DESIGN OF POWER SYSTEM STABILIZER

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology in
Electrical Engineering
By

AK Swagat Ranjan Swain (108EE085)

Ashit Kumar Swain (108EE087)

Abinash Mohapatra (108EE090)

Department of Electrical Engineering
National Institute of Technology, Rourkela
MAY 2012

DESIGN OF POWER SYSTEM STABILIZER

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology in
Electrical Engineering
By

AK Swagat Ranjan Swain (108EE085)

Ashit Kumar Swain (108EE087)

Abinash Mohapatra (108EE090)

Under the supervision of

Prof. BIDYADHAR SUBUDHI

Department of Electrical Engineering
National Institute of Technology, Rourkela
MAY 2012

DESIGN OF POWER SYSTEM STABILIZER

National Institute of Technology, Rourkela

CERTIFICATE

This is to certify that the thesis entitled **"Design of Power System Stabilizer"** submitted by **Arya Kumar Swagat Ranjan Swain (108EE085), Ashit Kumar Swain (108EE087), Abinash Mohapatra (108EE090)** in the partial fulfilment of the requirement for the degree of Bachelor of Technology in Electrical Engineering, National Institute of Technology, Rourkela, is an authentic work carried out by them under my supervision.

To the best of my knowledge the matter embodied in the thesis has not been submitted to any other university/institute for the award of any degree or diploma.

Date:

(Prof. Bidyadhar Subudhi)
Dept. of Electrical Engineering
National Institute of Technology
Rourkela-769008

ACKNOWLEDGEMENT

We wish to express our sincere gratitude to our guide and motivator *Prof. Bidyadhar Subudhi*, Electrical Engineering Department, National Institute of Technology, Rourkela for his invaluable guidance and co-operation, and for providing the necessary facilities and sources during the entire period of this project. The facilities and co-operation received from the technical staff of the Electrical Engineering Department is also thankfully acknowledged. Last, but not the least, we would like to thank the authors of various research articles and books that we referred to during the course of the project.

A.K. Swagat Ranjan Swain
Ashit Kumar Swain
Abinash Mohapatra

DESIGN OF POWER SYSTEM STABILIZER

National Institute of Technology, Rourkela

ABSTRACT

A power system stabilizer (PSS) installed in the excitation system of the synchronous generator improves the small-signal power system stability by damping out low frequency oscillations in the power system. It does that by providing supplementary perturbation signals in a feedback path to the alternator excitation system.

In our project we review different conventional PSS design (CPSS) techniques along with modern adaptive neuro-fuzzy design techniques. We adapt a linearized single-machine infinite bus model for design and simulation of the CPSS and the voltage regulator (AVR). We use 3 different input signals in the feedback (PSS) path namely, speed variation(w), Electrical Power (Pe), and integral of accelerating power (Pe*w), and review the results in each case.

For simulations, we use three different linear design techniques, namely, root-locus design, frequency-response design, and pole placement design; and the preferred non-linear design technique is the adaptive neuro-fuzzy based controller design.

The MATLAB package with Control System Toolbox and SIMULINK is used for the design and simulations.

CONTENTS:

CHAPTER-1

POWER SYSTEM STABILITY: INTRODUCTORY CONCEPTS

Power System Stability, its classification, and problems associated with it have been addressed by many CIGRE and IEEE publications. The CIGRE study committee and IEEE power systems dynamic performance committee defines power system stability as:

"Power system stability is the ability of an electrical power system, for given operating conditions, to regain its state of operating equilibrium after being subjected to a physical disturbance, with the system variables bounded, so that the entire system remains intact and the service remains uninterrupted" [3].

The figure below gives the overall picture of the stability problem:

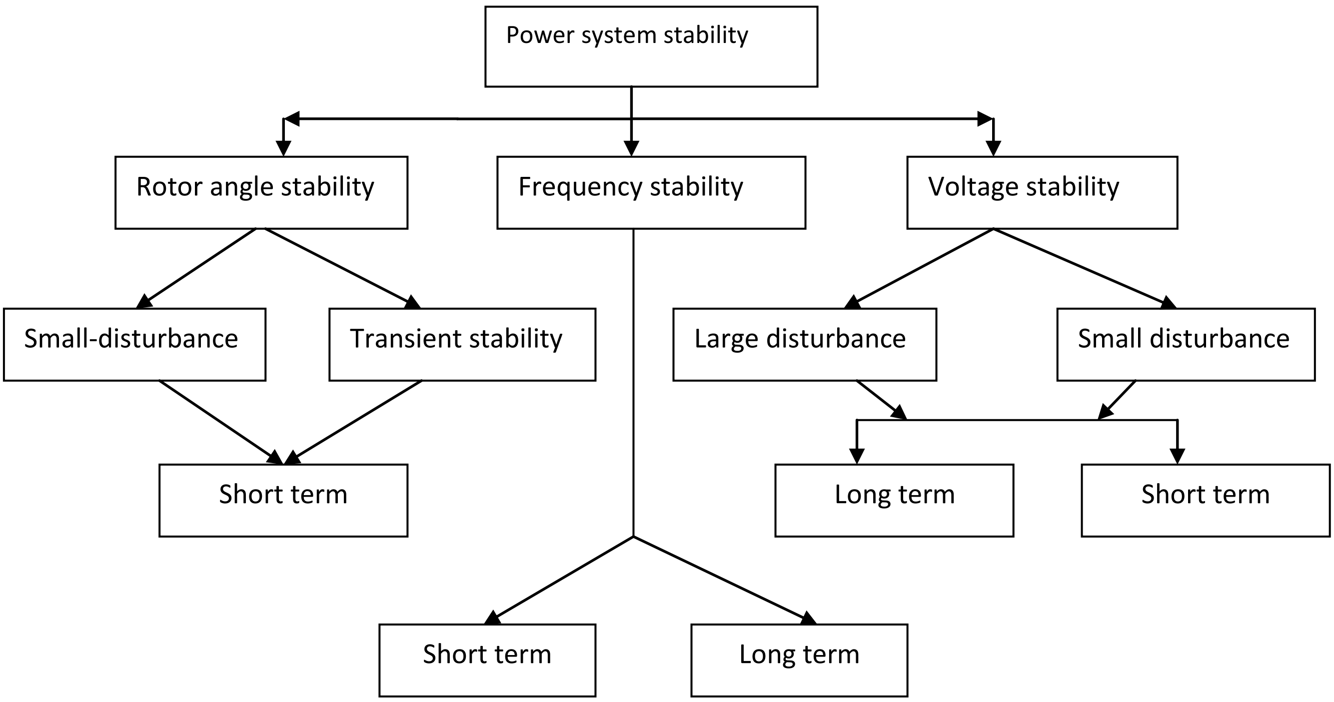


Fig.1. Power-system stability classification [24]

Out of all the stability problems mentioned above, our specific focus in this project is of **small disturbance stability** which is a part of the **rotor angle stability**. Also, the **voltage stability** due to **small disturbances** is covered.

Rotor angle stability:

This refers to the ability of the synchronous generator in an interconnected power system to remain in synchronism after being subjected to disturbances. It depends on the ability of the machine to maintain equilibrium between electromagnetic torque and mechanical torque of each synchronous machine in the system [24]. Instability of this kind occurs in the form of swings of the generator rotor which leads to loss of synchronism.

Small Disturbance Stability:

Small Disturbance stability may refer to small disturbance voltage or rotor angle stability. The disturbances are sufficiently small so as to assume a linearized system model. Small disturbances may be small incremental load changes, small control variations etc. It does not however include disturbances due to faults or short circuits.

CHAPTER-2

THE EXCITATION SYSTEM OF THE SYNCHRONOUS GENERATOR: AN OVERVIEW

In this chapter, we give a brief historical overview on the excitation system of the synchronous generator. Then we proceed to give the schematic diagram of the excitation system which we shall primarily use in this project to design the power system stabilizer.

The first step in the sophistication of the primitive excitation system was the introduction of the amplifier in the feedback path to amplify the error signal and make the system fast acting. With the increase in size of the units and interconnected systems, more and more complex excitation systems are being developed to make the system as stable as possible. With the advent of solid-state rectifiers, ac exciters are now in common use. [11]

A modern excitation system contains components like automatic voltage regulators (AVR), Power System stabilizers (PSS), and filters, which help in stabilizing the system and maintaining almost constant terminal voltage. These components can be analog or digital depending on the complexity, viability, and operating conditions. The final aim of the excitation system is to reduce swings due to transient rotor angle instability and to maintain a constant voltage. To do this, it is fed a reference voltage which it has to follow, which is normally a step voltage. The excitation voltage comes from the transmission line itself. The AC voltage is first converted into DC voltage by rectifier units and is fed to the excitation system via its components like the AVR, PSS etc. the different components are discussed later.

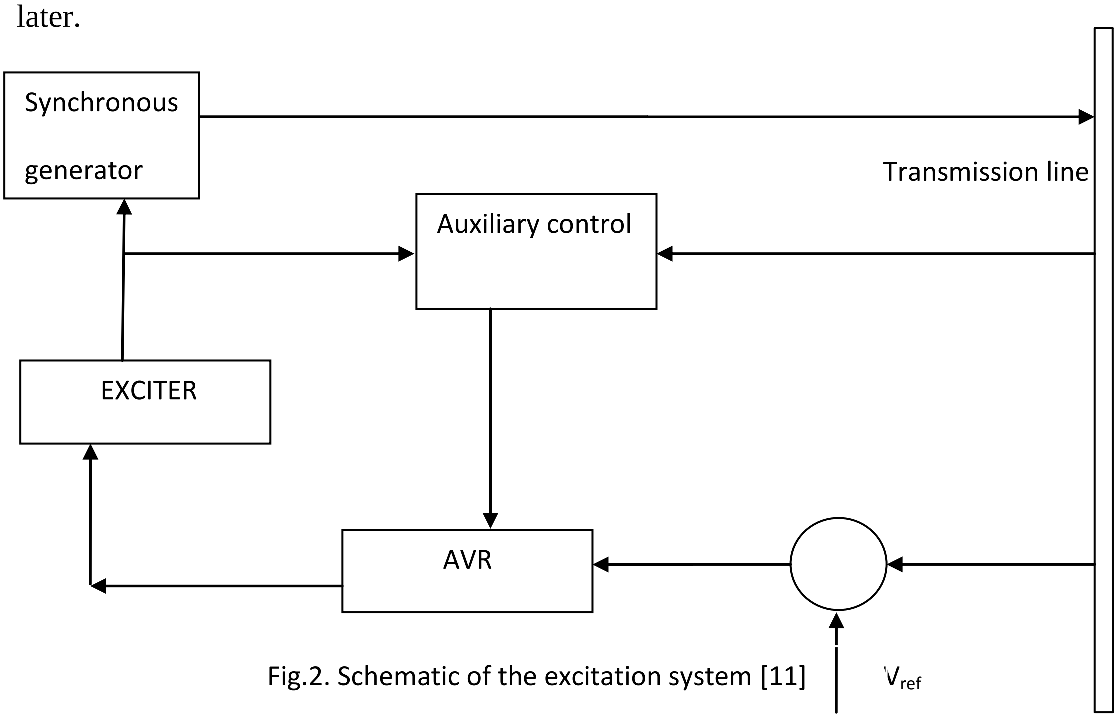


Fig.2. Schematic of the excitation system [11]

CHAPTER-3

POWER SYSTEM STABILIZER: AN INTRODUCTION

STABILITY ISSUES AND THE PSS:-

Traditionally the excitation system regulates the generated voltage and there by helps control the system voltage. The **automatic voltage regulators** (AVR) are found extremely suitable (in comparison to 'ammortisseur winding' and 'governor controls') for the regulation of generated voltage through excitation control. But extensive use of AVR has detrimental effect on the dynamic stability or steady state stability of the power system as oscillations of low frequencies (typically in the range of 0.2 to 3 Hz) persist in the power system for a long period and sometimes affect the power transfer capabilities of the system [4]. The **power system stabilizers** (PSS) were developed to aid in damping these oscillations by modulation of excitation system and by this supplement stability to the system [5]. The basic operation of PSS is to apply a signal to the excitation system that creates damping torque which is in phase with the rotor oscillations.

DESIGN CONSIDERATIONS:-

Although the main objective of PSS is to damp out oscillations it can have strong effect on power system transient stability. As PSS damps oscillations by regulating generator field voltage it results in swing of VAR output [1]. So the PSS gain is chosen carefully so that the resultant gain margin of Volt/VAR swing should be acceptable. To reduce this swing the time constant of the 'Wash-Out Filter 'can be adjusted to allow the frequency shaping of the input signal [5]. Again a control enhancement may be needed during the loading/un-loading or loss of generation when large fluctuations in the frequency and speed may act through the PSS and drive the system towards instability. A modified limit logic will allow these limits to be minimized while ensuring the damping action of PSS for all other system events. Another aspect of PSS which needs attention is possible interaction with other controls which may be part of the excitation system or external system such as HVDC, SVC, TCSC, FACTS. Apart from the low frequency oscillations the input to PSS also contains high frequency turbine-generator oscillations which should be taken into account for the PSS design. So emphasis should be on the study of potential of PSS-torsional interaction and verify the conclusion before commission of PSS [5].

PSS INPUT SIGNALS:-

Till date numerous PSS designs have been suggested. Using various input parameters such as speed, electrical power, rotor frequency several PSS models have been designed. Among those some are depicted below.

SPEED AS INPUT: - A power system stabilizer utilizing shaft speed as an input must compensate for the lags in the transfer function to produce a component of torque in phase with speed changes so as to increase damping of the rotor oscillations.

POWER AS INPUT: - The use of accelerating power as an input signal to the power system stabilizer has received considerable attention due to its low level torsional interaction. By utilising heavily filtered speed signal the effects of mechanical power changes can be minimized. The power as input is mostly suitable for closed loop characteristic of electrical power feedback.

FREQUENCY AS INPUT:- The sensitivity of the frequency signal to the rotor input increases in comparison to speed as input as the external transmission system becomes weaker which tend to offset the reduction in gain from stabilizer output to electrical torque ,that is apparent from the input signal sensitivity factor concept.

.

CHAPTER-4

METHODS OF PSS DESIGN: A REVIEW

In this chapter we shall design and review different aspects and methods of PSS design, its advantages, disadvantages and uses in field.

First, we discuss conventional methods of PSS design and then move onto more advanced methods and recent developments.

The schematic below represents different methods of PSS design:-

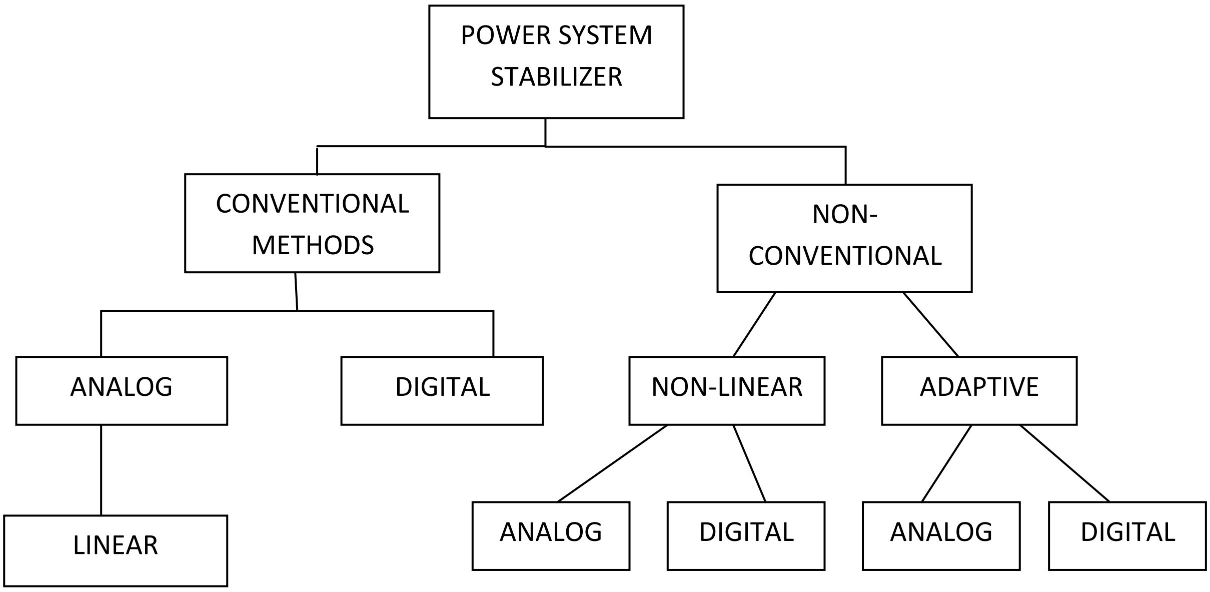


Fig.3. Methods of PSS design

We will mainly focus on analog methods of PSS design which can be further divided into linear and non-linear methods.

The linear methods are:-

1. Pole-placement method: Controllers designed using simultaneous stabilization design have fixed gain constant to adaptive controllers. The root locus technique can be utilized after designing gains separately to adjust the gains by which only dominant modes are selected. In a more efficient manner the pole-placement design was proposed in which participation factor were used to determine size and number of stabilizers in a multi machine system [8] [7].

2. Pole-shifting method: - By this method system input-output relationship are continuously estimated form the measured inputs and outputs and the gain setting of the self-tuning PID stabilizer was adjusted in addition to this the real part of the complex open loop poles can be shifted to any desired location [8] .

3. Linear Quadratic Regulation: - This is proposed using differential geometric linearization approach [8]. This stabilizer used information at the secondary bus of the step-up transformer as the input signal to the internal generator bus and the secondary bus is defined as the reference bus in place of an infinite bus.

4. Eigen value Sensitivity Analysis: - Based on second order Eigen-sensitivities an objective function can be utilized to carry out the co-ordination between the power system stabilizer and FACTS device stabilizer. The objective function can be solved by two methods the Levenberg-Marquardt method and a genetic algorithm in face of various operating conditions [8] [15].

5. Quantitative Feedback theory: - By simply retuning the PSS the conventional stabilizer performance can be extended to wide range of operating and system conditions. The parametric uncertainty can be handled using the Quantitative feedback Theory [8] [16].

6. Sliding Mode control: - Due to the inexact cancellation of non-linear terms the exact input output linearization is difficult. The sliding mode control makes the control design robust. The linearized system in controllable canonical form can be controlled by the SMC method. The control objective is to choose the control signal to make the output track the desired output [8] [17].

7. Reduced Order Model: -Through aggregation and perturbation reduced order model can be obtained but as it is based on open loop plant matrix only the results cannot be accurate.

But with suitable analytical tools reduced order model can be optimized to obtain state variables those are physically realizable and can be implemented with simple hard-wares [8] [18].

