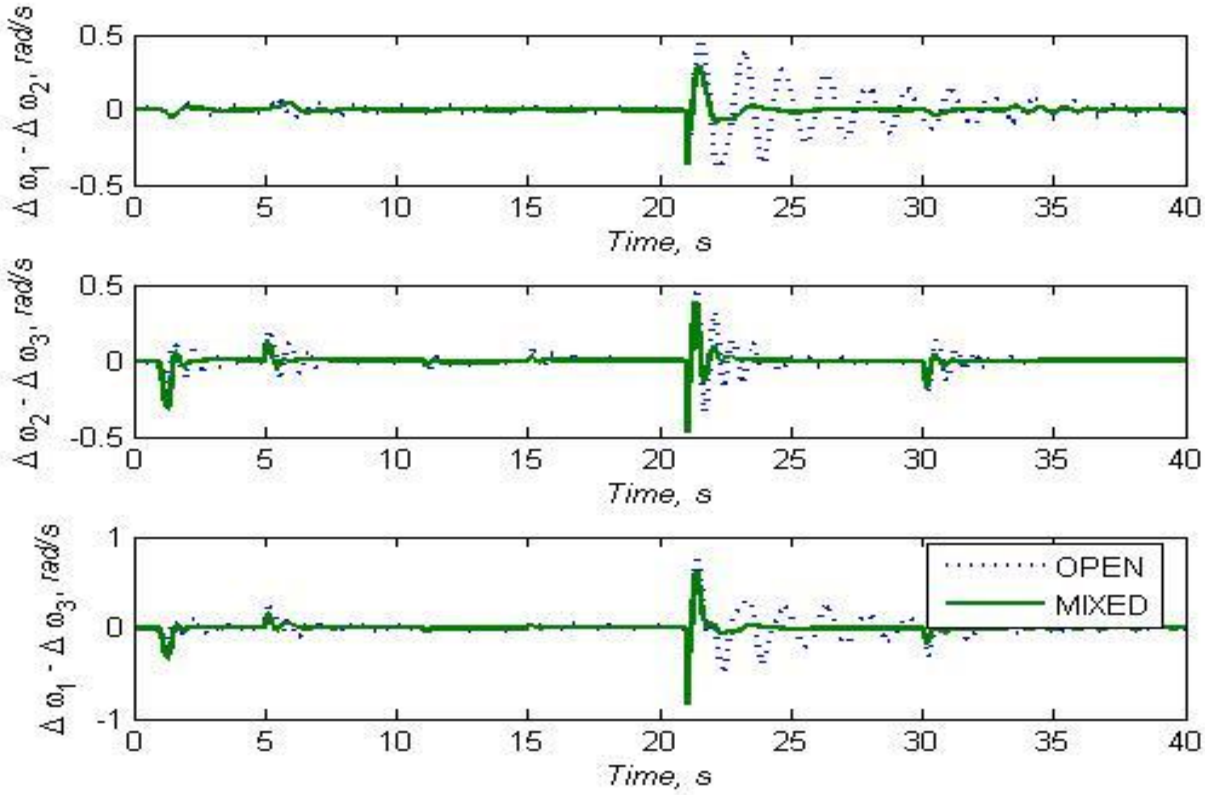


Fig 4.20 System response with APSS installed on generators G_1 and G_3 and CPSS on G_2, G_4 and G_5

4.4.2.4 Three-phase to ground fault test

A three phase to ground fault was applied at the middle of one transmission line between buses 3 and 6 at 1 s and cleared 50 ms later by removing the faulted line. The power system operates at the same operating conditions (operating condition #1, Appendix C.6). The faulted transmission line was restored successfully at 11 s. Fig 4.21 shows the response of the system under this disturbance.

