



Article

A Marine-Predator-Algorithm-Based Optimum FOPID Controller for Enhancing the Stability and Transient Response of Automatic Voltage Regulators

Abdullah M. Noman ^{1,*}, Sulaiman Z. Almutairi ¹, Mokhtar Aly ², Mohammed H. Alqahtani ¹,
Ali S. Aljumah ¹ and Emad A. Mohamed ^{1,3}

¹ Department of Electrical Engineering, College of Engineering, Prince Sattam bin Abdulaziz University, Al Kharj 16278, Saudi Arabia; s.almutairi@psau.edu.sa (S.Z.A.)

² Facultad de Ingeniería, Arquitectura y Diseño, Universidad San Sebastián, Bellavista 7, Santiago 8420524, Chile

³ Department of Electrical Engineering, Faculty of Engineering, Aswan University, Aswan 81542, Egypt

* Correspondence: a.noman@psau.edu.sa

Abstract: An improved design optimization method for fractional-order-based proportional integral derivative (FOPID) controllers is proposed in this paper to enhance the stability and transient response of automatic voltage-regulator systems. The FOPID represents a higher degree-of-freedom controller through having five tunable parameters, compared with only three parameters in the integer-order PID controller. In the literature, the performance of the FOPID is highly determined through the design method and its parameter-determination process. Recently, optimum design of the FOPID has found wide employment in several engineering applications through using metaheuristic optimization algorithms. In this paper, an improved method for the FOPID's parameter optimization is proposed for AVR applications using the marine predator optimization algorithm (MPA). The proposed MPA–FOPID controller is verified through comparing its performance with other featured and newly developed optimization algorithms. The proposed MPA–FOPID comparative analysis has been proven to have better stability, frequency response, robustness, faster response, and disturbance-rejection ability over the other studied methods in this paper.

Keywords: automatic voltage regulators; fractional-order PID controllers; marine predator algorithm (MPA); metaheuristic optimization; stability



Citation: Noman, A.M.; Almutairi, S.Z.; Aly, M.; Alqahtani, M.H.; Aljumah, A.S.; Mohamed, E.A. A Marine-Predator-Algorithm-Based Optimum FOPID Controller for Enhancing the Stability and Transient Response of Automatic Voltage Regulators. *Fractal Fract.* **2023**, *7*, 690. <https://doi.org/10.3390/fractalfract7090690>

Academic Editor: Norbert Herencsar

Received: 9 June 2023

Revised: 29 August 2023

Accepted: 30 August 2023

Published: 16 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Overview

Recently, wide continuous variations exist in electrical power systems due to the changes in the type of energy source, grid modeling, and load behaviors. Recently, renewable-based generation started to denominate the new sources' installation, followed by the corresponding grid properties changing. Based on that, achieving stable and constant voltage magnitudes and frequencies have become essential objectives for the design of control systems [1,2]. Fluctuating frequency and voltage magnitudes may lead to the degraded performance of their connected loads, especially their reliability and useful operating lifetime. The power system's active power, reactive power, and associated power losses are directly linked with the variations in their frequency and voltage magnitudes. In cases of existing slight deviations in voltage magnitudes, sufficient changes exist in the reactive power's magnitude. When voltage magnitude deviates out of the predefined $\pm 5\%$ limit around its rated voltage, the useful lifetime and operating efficiency of connected power system components/appliances are accordingly highly reduced [3].

Fortunately, voltage deviations can be properly controlled, and their effects consequently mitigated. The voltage can be regulated at various existing levels in power systems

on the generation side, and/or on the transmission/distribution sides. On the generation side, automatic voltage regulators (AVRs) are employed for controlling voltage deviations [4], whereas on the transmission/distribution sides, various existing reactive power compensation devices can be utilized, such as flexible AC transmission systems (FACTS), transformers' tap changers, filter devices, etc. The AVR systems have verified excellent regulation performance on the generation sides of synchronous generators. Optimized design of AVR parameters is essential for obtaining the best performance of these devices, considering the various uncertainties and parameter variations in AVRs, sensors, gains, etc. [5].

1.2. Literature Review

The integer order (IO) schemes have found wide use in AVR applications, especially the PID control scheme. It has proven better dynamic response and more stable performance, owing to its robustness and simple implementation. However, restrictions exist in the use of PID control schemes in modern control and industrial applications, due to their performance's dependency on and sensitivity to the gains' value-selection process [6]. Trial-and-error tuning has been presented in the literature; however, it consumes more time without ensuring an optimum control-parameter selection process. Diverse traditional tuning methods for PID controllers exist in the literature, such as the Ziegler–Nichols and Cohen–Coon methods. However, they are based on selecting a particular operating point for the tuning process when linearly evaluating the model. This, in turn, leads to the incapability to drive optimum gains and high overshoot peaks, and, accordingly, long oscillations exist in the system [7].

Additional improvements in AVR control have been proposed using fractional order (FO) control schemes. Using FO calculus with PID control in the FOPID provides enhanced performance and better degree of freedom in the control design [8]. The classical IO-based PID has three tunable control parameters (proportional (P), integral (I), and derivative (D) gains), whereas the FOPID has five tunable control parameters (P, I, and D gains in addition to the FO operators λ and μ for the I and D terms). Accordingly, FOPID-based AVR systems have better transients, more stability, and more robust performances than traditional IO-based PID AVR systems. Therefore, the FOPID has been utilized in this paper for AVR applications. The performance improvements of FOPID control can be achieved through optimizing its parameters to fulfill predefined criteria of system response (fitness/cost/objective function). Among the presented methods, metaheuristic-based optimization methods have been widely used, and are proven to have better tuning of parameters [9,10]. Another AVR control method based on PID plus the second-order derivative (PIDD2) AVR controller has been presented in [11]. A modified version of the Runge–Kutta optimization (RUN) algorithm has been proposed for optimizing the PIDD2 parameters.

Table 1 summarizes the applied optimization algorithms for tuning AVR control systems. The table shows that several algorithms have been applied in the literature for tuning PID and FOPID AVR controllers. The performance of these algorithms differs according to the operating principle of each algorithm. Recently, several new optimization algorithms have been proposed and showed improved performance in several applications. The marine predator algorithm (MPA) has been presented in [12]. It has proved better performance in the energy management of microgrids, several load-frequency control applications [13], parameter determinations, reconfiguration of PV and fuel cells [14,15], etc. Moreover, other optimizing algorithms have recently been presented for different applications. However, these algorithms have been applied in the AVR control design optimization. Therefore, their evaluation and comparisons are included in this paper.

Table 1. Summary of control schemes and optimization algorithms of AVR systems in the literature.

Ref.	Controller	Optimizer Algorithm
Ref. [16]	PID	Particle Swarm Optimization (PSO)
Ref. [17]	PID	Artificial Bee Colony (ABC)
Ref. [18]	PID	Biography-Based Optimization (BBO)
Ref. [19]	PID	Tree-Seed Algorithm (TSA)
Ref. [20]	PID	Grasshopper Optimization Algorithm (GOA)
Ref. [21]	PID	Pattern Search Algorithm (PSA)
Ref. [22]	PID	Whale Optimization Algorithm (WOA)
Ref. [23]	PID	Improved Whale Optimization Algorithm (IWOA)
Ref. [24]	PID	Genetic Algorithm (GA)
Ref. [25]	PID	Cuckoo Search (CS) Algorithm
Ref. [26]	PID	Sine–Cosine Algorithm (SCA)
Ref. [27]	PID	Improved Kidney Inspired Algorithm (IKA)
Ref. [17]	PID	Differential Evolution (DE)
Ref. [28]	PID	Continuous FireFly Algorithm (CFA)
Ref. [29]	PID	Symbiotic Organisms Search (SOS) Algorithm
Ref. [30]	PID	Salp Swarm Algorithm (SSA)
Ref. [31]	PID	Bacterial Foraging Optimization Algorithm (BFOA)
Ref. [32]	PID	Gravitational Search Algorithm (GSA)
Ref. [33]	PID	Ant Lion Optimizer (ALO)
Ref. [34]	PID	Local Unimodal Sampling (LUS) Algorithm
Ref. [35]	PID	Ant Colony Optimizer with Nelder–Mead (ACO-NM)
Ref. [36]	FOPID	Particle-Swarm Optimization (PSO)
Ref. [37]	FOPID	Genetic Algorithm (GA)
Ref. [38]	FOPID	Artificial-Bee Colony Optimizer (CNC-ABC)
Ref. [39]	FOPID	Chaotic Ant Swarm (CAS)
Ref. [40]	FOPID	Multi-Objective Extremal Optimization (MOEO)
Ref. [41]	FOPID	Sine–Cosine Algorithm (SCA)
Ref. [42]	FOPID	Chaotic Yellow Saddle Goatfish Algorithm (CYSGA)
Ref. [43]	FOPID (method 1)	Improved Multi-Objective NSGA-II with Henon Map
Ref. [44]	FOPID (method 2)	Cuckoo Search (CS)
Ref. [37]	FOPID (method 3)	Particle Swarm Optimization (PSO)
Ref. [45]	FOPID (method 4)	Salp Swarm Optimization (SSO)
Ref. [46]	FOPID (method 5)	Multi-Objective NSGA-II with Chaotic Map
Proposed	FOPID	Marine Predator Algorithm (MPA)

1.3. Motivation

Based on the above-described literature review, obtaining stable and satisfactory AVR responses can be achieved with the proper designs and determinations of the best parameters. In addition, various existing optimization algorithms differ in their principal operating

philosophy, which is reflected in their outputted best parameters for the optimization problem. Additionally, the FOPID-based AVR systems achieved better performance metrics than classical PID controllers. However, the evaluation of some recent optimization algorithms and the superior performance of the MPA optimizer has not been investigated. The MPA optimizer has achieved enhanced performance in several applications [12–15,47]. The main highlights of the operating principles and governing policies of MPA for achieving optimized foraging, interactions, and memories of marine predators are as follows [12]:

- The Lévy-based strategy is used by marine predators for environments with low prey concentrations, and Brownian movement is used for areas with abundant prey. Both strategies share the same percentage of traversing various habitats within their lifetime;
- The behavior of predators is changed by natural environment-based effects, such as eddy formations, or by human-based effects, such as fish aggregating devices (FAD), for seeking areas with prey distribution;
- The velocity ratio (VR), which is represented by the ratio of predator velocity to prey velocity, is used for determining the best strategy. At low VR ($VR = 0.1$), Lévy is the best predators' strategy, whether prey are moving using Brownian or Lévy strategies;
- At unity VR ($VR = 1$), Brownian movement represents the best predator's strategy when prey movement is through Lévy. The other scenarios depend on the system's size.
- At high VR ($VR \geq 10$), the predator's best strategy is not doing any movements at all, whereas prey's movement is made through the Brownian or Lévy strategies;
- The MPA benefits the good memory of marine predators at reminding their associates in addition to locations of successful foraging.

1.4. Paper Contributions

The paper's contributions are summarized as follows:

- The recent powerful MPA optimizer is presented and applied with the FOPID to improve the AVR controller. Based on authors' knowledge at the submission date, this is the first time the MPA optimizer is presented in AVR controller design. The study is not limited to applying the MPA optimizer, it also presents a performance evaluation of MPA with several recently developed optimization algorithms are presented in this paper.
- Additionally, the use of FOPID provides higher freedom with its additional parameters, which help improve the performance of AVR systems. The obtained optimum FOPID AVR controller is compared with the previously determined optimum FOPID using other optimizer techniques from the literature. The conducted design and analysis clearly demonstrate superior results of the proposed MPA-tuned FOPID AVR controller.
- Better convergence performance and the determined parameters' accuracy of the optimum FOPID AVR controller are presented in this paper using the MPA optimizer. The proposed MPA-based method is compared with recent and existing optimization methodologies. Additionally, several statistical tests are performed to make fair comparisons of optimization methods. The obtained results over 30 runs and the statistical analysis of the results confirm the superiority of MPA and its feasibility in AVR controller design.

The remaining of the paper is organized as follows: Section 2 presents state-of-the-art mathematical modeling of the AVR system, and the open loop analysis of AVR systems is presented in Section 3. A detailed description of the proposed optimization methodology is provided in Section 4. The AVR control and optimization process are detailed in Section 5. The obtained results are shown in Section 6, and paper conclusions are provided in Section 7.

