Comparative Assessment of PID and ANFIS Controllers in an Automatic Voltage Regulator

Sajesan Ghimire[1*], Bhriguraj Bhattrai[1], Sulav Shrestha[1], Sagar Poudel[1]
[1]Department of Electrical Engineering, IOE, Pashchimanchal Campus
Tribhuvan University, Nepal
*sajesan.772418@pasc.tu.edu.np

(Manuscript Received 4th September, 2024; Revised 9th November, 2024; Accepted 12th November, 2024)

Abstract

This research paper provides an in-depth analysis of the performance characteristics of PID (Proportional-Integral-Derivative) and ANFIS (Adaptive Neuro-Fuzzy Inference System) controllers within Automatic Voltage Regulator (AVR) systems. The primary objective is to evaluate these controllers' behavior and efficacy, potentially extending their application to other control systems in the power sector. Utilizing the robust capabilities of MATLAB-SIMULINK, the PID controller was finely tuned, while the ANFIS controller was trained using carefully selected data. The findings highlight the ANFIS controller's exceptional performance, characterized by a notably fast settling time of 1.7277 seconds and 1.8716% overshoot. In comparison, the PID controller exhibited greater overshoot and a longer settling time, demonstrating less efficiency. These results were compared with other published research papers, further validating the superior performance of the ANFIS controller. This detailed evaluation confirms the ANFIS controller's superiority, offering significant guidance for researchers and industry professionals in making informed decisions regarding the optimal choice of controllers for various control systems.
Keywords: ANFIS; AVR; Control; Fuzzy; MATLAB Simulink; PID; Power System.

1. Introduction

The variation in reactive load can cause fluctuations in the terminal voltage of a generator or an electrical system. These changes can lead to problems like system instability, equipment failure, and lower efficiency. To manage these fluctuations, Automatic Voltage Regulators (AVRs) are employed. AVRs automatically adjust the excitation of generators to maintain stable voltage levels (Ali et al., 2021). They continuously monitor the generator's output voltage and adjust the excitation current as needed either increasing it to raise the voltage or decreasing it to lower the voltage. This automatic adjustment ensures that voltage remains constant despite changes in load, stabilizing the power system and ensuring reliable and efficient operation.

A new control method has been introduced to improve the performance and efficiency of Automatic Voltage Regulators (AVRs). The Proportional-Integral-Derivative (PID) controller is widely used due to its simplicity and ease of use (Izci & Ekinci, 2022). Several advanced methods have been explored to enhance PID controllers, including PID with additional controls, whale-optimized PID, sigmoid PID (Suid & Ahmad, 2022), fuzzy PID with genetic algorithms (Suid & Ahmad, 2022), teaching-learning based optimized (TLBO) PID (Chatterjee & Mukherjee, 2016), fractional order PID (FOPID) (Zamani et al., 2009), (Li et al., 2017) and higher order differential feedback (Sikander et al., 2018). However, many of these methods struggle with providing the dynamic performance needed for complex systems (Izci et al., 2021), (Izci et al., 2022).

Researchers have also identified robustness and resilience as key factors for effective voltage regulation. One study focused on improving PID controller stability for

AVRs by calculating and analyzing stability intervals using specific criteria. Although this method helps, its analytical approach can slow down response times and limit adaptability in complex situations (Ali et al., 2021).

Another study proposed using fuzzy logic control to enhance AVR performance in synchronous motors. This approach involves a three-step process: setting rules to correct voltage errors, adjusting the field output, and programming the fuzzy logic system (Cuesta Cuesta & Santa, 2021).