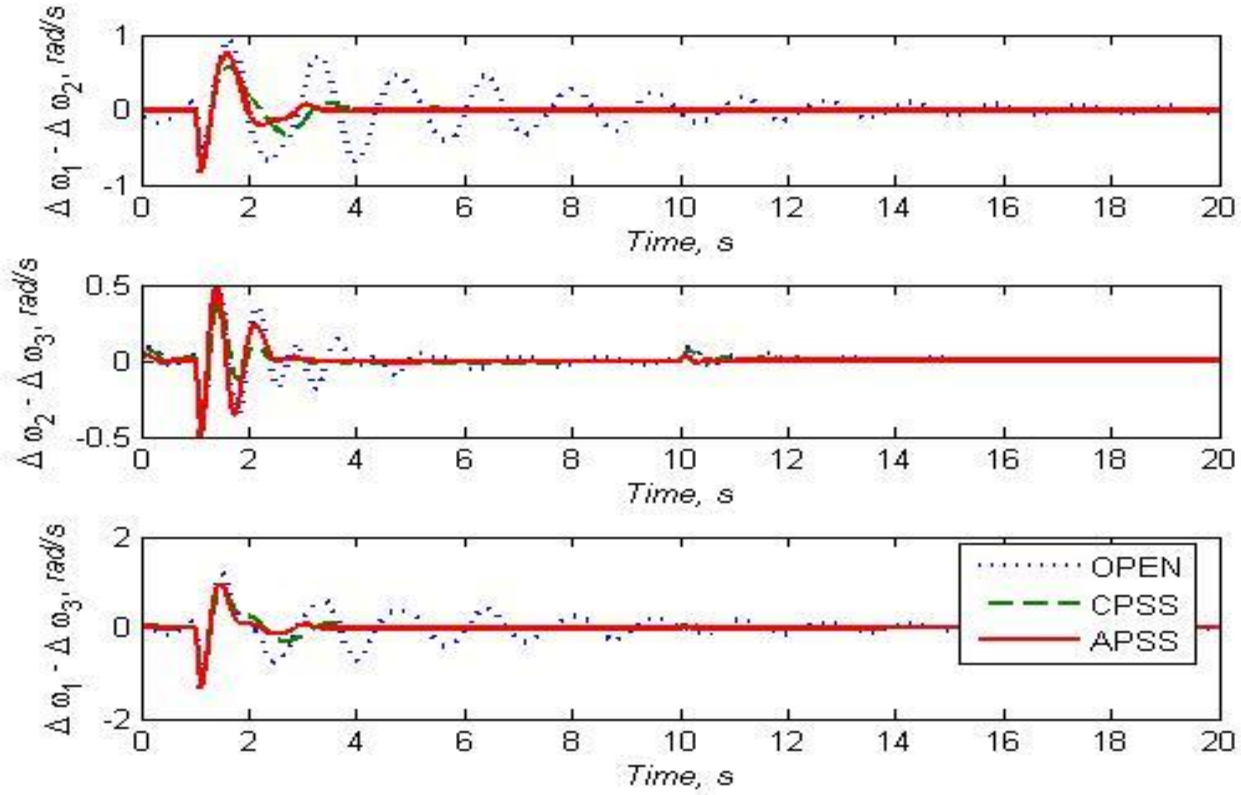


Fig 4.21 System response to a three phase to ground fault with PSSs installed on G_1, G_2 and G_3

4.5 Summary

In this chapter FLN-Identifier and PS Controller for APSS are described. Simulation studies for a single-machine infinite bus and multi-machine system are given in this chapter. Simulation results show that the given APSS with FLN-Identifier damps the local and inter-area mode oscillations effectively compared to a CPSS. Conclusion and reference to future work are given in the next chapter.

CHAPTER 5

CONCLUSIONS

This thesis presented an adaptive power system stabilizer consisting of a combined functional link network (FLN) identifier and Pole-shift (PS) control algorithm to damp low frequency oscillations in the power system. The advantage of using an adaptive power system stabilizer is that the parameters can be tuned on-line with the change in the operating conditions of the power system.

5.1 Summary

Chapter 1 introduced the concept of power system stability and control and gave an overview of the power system damping controllers. A summarized description of conventional power system stabilizers (CPSS) and adaptive power system stabilizers (APSS) were also provided in this chapter. A brief summary of the on-line parameter identifier and the control strategy were also given.

Chapter 2 discussed the various neural network architectures and their learning algorithms. The literature review summarized the advantages and disadvantages of using various artificial neural networks for system identification. Learning algorithm and linearization of the FLN identifier were also given in this chapter.

Chapter 3 described the Pole-Shift control algorithm and its advantages over moving average (MV) and pole-assignment (PA) control algorithms. A linear discrete system example was also given to prove the effectiveness of the proposed algorithm.

Chapter 4 explained the combined FLN identifier and the PS controller for adaptive power system stabilizers. Simulation results for a single-machine infinite bus and multi-machine system were given in this chapter. The results showed that the APSS can effectively damp local mode oscillations and also the inter-area oscillations to some extent.

5.2 Strengths of the Research

Power system stabilizers are the most cost-effective means of damping low frequency modes of oscillations in the power system. The North American Reliability Council has mandated the use of power system stabilizers for all generators after the August 15, 2003 blackout in Eastern Canada and the United States. Other transmission control devices, such as HVDC links, static var compensators and FACTS also enhance the stability of power system, but

the cost of these devices is so prohibitive that they are recommended only after the power system stabilizers are insufficient to damp all the low-frequency oscillations in the power system [44, 45]. Strengths of my research can be summarized as follows:

Firstly, the proposed adaptive power system stabilizer helped in the damping of the low frequency modes (local modes as well as inter-area modes) of oscillation in the power system. Low frequency oscillations, if they are not adequately damped, can lead to further instability conditions such as major transient oscillatory problems in the system, which may result in the isolation of generator from the power system and the interruption of power to the customers.

Secondly, the proposed adaptive power system stabilizer, provided optimal performance for wide range of operating conditions such as change in reference voltage of the power system, stressed leading power factor conditions, light load and three-phase fault conditions, whereas the conventional power system stabilizers used in industry are designed to provide optimal performance for normal load conditions only. Damping provided by these conventional stabilizers may not be as good for other operating conditions. In a multi-machine power system, conventional stabilizer parameters need to be re-tuned to coordinate with other machines and utilities, whereas re-tuning is not required for adaptive power system stabilizers, because of its inherent self-tuning capability [46, 47].