2. Mathematical Description Modeling for AVR Systems

Figure 1 shows simplified general AVR modeling for this study. The control system is responsible for regulating the output voltage of the generator. It controls the excitation of the generator using a signal representing the error of subtracting the sensed output terminal voltage of the generator from the predefined reference voltage value. The operation of the AVR control can be explained at voltage drop or raise as follows:

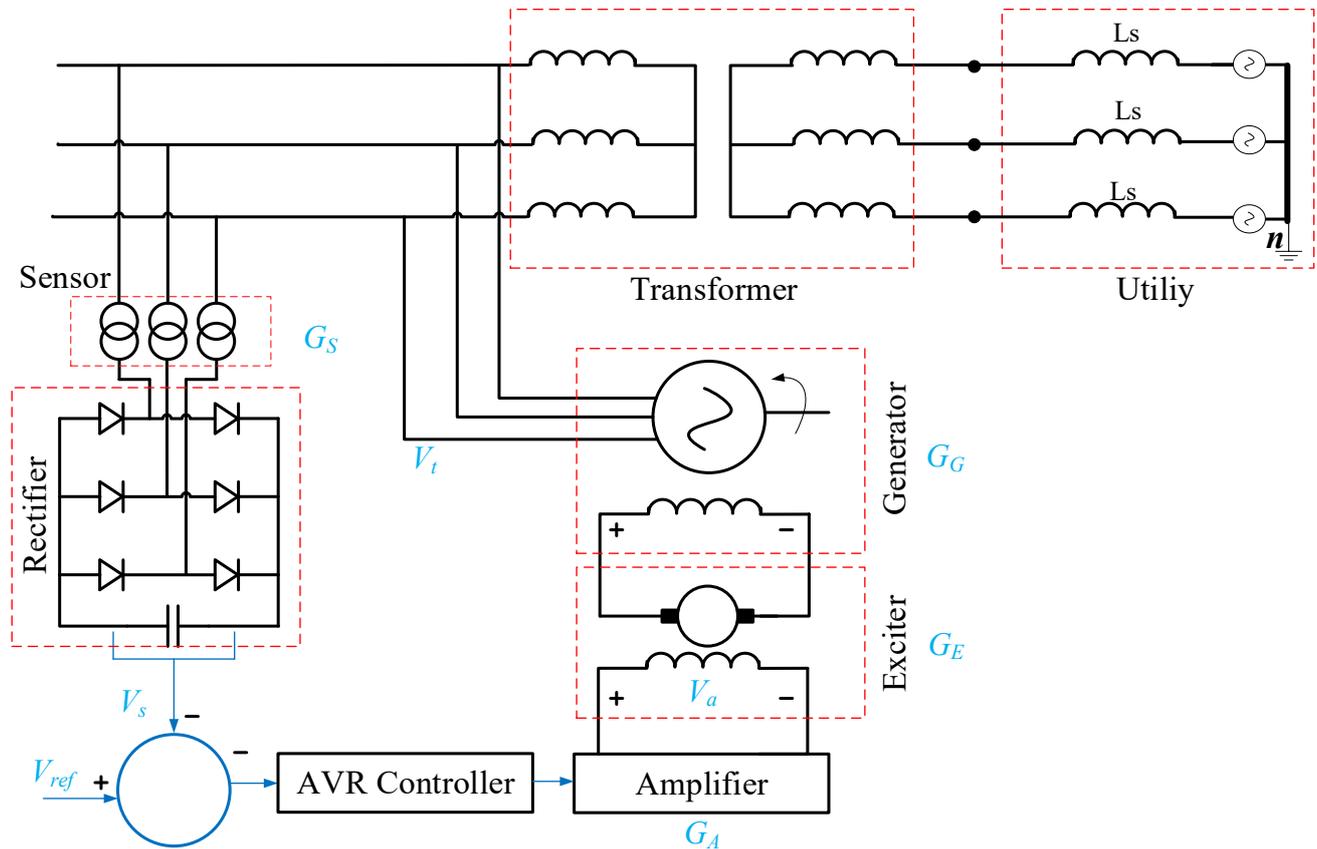


Figure 1. Schematic representation of generalized AVR components.

- **At voltage drop:** The output terminal voltage drops when there are increased loading conditions. In this scenario, the error between the sensed terminal voltage and the reference voltage increases with a positive value. Accordingly, the generator's excitation increases until the sensed voltage reaches the predefined reference voltage value. This process continues until the sensed voltage equals the reference voltage. After reaching this condition, the generator's excitation is maintained constant to preserve a stable supply voltage for all connected loads.
- **At voltage raise:** At reduced load values, the output voltage increases, and hence an increase in the error signal happens, but with a negative value. Then, the generator's excitation is reduced until it achieves equal sensed and reference terminal voltage values. Then, the generator's excitation is maintained constant to stabilize the output terminal voltage.

The field windings of the generator have very high values for their inductance, and hence unavoidable conditions of load switching are generated in power systems. This, in turn, imposes several challenges on the AVR design process. The transfer function (TF) of the AVR system is modeled using the Laplace transform. The various AVR components (amplifier, exciter, generator, and sensor) are modeled using linearized components to facilitate TF modeling processes. The TF of AVR components ($G_A(s)$ for amplifier, $G_E(s)$

for exciter, $G_G(s)$ for generator, and $G_S(s)$ for sensor) are represented with their reported parameters' ranges in the literature as follows [41,48]:

$$\begin{aligned}
 G_A(s) &= \frac{K_A}{1 + sT_A}, \quad \text{and} \quad 10 \leq K_A \leq 400, \quad 0.02 \text{ s} \leq T_A \leq 0.1 \text{ s} \\
 G_E(s) &= \frac{K_E}{1 + sT_E}, \quad \text{and} \quad 1 \leq K_E \leq 10, \quad 0.4 \text{ s} \leq T_E \leq 1 \text{ s} \\
 G_G(s) &= \frac{K_G}{1 + sT_G}, \quad \text{and} \quad 0.7 \leq K_G \leq 1, \quad 1 \text{ s} \leq T_G \leq 2 \text{ s} \\
 G_S(s) &= \frac{K_S}{1 + sT_S}, \quad \text{and} \quad 1 \leq K_S \leq 2, \quad 0.001 \text{ s} \leq T_S \leq 0.06 \text{ s}
 \end{aligned}
 \tag{1}$$

where $K_A, K_E, K_G,$ and K_S represent gains, and $T_A, T_E, T_G,$ and T_S represent time constants of the amplifier, exciter, generator, and sensor, respectively. The error between the measured voltage V_m and its reference voltage V_{ref} is represented as E_v in Figure 2. The AVR has to control the error E_v to be minimum and to equal zero at steady state, whereas the AVR controller $C(s)$ can be represented for PID $C_{PID}(s)$, and for FOPID $C_{FOPID}(s)$ as follows:

$$\begin{aligned}
 G_{PID}(s) &= K_P + \frac{K_I}{s} + K_D s \\
 G_{FOPID}(s) &= K_P + \frac{K_I}{s^\lambda} + K_D s^\mu
 \end{aligned}
 \tag{2}$$

The complete first-order TF modeling of AVR components is shown in Figure 2. The complete system's TF representing the output to input TF $G_{sys}(s)$ can be expressed as follows:

$$G_{sys}(s) = \frac{C(s)G_A(s)G_E(s)G_G(s)}{1 + C(s)G_A(s)G_E(s)G_G(s)G_S(s)}
 \tag{3}$$

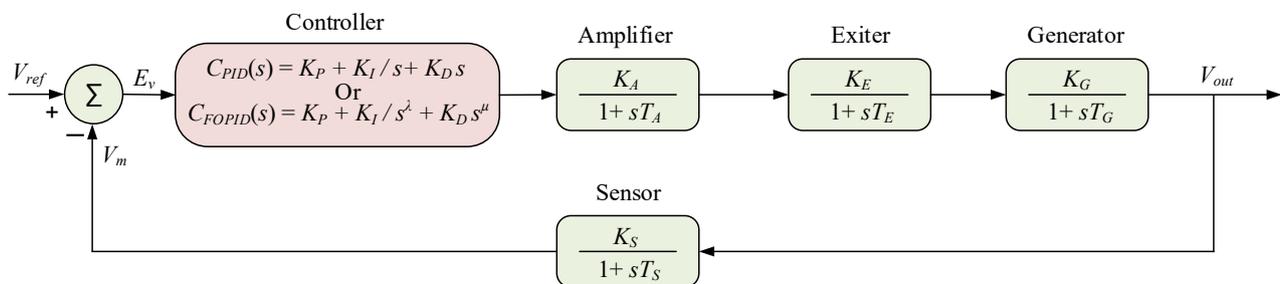


Figure 2. Model TFs of AVR components.

3. System Characteristics without Controller

The considered parameters' values (1) are as follows: $K_A = 10, K_E = 1, K_G = 1, K_S = 1, T_A = 0.1, T_E = 0.4, T_G = 1,$ and $T_S = 0.01$ [42], whereas the generator's gain K_G depends on its loading level. At the no-load condition, K_G equals 0.7, and it increases with loading to 1 at the nominal loading condition [42]. In this section, the analysis of the AVR system without the controller is carried out. Figure 3 shows different step responses of AVR systems at various loading conditions (varying K_G at 0.7, 0.8, 0.9, and 1.0). The associated frequency response and the root locus plots of the effects of different loading conditions are shown in Figures 4 and 5, respectively. It can be seen that the generator's gain affects the system response in step time, the frequency response, and the root locus pole/zero locations.

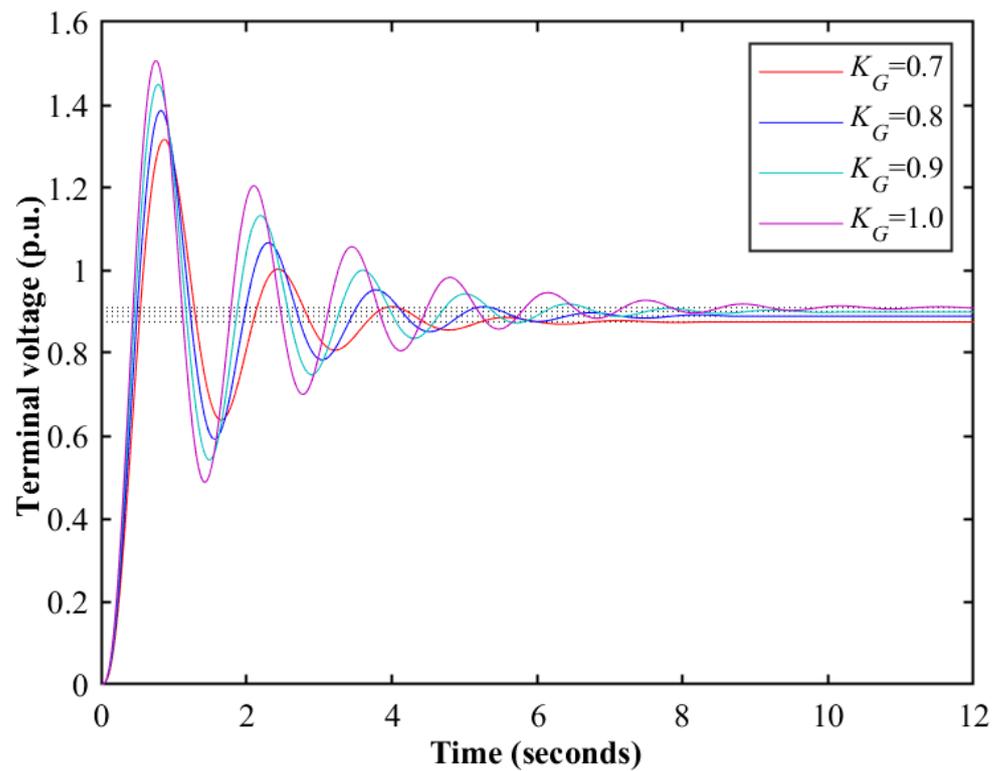


Figure 3. Step responses of AVR systems without controller at various loading conditions.

