

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Self-Tuning Fuzzy PI Controller (STFPIC)

Salman Hameed

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/59810>

1. Introduction

Traditionally, control systems are designed through mathematical analysis and synthesis techniques. A wealth of control design techniques through mathematical approaches is available in the literature. Now, the first step for control system design through mathematical techniques is to obtain a proper model of the process to be controlled. However, for a highly nonlinear system like power system, it is often quite difficult to obtain an accurate mathematical model. Because of the inaccurate model, it is often quite an arduous task to design a control system which would be effective over a wide range of operating conditions. Under these conditions, Fuzzy Logic Controllers (FLCs) can be used quite effectively for controlling the process. Fuzzy logic controllers are essentially “model free” controllers, which try to control a process, based on the experience of a skilled human operator without requiring a detailed model of the process to be controlled. In many cases, application of fuzzy logic approach makes it possible to design a control system that is more robust, cost-effective, and easier to design as compared to a controller developed based on mathematical techniques.

Motivated by the above features of FLC, an attempt has been made to design fuzzy logic TCSC (Thyristor controlled series capacitor) controllers to improve power system stability. Generally, fuzzy PI controllers (FPIC) have been used in which the inputs to the fuzzy controllers are error (e) and change in error (Δe), while the output of FPIC is the incremental change in the control output (Δu). The error (e) is defined as $e = P_{\text{ref}} - P_{\text{act}}$ where P_{ref} is the steady-state power flowing through the line (in which TCSC is installed) and P_{act} is the actual power flowing through that particular line following a disturbance. From these two inputs, the FPIC generates the incremental change in TCSC reactance as its output ($\Delta u = \Delta x_{\text{TCSC}}$). The main components of any FLC are fuzzification unit, fuzzy inference unit, fuzzy rule base, and defuzzification unit. The fuzzification unit maps the measured crisp inputs into the fuzzy linguistic values which are subsequently used by the fuzzy inference mechanism. It is to be noted that the actual values of the e and Δe are not submitted directly to the fuzzification unit. The quantities e and Δe are first converted to normalized quantities e_N and Δe_N respectively, by using scaling factors G_e

and $G_{\Delta e}$ and subsequently these normalized quantities are passed to the fuzzification unit. The fuzzy inference mechanism, utilizing the fuzzy rule base, performs fuzzy logic operations on the fuzzified inputs to infer the control action. Finally, the defuzzification unit converts the inferred fuzzy control action into crisp, normalized incremental change in control output (Δu_N), which, in turn, is converted into actual incremental change in control output (Δu) by using the scaling factor G_u .

These three scaling factors (G_e , $G_{\Delta e}$, and G_u) are the main parameters for tuning the FPIC because variation of scaling factors (SFs) changes the normalized universe of discourse for input and output variables and their corresponding membership functions. These SFs play a role similar to that of the gains of a conventional controller. Therefore, they are extremely important for the controller stability and performance. Generally, there is no well-defined method for computing the appropriate values of SFs for FLC and as a result, the values of these SFs are sometimes decided by trial-and-error technique. To circumvent this trial-and-error procedure, a genetic algorithm (GA)-based technique is proposed for finding the appropriate values for G_e , $G_{\Delta e}$, and G_u . The proposed GA-based technique is described in details in [1]. The scaling factors have been tuned such that the power system oscillations are minimized after the occurrence of a disturbance. Specifically, the aim is to minimize the error between the P_{ref} and P_{act} following a disturbance. Various performance indices can be used to represent the above goal mathematically. Here, the integral of squared error (ISE) index has been used as it tends to place a greater penalty on large errors.

In many practical cases, even after a controller is established, there is a need to further tune the controller on-line. The necessity of applying further on-line tuning arises in situations where a controller must operate under uncertainty and/or when the available information is so limited that it is impossible or impractical to design in advance a controller with fixed properties. The main goal of the auto-tuning controller is to make the controller robust as far as possible when subjected to a wide range of operating conditions.

In this chapter, a simple and effective fuzzy-logic-based method for on-line tuning of TCSC FPIC is presented. The resulting controller is termed as self-tuning fuzzy PI controller (STFPIC). The effectiveness of the developed STFPIC has been validated through detailed nonlinear dynamic simulation studies on two different multimachine power systems. These study systems are: (a) two-area 4-machine system and (b) 10-machine 39-bus system. The simulation studies have been carried out on MATLAB/SIMULINK environment. A large number of fault cases involving three-phase, 5-cycle, solid short-circuit faults have been investigated on both these study systems for different loading conditions as well as different fault locations. From the simulation results, it has been observed that the proposed STFPIC for TCSC improves the system stability significantly. The detailed description of STFPIC is given in the following sections.

2. Self-tuning scheme

Fuzzy logic controllers (FLCs) contain a number of sets of parameters that can be altered to modify the controller performance. These are:

- The scaling factors (SFs) for each variable
- The fuzzy sets representing the meaning of linguistic values
- The if-then rules

A nonadaptive FLC is one in which these parameters do not change once the controller goes on-line. Adaptive FLCs have the capability of altering these parameters on-line. Further, the adaptive fuzzy controllers that modify the fuzzy set definitions or the scaling factors are called self-tuning or auto-tuning fuzzy logic controllers (FLCs) [2, 3]. Alteration of these parameters essentially fine-tunes an already working controller. On the other hand, adaptive FLCs that alter the rules are called self-organizing controllers [3]. These two types of fuzzy logic controllers have been developed [4-11] and implemented for various practical processes. Of the various tunable parameters, SFs have the highest priority due to their global effect on the control performance because changing the SFs changes the normalized universe of discourse, the domains, and the membership functions of input/output variables of FLC [3].

2.1. Self-tuning fuzzy PI controller (STFPIC)

It has been experimentally observed that a conventional FLC with constant scaling factors and a limited number of IF-THEN rules may have limited performance for a highly nonlinear plant [12]. As a result, there has been significant research on tuning of FLCs where either the input or output scaling factors or the definitions of the membership functions (MFs) and sometimes the control rules are tuned to achieve the desired control objectives [5, 8, 10]. Here, only the output scaling factor (SF) is tuned due to its strong influence on the performance and stability of the system. In this scheme, the output SF of the FLC is adjusted on-line according to the current states of the controlled processes. Specifically, the tuning of the output SF is based on the value of instantaneous error and change in error (e and Δe). The block diagram of the Self-tuning Fuzzy PI type Controller (STFPIC) is shown in Fig. 1 [13, 14]. From this figure, it is observed that by the use of a self-tuning mechanism, the controller's output SF gets modified, as shown by the dotted boundary. Thus, the output scaling factor gets modified at each sampling time and is updated by the gain updating factor α . This process depends on the trend followed by the controlled process output and the value of α is obtained using fuzzy rules [13, 19].

In contrast to FPIC (which has only one fuzzy logic block), there are two fuzzy logic blocks in STFPIC as shown in Fig. The top fuzzy logic block in Fig. 1 computes the main control output (Δu) of the FLC. The membership functions (MFs) for e , Δe , and Δu corresponding to this fuzzy logic block are shown in Fig. 2. The rule base for computing Δu is shown in Table The second fuzzy logic block (shown in dotted boundary) calculates the gain updating factor α at each sampling instant. For this second fuzzy block also, the MFs for e and Δe are same as shown in Fig. 2. The membership functions used for the factor α are defined in the domain [1] and are shown in Fig. 3. As each of the two inputs has 7 fuzzified variables, the rule base has a total 49 rules for calculating α as shown in Table 2. This rule base improves the control performance under large disturbances such as sudden loss of a generating unit or a large load, three-phase short circuit on the transmission lines, etc. [13, 20]. For example, immediately after a large

disturbance, e may be small but Δe will be sufficiently large (they will be of same sign) and, for this case, α should be large to increase the gain. Therefore, under these circumstances, the appropriate rules are “IF e is PS and Δe is PM, THEN α is B” or “IF e is NS and Δe is NM, THEN α is B.” On the other hand, for steady-state conditions (i.e., $e \approx 0$ and $\Delta e \approx 0$), controller gain should be very small (e.g., IF e is ZE and Δe is ZE, THEN α is ZE) to avoid chattering problem around the set point. Further justification for using the rule base in Table 2 can be found in [13].

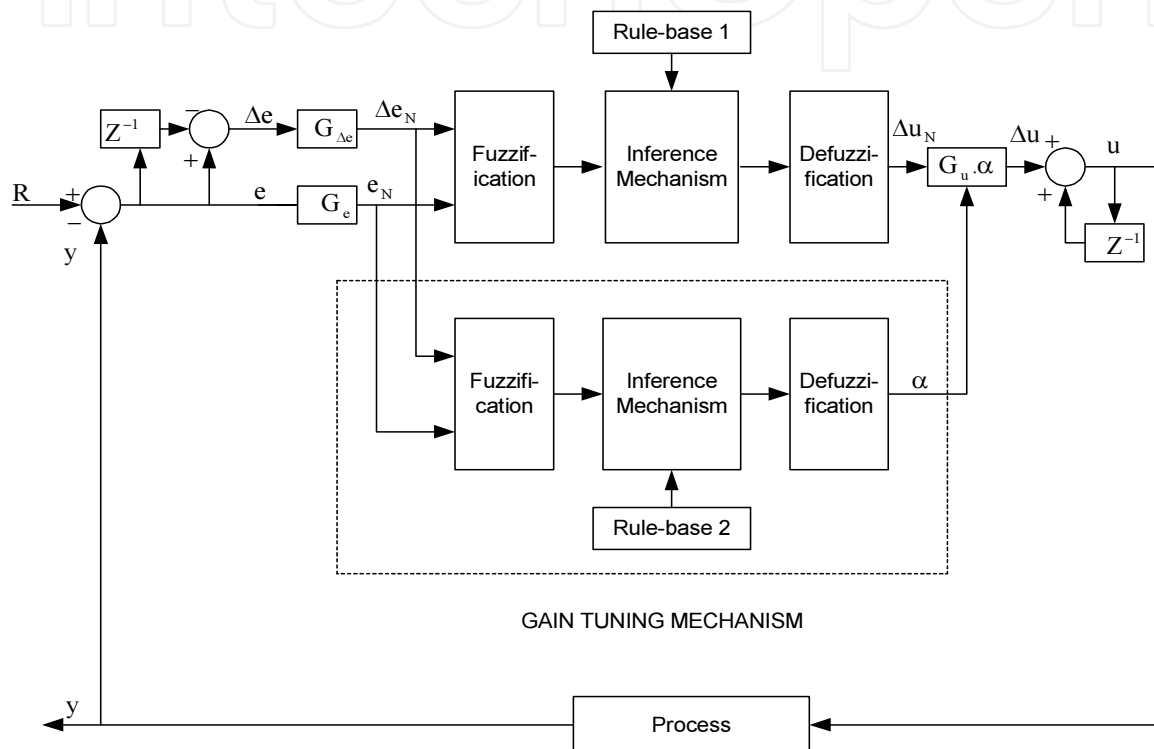


Figure 1. Block diagram of the STFPIEC

The following steps have been used for tuning the STFPIEC [13].

Step 1: Tune the SFs of the STFPIEC without the gain tuning mechanism and assuming $\alpha = 1$ (i.e., conventional FLC) for a given process to achieve a reasonably good control performance. For the tuning of the conventional FLC, GA [15, 20] has been used. At the end of this step, a good controller without self-tuning feature is obtained and this controller becomes the starting point (input) for the self-tuning controller in Step 2.

Step 2: Set the output SF (G_u) of the self-tuning FLC K times greater than that obtained in Step1, keeping the values of G_e and $G_{\Delta e}$ same as those of the conventional FLC. In this step, $\alpha \neq 1$, and is obtained from the rule base in Table 2. This factor K for the STFPIEC (to enhance the value of G_u) is found empirically with an objective to maintain the same rise time as that of the conventional FLC [13].

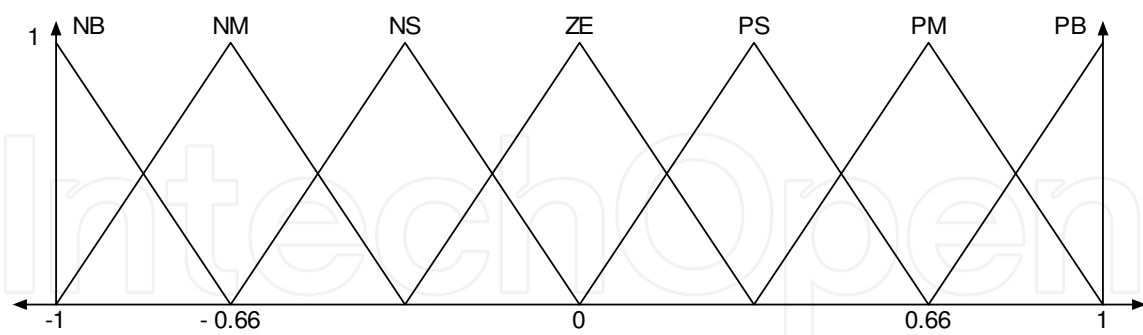


Figure 2. MFs for e , Δe , and Δu . N: Negative, P: Positive, ZE: Zero, B: Big, M: Medium, S: Small

$\Delta e/e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

Table 1. Rule Base for Δu

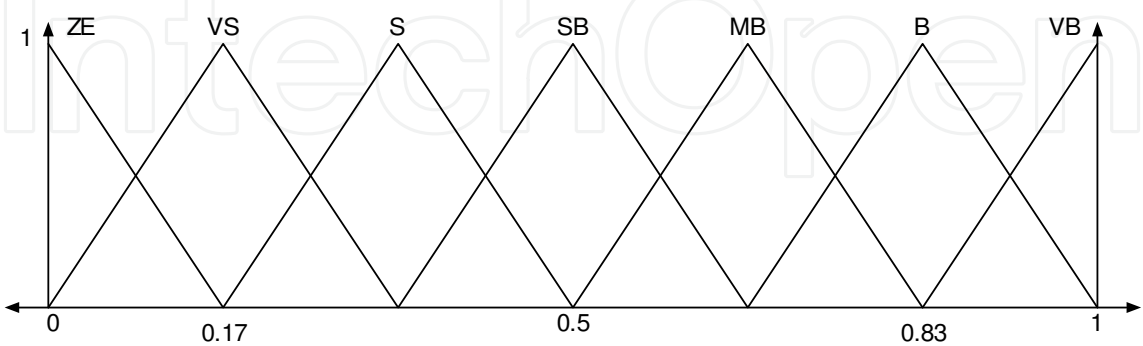


Figure 3. MFs for gain updating factor (α). ZE: Zero, V: Very, B: Big, M: Medium, S: Small