From the results discussed in Section 4.3.1 to 4.3.5, we can see that proposed self-tuning power system stabilizer offered more or less the same performance as the conventional stabilizer for normal load condition, but the performance of the proposed adaptive stabilizer was better than the conventional stabilizer for other operating conditions because of its self-tuning capabilities. This is one of the main strength of the proposed adaptive power system stabilizer.

Thirdly, as the proposed functional link network does not have any hidden layer, the identifier architecture implementation was simple and the computational complexity was also reduced. The simple functional link network identifier demonstrated its capacity to learn changes in the plant dynamics and its capacity to adjust to the new power system operating conditions quickly and effectively. Also, the proposed adaptive stabilizer used a linear feedback controller namely the Pole-shift controller. The proposed Pole-shift controller overcame the difficulty of choosing suitable values for the performance index parameters often-encountered in other types of pole-placement controllers discussed in the optimal control theory literature.

Fourthly, as discussed in Section 3.2.8, the proposed Pole-shift controller moved the unstable poles inside the unit circle in the z-plane and thereby ensured the closed loop stability of the system at every sampling instant. The proposed adaptive stabilizer with pole-shift control technique provided very good performance in terms of shorter settling time (2-3 seconds) and small overshoot (<5%). Other modern techniques like neuro-control or fuzzy logic based control found in the literatures do not offer this advantage as there are no theoretical analyses available to prove the stability of the overall feedback control system using such controllers.

5.3 Limitations of the Research Work

The limitations of the research can be summarized as follows:

Firstly, the simulation results showed that the power system stabilizer was very effective in damping local modes of oscillation. But the stabilizer was not equally effective for damping the inter-area oscillations. The previous works suggest that, the power system stabilizers are primarily employed to damp the local area oscillations [48, 49]. Further investigations are needed using power system stabilizers in conjunction with transmission control devices. For example, transmission control devices such as thyristor controlled series capacitors (TCSC) could be installed on the line. Adaptive power system stabilizers along with such transmission line control devices would be able to damp both local and inter-area oscillations effectively.

Secondly, the proposed adaptive stabilizer was shown to give better performance compared to the conventional stabilizers using simulation studies, but this has to be verified using actual experimental testing. True validation of the proposed adaptive stabilizer would be by testing, using an experimental set up consisting of a synchronous generator connected through a transmission line to the city grid. Such a testing would provide practical validation of the proposed adaptive stabilizer for industry acceptability.

Thirdly, the proposed adaptive controller used a 3^{rd} order model for control calculations based on the findings of some of the previous research works. A simplified 3^{rd} order model could have been one of the reasons that the adaptive stabilizer did not perform well in the multi-machine environment (especially for damping the inter-area oscillations). The performance of the proposed adaptive power system stabilizer was compared with that of the conventional stabilizer, and the simulation results proved that the adaptive power system stabilizer can damp local mode oscillations effectively for varying operating conditions; however, its performance

was not equally effective in damping the inter-area oscillations. The proposed functional link network identifier has demonstrated the capacity to learn changes in the plant dynamics and to adjust to the new operating conditions quickly and effectively. This was a limitation of the controller and further investigations are needed with higher-order controllers and its impact on damping inter-area oscillations.

5.4 Suggestions for Future Work

1. It was shown in this thesis that the power system stabilizers are very effective for damping the local mode oscillations and are also useful for damping the inter area oscillations to some degree. The simulation study on the multi-machine system was done to show that adaptive power system stabilizer was capable of damping inter-area oscillations. However, in some cases, the conventional power system stabilizer gave slightly better performance compared to the adaptive power system stabilizer. Further studies are needed in the multi-machine environment to get a true assessment of the adaptive power system stabilizer in comparison to the conventional power system stabilizer.
2. In real world power systems, all the controllers will be implemented on digital processor boards with certain clock speeds. Each sample control calculation can take different times, depending on the controller complexity. The choice of sampling frequency is very essential in real-time scenarios and therefore testing the proposed adaptive controller in real environment will be required. The Real Time Digital Simulator (RTDS) available in the Power Research Laboratory is capable of performing continuous real time operation. It uses a combination of custom hardware (Digital Signal Processors utilizing parallel processing technique) and software (accurate power system component models) models. Real time simulations provide a solid framework to test the proposed control algorithm. It would be useful to test the proposed Pole-shift control algorithm on RTDS and this would provide an excellent verification of the proposed FLN identifier and Pole shift control algorithm.

REFERENCES

[1] P. Kundur, J. Paserba, V. Ajjarapu, G. Andersson, A. Bose, C. Canizares, N. Hatziargyriou, D. Hill, A. Stankovic, C. Taylor, T. Van Cutsem, and V. Vittal, "Definition and classification of power system stability IEEE/CIGRE joint task force on stability terms and definitions," *Power Systems, IEEE Transactions on,* vol. 19, pp. 1387-1401, 2004.