8. H2 Control: - Application of H2 optimal adaptive control can be utilized for disturbance attenuation in the sense of H2 norm for nonlinear systems and can be successful for the control of non-linear systems like synchronous generators [8].

The Non-linear methods are:-

1. Adaptive control:-Several adaptive methods have been suggested like Adaptive Automatic Method, Heuristic Dynamic programming. In adaptive automatic method the lack of adaptability of the PSS to the system operating changes can be overcome. Heuristic Dynamic programming combines the concepts of dynamic programming and reinforcement learning in the design of non-linear optimal PSS [8].

2. Genetic Algorithm: - Genetic algorithm is independent of complexity of performance indices and suffices to specify the objective function and to place the finite bounds on the optimized parameters. As a result it has been used either to simultaneously tune multiple controllers in different operating conditions or to enhance the power system stability via PSS and SVC based stabilizer when used independently and through coordinated applications [8].

3. Particle Swarm Optimization:- Unlike other heuristic techniques ,PSO has characteristics of simple concept, easy implementation, computationally efficient , and has a flexible and well balanced mechanism to enhance the local and global exploration abilities [8].

4. Fuzzy Logic: - These controllers are model-free controllers. They do not require an exact mathematical model of the control system. Several papers have been suggested for the systematic development of the PSS using this method [19] [22].

5. Neural Network: - Extremely fast processing facility and the ability to realize complicated nonlinear mapping from the input space to the outer space has put forward the Neural Network. The work on the application of neural networks to the PSS design includes online tuning of conventional PSS parameters, the implementation of inverse mode control, direct control, and indirect adaptive control [19] [22].

6. Tabu Search: - By using Tabu Search the computation of sensitivity factors and Eigen vectors can be avoided to design a PSS for multi machine systems.

7. Simulated Annealing: - It is derivative free optimization algorithm and to evaluate objective function no sensitivity analysis is required [8].

8. Lyapunov Method: - With the properly chosen control gains the Lyapunov Method shows that the system is exponentially stable.

9. Dissipative Method:-A framework based on the dissipative method concept can be used to design PSS which is based on the concept of viewing the role of PSS as one of dissipating rotor energy and to quantify energy dissipation using the system theory notation of passivity [8].

10. Gain Scheduling Method: - Due to the difficulty of obtaining a fixed set of feedback gains design of optimum gain scheduling PSS is proposed to give satisfactory performance over wide range of operation. As time delay can make a control system to have less damping and eventually result in loss of synchronism, a centralized wide area control design using system wide has been investigated to enhance large interconnected power system dynamic performance. A gain scheduling model was proposed to accommodate the time delay [8].

11. Phasor Measurement: -An architecture using multi-site power system control using wide area information provided by GPS based phasor measurement units can give a step wise development path for the global control of power system [8].

CHAPTER-5

THE ALTERNATOR STATE SPACE MODEL

The model which was used for the design of the final PSS consists of a “single-machine infinite bus". It consists of a single generator and delivers electrical power P_e to the infinite bus. It has been modelled taking into consideration sub transient effects.

The below schematic diagram shows the model:-

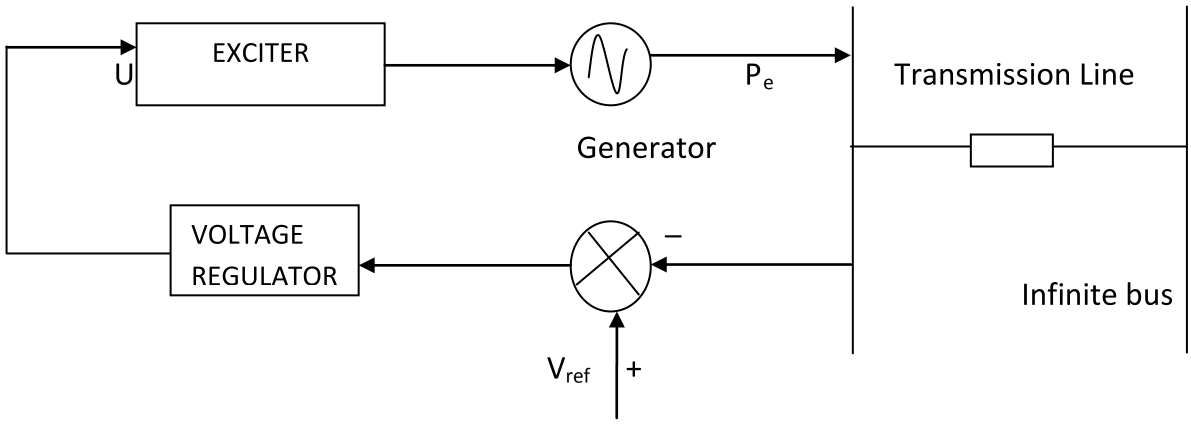


Fig.4. Excitation system control model [1]

The voltage regulator controls the input u to the excitation system which provides the field voltage so as to maintain the generator terminal voltage V_{term} at a desired value V_{ref}. We consider the state –space representation of the above system [1] as follows:-

There are 7 state variables, 1 input variable and 3 output variables y.

Where state variables $x = [\delta\ \ \omega\ \ E_q'\ \ \psi_d\ \ E'_d\ \ \psi_q\ \ V_r\ ]^T$

Output variables $y = [V_{term}\ \omega\ \ P_e]^T$

Input variable $u = V_{ref}$

Where, $\delta =$ rotor angle in radian.

$\omega =$ angular frequency in radian/sec.

ψ_d, $E_d' =$ direct axis flux and field.

Ψ_q, $E'_q =$ quadrature axis flux and field

$V_{term} =$ terminal voltage

P_e= Power delivered to the infinite bus.

The state eqn are:-

$\Delta x' = A\Delta x + B\Delta u$;

$\Delta y = C\Delta x$

Here, the matrices A, B depends on a wide range of system parameters and operating conditions [1].

A=

$$\begin{bmatrix} 0 & 377.0 & 0 & 0 & 0 & 0 & 0 \\ -0.246 & -0.156 & -0.137 & -0.123 & -0.0124 & -0.0546 & 0 \\ 0.109 & 0.262 & -2.17 & 2.30 & -0.171 & -0.0753 & 1.27 \\ -4.58 & 0 & 30.0 & -34.3 & 0 & 0 & 0 \\ -0.161 & 0 & 0 & 0 & -8.44 & 6.33 & 0 \\ -1.70 & 0 & 0 & 0 & 15.2 & -21.5 & 0 \\ -33.9 & -23.1 & 6.86 & -59.5 & 1.5 & 6.63 & -114 \end{bmatrix}$$

$B = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 16.4]^T$

$$C=\begin{bmatrix} -0.123 & 1.05 & 0.230 & 0.207 & -0.105 & -0.460 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1.42 & 0.9 & 0.787 & 0.708 & 0.0713 & 0.314 & 0 \end{bmatrix}$$

CHAPTER-6

DESIGN OF THE PSS: THE EXCITATION SYSTEM MODEL

The SIMULINK™ model of the single machine excitation system is given below:

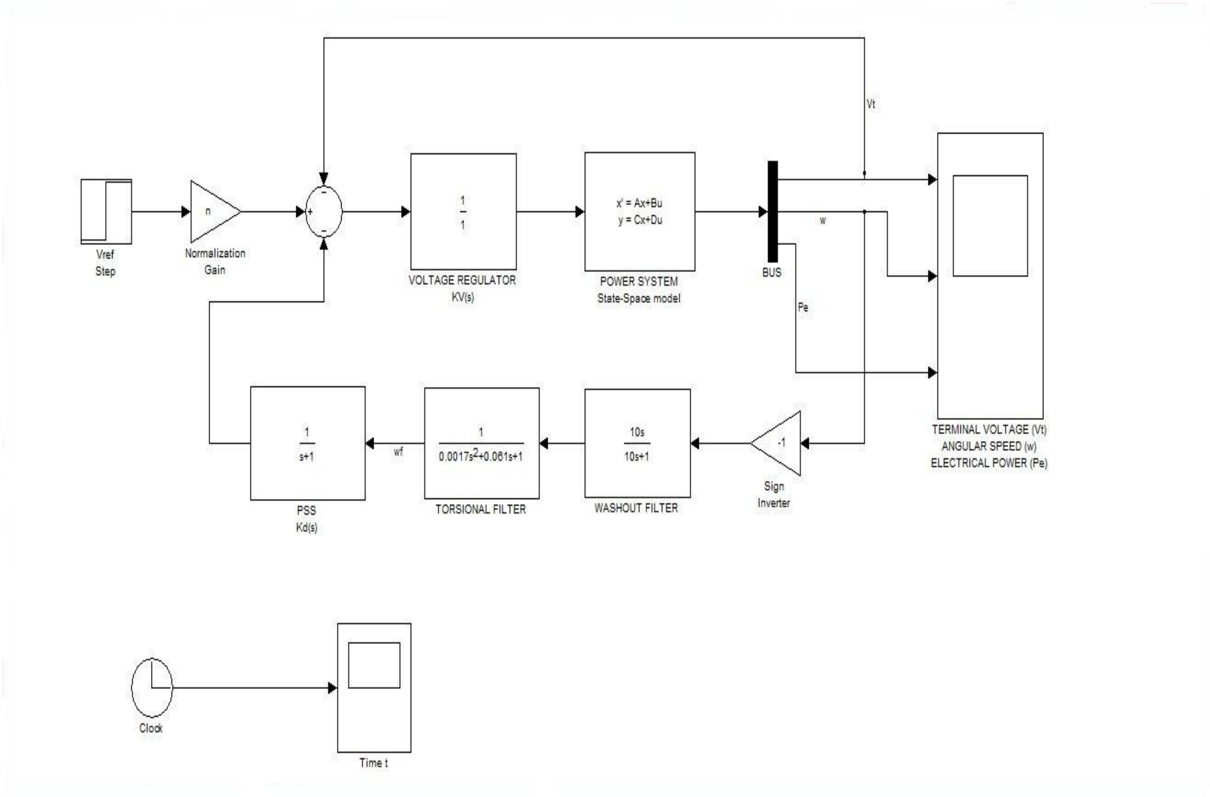


Fig.5. SIMULINK model of the 1-machine infinite bus [1]

The above SIMULINK model adapted from [1] was used by us to design an optimum Voltage regulator and the "power system stabilizer" using various design methods that we discuss later.

<u>The different parts of the model are discussed as follows:</u>

1. Vref- the reference voltage signal is a step voltage of 0.1 V. the final aim is to maintain the voltage at a constant level without oscillations.

2. Voltage regulator (AVR) - The excitation of the alternator is varied by varying the main exciter output voltage which is varied by the AVR. The actual AVR contains:

- Power magnetic amplifier
- Voltage correctors

- Bias circuit
- Feedback circuit
- Matching circuit etc.

For our simulations, we have utilized a

1. Proportional VR Kv(s) =K_p (10, 20, 30…)
2. PI VR $K_v(s) = k_{pi} = k_p(1+k_i/s)$
3. Lag VR (compensator or filter)
4. Observer based controller VR (5^{th} order and 1^{st} order)

The effect of different types of control and different values of kp and ki on the AVR and the overall power system has been shown in the simulated results.

3. POWER SYSTEM MODEL: -

As described in the previous section, we use a state space model [1] of the power system having 7 state variables, 1input and 3 output variables.The details of the model are given in the previous chapter.

4. WASHOUT FILTER:-

The output w is fed back through a sign inverter to the washout filter which is a high pass filter having a dc gain of 0. This is provided to cut-out the PSS path when the steady state [1]. In our simulation we take the filter as a transfer function model of

$F(s) = (10s/10s+1)$

5. TORSIONAL FILTER: - This block filters out the high frequency oscillatins due to the torsional interactions of the alternator. In our simulation, we take the transfer function model of this filter as $Tor(s) = (1/1+0.06s+0.0017s^2)$ [1].

6. PSS: -

This is the main part of our design problem. The power system stabilizer takes input from the filter outputs of the rotor speed variables and gives a stable output to the voltage regulator. The pss acts as a damper to the oscillation of the synchronous machine rotor due to unstable operating condition. It does this task by taking rotor speed as input (with the swings in the rotor) and feeding a stabilized output to the voltage regulator. A PSS is tuned by several methods to provide optimal damping for a stable operation. They are tuned around a steady state operating point which we shall try to design.

CHAPTER-7

DESIGN OF AVR AND PSS USING COVENTIONAL METHODS

In our model for the control of the single-machine excitation system, we have two aspects of design namely:

a) Voltage regulator (AVR)
b) Power system stabilizer (PSS)

The power system stabilizer design performed by us has been grouped under three heads:

1. Root-Locus approach (Lead-Lead compensator)
2. Frequency response approach (Lead-Lead compensator)
3. State-Space approach (Observer based Controllers)

We now discuss each method in details; the steps involved, the results obtained and finally, give a brief review on the merits and demerits of each method.

1. ROOT LOCUS METHOD:

The root locus design method of the PSS involves the following steps:

a) ***Design of the AVR****:* We take a PI controller as the voltage regulator having the transfer function, V(s) =$K_p\left(1+\frac{K_i}{s}\right)$. The constants Kp and Ki are to be chosen such that the design specs**:** $\mathbf{t_r < 0.5}$ sec and $\mathbf{M_p < 10\%}$ are satisfied. For this, we make a table of different Kr and Kp values and their corresponding Tr and Mp values and choose the appropriate value as given in [**Appendix-1.4, 1.5**].
We get $\mathbf{K_p=35}$ **and** $\mathbf{K_i=0.6}$ which satisfy the above specifications.

The output V_{term} for different values of Ki is plotted below in fig.6:

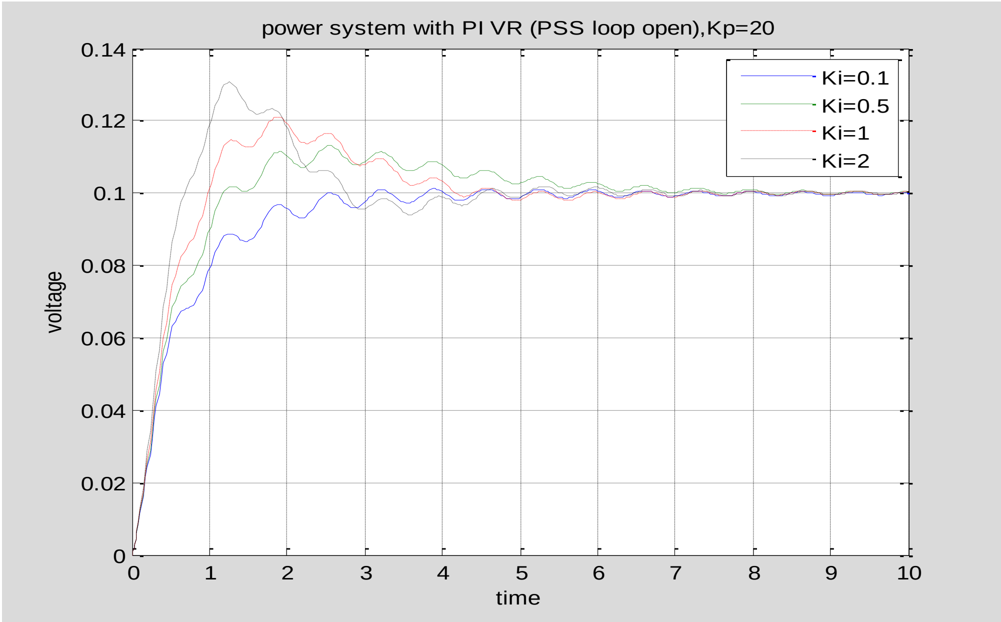


Fig.6. Step response for regulation loop for different Ki values.

b) ***Design of PSS:*** We close the VR loop with the above Kp and Ki and simulate the system response for a step input. The above plot shows that the steady state error =0. Hence, the system is able to follow the step input by introduction of the AVR; but due to the PI controller of the AVR, the swing mode (dominant complex poles) becomes unstable and oscillations are introduced in the output V_{term}. Now, to reduce the oscillations, we have to introduce a feedback loop involving the swing in rotor angular speed ($\Delta\omega$) as input to the PSS loop.

First we analyse the root locus of the PSS loop from u to w_f:

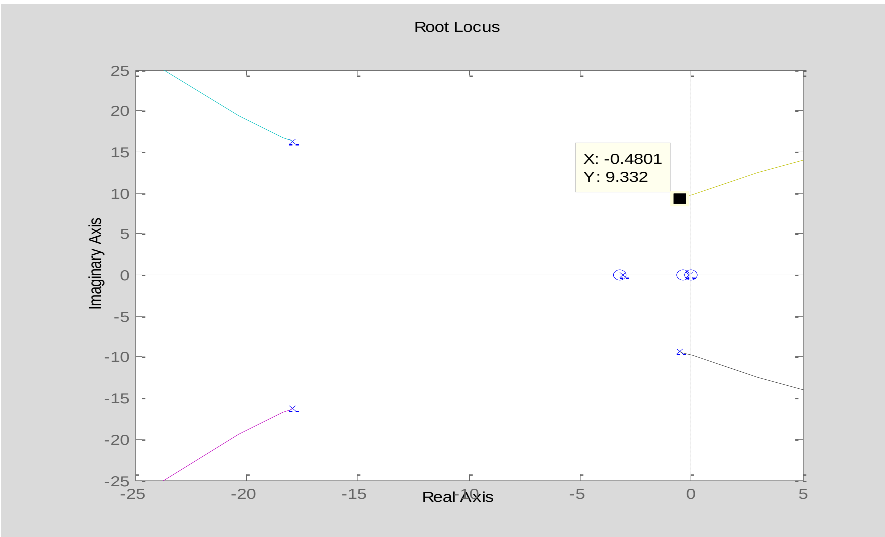


Fig7. Root locus of PSS loop showing the dominant complex poles.

