



## CONTROLLER TUNING APPROACH WITH TLBO ALGORITHM FOR THE AUTOMATIC VOLTAGE REGULATOR SYSTEM

Banu ATAŞLAR-AYYILDIZ<sup>1,\*</sup>, Oğuzhan KARAHAN<sup>1</sup>

<sup>1</sup> Department of Electrical and Communications Engineering, Kocaeli University, Kocaeli, Turkey

### ABSTRACT

In this paper, an optimal parameter set tuning method for proportional-integral-derivative (PID) controller and fractional order PID controller is proposed using teaching-learning based optimization (TLBO) algorithm. During the optimization of the PID and FOPID controller parameters, an objective function consisting of overshoot, rise time, settling time and steady state error is formulated to achieve a satisfactory trade-off between the dynamic response characteristics. TLBO algorithm is used as the optimizer to find the best parameters of the proposed controllers. The designed PID and FOPID controllers are applied to an automatic voltage regulator (AVR) system. The performances of the proposed controllers are confirmed by comparing their results with those obtained from different optimized PID and FOPID controllers previously published in the literature for the same AVR system. The numerical simulations and comparisons show that the proposed controllers provide the better dynamic response characteristics as well as more robust to model uncertainties than the other different optimized controllers. The results obtained with the proposed controllers show a better trade-off between the set point tracking performance, robustness and stability.

**Keywords:** AVR system; TLBO Algorithm; PID control; FOPID control

### 1. INTRODUCTION

In power systems, it is a crucial to design a voltage regulator in order to automatically maintain the reactive power and the voltage profile. For this reason, an automatic voltage regulator (AVR) system whose main objective is to control the primary voltage for the stable electrical power service is developed for synchronous generator with a high efficiency and a fast response [1]. Due to parameter uncertainties and external disturbances, the longtime electrical oscillations may arise and can cause harmful to stability of the system. As a consequence, the optimal control algorithms are anticipated to effectively cope with these issues.

Various controller tuning methods have been proposed in literature. Recently, different types of intelligent optimization algorithms and performance indices have been proposed to tune the controller parameters in the AVR system for obtaining improved transient and dynamic voltage stability. Proportional integral derivative (PID) controller, is a popular control technique since it is easy to design and implement, is being tuned by many artificial intelligence-based optimization algorithms for the AVR system. An approach for finding the optimal gains of PID controller by using particle swarm optimization (PSO) algorithm with a time domain performance criterion was proposed by Gaing [2] for the AVR system. Moreover, in that paper, the comparison of the results obtained from the PSO based approach and the genetic algorithm (GA) based approach has been given. Mohanty et al. [3] presented a method for tuning the PID controller parameters of the AVR system based on Local unimodal sampling (LUS) optimization algorithm. Also, in that paper, various conventional time domain integral performance indices and modified objective functions regarding peak overshoot and settling time with some appropriate weighting factors were employed during the optimization of the controller parameters. Zhu et al. [4] proposed a novel design method using the chaotic ant swarm (CAS) algorithm for the tuning of the PID controller parameters in the AVR system. In the tuning process of the parameters using CAS and GA, the fitness theory concerning the integral of time multiplied by the absolute value of error (ITAE)

and the overshoot was used. Moreover, the results of comparison between the CAS-PID and GA-PID controllers were given for showing the superior performance of the proposed method. Pande et al. [5] suggested many optimizing liaisons (MOL) algorithm to obtain optimum PID controller parameters for the AVR system. Also, the time domain integral performance indices such as the (ITAE), integral of absolute value of error (IAE), integral of time multiplied by squared error (ITSE) and integral of squared error (ISE) were considered. Moreover, a comparative analysis between the results from the proposed approach and the results from the other similar recently proposed optimization algorithms has been reported. Gozde and Taplamacioglu [6] presented comparative tuning performance of ABC algorithm to determine the optimal PID gains for the AVR system. A comparison for the robustness analysis of the proposed method was carried out with PSO and differential evolution (DE) algorithms in terms of transient response, root locus, bode and statistically receiver operating characteristic analyses. Kim [7] developed a hybrid method comprising GA and bacterial foraging (BF) optimization technique for optimizing the PID controller parameters in AVR control system. During the optimization, a time domain objective performance index including the dynamic response characteristics was used and also the results from the proposed approach were compared with that of GA, PSO and GA-PSO algorithms. Chatterjee and Mukherjee [8] presented robust PID controller design consisting of gains with first order low pass filter installed in the AVR system using TLBO algorithm. Also, the simulation results obtained from the proposed TLBO based PID controller were compared to those offered by the other algorithms reported in the literature. Blondin et al. [9] proposed a novel combination of a simplified ant colony optimization (ACO) algorithm and Nelder-Mead method (ACO-NM) for optimum design of PID controller in the AVR system. Also, a comparison between the proposed tuning method and other AVR tuning propositions was presented in terms of the overall transient response. Mouayad [10] proposed a new type of PID control consisting of proportional, integral, derivative, and second order derivative terms for the AVR system. The parameters of the proposed controller were tuned by the PSO algorithm using the ITAE. Also, in that paper, the performance of the proposed controller was compared to various PID controllers optimized with modern heuristic optimization algorithms using the different types of performance criterion. Bingul and Karahan [11] suggested a novel tuning design of PID controller with the cuckoo search (CS) algorithm using a new time domain performance criterion for the AVR system. In that paper, performance comparison between the proposed PID controller and different optimized PID controllers with different objective functions proposed in the literature were performed in detail.

During the past decades, many researches have investigated the concept of fractional calculus in many control applications to enhance the closed loop performance of existing control strategy. Hence, among fractional order (FO) controllers, much interest is devoted to fractional order PID (FOPID) controller. FOPID controller has more tuning parameters compared with PID controller. Accordingly, FOPID controller provides more flexibility for the designer. In 1999, Podlubny [12] firstly proposed FOPID controller. In that paper, the better closed loop performance was provided with the proposed FOPID controller as compared to PID controller. One of the most remarkable challenges is that tuning of the FOPID controller parameters is not easy. As a result, finding a suitable tuning approach for FOPID controller has been the subject of several examinations in order to spread usage of this controller in control engineers. Recently, the application of FOPID controller has been used in the design of the AVR system for maintaining the specific constant voltage of a synchronous generator, and also it has been reported to outperform the PID in many cases. For tuning the FOPID controller parameters in the AVR system, Zamani et al. [13] employed the PSO algorithm using a performance criterion consisting of the IAE, time domain measures and control effort. Tang et al. [14] designed FOPID controller using CAS optimization algorithm employing an objective function composed of the step response characteristics. Pan and Das [15] suggested performance measures consisting of various contradictory objective functions for tuning FOPID controller with the non-dominated sorting genetic algorithm (NSGA-II) for the AVR system. Despite of having so many FOPID controller tuning approaches, the settings of the most suitable FOPID controller parameters is still a challenging task. In [16], a novel optimum design technique of FOPID controller for the AVR system was developed using the CS algorithm. In that paper, during the optimization, time response specifications including the overshoot, rise time, settling time and steady state

error was used. Moreover, parameter variations in the AVR system have been considered to show robustness of the proposed FOPID controller and compared to recently published and other existing design methods. To enhance the stability and dynamic response of the AVR system, Suri babu and Chiranjeevi [17] presented FOPID controller by employing the GA and the ACO algorithms with the ITAE. In that paper, Comparisons were made with PID controller in terms of transient response and robustness characteristics. Zhang et al. [18] proposed an improved artificial bee colony algorithm (CNC-ABC) to optimize the FOPID controller parameters for the AVR system. Also, in that paper, the better performance was provided with the CNC-ABC FOPID controller as compared to other FOPID and PID controllers.

As observed from the studies given above, some investigations were allocated to determine a new tuning approach by using the aforementioned optimization algorithms with different performance criterion and constraints for the AVR system. Regarding above survey, in this paper, the optimal FOPID controller parameters are achieved by minimizing the dynamic response characteristics using TLBO algorithm. The simulation results are compared with recently published and other existing tuning methods. Moreover, the dynamic response and robustness of the proposed FOPID controller is investigated under the condition of parameter variations and compared with some PID and FOPID controllers tuned by recently published modern heuristic optimization algorithms mentioned above.

## 2. MODELLING OF THE AVR SYSTEM

The main task of the AVR system in a power generating station is to maintain the terminal voltage of the synchronous generator automatically at a specified level. The AVR system basically controls the output voltage by controlling the exciter voltage of the synchronous generator. Since the stability of the AVR system significantly affects the safety of the power system, it is necessary to design a controller for an AVR system to improve its stability and transient performance and to achieve a robust response despite external disturbance.

Generally, an AVR system is modelled by consisting of the amplifier, exciter, and sensor. The block diagram of the AVR system is shown in Figure 1 [2]. In this figure,  $V_s(s)$  is the output of the sensor measuring the terminal voltage  $V_t(s)$ .  $V_e(s)$  is the error voltage determined by the difference between the reference voltage,  $V_{ref}(s)$ , and  $V_s(s)$ .