[2] M. K. El-Sherbiny and D. M. Mehta, "Dynamic System Stability Part I - Investigation of the Effect of Different Loading and Excitation Systems," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-92, pp. 1538-1546, 1973.

[3] H. A. M. Moussa and Y. Yao-nan, "Dynamic Interaction of Multi-Machine Power System and Excitation Control," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-93, pp. 1150-1158, 1974.

[4] V. A. Venikov and V. A. Stroev, "Power System Stability as Affected by Automatic Control of Generators-Some Methods of Analysis and Synthesis," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-90, pp. 2483-2487, 1971.

[5] P. C. Krause and J. N. Towle, "Synchronous Machine Damping by Excitation Control with Direct and Quadrature Axis Field Windings," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-88, pp. 1266-1274, 1969.

[6] M. K. El-Sherbiny and A. A. Fouad, "Digital Analysis of Excitation Control for Interconnected Power Systems," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-90, pp. 441-447, 1971.

[7] W. A. Mittelstadt, "Four Methods of Power System Damping," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-87, pp. 1323-1329, 1968.

[8] F. R. Schleif, H. D. Hunkins, G. E. Martin, and E. E. Hattan, "Excitation Control to Improve Powerline Stability," *Power Apparatus and Systems, IEEE Transactions on,* vol. PAS-87, pp. 1426-1434, 1968.

[9] A. Ghafouri, M. R. Zolghadri, M. Ehsan, O. Elmatboly, and A. Homaifar, "Fuzzy Controlled STATCOM for Improving the Power System Transient Stability," in *Power Symposium, 2007. NAPS '07. 39th North American,* 2007, pp. 212-216.

[10] K. M. Sze, L. A. Snider, T. S. Chung, and K. W. Chan, "An intelligent fuzzy controlled SSSC to enhance power system stability," in *Power System Technology, 2004. PowerCon 2004. 2004 International Conference on,* 2004, pp. 1183-1188 Vol.2.

[11] D. K. Chaturvedi and O. P. Malik, "Generalized neuron-based adaptive PSS for multimachine environment," *Power Systems, IEEE Transactions on,* vol. 20, pp. 358-366, 2005.

[12] P. Zhao and O. P. Malik, "Operating-condition-dependent ARMA model for PSS application," in *Power Engineering Society General Meeting, 2004. IEEE,* 2004, pp. 1749-1754 Vol.2.

[13] L. H. Hassan, M. Moghavvemi, and H. A. F. Mohamed, "Power system stabilization based on artificial intelligent techniques; A review," in *Technical Postgraduates (TECHPOS), 2009 International Conference for,* 2009, pp. 1-6.

[14] P. Shamsollahi and O. P. Malik, "Application of neural adaptive power system stabilizer in a multi-machine power system," *Energy Conversion, IEEE Transactions on,* vol. 14, pp. 731-736, 1999.

[15] Z. Peng and O. P. Malik, "Design of an Adaptive PSS Based on Recurrent Adaptive Control Theory," *Energy Conversion, IEEE Transactions on*, vol. 24, pp. 884-892, 2009.
[16] O. P. Malik, "Adaptive and artificial intelligence based PSS," in *Power Engineering Society General Meeting, 2003, IEEE*, 2003, pp. 1792-1797 Vol. 3.
[17] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *Neural Networks, IEEE Transactions on*, vol. 1, pp. 4-27, 1990.
[18] J. C. Patra, R. N. Pal, B. N. Chatterji, and G. Panda, "Identification of nonlinear dynamic systems using functional link artificial neural networks," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 29, pp. 254-262, 1999.
[19] D. K. Chaturvedi, O. P. Malik, and P. K. Kalra, "Performance of a generalized neuron-based PSS in a multimachine power system," *Energy Conversion, IEEE Transactions on*, vol. 19, pp. 625-632, 2004.
[20] S. Chen and S. A. Billings, "Neural networks for nonlinear dynamic system modelling and identification," *International Journal of Control*, vol. 56, pp. 319-346, 1992.
[21] J. C. Patra and A. C. Kot, "Nonlinear dynamic system identification using Chebyshev functional link artificial neural networks," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 32, pp. 505-511, 2002.
[22] D. A. Pierre, "A Perspective on Adaptive Control of Power Systems," *Power Engineering Review, IEEE*, vol. PER-7, pp. 45-46, 1987.
[23] H. T. Siegelmann, B. G. Horne, and C. L. Giles, "Computational capabilities of recurrent NARX neural networks," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 27, pp. 208-215, 1997.
[24] W. Jianzhou, L. Jinzhao, C. Jinxing, and S. Donghuai, "ARMA Model identification using Particle Swarm Optimization Algorithm," in *Computer Science and Information Technology, 2008. ICCSIT '08. International Conference on*, 2008, pp. 223-227.
[25] D. K. Seidel and P. Davies, "Modeling nonlinear systems by using nonlinear autoregressive moving average models," in *Acoustics, Speech, and Signal Processing, 1990. ICASSP-90., 1990 International Conference on*, 1990, pp. 2559-2562 vol.5.
[26] Y. Zhang, G. P. Chen, O. P. Malik, and G. S. Hope, "An artificial neural network based adaptive power system stabilizer," *Energy Conversion, IEEE Transactions on*, vol. 8, pp. 71-77, 1993.
[27] M. Zhang, *Artificial higher order neural networks for computer science and engineering : trends for emerging applications*. Hershey, PA: Information Science Reference.
[28] Y. Shin and J. Ghosh, "The pi-sigma network: an efficient higher-order neural network for pattern classification and function approximation," in *Neural Networks, 1991., IJCNN-91-Seattle International Joint Conference on*, 1991, pp. 13-18 vol.1.
[29] I. Hassanzadeh, M. B. B. Sharifian, S. Khanmohammadi, and R. Kenarangui, "A FLN artificial neural network based fuzzy controller for generator excitation control," in *Electrical and Computer Engineering, 2000 Canadian Conference on*, 2000, pp. 702-706 vol.2.
[30] Y. Pao, *Adaptive pattern recognition and neural networks*, 1989.
[31] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural Netw.*, vol. 4, pp. 251-257, 1991.
[32] G. P. Chen and O. P. Malik, "Tracking constrained adaptive power system stabiliser," *Generation, Transmission and Distribution, IEE Proceedings-*, vol. 142, pp. 149-156, 1995.