Additionally, research explored using an improved Particle Swarm Optimization (PSO) approach with a unique sliding surface function to control AVRs reliably. This method aims to reduce chattering (unwanted oscillations) by using a special function, but it does not consider other strategies that might better minimize chattering, which can impact system performance (Furat & Cucu, 2022). As the techniques discussed in previous section didn’t provide enough adaptability and robustness. When fuzzy logic and ANN are combined together. It is supposed to perform better than either system alone because fuzzy logic uses an imprecise spectrum of data to generate inferential rules that quickly produce an array of accurate responses or conclusions, and ANN extracts the necessary information from the AVR system. Furthermore, the combined feature will increase the system's capacity for rapid learning, improve adaptability, and capture the non-linear structure of a process (Sahu et al., 2020). Hence, an adaptive controller is required in the closed loop for improved responsiveness. Thus, this work presents an adaptive neuro-fuzzy inference system (ANFIS), which combines the use of neural networks with fuzzy inference, emphasizing the neural network's capacity to learn and derive a set of rules that are necessary to design a desirable fuzzy inference system (FIS) for the efficient control of the AVR system.

Apart from the previously mentioned explanations, the study also makes contributions in terms of designing an appropriate method for getting the training dataset ready, using a hybrid learning training algorithm, and performing a comparative analysis of the controllers' performance characteristics for the AVR system. As a result, the study evaluates the effectiveness of the PID controller, the suggested ANFIS, and the system operating without a controller. This paper presents a unique use of the hybrid learning method for the AVR system with ANFIS. As a result, the procedure produces results that are comparatively better than those of other methods.

2. Mathematical Modeling of System

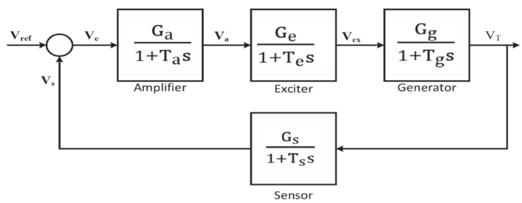

Fig. 1: Block Diagram of AVR without controller

The AVR system can typically be divided into four components: the excitation unit, which supplies direct current to the generator's field windings; the amplifier unit, which amplifies the error signal to a usable level for the excitation unit; the generator and rectifier; and the sensor unit. These components are connected as illustrated in Fig. 1.

2.1 Sensor Unit

This unit takes the terminal voltage V_T as input and produces a sensor voltage V_s. This sensor voltage is then compared with the reference voltage V_{ref} to generate an error signal V_e, which serves as the input for the amplifier unit. Thus, the sensor unit can be represented as (Mosaad et al., 2019), (Melendez-Perez et al., 2020), (Celik & Durgut, 2018)

$$V_S(s) = \frac{G_S}{1+T_S s} V_T(s) \tag{1}$$

Where G_s represents the sensor gain and T_s is the time constant, usually between 0.001 and 0.06 seconds. The associated transfer function is:

$$\frac{V_S(s)}{V_T(s)} = \frac{G_S}{1+T_S s} \tag{2}$$

Note that : $V_e = V_{ref} - V_s$

2.2 Amplifier unit

The amplifier model in the AVR system generally takes the error signal V_e as input and produces an amplified voltage signal V_a, which then serves as the input to the exciter unit. Mathematically, as (Mosaad et al., 2019), (Melendez-Perez et al., 2020), (Celik & Durgut, 2018)

$$V_a(s) = \frac{G_a}{1+T_a s} V_e(s) \quad (3)$$

G_a and T_a represent the amplifier gain and time constant, respectively, with values ranging from 10 to 40 and 0.02 seconds to 0.1 seconds. Thus, the transfer function is:

$$\frac{V_a(s)}{V_e(s)} = \frac{G_a}{1+T_a s} \quad (4)$$

2.3 Exciter unit

In this unit, V_a is fed into the exciter. The exciter output, V_{ex}, is then supplied to the generator's field winding and is sufficient to overcome the winding resistance. The generic equivalent model of this unit is provided in as (Mosaad et al., 2019), (Melendez-Perez et al., 2020), (Celik & Durgut, 2018)

$$V_{ex}(s) = \frac{G_e}{1+T_e s} V_e(s) \quad (5)$$

Here, G_e and T_e denote the exciter gain and time constant, with values ranging from 0.8 to 1 and 0.02 seconds to 0.1 seconds, respectively. Equation (6) represents the first-order transfer function derived from Eqn. (5).