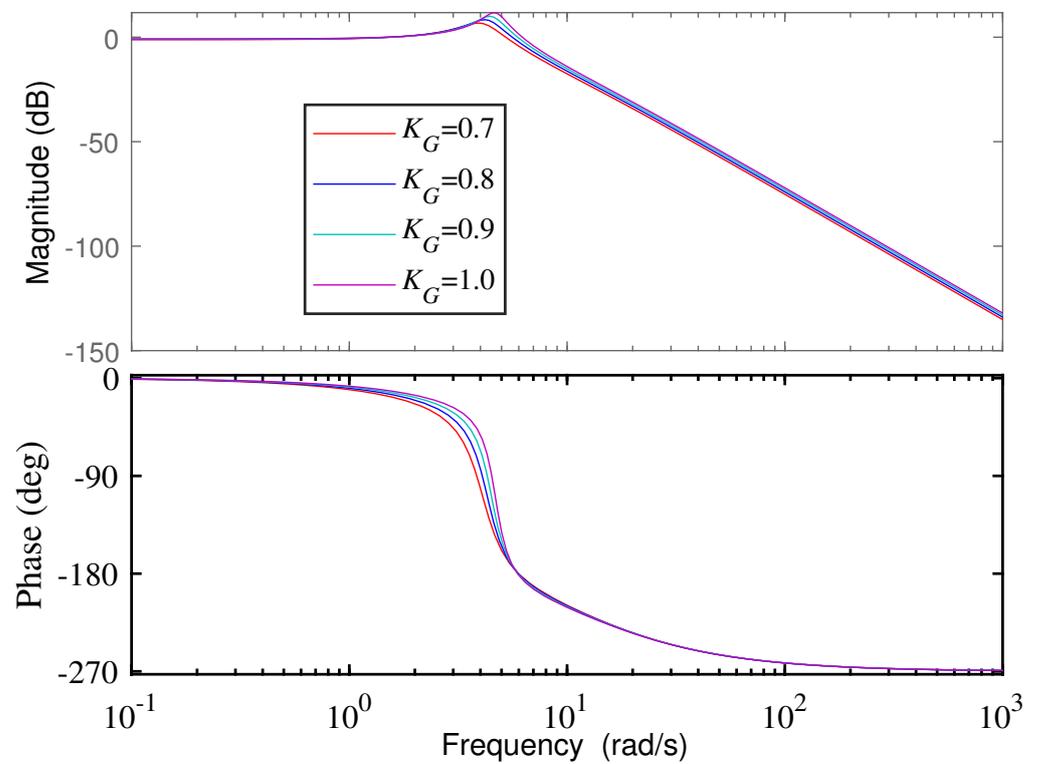


Figure 4. Frequency responses of AVR systems without controller at various loading conditions.

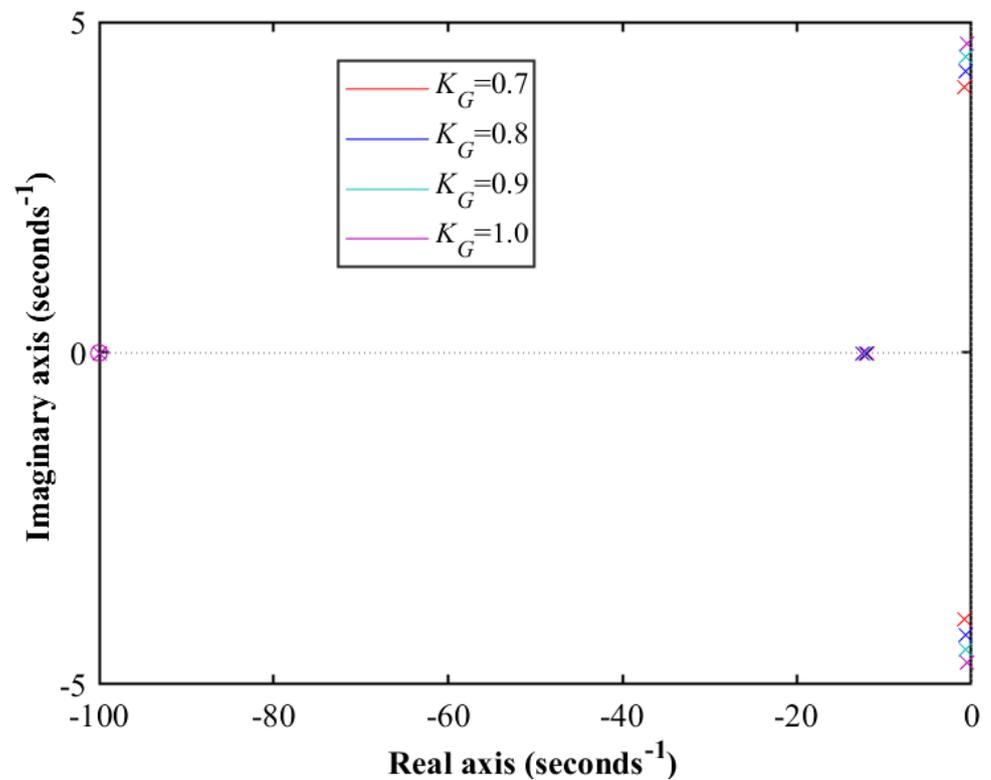


Figure 5. Root locus plots of AVR systems without controller at various loading conditions.

The measured effects of the AVR system's response are summarized in Table 2. The overshoot percentage and settling time increase at higher values of K_G , whereas the rise time and steady state error decrease with the increase in K_G . For the frequency response bode diagram, the gain margin and the phase margin decrease with increasing K_G . In addition, the variations in K_G values affect the closed-loop poles locations of the AVR system. Comparisons of the open-loop response values are included in the table.

Table 2. AVR system's responses at various loading conditions.

Type	Parameters	Units	$K_G = 0.7$	$K_G = 0.8$	$K_G = 0.9$	$K_G = 1.0$
Step Resp.	Overshoot	Percentage %	50.4956	55.9611	61.0183	65.7226
	Rise Time	Seconds	0.3172	0.2944	0.2760	0.2607
	Settling Time	Seconds	4.8980	5.4149	6.4257	6.9865
	Steady-State Err.	Per Unit (p.u.)	0.1220	0.1158	0.1024	0.0938
Freq. Resp.	Gain Margin	dB	4.19	2.11	0.07	-2
	Phase Margin	Degree (°)	14.6	6.7	0.2	-5.3
Root Locus	Closed Loop System Poles		-12.49	-12.31	-12.13	-11.93
			-99.97	-99.97	-99.98	-99.98
			-0.52 + 4.66i	-0.61 + 4.47i	-0.7 + 4.25i	-0.8 + 4.02i
			-0.52 - 4.66i	-0.61 - 4.47i	-0.7 - 4.25i	-0.8 - 4.02i

4. MPA Optimization Methodology

The MPA represents a population-based optimizer, wherein uniform distribution over searching space exists in its initial solution. It is expressed as follows [12]:

$$X_0 = X_{min} + rand(X_{max} - X_{min}) \quad (4)$$

where X_{max} and X_{min} are upper and lower boundaries of variables, and $rand$ represents a uniform-random vector (between 0 and 1). The Elite, which is used for representing top predators, and prey are expressed as follows [47]:

$$Elite = \begin{bmatrix} X_{1,1}^I & X_{1,2}^I & \cdots & \cdots & X_{1,d}^I \\ X_{2,1}^I & X_{2,2}^I & \cdots & \cdots & X_{2,d}^I \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ X_{n,1}^I & X_{n,2}^I & \cdots & \cdots & X_{n,d}^I \end{bmatrix}_{n \times d} \quad (5)$$

$$Prey = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & \cdots & X_{1,d} \\ X_{2,1} & X_{2,2} & \cdots & \cdots & X_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ X_{n,1} & X_{n,2} & \cdots & \cdots & X_{n,d} \end{bmatrix}_{n \times d} \quad (6)$$

where \vec{X}^I is top predators' vector, n stands for searching agents' number, d stands for dimensions' number, and $X_{i,j}$ stands for the j^{th} dimension of i^{th} prey. The three operating scenarios of MPA can be expressed as follows:

At High VR Scenario: Within this scenario, the predators' velocity is higher than prey's velocity. It happens during initial optimization iterations. It is mathematically expressed as: while $Iter < \frac{1}{3}Max_{Iter}$ [14],

$$\begin{aligned} \vec{S}_i &= \vec{R}_B \otimes (\vec{Elite}_i - \vec{R}_B \otimes \vec{Prey}_i), \quad i = 1, 2, \dots, n \\ \vec{Prey}_i &= \vec{Prey}_i + P \cdot \vec{R} \otimes \vec{S}_i \end{aligned} \quad (7)$$

where R_B is vector of random numbers for Brownian motion representation, \otimes stands for entry-wise multiplication, P is a constant equal to 0.5, R stands for a random number vector between 0 and 1, and $Iter$ stands for the current iteration's number within the maximum iterations number Max_{Iter} , whereas multiplying R_B with $Prey_i$ simulates prey movements.

At Unity VR Scenario: This scenario happens while $\frac{1}{3}Max_{Iter} \leq Iter < \frac{2}{3}Max_{Iter}$, wherein predators use Brownian strategy and prey use Lévy strategy. Both of exploration and the exploitation happen in this scenario. Then, the population is divided between them, where prey are responsible for the exploitation part and predators are responsible for the exploration part. For the prey's half of the population, it is expressed as [12]:

$$\begin{aligned} \vec{S}_i &= \vec{R}_L \otimes (\vec{Elite}_i - \vec{R}_L \otimes \vec{Prey}_i), \quad i = 1, 2, \dots, n/2 \\ \vec{Prey}_i &= \vec{Prey}_i + P \cdot \vec{R} \otimes \vec{S}_i \end{aligned} \quad (8)$$

where \vec{R}_L stands for Lévy distribution-based random number vector for Lévy movement representation. For the predators' half of the population, it is expressed as [12]:

$$\begin{aligned} \vec{S}_i &= \vec{R}_B \otimes (\vec{R}_B \otimes \vec{Elite}_i - \vec{Prey}_i), \quad i = n/2, \dots, n \\ \vec{Prey}_i &= \vec{Elite}_i + P.CF \otimes \vec{S}_i \\ CF &= \left(1 - \frac{t}{MaxIter}\right)^{2 \frac{Iter}{MaxIter}} \end{aligned} \tag{9}$$

At Low VR Scenario: This scenario happens while $Iter \geq \frac{2}{3}MaxIter$. It is mathematically expressed as [14]:

$$\begin{aligned} \vec{S}_i &= \vec{R}_L \otimes (\vec{R}_L \otimes \vec{Elite}_i - \vec{Prey}_i), \quad i = 1, 2, \dots, n \\ \vec{Prey}_i &= \vec{Elite}_i + P.CF \otimes \vec{S}_i \end{aligned} \tag{10}$$

In addition, FAD effects are mathematically expressed as [12]:

$$\vec{Prey}_i = \begin{cases} \vec{Prey}_i + CF[X_{min} + \vec{R} \otimes (X_{max} - X_{min})] \otimes U & \text{If } r \leq FADs \\ \vec{Prey}_i + [FADs(1 - r) + r](\vec{Prey}_{r1} - \vec{Prey}_{r2}) & \text{If } r > FADs \end{cases} \tag{11}$$

where U stands for binary vector including arrays of 0 and 1, $FADs$ stands for FADs' effects probability, which equals 0.2, r stands for a random number between 0 and 1, and \vec{Prey}_{r1} and \vec{Prey}_{r2} are random indexes within the prey matrix.