We see that the dominant complex poles are at **(-0.4801+9.332i, -0.4801-9.332i).** Next, we find the angle of departure (Φ_p) from the pole using MATLAB. We get **Φp = 43.28**. Based on this angle we design the lead-lead compensator :

P(s) = $K\left[K_\alpha\left(\frac{s+z}{s+p}\right)\right]\left[K_\alpha\left(\frac{s+z}{s+p}\right)\right]$ such that $\mathbf{\Phi_p=180^o}$ for perfect damping. Hence we have to add angle of 137° which cannot be done using a single lead compensator. So we use two lead compensators in series each adding an angle of 68.5°. K is chosen from the root locus plot of the final PSS loop such that damping ratio ζ > **15%.**

After the design we find that:

z= 3.5 p= 24 Kα= 13.8 K= 0.4

The final lead-lead compensator is given by:

$$\mathbf{P(s) = 0.4\left[13.8\left(\frac{s+3.5}{s+24}\right)\right]\left[13.8\left(\frac{s+3.5}{s+24}\right)\right]}$$

Next, we implement this PSS and close the loop and simulate the response. The root-locus plot of the final PSS loop and the comparison of responses are given below:

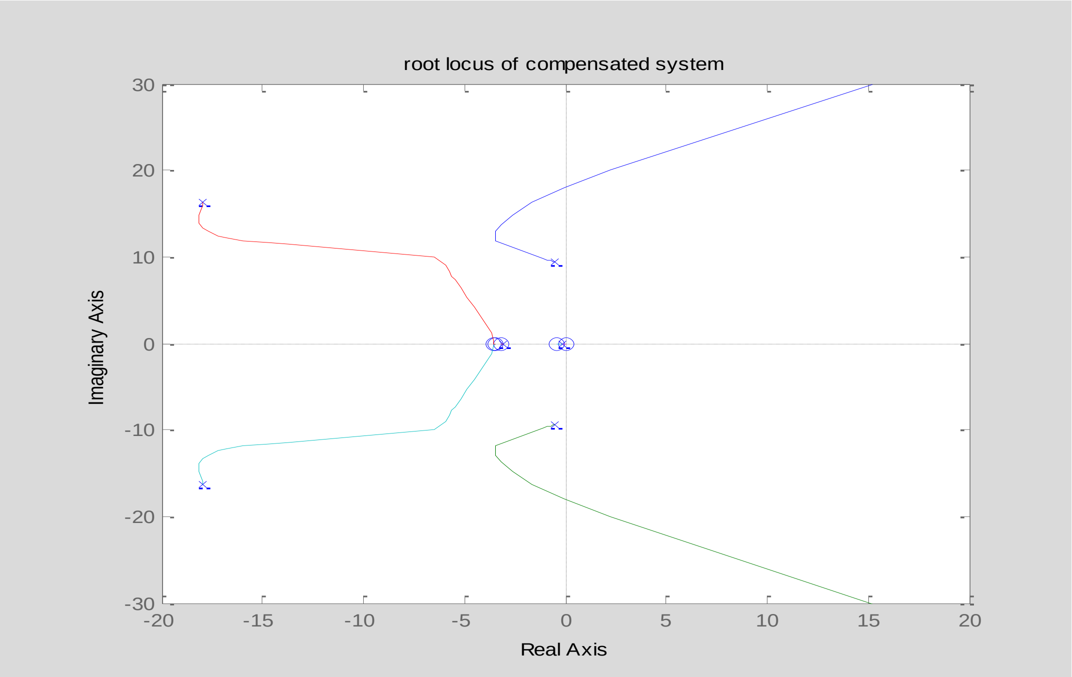


Fig.8. Root-locus of the final PSS loop showing $\Phi_p \approx 180^o$ for dominant poles

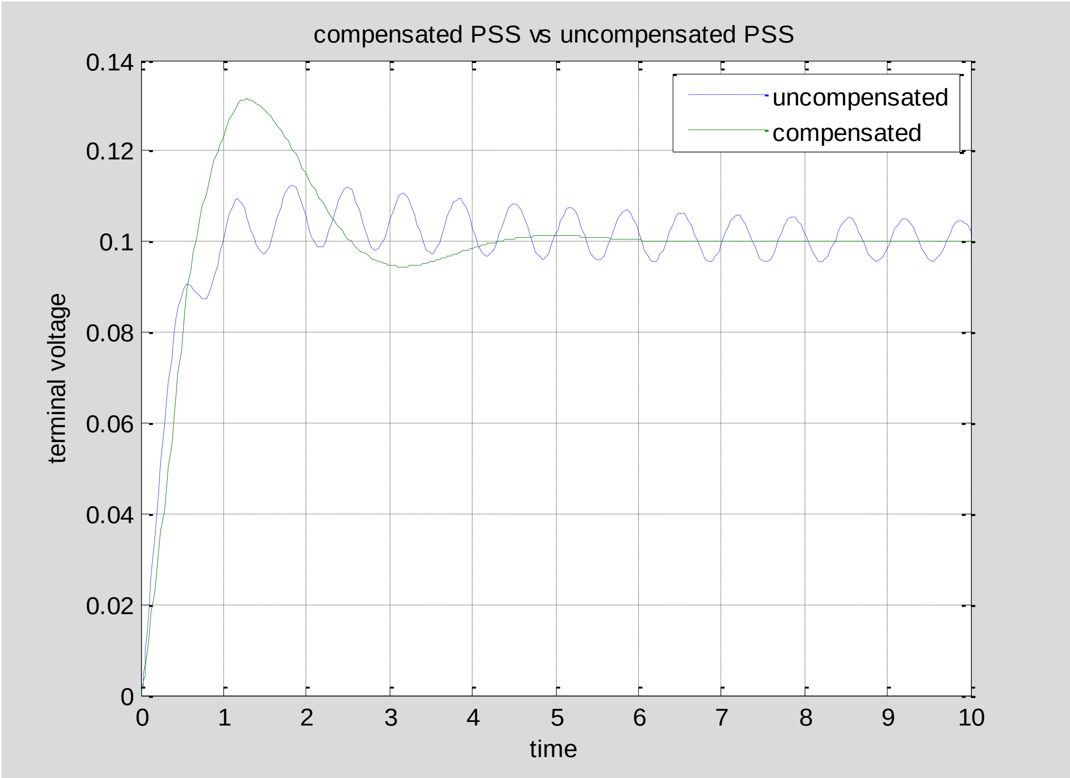


Fig.9. Comparison of step response of uncompensated and compensated systems

2. FREQUENCY RESPONSE METHOD:

The frequency response design method involves the use of bode-diagrams to measure the phase and gain margin of the system and compensating the phase by using lag controller for AVR and lead controller for PSS. The design details are as below:

a) ***Design of the AVR:*** First, we plot and analyse the bode plot of the open-loop Power system. From this, we find that:

Gain margin G_m = 35dB **Phase margin P_m = inf.** **DC gain= -2.57dB (0.74)**

The design specs [1] require the *DC gain > 200 (=46dB) and phase margin > 80º.*

Thus the required gain K_c=10^((200+0.74)/20) = 269. Now, for the phase margin to be >80º, the new gain crossover frequency = 5rad/sec.

To give the required phase lag to the system at this crossover frequency, we take a lag-compensator as the AVR, having transfer function:

V(s) = $K_l\left[\frac{s+z}{s+p}\right]$, where $K_l=\frac{K_c}{\beta}$, $p=\frac{z}{\beta}$

Now, the lag required at 5rad/sec is -18dB. Hence, $\mathbf{20log\frac{1}{\beta} = -18}$**, i.e. β=8.**

We choose the corner frequency $\frac{1}{T}$ = 0.1 to make the system faster. So, z = 0.1. Hence, p=0.1/8 = 0.0125, K_l= 269/8 = 35. Thus the final AVR is:

$\mathbf{V(s) = 35\left[\frac{s+0.1}{s+0.0125}\right].}$

The frequency response of the uncompensated and the compensated system are shown below:

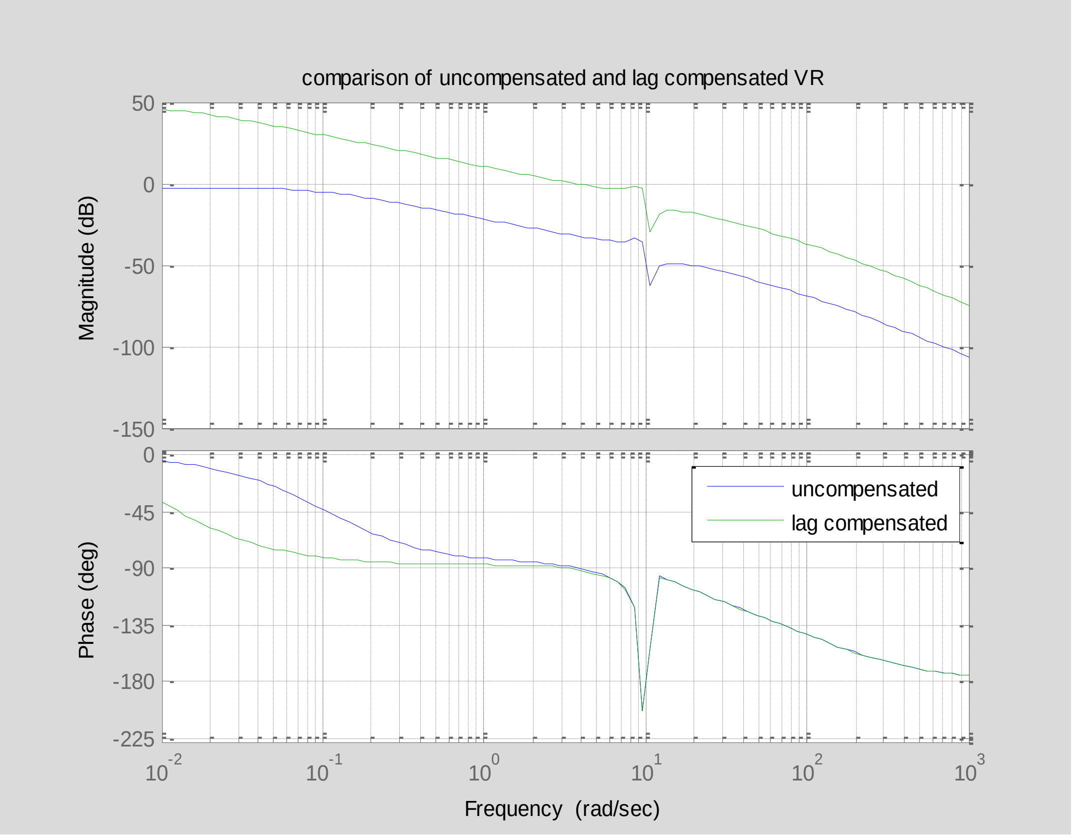


Fig.10. Comparison of frequency response with and without VR loop

Next, we implement this AVR in the SIMULINK model and get the step-response:

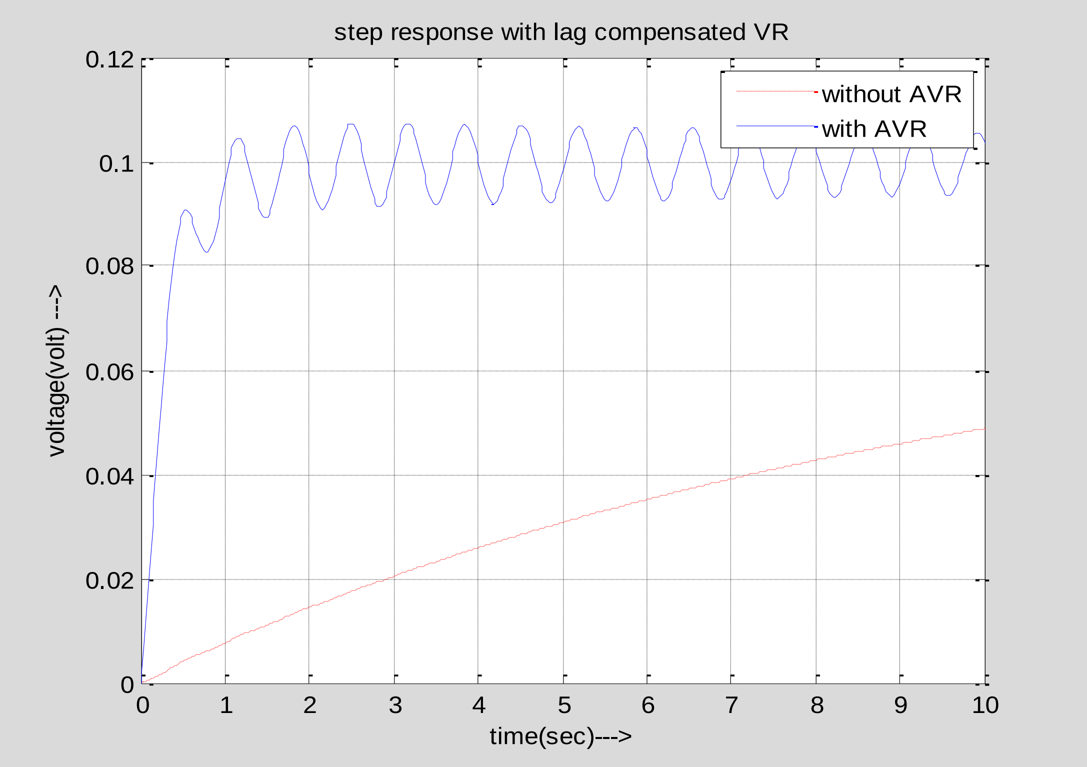


Fig.11. Step response of the lag compensated VR

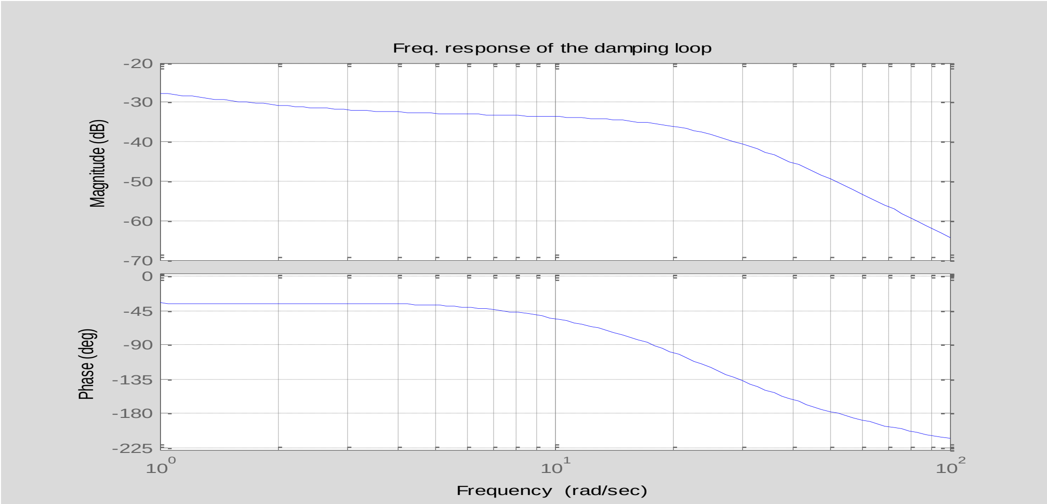
Rise time t_r= 0.48sec. **Maximum overshoot M_p= 7.36%**

b) ***Design of the PSS:*** As in case of the previous design method, we find that the introduction of the voltage regulator eliminates the steady state error and makes the system much faster. But it also introduces low frequency oscillations in the system. Hence we have to design the PSS loop taking input as the perturbation in rotor angular speed ($\Delta\omega$).

First, we generate the state-space model from V_{ref} to ω with the regulation loop closed. As given in [1], figure 8, we isolate the path *Q(s)= effect of speed on electric torque due to machine dynamics* and find A_ω matrix from the main matrix A.

The resulting state-space model has input $\Delta\omega$ and output τ (balancing torque). Thus we get A_{33}(5*5 matrix) , a_{32} (5*1 vector), a_{23}(1*5 matrix). ***[see Appendix1.2]***

We convert this state space model to transfer function and connect Q(s) to the torsional and washout filters to get F(s). Then we plot and analyze the frequency response of F(s) from 1rad/sec to 100 rad/sec.



From the above Fig.12, we find that:

Phase at 2rad/sec = -37º **Phase at 20 rad/sec = -105º**

As per the design specs [1], we have to increase this phase at 2 to 20 rad/sec from the above values to approximately 0º to -15º, such that the feedback loop will add pure damping to the dominant poles. Thus we require a lead compensator of the form:

P(s)= $K\left[K_\alpha\left(\frac{s+z}{s+p}\right)\right]\left[K_\alpha\left(\frac{s+z}{s+p}\right)\right]$ where $K_\alpha = \frac{Kc}{\alpha}$

We need an additional phase of:

35º at 2 rad/sec 60º at 12 rad/sec 100º at 20 rad/sec

Hence, maximum phase addition Φ_m is at 20 rad/sec =100º. This is too large for a single lead compensator as shown in figure 13. below:

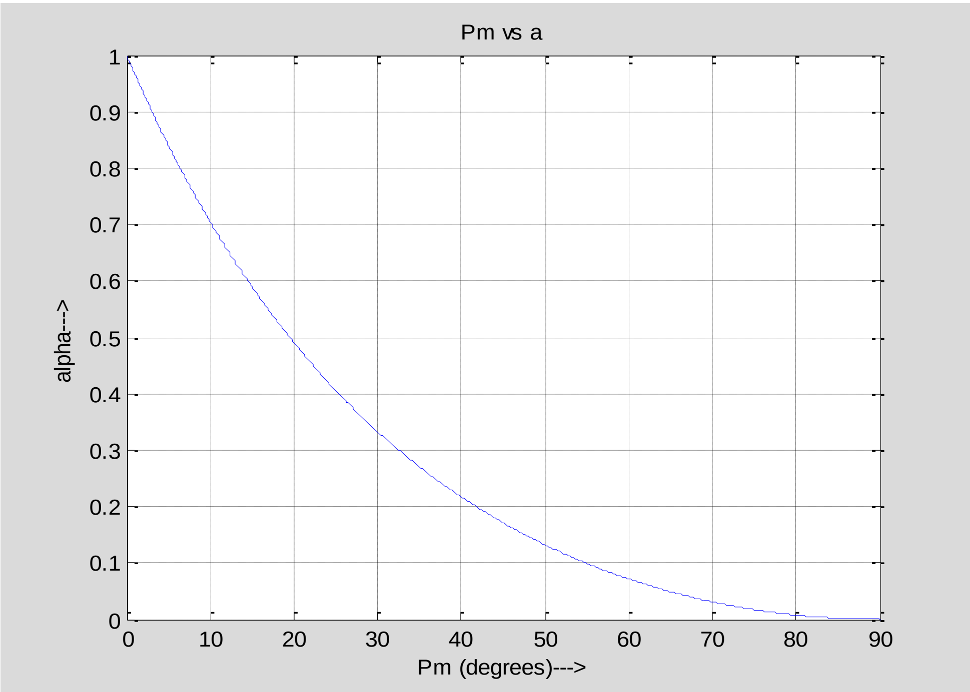


Fig. 13. Maximum phase addition Φ_m vs alpha α

From the above figure, we see that for $\Phi_m > 60^o$, α is too small. Hence we use two identical lead-compensators in series. Thus for each compensator $\Phi_m = 50^o$.