$\Delta e/e$	NB	NM	NS	ZE	PS	PM	PB
NB	VB	VB	VB	B	SB	S	ZE
NM	VB	VB	B	B	MB	S	VS
NS	VB	MB	B	VB	VS	S	VS
ZE	S	SB	MB	ZE	MB	SB	S
PS	VS	S	VS	VB	B	MB	VB
PM	VS	S	MB	B	B	VB	VB
PB	ZE	S	SB	B	VB	VB	VB

Table 2. Rule Base for α

2.2. Scaling factor's tuning using genetic algorithm (GA)

Here, the power system oscillations are minimized by tuning of the scaling factors using GA. The aim is to minimize the error between the steady-state power (P_{ref}) flowing through the line in which the TCSC is installed and the actual power flowing through that line (P_{actual}) after a disturbance [12, 19]. Several performance indices can be used to represent the above goal mathematically. The integral of the squared error (ISE) [12] as shown in eqn. (1) has been selected as it tends to place a greater penalty on large errors. This goal can be formulated as the minimization of the objective function F where,

$$F = \int_0^{t_{sim}} e^2(t) dt \quad (1)$$

In eqn. (1), $e(t) = P_{ref} - P_{actual}$ is the error in power flow in the line following a disturbance and t_{sim} is the total time period of simulation. As the objective function of eqn. (1) is nonconvex in nature, GA has been used to minimize F. A basic introduction to GA is given in Section 2.

2.2.1. Genetic algorithm (GA)

Various conventional optimization techniques are available in the literature, which can be used to optimize the controller parameters effectively. The conventional optimization approach, based on gradient descent technique, sometimes gets stuck at local minima giving suboptimal controllers.

Genetic Algorithms (GAs) offer promising potential for optimizing controller parameters and it uses the concept of Darwin's theory of evolution. As per this theory, the existence of all living things is based on the rule of "survival of the fittest" [1]. GA makes use of three fundamental operators, namely, reproduction, crossover, and mutation, which are explained later in this section. The basic principle of GA is to initially form different possible solutions to a problem and then test them for their performance. By using the rule of "survival of the fittest," only a fraction of the better solutions are selected, eliminating the remaining ones. Genetic operators (reproduction, crossover, and mutation) are then applied to the selected solutions resulting in

a new generation of possible solutions (which are expected to perform better than the previous generation). The above steps are repeated until a specific termination criterion is met (such as maximum number of generations, negligible variation in performance or fitness value, etc.). Search space of GA does not confine to a narrow expected region (as in the case of gradient descent), but rather includes a broad spectrum of possible solutions. It is capable of obtaining the global solution of a wide variety of functions such as differentiable or nondifferentiable, linear or nonlinear, continuous or discrete, and analytical or procedural [1]. GA tries to perform an intelligent search for a solution from a nearly infinite number of possible solutions.

2.2.2. Basic tuning procedure

The overall flowchart for optimization using GA is shown in Fig. 4. Initially, a number of populations (N) have been generated for the scaling factors. Each of the populations consists of the binary strings corresponding to the scaling factors G_e , $G_{\Delta e}$, and G_u . These strings are created in a random fashion with the constrain that the values of G_e , $G_{\Delta e}$, and G_u should lie within their specified ranges. The ranges chosen for each of these scaling factors are shown in Table 3. For each of these N sets of values of G_e , $G_{\Delta e}$, and G_u , time domain nonlinear simulation studies have been carried out for evaluating the objective function F of eqn. (1). For this, t_{sim} has been taken as 80 s. Based on the values of the objective function, out of these N possible solutions, the good solutions are retained and the others are eliminated (following the principle of survival of the fittest) [1, 15, 19]. The selected solutions undergo the processes of reproduction, crossover, and mutation to create a new generation of possible solutions (which are expected to perform better than the previous generation). This process of production of a new generation and its evaluation is performed repetitively. The algorithm stops when a predefined maximum number of generations is achieved [1, 15, 19]. The parameters used for GA are shown in Table 4.

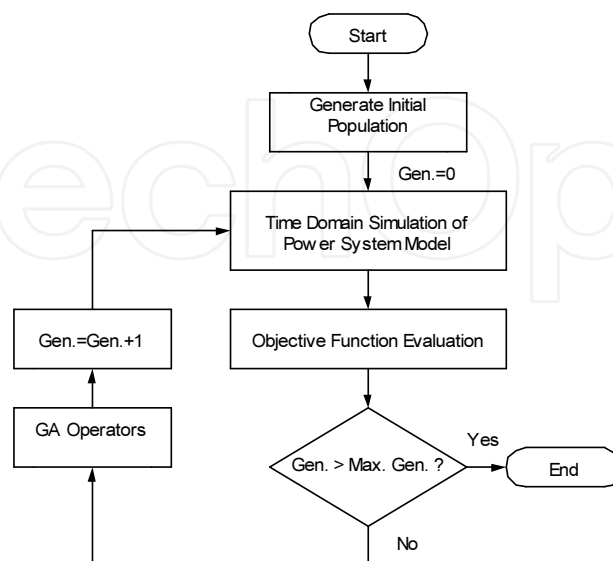


Figure 4. Flowchart of the GA optimization algorithm

Parameters	G_e	$G_{\Delta e}$	G_u
Minimum range	0	0	0
Maximum range	0.5	1	1

Table 3. Ranges for the different scaling factors

Parameter	Value/Type
Maximum generations	50
Population size	25
Mutation rate	0.1
Crossover operator	Scattered

Table 4. Parameters used in genetic algorithm

3. Case studies

The efficacy of the projected STFPIC has been validated on two different multimachine power systems: (a) two-area 4-machine system [16] and (b) 10-machine 39-bus system [16, 19]. The details of these two systems are presented in Section 3.

3.1. Study systems

The schematic diagram of the two-area 4-machine system is shown in Fig. 5. In this system, machines 1, 2 and 3, 4 form two different coherent groups. The three tie lines connect these two coherent areas. As shown in Fig. 5, a TCSC has been installed in the middle of one of these tie lines [17, 19].

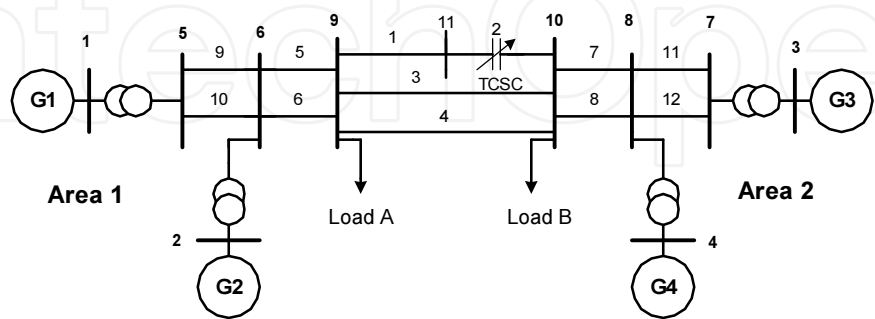


Figure 5. TCSC in a two-area 4-generator system

The one line diagram of the second study system of this work, i.e., the 10-machine 39-bus system is shown in Fig. 6. In this diagram, the generators are identified by their bus numbers

to which they are connected. In this system, a TCSC has been assumed to be installed in the middle of the tie line connecting buses #39 and #36.

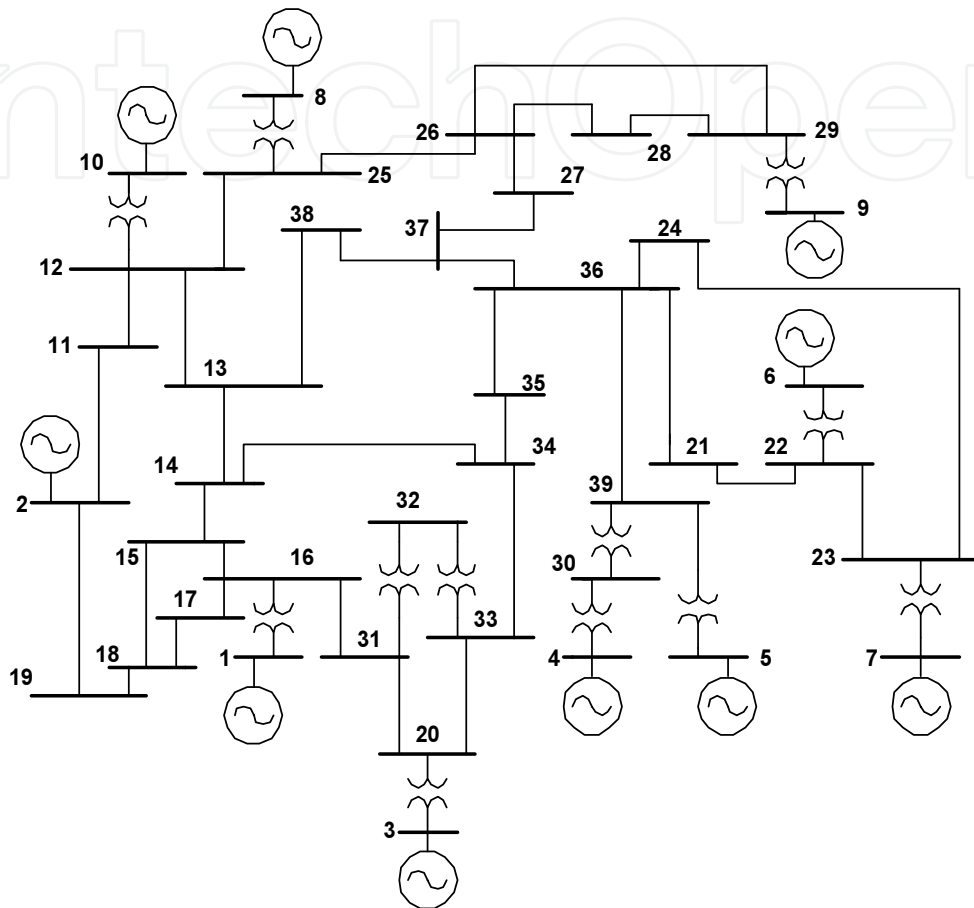


Figure 6. Schematic diagram of 10-machine, 39-bus system

3.2. Simulation results

The effectiveness of the self-tuning fuzzy PI controller (STFPIC) has been studied through detailed nonlinear time domain simulation studies for a three-phase, 5-cycle, solid short-circuit faults on both the test systems shown above. The short-circuit faults have been assumed to occur at $t = 5$ s. The simulation studies have been carried out in the MATLAB/SIMULINK environment [18]. For illustrating the efficacy of the fuzzy controllers developed in this work, results pertaining to two different situations are presented. These situations are: (a) study system with a TCSC controlled by FPIC and (b) study system with a TCSC controlled by STFPIC. For following figures, the rotor angles of the machines have been measured with respect to the Centre of Inertia (COI) reference [16].

3.2.1. Two-area 4-machine system

For this system, initially the short-circuit fault has been assumed to occur at bus 9. The variations of the rotor angles of the two machines (machine 1 and machine 4) and active power flow in line 11–10 (P_{11-10}) for the base case loading condition (the loading condition as described in [16, 21]) are displayed in Figs. 7 and 8, and 9(a), corresponding to scenarios (a) and (b), respectively. In Figs. 7–9, the dotted lines show the variation of rotor angle with TCSC FPIC, whereas the variation of rotor angle with TCSC STFPIC is depicted with solid lines. For implementing STFPIC, K has been set equal to 3. From Figs. 7 to 8, and 9(a), it is observed that there is perceptible improvement in the performance of the system as the oscillations are damped earlier by the STFPIC proposed for TCSC. The variations of the TCSC reactance (X_{TCSC}) are shown in Fig. 9(b) for situations (a) and (b).

To investigate the performance of the TCSC STFPIC controller at enhanced loading condition, simulation studies were carried out by increasing the system loads by 10% from the base case loading condition. The results are shown in Figs. 10 and 11, and 12, corresponding to scenarios (a) and (b), respectively. From Figs. 10 to 11 and 12(a), it is evident that at 10% higher loading, the TCSC STFPIC improves the system performance. Figure 12(b) shows the variation of the TCSC reactance for this case with TCSC FPIC and TCSC STFPIC. It can be observed from Figs. 9(b) and 12(b) that STFPIC has been able to improve the damping with smaller variation in X_{TCSC} as compared to FPIC for the base case and 110% loading conditions respectively with fault at bus 9. The variations of α for fault at bus 9 are shown in Fig. 13, corresponding to both base case and 10% increased loading conditions. From these figures it is observed that for different loading conditions, the TCSC STFPIC always improves the system performance, showing the robustness of the controller.

The performance of the TCSC STFPIC has also been studied with faults at different locations for both base case and 10% increased loading conditions. The simulation results for faults at bus 8 are shown in Figs. 14–19 with the variations of α depicted in Fig. 20. Similarly, the results corresponding to fault at bus 5 are shown in Figs. 21–26 with the variations of α displayed in Fig. 27. From these figures, it is observed that irrespective of the fault location, the TCSC STFPIC always improves the system performance.