5. The AVR Control and Optimization Process

5.1. FO Theory and Representation

The FO control theory offers better flexibility and freedom compared with integer ones. The common representation of FO theory is provided by the Riemann–Liouville methodology, the Grunwald–Letnikov methodology, and the Caputo representation. For Grundwald–Letnikov, the α_{th} FO derivative part of function f and within a to t range limits is:

$$D^\alpha |^t_a = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{r=0}^{\frac{t-a}{h}} (-1)^r \binom{n}{r} f(t - rh) \tag{12}$$

where h refers to step time, and the use of n is for fulfilling the $(n - 1 < \alpha < n)$ condition. The coefficients in binomial are obtained as:

$$\binom{n}{r} = \frac{\Gamma(n + 1)}{\Gamma(r + 1)\Gamma(n - r + 1)} \tag{13}$$

where Gamma refers to function and is defined as:

$$\Gamma(n + 1) = \int_0^\infty t^{n-1} e^{-t} dt \tag{14}$$

Additionally, Liouville and Riemann have provided definitions of derivatives that are capable of avoiding sums and limits. Wherein this definition uses the IO derivatives, the integrals are represented as:

$$D^\alpha |^t_a = \frac{1}{\Gamma(n - \alpha)} \left(\frac{d}{dt}\right)^n \int_a^t \frac{f(\tau)}{(t - \tau)^{\alpha - n + 1}} d\tau \tag{15}$$

The Caputo method representation defines FO derivatives as:

$$D^\alpha |^t_a = \frac{1}{\Gamma(n - \alpha)} \int_a^t \frac{f^{(n)}(\tau)}{(t - \tau)^{\alpha - n + 1}} d\tau \tag{16}$$

whereas $D^\alpha|_a^t$ is referred as:

$$D^\alpha|_a^t = \begin{cases} \alpha > 0 \rightarrow \frac{d^\alpha}{dt^\alpha} & \text{FO derivative} \\ \alpha < 0 \rightarrow \int_{t_0}^{t_f} dt^\alpha & \text{FO integral} \\ \alpha = 0 \rightarrow 1 \end{cases} \quad (17)$$

Implementing of an FO-based control is commonly performed using an Oustaloup recursive approximation (ORA). It showed a suitable representation of FO derivatives for digital implementations in real-time applications. Also, it showed a better way to optimize FO control design using optimization techniques. Therefore, ORA FO representation is selected in this work to represent the FOPID AVR controller. The approximate mathematical ORA representation of the α^{th} derivative (s^α) is expressed as:

$$s^\alpha \approx \omega_h^\alpha \prod_{k=-N}^N \frac{s + \omega_k^z}{s + \omega_k^p} \quad (18)$$

where ω_k^p and ω_k^z refer to pole and zero locations in ω_h sequences, respectively, and they are calculated as:

$$\omega_k^z = \omega_b \left(\frac{\omega_h}{\omega_b}\right)^{\frac{k+N+\frac{1-\alpha}{2}}{2N+1}} \quad (19)$$

$$\omega_k^p = \omega_b \left(\frac{\omega_h}{\omega_b}\right)^{\frac{k+N+\frac{1+\alpha}{2}}{2N+1}} \quad (20)$$

$$\omega_h^\alpha = \left(\frac{\omega_h}{\omega_b}\right)^{-\frac{\alpha}{2}} \prod_{k=-N}^N \frac{\omega_k^p}{\omega_k^z} \quad (21)$$

where the ORA approximated representations contains $(2N + 1)$ number of poles and zeros. Thence, N refers to ORA's filter order within $(2N + 1)$ frequency ranges. The implemented ORA representation in this paper relies on using $(N = 5)$ in frequency range of $(\omega \in [\omega_b, \omega_h])$, and accordingly selected between $[0.001, 1000]$ rad/s.

5.2. Proposed Optimized FOPID AVR Control

Figure 6 shows the schematic diagrams of PID- and FOPID-based AVR systems. It can be seen that the PID possesses three tunable gains, whereas the FOPID possesses five different tunable parameters. The FOPID has three gains and two FO operators, which increase its flexibility and degree of freedom compared to IO-based PID controller. The parameter optimization process is highly dependent on the objective function employed to drive the optimizer algorithm. The controller parameters are selected to minimize the predefined objective function. Usually, the measured output voltage and/or the error signals are used for constructing the objective functions.

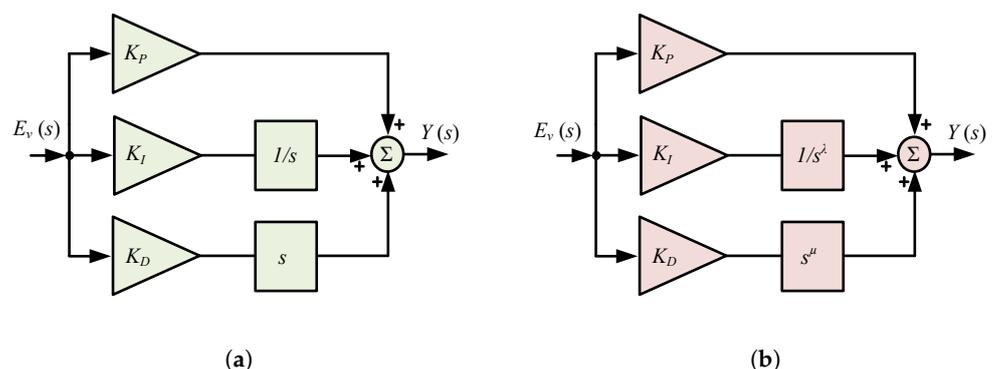


Figure 6. Schematic structure of AVR control schemes. (a) PID controller; (b) FOPID controller.

There are four main error-based objective functions in the literature. Their general representation is as follows:

1. Integral-squared-errors (ISE),

$$ISE = \int \sum_{i=1}^m (e_i^2) dt \tag{22}$$

2. Integral time-squared-errors (ITSE),

$$ITSE = \int \sum_{i=1}^m (e_i^2) t \cdot dt \tag{23}$$

3. Integral-absolute-errors (IAE),

$$IAE = \int \sum_{i=1}^m abs(e_i) dt \tag{24}$$

4. Integral time-absolute-errors (ITAE),

$$ITAE = \int \sum_{i=1}^m abs(e_i) t \cdot dt \tag{25}$$

Moreover, the various error and/or performance indices can be combined in the summation and employed for constructing functions with multiple objectives. Weighting factors for each error or objective can also be added to the general form to manage the priority of each element. Table 3 summarizes the various objective function representations in the literature.

Table 3. Existing objective functions in the literature.

Objective Function	Type	Ref.
$OF = IAE = \int e_v dt$	Single	Ref. [38]
$OF_1 = IAE, OF_2 = 1000 E_{ss} , OF_3 = T_s$	Multiple	Ref. [40]
$OF_1 = \omega_{cf}, OF_2 = P_m$	Multiple	Ref. [43]
$OF = ITAE = \int t \cdot e_v dt$	Single	Ref. [45]
$OF = (\omega_1 \cdot OS)^2 + \omega_2 \cdot T_s^2 + \frac{\omega_3}{dV_{max}^2}$	Single	Ref. [37]
$OF = \omega_1 \cdot OS + \omega_2 \cdot T_r + \omega_3 \cdot T_s + \omega_4 \cdot E_{ss} + \int (\omega_5 \cdot e_v dt + \omega_6 \cdot V_f(t)^2) dt + \frac{\omega_7}{P_m} + \frac{\omega_8}{G_m}$	Single	Ref. [36]
$OF = (1 - e^{-\beta}) \cdot (OS + E_{ss}) + e^{-\beta} \cdot (T_s - T_r)$	Single	Ref. [39]
$OF_1 = ITSE = \int t \cdot e_v^2 dt, OF_2 = \int t \cdot \Delta u^2(t) dt, OF_3 = ITSE = \int t \cdot e_{load}^2 dt$	Multiple	Ref. [46]
$OF = \omega_1 \cdot OS + \omega_2 \cdot T_s + \omega_3 \cdot E_{ss} + \omega_4 \cdot \int e_v dt + \omega_5 \cdot \int u^2 dt$	Single	Ref. [37]

OS = overshoot, T_r = rise time, T_s = settling time, E_{ss} = steady-state error, e_v = error voltage, u = control signal, P_m = phase margin, G_m = gain margin, ω_{cf} = gain crossover-frequency, e_{load} = error signal during load disturbance, dV_{max} = maximum-point of voltage signal-derivative, ω_1 – ω_8 = weighting factors.

The ITAE is used as an objective function in this paper for the optimization problem due to its generality. Figure 7 shows the schematic diagram of the proposed AVR controller optimization process. The measured error voltage signal is used to calculate the objective function for each simulation run, and the MPA optimizer works to determine the optimum parameters as clarified in Algorithm 1. It can be seen that there are three tunable parameters

with the PID controller and five tunable ones with the FOPID controller. The upper and lower boundaries for the proposed optimization process are as follows:

$$\begin{aligned}
 K_P^{min} &\leq K_P \leq K_P^{max} \\
 K_I^{min} &\leq K_I \leq K_I^{max} \\
 K_D^{min} &\leq K_D \leq K_D^{max} \\
 \lambda^{min} &\leq \lambda \leq \lambda^{max} \\
 \mu^{min} &\leq \mu \leq \mu^{max}
 \end{aligned}
 \tag{26}$$

where $(f)^{min}$ and $(f)^{max}$ are lower and upper bounds of the tunable AVR controller's parameter, respectively, whereas K_P^{min} , K_I^{min} , K_D^{min} are set at 1, 0.1, and 0.1, respectively, and K_P^{max} , K_I^{max} , and K_D^{max} are set at 2, 1, and 0.4, respectively. In addition, μ^{min} and λ^{min} are set at 1, whereas μ^{max} and λ^{max} are set at 2 in the proposed optimization process. For a fair comparison, all the optimization algorithms are set at 50 iterations, and the population size is set at 8.

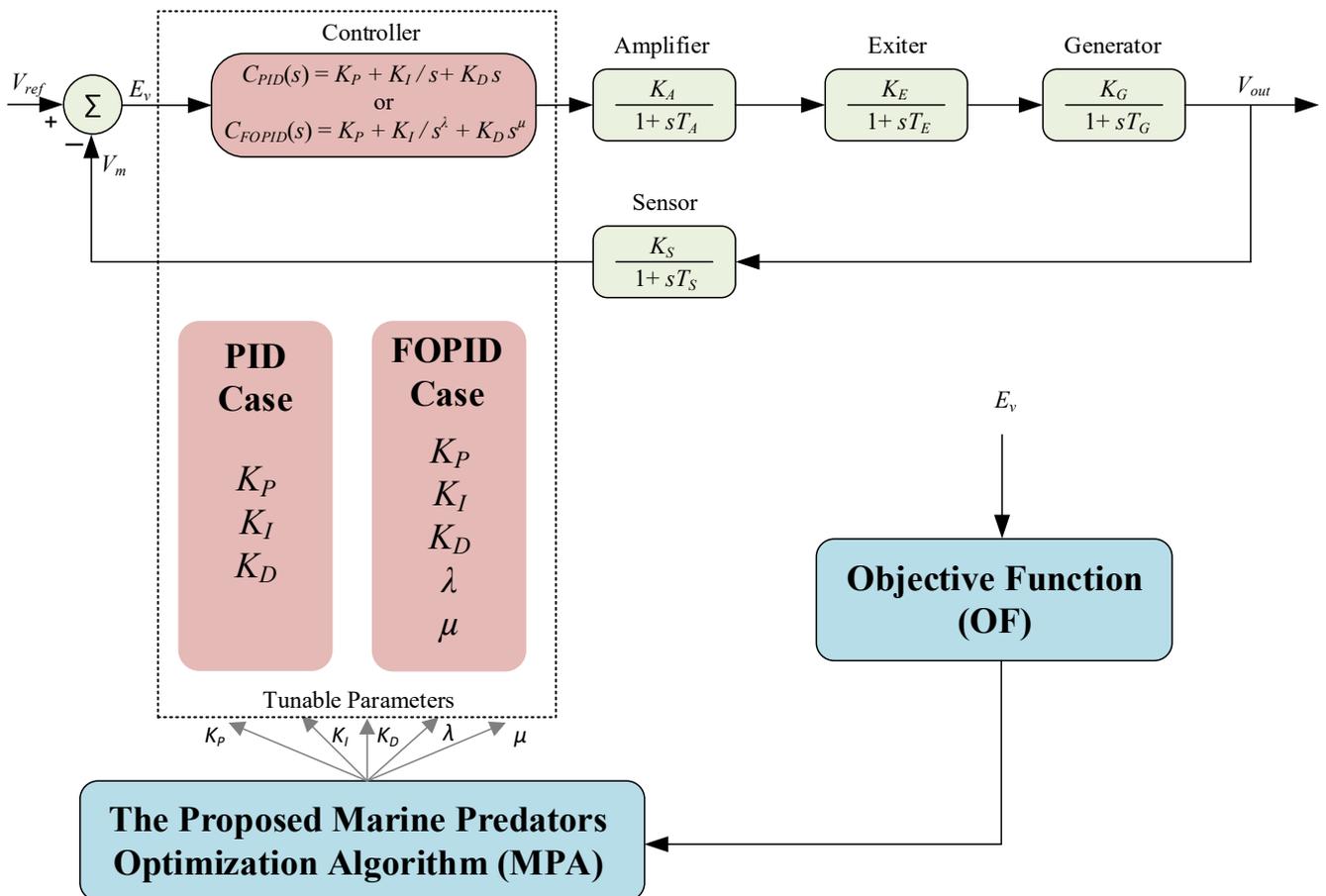


Figure 7. Proposed MPA-based AVR controller optimization.