[33] G. P. Chen, O. P. Malik, G. S. Hope, Y. H. Qin, and G. Y. Xu, "An adaptive power system stabilizer based on the self-optimizing pole shifting control strategy," *Energy Conversion, IEEE Transactions on*, vol. 8, pp. 639-645, 1993.

[34] T. Abdelazim and O. P. Malik, "Fuzzy logic based identifier and pole-shifting controller for PSS application," in *Power Engineering Society General Meeting, 2003, IEEE*, 2003, pp. 1680-1685 Vol. 3.

[35] K. J. Åström, U. Borisson, L. Ljung, and B. Wittenmark, "Theory and applications of self-tuning regulators," *Automatica*, vol. 13, pp. 457-476, 1977.

[36] A. Ghosh, G. Ledwich, O. P. Malik, and G. S. Hope, "Power System Stabilizer Based on Adaptive Control Techniques," *Power Apparatus and Systems, IEEE Transactions on*, vol. PAS-103, pp. 1983-1989, 1984.

[37] S.-j. Cheng, Y. S. Chow, O. P. Malik, and G. S. Hope, "An Adaptive Synchronous Machine Stabilizer," *Power Systems, IEEE Transactions on*, vol. 1, pp. 101-107, 1986.

[38] O. P. Malik, G. S. Hope, and S. J. Cheng, "Some issues on the practical use of recursive least squares identification in self-tuning control," *International Journal of Control*, vol. 53, pp. 1021 - 1033, 1991.

[39] O. P. Malik, G. P. Chen, G. S. Hope, Y. H. Qin, and G. Y. Xu, "Adaptive self-optimising pole shifting control algorithm," *Control Theory and Applications, IEE Proceedings D*, vol. 139, pp. 429-438, 1992.

[40] J. Teeter and C. Mo-Yuen, "Application of functional link neural network to HVAC thermal dynamic system identification," *Industrial Electronics, IEEE Transactions on*, vol. 45, pp. 170-176, 1998.

[41] B. Changaroon, S. C. Srivastava, and D. Thukaram, "A neural network based power system stabilizer suitable for on-line training-a practical case study for EGAT system," *Energy Conversion, IEEE Transactions on*, vol. 15, pp. 103-109, 2000.

[42] W. Hussein and O. P. Malik, "GA-identifier and predictive controller for multi-machine power system," in *Power India Conference, 2006 IEEE*, 2006, p. 6 pp.

[43] E. V. Larsen and D. A. Swann, "Applying Power System Stabilizers. Part I: General Concepts," *Power Engineering Review, IEEE*, vol. PER-1, pp. 62-63, 1981.

[44] *Power system stability and control*: McGraw-Hill Education (India) Pvt Ltd, 1994.

[45] G. Rogers, "Demystifying power system oscillations," *Computer Applications in Power, IEEE*, vol. 9, pp. 30-35, 1996.

[46] Y. L. Abdel-Magid, M. A. Abido, and A. H. Mantaway, "Robust tuning of power system stabilizers in multimachine power systems," *Power Systems, IEEE Transactions on*, vol. 15, pp. 735-740, 2000.

[47] A. L. B. Do Bomfim, G. N. Taranto, and D. M. Falcao, "Simultaneous tuning of power system damping controllers using genetic algorithms," *Power Systems, IEEE Transactions on*, vol. 15, pp. 163-169, 2000.

[48] P. Kundur, M. Klein, G. J. Rogers, and M. S. Zywno, "Application of power system stabilizers for enhancement of overall system stability," *Power Systems, IEEE Transactions on*, vol. 4, pp. 614-626, 1989.