3.2.2. 10-machine 39-bus system

In this system, initially the short-circuit fault has been assumed to occur at bus 24. From the fault simulation studies carried out at the base case loading condition (the loading condition as described in [16, 19]), it has been observed that the proposed TCSC STFPIC improved the system performance. Subsequently, a great number of fault studies have been carried out at several increased system loading situations to inspect the effectiveness of the developed STFPIC for improving the performance of the system. After all these studies, it has been found that the TCSC STFPIC helps to further increase the damping for the system. Only a few representative results are presented below to illustrate the effectiveness of the developed STFPIC in further enhancing the performance of the system under study.

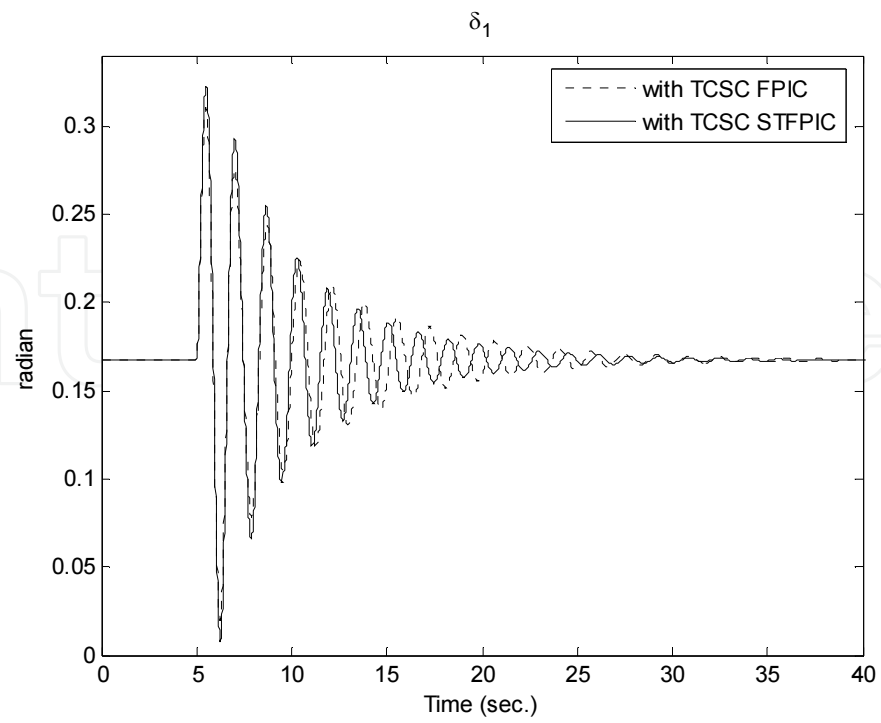


Figure 7. Rotor angle of generator 1 with STFPIC for base case with fault at bus 9

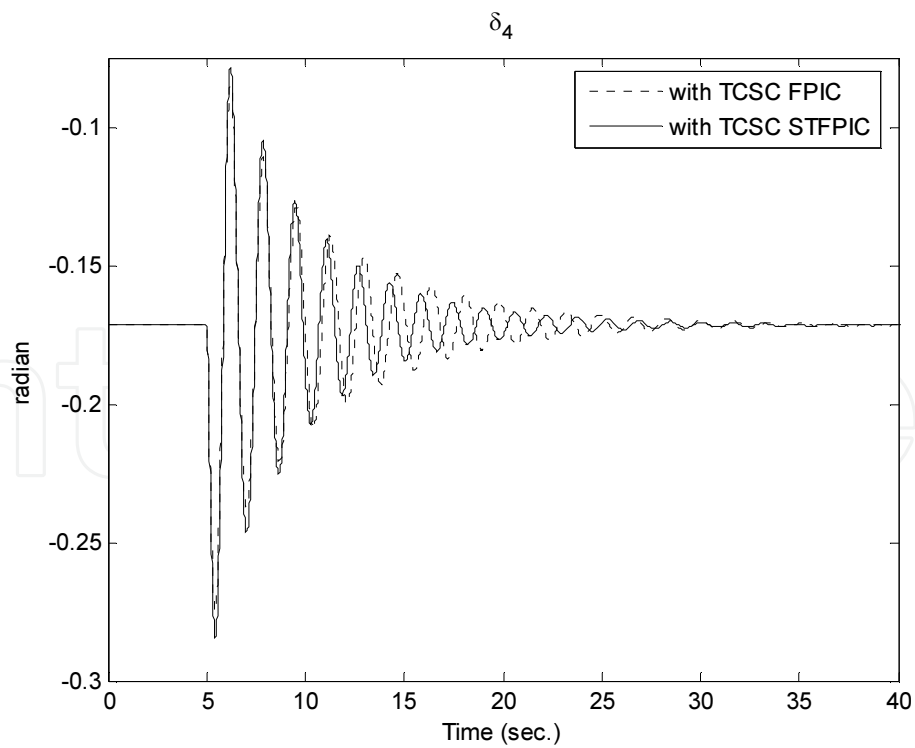


Figure 8. Rotor angle of generator 4 with STFPIC for base case with fault at bus 9

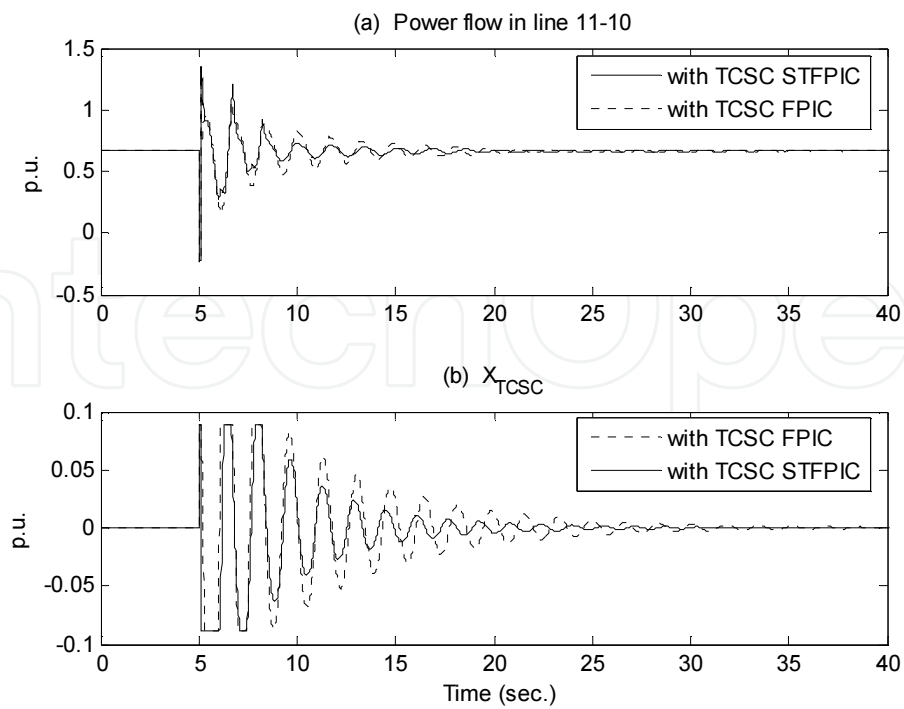


Figure 9. Active power flow in line 11-10 and TCSC capacitive reactance with STFPIC for base case with fault at bus 9

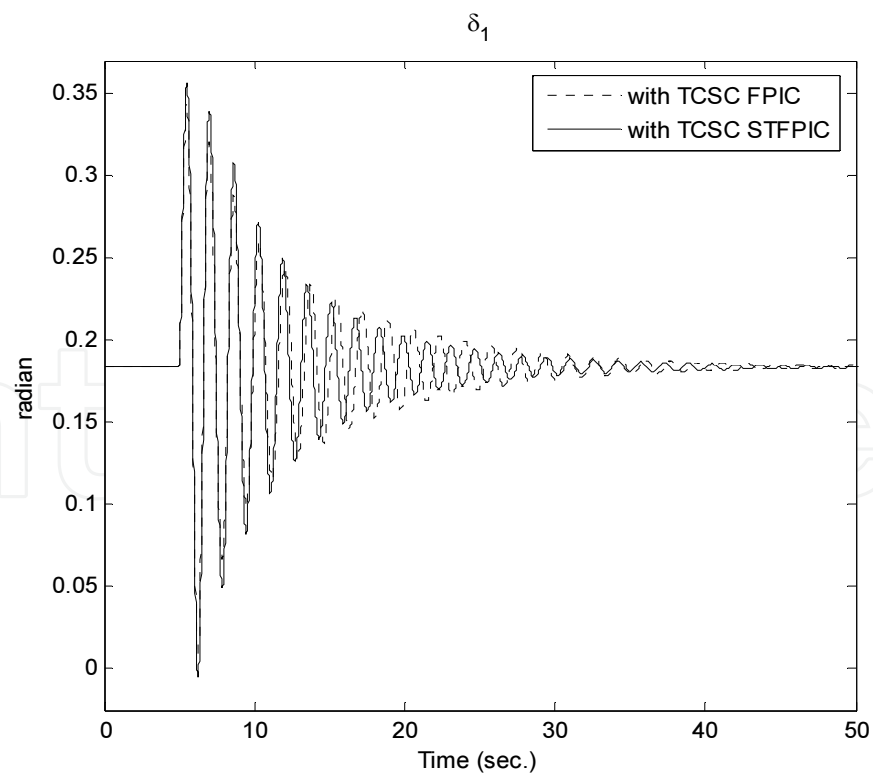


Figure 10. Rotor angle of generator 1 with STFPIC for 110% loading with fault at bus 9

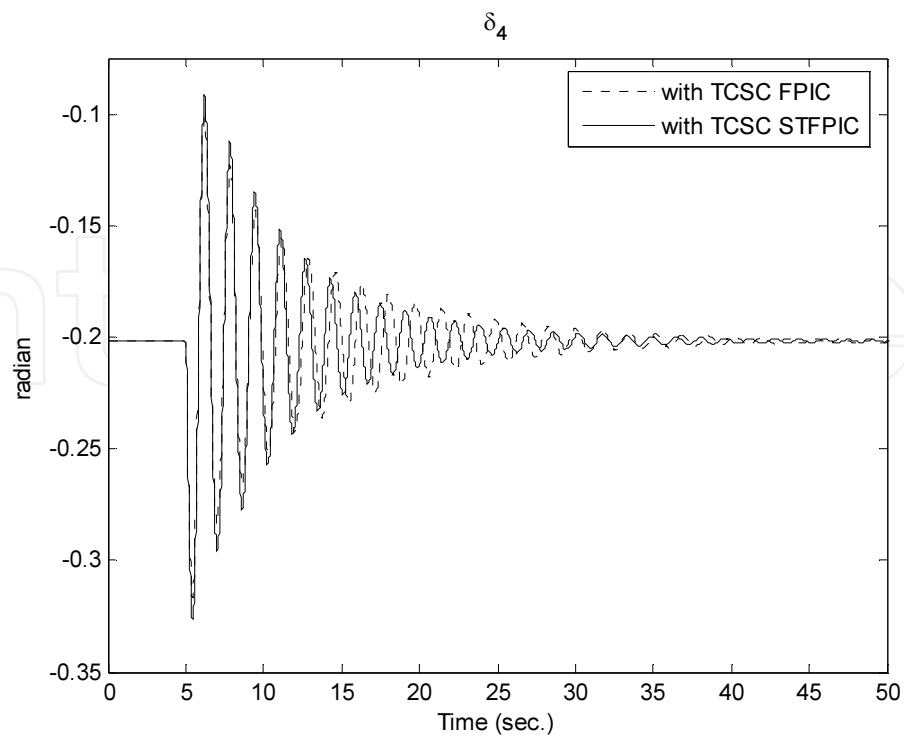


Figure 11. Rotor angle of generator 4 with STFPIC for 110% loading with fault at bus 9

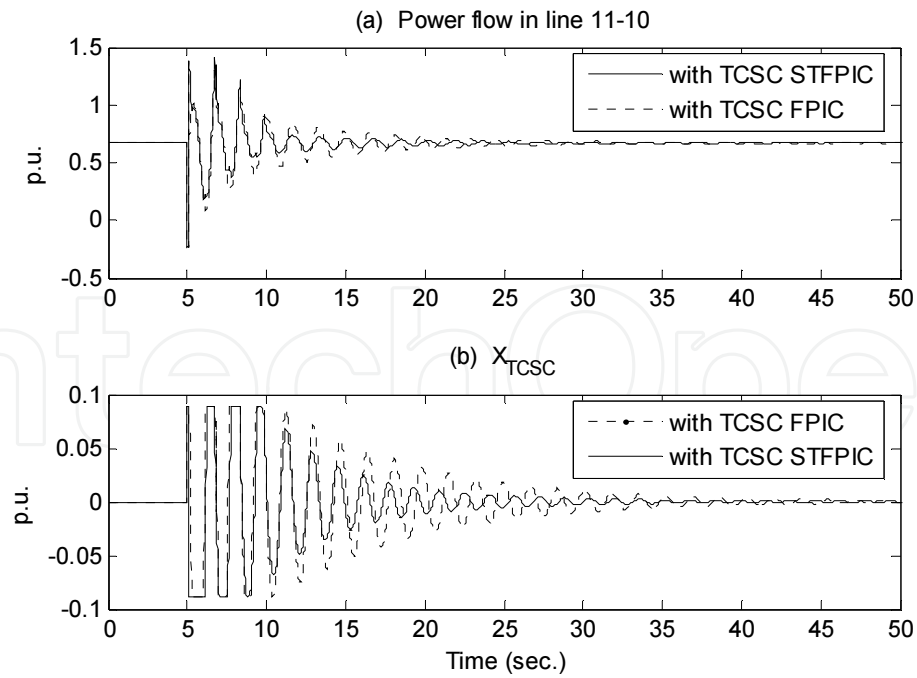


Figure 12. Active power flow in line 11-10 and TCSC capacitive reactance with STFPIC for 110% loading with fault at bus 9