From the relation $\sin \boldsymbol{\Phi_m} = \frac{1-\alpha}{1+\alpha}$, we get **α= 0.1325.** Hence,

$K_\alpha = 1/\alpha = 7.5$ $\quad$ $T = \frac{1}{\sqrt{\alpha}\omega} = 0.137$ $\quad$ $z = \frac{1}{T} = 7.28$ $\quad$ $p = \frac{1}{\alpha T} = 55$

From the root locus plot of the PSS loop we get K for **ζ >15% K=5**. Thus

$$P(s) = 5\left[7.5\left(\frac{s+7.28}{s+55}\right)\right]\left[7.5\left(\frac{s+7.28}{s+55}\right)\right].$$

Then we implement this PSS and close the loop and simulate the resultant model. We find the step response and the rise time and maximum overshoot of the compensated system.

Below fig. 14 shows the root locus plot of the damping loop and fig15. Shows the step-response of the final system:

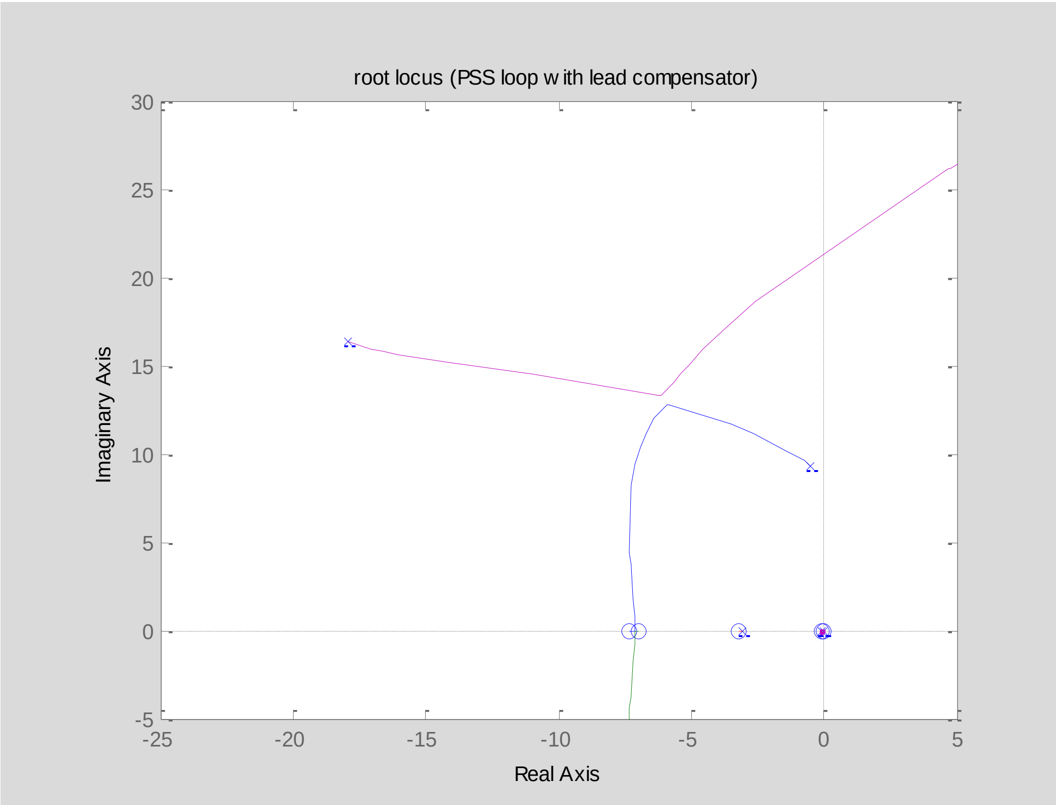


Fig.14. Root locus plot of the PSS loop showing the dominant poles

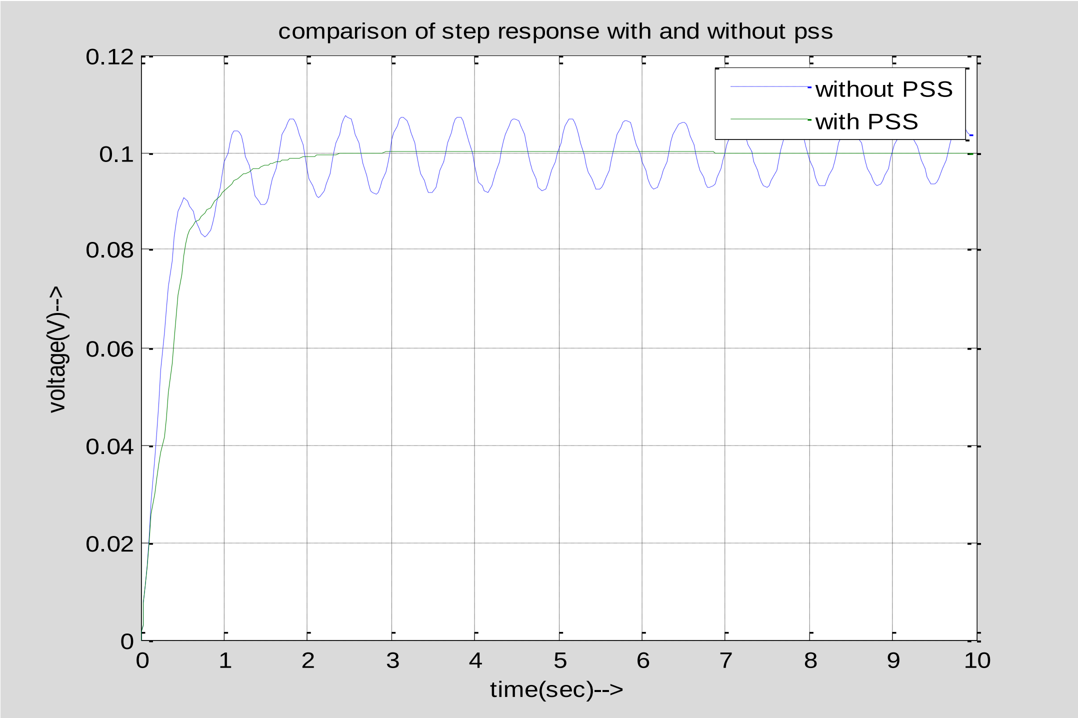


Fig.15. Step response of the final system with and without PSS loop

3. STATE-SPACE METHOD:

The state space design involves designing full state observers using pole placement to measure the states and then designing the controller such that the closed loop poles lie in the desired place. As before, we first design the voltage controller AVR such that the dominant pole is made faster by placing it away from the jω axis. Then, we design the PSS to stabilize the oscillations due to the VR loop by manipulating the swing mode (dominant poles). The details are given below:

a) ***Design of the AVR:*** We first obtain the 1-input 1-output model of the power system as given in [1] from V_{ref} to V_{term}. Hence, we get A_1 (7*7 matrix), B_1 (7*1 vector), C_1 (1*7 matrix), and D_1(1*1) as given in **Appendix-1** in this text**.** We find the open loop poles of this system:

   (-114.33, -35.36, -26.72, -0.48±9.33j, -3.08, -0.1054). Hence the dominant real pole is -0.1054.

   For the controller design, we have to make this dominant pole faster and steady state error zero. We choose the shifted pole at **-4.0+0.0j** and leave the other poles unchanged. Then, using MATLAB, we find the gain matrix K_c for the controller.

K_c= acker(A_1, B_1, modified poles)

Next, we design the full-order observer to measure the states. We choose the observer dominant pole such that it is far from the jw axis, hence it decays very fast. We take it to be **-8.0+0.0j** and leave other poles unchanged. Again, using MATLAB, we find the observer gain matrix $K_{o.}$

K_o= place(A_1', C_1', modified poles)'

Finally we find the state space representation and the transfer function of the above designed observer-controller as:

$A_o = A_1 - (K_o * C_1) - (B_1 * K_c)$

$B_o = K_o$

$C_o = K_c$

$D_o = 0$

We get the 7th order observer-controller as given in **Appendix-1.3** in this text. We then minimize the order of this controller to 1st order by approximate pole-zero cancellations as given below:

Poles of observer-controller	Zeros of observer-controller
-114.22	-114.33
-35.86	-35.36
-26.72	-26.72
-13.13	
-0.6129+9.58j	-0.48+9.33j
-0.6129-9.58j	-0.48-9.33j
-2.41	-3.07

Thus, we are left with a single pole **-13.13**. So, the VR is given by:

$$\mathbf{V(s)} = \frac{\mathbf{480}}{\boldsymbol{s}\mathbf{+13.13}}$$

We show the step response of the system after implementing the 7th order VR and the 1st order VR below in fig.16:

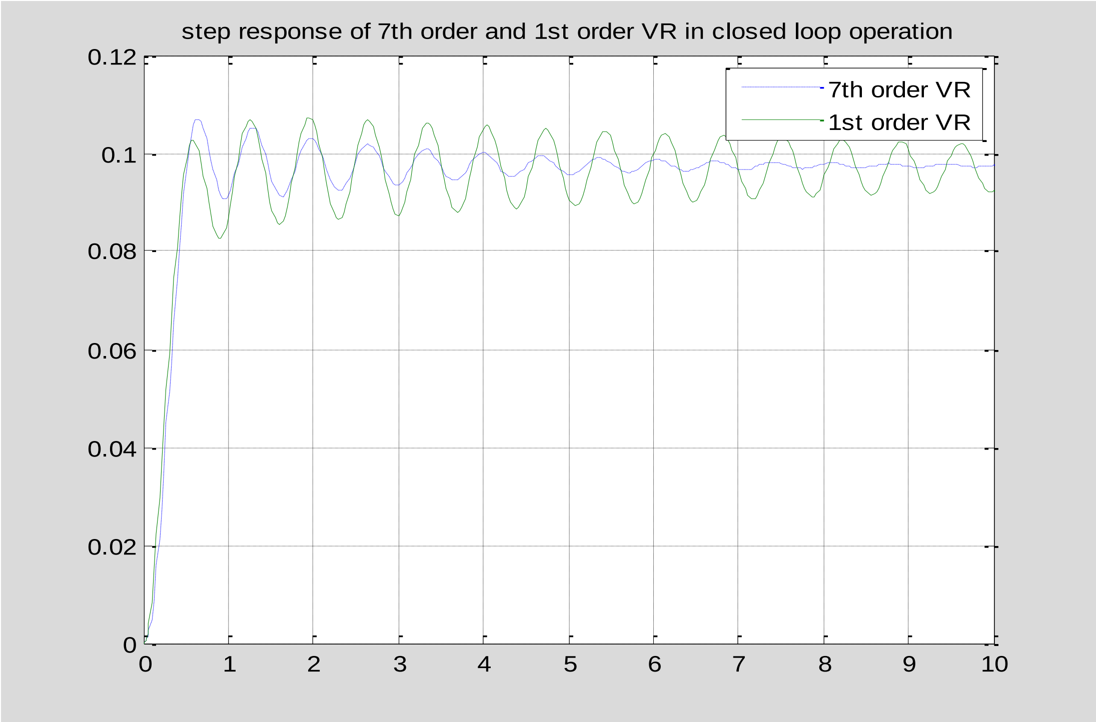


Fig.16. Step response comparison of 7th order and 1st order VR

We find that the step response is identical except that due to minimization of order, oscillations are introduced in the 1st order VR. Hence, we design the damping (PSS) loop to stabilize the system.

b) Design of PSS: As mentioned above, use of the 1st order AVR introduces oscillations in the system. Hence we design the PSS loop.

First we find the 1-input, 1-output model of the system from V_{ref} to $\omega_{f,}$ including the 1st order VR designed previously. This is an 11th order transfer function as given in **Appendix-1** in this text. Thus we get the state space model A_g, B_g, C_g, D_g. From the root locus plot of this system, we find that the dominant complex pole is **at (-0.48 ± 9.33j).**

For the controller design, we have to shift the swing mode to get a faster response.

We shift it to: **(-1.5 ± 9.33j),** leaving all other poles unchanged.

Using MATLAB, we get the controller gain matrix K_c=acker (A_g, B_g, mod_poles).

For the observer design, we choose the poles as (**-4.5 ± 9.33j)** so that it decays faster.

K_o=place (A_g', C_g', poles_obs)'.

Thus we get the 11th order observer-controller as:

$A_o = A_1 - (K_o * C_1) - (B_1 * K_c)$

$B_o = K_o$

$C_o = K_c$

$D_o = 0$

Next, we minimize this PSS from 11th order to 5th order by approximate pole-zero cancellations.

Poles of observer controller	Zeros of observer-controller
-114.34	-114.33
-36.106	-35.4
-20.9+16.3j	-18.01+16.3j
-20.9-16.3j	-18.01-16.3j
-28.61	-193.03
-26.74	-26.72
-5.02+13.7j	
-5.02-13.7j	
-3.62	-3.10
-0.091+0.0325	-0.105
-0.091-0.0325	-0.100

We incorporate these poles and zeros for the 5th order PSS [**Appendix-1.3**] After implementing the PSS, we plot the root locus of the damping loop as below:

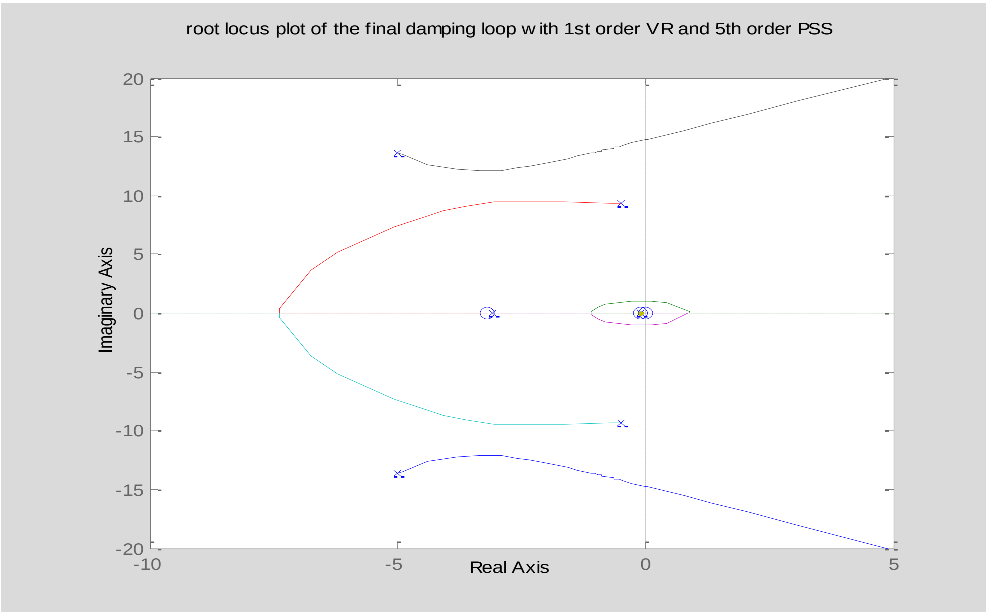


Fig. 17. Root locus plot of the damping (PSS) loop with 5th order PSS implemented

From the previous root locus plot, we find that the 5th order PSS manifests a pure damping at the dominant pole as the angle of departure is approximately= 180º. The gain for ζ=15% is found to be 0.7.

Finally, we implement the above design in the SIMULINK model and find the step response. It is shown in figure 18. below:

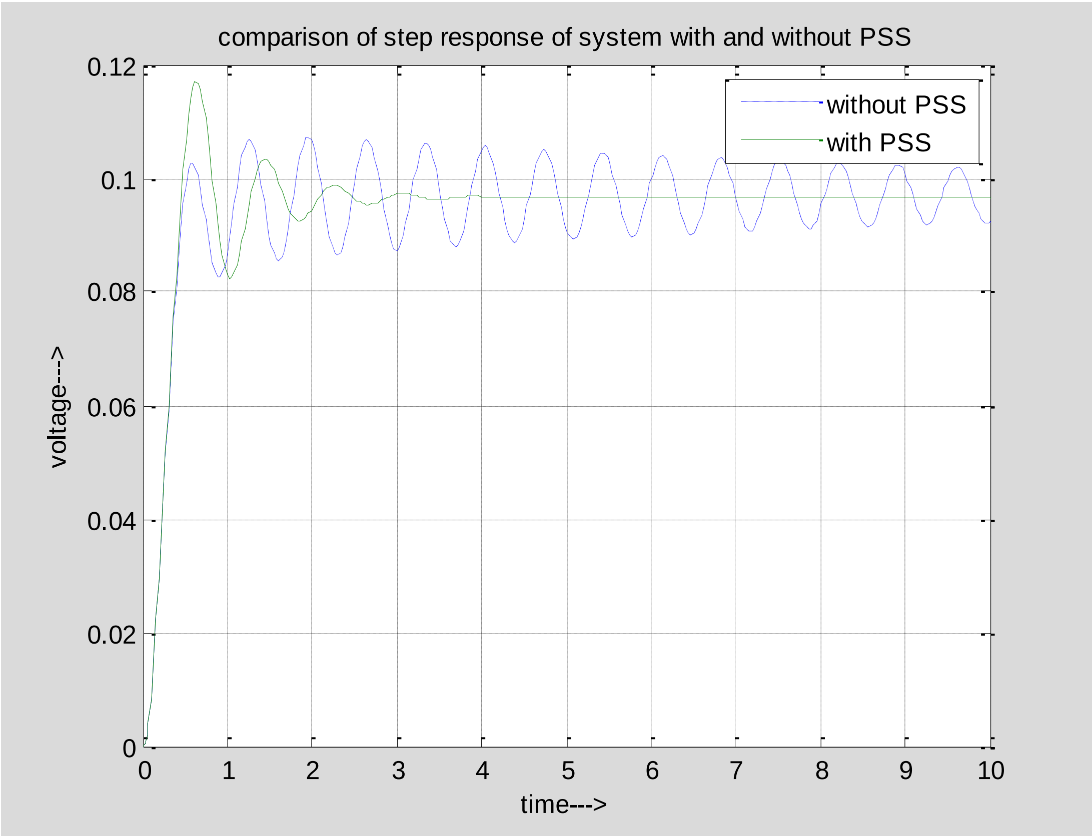


Fig.18. Comparison of the step response of system with and without PSS

We see that the PSS has reduced the oscillations to a large extent and improved the rise time.

CHAPTER-8

REVIEW OF THE CONVENTIONAL DESIGN TECHNIQUES:

Having completed the design of the AVR and the PSS in the above three methods, we now are able to give a brief review on the methods and their merits and demerits.

AVR design:

- We see that the root locus method (method-1) involves designing the voltage regulator as a PI controller by tuning it to achieve a particular value of M_p and t_r. This although simpler is quite arbitrary and is achieved by trial and error.