[49] N. M. Muhamad Razali, V. K. Ramachandaramurthy, and R. N. Mukerjee, "Power System Stabilizer Placement and Tuning Methods for Inter-area Oscillation Damping," in *Power and Energy Conference, 2006. PECon '06. IEEE International*, 2006, pp. 173-178.

[50] "IEEE Recommended Practice for Excitation System Models for Power System Stability Studies," *IEEE Std 421.5-1992*, p. 0_1, 1992.

APPENDIX A

BACK PROPAGATION ALGORITHM

A back-propagation neural network is a layered network consisting of an input layer, an output layer and at least one layer of nonlinear processing elements. The nonlinear processing elements, which sum incoming signals and generate output signals according to some predefined function, are called *neurons*. The neurons are connected by terms with variable weights. The output of one neuron multiplied by a weight becomes the input of an adjacent neuron of the next layer. The connection weights between the neurons must be determined before it can be used in the application. The process of determining the weights is called the training process. MLPs employ the error back-propagation method to train the interconnecting weights. Back propagation learning requires a set of input and output pairs. Before the training, small random weights are assigned to the network. For each training data set p, the input pattern is presented at the input along with a desirable output pattern. Weights are then adjusted to eliminate the total squared error, Err, which is the sum of the difference squared between the set of required outputs and the set of actual outputs of the ANN:

$$Err_p = \frac{1}{2}\sum(t_{pj} - out_{pj})^2 \quad \text{(A.1)}$$

where t_{pj} is one of the required outputs and out_{pj} is one of the actual ANN output.

The weight w_{ij} can be adjusted to minimize Err_p for the set of p training data set by a gradient descent method,

$$w_{ji}(n+1) = w_{ji}(n) + \Delta w_{ji}(n) \quad \text{(A.2)}$$

$$\Delta w_{ji}(n) = \eta \delta_{pj} out_{pj} \quad \text{(A.3)}$$

where n is the iteration number, η is the learning rate and

$$\delta_{pj} = \frac{d\ out_{pj}}{d\ sum_{pj}}\ (t_{pj} - out_{pj}) \quad \text{(A.4)}$$

If neuron j is in the hidden layer,

$$\delta_{pj} = \frac{d\ out_{pj}}{d\ sum_{pj}} \sum_k \delta_{pk} w_{kj} \quad \text{(A.5)}$$

Better convergence can be obtained if a momentum term is added to equation (A.3), as

$$\Delta w_{ji}(n) = \eta \delta_{pj} out_{pj} + \beta \Delta w_{ji}(n-1) \quad \text{(A.6)}$$

where β is the momentum factor. In the above equations, learning rate, η, and the momentum factor, β, are between 0.0 and 1.0.

Disadvantages of BP algorithm:

1. BO learning algorithm gets trapped in local minima easily, especially for non-linearly separable problems
2. The convergent speed of the BP learning is too slow even if the learning goal can be achieved.
3. The convergent behavior of the BP learning algorithm depends very much on the choices of initial values of the network connection weights as well as the parameters in the algorithm such as the learning rate and momentum.

APPENDIX B

SINGLE-MACHINE POWER SYSTEM

B.1 The generating unit is modeled by seven first order differential equations given below:

$$\dot{\delta} = \omega_0 \omega \tag{B.1}$$

$$\dot{\omega} = \frac{1}{2H}(T_m + g - T_e) \tag{B.2}$$

$$\dot{\lambda}_d = e_d + r_a i_d + \omega_0(\omega + 1)\lambda_q \tag{B.3}$$

$$\dot{\lambda}_q = e_q + r_a i_q - \omega_0(\omega + 1)\lambda_d \tag{B.4}$$

$$\dot{\lambda}_f = e_f - r_f i_f \tag{B.5}$$

$$\dot{\lambda}_{kd} = -r_{kd} i_{kd} \tag{B.6}$$

$$\dot{\lambda}_{kq} = -r_{kq} i_{kq} \tag{B.7}$$

B.2 The AVR and exciter combination used in the system is from the IEEE Standard P421.5/D15, Type ST1A shown in Fig B.1[50]