Algorithm 1 Pseudo-code representing proposed optimization process using MPA

Define MPA parameters (Size of population, and maximum iteration Max_{Iter})

Start algorithm initialization using (4)

Construct initial *Elite* using (5)

Initialize the searching agents (Preys) population using (6)

while $Iter < Max_{Iter}$ **do**

 Calculate objective function, and construct *Elite* matrix

if $Iter < Max_{Iter}/3$ **then**

 Update preys using (7)

else if $Max_{Iter}/3 \leq Iter < 2Max_{Iter}/3$ **then**

 For preys' half of population ($i = 1, \dots, n/2$), update preys using (8)

 For predators' half of population ($i = n/2, \dots, n$), update preys using (9)

else if $Iter \geq 2Max_{Iter}/3$ **then**

 Update preys using (10)

end if

 Accomplish the memory savings, and the *Elite* updates

 Applying the FADs effects using (11)

 Calculate the objective function

end while

Return the best solution

6. Results and Discussion

Performance Comparisons with Recent Algorithms

Firstly, comparisons of the performance of MPA optimizers was performed with optimizers featured in the literature recently. The considered optimizers include the Artificial Hummingbirds Algorithm (AHA) [49], arithmetics trigonometric optimizer algorithm (ATOA) [50], Differential Evolutions (DE) [51], Heap-Based Optimization (HBO) [52], Sine-Cosine Optimizer Algorithm (SCA) [53], Slime Mould Algorithm (SMA) [54], and Manta Ray-Foraging Optimization (MRFO) [55]. The proposed AVR controller and optimization algorithms are simulated using MATLAB R2021a software. The AVR system and controller are implemented in Simulink and linked with the m-file program, which contains the optimization algorithms. The optimization process is made 30 times using the same number of iterations, population size, and parameters limits. The obtained results over the 30 runs are summarized in Table 4.

The obtained best FOPID parameters for each optimizer over the 30 runs are summarized in Table 5. It also provides statistical analysis of the best value, worst value, mean value, median value, and standard deviation for each optimizer during the 30 runs. The proposed MPA gives the lowest value of the objective function compared to the studied optimizers of 0.0151, whereas the SMA optimizer provides the highest value of the objective function of 0.0233. This proves the best performance of MPA over the all runs, whereas the statistical analysis shows that the SCA provides the highest value of the worst objective of 0.0763 compared to the lowest value of worst objective of 0.0176 using the proposed MPA optimizer. Additionally, the proposed MPA provides the best results in terms of mean, median, and standard deviation over the 30 runs compared to the studied optimizers. This proves the ability and stability of proposed method over the performed runs and random processes of different optimizers.

Another test based on 100 runs of the MRFO and MPA is performed, and the obtained results are shown in Table A1 in Appendix A. The associated statistical results are summarized in Table 6. It can be seen from the table that the best objective obtained by MPA is 0.01507709, compared to 0.01534512 by the MRFO algorithm. Additionally, the mean value achieved by the MPA is 0.01564708, compared to 0.01704896 by MRFO. This, in turn, confirms the improved design performance using MPA compared to the MRFO algorithm.

Table 4. Obtained objective functions over 30 runs of the studied optimizers.

Run No.	AHA	ATOA	DE	HBO	SCA	SMA	MRFO	MPA
1	0.0294	0.0325	0.0175	0.0178	0.0532	0.0507	0.0168	0.0165
2	0.0175	0.0312	0.0164	0.0170	0.0342	0.0414	0.0156	0.0154
3	0.0192	0.0358	0.0233	0.0209	0.0255	0.0367	0.0162	0.0162
4	0.0159	0.0251	0.0205	0.0202	0.0368	0.0500	0.0155	0.0151
5	0.0265	0.0288	0.0180	0.0206	0.0326	0.0484	0.0165	0.0152
6	0.0196	0.0598	0.0152	0.0205	0.0420	0.0486	0.0178	0.0154
7	0.0191	0.0253	0.0232	0.0173	0.0589	0.0296	0.0176	0.0153
8	0.0166	0.0227	0.0162	0.0233	0.0366	0.0356	0.0159	0.0162
9	0.0257	0.0237	0.0280	0.0180	0.0584	0.0593	0.0174	0.0152
10	0.0207	0.0250	0.0203	0.0314	0.0215	0.0284	0.0156	0.0152
11	0.0320	0.0283	0.0202	0.0200	0.0663	0.0512	0.0167	0.0166
12	0.0264	0.0321	0.0205	0.0265	0.0374	0.0512	0.0159	0.0151
13	0.0209	0.0264	0.0205	0.0242	0.0378	0.0358	0.0155	0.0174
14	0.0287	0.0356	0.0202	0.0162	0.0376	0.0415	0.0159	0.0170
15	0.0271	0.0204	0.0255	0.0445	0.0388	0.0546	0.0155	0.0165
16	0.0231	0.0322	0.0171	0.0166	0.0436	0.0527	0.0155	0.0152
17	0.0219	0.0321	0.0208	0.0257	0.0470	0.0478	0.0155	0.0153
18	0.0196	0.0294	0.0178	0.0161	0.0422	0.0233	0.0183	0.0176
19	0.0273	0.0324	0.0185	0.0237	0.0304	0.0550	0.0180	0.0156
20	0.0223	0.0287	0.0184	0.0228	0.0363	0.0488	0.0175	0.0151
21	0.0157	0.0200	0.0223	0.0267	0.0611	0.0375	0.0155	0.0154
22	0.0178	0.0368	0.0170	0.0187	0.0558	0.0257	0.0157	0.0156
23	0.0280	0.0338	0.0157	0.0301	0.0327	0.0493	0.0153	0.0151
24	0.0170	0.0254	0.0238	0.0267	0.0298	0.0503	0.0154	0.0165
25	0.0222	0.0216	0.0270	0.0182	0.0308	0.0558	0.0179	0.0162
26	0.0220	0.0260	0.0204	0.0200	0.0285	0.0491	0.0155	0.0151
27	0.0187	0.0326	0.0343	0.0373	0.0346	0.0363	0.0160	0.0151
28	0.0305	0.0360	0.0199	0.0185	0.0763	0.0365	0.0170	0.0172
29	0.0245	0.0387	0.0195	0.0206	0.0280	0.0328	0.0155	0.0160
30	0.0198	0.0261	0.0167	0.0251	0.0372	0.0358	0.0155	0.0159

Table 5. Best FOPID parameters and statistical analysis of the studied optimizers over the performed 30 runs.

	AHA	ATOA	DE	HBO	SCA	SMA	MRFO	MPA
Optimum FOPID Parameters								
K_p	1.8295	2.0000	1.7057	1.5215	1.8515	1.5791	1.6506	1.7061
K_i	0.8152	1.0000	0.8263	0.7293	1.0000	0.9417	0.7878	0.8068
K_d	0.3968	0.4000	0.4000	0.3601	0.4000	0.3896	0.3932	0.4000
λ	1.1636	1.1335	1.1158	1.1269	1.0584	1.0234	1.1235	1.1286
μ	1.2512	1.1797	1.2129	1.2073	1.1911	1.1344	1.2093	1.2164
Statistical Parameters								
Best	0.0157	0.0200	0.0152	0.0161	0.0215	0.0233	0.0153	0.0151
Worst	0.0320	0.0598	0.0343	0.0445	0.0763	0.0593	0.0183	0.0176
Mean	0.0225	0.0301	0.0205	0.0228	0.0411	0.0433	0.0163	0.0158
Median	0.0220	0.0291	0.0202	0.0206	0.0373	0.0481	0.0159	0.0155
Std. Deviation	0.0047	0.0076	0.0042	0.0065	0.0131	0.0098	0.0010	0.0008

Table 6. Statistical analysis based on 100 runs of MPA and MRFO algorithms.

Algorithm	Best	Worst	Mean	Median	Std. Deviation
MRFO	0.01534512	0.02311189	0.01704896	0.01641032	0.00168091
MPA	0.01507709	0.01848793	0.01564708	0.01546613	0.00059616

An analysis of variance (ANOVA) test has been performed on the obtained results of 30 runs. The results are shown in Table 7, and their corresponding rankings are shown in Figure 8. In the ANOVA test, when the obtained value for F is higher than the obtained *p*-value, the null hypothesis will be true. The obtained results verify that the obtained F-value is much higher than the obtained *p*-value, showing a significant difference among the obtained results. From Figure 8, the proposed optimization using MPA outperforms other recently developed methods that are considered in this analysis. The proposed MPA-based design provides the smallest variance range with the lowest objective value (for our minimization problem for optimization). This confirms the resilience and accuracy of proposed method. Additionally, the Tukey Honestly Significant Difference (the Tukey HSD) post hoc analysis has been provided to provide support for the obtained ANOVA test-based results. Figure 9 provides the obtained HSD post hoc results. The proposed MPA-based optimization provides the lowest value for the mean fitness function. It is followed by the MRFO and SMA optimizers. The obtained results from the two tests confirm the superior algorithm robustness of the MPA based design for determining optimal FOPID AVR controller parameters.

Table 7. Performance comparisons using ANOVA test.

ANOVA Table					
Source	SS	df	MS	F	Prob > F
Columns	0.02375	7	0.00339	66.72	4.53626×10^{-52}
Error	0.0118	232	0.00005		
Total	0.03555	239			

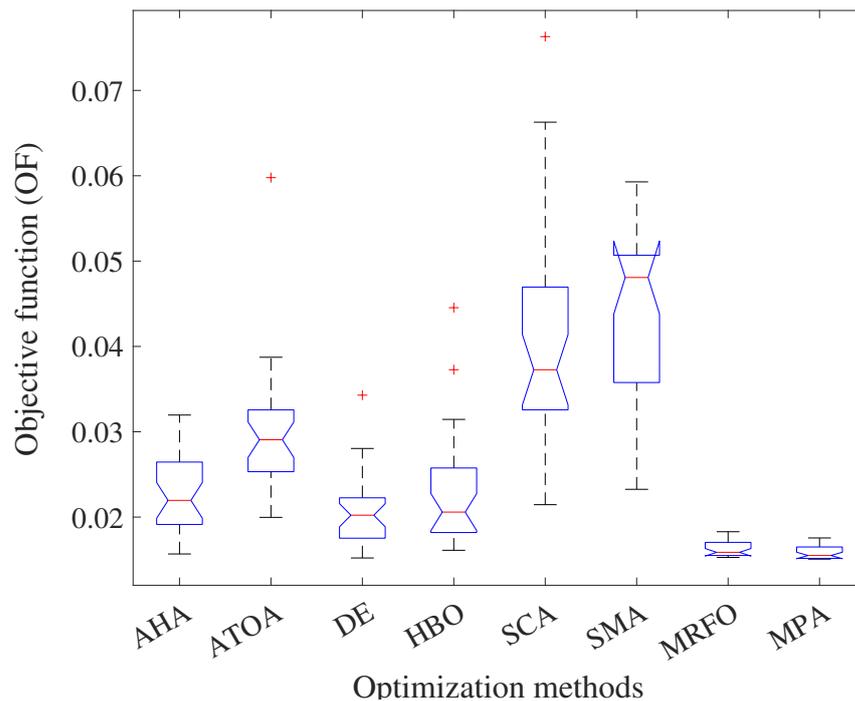


Figure 8. Statistical analysis using ANOVA test.

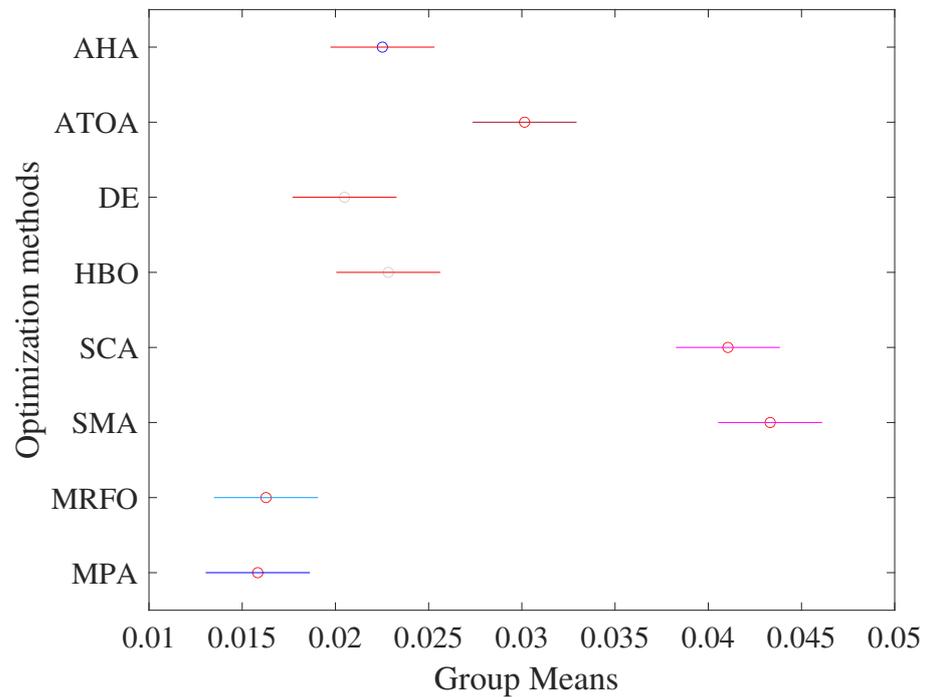


Figure 9. Tukey test of obtained results.