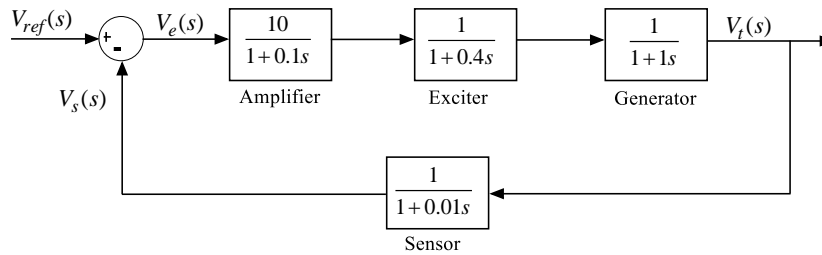


Figure 1. The block diagram of the AVR system

The amplifier in the AVR system is modelled as:

$$G_a(s) = \frac{K_a}{1 + \tau_a s} \quad (1)$$

where  $K_a$  and  $\tau_a$  are the gain and the time constant of the amplifier. The value of  $K_a$  is in the range of 100 – 400 while the amplifier time constant  $\tau_a$  ranges from 0.02s. to 0.1s. The transfer function of the exciter is:

$$G_e(s) = \frac{K_e}{1 + \tau_e s} \quad (2)$$

where  $K_e$  and  $\tau_e$  are the gain and the time constant of the exciter. The value of  $K_e$  is in the range of 1 – 400, while the amplifier time constant  $\tau_e$  ranges from 0.5s. to 1.0s. The transfer function of the generator is:

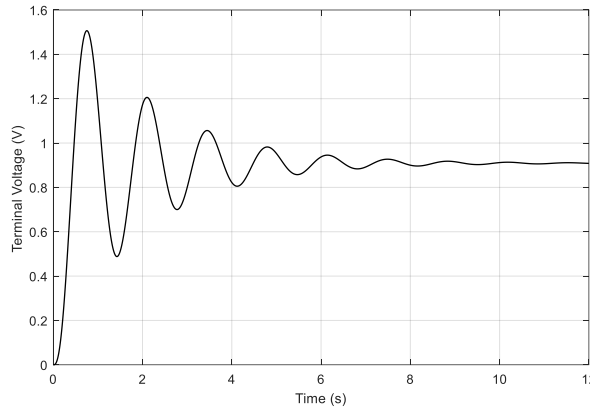
$$G_g(s) = \frac{K_g}{1 + \tau_g s} \quad (3)$$

where  $K_g$  and  $\tau_g$  are the gain and the time constant of the generator. The value of  $K_g$  is in the range of 0.7 – 1.0, while the amplifier time constant  $\tau_g$  ranges from 1.0s. to 2.0s. The sensor is modelled as:

$$G_s(s) = \frac{K_s}{1 + \tau_s s} \quad (4)$$

where  $K_s$  and  $\tau_s$  are the gain and the time constant of the sensor. The value of  $K_s$  is 1.0, while the amplifier time constant  $\tau_s$  ranges from 0.001s. to 0.06s.

By choosing, the parameters for the AVR system are as follows:  $K_a = 10.0$ ,  $\tau_a = 0.1s.$ ,  $K_e = 1.0$ ,  $\tau_e = 0.4s.$ ,  $K_g = 1.0$ ,  $\tau_g = 1s.$ ,  $K_s = 1.0$ ,  $\tau_s = 0.01s.$ , the step response of the AVR system is obtained as shown in Figure 2.



**Figure 2.** Step response of the AVR system

From the step response graphic, it is observed that it has high oscillations with  $M_p = \%66$  (maximum overshoot),  $t_r = 0.31 s.$  (rising time),  $t_s = 5.57 s.$  (settling time), and  $E_{ss} = 0.091$  (steady-state error).

### 3. CONTROLLER DESIGN

#### 3.1. PID Controller

The output of the conventional PID controller  $u(t)$  can be expressed in terms of the input  $e(t)$  as:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) + K_d \frac{de(t)}{dt} \quad (5)$$

where  $K_p$ ,  $K_i$ ,  $K_d$  are the proportional, integral and derivative gains, respectively. In this case, the transfer function of the PID controller is formed as:

$$C_{PID}(s) = K_p + K_i \frac{1}{s} + K_d s \quad (6)$$

#### 3.2. FOPID Controller

FOPID controller is a generalization of the traditional PID controller. Compared to the PID controller, FOPID controller is characterized by two additional control parameters in which the orders of the

integral part  $\lambda$  and derivative part  $\mu$  are non-integer. These extra parameters provide a greater flexibility in the design of controller and also can lead to achieve an improving dynamic performance. Accordingly, in FOPID controller, the fractional calculus is brought into PID controller to increase its performance. The output of the FOPID ( $PI^\lambda D^\mu$ ) controller is defined by:

$$u(t) = K_p e(t) + K_i D_t^{-\lambda} e(t) + K_d D_t^\mu e(t) \quad (7)$$

where the operator  ${}_a D_t^\alpha$  represents fractional differentiation or integration [13, 15],  $a$  and  $t$  are respectively the lower and upper bound, and  $\alpha$  is the order of derivative or integral.

The transfer function of the FOPID is also given as:

$$C_{FOPID}(s) = K_p + K_i \frac{1}{s^\lambda} + K_d s^\mu \quad (8)$$

For designing the FOPID, three parameters  $K_p$ ,  $K_i$ ,  $K_d$  and two non-integer orders  $\lambda$ ,  $\mu$  for the system are determined.

#### 4. TLBO ALGORITHM

Teaching-learning based optimization (TLBO) is one of the recently proposed population based algorithms which is inspired from the teaching and learning phenomenon of a class [19]. For the population based algorithm TLBO, class is considered as a population, and each student in the class (learner) is considered as individual in this population. In order to obtain the optimum solution, it is aimed to increase the knowledge level of the students in the class. As a basis for this aim, operations are carried out in two phases: teacher phase and learner phase.

##### **Teacher phase:**

The student who gives the best solution in the class is elected as a teacher. A teacher is being considered as the best learner in the entire population, so the teacher is the most experienced and knowledgeable person of a subject. Accordingly, other students increase his or her knowledge by using the knowledge of this teacher. If a student gives a better solution than the teacher, he/she will replace the teacher. For the  $k$  th iteration, the knowledge of the  $i$  th learner is updated by:

$$X_i^{new} = X_i + r_i (X_T - T_F^i \cdot X_{mean}) \quad (9)$$

In here,  $X_T$  is the knowledge of the teacher,  $r_i$  is the random number in the range of [0, 1],  $X_{mean}$  is the mean results of the learners and  $T_F^i$  is the teaching factor obtained by:

$$T_F^i = \text{round}[1 + \text{rand}(0,1)] \quad (10)$$

At the end of the teacher phase, all the knowledge of the learners are being updated, and these values are ready for the learner phase as input values.

##### **Learner phase:**

In a real classroom, the students can improve their knowledge by interacting with the other students in the class. This is the base of the learner phase of the TLBO algorithm. For this phase the knowledge of the  $i$  th student is compared by the any  $j$  th student which is randomly selected form the class.

$$\begin{aligned} \text{If } f(X_i) < f(X_j) \text{ than, } X_i^{new} &= X_i + r_i (X_i - X_j) \\ \text{If } f(X_i) > f(X_j) \text{ than, } X_i^{new} &= X_i + r_i (X_j - X_i) \end{aligned} \quad (11)$$

At the end of the learner phase, if the knowledge of the  $j$  th student is better,  $i$  th student can improve his/her knowledge:

$$\text{If } f(X_i^{new}) < f(X_i) \text{ than, } X_i = X_i^{new} \quad (12)$$

The TLBO algorithm consisting of the teacher phase and learner phase continues until the termination condition or maximum number of iterations are reached. The algorithm can be summarized as below:

**Algorithm 1. TLBO Algorithm**

Decide number of students (population size), decision variables and termination criterion or max. number of iteration.

Set  $k=1$ .

Initialize all variables and initiate all population array ( $X$ ) (students of the classroom) randomly.

Compute the objective function  $f(X)$  for all students of the classroom.

**WHILE** (the termination condition or maximum number of iterations are not reached)

**FOR**  $i=1 \rightarrow n$

**{teacher phase}**

Evaluate the mean of each design variable,  $X_{mean}$ . Identify the best solution (teacher).

Compute teaching factor:  $T_F^i = \text{round}[1 + \text{rand}(0,1)]$ .

Update the knowledge of the learners according to the knowledge of the teacher ( $X_T$ ):

$$X_i^{new} = X_i + r_i (X_T - T_F^i \cdot X_{mean})$$

Compute the new value of the objective function  $f(X_i^{new})$

IF  $f(X_i^{new}) < f(X_i)$  than  $X_i = X_i^{new}$

**{end of the teacher phase}**

**{learner phase}**

Select a learner  $X_j$ , randomly ( $j \neq i$ ).

$$\text{IF } f(X_i) < f(X_j) \text{ than } X_i^{new} = X_i + r_i (X_i - X_j) \quad \text{else } X_i^{new} = X_i + r_i (X_j - X_i)$$

IF  $f(X_i^{new}) < f(X_i)$  than  $X_i = X_i^{new}$

**{end of the learner phase}**

**END FOR**

Set  $k=k+1$

**END WHILE**

## 5. CHOICE OF OBJECTIVE FUNCTION

During the controller design by using an optimization algorithm, the most crucial step is to select the most appropriate objective function. Time domain objective functions can be divided into two categories: Integral based objective functions and dynamic performance indices based objective functions.