- The frequency response method (method-2) involves measuring the dc gain and phase margin of the system without the regulation loop; and increasing the dc gain to achieve zero steady state error. Then we adjust the phase margin by a lag compensator to achieve the required M_p and t_r. This method, although less arbitrary than the PI controller, still does not give a direct idea about the time response, i.e. we cannot measure M_p and t_r directly from the phase margin.

- Finally, in the state-space method, we make use of a full-state observer based controller to directly shift the dominant pole of the regulation loop to its left to make it faster and satisfy the specifications. Although this gives an exact controller, the order of the controller is very high and hence is impractical to implement. Thus, it requires reduction of order by approximate pole-zero cancellations. Hence the system becomes slightly oscillatory. Thus, this method is a little cumbersome and time-consuming, and the benefits of the higher order VR is negated by the approximate VR.

PSS design After designing the voltage regulator in any of the above methods, we compare the step response after implementing the regulation loop in each case and find that, although the steady state error e_{ss}, Max. Overshoot M_p and the rise-time T_r conditions are satisfied, the system is not perfectly damped and there are oscillations in it. Hence, we design a feedback loop (PSS) involving the perturbation in rotor velocity $\Delta\omega$ as input which reduces the oscillations.

- For PSS design using root-locus method, we find the dominant complex pole (swing mode) from the root locus plot of the open PSS loop and calculate the angle of departure from this pole. For perfect damping, the angle of departure should be -180. Hence we design a lead-lead compensator to adjust the angle of departure. This method is elegant and simple, yet manual calculation and plotting is required to find the zero and pole of the compensator.

- In the frequency response-method, we have to first decompose the system into its damping component to perform the analysis [1], figure.8. Hence it requires the detailed understanding of the power-system model and its states. Then we manipulate the phase of the system in a frequency range (2rad/sec to 20rad/sec) by a lead-lead compensator to achieve the desired damping effect. Again, this does not give an idea about the actual time-response characteristics and we have to perform a root locus analysis again to find the Gain for the specified damping.

- Finally, in the state-space method, an exact 11^{th} order controller is derived from a full order state-observer. This is highly impractical and expensive, and thus we need to minimize the order of the system by approximate pole-zero cancellations which make it a lengthy and cumbersome process.

CHAPTER-9

DESIGN OF PSS BY ADAPTIVE METHODS

In the preceding chapters the low frequency oscillation problem is dealt with using conventional POWER SYSTEM STABILIZER. As explained earlier these PSS provide the supplementary damping signal to suppress the above mentioned oscillations and increase overall stability of the system. But these conventional PSS use transfer functions of highly linearized models around a particular operating point. So these systems are unable to provide satisfactory operations over wide ranges of operating conditions [22]. To overcome this problem artificial intelligence based approaches has been developed. These include fuzzy logic (FL), neural network (NN), and genetic algorithm (GA). Fuzzy Logic based controller shows great potential to damp out local mode oscillations especially when made adaptive. The adaptability is achieved through tuning with Neural-Network [19].

FUZZY LOGIC:

Fuzzy logic is based on data sets which have non-crisp boundaries. The membership functions map each element of the fuzzy set to a membership grade. Also fuzzy sets are characterized by several linguistic variables. Each linguistic variable has its unique membership function which maps the data accordingly [20]. Fuzzy rules are also provided along with to decide the output of the fuzzy logic based system. A problem associated with this is the parameters associated with the membership function and the fuzzy rule; which broadly depends upon the experience and expertise of the designer [23].

ANFIS:

ANFIS is the abbreviation for the ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM. In it a class of adaptive networks are used which is similar to fuzzy inference system. As the name adaptive suggests it consists of a number of nodes connected through directional links. Each node represents a process unit and the link between them specifies the causal relationship between them. All or some part of these nodes can be made adaptive which means that these node parameters can be varied depending on the output of the nodes. This adaptation depends on the rule table which is designed intuitively by the designer [20].

Adaptive Neuro-Fuzzy design of PSS

In the following chapters a design technique for the off-line training of the power system will be elaborated. The design is divided into two parts. The first one is the design of an identifier for the identification of the plant parameters which cannot be obtained otherwise as the power plants are highly nonlinear systems. The second one is the design of the ANFIS controller which is trained off-line to control the plant outputs $\Delta\omega$ and Δp.

SYSTEM IDENTIFIER

The plant identifier is of immense importance for the determination of the plant parameters in order to successfully tune the PSS. The identifier parameters are estimated on the basis of the error between the estimated generator speed deviation and the actual value. A third order Auto Regression Moving Average (ARMA) model is used for the generating system and the Recursive Least Square (RLS) method with a variable forgetting factor [19] is used to obtain the coefficient vector of the generator system model.

The identifier is a third order ARMA model of the form

$$\nabla\omega(t) = \emptyset^T(t-1)\hat{\theta}(t-1) + e(t)$$

Where $\emptyset^T(t-1) = [-\nabla\mathrm{\omega}(t-1), -\nabla\mathrm{\omega}(t-2), -\nabla\mathrm{\omega}(t-3), \mathrm{u(t-1)}, \mathrm{u(t-2)}, \mathrm{u(t-3)}]$

$\hat{\theta}(t-1) = [\theta_{1,}\theta_2, \theta_3, \theta_4, \theta_5, \theta_6]^T$ is a randomly chosen constant vector and e(t) is the identified error [19].

The co-efficient vector is updated using the following SIMULINK model which consists of the power system model and the special embedded function blocks. In it the delayed inputs both for power and angular velocity variation are obtained from the '**delayed inputs' block**.

The '**rls' block** implements the step

$$e(t) = x(t) - \hat{x}(t)$$

where $\hat{x}(t) = \emptyset^T(t-1)\hat{\theta}(t-1)$

the co-variance matrix is determined by the step which is implemented by the **block 'covar'**

$$P(t) = \frac{[1-K(t)\emptyset^T(t-1)]P(t-1)}{\rho}$$ where ρ is the forgetting factor which in this case is taken as 1.

The gain is determined from the step below which is implemented using the **‘k’ block** in the SIMULINK model

$$K(t) = \frac{P(t-1)\emptyset(t-1)}{(\rho(t)+\emptyset^T(t-1)P(t-1)\emptyset(t-1))}$$

Here also forgetting factor ρ is taken 1.

The parameter θ is updated using the following step implemented by the **block ‘theta’** in Simulink diagram.

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)e(t)$$

The Simulink model is given in the following figure

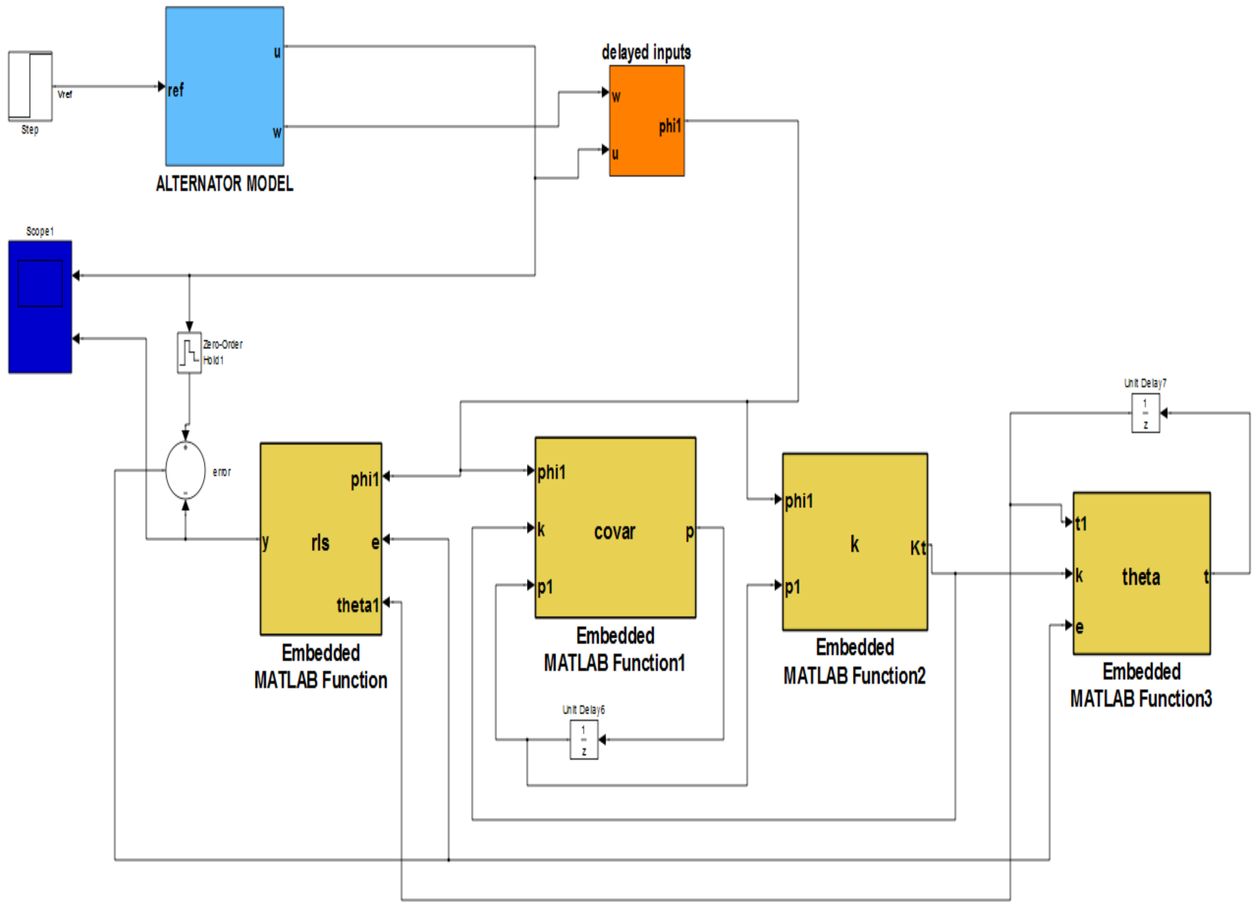


Fig.19. SIMULINK model of the ARMA implementation of the system identifier

The output RLS block is compared with the desired output signal obtained from the PSS as given in figure 20.

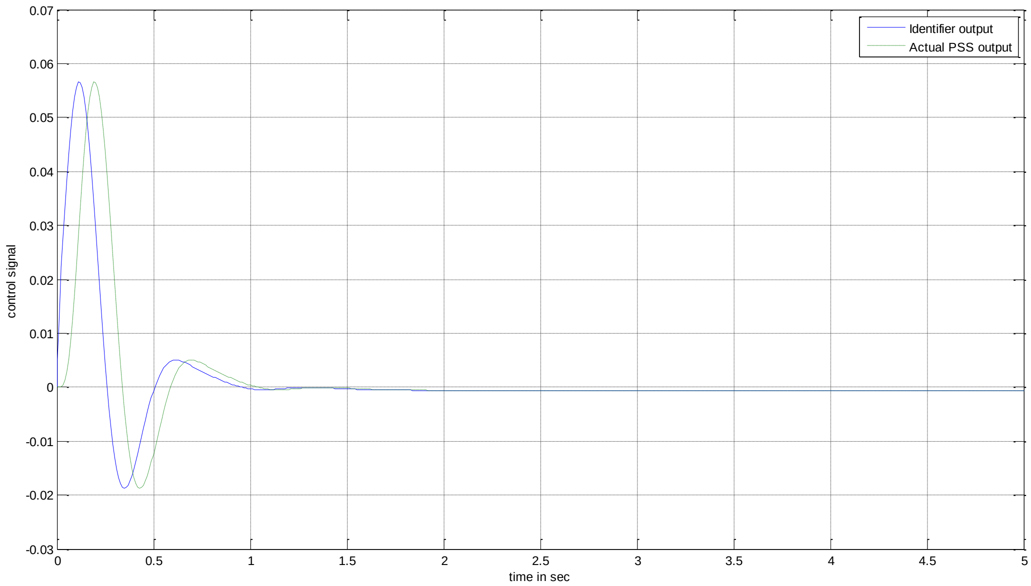


Fig.20. Comparison between ARMA output and actual output

The above figure shows that the identifier output follows the desired PSS output and the error signal reduced to zero subsequently.

PSS DESIGN USING ANFIS

The ANFIS PSS uses a zero order Sugeno type fuzzy controller with 49 rules. The input to the PSS is the speed and electrical power which are obtained from the wash-out filter that is used to eliminate any existing dc offsets. The fuzzy inference system consists of the fuzzification block, rule table block and the sugeno defuzzification block.

For fuzzification Gaussian membership function is used which is of the form

$$\mu_{A_i^j} = \exp(\frac{(x_j - c_i^j)^2}{2\sigma_i^{j^2}})$$

Where x_j is the jth input, A_i^j represents the ith linguistic term related to the jth input and C_i^j, σ_i^j are the centres and the spreads of the membership function related to A_i^j which are adjustable by the neural network block of the ANFIS. Seven linguistic variables are used for each input for the fuzzifiacation.

The fuzzy logic based controller is made adaptive by using feed forward neural-network using a multilevel perceptron. The multilevel perceptron is implemented using the ANFIS-GUI block of MATLAB. The neural network can be trained using either OFFLINE method or ONLINE method. The details are as follows.

OFFLINE ADAPTATION USING ANFIS:

Here we first generate the input-output data pair of the system using the identifier or directly from the model. Then, we use the ANFIS module in MATLAB to generate a fuzzy inference system. Two inputs are used, namely $\Delta\omega$ and ΔP, and a single control output for the feedback. A Sugeno type FIS model is used.

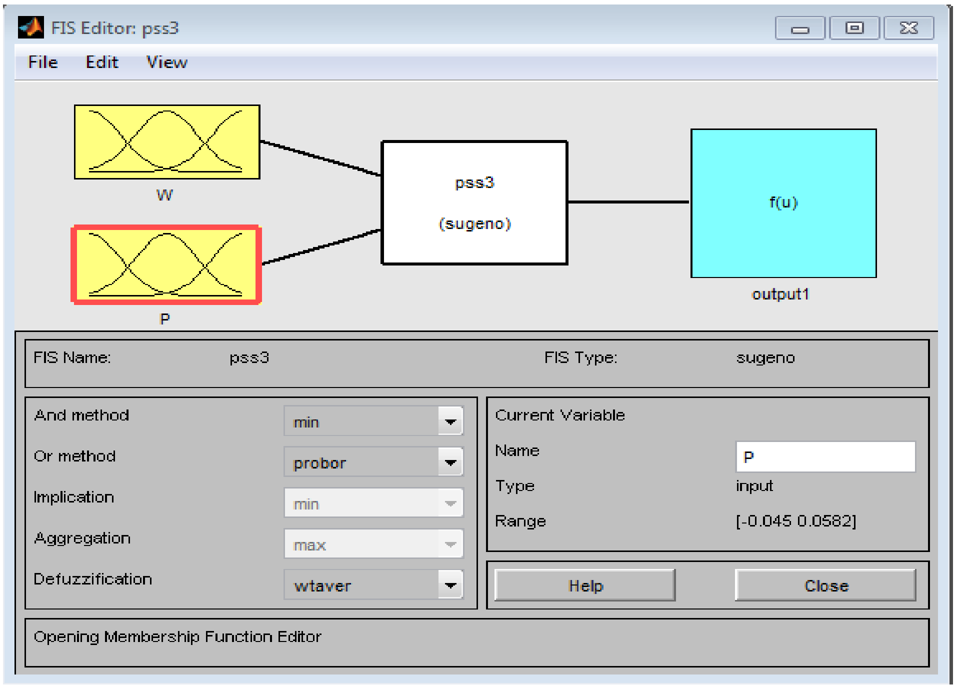


Fig.21. FIS model of the PSS

The membership functions of the inputs are of Gaussian distribution type. We use 7 membership functions for each input to cover the full range of the respective inputs. Thus, we get 49 rules for the output function which is linear relation of the inputs. The initial input parameters are arbitrarily chosen and output parameters are given in table5 (appendix-1). The output is governed by the AND function and thus the rules are generated.

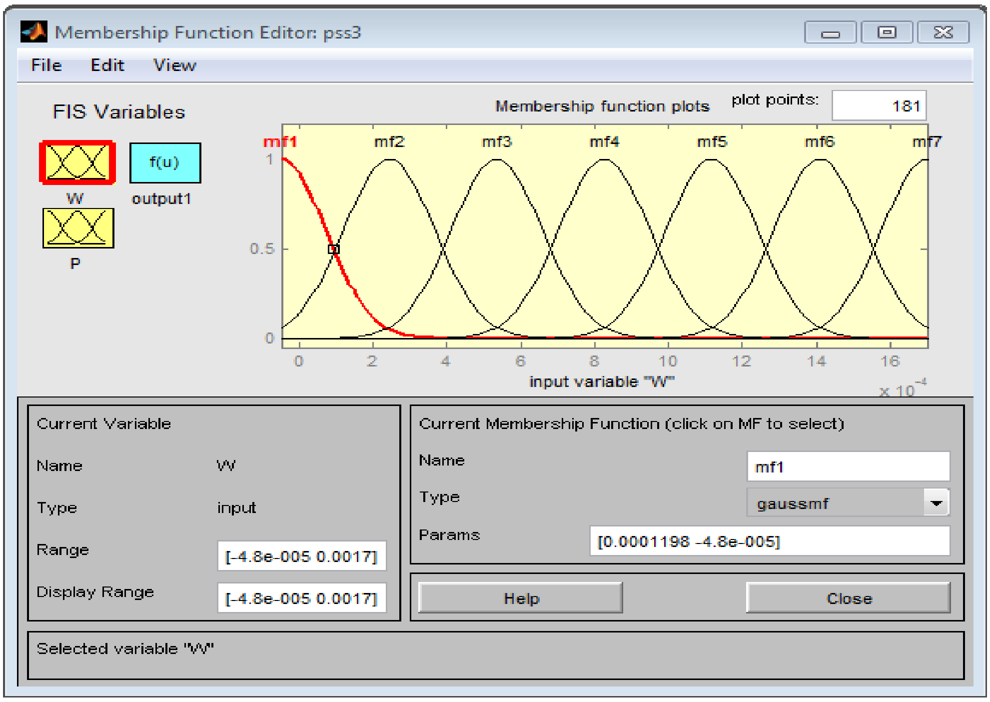


Fig.22. Gaussian membership functions of the inputs

The above generated fis file is opened in the ANFIS GUI for training. We also import the training data which was previously generated to the GUI.The neural network thus has four layers as given below:

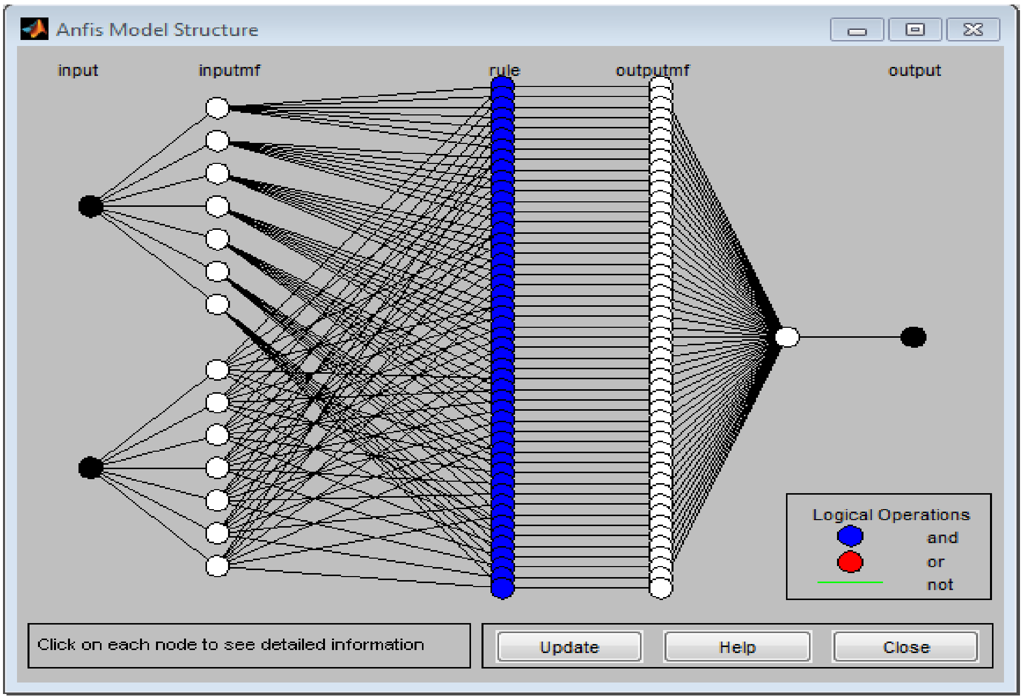


Fig.23. Structure of the Neural Network

The first layer represents the input membership functions (MFs) which is Gaussian. The second layer represents the AND function. The third layer represents the normalized firing strength as given in the sugeno model and, the fourth layer represents the combination of the rules and their weighted average to find the final output using sugeno defuzzification technique.