The obtained optimal parameters of various considered algorithms are simulated in a closed-loop control with the AVR model. The control parameters associated with the best objective function from the 30 runs are used in the simulation study for fair comparisons. The obtained results with step change in the reference voltage at zero second and full-load condition with $K_G = 1$ are shown in Figure 10. It can be seen that the worst response is obtained by the ATOA, SMA, and SCA optimizers, whereas the MPA and MRFO provide the best results. Although MPA appears to have a slightly higher peak, it provides faster, reaching a steady-state operating point. The measured ISE, IAE, ITSE, and ITAE for MPA are 2.7334, 2.857, 4.1826, and 4.3812, respectively, whereas the MPA provides 2.8728, 2.9025, 4.4870, and 4.6129, respectively.

In this part, a comparison of the step response of the proposed design is performed with some featured designs from the literature. The obtained step response comparisons are shown in Figure 11 at full-load condition with $K_G = 1$. The considered FOPID designs and parameters include method 1 as in [43], method 2 as in [44], method 3 as in [37], method 4 as in [45], method 5 as in [46], and the proposed MPA design. The proposed MPA-based FOPID AVR regulator provides reduced peak values and fast tracking in this step change scenario. Although method 2 provides a lower peak overshoot value, it has a steady state error as shown in the zoomed-in results. The proposed MPA-based design achieves zero steady state error in this scenario. The method 5 design has the highest peak overshoot value, followed by the methods 3 and 4. The design of method 1 has a high rise time and oscillations. Another case study at the no-load condition with $K_G = 0.7$ is performed and shown in Figure 12. The results confirm the superior response of the proposed method compared with studied methods from the literature at no-load condition. The proposed MPA-based design has a low peak overshoot value, a low rise time, and zero steady state error. This proves the superior performance of the proposed design method compared to existing methods in the literature.

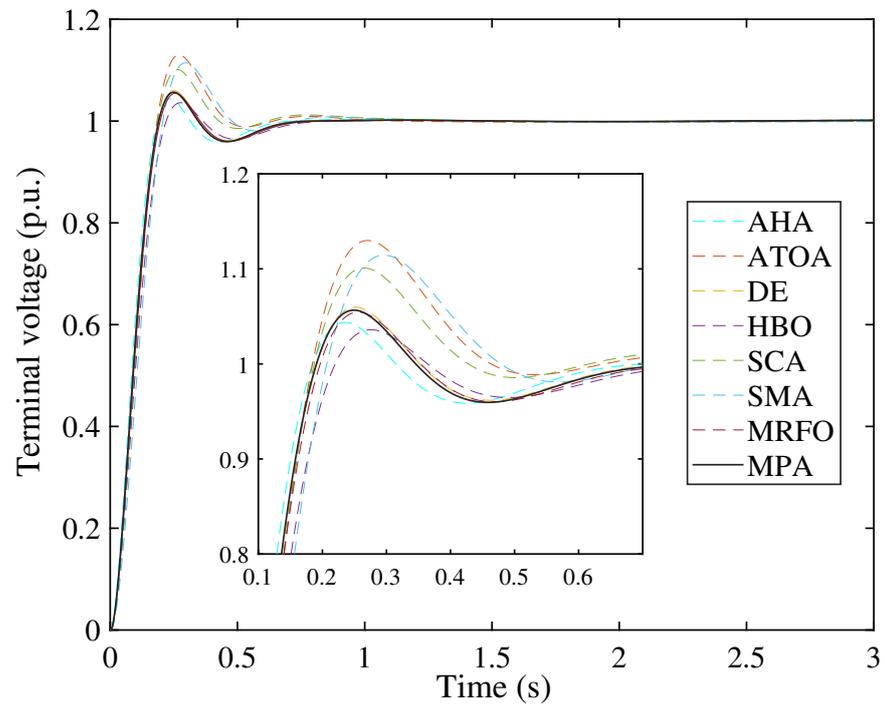


Figure 10. Step response comparisons of the various obtained optimization results at full-load condition with $K_G = 1$.

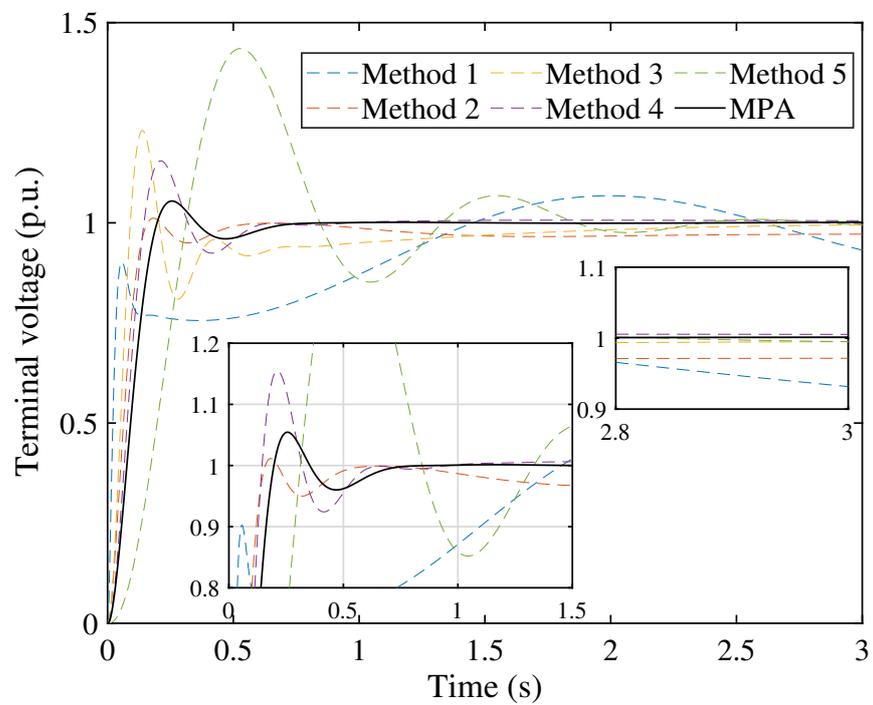


Figure 11. Step response comparisons of the proposed AVR controller with the existing designs in the literature at full-load condition with $K_G = 1$.

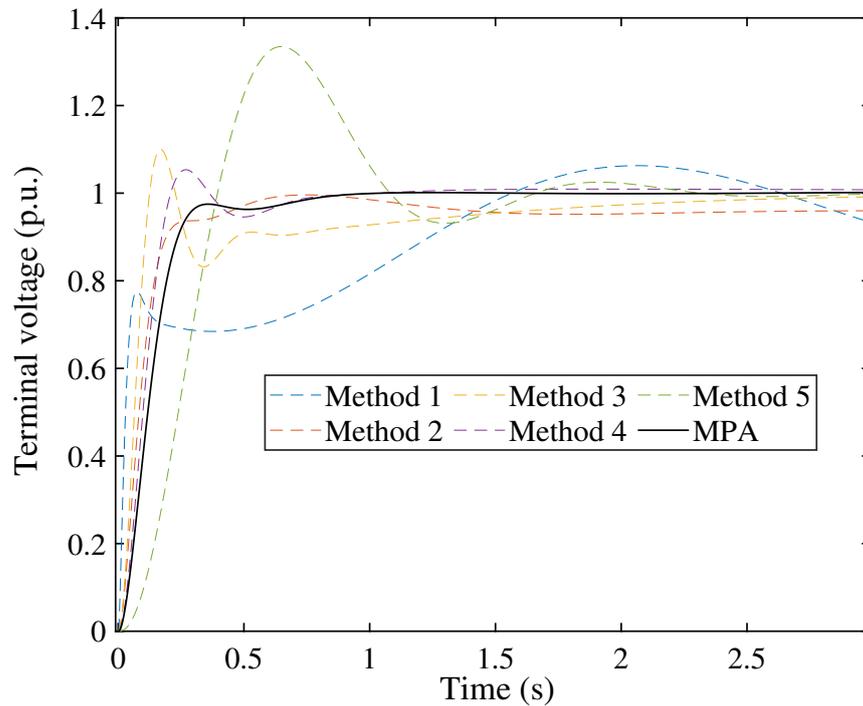


Figure 12. Step response comparisons of the proposed AVR controller with the existing designs in the literature at no-load condition with $K_G = 0.7$.

A comparison of the convergence characteristics is provided in Figure 13. The selected algorithms for the comparison include HBO, SMA, and ATOA methods. It can be seen that the MPA optimizer has good convergence speed compared with the other algorithms. While the most important criteria is the ability of MPA to have a reduced objective function at the end of the optimization process, it can be seen that MPA has the minimum objective function.

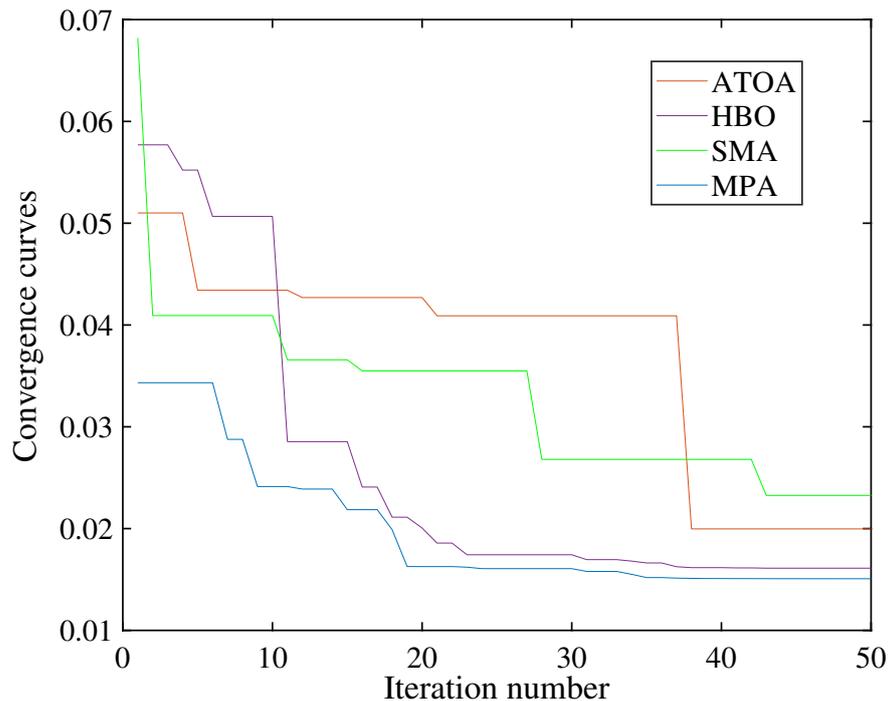


Figure 13. Conversion speed evaluation of MPA method.

Another check is made by adding a disturbance of 0.3 p.u. to the output voltage measurement. The obtained results are shown in Figure 14. The disturbance is injected at 6 s and removed at 10 s. It can be seen that the MPA-based FOPID is robust against disturbances, and it can recover its state very fast with lower fluctuations, whereas the other studied designs have higher transients and/or higher steady state error and response times. Another comparison is made with the PIDD2 tuned by the RUN algorithm, and the parameters are taken from [11]. The obtained results are shown in Figure 15. It can be seen that the proposed MPA-based FOPID has better performance and response characteristics compared with the existing PIDD2 method.