$$\frac{V_{ex}(s)}{V_e(s)} = \frac{G_e}{1+T_e s} \quad (6)$$

2.4 Generator unit

The generator output, V_T, is influenced by the voltage supplied by the exciter. The first-order model of the generator's transfer function is provided in as (Mosaad et al., 2019), (Melendez-Perez et al., 2020), (Celik & Durgut, 2018)

$$V_T(s) = \frac{G_g}{1+T_g s} \quad (7)$$

Here, G_g and T_g represent the generator gain and time constant.

Resolving Equations (2), (4), and (6) provides the overall transfer function for the three units in the feedforward loop: the amplifier, the exciter, and the generator.

$$V_T = \left(\frac{G_a}{1+T_a s} \times \frac{G_e}{1+T_e s} \times \frac{G_g}{1+T_g s}\right)(V_{ref} - V_s) = \frac{G_a G_e G_g}{(1+T_a s)\times(1+T_e s)\times(1+T_g s)}(V_e) \quad (8)$$

$$\text{Thus, } \frac{V_T(s)}{V_e(s)} = \frac{G_a G_e G_g}{(1+T_a s)\times(1+T_e s)\times(1+T_g s)} \quad (9)$$

Hence combining (9) and the feedback path represented by the sensor unit as shown in Fig.1 will give (10), which is the transfer function of the entire AVR system without a controller,

$$TF_{AVR} = \frac{G_a G_e G_g (1+T_s s)}{(1+T_a s)(1+T_e s)(1+T_g s)(1+T_s s)+G_a G_e G_g G_s} \quad (10)$$

For this work, the values of the parameters used are shown in Table 1 .

Table 1 AVR Parameters used (Mosaad et al., 2019).

Parameters	Gain	Time Constant
Sensor	$G_s = 1$	$T_s = 0.05$
Amplifier	$G_a = 10$	$T_a = 0.1$
Exciter	$G_e = 1$	$T_e = 0.4$
Generator	$G_g = 1$	$T_g = 1$

3. Modelling the AVR System Using PID and Aneis Controllers

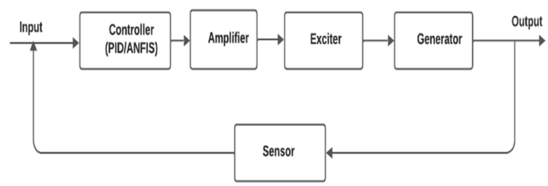

Fig. 2: Diagram of AVR System with Controller

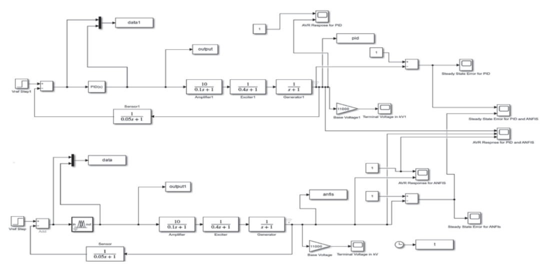

Fig. 3: Simulink Diagram of PID and ANFIS Controller

In the Automatic Voltage Regulator (AVR) model, two types of controllers PID and ANFIS are used one after the other The basic model was the PID controller. The values of P, I, and D were taken by the PID tuning method. Since the ANFIS controller needs both input and output data for its training, this data is taken from the PID controller. A 'To-Workspace' block in Simulink is used to capture this data, with the 'Save format' set to 'Array' and 'Sample time' set to -1. A MUX block is then used to combine the input and output data. After the ANFIS controller is trained, it is implemented in the model using a FUZZY block in Simulink (Mosaad et al., 2019).