Integral based objective functions commonly used in literature are: IAE (Integral of Absolute Error), ITAE (Integral of Time Absolute Error), ISE (Integral of Squared Error) and ITSE (Integral of Time Squared Error). For these objective functions, the difference between the system output and reference signal is indicated by the error. Each one of these integral based objective functions has advantages and disadvantages. For example, since IAE and ISE criteria are independent of time, the obtained results have relatively small overshoot but a long settling time. On the other hand, ITAE and ITSE can overcome this disadvantage; but, they can not provide a desirable stability margin.

The second category of the time domain objective functions is based on the performance indices of the system dynamic output. These functions usually involve the maximum overshoot ( $M_p$ ), rising time ( $t_r$ ), settling time ( $t_s$ ), and steady-state error ( $E_{ss}$ ).

In this study, one of time domain objective functions used in literature is considered [2]:

$$W(\beta) = (1 - e^{-\beta})(M_p + E_{ss}) + e^{-\beta}(t_s - t_r) \quad (13)$$

where  $\beta$  is weighting factor to allow the designer to determine the significance of performance criteria to others.

## 6. RESULTS AND DISCUSSIONS

In this section, simulations are carried out on the AVR system model for discussing thoroughly the transient responses of the optimized PID and FOPID controllers with TLBO. Moreover, the robustness of the PID and FOPID controllers' parameters obtained at the nominal condition is observed for parametric uncertainties. The study is carried out in three cases of weighting factor  $\beta$ : the optimization process is employed for  $\beta = 0.8$ ,  $\beta = 1$  and  $\beta = 1.5$ , respectively. Hence, for obtaining good dynamic stability of the AVR system, the optimized PID and FOPID controllers according to the values of weighting factor  $\beta$  are used. To achieve the best outputs from the PID and FOPID, the controller parameters are simultaneously tuned with TLBO. Based on the detailed literature review, the lower bounds of the controller parameters are zero and their upper bounds are set to  $K_{pmax} = K_{imax} = K_{dmax} = 1.5$  and  $\mu = \lambda = 2$ . Also, the transfer function of the FOPID controllers given in Table 1 is approximated using by the Oustaloup continuous approximation method which is one of the most known approximation methods [20]. The frequency band is in the range of [0.001 Hz, 1000 Hz] and the order of the approximation is 6 (number of poles and zeros). The model of the system under study illustrated in Figure 1 has been designed in MATLAB/SIMULINK environment. During the optimization, the parameters of TLBO should be chosen carefully. In this work, a population size of NP is set to 10, number of iterations is taken as 100 and the simulation time  $t$  is set to 15 s. The simulations were conducted on an Intel, Core™ i7-6700 HQ CPU of 2.6 GHz and 32.00 GB RAM computer in MATLAB 9.0.0.341360 (R2016a) environment. The optimization process was independently run with random initial conditions for ten times in the three cases of weighting factor  $\beta$ . The best final solutions obtained in the 10 runs are shown in Figure 3. Accordingly, the best results obtained from successive ten runs for the three cases are chosen as optimum parameters of the PID and FOPID controllers as reported in Table 1.

**Table 1.** Performance indice of the AVR system with the best controller parameters for different  $\beta$  values

$\beta$	Controller	$M_p$ %	$t_r$ 0.1 → 0.9	$t_s$ $\pm 5\%$	$E_{ss}$	$K_p$	$K_i$	$K_d$	$\mu$	$\lambda$	Obj. value
0.8	TLBO-PID	0.0335	0.2620	0.3620	0.0004	0.6735	0.4327	0.2619	-	-	0.5960
	TLBO-FOPID	0	0.1280	0.1780	0.0081	1.4992	0.6435	0.4333	1.1855	1.0010	0.5765
1	TLBO-PID	0	0.2630	0.3630	0.0005	0.6725	0.4314	0.2613	-	-	0.6692
	TLBO-FOPID	0	0.1620	0.2250	0.0017	1.0633	0.6642	0.3943	1.1126	0.9805	0.6553
1.5	TLBO-PID	0	0.2670	0.3690	0.0021	0.6706	0.4149	0.2556	-	-	0.8008
	TLBO-FOPID	0.0047	0.1400	0.1950	0.0057	1.3487	0.7124	0.4142	1.1603	0.9672	0.7936

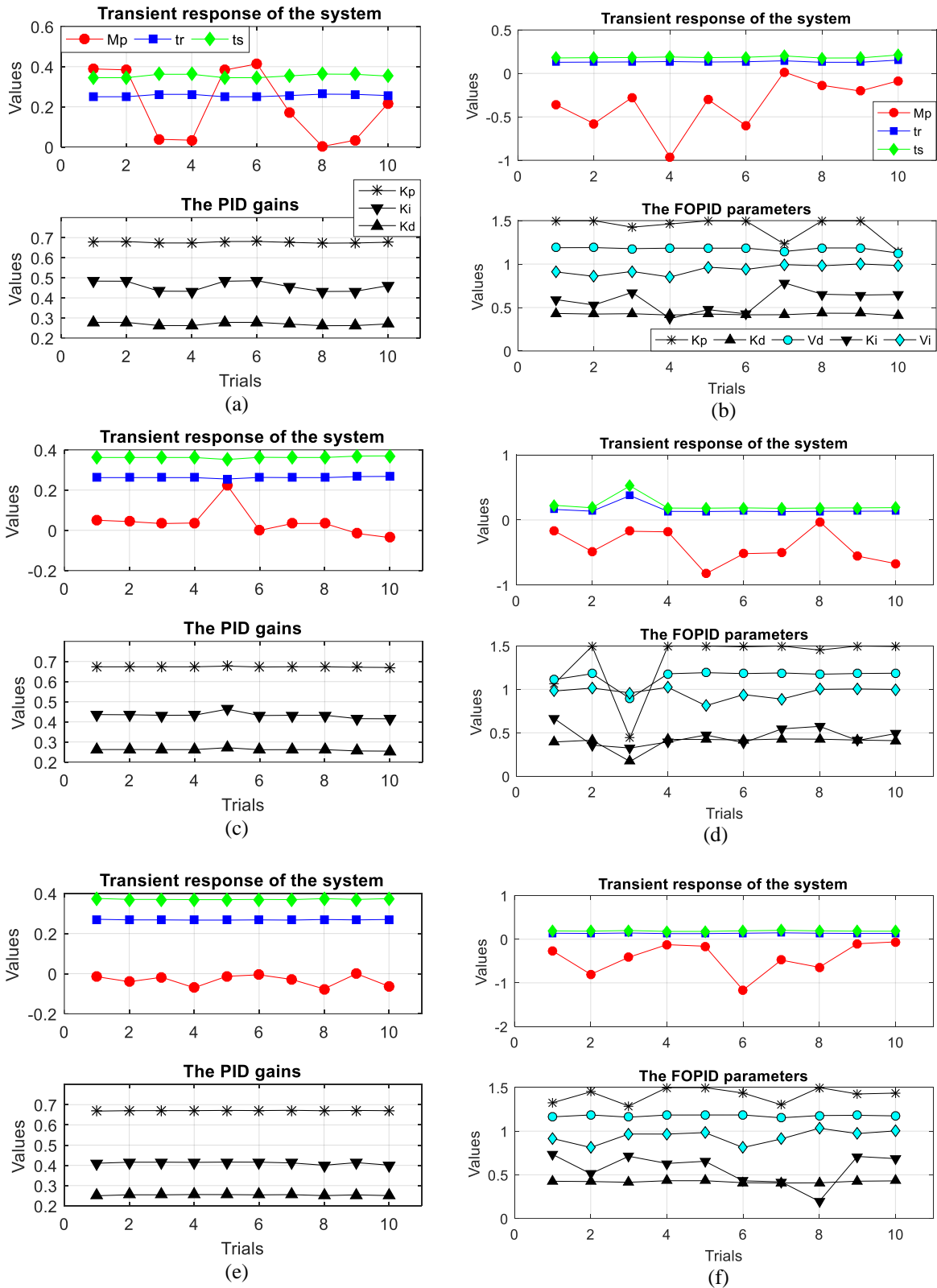


Figure 3. Results of ten trials for the PID and FOPID controllers using the values of  $\beta = 0.8$  (a - b),  $\beta = 1$  (c - d) and  $\beta = 1.5$  (e - f).