Now, the training is started using the back-propagation method and the model is trained for 100 epochs for greater reliability. The error is given as below:

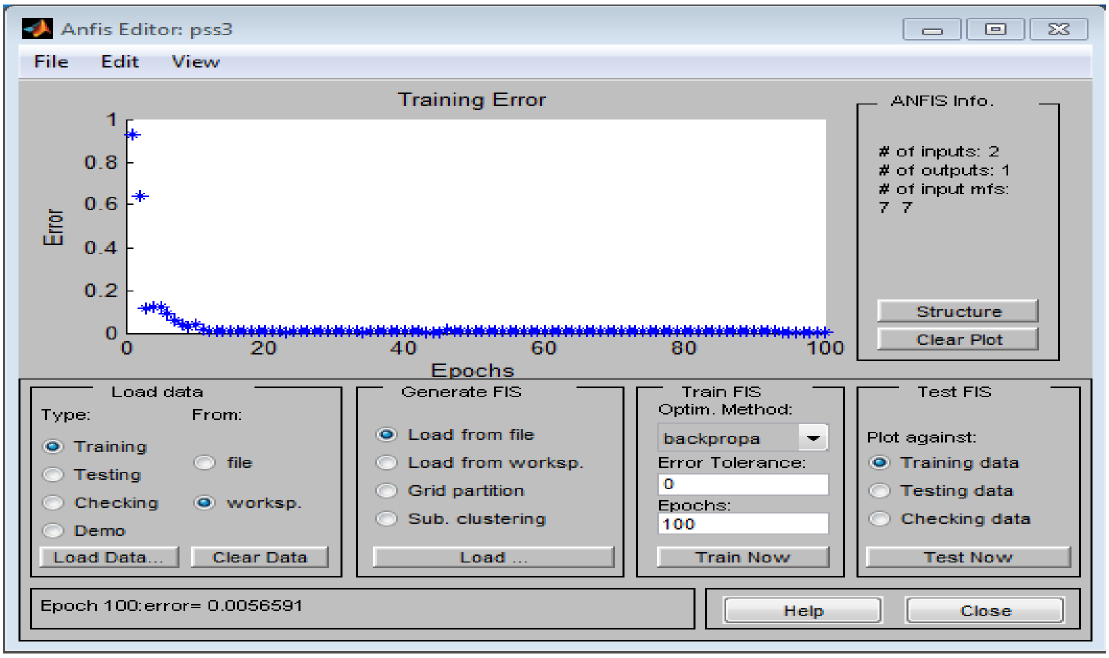


Fig.24. The training of ANFIS showing the training error

Finally the trained model is tested against the output data as below:

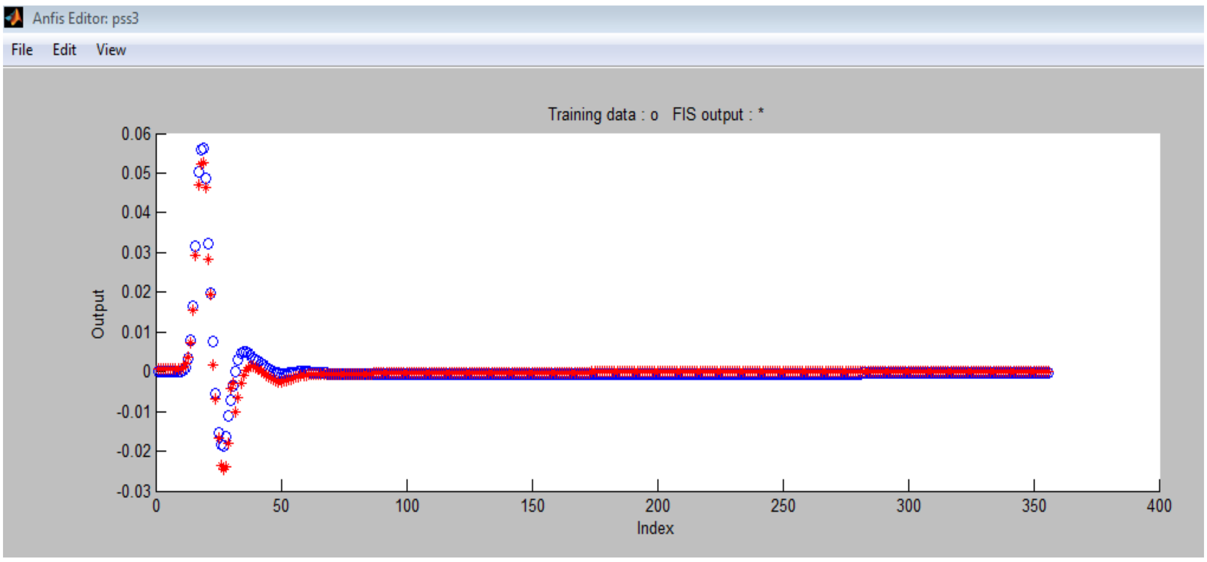


Fig.25. Comparison between trained and test data

As seen in the figure above, the trained data (red stars) almost faithfully follows the output (blue circles). This trained FIS model is exported for use in our fuzzy logic controller block (PSS). Thus, the offline-trained fis was used in the fuzzy controller to simulate the PSS.

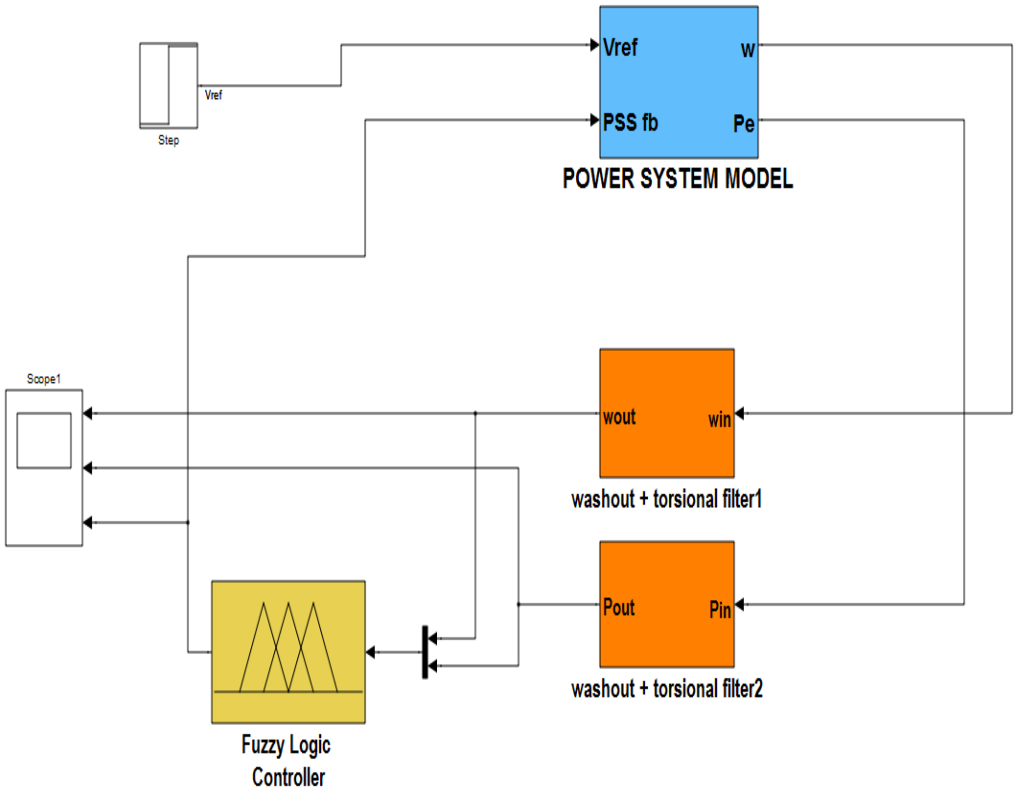


Fig.26. SIMULINK implementation of the fuzzy controller

The output responses as seen from the simulation results are crisp and have good design specifications such as rise time, overshoot and settling time.

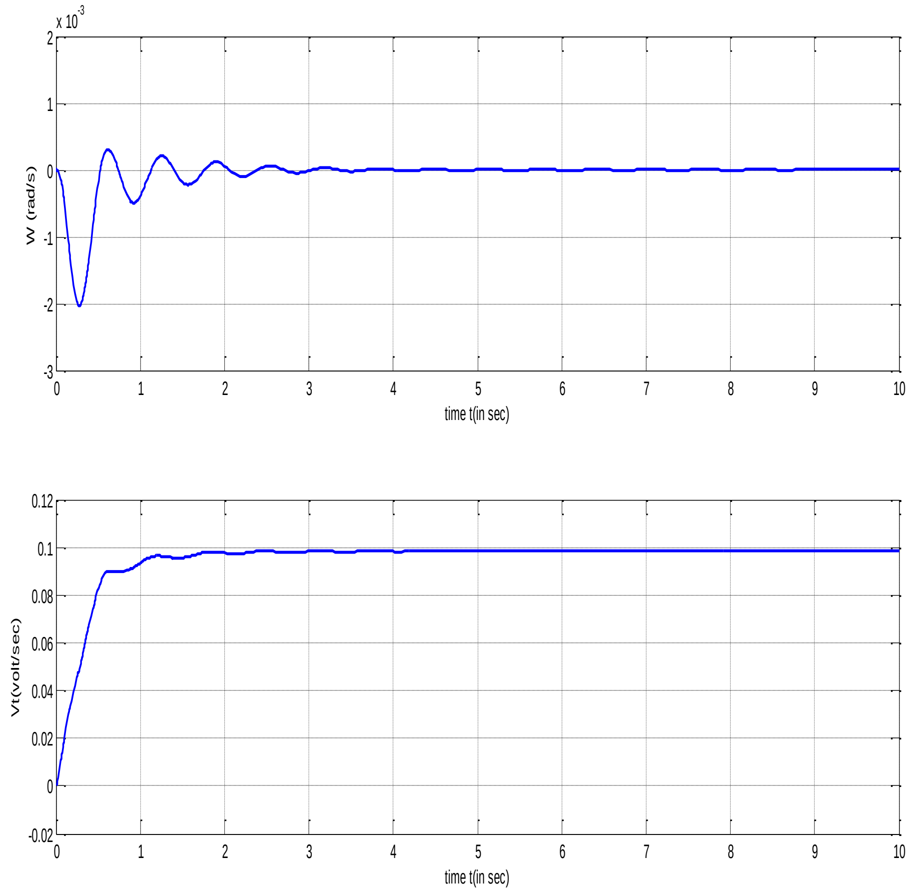
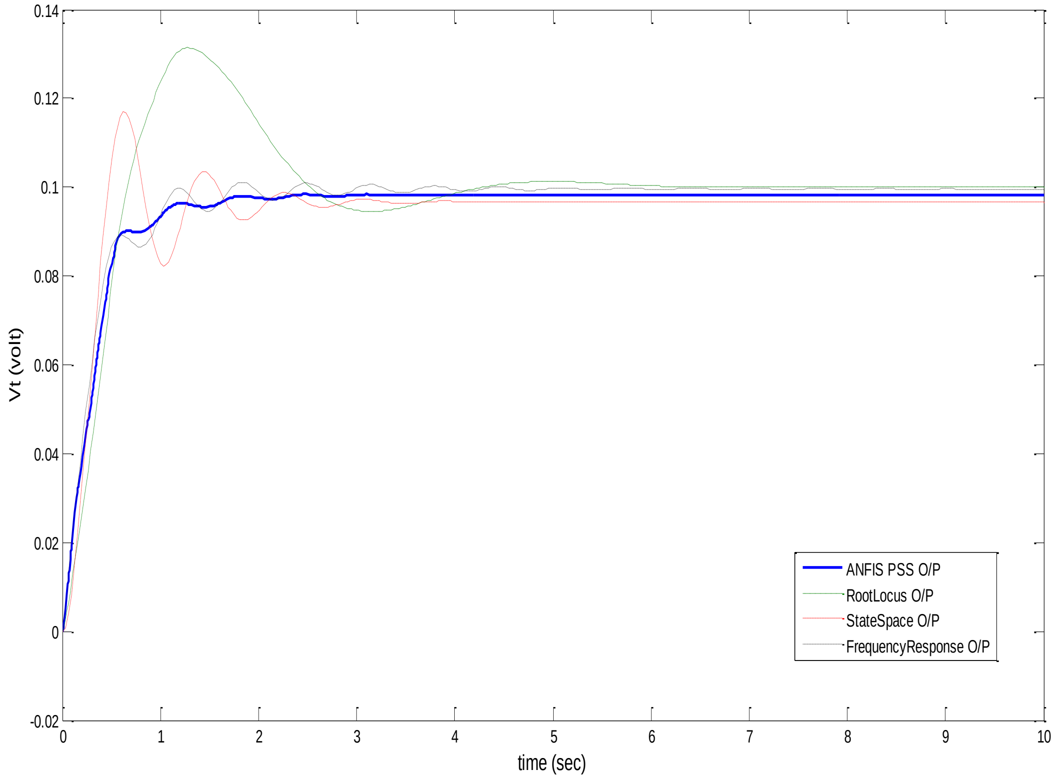


Fig.27. w and Vt outputs using the fuzzy controller

COMPARISON OF THE ANFIS PSS CONTROLLER WITH CPSS:

Finally, we are in a position to compare the conventional PSS or CPSS with the PSS developed using Fuzzy inference system. As seen in Figure 28, the fuzzy PSS has the best output response (Vt), the least overshoot and settling time. Also, it produces the best damping which is manifested in the plot showing the rotor speed perturbation (w). Thus, by proper training algorithms, the fuzzy PSS can surpass the performance of the CPSS.



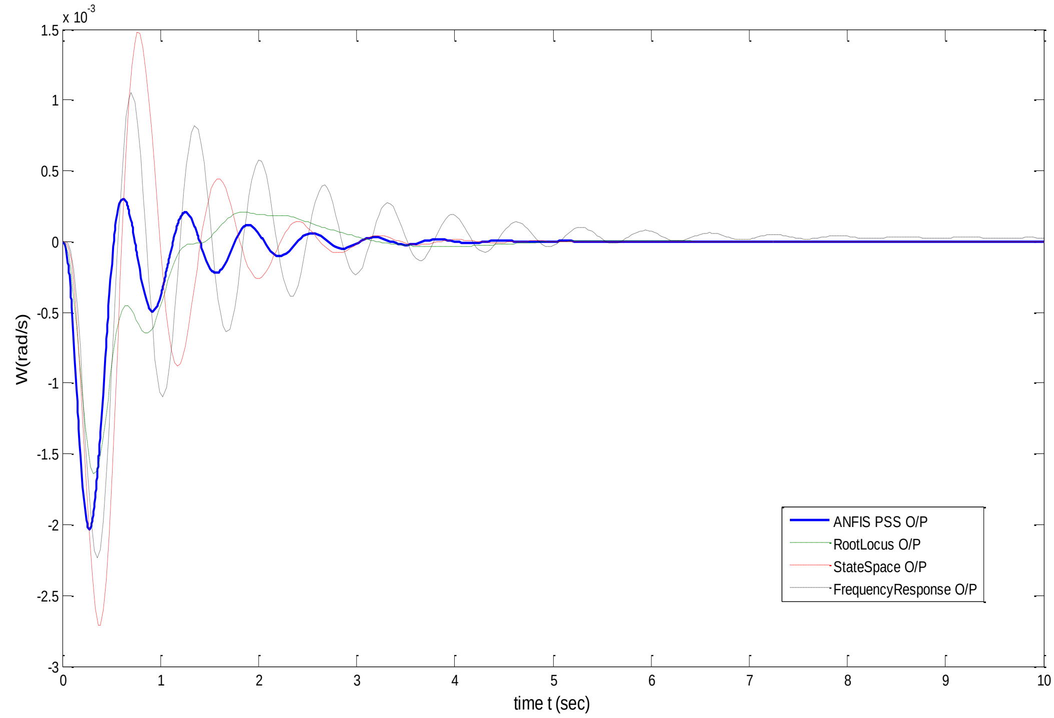


Fig.28. Comparison of Vt and w between CPSS and ANFIS PSS

CONCLUSION

The optimal design of Power System Stabilizer (PSS) involves a deep understanding of the dynamics of the single machine infinite bus system. In this project, we have tried to design the PSS using control system principles and hence view the problem as a feedback control problem. Both conventional control design methods like root-locus method, frequency response method and pole placement method as well as more modern adaptive methods like neural networks and fuzzy logic are used to design the PSS. By comparison of these methods, it is found that each method has its advantages and disadvantages.