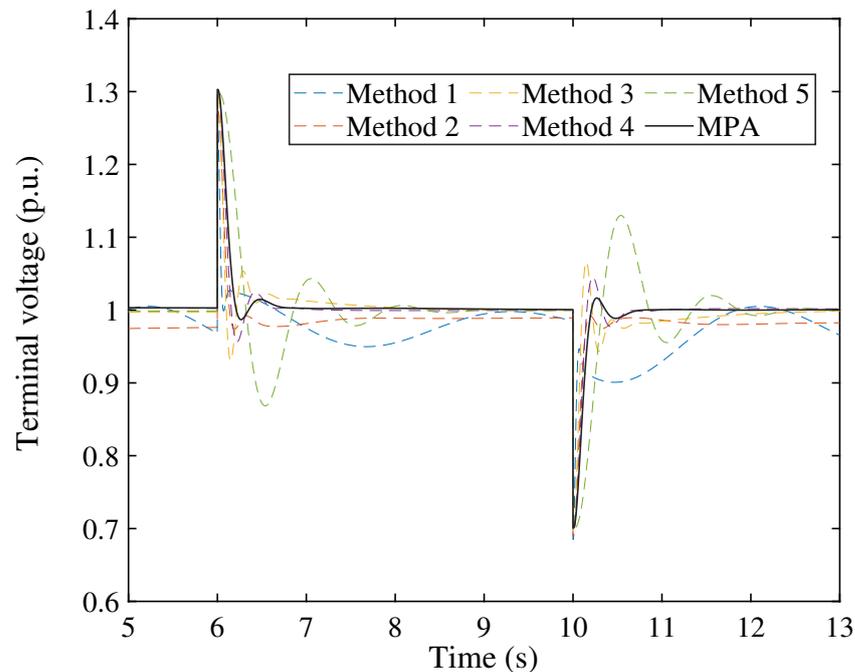


Figure 14. Step responses with load disturbance of 0.3 p.u.

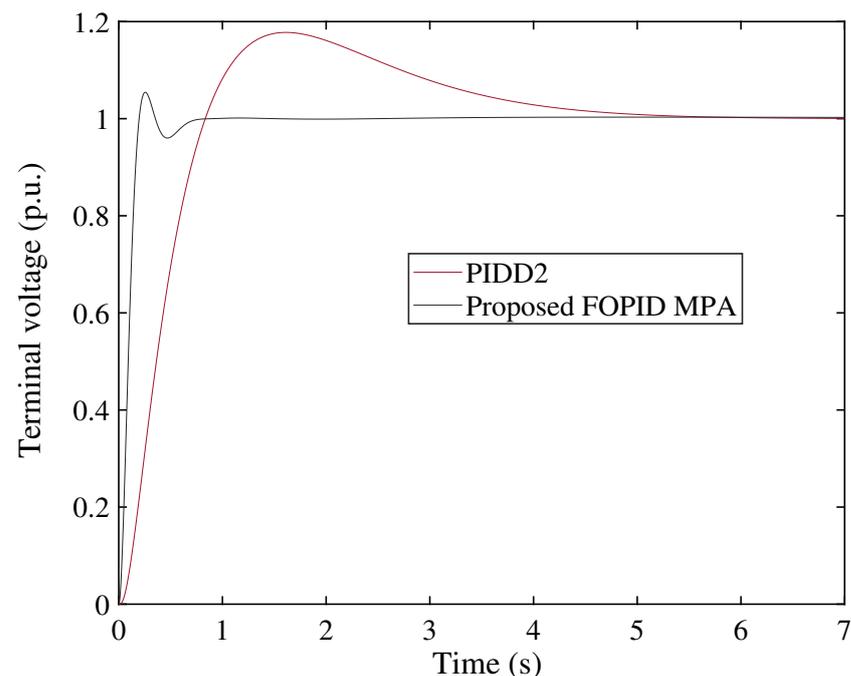


Figure 15. Performance comparison with PIDD2 with parameters from [11].

7. Conclusions

This paper deals with an important issue of modern power systems that is related to the AVR controller and its design optimization. The paper proposes an optimized FOPID AVR controller that is optimized with the powerful marine predator optimization algorithm (MPA). The proposed AVR controller achieves an improved transient response, while preserving system stability. Additionally, the FOPID provides additional design freedom due to the extra included fractional order operators. Compared with the integer-order PID, the FOPID has two additional design flexibilities, which are reflected in its response and optimization process. Comprehensive statistical analysis of the MPA and obtained results over 30 runs are provided and compared with featured methods from the literature. The proposed FOPID-MPA method provides the best objective function minimization with reduced variations with runs compared to the other studied methods. The ANOVA and Tukey tests are provided in addition to statistical parameters, which proved to be the best performance obtained by the proposed FOPID-MPA method. Future research includes consideration of various constraints of AVR elements and further comparisons with other control methods. Additionally, more considerations can be given to other measurements, such as noise and stator current limitations.

Author Contributions: A.M.N., M.A., M.H.A., S.Z.A. and A.S.A.; Methodology, A.M.N., M.A., S.Z.A. and E.A.M.; Software, M.A., M.H.A., A.S.A., S.Z.A. and E.A.M.; Validation, A.M.N.; Formal analysis, A.M.N., M.H.A., and S.Z.A.; Investigation, A.M.N., E.A.M., M.A., A.S.A. and S.Z.A.; Resources, A.M.N., M.H.A., A.S.A. and S.Z.A.; Data curation, M.A. and E.A.M.; Writing—original draft, M.H.A., M.A. and E.A.M.; Writing—review and editing, A.M.N., M.A., A.S.A., S.Z.A., M.H.A. and E.A.M.; Visualization, A.S.A. and M.H.A.; Supervision, A.S.A., S.Z.A., A.M.N. and E.A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded By Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia, Project Number IF2/PSAU/2022/01/23150.

Data Availability Statement: Not applicable.

Acknowledgments: The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number (IF2/PSAU/2022/01/23150).

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Table A1. Obtained objective functions for the MRFO and MPA over 100 runs.

Run No.	MPA	MRFO	Run No.	MPA	MRFO
1	0.0160902309153365	0.0186072735390579	51	0.0159763953627226	0.0164159935337242
2	0.0157754214386025	0.0157790581530151	52	0.0154625364235344	0.0161561738059689
3	0.0163839755300565	0.0197874911366351	53	0.0154673980808997	0.0216327342483548
4	0.0158614360293316	0.0207406428821346	54	0.0151171665779289	0.0171331501582164
5	0.0151815141506785	0.0178342633107935	55	0.0151615813344883	0.0177126323347514
6	0.0166839473808019	0.0199929483957633	56	0.0156267887327416	0.0164519137830847
7	0.0170801125246933	0.0155652253629703	57	0.0165867956883701	0.0154761204614243
8	0.0170861857896019	0.0163012176667067	58	0.0155362943324246	0.0156001843471987
9	0.0156217377981934	0.0174133359370687	59	0.0152952531265295	0.0191971202041615
10	0.0161582631746110	0.0164046438255427	60	0.0154008521226980	0.0169098714187219
11	0.0170467975859854	0.0180406862499614	61	0.0151632456741165	0.0161135094007944
12	0.0182722351800569	0.0185911187833772	62	0.0153527356134189	0.0154955036509552
13	0.0153036284462915	0.0153519464778942	63	0.0151261669951796	0.0160335763536030

Table A1. Cont.

Run No.	MPA	MRFO	Run No.	MPA	MRFO
14	0.0158050783963908	0.0153868044792599	64	0.0152063617495559	0.0207565298970893
15	0.0152747700347015	0.0158201433770507	65	0.0152006227143923	0.0155734274747462
16	0.0160724506267845	0.0187138746721933	66	0.0152126399419112	0.0169535389239530
17	0.0155061948859095	0.0186836337099605	67	0.0155041942054795	0.0153554028201223
18	0.0165419455259726	0.0185533825453101	68	0.0155097483622622	0.0160391558592351
19	0.0153239301220592	0.0223799180060417	69	0.0155229664833601	0.0161831344490288
20	0.0157248521583907	0.0158234562111405	70	0.0152269394223403	0.0171716717715401
21	0.0152544299495255	0.0156292615025043	71	0.0157237013082333	0.0182032626875426
22	0.0156251526403613	0.0160019396955112	72	0.0157714234694387	0.0185312561464443
23	0.0159162308142132	0.0178834956451730	73	0.0157236934238772	0.0160178288055126
24	0.0161622408281447	0.0166977069643665	74	0.0150855154798788	0.0156785500585492
25	0.0150770912799963	0.0160946478966843	75	0.0156749031336672	0.0183338207041531
26	0.0159260002629224	0.0202221745201756	76	0.0152678788613815	0.0177098353253446
27	0.0155029436270759	0.0178024023309439	77	0.0154648527668256	0.0213044716721203
28	0.0158256473703595	0.0159423598815060	78	0.0152115531728442	0.0175407533787249
29	0.0152981824451236	0.0154539311942310	79	0.0151919556809319	0.0154529560023485
30	0.0163552680101254	0.0168646119783646	80	0.0184879256653829	0.0160860584564143
31	0.0153077796595018	0.0166950620491384	81	0.0150963550483734	0.0161007964844298
32	0.0155507585576659	0.0163582696073985	82	0.0158997972385465	0.0169265091642214
33	0.0155108649307797	0.0155126486258078	83	0.0153172163577543	0.0158997972385465
34	0.0160034803104551	0.0183471186539829	84	0.0154046935066751	0.0174072210013845
35	0.0153675694782160	0.0231118909681678	85	0.0151392551367036	0.0194813295050308
36	0.0157011613280708	0.0162058702232047	86	0.0155098581205377	0.0159186921213136
37	0.0154641600824197	0.0159433449512397	87	0.0154122382825814	0.0166866119939868
38	0.0152287469911507	0.0158987438068760	88	0.0155100829613650	0.0161010349179289
39	0.0153853897470870	0.0157847475047855	89	0.0156897018278160	0.0166741742074250
40	0.0157703358883489	0.0173888037944703	90	0.0153128334831947	0.0159414326243861
41	0.0154313596386918	0.0170021705087658	91	0.0153575265265852	0.0168227013385350
42	0.0160520172501177	0.0157239653675728	92	0.0154502895278778	0.0164106715341243
43	0.0153801132684594	0.0185347128325756	93	0.0151150340218386	0.0160006097916894
44	0.0164614923840220	0.0153723037170996	94	0.0157887027725565	0.0155413006166845
45	0.0153702421725225	0.0153723037170996	95	0.0151212550739855	0.0155006512376765
46	0.0161266891964663	0.0153451195810031	96	0.0155500689142018	0.0161683560384650
47	0.0153867052299926	0.0172731580353522	97	0.0151789038204640	0.0178129214344732
48	0.0153635291223803	0.0154699708648064	98	0.0152734520589117	0.0154727513622186
49	0.0153284443691374	0.0163318577343206	99	0.0150910942530682	0.0174838967070398
50	0.0152902130049922	0.0156309924834781	100	0.0156344970037647	0.0179487980457389

References

- Hassan, A.; Aly, M.; Elmelegi, A.; Nasrat, L.; Watanabe, M.; Mohamed, E.A. Optimal Frequency Control of Multi-Area Hybrid Power System Using New Cascaded TID-PIAD μ N Controller Incorporating Electric Vehicles. *Fractal Fract.* **2022**, *6*, 548. [[CrossRef](#)]
- Amin, A.; Ebeed, M.; Nasrat, L.; Aly, M.; Ahmed, E.M.; Mohamed, E.A.; Alnuman, H.H.; Hamed, A.M.A.E. Techno-Economic Evaluation of Optimal Integration of PV Based DG with DSTATCOM Functionality with Solar Irradiance and Loading Variations. *Mathematics* **2022**, *10*, 2543. [[CrossRef](#)]
- Said, S.M.; Aly, M.; Hartmann, B.; Mohamed, E.A. Coordinated fuzzy logic-based virtual inertia controller and frequency relay scheme for reliable operation of low-inertia power system. *IET Renew. Power Gener.* **2021**, *15*, 1286–1300. [[CrossRef](#)]
- Alghamdi, S.; Sindi, H.F.; Rawa, M.; Alhussainy, A.A.; Calasan, M.; Micev, M.; Ali, Z.M.; Aleem, S.H.E.A. Optimal PID Controllers for AVR Systems Using Hybrid Simulated Annealing and Gorilla Troops Optimization. *Fractal Fract.* **2022**, *6*, 682. [[CrossRef](#)]
- Ghasemi, M.; Rahimnejad, A.; Gil, M.; Akbari, E.; Gadsden, S.A. A self-competitive mutation strategy for Differential Evolution algorithms with applications to Proportional–Integral–Derivative controllers and Automatic Voltage Regulator systems. *Decis. Anal. J.* **2023**, *7*, 100205. [[CrossRef](#)]
- Alilou, M.; Azami, H.; Oshnoei, A.; Mohammadi-Ivatloo, B.; Teodorescu, R. Fractional-Order Control Techniques for Renewable Energy and Energy-Storage-Integrated Power Systems: A Review. *Fractal Fract.* **2023**, *7*, 391. [[CrossRef](#)]