3.1 PID Controller

A Proportional-Integral-Derivative (PID) controller is a crucial control mechanism used for system stabilization and regulation. It calculates the control signal by evaluating the difference between the desired set point and the current process value. The controller consists of three key components:

Proportional (P) Term: Produces an output that is directly proportional to the current error, helping to reduce steady-state error and enhance responsiveness.

Integral (I) Term: Integrates the error over time to eliminate steady-state error and correct long-term biases.

Derivative (D) Term: Addresses the rate of change of the error, aiding in system stabilization and reducing abrupt or oscillatory fluctuations.

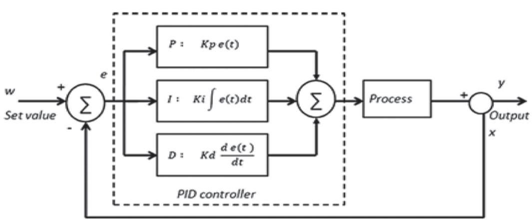

Fig. 4: Block Diagram of PID controller

In this study, the initial setup of the PID controller in Simulink involves using the parallel form of the PID. The time domain is set to 'Continuous Time,' and the 'Internal' option is selected for the source. The 'Filter coefficient' (N) is set to 100.The compensator is used. The value of P, I, and D for this experiment is taken by PID tuning.

3.2 PID Tuning

PID tuning involves selecting the appropriate gains K_p (proportional), K_i (integral), and K_d (derivative) for a PID controller to achieve the desired control performance. This process can be carried out through manual adjustment, using techniques like Ziegler-Nichols or Cohen-Coon, or by employing frequency response methods or model-based approaches. Proper tuning is essential to ensure system stability and optimal response. After several times of tuning the best value obtained as Fig. 6.

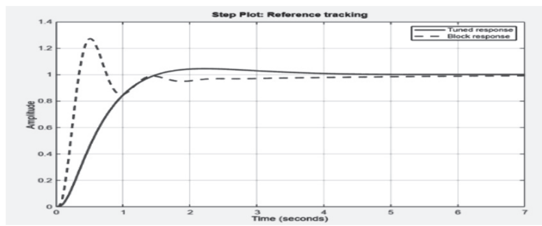

Fig. 5: PID tuning process

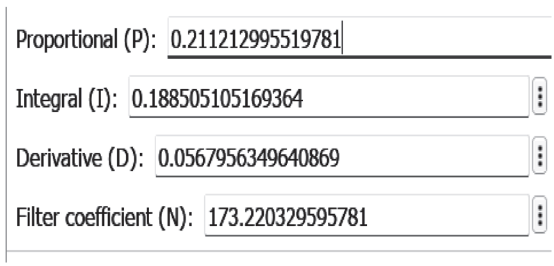

Fig. 6: Tuned value of PID Controller

3.3 ANFIS Controller

ANFIS (Adaptive Neuro-Fuzzy Inference System) combines the strengths of neural networks and fuzzy logic to effectively represent complex and nonlinear systems with minimal training data. It integrates the learning capabilities of Artificial Neural Networks (ANNs) with the ability of fuzzy logic to handle uncertainty and imprecision. This fusion allows ANFIS to approximate pure nonlinear mathematical models efficiently, leveraging ANN's learning from processes and fuzzy logic's control of ambiguous data for optimal performance.

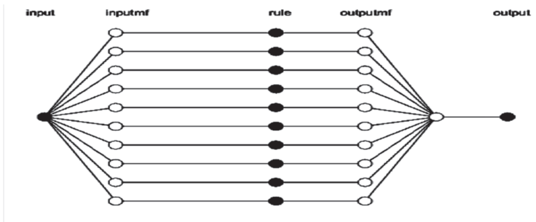

Fig. 7: The Structural Design of ANFIS

ANFIS consists of five interconnected layers, with each layer's outputs being weighted inputs to the subsequent layer. Initially, data from the input layer is transformed into membership functions, which quantify the degree of membership for each input. In the second layer, fuzzy rules link the input to the output. The third layer normalizes the data, which is then passed to the fourth layer, where the output membership function is provided. Finally, the fifth layer combines these outputs to generate a single result.