The actual design method should be chosen based on real time application and dynamic performance characteristics. In general, it is found from our simulations that the ANFIS based adaptive PSS provides good performance if the training data and algorithms are selected properly. However, adaptive control involves updating controller parameters in real time using a system identifier which can be complicated and expensive. Hence, the economics of the process is also a constraint.

Although the first power system stabilizers were developed and installed during the 1960s and a lot of work has been done to improve its performance, modern control design algorithms can further enhance the performance of the PSS. In particular, adaptive control of PSS is still an active area. Digital design of the PSS is also possible. Hence, the design of the Power System Stabilizer has a lot of scope for future research.

REFERENCES

1. Joe H.Chow, G.E. Boukarim, A.Murdoch, *"Power System Stabilizers as Undergraduate Control Design Projects"*, IEEE Transactions on Power systems, vol.19, no.1, Feb.2004.
2. K.Ogata, *Modern Control Systems*, 5th edition, Prentice Hall Publications-2002.
3. P.Kundur, *Power System Stability and Control*, Mc-Graw Hill-1994.
4. F.P.deMello, C.Concordia, "*Concepts of synchronous machine stability as affected by excitation control"*, IEEE Trans. on Power App. Syst., vol-88 pp.189-202, April 1969.
5. E.V.Larsen, D.A.Swann, "*Applying Power System Stabilizers", Part-1, Part-2, Part-3,* IEEE Trans. on Power App. Syst, vol. PAS-100, pp. 3017-3046, June 1981.
6. P.Kundur, M.Klein , G J Rogers , M S Zywno, *"Application of power system stabilizers for enhancement of overall system stability"*, IEEE Trans. on Power App. Syst, vol-PWRS-4, pp.614-626, May 1989.
7. J.H.Chow, J.J. Sachez Gasca, "*Pole placement design of power system stabilizers",* IEEE Trans. on Power App. Syst., vol-4, pp.271-277, Feb.1989.
8. Ahmed A. Ba-Muquabel, Dr Mohammed A.Abido, "*Review of conventional PSS design methods"*, GCC Conference IEEE 2006.
9. Rajeev Gupta, B.Bandyopadhyay and A.M.Kulkarni, "*Design of Decentralized Power System Stabilizers for Multimachine Power System using Model Reduction"*, 5th Asian Control Conference, 2004.
10. K. Bollinger, A. Laha, **R.** Hamilton, T. Harras, "*Power Stabilizer Design using root locus methods",* IEEE Transactions on power apparatus and systems, vol. PAS-94, no. 5, September/October 1975.
11. C.L.Wadhwa, *Electrical Power Systems*, Sixth Edition, 2010, New Age International Publishers.
12. F.P. de Mello, P.J. Nolan, T.F. Laskowski, and J.M. Undrill, "*Co-ordinated application of stabilizers in multimachine power systems*", IEEE trans. on PAS-99, No.3, May/June 1980.

13. Hsu, Yuan-Yih; Chen, Chern-Lin, "*Identification of optimum location for stabilizer applications using participation factor*", Generation, Transmission and Distribution, IEEE Proceedings C , Vol-134, Issue-3, pp.238-244, 1987.
14. Zhou, E.Z., Malik O.P., Hope G.S., "*Theory and method of power system stabilizer location*", IEEE Transactions on Energy Conversion, Vol-6, Issue-1, pp.170-176, 1991.
15. J. M. Ramirez, I. Castillo, "*PSS & FDS simultaneous tuning*," EPSR 68 (2004) pp 33-40.
16. P. S. Rao, I. Sen, "*Robust tuning of power system stabilizers using QFT*," IEEE Trans. Control System Tech., vol. 7, no. 4, pp. 478-486, July 1999.
17. S. S. Sharif, "*Nonlinear PSS design technique*," IEEE, pp. 44-47, 1995.
18. A. Feliachi, et al., "*PSS design using optimal reduced order models part II: design*," IEEE Trans. Power Sys., vol. 3, no. 4, pp. 1676-1684, November 1988.
19. Ruhuao You, Hassan J.Eghbali, M.HashemNehrir, "*An Online Adaptive Neuro-Fuzzy Power System Stabilizer For Multi Machine System",* IEEE transactions on Power Systems, Vol-18, No-1, Feb-2003.
20. J. S. R. Jang et al., "Neuro Fuzzy and Soft Computing", Prentice Hall of India, 1997.
21. J. S. R. Jang, "*ANFIS: Adaptive-Network-Based Fuzzy Inference System*", IEEE Transactions on Systems, Man and Cybernetics, Vol. 23, 1993, pp. 665-684.
22. A. S. Venugopal, G Radman , M. Abdelrahman, "*An Adaptive Neuro Fuzzy Stabilizer For Damping Inter Area Oscillations in Power Systems*", Proceedings of the Thirty-Sixth Southeastern Symposium on System Theory, pp.41-44, 2004.
23. P. Mitra, S. P. Chowdhury, S. K. Pal et al., "*Intelligent AVR and PSS With Hybrid Learning Algorithm*", Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, 2008 IEEE, pp-1-7.
24. D.P. Kothari, I. J. Nagrath, "*Modern Power System Analysis*", 4th edition, Tata Mc-Graw Hill Publcation, New-Delhi, 2011.

APPENDIX-1

1. **The power system model [1]:**

   The state equations are:-

   $\Delta x' = A\Delta x + B\Delta u$

   $\Delta y = C\Delta x$

   Where state variables $x=[\delta \;\; \omega \;\; E_q' \;\; \psi_d \;\; E'_d \;\; \psi_q \;\; V_r\,]^T$

   Output variables $y=[V_{term} \;\; \omega \;\; P_e]^T$

   Input variable $u=V_{ref}$

   Where, δ= rotor angle in radian.

   ω= angular frequency in radian/sec.

   ψ_d, E_d'= direct axis flux and field.

   Ψ_q, E'_q= quadrature axis flux and field

   V_{term}= terminal voltage

   P_e= Power delivered to the infinite bus.

A=

$$\begin{vmatrix} 0 & 377.0 & 0 & 0 & 0 & 0 & 0 \\ -0.246 & -0.156 & -0.137 & -0.123 & -0.0124 & -0.0546 & 0 \\ 0.109 & 0.262 & -2.17 & 2.30 & -0.0171 & -0.0753 & 1.27 \\ -4.58 & 0 & 30.0 & -34.3 & 0 & 0 & 0 \\ -0.161 & 0 & 0 & 0 & -8.44 & 6.33 & 0 \\ -1.70 & 0 & 0 & 0 & 15.2 & -21.5 & 0 \\ -33.9 & -23.1 & 6.86 & -59.5 & 1.50 & 6.63 & -114 \end{vmatrix}$$

B=

$$\begin{vmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 16.4 \end{vmatrix}$$

C=

$$\begin{vmatrix} -0.123 & 1.05 & 0.230 & 0.207 & -0.015 & -0.460 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1.42 & 0.900 & 0.787 & 0.708 & 0.0713 & 0.314 & 0 \end{vmatrix}$$

2. **The model for $G_w(s)$. i.e. effect of the speed on electrical torque due to machine dynamics [1].**

$$\mathbf{A_w} = \begin{vmatrix} 0 & 377 & 0 \\ -K & -D & a_{23} \\ a_{31} & a_{32} & A_{33} \end{vmatrix}$$

K=0.2462, **D**=0.1563

The model of the damping loop [1] is

$\xi' = A_{33}\xi + a_{32}\omega$

$\tau = a_{23}\xi$

Where,

$A_{33}=$

$$\begin{vmatrix} -2.17 & 2.30 & -0.0171 & -0.0753 & 1.27 \\ 30.0 & -34.3 & 0 & 0 & 0 \\ 0 & 0 & -8.44 & 6.33 & 0 \\ 0 & 0 & 15.2 & -21.5 & 0 \\ 6.86 & -59.5 & 1.50 & 6.63 & -114 \end{vmatrix}$$

$a_{32}=$

$$\begin{vmatrix} 0.262 \\ 0 \\ 0 \\ 0 \\ -23.1 \end{vmatrix}$$

$a_{23}=$

$$\begin{vmatrix} -0.137 & -0.123 & -0.0124 & -0.0546 & 0 \end{vmatrix}$$

3. transfer function of the 7th order observer-controller VR :

a) $$\frac{405.6s^6+7.32e4s^5+3.588e6s^4+6.35e7s^3+4.90e8s^2+4.83e9s+11.77e10}{s^7+193.3s^6+1.12e4s^5+2.7e5s^4+3.26e6s^3+2.81e7s^2+1.85e8s+3.2e8}$$

b) **transfer function of 1st order minimized VR**:

$$\frac{K}{s+13.13}$$

c) **transfer function of the 5th order minimized PSS**

$$\frac{-20s^4 - 4120s^3 - 51580s^2 - 10440s - 540}{s^5 + 39s^4 + 507s^3 + 6183s^2 + 1112s + 57}$$

4. Tabulation of rise-time t_r (sec) in a grid of K_p and K_i:

T_r(sec)	K_i=0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
K_p=5	6.072	4.228	3.487	3.046	2.745	2.505	2.364	2.224	2.104	1.984
10	3.046	2.364	2.004	1.783	1.683	1.583	1.483	1.383	1.282	1.222
15	1.843	1.683	1.563	1.262	1.162	1.102	1.062	1.022	1.002	0.962
20	1.623	1.142	1.082	1.022	1.002	0.962	0.922	0.902	0.862	0.822
25	1.062	1.022	0.982	0.942	0.902	0.882	0.821	0.761	0.641	0.581
30	0.982	0.962	0.922	0.882	0.841	0.561	0.521	0.501	0.481	0.481
35	0.942	0.922	0.541	0.501	0.461	0.461	0.441	0.441	0.421	0.421
40	0.481	0.461	0.441	0.421	0.420	0.401	0.401	0.381	0.381	0.381
45	0.421	0.401	0.381	0.380	0.380	0.360	0.360	0.360	0.340	0.340
50	0.381	0.361	0.360	0.341	0.340	0.340	0.340	0.321	0.320	0.320

5. Tabulation of Maximum-overshoot M_p (%) in a grid of K_p and K_i:

M_p(%)	K_i=0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
K_p=5	-2.737	6.508	11.98	16.31	19.80	22.70	25.18	27.42	29.43	30.93
10	-0.318	5.237	9.496	12.80	15.85	18.00	20.50	22.37	23.71	25.23
15	0.131	4.822	8.375	11.40	13.63	15.91	18.00	19.57	20.73	21.62
20	1.156	5.201	8.220	10.86	13.05	14.63	16.27	18.19	19.80	21.11
25	2.570	5.953	8.783	10.88	12.75	14.74	16.38	17.69	18.70	19.44
30	4.156	7.233	9.523	11.55	13.45	14.99	16.19	17.09	17.81	19.41
35	5.941	8.586	10.63	12.59	14.15	15.36	16.27	17.51	19.00	20.35
40	7.769	10.11	12.00	13.69	15.02	16.02	17.16	18.60	19.88	21.03
45	10.48	11.89	13.39	14.88	16.04	16.91	18.23	19.51	20.64	21.63
50	17.67	18.74	19.28	19.82	20.53	21.26	22.00	22.91	23.85	24.79

APPENDIX-2 [LIST OF FIGURES]

APPENDIX-3 (MATLAB CODES)

Here, we provide some of the MATLAB™ scripts used in the design and the simulation process:

ROOT LOCUS DESIGN:

1. To convert the power system model into transfer function:

```
% this function converts the power system
% model from state space to transfer function.

% A,B,C,D are the state parameters

% PS0 refers to transfer function matrix having 3 outputs
% PS refers to transfer function with output=w
% PS1 refers to transfer function with output=Vterm

% all coeff having very small values are approximated
% to zero in the saved variables

clc
clear
A=[0, 377.0, 0, 0, 0, 0, 0; -0.246, -0.156, -0.137, -0.123, -
0.0124, -0.0546, 0; 0.109, 0.262, -2.17, 2.30, -0.0171, -0.0753,
1.27; -4.58, 0, 30.0, -34.3, 0, 0, 0; -0.161, 0, 0, 0, -8.44,
6.33, 0; -1.70, 0, 0, 0, 15.2, -21.5, 0; -33.9, -23.1, 6.86, -
59.5, 1.50, 6.63, -114];
B=[0; 0; 0; 0; 0; 0; 16.4];
C=[-0.123, 1.05, 0.230, 0.207, -0.105, -0.460, 0; 0, 1, 0, 0, 0,
0, 0; 1.42, 0.900, 0.787, 0.708, 0.0713, 0.314, 0];
D=[0; 0; 0];
[numPS0,denPS0]=ss2tf(A,B,C,D);
numPS=numPS0(2,:);
denPS=denPS0;
numPS1=numPS0(1,:);
denPS1=denPS0;
save 'tf_ps.mat'    % saves the workspace variables to tf_ps.mat
```

2. To compare the rise-time and maxium overshoot by taking a proportional VR

```
% ROOT LOCUS DESIGN

% to display the RISE-TIME & MAX-OVERSHOOT
% by taking a PROPORTIONAL voltage regulator
% and varying Kp
% tolerance=0.08 of final value(unit step)
% 10<= Kp <=100

% also plots the step-response

clc
clear
load tf_ps.mat;
```

```
denVR=1;
numVR=0;
t=linspace(0,10,500);
for n=1:10
    numVR=numVR+10;    numVR  % display gain of the VR
    numG=conv(numPS1,numVR);
    denG=conv(denPS1,denVR);
    [numTotal,denTotal]=feedback(numG,denG,1,1);
    [y,x,t]=step(numTotal,denTotal,t);
    y=0.1.*y;
    r=1;
    while y(r)<0.08
        r=r+1;
    end;
    rise_time=t(r-1)  %display rise time
    ymax=max(y);
    max_overshoot=(ymax-0.1).*1000  %display max overshoot
    figure(n)
    plot(t,y)
end
```

3. **To tabulate the rise-time and maximum overshoot vs the K_p and K_i values by taking a PI VR:**

```
% ROOT LOCUS DESIGN

% to tabulate the RISE-TIME & MAX-OVERSHOOT
% by taking a PI voltage regulator
% and varying Kp and Ki
% tolerance=0.09 of final value(unit step)

% 5<= Kp <=50 (different rows have diff Kp)
% 0.1<= Ki <=1 (diff columns have diff Ki)

clc
clear
load tf_ps.mat;
denVR=[1,0];
numVR=0; Kp=0;
t=linspace(0,10,500);
for m=1:10
   Kp=Kp+5;
   Ki=0;
   for n=1:10
       Ki=Ki+0.1;
       numVR=[Kp,Kp*Ki];
       numG=conv(numPS1,numVR);
       denG=conv(denPS1,denVR);
       [numTotal,denTotal]=feedback(numG,denG,1,1);
       [y,x,t]=step(numTotal,denTotal,t);
       y=0.1.*y;
       r=1;
        while y(r)<0.09
            r=r+1;
        end;
```

```
        rise_time(m,n)=t(r-1);  %store rise time
        ymax=max(y);
        max_overshoot(m,n)=(ymax-0.1).*1000;        %store    max
overshoot
   end
end
```

4. **To plot the step response and the root-locus plot (of regulation loop) for PI VR**

```
% ROOT LOCUS DESIGN

% plots the step response taking PI Vr and
% varying Ki

% Vtpi imported from simulink model simulation
clc
clear
load Vtpi;
plot(t,Vtpi1,t,Vtpi2,t,Vtpi3,t,Vtpi4,t,Vtpi5,t,Vtpi6,t,Vtpi7)
xlabel('time');
ylabel('voltage');
title('power system with PI VR (PSS loop open),Kp=20');
legend('Ki=0.1','Ki=0.5','Ki=1','Ki=2','Ki=2.5','Ki=3','Ki=3.5');

% also plots the root locus for the feed-forward loop
% with PI VR
load tf_ps;
numVRpi=[35,14];
denVRpi=[1,0];
numG3=conv(numPS1,numVRpi);
denG3=conv(denPS1,denVRpi);
figure(2)
rlocus(numG3,denG3)
axis([-1,1,-20,20])
```

5. **To calculate the final transfer function of the open-loop system incorporating the PI Voltage-regulator, the washout filters and the torsional filter:**

```
% calculates the transfer function of the open loop system

%  the  system  consists  of  the  VR(numVR,denVR),Power  system
TF(numPS,denPS),
% the filters WF,TOR and the PSS(numPSS,denPSS)
% the open loop tf= [numFINAL,denFINAL]

% it also shows the root locus plot of the open loop
% shows the dominant poles and zeros only

clc
clear
load tf_ps.mat
numVR=[35,14];
denVR=[1,0];
numPSS=[0,1];
denPSS=[0,1];
```

```
numWF=[10,0];
denWF=[10,1];
numTOR=[0,0,-1];
denTOR=[0.0017, 0.061, 1];
numG=conv(numPS,numVR);
numFilters=conv(numWF,numTOR);
numH=conv(numFilters,numPSS);
numFINAL=conv(numG,numH);
denG=conv(denPS,denVR);
denFilters=conv(denWF,denTOR);
denH=conv(denFilters,denPSS);
denFINAL=conv(denG,denH);
save 'finalTF.mat'

rlocus(numFINAL,denFINAL);
axis([-30,30,-50,50]);
title('root locus (PSS loop)');
```

6. To calculate angle of departure from the positive swing mode without the PSS:

```
% ROOT LOCUS DESIGN

% to calculate the angle of departure from the dominant pole
% of the uncompensated system

% we have to design the lead compensator so as to
% make this angle of departure 180 deg
% keeping other parameters as specified in design data

% finalTF stores the tf of the complete open pss-loop

clc
clear
load finalTF.mat;
poles_ol=roots(denFINAL);
p1=poles_ol(7);
zeros_ol=roots(numFINAL);

for m=1:11
    angpole(m)=180./pi.*angle(p1-poles_ol(m));
end

for n=1:6
    angzero(n)=180./pi.*angle(p1-zeros_ol(n));
end

sum1=0;
for m=1:11
    sum1=sum1+angpole(m);
end

sum2=0;
for n=1:6
    sum2=sum2+angzero(n);
end
```

```
angle_dep=180-sum1+sum2;
angle_dep  %display angle of departure
```