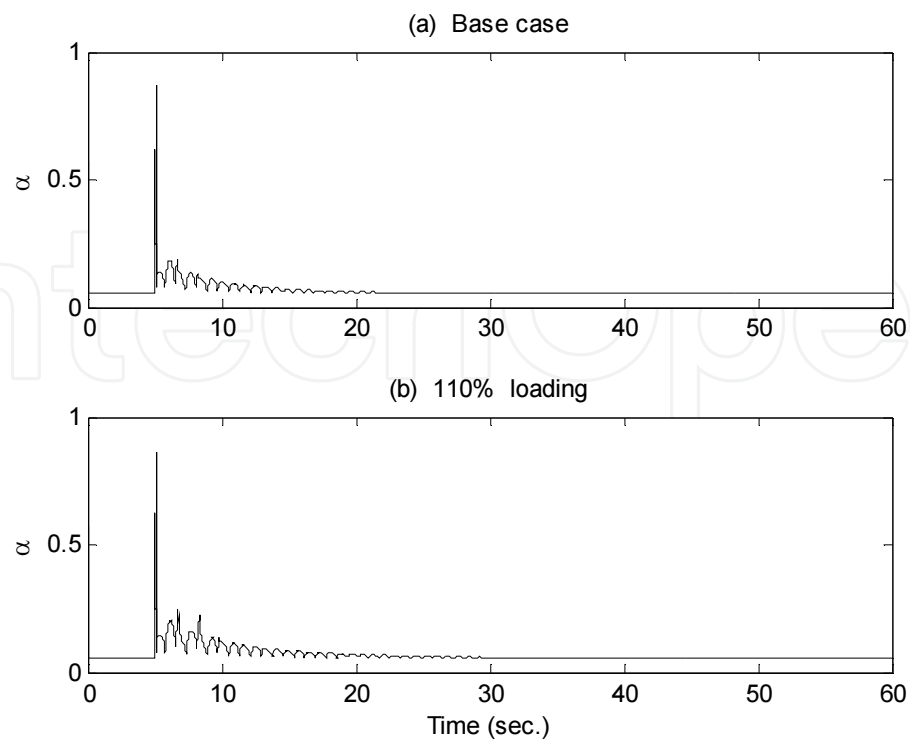


Figure 13. Variation of α for base case and 110% loading with fault at bus 9

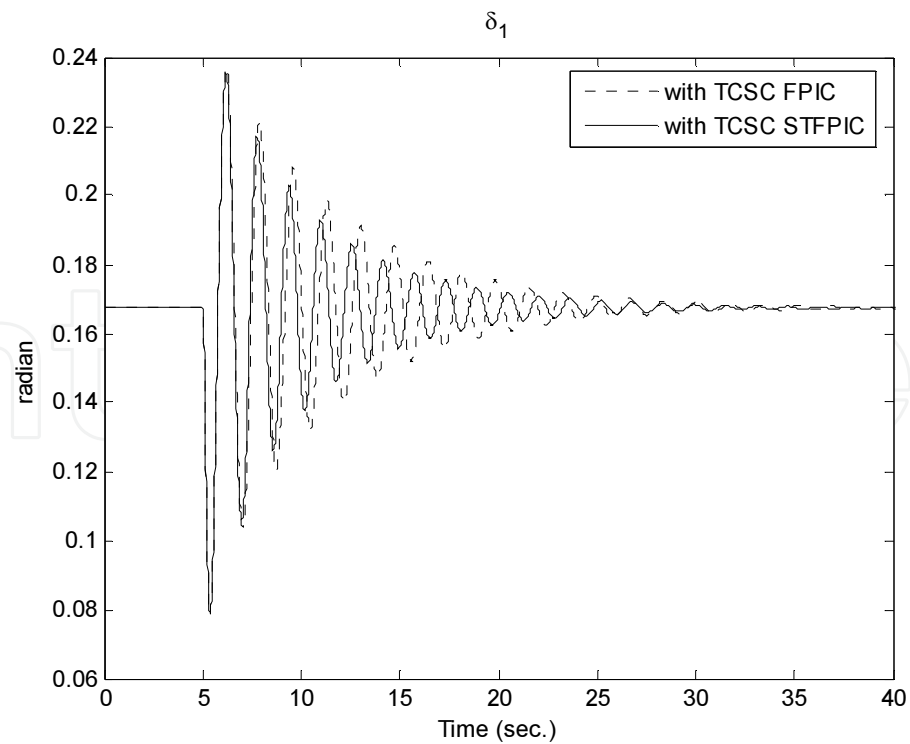


Figure 14. Rotor angle of generator 1 with STFPIC for base case with fault at bus 8

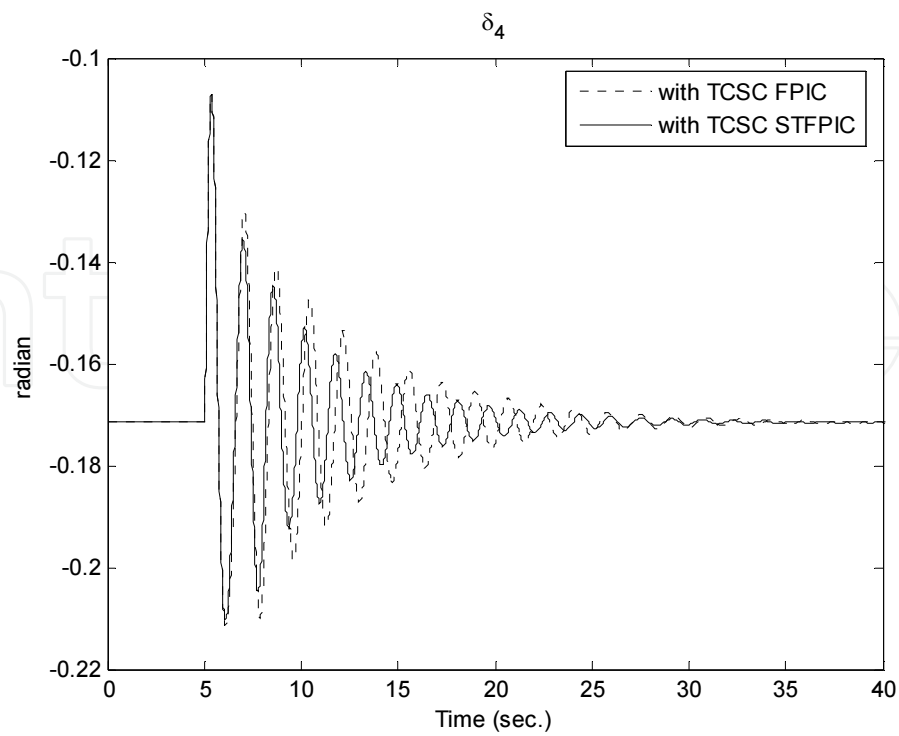


Figure 15. Rotor angle of generator 4 with STFPIC for base case with fault at bus 8

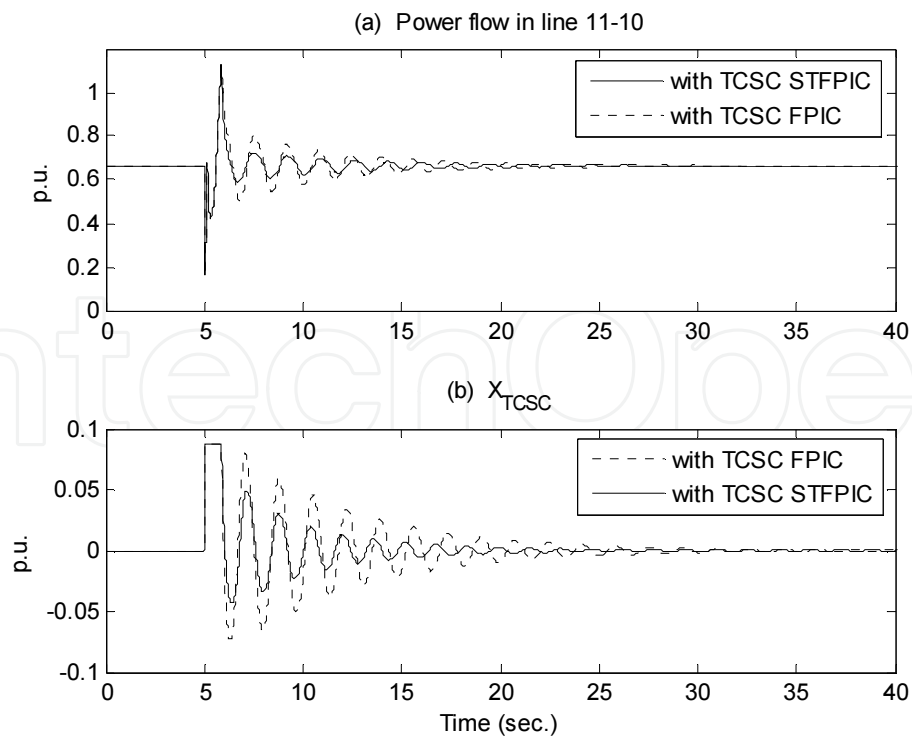


Figure 16. Active power flow in line 11-10 and TCSC capacitive reactance with STFPIC for base case with fault at bus 8

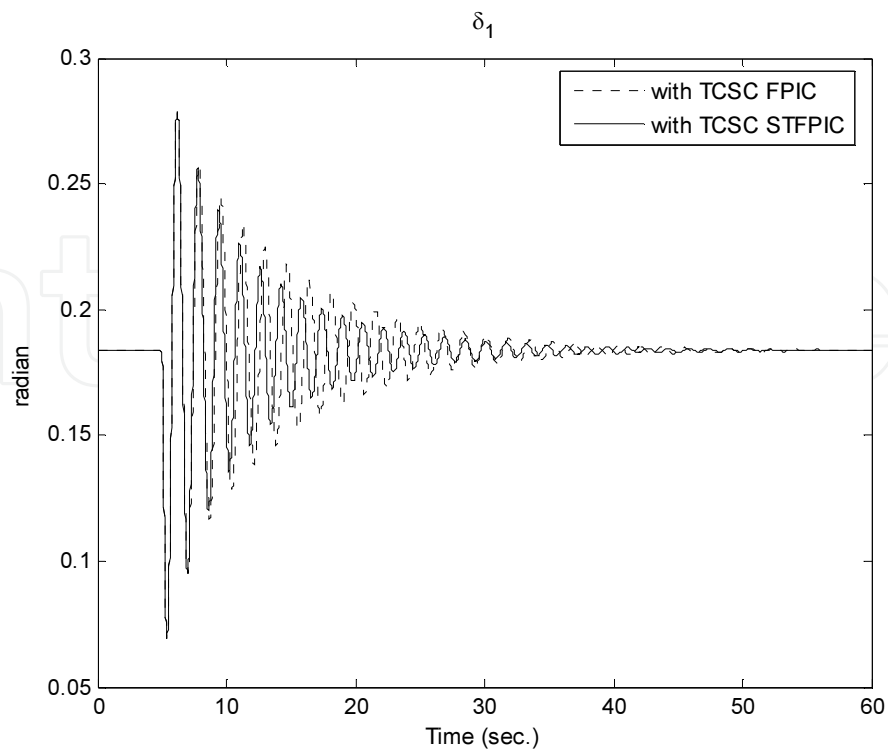


Figure 17. Rotor angle of generator 1 with STFPIC for 110% loading with fault at bus 8

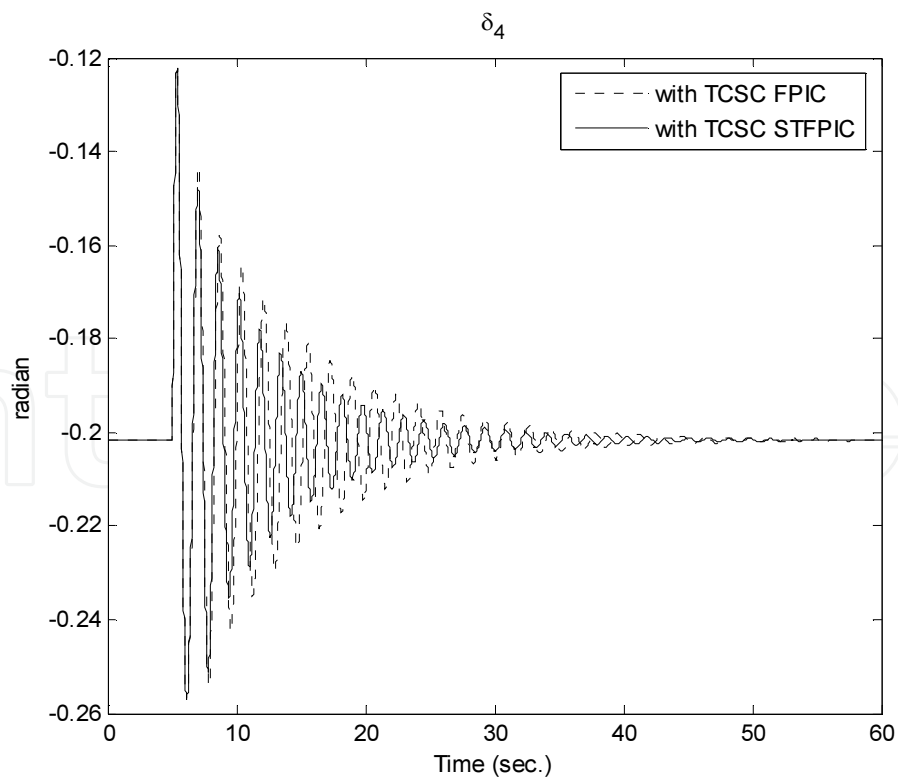


Figure 18. Rotor angle of generator 4 with STFPIC for 110% loading with fault at bus 8