For the training process, ten membership functions of type 'trimf' (triangular membership function) and constant are used. The 'Hybrid' method is chosen for testing, with 100 epochs. The input and output training data are sourced from the PID controller.

3.4 Matlab Simulation

The AVR system simulation model is developed using various toolboxes available in MATLAB SIMULINK.

Step 1: Designing the AVR System with a PID Controller

Initially, the AVR system model is constructed using a PID controller. The PID tuner is employed to determine the appropriate values for P, I, and D to achieve the desired output. The 'To Workspace' block is used to log output data to the command window.

Step 2: Data Generation for ANFIS Training

For this study, data is collected from the PID controller to train both ANN and ANFIS. The 'To Workspace' block is added to the PID controller's input and output sections to obtain the necessary training data.

Step 3: ANFIS Training

ANFIS training requires input and output data in a combined format. The MATLAB command ‘anfisedit’ opens the training window, where basic parameters, such as 100 iterations, are set. Successful ANFIS training is indicated when the error approaches zero.

Step 4: Comparing the Performance of PID and ANFIS Controllers

Once training is complete, the results from all three models are brought together on a single platform. A SCOPE block is added to measure and compare the output signals.

4. Result and Discussion

This section entails the results obtained from the design and simulation of the Automatic Voltage Regulator with the PID and the ANFIS controllers

4.1 Comparative analysis of AVR System with PID and ANFIS Controllers

This section presents a comparative analysis of the Automatic Voltage Regulator (AVR) system's performance under PID (Proportional-Integral-Derivative) and ANFIS (Adaptive Neuro-Fuzzy Inference System) controllers. Figures 8 and 9 illustrate the system’s terminal voltage response and steady-state error when a unit step input is applied for each controller.

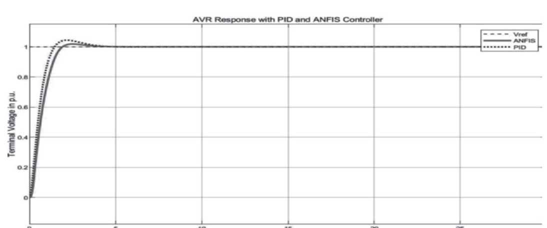

Fig 8 : Steady-state error of different setups

(Unit step response)

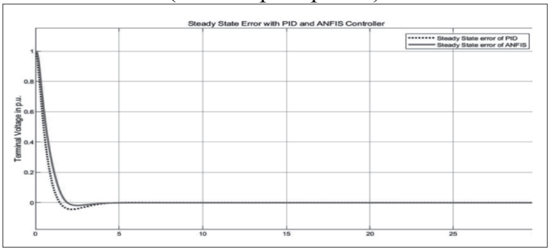

Fig 9 : Terminal voltage in per unit with PID and ANFIS controller

In Figure 8, the response of the AVR system to a unit step input shows the behavior of the terminal voltage in reaching the reference voltage (Vref). Under PID control, the voltage initially rises rapidly with minimal overshoot, stabilizing at the desired voltage. This behavior demonstrates the PID controller's ability to achieve a stable output with a minimal transient response.

In contrast, the AVR system under ANFIS control also rises to the reference voltage but does so with even faster response time and reduced overshoot compared to the PID controller. The ANFIS controller's adaptive nature enables it to handle transient behavior more efficiently, reaching the desired voltage with smoother transitions and achieving a stable output more quickly.