### 6.1. Transient Response Analysis

For comparison purpose, the transient responses of the proposed PID and FOPID controllers tuned by TLBO are analyzed by comparing with different optimized PID and FOPID controllers depending upon the different values of weighting factor  $\beta$ . The PID and FOPID controllers were tuned by CAS [14], PSO [2, 10, 16], GA [16], CS [16], TLBO [8] algorithms for the same AVR system. The terminal voltage step responses of the proposed controllers and the different optimized controllers with one of the aforementioned tuning algorithms according to the values of  $\beta$  for the AVR system are compared as shown in Figure 4. Also, the numerical results of the step response comparison including controller parameters, the best dynamic response characteristics and the objective function values are listed in Table 2.

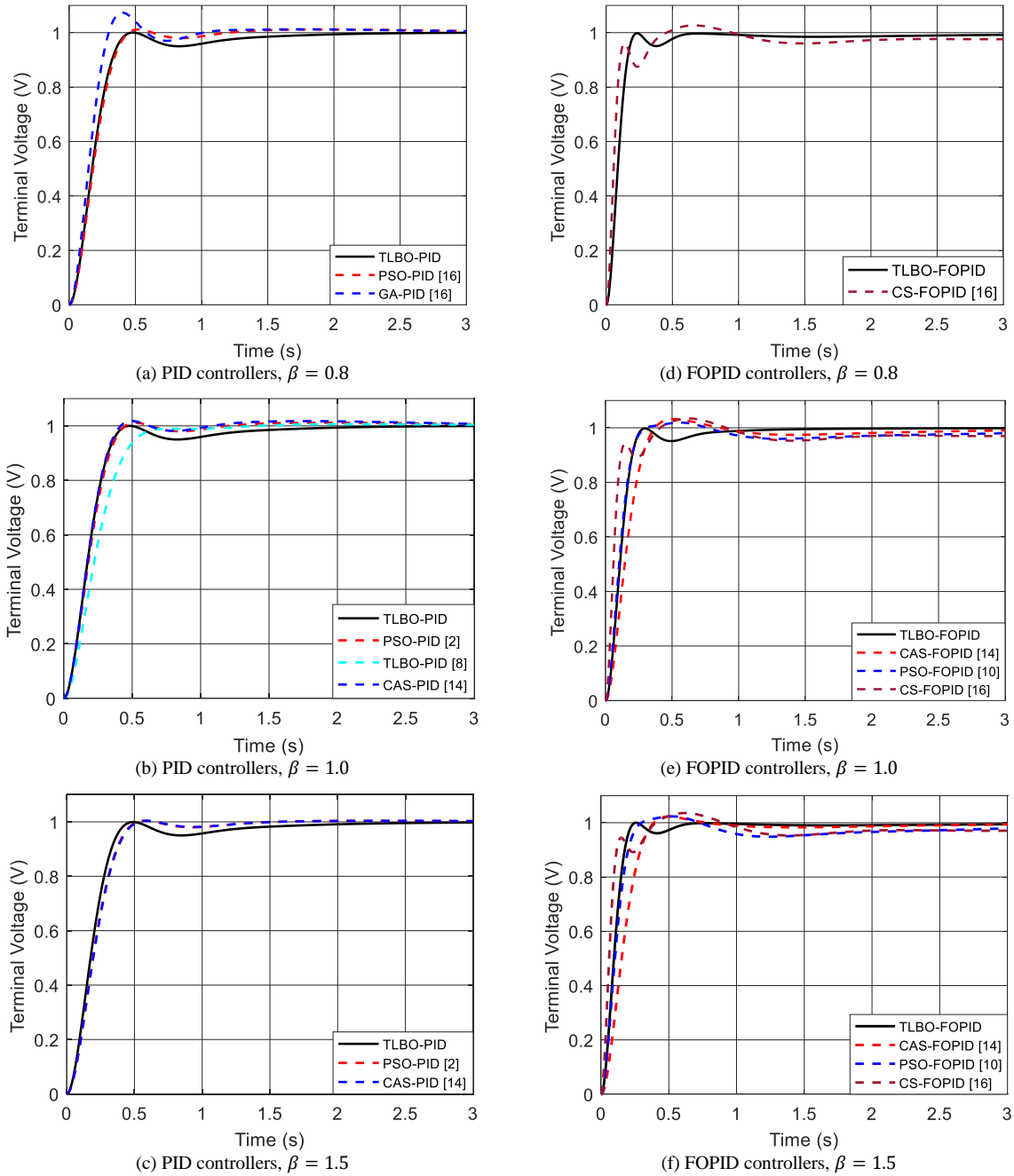


Figure 4. Terminal voltage step responses of the AVR system with different controllers.

**Table 2.** Different optimized controller parameters and the time domain performance indices under different  $\beta$  values.

Algorithm-Controller		Controller Parameters					$M_p$ %	$t_r$ 0.1 → 0.9	$t_s$ ±5%	$E_{ss}$	Obj. value
		$K_p$	$K_i$	$K_d$	$\mu$	$\lambda$					
0.8	PSO-PID [16]	0.6570	0.5390	0.2480	-	-	1.1613	0.2710	0.3750	0.0061	0.6072
	GA-PID [16]	0.8663	0.7531	0.3365	-	-	7.3438	0.1990	0.4810	0.0050	0.7206
	<b>TLBO-PID</b>	0.6735	0.4327	0.2619	-	-	0.0335	0.2620	0.3620	0.0004	0.5960
0.8	CS-FOPID [16]	2.4676	0.3020	0.4230	1.3800	0.9700	2.6692	0.0850	0.3370	0.0241	0.6919
	<b>TLBO-FOPID</b>	1.4992	0.6435	0.4333	1.1855	1.0010	0	0.1280	0.1780	0.0081	0.5765
1.0	PSO-PID [2]	0.6570	0.5389	0.2458	-	-	1.1601	0.2710	0.3750	0.0061	0.6816
	TLBO-PID [8]	0.5302	0.4001	0.1787	-	-	0.6376	0.3720	0.5220	0.0041	0.6939
	CAS-PID [14]	0.6746	0.6009	0.2618	-	-	1.7702	0.2580	0.3540	0.0073	0.6833
	<b>TLBO-PID</b>	0.6725	0.4314	0.2613	-	-	0	0.2630	0.3630	0.0005	0.6692
1.0	CAS-FOPID [14]	1.0537	0.4418	0.2510	1.1122	1.0624	3.1987	0.2300	0.3140	0.0100	0.6896
	PSO-FOPID [10]	1.6264	0.2956	0.3226	1.1980	1.3183	1.9084	0.1630	0.2280	0.0201	0.6808
	CS-FOPID [16]	2.5150	0.1629	0.3888	1.3800	0.9700	3.4181	0.0950	0.3280	0.0303	0.7586
	<b>TLBO-FOPID</b>	1.0633	0.6642	0.3943	1.1126	0.9805	0	0.1620	0.2250	0.0017	0.6553
1.5	PSO-PID [2]	0.6254	0.4577	0.2187	-	-	0.4412	0.3000	0.4150	0.0030	0.8083
	CAS-PID [14]	0.6202	0.4531	0.2152	-	-	0.4050	0.3050	0.4210	0.0028	0.8081
	<b>TLBO-PID</b>	0.6706	0.4149	0.2556	-	-	0	0.2670	0.3690	0.0021	0.8008
1.5	CAS-FOPID [14]	0.9315	0.4776	0.2536	1.0838	1.0275	2.2428	0.2380	0.3270	0.0064	0.8191
	PSO-FOPID [10]	1.6986	0.1797	0.3122	1.2081	1.8373	2.3180	0.1640	1.3690	0.0210	1.0801
	CS-FOPID [16]	2.5490	0.1759	0.3904	1.3800	0.9700	3.5656	0.0940	0.3250	0.0294	0.8790
	<b>TLBO-FOPID</b>	1.3487	0.7124	0.4142	1.1603	0.9672	0.0047	0.1400	0.1950	0.0057	0.7936