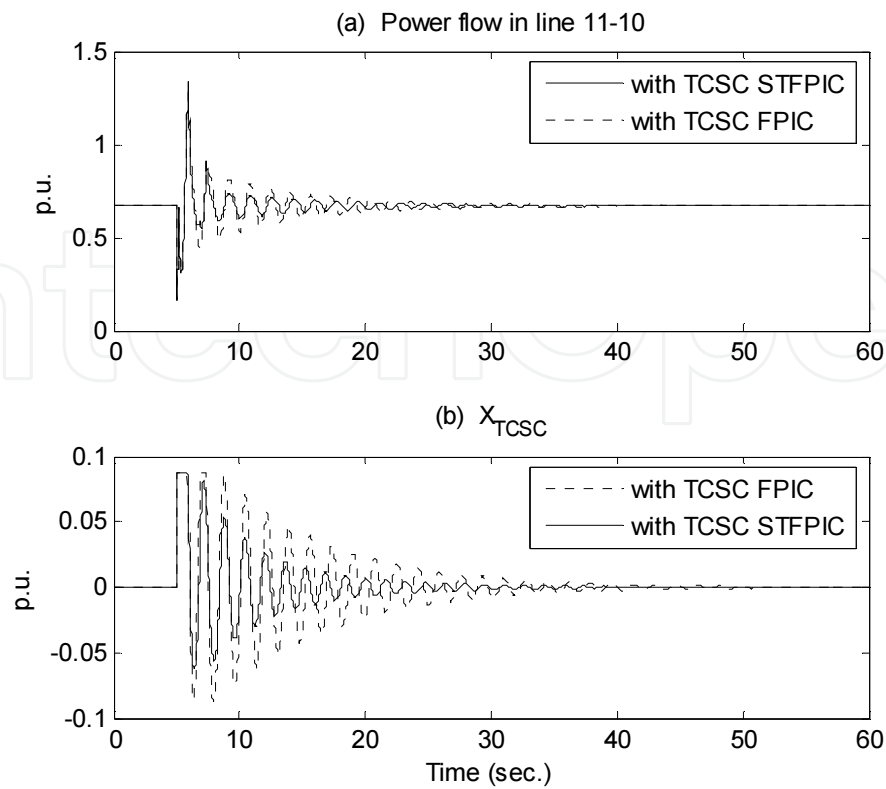


Figure 19. Active power flow in line 11-10 and TCSC capacitive reactance with STFPIC for 110% loading with fault at bus 8

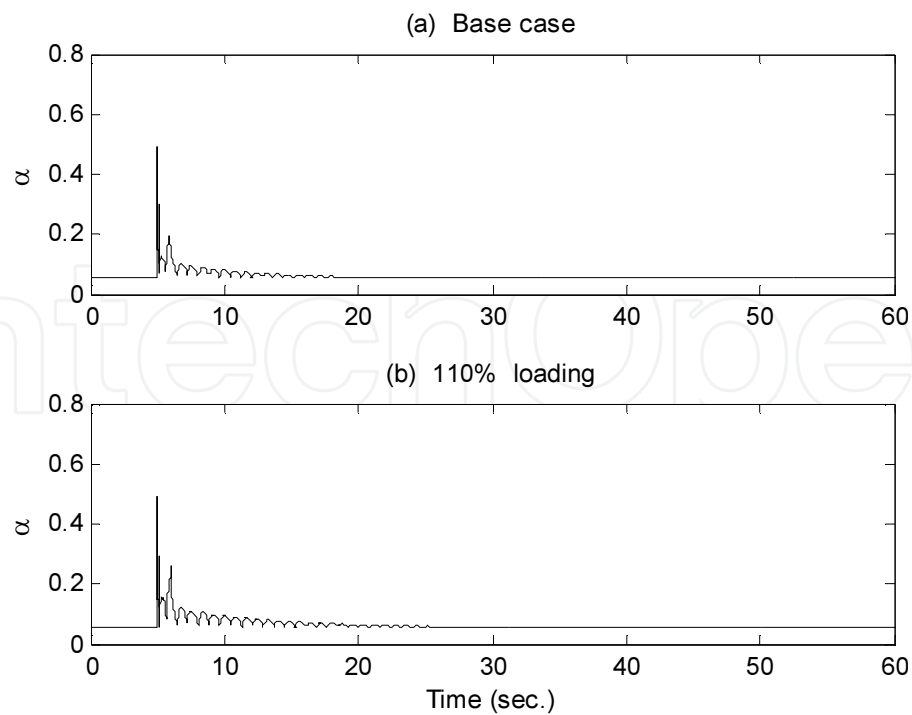


Figure 20. Variation of α for base case and 110% loading with fault at bus 8

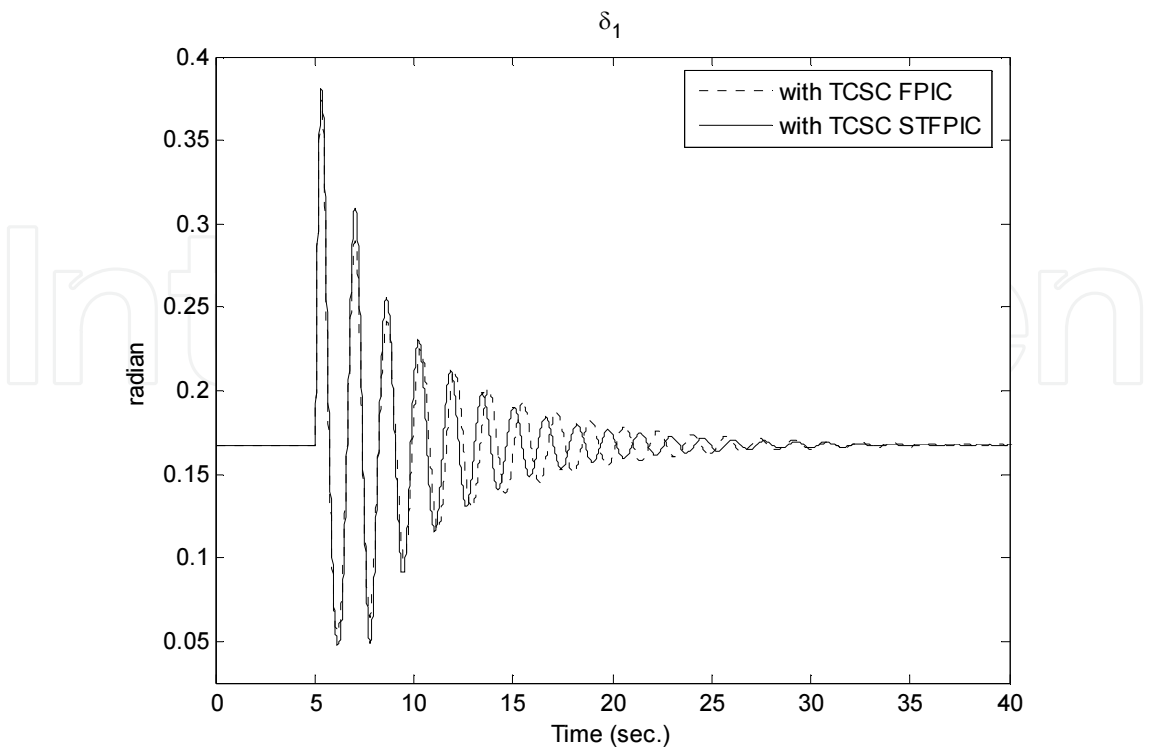


Figure 21. Rotor angle of generator 1 with STFPIC for base case with fault at bus 5

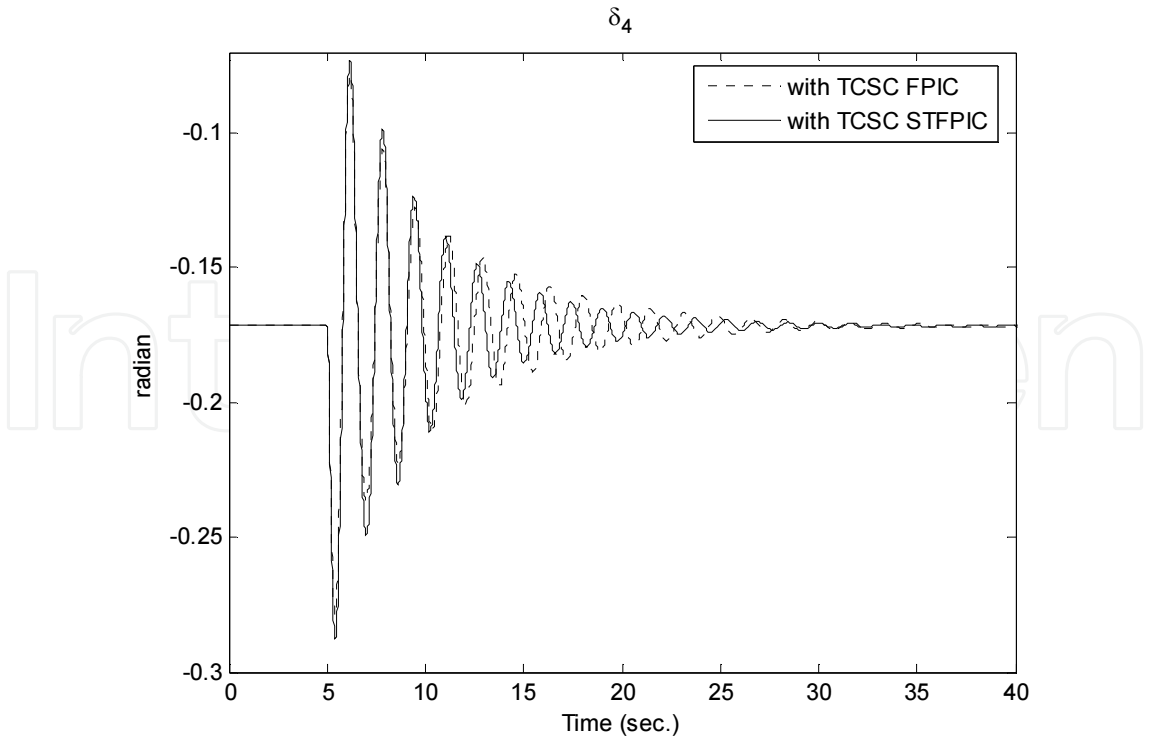


Figure 22. Rotor angle of generator 4 with STFPIC for base case with fault at bus 5

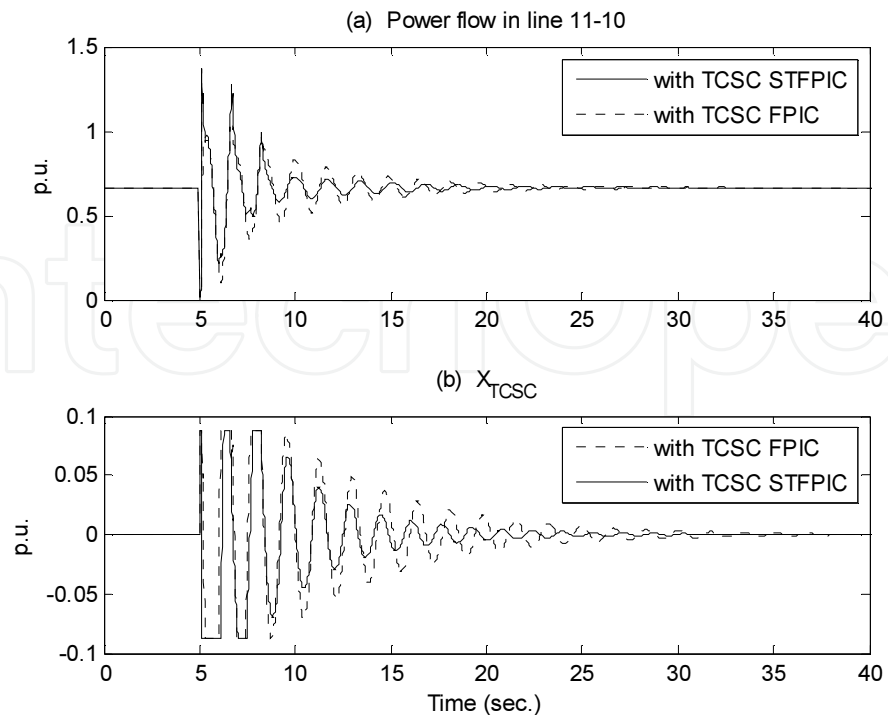


Figure 23. Active power flow in line 11-10 and TCSC capacitive reactance with STFPIC for base case with fault at bus 5

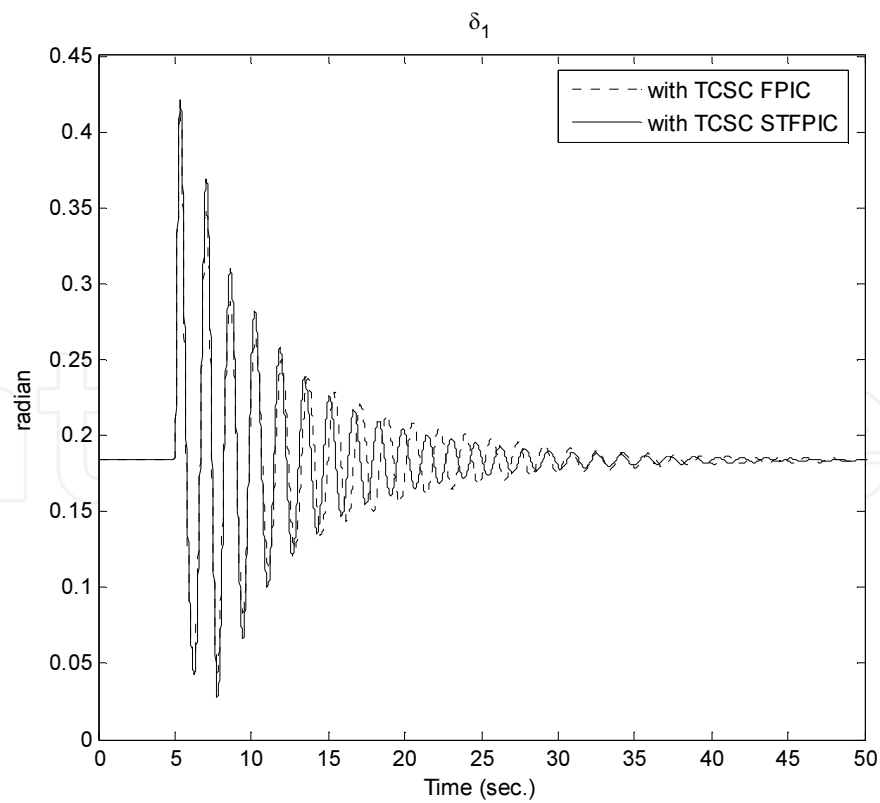


Figure 24. Rotor angle of generator 1 with STFPIC for 110% loading with fault at bus 5