Figure 9 illustrates the steady-state error, which reflects the difference between the terminal voltage and the reference voltage over time. Both controllers effectively minimize the steady-state error to zero. However, the ANFIS controller achieves this reduction more quickly, as seen by its error curve reaching zero faster than that of the PID controller. This demonstrates ANFIS's enhanced capability to reduce residual deviations and maintain the desired voltage level efficiently.

The table 2 summarizes the key performance parameters of the AVR system under both PID and ANFIS control:

Table 2 Responses to unit step input.

Parameters	AVR with PID	AVR with ANFIS
Rise Time (s)	0.9198	1.1493
Settling Time (s)	3.2804	1.7277
Overshoot (%)	4.4286	1.8716
Undershoot (%)	0	0
Peak(p.u)	1.0443	1.0187
Peak time (s)	2.1585	2.4871
Transient time (s)	3.2804	1.7277
Steady State Error	0.0037	0.0012

4.2 Notable Findings

a. Settling Time: The ANFIS controller reduced the settling time by 47.34%, leading to quicker system stability compared to the PID controller.

b. Overshoot: The overshoot decreased by 57.74% with the ANFIS controller, enhancing precision and reducing the risk of instability.

c. Transient Time: The transient time showed a 47.34% improvement with the ANFIS controller, indicating a faster response to changes or disturbances.

d. Peak Value: The peak value with the ANFIS controller was closer to the reference value, showing a 2.45% improvement over the PID controller.

Using comparable parameters, this study's performance was assessed against other published works, with the results summarized in Table 3. The data in Tables 2 and 3 indicate that the ANFIS controller setup achieved superior performance, particularly in terms of response and rise time, making it the preferred choice.

Table 3 Comparison with other Published Works

Control Method	Settling Time (s)	Overshoot (%)
FPID [15]	3.6817	1
MPC-PSO [16]	30.1	0
ANN-PID [12]	3.69	7.75
FOPID [7]	52	0
ANFIS	1.7277	1.8716

5. Conclusion

In conclusion, this research has demonstrated the clear advantages of the Adaptive Neuro-Fuzzy Inference System (ANFIS) over the conventional PID controller in the context of Automatic Voltage Regulator (AVR) systems. The ANFIS controller significantly outperformed the PID controller, achieving a faster settling time of 1.7277 seconds compared to the PID's 3.516 seconds, and a reduced overshoot of 1.8716% against the PID's 7.0817%, along with a minimal steady-state error. These results highlight the efficiency of ANFIS in managing the nonlinearities and uncertainties within the AVR system, owing to its hybrid nature that combines fuzzy logic and neural networks. The findings not only establish the ANFIS controller as a superior alternative for AVR systems but also suggest its potential for broader applications in other control systems. Future research could focus on integrating advanced optimization techniques with ANFIS and testing its performance in real-time environments to further validate its practical utility.

References

Ali, M., Soliman, M., Mahmoud, K., Guerrero, J., Lehtonen, M., & Darwish, M. M. F. (2021). Resilient design of robust multi-objectives PID controllers for automatic voltage regulators: D-Decomposition approach. IEEE Access, 1(PP), 1–11. https://doi.org/10.1109/ACCESS.2021.3100415

Mosaad, A. M., Attia, M. A., & Abdelaziz, A. Y. (2019). Whale optimization algorithm to tune PID and PIDA controllers on AVR system. Ain Shams Engineering Journal, 10(4), 755–767. https://doi.org/10.1016/j.asej.2019.07.004

Cuesta Cuesta, D. N., & Santa, F. M. (2021). Optimization of an automatic voltage regulator (AVR) on a synchronous machine using fuzzy control. Journal of Physics: Conference Series, 2135(1), 012004. https://doi.org/10.1088/1742-6596/2135/1/012004

Izci, D., & Ekinci, S. (2022). An improved RUN optimizer based real PID plus second-order derivative controller design as a novel method to enhance transient response and robustness of an automatic voltage regulator. e-Prime - Advances in Electrical Engineering, Electronics, and Energy, 2(September), 100071. https://doi.org/10.1016/j.prime.2022.100071