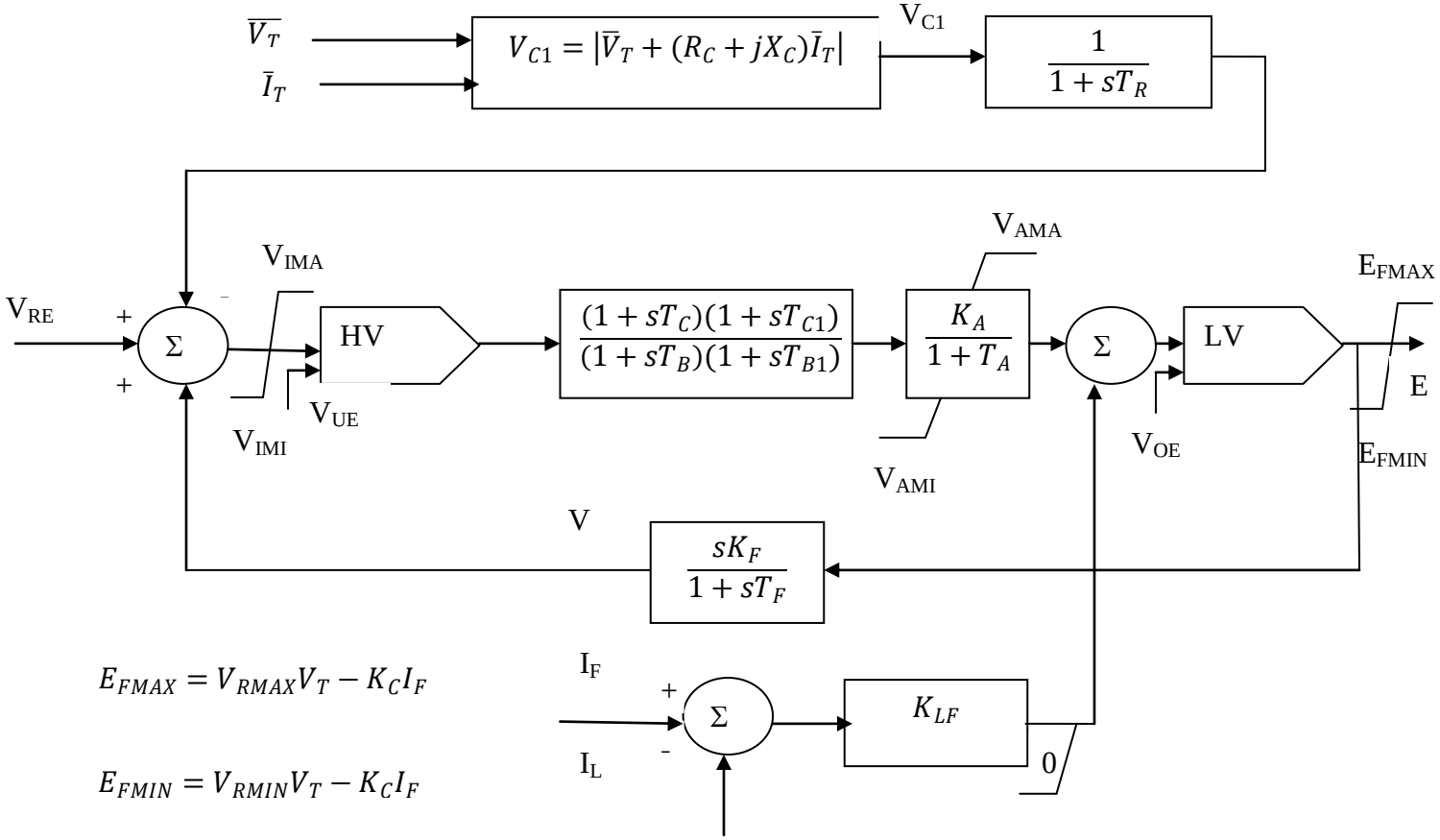


Fig B.1. AVR and exciter model Type ST1A, IEEE Standard P421.5/D15

B.3 The governor used in the system has the transfer function

$$g(s) = \left[a + \frac{b}{1+sT_g}\right]\dot{\delta} \qquad \text{(B.8)}$$

B.4 The conventional power system stabilizer is Type PSS1A from IEEE Standard P421.5/D15 shown in Fig B.2[50]