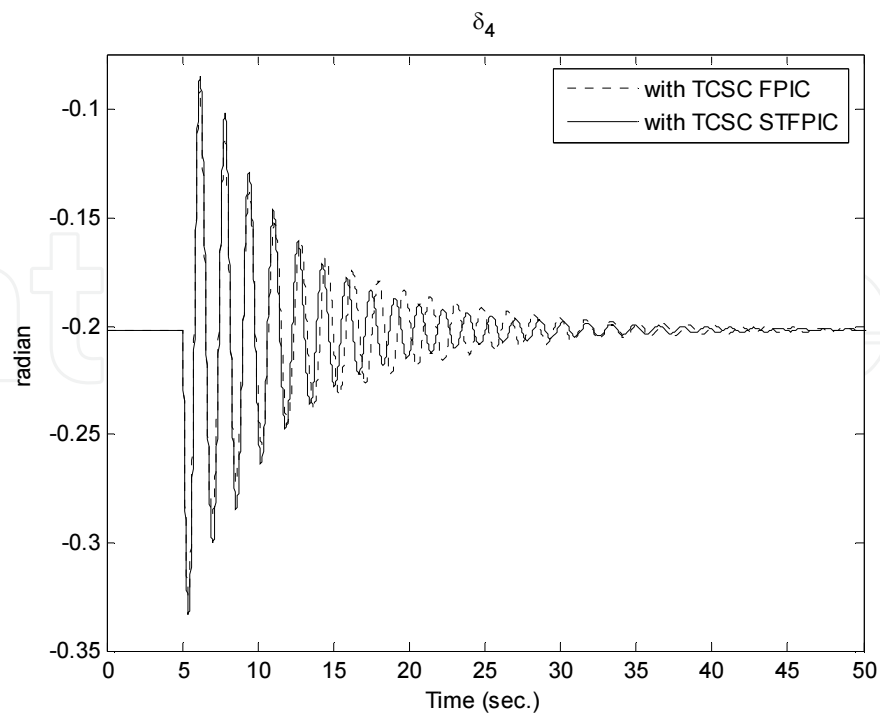


Figure 25. Rotor angle of generator 4 with STFPIC for 110% loading with fault at bus 5

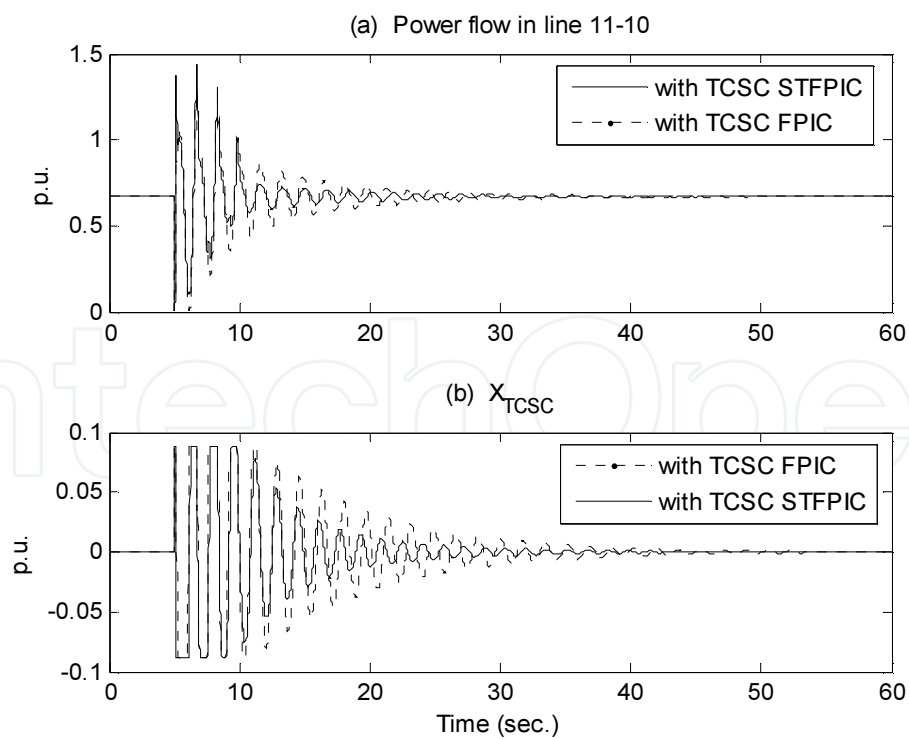


Figure 26. Active power flow in line 11-10 and TCSC capacitive reactance with STFPIC for 110% loading with fault at bus 5

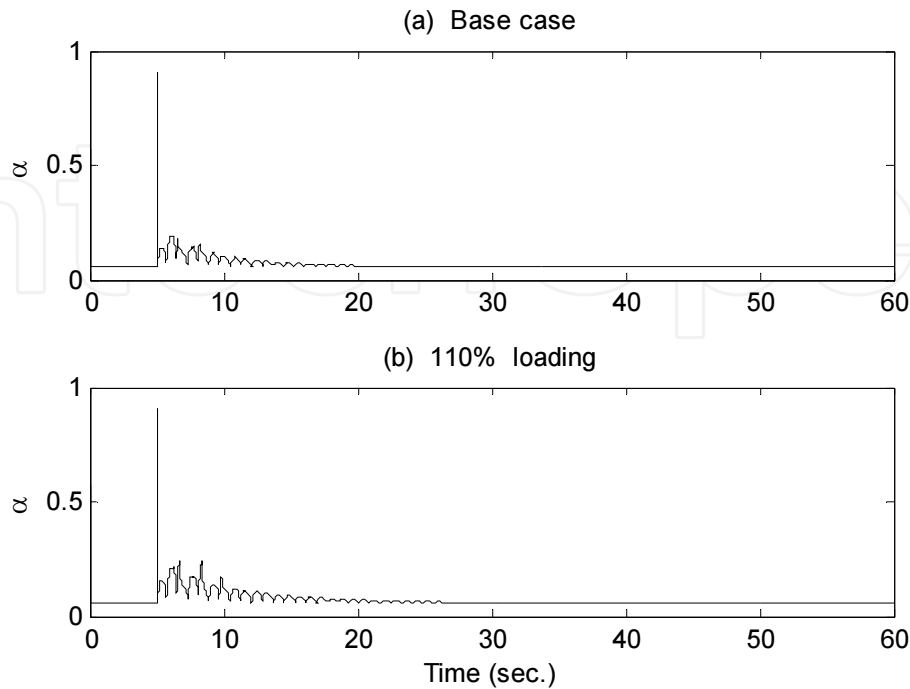


Figure 27. Variation of α for base case and 110% loading with fault at bus 5

The performance of the TCSC STFPIC at 160% loading condition (w.r.t. the base case loading condition as described in [16]) is shown in Figs. 28–3. From these figures it is observed that at 160% loading condition, substantial improvement in the system performance is achieved by the proposed TCSC STFPIC. For implementing STFPIC, K has been set equal to 10. When the system loading level is further increased to 175% of the base case loading condition, the proposed TCSC STFPIC is able to improve the system performance significantly as illustrated in Figs. 32 and 33. It can be noticed from Figs. 31(b) and 33(b) that STFPIC has been able to improve the damping with smaller variation in X_{TCSC} as compared to FPIC for 160% and 175% loading conditions, respectively, with fault at bus 24. The variations of α for fault at bus 24 are shown in Fig. 34, corresponding to 160% and 175% loading conditions.

Additional fault simulation studies have also been carried out with faults at different locations, for various loading conditions, to investigate the robustness of the proposed control scheme. For all the simulation studies carried out in this work, the TCSC STFPIC has improved the damping of the system (as compared to the damping obtained with FPIC). Some representative results obtained at 175% loading condition are shown in Figs. 35–38. Figures 35 and 36 show the results for fault at bus 38, while the results for fault at bus 23 are shown in Figs. 37 and 38. These results amply demonstrate that the TCSC STFPIC helps to enhance the damping capability of the system quite significantly. The variations of α for fault at bus 38 and 23 corresponding to 175% loading conditions are shown in Fig. 39. Variation of the rotor angle of generator 3 only has been shown for 175% loading condition.

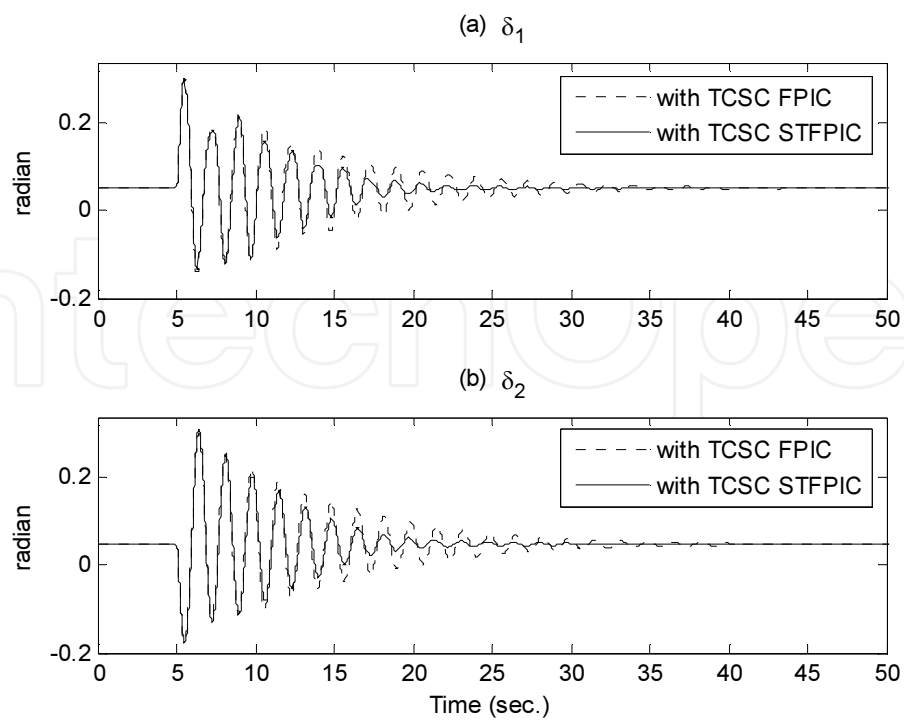


Figure 28. Rotor angles of generator 1 and 2 with STFPIC for 160% loading with fault at bus 24

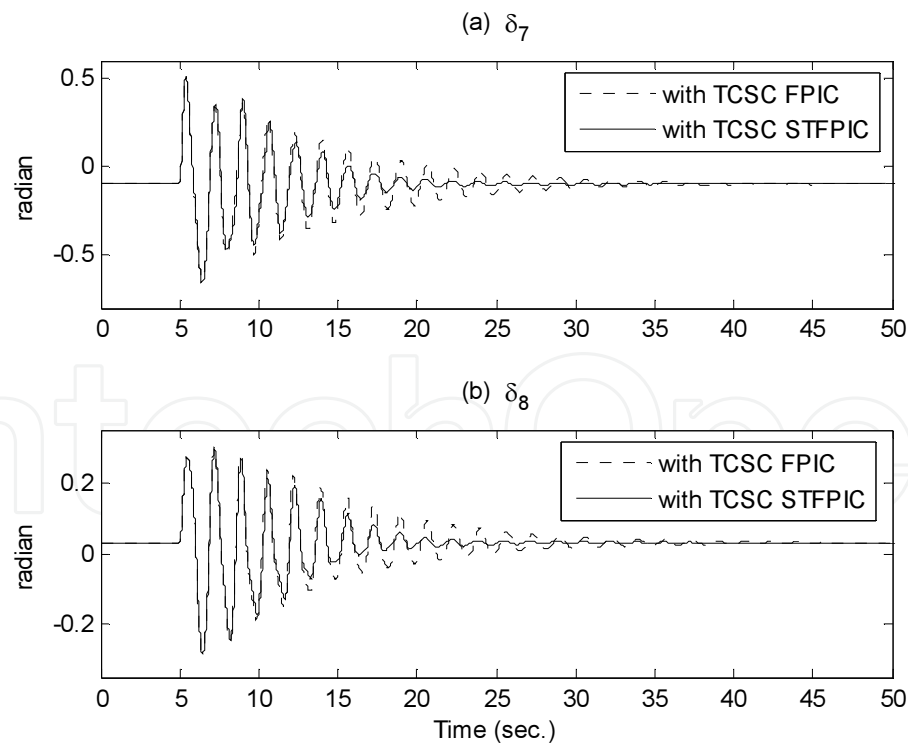


Figure 29. Rotor angles of generator 7 and 8 with STFPIC for 160% loading with fault at bus 24