Chatterjee, S., & Mukherjee, V. (2016). PID controller for automatic voltage regulator using teaching–learning based optimization technique. International Journal of Electrical Power & Energy Systems, 77, 418–429. https://doi.org/10.1016/j.ijepes.2015.11.010

Suid, M. H., & Ahmad, M. A. (2022). Optimal tuning of sigmoid PID controller using nonlinear sine cosine algorithm for the automatic voltage regulator system. ISA Transactions, 128, 265–286. https://doi.org/10.1016/j.isatra.2021.11.037

Zamani, M., Karimi-Ghartemani, M., Sadati, N., & Parniani, M. (2009). Design of a fractional order PID controller for an AVR using particle swarm optimization. Control Engineering Practice, 17(12), 1380–1387. https://doi.org/10.1016/j.conengprac.2009.07.005

Li, X., Wang, Y., Li, N., Han, M., Tang, Y., & Liu, F. (2017). Optimal fractional order PID controller design for automatic voltage regulator system based on reference model using particle swarm optimization. International Journal of Machine Learning and Cybernetics, 8(5), 1595–1605. https://doi.org/10.1007/s13042-016-0530-2

Sikander, A., Thakur, P., Bansal, R. C., &

Rajasekar, S. (2018). A novel technique to design cuckoo search based FOPID controller for AVR in power systems. Computers & Electrical Engineering, 70, 261–274. https://doi.org/10.1016/j.compeleceng.2017.07.005

Izci, D., Ekinci, S., Zeynelgil, H. L., & Hedley, J. (2021). Fractional order PID design based on novel improved slime mould algorithm. Electrical Power Components and Systems, 49(9–10), 901–918. https://doi.org/10.1080/15325008.2022.2049650

Izci, D., Ekinci, S., & Hekimoğlu, B. (2022). A novel modified Lévy flight distribution algorithm to tune proportional, integral, derivative, and acceleration controller on buck converter system. Transactions in Instrumentation and Measurement, 44(2), 393–409. https://doi.org/10.1177/01423312211036591

Chadar, G., Mohaney, S. K., & Lakra, P. (2022). Improvement in voltage profile of synchronous generator using PID controller and artificial neural network in automatic voltage regulator. In Proceedings of the 2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT) (pp. 89–94). https://doi.org/10.1109/CSNT54456.2022.9787625

Furat, M., & Cucu, G. G. (2022). Design, implementation, and optimization of sliding mode controller for automatic voltage regulator system. IEEE Access, 10, 55650–55674. https://doi.org/10.1109/ACCESS.2022.3177621

Sahu, J., Satapathy, P., Debnath, M. K., Mohanty, P. K., Sahu, B. K., & Padhi, J. R. (2020). Automatic voltage regulator design based on fractional calculus plus PID controller. In 2020 International Conference on Computational Intelligence for Smart Power System and Sustainable Energy (CISPSSE) (pp. 1–4). https://doi.org/10.1109/CISPSSE49931.2020.9212263

Melendez-Perez, R., Ortiz-Rodriguez, F., & Ruiz-Vega, D. (2020). Design of an ANFIS automatic voltage regulator of a synchronous generator. In 2020 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC) (pp. 1–6). https://doi.org/10.1109/ROPEC50909.2020.9258731

Kumar, V., & Sharma, V. (2020). Automatic voltage regulator with particle swarm optimized model predictive control strategy. In 2020 First IEEE International Conference on Measurement, Instrumentation, Control and Automation (ICMICA) (pp. 1–5). https://doi.org/10.1109/ICMICA48462.2020.9242783

Celik, E., & Durgut, R. (2018). Performance enhancement of automatic voltage regulator by modified cost function and symbiotic organisms search algorithm. Engineering Science and Technology, an International Journal, 21(5), 1104–1111.https://doi.org/10.1016/j.jestch.2018.08.006