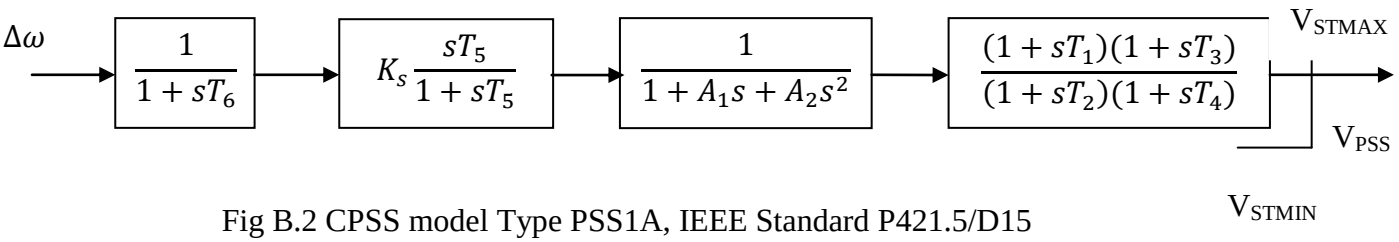


Fig B.2 CPSS model Type PSS1A, IEEE Standard P421.5/D15

B.5 Parameters

$r_a = 0.007$	$H = 4.000$	$r_f = 0.00089$	$r_{kd} = 0.023$
$r_{kq} = 0.023$	$x_d = 1.24$	$x_f = 1.33$	$x_{kd} = 1.15$
$x_{md} = 1.126$	$x_q = 0.743$	$x_{kq} = 0.652$	$x_{mq} = 0.626$
$K_d = -0.0027$	$r_t = 0.05$	$x_t = 0.06$	$R_C = 0.0$
$X_C = 0.0$	$T_R = 0.04$	$K_A = 190.0$	$K_C = 0.08$
$K_F = 0.0$	$K_{LF} = 0.0$	$I_{LR} = 0.0$	$T_A = 0.0$
$T_B = 10.0$	$T_C = 1.0$	$T_{B1} = 0.0$	$T_{C1} = 0.0$
$T_F = 1.00$	$V_{OEL} = 999$	$V_{UEL} = -999$	$V_{IMAX} = 9999.0$
$V_{IMIN} = -9999.0$	$V_{AMAX} = 9999.0$	$V_{AMIN} = -9999.0$	$V_{RMAX} = 7.8$
$V_{RMIN} = -6.7$	$a = -0.00133$	$b = -0.17$	$T_g = 0.25$
$T_1 = 0.1$	$T_2 = 0.01$	$T_3 = 0.1$	$T_4 = 0.01$
$T_5 = 2.85$	$T_6 = 0.005$	$K_s = 0.02$	$A_1 = 0.0$
$A_2 = 0.0$	$V_{STMAX} = 0.1$	$V_{STMIN} = -0.1$	

All resistances and reactances are in per-unit and time constants in seconds.

APPENDIX C

MULTI-MACHINE POWER SYSTEM

C.1 The generating unit is modeled by five first order differential equations given below

$$\dot{\delta} = \omega_0 \omega \tag{C.1}$$

$$\dot{\omega} = \frac{1}{2H}(T_m + g + K_d\dot{\delta} - T_e) \tag{C.2}$$

$$T'_{d0}\dot{e'}_q = e_f - (x_d - x'_d)i_d - e'_q \tag{C.3}$$

$$T''_{d0}\dot{e''}_q = \left[e'_q - (x'_d - x''_d)i_d - e''_q\right] + T''_{d0}\dot{e'}_q + T''_{d0}\dot{e'}_q \tag{C.4}$$

$$T''_{q0}\dot{e''}_d = \left(x_q - x''_q\right)i_q - e''_d \tag{C.5}$$

C.2 Parameters of the generators

	Gen#1	Gen#2	Gen#3	Gen#4	Gen#5
x_d	0.1026	0.1026	1.0260	0.1026	1.0260
x_q	0.0658	0.0658	0.6580	0.0658	0.6580
x'_d	0.0339	0.0339	0.3390	0.0339	0.3390
x''_d	0.0269	0.0269	0.2690	0.0269	0.2690
x''_q	0.0335	0.0335	0.3350	0.0335	0.3350
T'_{d0}	5.6700	5.6700	5.6700	5.6700	5.6700
T''_{d0}	0.6140	0.6140	0.6140	0.6140	0.6140
T''_{q0}	0.7230	0.7230	0.7230	0.7230	0.7230
H	80.000	80.000	10.000	80.000	10.000

C.3 Parameters of AVRs and simplified ST1A exciters

	Gen#1	Gen#2	Gen#3	Gen#4	Gen#5
T_R	0.0400	0.0400	0.0400	0.0400	0.0400
K_A	190.00	190.00	190.00	190.00	190.00
K_C	0.0800	0.0800	0.0800	0.0800	0.0800
T_B	10.000	10.000	10.000	10.000	10.000
T_C	1.0000	1.0000	1.0000	1.0000	1.0000

The output of all exciters is limited within - 6.7 to 7.8 p.u.

C.4 Parameters of the governors

	Gen#1	Gen#2	Gen#3	Gen#4	Gen#5
T_g	0.25000	0.25000	0.25000	0.25000	0.25000
a	-0.00015	-0.00015	-0.00133	-0.00015	-0.00133
b	-0.01500	-0.01500	-0.17000	-0.01500	-0.17000

C.5 Parameters of transmission lines in p.u.

Bus No.	R	X	B/2	Bus No.	R	X	B/2
1-7	.00435	.01067	.01536	2-6	.00213	.00468	.00404
3-6	.01002	.03122	.03204	3-6	.01002	.03122	.03204
4-8	.00524	.01184	.01756	5-6	.00711	.02331	.02732
6-7	.04032	.12785	.15858	7-8	.01724	.04153	.06014

C.6 Operating conditions and loads for operating point #1

	Gen#1	Gen#2	Gen#3	Gen#4	Gen#5
$P\ (p.u.)$	5.1076	8.5835	1.8055	8.5670	0.8501
$Q\ (p.u.)$	6.8019	4.3836	0.4353	4.6686	0.2264
$V\ (p.u.)$	1.0750	1.0500	1.0250	1.0750	1.0250
$\delta\ (rad.)$	0.0000	0.3167	0.2975	0.1174	0.3051

Load in admittances in p.u.

$L_1 = 7.5 - j5.0$ $L_2 = 8.5 - j5.0$ $L_3 = 7.0 - j4.5$