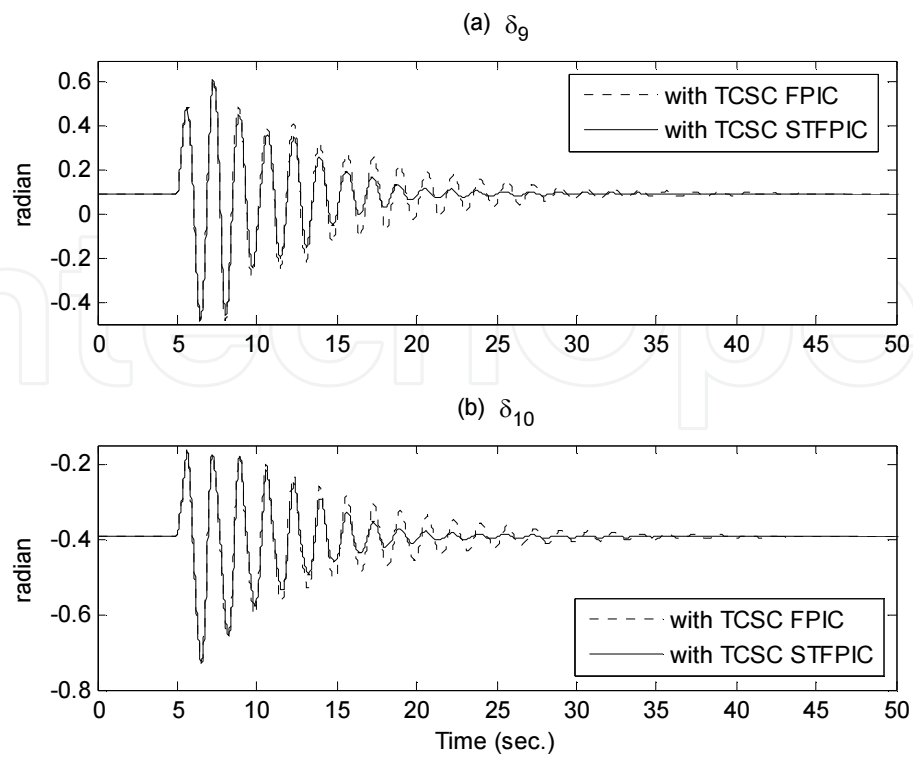


Figure 30. Rotor angles of generator 9 and 10 with STFPIC for 160% loading with fault at bus 24

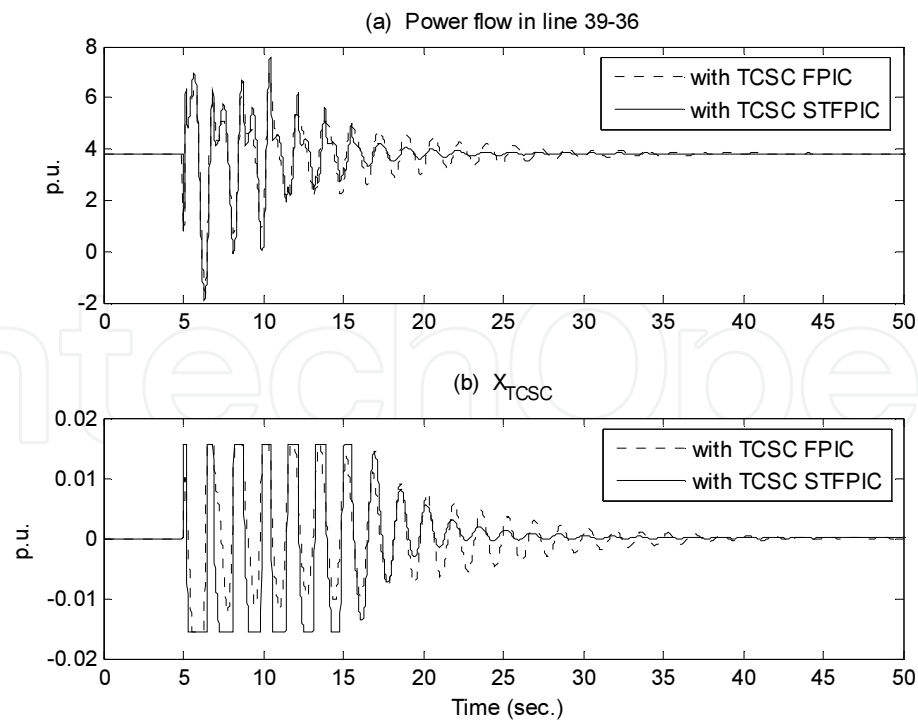


Figure 31. Active power flow in line 39-36 and TCSC capacitive reactance with STFPIC for 160% loading with fault at bus 24

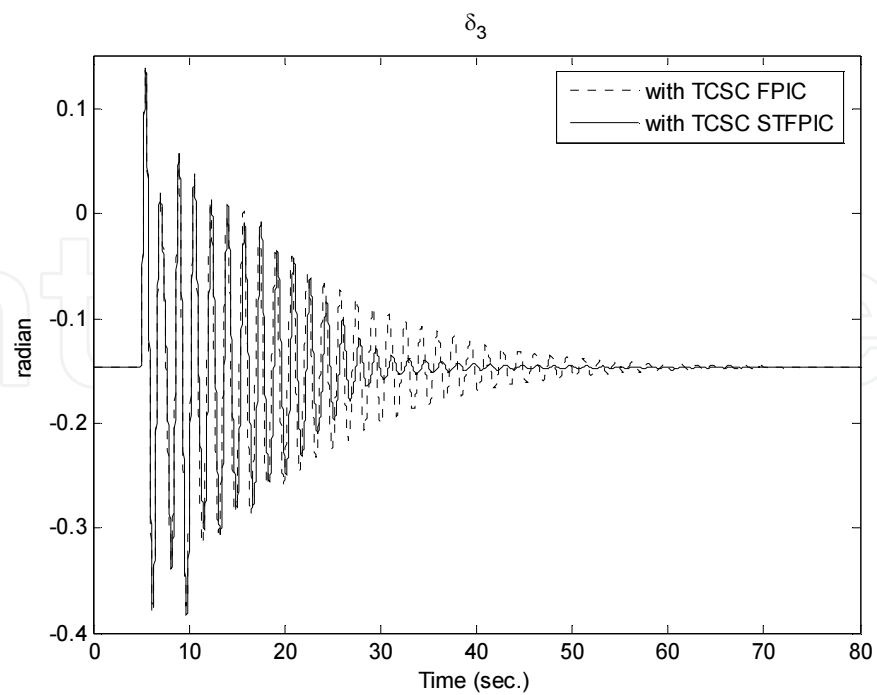


Figure 32. Rotor angle of generator 3 for 175% loading with STFPIC for fault at bus 24

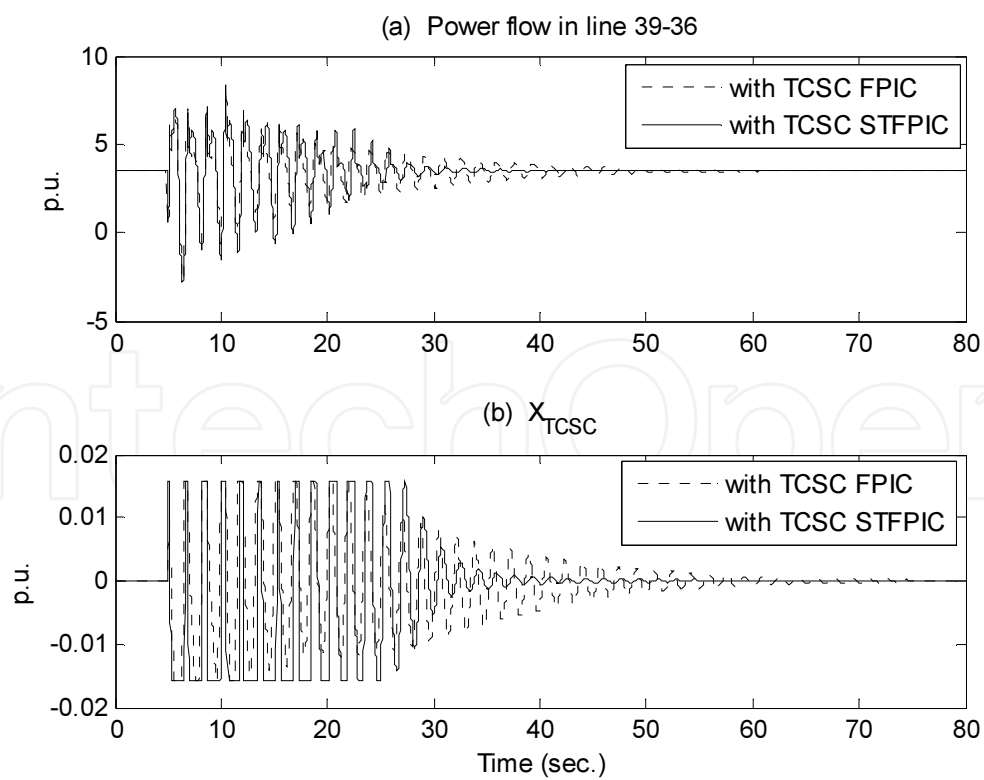


Figure 33. Active power flow in line 39-36 and TCSC capacitive reactance with STFPIC for 175% loading with fault at bus 24

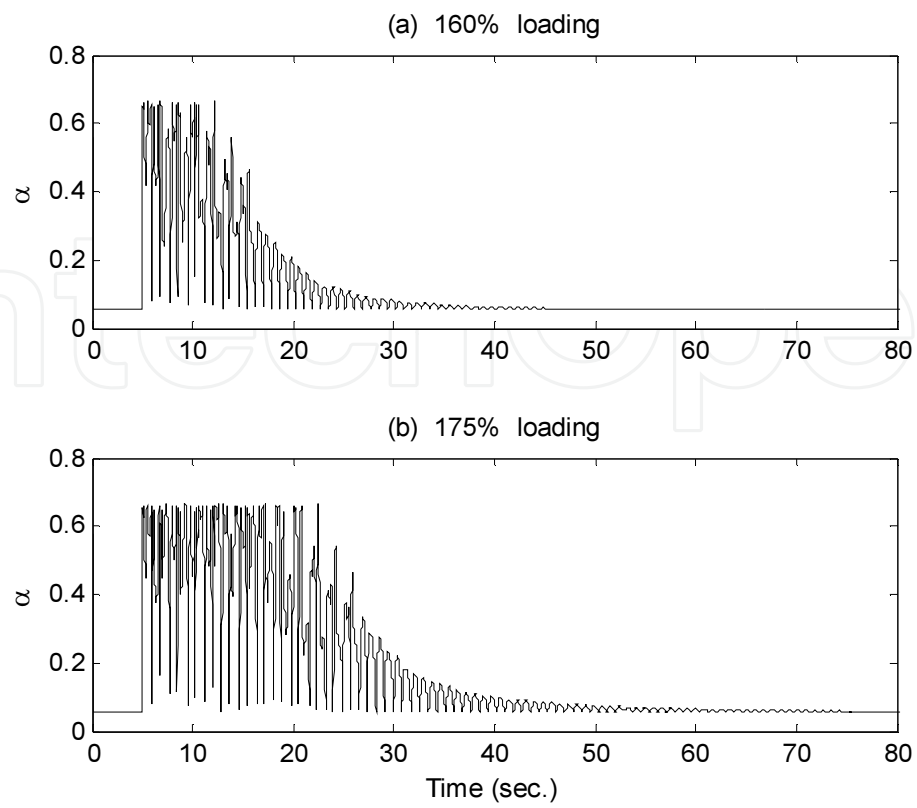


Figure 34. Variation of α for 160% and 175% loading with fault at bus 24

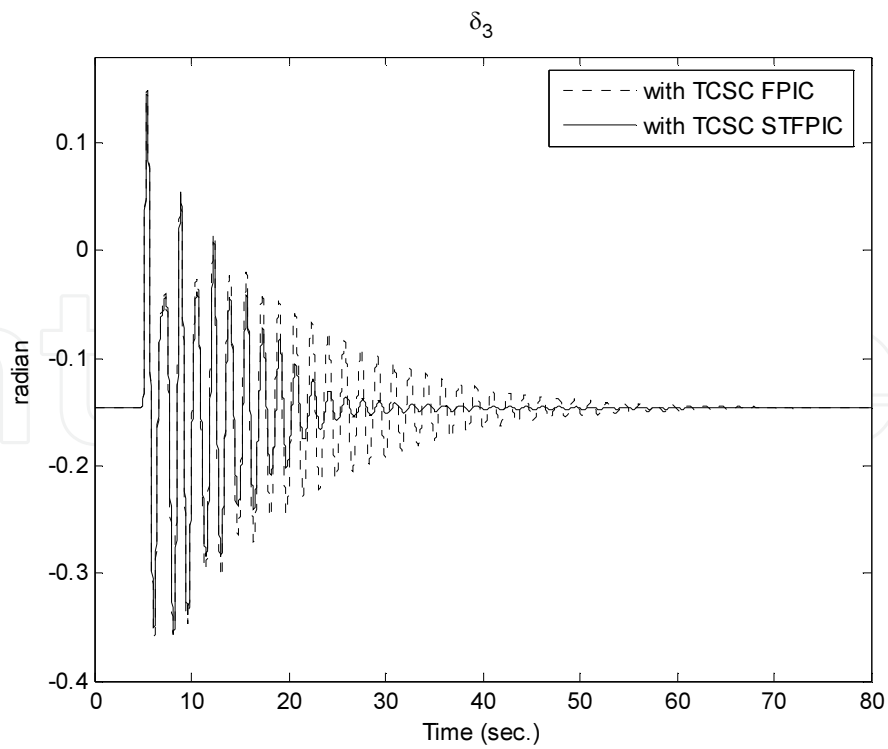


Figure 35. Rotor angle of generator 3 for 175% loading with STFPIC for fault at bus 38