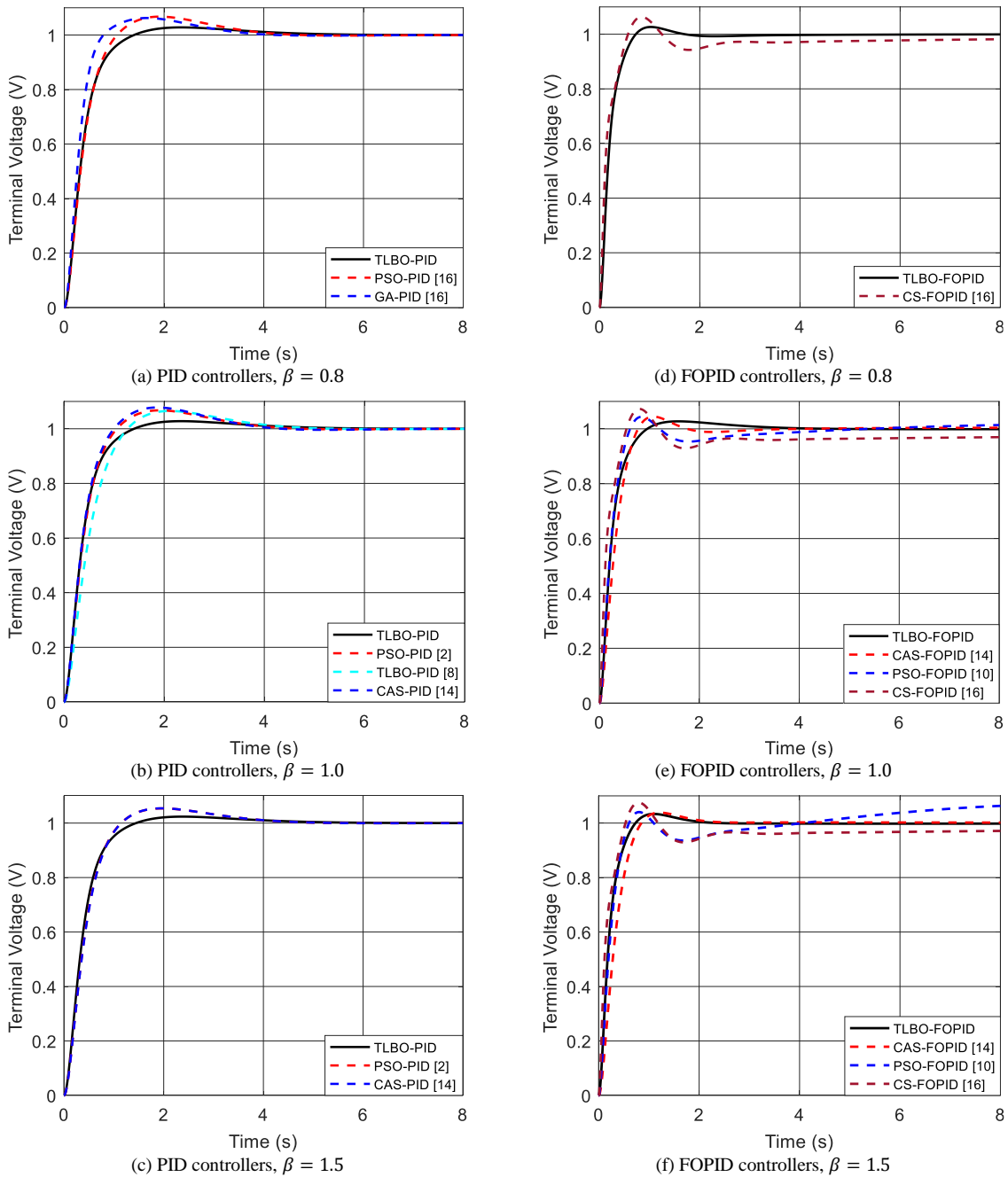
During comparison as reported in Table 2 and as shown in Figure 4, the proposed TLBO-PID and TLBO-FOPID controllers have superior performance over the other optimized controllers for each case of  $\beta$ . In comparison to PSO-PID [2, 16], GA-PID [16], TLBO-PID [8], CAS-PID [14], CS-FOPID [16], CAS-FOPID [14], PSO-FOPID [10], it may be found that the proposed controllers exhibit better performance characteristics which are  $M_p$ ,  $t_s$ ,  $E_{ss}$ , objective function value, and only little worse  $t_r$ . As a result, the proposed TLBO-FOPID controller has the ability to achieve the most accurate and most stable unit step response in comparison with all the PID and the other FOPID controllers in all cases (i.e.  $\beta = 0.8$ ,  $\beta = 1.0$  and  $\beta = 1.5$ ).

### 6.2. Robustness Analysis

The behavior of the system is observed by the robustness analysis under the parametric uncertainties. Here, to observe the robustness of the proposed PID and FOPID controllers tuned by TLBO, three kinds of parameter uncertainties in the AVR system are considered. Thus, it is assumed that the generator, exciter and amplifier model parameters are respectively changed to  $K_g = 0.7$  and  $\tau_g = 1.6$ ,  $K_e = 1.4$  and  $\tau_e = 0.8$ ,  $K_a = 7$  and  $\tau_a = 0.3$  from actual values as previously specified.

Figure 5-7 show step responses of the proposed controllers and different optimized controllers under variations in the generator, exciter and amplifier model parameters of the AVR system, respectively. Also, the

corresponding dynamic response measurements are given in Table 3-5 for the proposed controllers and the other compared controllers under system parameter variations in cases of  $\beta = 0.8$ ,  $\beta = 1.0$  and  $\beta = 1.5$ .



**Figure 5.** Terminal voltage step responses with parameter uncertainties in generator model.

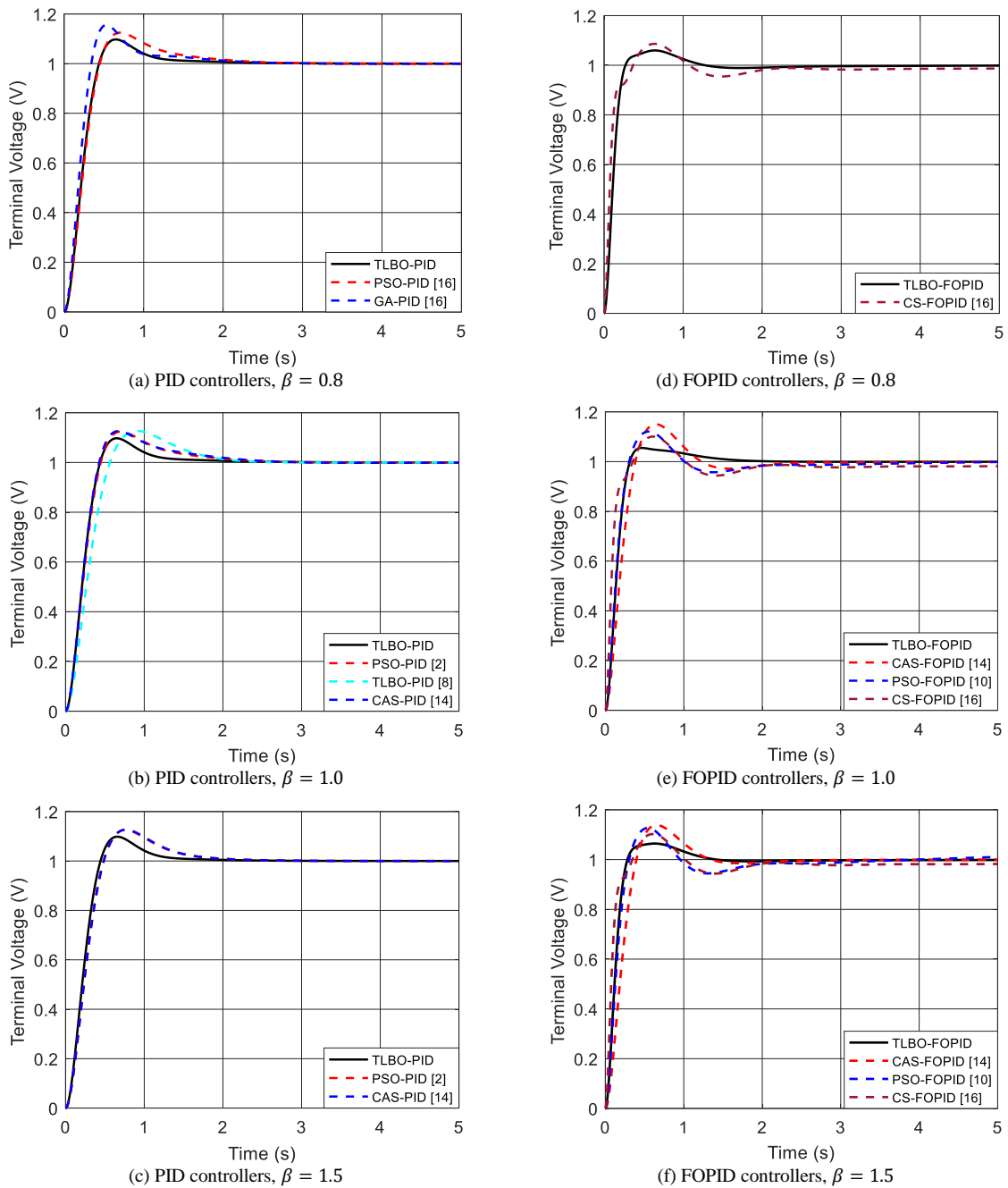
It is evident from Figure 5 that the terminal voltage stability has been successfully improved in case of the proposed PID and FOPID controllers under parameter uncertainties in generator model of the AVR system. From dynamic responses given in Table 3, it is found that the proposed controllers have the more robust performance and lower maximum overshoot, lower settling time, lower steady state error as compared to the other optimized controllers in all cases of weighting factor  $\beta$ .

**Table 3.** Comparative performance of different optimized controllers under parameter uncertainties in generator model.