7. **To incorporate the lead-compensator in the PSS and plot the root locus and the step response:**

```
%ROOT LOCUS DESIGN

% plots the root locus of the final compensated system
% the angle of departure from the swing mode
% of the dominant pole should be close to 180 degrees

clc
clear
load finalTF;
numCMP=[247,1729,3025];
denCMP=[1,48,576];
NUM=conv(numFINAL,numCMP);
DEN=conv(denFINAL,denCMP);
rlocus(NUM,DEN)
axis([-20,20,-30,30]);
title('root locus of compensated system');

%ROOT LOCUS DESIGN

% step-response of the compensated and uncompensated systems
% for comparison.

% the data Vtcomp is taken from simulink model simulation
% Vtcomp contains Vtcl and Vtclcom

clc
clear
load Vtcomp;
plot(tout,Vtcl,tout,Vtclcom); grid on;
title('compensated PSS vs uncompensated PSS');
xlabel('time');
ylabel('terminal voltage');
legend('uncompensated','compensated');
```

<u>FREQUENCY RESPONSE DESIGN</u>:

8. **To plot the frequency-response of the regulation loop without the VR**

```
% FREQUENCY RESPONSE DESIGN

% plotting the frequency response from u to Vterm
% also shows the gain and phase margin
% VR is assumed to have gain=1

% also displays the uncompensated dc gain

clc
```

```
clear
load tf_ps
w=logspace(-2,3,100);
[mag,phase,w]=bode(numPS1,denPS1,w);
margin(mag,phase,w); grid on;
[Gm,Pm,wg,wp]=margin(mag,phase,w);
Gm=20*log10(Gm);
dcgain_uncomp=20*log10(mag(1));
sprintf('uncompensated dc gain= %f',dcgain_uncomp)
```

9. Design of the lag-compensator for the VR and comparison of the frequency resp:

```
% FREQUENCY RESPONSE DESIGN
% this script is for the lag compensator design of VR

% Reqd: min dc gain=200(~46dB), min phase margin=80 degrees
% uncompensated dc gain=-2.57dB
% hence K is calculated from above data
clc
clear
dcgain_req=20*log10(200);
K=ceil(10^((dcgain_req+2.57)/20));
sprintf('reqd gain addition: K=%d',K)

% now the bode plot is drawn multiplying the calc K
load tf_ps
w=logspace(-2,3,100);
figure(1);
[mag1,phase1,w]=bode(numPS1*K,denPS1,w);
margin(mag1,phase1,w); grid on;
[Gm,Pm,wg,wp]=margin(mag1,phase1,w);
sprintf('dc gain of gain compensated system = %f',mag1(1))

% now the compensator is designed so that phase margin
% is close to 80 degrees

% from the bode plot, we find that the
% new gain crossover frequency should be = 5 rad/sec.
% we have to bring the magnitude curve to 0dB at this frequency
% i.e. approx 18dB attenuation
% hence, 20log(1/B)= -18. or B=8 (approx)

% also we choose zero position= 0.1
% ( i.e. 1 octave to 1 decade below the new gain crossover freq.)
% hence pole position = 0.1/8=.0125
% reqd compensator is (270/8)*(s+0.1/s+0.0125)

% now we plot the bode diag. of the compensated system
sprintf('Kc=%d',ceil(K/8))
numCOMP=conv(numPS1,[40,4.0]); % we have taken Kc=40 here instead
of 34
denCOMP=conv(denPS1,[1,0.0125]);
figure(2);
[mag2,phase2,w]=bode(numCOMP,denCOMP,w);
margin(mag2,phase2,w); grid on;
```

```
% comparison of the compensated and uncompensated bode plots
figure(3);
bode(numPS1,denPS1,w); hold;
bode(numCOMP,denCOMP,w); grid on;
title('comparison of uncompensated and lag compensated VR');
legend('uncompensated','lag compensated');
% end of code
```

10. To plot the step-response of the regulation-loop and comparing the t_r and M_p:

```
% FREQUENCY RESPONSE DESIGN
% this script calculates the rise time and max overshoot
% of the compensated VR loop
% the compensated VR parameters have been calculated in
lag_compVR_F2.m
% tolerance taken is 0.09 of unit step

clc
clear
load tf_ps.mat;
numVRcomp=[40,4.0];
denVRcomp=[1,0.0125];
t=linspace(0,10,500);
numG=conv(numPS1,numVRcomp);
denG=conv(denPS1,denVRcomp);
[numVRloop,denVRloop]=feedback(numG,denG,1,1);
[y,x,t]=step(numVRloop,denVRloop,t);
y=0.1.*y;
r=1;
    while y(r)<0.09
        r=r+1;
    end;
rise_time=t(r-1)  %display rise time
ymax=max(y);
max_overshoot=(ymax-0.1).*1000  %display max overshoot
plot(t,y); grid on; title('step response with lag compensated
VR');
xlabel('time(sec)--->'); ylabel('voltage(volt) --->');

% end of code
```

11. To decompose the system into Q(s) i.e. the part of speed which affects the damping torque and plot the frequency response of the damping loop:

```
% FREQUENCY RESPONSE DESIGN
% this script develops the new state space matrices
% after decomposing the PS state space model
% into paths: Q(s)-> effect of w on electrical torque
%             K=0.2462-> synchronizing torque loop
%             D=0.1563-> damping torque loop
% we get the new state space matrices from main A matrix as:
% A=A33(square matrix 5*5), B=a32(column vector 5*1), C=a23(row
vector 1*5)
```

```
% then we connect resultant t-f to the filters
% and plot the freq. response of F(s)

clc
clear
numVRcomp=[40,4.0];
denVRcomp=[1,0.0125];
A1=[-2.17,2.30,-0.0171,-0.0753,1.27;30.0,-34.3,0,0,0;0,0,-
8.44,6.33,0;0,0,15.2,-21.5,0;6.86,-59.5,1.50,6.63,-114];
B1=[0.262;0;0;0;-23.1];
C1=[-0.137,-0.123,-0.0124,-0.0546,0];
D1=0;
[numQ,denQ]=ss2tf(A1,B1,C1,D1);
numGw=conv(numQ,numVRcomp);
denGw=conv(denQ,denVRcomp);

numWF=[10,0];
denWF=[10,1];
numTOR=[0,0,-1];
denTOR=[0.0017, 0.061, 1];

numFilters=conv(numWF,numTOR);
denFilters=conv(denWF,denTOR);
numF=conv(numFilters,numGw);
denF=conv(denFilters,denGw);

w=logspace(0,2,100);
[magF,phaseF,w]=bode(numF,denF,w);
bode(numF,denF,w); grid on;
title('Freq. response of the damping loop');
save 'decomp.mat'  % saves the workspace variables

% end of code
```

12. To design the lead-compensator for the PSS:

```
% FREQUENCY RESPONSE DESIGN

% from the bode plot of F(s) in decomp_speedtorque_mat.m
%  we find that the phase at 2rad/sec=-37 degrees; phase at 12
rad/sec=-65 d
% and phase at 20 rad/sec= -105 degrees.

% from the design specifications, we need:
% phase of F(s).Kd(s) to be 0 to -20 degrees in the range 2 to 20
rad/sec

% hence, we require to add a phase of approximately:
% 35 deg at 2 rad/sec; 50 to 60 deg at 12 rad/sec; 90 to 100 deg
at 20 rad/sec

% we plot Pm vs alpha to show the relation
clc
clear
phi=linspace(0,90,1000);
```

```
alpha=(1-sind(phi))./(1+sind(phi));
plot(phi,alpha); grid on;
title('Pm vs a'); xlabel('Pm (degrees)--->'); ylabel('alpha--->');

% from the lead compensator design metod in K.Ogata:
% we select max phase addition to be achieved in freq. 20 rad/sec= 100 deg.
% since the phase addition is too large for a single lead-compensator,
% we take 2 series lead-compensators, each providing 50deg add at 20rad/s
% sin(Pm)=(1-a)/(1+a); or, a=(1-sin(Pm))/(1+sin(Pm))

clc
clear
Pm=50; wm=20;
a=(1-sind(50))/(1+sind(50));
T=1/(sqrt(a)*wm);
z=(ceil((1/T)*100))/100;
p=ceil(1/(a*T));
Kc=(1/a);
sprintf('There are 2 identical lead-compensators in series')
sprintf('for each compensator:')
sprintf('max phase addition Pm = %d deg at wm = %d radian/sec',Pm,wm)
sprintf('alpha=%f, T=%f',a,T)
sprintf('zero at=%0.2f  pole at=%d',z,p)
sprintf('gain Kc=%0.1f',Kc)
```

13. To implement the lag compensated VR and lead compensated PSS and plot the root locus of the damping loop:

```
%FREQUENCY RESPONSE DESIGN

% from lead_control_design_F5.m, we get the lead controller for PSS loop:
% Kd(s)=K*[7.5*(s+7.14)/(s+55)]^2

% we implement this in the design and plot the root locus of it
% from this root locus, we get K=15 to 20 for Z>=15%
% we then imlement the full system in simulink model
% final_compensated_systemF7.mdl

clc
clear
load tf_ps.mat
numVRcomp=[40,4];
denVRcomp=[1,0.0125];

numPSS=[56,800,2855];
denPSS=[1,110,3025];

numWF=[10,0];
denWF=[10,1];
```

```
numTOR=[0,0,-1];
denTOR=[0.0017, 0.061, 1];
numG=conv(numPS,numVRcomp);
numFilters=conv(numWF,numTOR);
numH=conv(numFilters,numPSS);
denG=conv(denPS,denVRcomp);
denFilters=conv(denWF,denTOR);
denH=conv(denFilters,denPSS);

numFINAL1=conv(numG,numH);
numFINAL=15.*numFINAL1;
denFINAL=conv(denG,denH);

rlocus(numFINAL1,denFINAL);
axis([-25,5,-5,30]);
title('root locus (PSS loop with lead compensator)');
```

14. To plot the step-response and frequency-response of the compensated and uncompensated system and compare the t_r and M_p:

```
% FREQUENCY RESPONSE DESIGN

% plotting the step response of system with and without PSS loop
% the variables are taken from simulink simulation
% Vt1= output without PSS loop
% Vtclcom= output with PSS loop

clc
clear
load final_stepresponse.mat
plot(tout,Vt1,tout,Vtclcom);grid on;
axis([0,10,0,0.12]);
title('comparison of step response with and without pss');
xlabel('time(sec)-->'); ylabel('voltage(V)-->');
legend('without PSS','with PSS');

% rise time and max overshoot of final system i.e. Vtclcom
% tolerance= 85% of final value

r=1;
    while Vtclcom(r)<0.085
        r=r+1;
    end;
rise_time=tout(r-1)  %display rise time
ymax=max(Vtclcom);
max_overshoot=(ymax-0.1).*1000  %display max overshoot

% FREQUENCY RESPONSE DESIGN

%  the  frequency  response  of  compensated  and  uncompensated(PSS)
system
% is compared.
% decomp.mat contains the decomposed matrices, Gw(s), and F(s)

clc
```

```
clear
load decomp.mat

numPSS=[56,800,2855];
denPSS=[1,110,3025];

numF2=conv(numF,numPSS);
denF2=conv(denF,denPSS);
```

STATE-SPACE DESIGN:

15. To design the full-order observer-controller for the VR:

```
% STATE SPACE DESIGN

% this script is to design the state feedback observer based
% controller to shift the closed loop VR pole of the system
% to a desirable position for a given time-const.

clc
clear
load tf_ps.mat;
[A1,B1,C1,D1]=tf2ss(numPS1,denPS1);  %  1-output(Vt)  state  space
model from u to Vt
ol_poles=roots(denPS1);
ol_zeros=roots(numPS1);
rlocus(numPS1,denPS1);  axis([-3,3,-12,12]);  title('root  locus  of
VR loop showing dominant pole');

dominant_pole=-(min(abs(ol_poles)));
sprintf('dominant pole= %0.4f',dominant_pole)

% this pole at -0.1054 needs to be made faster by shifting it to
-4.0 in
% the controller
modified_poles=ol_poles;
modified_poles(7)=-4.0+0.0i;

%  now  we  find  the  controller  gain  matrix  using  the  ackerman's
formula
Kc=acker(A1,B1,modified_poles);

% now we find the observer gain matrix by first shifting the
% dominant pole to -8
modified_poles_obs=ol_poles;
modified_poles_obs(7)=-8.0+0.0i;
Ko=place(A1',C1',modified_poles_obs);% observer gain matrix
Ko=Ko.'; % transpose operation

% now we obtain the transfer function of the observer-controller
Ao=A1-(Ko*C1)-(B1*Kc);       Bo=Ko;
Co=Kc;                   Do=0;
[numVR_obs,denVR_obs]=ss2tf(Ao,Bo,Co,Do);
tf(numVR_obs,denVR_obs)
```

```
% the poles and zeroes of the observer controller
obs_poles=roots(denVR_obs);
obs_zeros=roots(numVR_obs);

% hence we have obtained the 7th order observer controller for
the VR. now we save the variables in obs_cont.mat
save 'obs_cont.mat' Ao Bo Co Do numVR_obs denVR_obs

% in the next step, we minimize this controller to a
% 1st order controller by approx pole-zero cancellations
% end of code.
```

16. To minimize the order of the VR controller and plot the step-response and frequency response of the regulation loop:

```
% STATE SPACE DESIGN

% this script is to reduce the order of the observer-controller
% of the VR from 7th to 1st order by cancelling approx poles and
zeroes

% also it plots the freq response of both 7th and 1st order
controller
% also it plots the step-response of both 7th and 1st order
controller

clc
clear
load obs_cont_VR.mat;

sprintf('Poles of the 7th order VR')
sprintf('%d\n',obs_poles)
sprintf('Zeros of the 7th order VR')
sprintf('%d\n',obs_zeros)

% by approx pole-zero cancellation, we find that all poles and
zeros are
% cancelled out except the pole at -13.14.
% hence the reqd controller tf is:
% Kv = 405/(s+13.14), where 405 is the gain got by comparing
poly(obs_zeros) and numVR_obs.
numVR_final=480; % for zero steady state voltage error, we set
gain=405*1.185
denVR_final=[1,13.14];
tf(numVR_final,denVR_final)

% now we plot the freq response of both the 1st and 7th order
controllers
w=logspace(-1,2,100);
figure(1);
bode(numVR_obs,denVR_obs,w); grid on; hold;
bode(numVR_final,denVR_final,w);
title('comparison of freq. response of 1st and 7th order
controllers for VR');
legend('7th order VR', '1st order VR');
```

```
% also we plot and compare the step response of 1st and 7th order
controllers.
% the variables are imported from simulink models for S1 and S2
load step_resp_VR.mat;
figure(2);
plot(tout,Vt7th_order,tout,Vt1st_order); grid on;
title('step response of 7th order and 1st order VR in closed loop
operation');
legend('7th order VR','1st order VR');

% end of code
```

17. To design the full-order observer based controller for the PSS, minimize the order, and finally implement it and plot the root locus, frequency response of the damping-loop and comparison of the step-response:

```
% STATE SPACE DESIGN

% this script is to design the PSS which is an
% observer controller of 11th order
% then we minimize the order with approx pole-zero cancellations
% to get a 5th order PSS

clc;
clear;
load tf_ps.mat;

% now we find the transfer-function from Vref to wf
% taking the 1st order VR
numVR_final=[480];
denVR_final=[1,13.14];
numWF=[10,0];
denWF=[10,1];
numTOR=[0,0,-1];
denTOR=[0.0017, 0.061, 1];
numG=conv(numPS,numVR_final);
denG=conv(denPS,denVR_final);
numFilters=conv(numWF,numTOR);
denFilters=conv(denWF,denTOR);

numGw=conv(numG,numFilters);
denGw=conv(denG,denFilters);

% now we convert Gw from t-f to state-space model
[Ag,Bg,Cg,Dg]=tf2ss(numGw,denGw);

% we also see the swing mode of Gw in Root-Locus Plot
figure(1);
rlocus(numGw,denGw); axis([-10,5,-20,20]);
title('swing mode of Gw(s)');

% from the root locus, we see that the swing mode is at:
% -0.48 + 9.33i      We have to shift this to -1.5 + 9.33i
% for 15% damping in the controller and -4.5 +9.33i
```

```
% for the observer to enable fast tracking of states.

% now we design the controller by shifting the swing mode to -
1.5+9.33i
poles_ol=roots(denGw);
zeros_ol=roots(numGw);

poles_control=poles_ol;
poles_control(7)= -1.5+9.33i;
poles_control(8)= -1.5-9.33i; % modified poles for the controller

Kc=acker(Ag,Bg,poles_control); % the controller gain matrix for
PSS

% then we design the observer by shifting the swing mode to -
4.5+9.33i
poles_obs=poles_ol;
poles_obs(7)= -4.5+9.33i;
poles_obs(8)= -4.5-9.33i; % modified poles for the observer

Ko=place(Ag',Cg',poles_obs);      Ko=Ko.'; % observer gain matrix
for PSS

% the final step is obtaining the T-F of the observer controller
Ao=Ag-(Ko*Cg)-(Bg*Kc);      Bo=Ko;
Co=Kc;                   Do=0;
[numPSS_obs,denPSS_obs]=ss2tf(Ao,Bo,Co,Do);
sprintf('TRANSFER FUNCTION OF THE 11TH ORDER CONTROLLER:')
tf(numPSS_obs,denPSS_obs)


% this is an 11th order controller
% next, we reduce the order by approx pole-zero cancellations
Po=roots(denPSS_obs);
Zo=roots(numPSS_obs);
sprintf('Poles of the 11th order controller:')
sprintf('%d\n',Po)
sprintf('Zeros of the 11th order controller:')
sprintf('%d\n',Zo)

% we get the final 5th order controller
numPSS_final=-20*[1,206,2579,522,27]; % 20 is the gain constant
when converting to polynomial from its roots
denPSS_final=[1,39,507,6183,1112,57];
sprintf('TRANSFER FUNCTION OF THE REDUCED 5TH ORDER CONTROLLER:')
tf(numPSS_final,denPSS_final)

% we plot the frequency response of 11th order PSS and 5th order
PSS
figure(2);
w=logspace(-1,2,100);
bode(numPSS_obs,denPSS_obs,w); grid on; hold;
bode(numPSS_final,denPSS_final,w);
title('Freq. Response comparison of 11th order and 5th order
PSS');
legend('11th order','5th order');
```

```
% we also plot the root locus of the damping loop
% we see that the swing mode has almost -180 deg angle of
departure
% thus providing pure damping
figure(3);
rlocus(conv(numGw,numPSS_final),conv(denGw,denPSS_final));
axis([-10,5,-20,20]);
title('root locus plot of the final damping loop with 1st order
VR and 5th order PSS');


% STATE SPACE DESIGN

% this script plots and compares the step response of the system
% with and without implementing the 5th order PSS

% in both the cases, the 1st order VR has been implemented

clc
clear
load step_resp_PSS.mat

plot(tout,Vt1st_order,tout,Vtfinal); grid on;
title('comparison of step response of system with and without
PSS');
legend('without PSS','with PSS');
xlabel('time--->');
ylabel('voltage--->');

% end of code
```