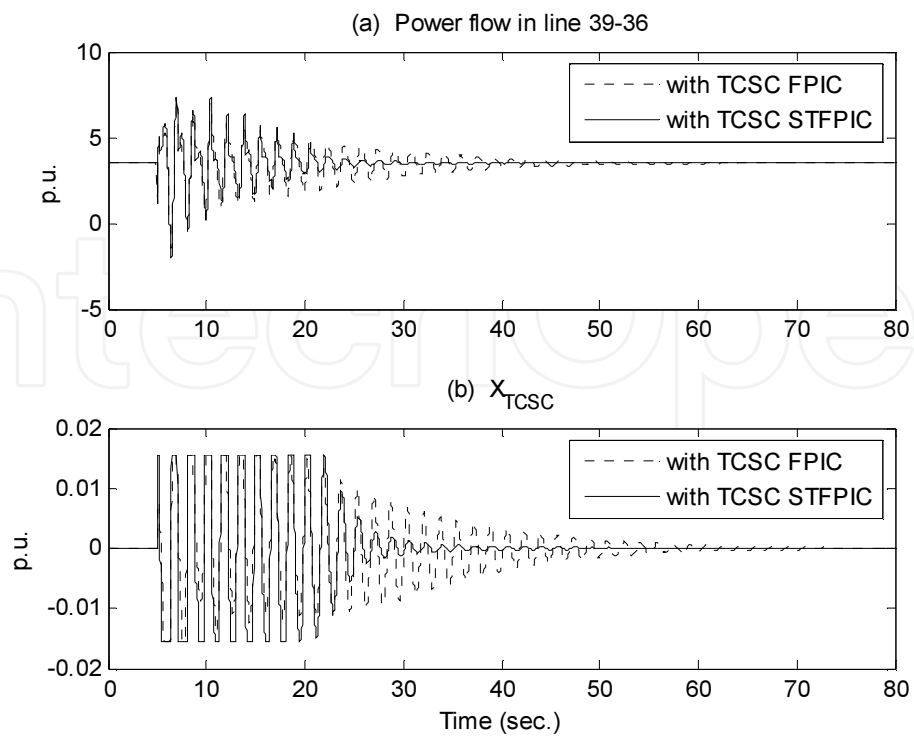


Figure 36. Active power flow in line 39-36 and TCSC capacitive reactance with STFPIC for 175% loading with fault at bus 38

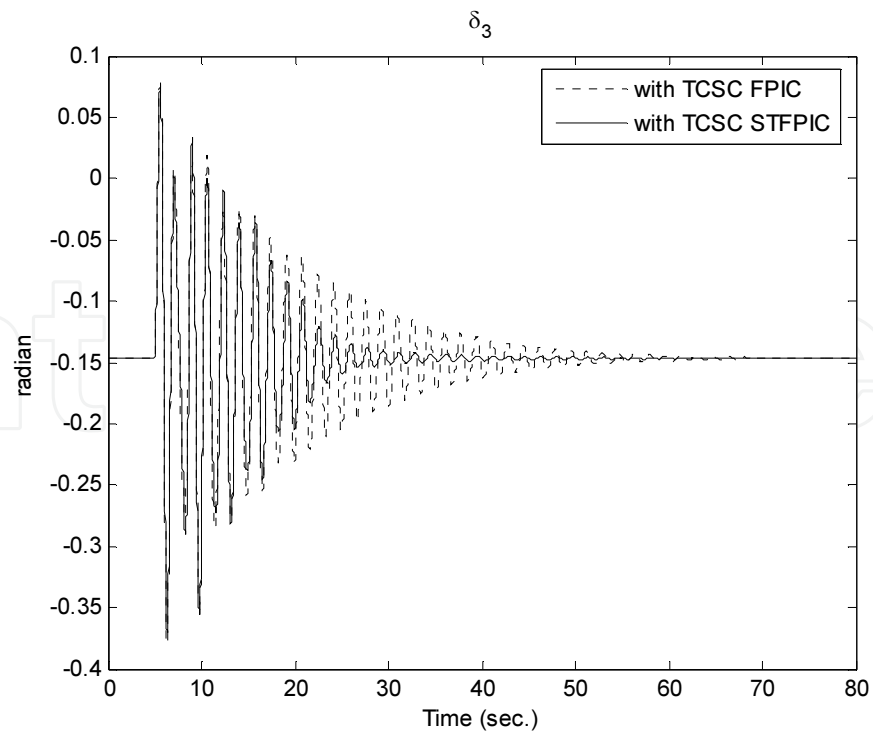


Figure 37. Rotor angle of generator 3 for 175% loading with STFPIC for fault at bus 23

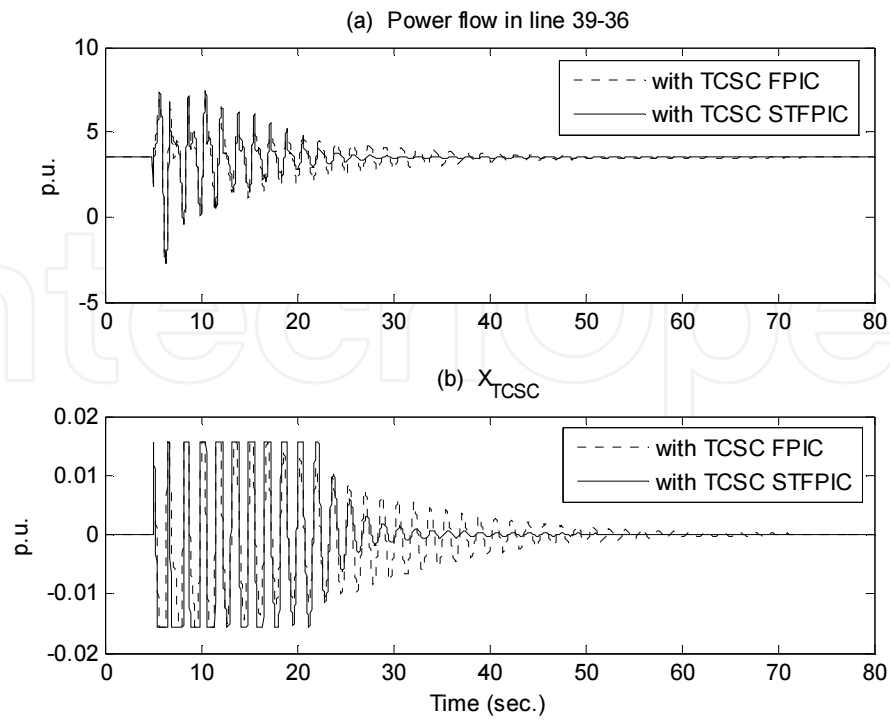


Figure 38. Active power flow in line 39-36 and TCSC capacitive reactance with STFPIC for 175% loading with fault at bus 23

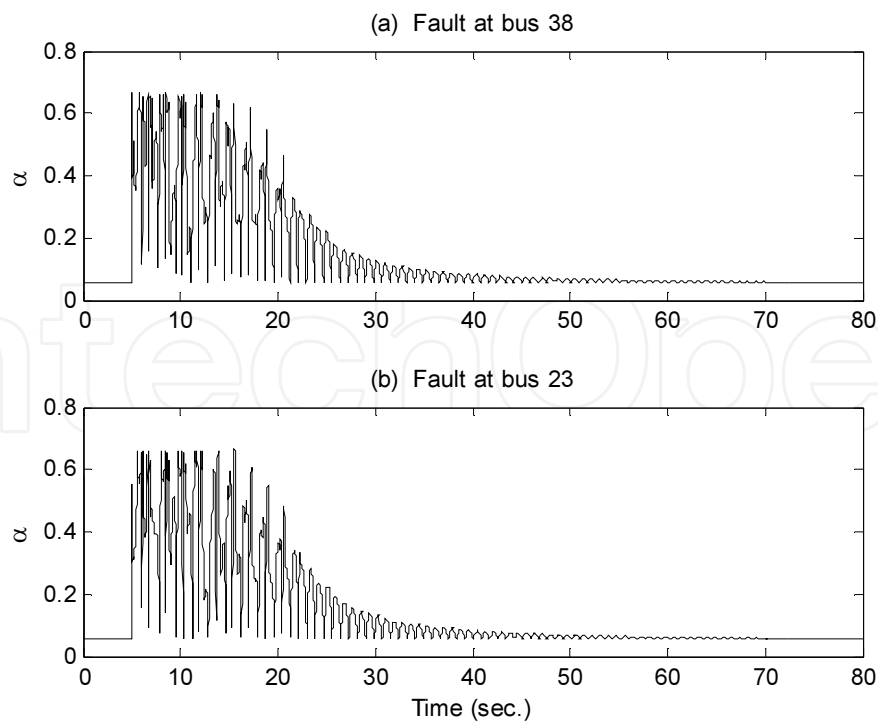


Figure 39. Variation of α for 175% loading with fault at bus 38 and 23

4. Conclusion

In this chapter, a methodology has been discussed for designing an adaptive FPIC using self-tuning mechanism (STFPIC). The simulation studies have been carried out in MATLAB/SIMULINK environment. A large number of fault cases involving three-phase, 5-cycle, solid short-circuit faults have been investigated on both the study systems discussed earlier, for different loading conditions as well as different fault locations. From the simulation results, it has been observed that the proposed STFPIC for TCSC improves the system stability significantly.

In this scheme, the output scaling factor of the FPIC is modulated on-line by a gain updating factor (α) whose value is determined by a second fuzzy logic scheme. Hence, the output scaling factor of STFPIC does not remain fixed while the controller is in operation, rather it is modified at each sampling instant by a gain updating factor " α ."

The effectiveness of the developed STFPIC has been studied through detailed nonlinear fault simulation studies on both the test systems. The same fault cases have also been studied with FPIC and the performances obtained by STFPIC have been compared to those obtained by FPIC. From the comparative simulation results, it can be observed that the proposed STFPIC for TCSC improves the system damping further (as compared to the system damping obtained by FPIC).

The STFPIC, which comprises of two fuzzy logic blocks, requires increased computational time and memory for implementation. Therefore, it would be advantageous to reduce the size of the rule base so that the requirement of computational time and memory for implementation is reduced.

The performances of FPIC and STFPIC discussed in this chapter have been tested with predesigned scaling factors. However, in real life, the operating point in the system may vary continuously, thereby making it almost impossible to predesign the scaling factors. Thus, it is necessary to determine the scaling factors on-line based on some suitable measurements. In future, such a scheme for on-line determination of the scaling factors will be described.

The proposed self-tuning philosophy may possibly be applied for the tuning of input SFs or for tuning of both input and output SFs simultaneously, which may lead to superior performances of the FLCs.

The efficacy and suitability of the proposed techniques can be explored further by applying it to larger power systems for different operating conditions.

Author details

Salman Hameed*

Address all correspondence to: hameeddee@gmail.com

Department of Electrical Engineering, Aligarh Muslim University, Aligarh, India

References

- [1] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*: Pearson Education Asia, 200
- [2] J. Yan, M. Ryan and J. Power, *Using Fuzzy Logic Towards intelligent systems*, Prentice Hall International (UK) Ltd, 1994.
- [3] D. Driankov, H. Hellendoorn and M. Reinfrank, *An Introduction to Fuzzy Control*, Springer-Verlag, New York, 1993.
- [4] S. Z. He, S. Tan, F. L. Xu and P. Z. Wang, "Fuzzy self-tuning of PID controller", *Fuzzy Sets and Systems*, vol. 56, no. 1, May 1993, pp. 37–46.
- [5] M. Maedo and S. Murakami, "A self-tuning fuzzy controller", *Fuzzy Sets and Systems*, vol. 51, no. 1, October 1992, pp. 29–40.
- [6] S. Shao, "Fuzzy self-organizing controller and its application for dynamic processes", *Fuzzy Sets and Systems*, vol. 26, no. 2, May 1988, pp. 151–164.
- [7] Y. Park, U. Moon and K. Y. Lee, "A self-organizing fuzzy logic controller for dynamic systems using a fuzzy auto-regressive moving average (FARMA) model", *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 1, February 1995, pp. 75–82.
- [8] R. K. Mudi and N. K. Pal, "A self-tuning fuzzy PI controller", *Fuzzy Sets and Systems*, vol. 115, no. 2, October 2000, pp. 327–338.
- [9] C. T. Chao and C. C. Teng, "A PD-like self-tuning fuzzy controller without steady-state error", *Fuzzy Sets and Systems*, vol. 87, no. 2, April 1997, pp. 141–154.
- [10] Z. W. Woo, H. Y. Chung and J. J. Lin, "A PID type fuzzy controller with self-tuning scaling factors", *Fuzzy Sets and Systems*, vol. 115, no. 2, October 2000, pp. 321–326.
- [11] Y. P. Singh, "A modified self-organizing controller for real-time process control applications", *Fuzzy Sets and Systems*, vol. 96, no. 2, June 1998, pp. 147–159.
- [12] Y. Zhao and E. G. Collins, Jr., "Fuzzy PI control design for an industrial weigh belt feeder", *IEEE Transactions on Fuzzy Systems*, vol. 11, no. 3, June 2003, pp. 311–319.
- [13] R. K. Mudi and N. R. Pal, "A robust self-tuning scheme for PI- and PD-type fuzzy controllers", *IEEE Transactions on Fuzzy Systems*, vol. 7, no. 1, February 1999, pp. 2–16.
- [14] K. Pal, R. K. Mudi and N. R. Pal, "A new scheme for fuzzy rule-based system identification and its application to self-tuning fuzzy controllers", *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 32, no. 4, August 2002, pp. 470–482.
- [15] A. Popov, *Genetic Algorithm for Optimization*. web-site: <http://automatics.hit.bg/>

- [16] K. R. Padiyar, *Power System Dynamics - Stability and Control*, Interline Publishing Ltd., Bangalore, India 1996.
- [17] B. Chowdhury, R. Majumder and B. C. Pal, "Application of multiple-model adaptive control strategy for robust damping of interarea oscillations in power system", *IEEE Transactions on Control Systems Technology*, vol. 12, no. 5, September 2004, pp. 727-736.
- [18] MATLAB/SIMULINK Software, Version 7.0.1, Release 14, The Math Works Inc.
- [19] S. Hameed, "A self-tuning fuzzy PI controller for TCSC to improve power system stability", *Elect Power Sys Res*, 10/2008
- [20] S. Hameed, "Power system stability enhancement using reduced rule base self-tuning fuzzy PI controller for TCSC", *IEEE PES T&D*, 2010, 04/2010
- [21] S. Hameed, "Reduced rule base self-tuning fuzzy PI controller for TCSC", *Int J Elect Power Energy Sys*, 11/2010

IntechOpen