7. Daraz, A.; Malik, S.A.; Basit, A.; Aslam, S.; Zhang, G. Modified FOPID Controller for Frequency Regulation of a Hybrid Interconnected System of Conventional and Renewable Energy Sources. *Fractal Fract.* **2023**, *7*, 89. [[CrossRef](#)]
8. Almasoudi, F.M.; Magdy, G.; Bakeer, A.; Alatawi, K.S.S.; Rihan, M. A New Load Frequency Control Technique for Hybrid Maritime Microgrids: Sophisticated Structure of Fractional-Order PIDA Controller. *Fractal Fract.* **2023**, *7*, 435. [[CrossRef](#)]
9. Ahmed, E.M.; Mohamed, E.A.; Elmelegi, A.; Aly, M.; Elbaksawi, O. Optimum Modified Fractional Order Controller for Future Electric Vehicles and Renewable Energy-Based Interconnected Power Systems. *IEEE Access* **2021**, *9*, 29993–30010. [[CrossRef](#)]
10. Ahmed, E.M.; Selim, A.; Alnuman, H.; Alhosaini, W.; Aly, M.; Mohamed, E.A. Modified Frequency Regulator Based on TIA-TD μ FF Controller for Interconnected Microgrids with Incorporating Hybrid Renewable Energy Sources. *Mathematics* **2022**, *11*, 28. [[CrossRef](#)]
11. Izci, D.; Ekinici, S.; Mirjalili, S. Optimal PID plus second-order derivative controller design for AVR system using a modified Runge–Kutta optimizer and Bode’s ideal reference model. *Int. J. Dyn. Control.* **2022**, *11*, 1247–1264. [[CrossRef](#)]
12. Faramarzi, A.; Heidarinejad, M.; Mirjalili, S.; Gandomi, A.H. Marine Predators Algorithm: A nature-inspired metaheuristic. *Expert Syst. Appl.* **2020**, *152*, 113377. [[CrossRef](#)]
13. Sobhy, M.A.; Abdelaziz, A.Y.; Hasanien, H.M.; Ezzat, M. Marine predators algorithm for load frequency control of modern interconnected power systems including renewable energy sources and energy storage units. *Ain Shams Eng. J.* **2021**, *12*, 3843–3857. [[CrossRef](#)]
14. Aly, M.; Ahmed, E.M.; Rezk, H.; Mohamed, E.A. Marine Predators Algorithm Optimized Reduced Sensor Fuzzy-Logic Based Maximum Power Point Tracking of Fuel Cell-Battery Standalone Applications. *IEEE Access* **2021**, *9*, 27987–28000. [[CrossRef](#)]
15. Yousri, D.; Babu, T.S.; Beshr, E.; Eteiba, M.B.; Allam, D. A Robust Strategy Based on Marine Predators Algorithm for Large Scale Photovoltaic Array Reconfiguration to Mitigate the Partial Shading Effect on the Performance of PV System. *IEEE Access* **2020**, *8*, 112407–112426. [[CrossRef](#)]
16. Gaing, Z.L. A particle swarm optimization approach for optimum design of PID controller in AVR system. *IEEE Trans. Energy Convers.* **2004**, *19*, 384–391. [[CrossRef](#)]
17. Gozde, H.; Taplamacioglu, M. Comparative performance analysis of artificial bee colony algorithm for automatic voltage regulator (AVR) system. *J. Frankl. Inst.* **2011**, *348*, 1927–1946. [[CrossRef](#)]
18. GÜVENÇ, U.; YİĞİT, T.; İŞİK, A.H.; AKKAYA, İ. Performance analysis of biogeography-based optimization for automatic voltage regulator system. *Turk. J. Electr. Eng. Comput. Sci.* **2016**, *24*, 1150–1162. [[CrossRef](#)]
19. Köse, E. Optimal Control of AVR System With Tree Seed Algorithm-Based PID Controller. *IEEE Access* **2020**, *8*, 89457–89467. [[CrossRef](#)]
20. Hekimoglu, B.; Ekinici, S. Grasshopper optimization algorithm for automatic voltage regulator system. In Proceedings of the 2018 5th International Conference on Electrical and Electronic Engineering (ICEEE), IEEE, Istanbul, Turkey, 3–5 May 2018. [[CrossRef](#)]
21. Sahu, B.K.; Panda, S.; Mohanty, P.K.; Mishra, N. Robust analysis and design of PID controlled AVR system using Pattern Search algorithm. In Proceedings of the 2012 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES), Bengaluru, India, 16–19 December 2012; pp. 1–6. [[CrossRef](#)]
22. Mosaad, A.M.; Attia, M.A.; Abdelaziz, A.Y. Whale optimization algorithm to tune PID and PIDA controllers on AVR system. *Ain Shams Eng. J.* **2019**, *10*, 755–767. [[CrossRef](#)]
23. Habib, S.; Abbas, G.; Jumani, T.A.; Bhutto, A.A.; Mirsaedi, S.; Ahmed, E.M. Improved Whale Optimization Algorithm for Transient Response, Robustness, and Stability Enhancement of an Automatic Voltage Regulator System. *Energies* **2022**, *15*, 5037. [[CrossRef](#)]
24. Dogruer, T.; Can, M.S. Design and robustness analysis of fuzzy PID controller for automatic voltage regulator system using genetic algorithm. *Trans. Inst. Meas. Control.* **2022**, *44*, 1862–1873. [[CrossRef](#)]
25. Bingul, Z.; Karahan, O. A novel performance criterion approach to optimum design of PID controller using cuckoo search algorithm for AVR system. *J. Frankl. Inst.* **2018**, *355*, 5534–5559. [[CrossRef](#)]
26. Hekimoğlu, B. Sine-cosine algorithm-based optimization for automatic voltage regulator system. *Trans. Inst. Meas. Control.* **2018**, *41*, 1761–1771. [[CrossRef](#)]
27. Ekinici, S.; Hekimoğlu, B. Improved Kidney-Inspired Algorithm Approach for Tuning of PID Controller in AVR System. *IEEE Access* **2019**, *7*, 39935–39947. [[CrossRef](#)]
28. Bendjehaba, O. Continuous Firefly Algorithm for Optimal Tuning of Pid Controller in AVR System. *J. Electr. Eng.* **2014**, *65*, 44–49. [[CrossRef](#)]
29. Çelik, E.; Durgut, R. Performance enhancement of automatic voltage regulator by modified cost function and symbiotic organisms search algorithm. *Eng. Sci. Technol. Int. J.* **2018**, *21*, 1104–1111. [[CrossRef](#)]
30. Ekinici, S.; Hekimoğlu, B.; Kaya, S. Tuning of PID Controller for AVR System Using Salp Swarm Algorithm. In Proceedings of the 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), Malatya, Turkey, 28–30 September 2018; pp. 1–6. [[CrossRef](#)]
31. Anbarasi, S.; Muralidharan, S. Enhancing the Transient Performances and Stability of AVR System with BFOA Tuned PID Controller. *J. Control. Eng. Appl. Inform.* **2016**, *18*, 20–29.
32. DUMAN, S.; YÖRÜKEREN, N.; ALTAŞ, İ.H. Gravitational search algorithm for determining controller parameters in an automatic voltage regulator system. *Turk. J. Electr. Eng. Comput. Sci.* **2016**, *24*, 2387–2400. [[CrossRef](#)]

33. Pradhan, R.; Majhi, S.K.; Pati, B.B. Design of PID controller for automatic voltage regulator system using Ant Lion Optimizer. *World J. Eng.* **2018**, *15*, 373–387. [[CrossRef](#)]
34. 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 Components Syst.* **2014**, *42*, 959–969. [[CrossRef](#)]
35. Blondin, M.; Sicard, P.; Pardalos, P. Controller Tuning Approach with robustness, stability and dynamic criteria for the original AVR System. *Math. Comput. Simul.* **2019**, *163*, 168–182. [[CrossRef](#)]
36. Zamani, M.; Karimi-Ghartemani, M.; Sadati, N.; Parniani, M. Design of a fractional order PID controller for an AVR using particle swarm optimization. *Control. Eng. Pract.* **2009**, *17*, 1380–1387. [[CrossRef](#)]
37. Ortiz-Quisbert, M.E.; Duarte-Mermoud, M.A.; Milla, F.; Castro-Linares, R.; Lefranc, G. Optimal fractional order adaptive controllers for AVR applications. *Electr. Eng.* **2016**, *100*, 267–283. [[CrossRef](#)]
38. 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 Autom. Sin.* **2014**, *40*, 973–979. [[CrossRef](#)]
39. Tang, Y.; Cui, M.; Hua, C.; Li, L.; Yang, Y. Optimum design of fractional order PIAD μ controller for AVR system using chaotic ant swarm. *Expert Syst. Appl.* **2012**, *39*, 6887–6896. [[CrossRef](#)]
40. Zeng, G.Q.; Chen, J.; Dai, Y.X.; Li, L.M.; Zheng, C.W.; Chen, M.R. Design of fractional order PID controller for automatic regulator voltage system based on multi-objective extremal optimization. *Neurocomputing* **2015**, *160*, 173–184. [[CrossRef](#)]
41. Ayas, M.S.; Sahin, E. FOPID controller with fractional filter for an automatic voltage regulator. *Comput. Electr. Eng.* **2021**, *90*, 106895. [[CrossRef](#)]
42. Micev, M.; Čalasan, M.; Oliva, D. Fractional Order PID Controller Design for an AVR System Using Chaotic Yellow Saddle Goatfish Algorithm. *Mathematics* **2020**, *8*, 1182. [[CrossRef](#)]
43. Pan, I.; Das, S. Frequency domain design of fractional order PID controller for AVR system using chaotic multi-objective optimization. *Int. J. Electr. Power Energy Syst.* **2013**, *51*, 106–118. [[CrossRef](#)]
44. Sikander, A.; Thakur, P.; Bansal, R.; Rajasekar, S. A novel technique to design cuckoo search based FOPID controller for AVR in power systems. *Comput. Electr. Eng.* **2018**, *70*, 261–274. [[CrossRef](#)]
45. Khan, I.A.; Alghamdi, A.S.; Jumani, T.A.; Alamgir, A.; Awan, A.B.; Khidrani, A. Salp Swarm Optimization Algorithm-Based Fractional Order PID Controller for Dynamic Response and Stability Enhancement of an Automatic Voltage Regulator System. *Electronics* **2019**, *8*, 1472. [[CrossRef](#)]
46. Pan, I.; Das, S. Chaotic multi-objective optimization based design of fractional order PIAD μ controller in AVR system. *Int. J. Electr. Power Energy Syst.* **2012**, *43*, 393–407. [[CrossRef](#)]
47. Ahmed, E.M.; Elmelegi, A.; Shawky, A.; Aly, M.; Alhosaini, W.; Mohamed, E.A. Frequency Regulation of Electric Vehicle-Penetrated Power System Using MPA-Tuned New Combined Fractional Order Controllers. *IEEE Access* **2021**, *9*, 107548–107565. [[CrossRef](#)]
48. Bakir, H.; Guvenc, U.; Kahraman, H.T.; Duman, S. Improved Lévy flight distribution algorithm with FDB-based guiding mechanism for AVR system optimal design. *Comput. Ind. Eng.* **2022**, *168*, 108032. [[CrossRef](#)]
49. Zhao, W.; Wang, L.; Mirjalili, S. Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications. *Comput. Methods Appl. Mech. Eng.* **2022**, *388*, 114194. [[CrossRef](#)]
50. Devan, P.A.M.; Hussin, F.A.; Ibrahim, R.B.; Bingi, K.; Nagarajapandian, M.; Assaad, M. An Arithmetic-Trigonometric Optimization Algorithm with Application for Control of Real-Time Pressure Process Plant. *Sensors* **2022**, *22*, 617. [[CrossRef](#)]
51. Ahmad, M.F.; Isa, N.A.M.; Lim, W.H.; Ang, K.M. Differential evolution: A recent review based on state-of-the-art works. *Alex. Eng. J.* **2022**, *61*, 3831–3872. [[CrossRef](#)]
52. Askari, Q.; Saeed, M.; Younas, I. Heap-based optimizer inspired by corporate rank hierarchy for global optimization. *Expert Syst. Appl.* **2020**, *161*, 113702. [[CrossRef](#)]
53. Mirjalili, S. SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowl.-Based Syst.* **2016**, *96*, 120–133. [[CrossRef](#)]
54. Li, S.; Chen, H.; Wang, M.; Heidari, A.A.; Mirjalili, S. Slime mould algorithm: A new method for stochastic optimization. *Future Gener. Comput. Syst.* **2020**, *111*, 300–323. [[CrossRef](#)]
55. Zhao, W.; Zhang, Z.; Wang, L. Manta ray foraging optimization: An effective bio-inspired optimizer for engineering applications. *Eng. Appl. Artif. Intell.* **2020**, *87*, 103300. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.