System dynamic parameters	$\beta$	Algorithm-Controller	$M_p(\%)$	$t_r$ 0.1 → 0.9	$t_s$ ±5%	$E_{ss}$	
Change to $K_g = 0.7, \tau_g = 1.6$ from actual value $K_g = 1.0, \tau_g = 1.0$	0.8	PSO-PID [16]	6.7522	0.6400	2.6330	0.0001	
		GA-PID [16]	6.2480	0.4540	2.2280	0.00001	
		<b>TLBO-PID</b>	2.8000	0.6790	0.9780	0.0001	
	1.0	PSO-PID [2]	6.7490	0.6400	2.6330	0.0001	
		TLBO-PID [8]	6.5162	0.8100	2.8080	0.0003	
		CAS-PID [14]	7.8030	0.6080	2.7270	0.0001	
	<b>TLBO-PID</b>	2.7864	0.6800	0.9800	0.00001		
	1.5	PSO-PID [2]	5.3874	0.6920	2.3090	0.0002	
		CAS-PID [14]	5.4293	0.6980	2.3250	0.0002	
		<b>TLBO-PID</b>	2.3709	0.6870	0.9920	0.0001	
	Change to $K_e = 1.4, \tau_e = 0.8$ from actual value $K_e = 1.0, \tau_e = 0.4$	0.8	CS-FOPID [16]	6.3399	0.3870	2.0440	0.0183
			<b>TLBO-FOPID</b>	2.6533	0.4090	0.5620	0.0006
		1.0	CAS-FOPID [14]	4.4656	0.5040	0.6610	0.0040
			PSO-FOPID [10]	4.3880	0.4010	0.5230	0.0140
			CS-FOPID [16]	7.2521	0.3760	2.1480	0.0303
<b>TLBO-FOPID</b>			2.7417	0.4890	0.6950	0.0013	
1.5		CAS-FOPID [14]	3.8348	0.5360	0.7110	0.0018	
		PSO-FOPID [10]	6.3244	0.3950	8.0000	0.0632	
		CS-FOPID [16]	7.5283	0.3710	2.1160	0.0288	
		<b>TLBO-FOPID</b>	3.3482	0.4190	0.5750	0.0019	

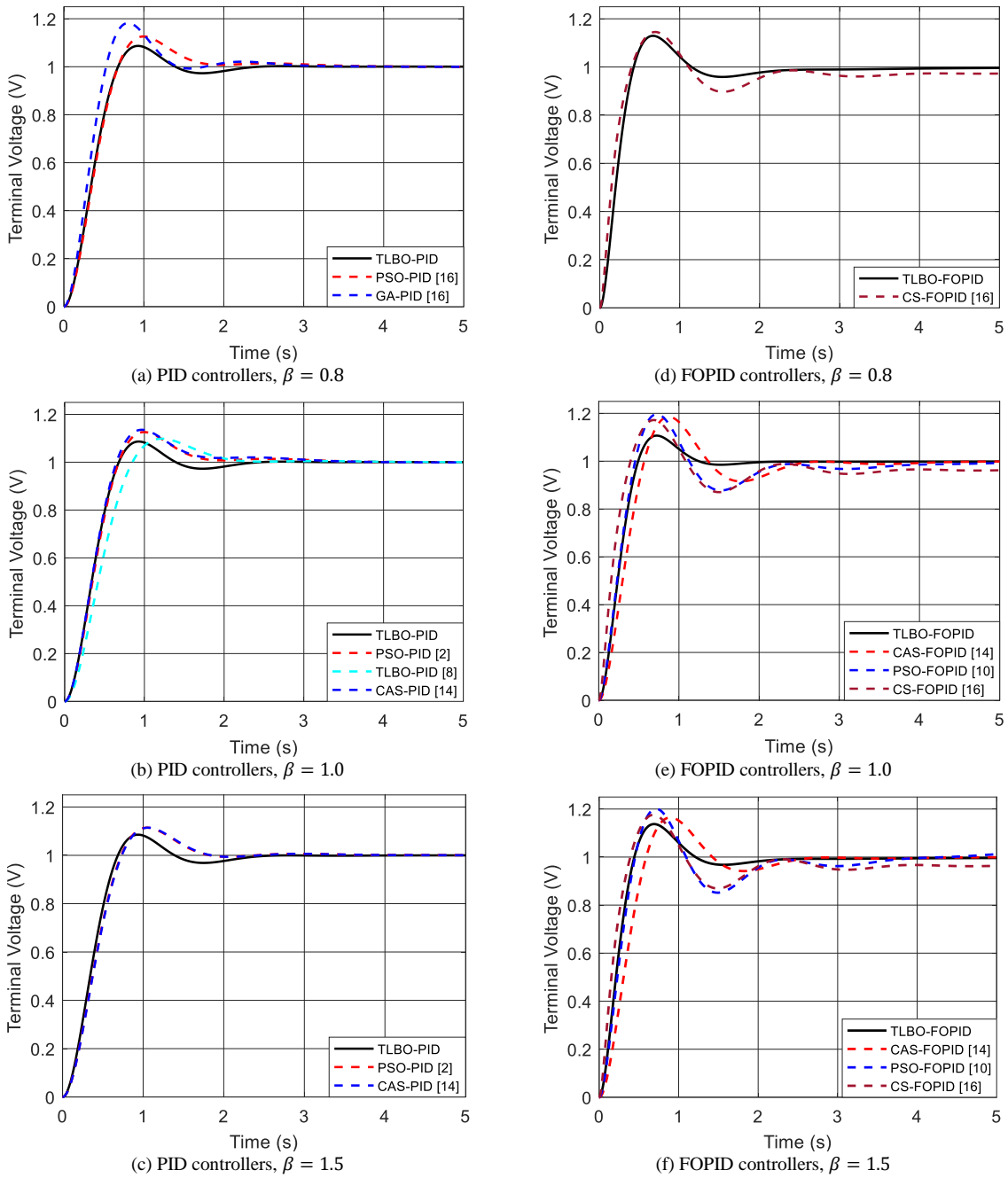
**Table 4.** Comparative performance of different optimized controllers under parameter uncertainties in exciter model

System dynamic parameters	$\beta$	Algorithm-Controller	$M_p(\%)$	$t_r$ 0.1 → 0.9	$t_s$ ±5%	$E_{ss}$	
Change to $K_e = 1.4, \tau_e = 0.8$ from actual value $K_e = 1.0, \tau_e = 0.4$	0.8	PSO-PID [16]	12.4692	0.3030	1.2550	0.0003	
		GA-PID [16]	15.5454	0.2320	0.9060	0.0003	
		<b>TLBO-PID</b>	9.7484	0.2940	0.9410	0.0001	
	1.0	PSO-PID [2]	12.4673	0.3030	1.2540	0.0003	
		TLBO-PID [8]	12.7027	0.3840	1.5220	0.00001	
		CAS-PID [14]	12.6386	0.2890	1.2950	0.0005	
		<b>TLBO-PID</b>	9.7342	0.2940	0.9420	0.0001	
	1.5	PSO-PID [2]	12.5669	0.3270	1.2660	0.00001	
		CAS-PID [14]	12.6783	0.3310	1.2820	0.00001	
		<b>TLBO-PID</b>	9.8346	0.2980	0.9530	0.00001	
	Change to $K_g = 0.7, \tau_g = 1.6$ from actual value $K_g = 1.0, \tau_g = 1.0$	0.8	CS-FOPID [16]	8.6341	0.1350	0.8750	0.0130
			<b>TLBO-FOPID</b>	5.9682	0.1660	0.7940	0.0016
		1.0	CAS-FOPID [14]	15.0394	0.2620	1.0280	0.0001
			PSO-FOPID [10]	12.0840	0.2010	0.8470	0.0017
			CS-FOPID [16]	10.0780	0.1550	1.5650	0.0184
<b>TLBO-FOPID</b>			5.4964	0.2010	0.5850	0.0008	
1.5		CAS-FOPID [14]	13.8527	0.2700	1.0600	0.0001	
		PSO-FOPID [10]	12.6788	0.2020	1.4590	0.0106	
		CS-FOPID [16]	10.2493	0.1520	1.5540	0.0177	
		<b>TLBO-FOPID</b>	6.4400	0.1790	0.8490	0.0006	



**Figure 6.** Terminal voltage step responses with parameter uncertainties in exciter model.

From Figure 6 and Table 4, it can be observed that the proposed TLBO based FOPID controller outperforms the PSO, CAS and CS tuned controller in terms of minimum  $\%M_p$ , lower  $t_r$  and achieve lower  $E_{SS}$  in faster  $t_s$  in most of cases. On the other hand, in case of  $\beta = 0.8$  and  $\beta = 1.0$ , although the rise time of the proposed TLBO based PID controller is closely higher as compared to that of other different optimized PID controllers, all other time domain specifications such as  $M_p$ ,  $t_s$  and  $E_{SS}$  by the proposed PID controller are superior in case of all.

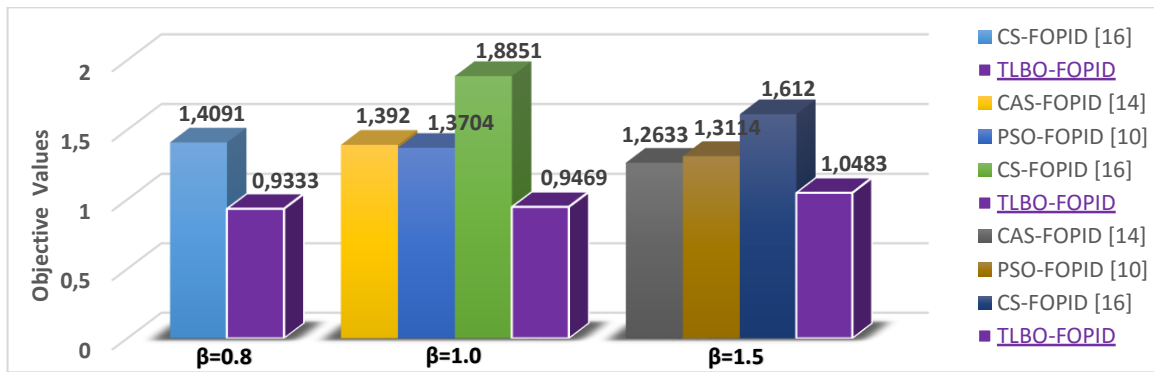
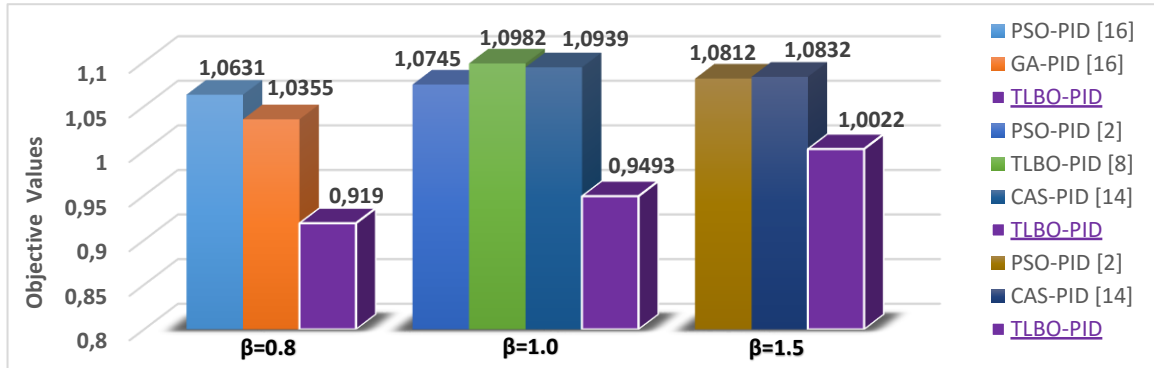


**Figure 7.** Terminal voltage step responses with parameter uncertainties in amplifier model.

As can be seen from comparing the dynamic responses in Table 5 and Figure 7, this examination reveals that performances of the proposed PID and FOPID controllers are far better as compared with aforementioned controllers in terms of less overshoot, settling time and steady state error. Moreover, observing the data in Table 5, for all cases of  $\beta$ , it is clearly seen that the responses of all PID controllers are nearly same with less rise time and steady state error whereas the performance of the proposed TLBO based PID controller remarkably improves the settling time with less overshoot as compared to the other PID controllers.

**Table 5.** Comparative performance of different optimized controllers under parameter uncertainties in amplifier model.

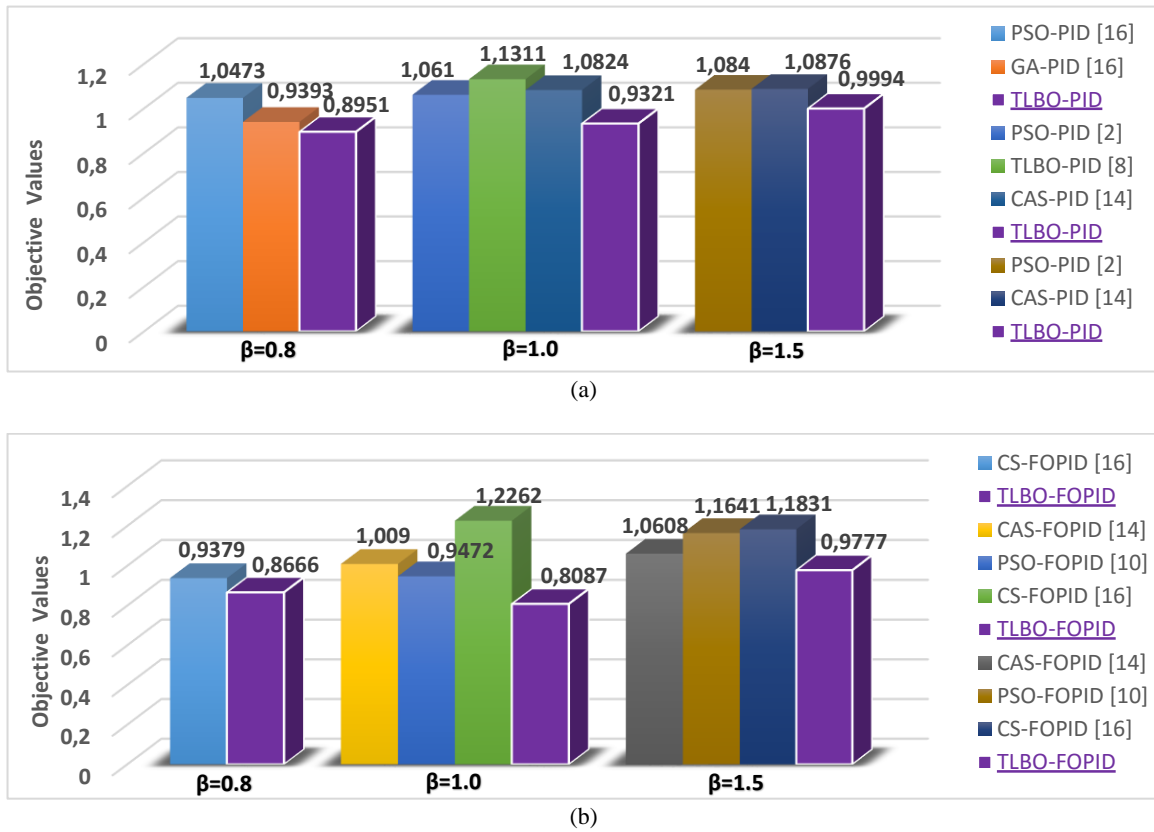
System dynamic parameters	$\beta$	Algorithm-Controller	$M_p$ (%)	$t_r$ 0.1 → 0.9	$t_s$ $\pm 5\%$	$E_{ss}$	
Change to $K_a = 7, \tau_a = 0.3$ from actual value $K_a = 10, \tau_a = 0.1$	0.8	PSO-PID [16]	12.6076	0.4570	1.4430	0.00001	
		GA-PID [16]	18.2364	0.3570	1.2120	0.0003	
		<b>TLBO-PID</b>	8.6794	0.4520	1.1650	0.0004	
	1.0	PSO-PID [2]	12.6044	0.4570	1.4430	0.00001	
		TLBO-PID [8]	9.8465	0.5690	1.6660	0.0004	
		CAS-PID [14]	13.5595	0.4390	1.4600	0.0007	
		<b>TLBO-PID</b>	8.6448	0.4530	1.1660	0.0003	
	1.5	PSO-PID [2]	11.4441	0.4900	1.4540	0.0005	
		CAS-PID [14]	11.4460	0.4950	1.4680	0.0004	
		<b>TLBO-PID</b>	8.5469	0.4570	1.1690	0.0001	
	Change to $K_a = 7, \tau_a = 0.3$ from actual value $K_a = 10, \tau_a = 0.1$	0.8	CS-FOPID [16]	14.5262	0.2730	1.9720	0.0272
			<b>TLBO-FOPID</b>	12.9370	0.2910	0.9790	0.0041
		1.0	CAS-FOPID [14]	18.5683	0.3930	2.1370	0.0015
			PSO-FOPID [10]	19.6566	0.3190	1.9750	0.0076
			CS-FOPID [16]	17.2007	0.2780	3.3230	0.0380
<b>TLBO-FOPID</b>			10.6845	0.3330	1.0020	0.0016	
1.5		CAS-FOPID [14]	16.4754	0.4070	2.0110	0.0007	
		PSO-FOPID [10]	20.0937	0.3180	1.9720	0.0120	
		CS-FOPID [16]	17.6006	0.2750	3.2780	0.0365	
		<b>TLBO-FOPID</b>	13.7326	0.3040	1.0290	0.0038	



**Figure 8.** The objective function comparison of different optimized PID controllers (a) and FOPID controllers (b) under parameter uncertainties in amplifier model of the AVR system.

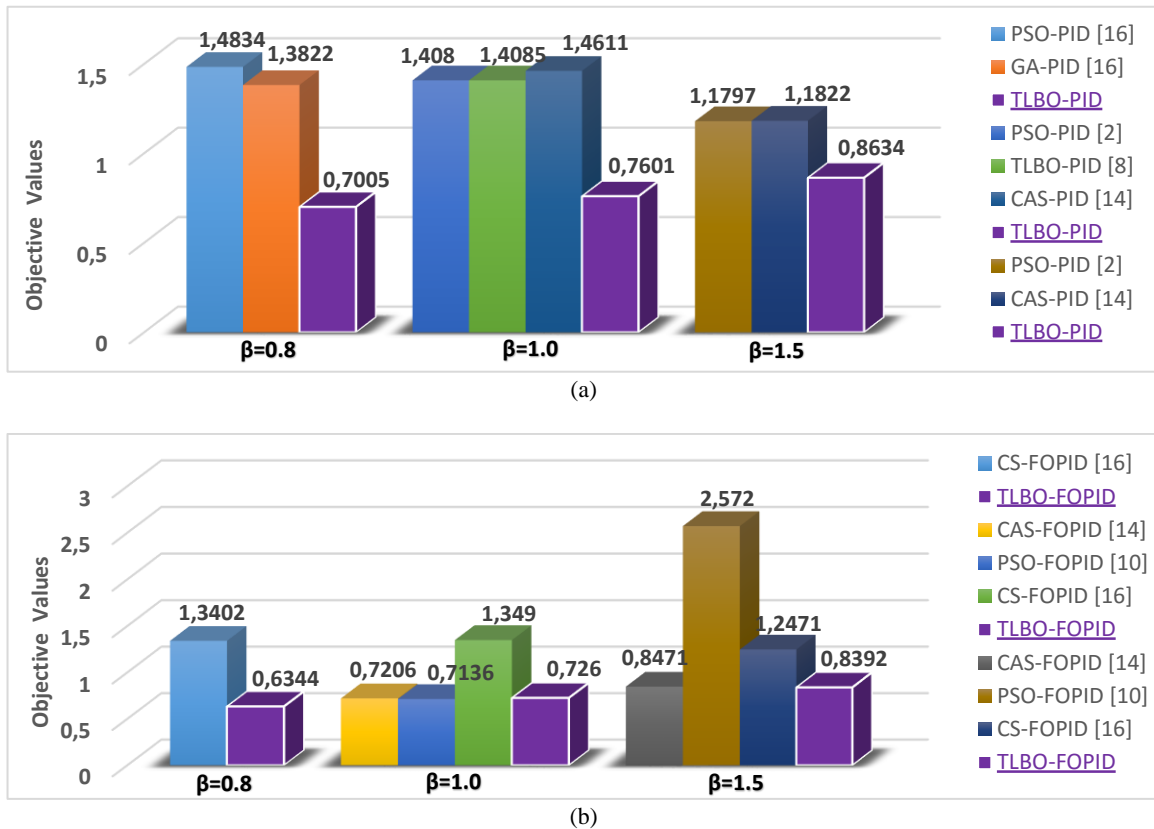
Figure 8-10 show the values of performance index for the proposed controllers and the different optimized controllers under parameter uncertainties in amplifier, exciter and generator models of the AVR system in all cases of  $\beta$ . In case of the uncertainty of amplifier model, it is clear that the proposed TLBO based controllers provide very low objective function values as shown in Figure 8. From Figure 9 for the uncertainty case in exciter model, the objective values in all cases of  $\beta$  show that the TLBO algorithm leads to the PID and FOPID controllers with a smaller objective function value. As for Figure 10, it is evident that the values of objective function for all  $\beta$  by the proposed PID and FOPID controllers are smaller than other optimized controllers.

The results in Figure 5-7 and Table 3-5 demonstrate that the PID and FOPID controllers tuned by the TLBO algorithm can attain a steady state of step response curve. Finally, these results overall show that the TLBO algorithm can more effectively optimize the PID and FOPID controllers using the objective function with different weighting factors. Finally, analyzing the results obtained for the robustness, it is concluded that TLBO-PID and TLBO-FOPID are more robust to uncertainties as compared with different optimized PID and FOPID controllers.



**Figure 9.** The objective function comparison of different optimized PID controllers (a) and FOPID controllers (b) under parameter uncertainties in exciter model of the AVR system.





**Figure 10.** The objective function comparison of different optimized PID controllers (a) and FOPID controllers (b) under parameter uncertainties in generator model of the AVR system.

## 7. CONCLUSIONS

In this paper, comparative performance study of the time domain optimal tuning approach with TLBO for the PID and FOPID controller parameters is done to control AVR system. During the optimization of the PID and FOPID controller parameters, a nonlinear objective function consisting of overshoot, rise time, settling time and steady-state error is used. The performance of the AVR with the proposed controllers is compared to the other PID and FOPID controllers tuned by already proposed optimization algorithms in the literature. The comparative simulation results show that the proposed controllers exhibit more improved dynamic responses as compared to the existing controllers such as PSO, GA, CAS and CS based PID and FOPID controllers. Moreover, robustness testing of the proposed controllers as well as the other different optimized controllers are carried out under dynamic parameter uncertainties of the AVR. It is found that the proposed PID and FOPID controllers have more robust stability and better dynamic responses as compared to the other controllers under model uncertainties.

## REFERENCES

- [1] Wong C-C, Li S-A, Wang H-Y. Hybrid evolutionary algorithm for PID controller design of AVR system, Journal of the Chinese Institute of Engineers, 2009; 32 (2): 251-264.
- [2] Gaing Z-L. A particle swarm optimization approach for optimum design of PID controller in AVR system, IEEE Trans. Energy Convers., 2004; 19 (2): 384-391.

- [3] Mohanty P-K, Sahu B-K, Panda S. Tuning and assessment of proportional–integral– derivative controller for an automatic voltage regulator system employing local unimodal sampling algorithm, *Electr. Power Compon. Syst.*, 2014; 42 (9): 959-969.
- [4] Zhu H, Li L, Zhao Y, Guo Y, Yang Y. CAS algorithm-based optimum design of PID controller in AVR system, *Chaos Solitons Fractals*, 2009; 42 (2): 792-800.
- [5] Panda S, Sahu B-K, Mohanty P-K. Design and performance analysis of PID controller for an automatic voltage regulator system using simplified particle swarm optimization, *Journal of the Franklin Institute*, 2012; 349 (8): 2609-2625.
- [6] Gozde H, Taplamacioglu M C. Comparative performance analysis of artificial bee colony algorithm for automatic voltage regulator (AVR) system, *J. Frankl. Inst.*, 2011; 348: 1927-1946.
- [7] Kim D H. Hybrid GA–BF based intelligent PID controller tuning for AVR system, *Applied Soft Computing*, 2011; 11 (1); 11-22.
- [8] Chatterjee S, Mukherjee V. PID controller for automatic voltage regulator using teaching–learning based optimization technique, *Electr. Power Energy Syst.*, 2016; 77: 418-429.
- [9] Blondin M J, Sanchis J, Sicard P, Herrero J M. New optimal controller tuning method for an AVR system using a simplified ant colony optimization with a new constrained Nelder–Mead algorithm, *Appl. Soft Comput.*, 2018; 62: 216-229.
- [10] Mouayad A S. A novel optimal PID plus second order derivative controller for AVR system, *Engineering Science and Technology, an International Journal*, 2015; 18 (2): 194-206,
- [11] Bingul Z, Karahan O. A novel performance criterion approach to optimum design of PID controller using cuckoo search algorithm for AVR system, *Journal of the Franklin Institute*, 2018; 355 (13): 5534-5559.
- [12] Podlubny I. Fractional-order systems and  $PI^\lambda D^\mu$  controllers, *IEEE Transactions on Automatic Control*, 1999; 44 (1): 208-2014.
- [13] Zamani M, Karimi-Ghartemani M, Sadati N, Parniani M. Design of a fractional order PID controller for an AVR using particle swarm optimization, *Control Engineering Practice*, 2009; 17 (12): 1380-1387.
- [14] Tang Y, Cui M, Hua C, Li L, Yang Y. Optimum design of fractional order  $PI^\lambda D^\mu$  controller for AVR system using chaotic ant swarm, *Expert Systems with Applications*, 2012; 39 (8): 6887-6896.
- [15] Pan I, Das S. Chaotic multi-objective optimization based design of fractional order  $PI^\lambda D^\mu$  controller in AVR system, *International Journal of Electrical Power & Energy Systems*, 2012; 43 (1): 393-407.
- [16] Sikander A, Thakur P, Bansal R C, Rajasekar S. A novel technique to design cuckoo search based FOPID controller for AVR in power systems, *Computers & Electrical Engineering*, 2018; 70: 261-274.

- [17] Suri babu A G, Chiranjeevi B T. Implementation of Fractional Order PID Controller for an AVR System Using GA and ACO Optimization Techniques, IFAC-PapersOnLine, 2016; 49 (1): 456-461.
- [18] Zhang D L, Tang Y G, Guan X P. Optimum Design of Fractional Order PID Controller for an AVR System Using an Improved Artificial Bee Colony Algorithm, Acta Automatica Sinica, 2014; 40 (5): 973-979.
- [19] Rao R V, Savsani V J, Vakharia D P. Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems, Computer-Aided Design, 2011; 43 (3), 303-315.
- [20] Oustaloup A. La commande CRONE: commande robuste d'ordre non entire, Herme's, Paris, 1991.