Vokasi Unesa Bulletin of Engineering, Technology and Applied Science (VUBETA)
https://journal.unesa.ac.id/index.php/vubeta
Vol. 2, No. 2, 2025, pp. 270~280 DOI: 10.26740/vubeta.v2i2.36821
ISSN: 3064-0768

PID controller tuning for an AVR system using Particle Swarm Optimisation and Genetic Algorithm; A comparison-based approach

Sabo Aliyu[1], Mahmud Bawa[2*], Yunusa Yakubu[3] Alan Audu Ngyarmunta[4], Yunusa Aliyu[5], Alama Musa[6], Mohammed Katun[7].
[1,2,3,4,5,6,7] Department of Electrical Engineering, Faculty of Engineering Technology, Nigerian Defense Academy, Kaduna Nigeria

Article Info

Article history:

Received December 13, 2024
Revised February 28, 2025
Accepted May 15, 2025

Keywords:

Proportional Integral Derivative (PID)
Automatic Voltage Control
Particle Swarm Optimization (PSO)
Genetic algorithm
Adaptive control

ABSTRACT

This paper discusses tuning a Proportional-Integral-Derivative (PID) controller for an Automatic Voltage Regulator (AVR) system utilizing a particle swarm optimization technique and genetic algorithm. The primary objective is to compare the two methods. The AVR system was modeled and simulated using MATLAB, and the performance of the optimized PID controller was analyzed. The results demonstrate significant improvements in system performance with the metaheuristic-tuned PID controllers. Specifically, the GA-tuned PID controller achieved the best overshoot reduction (0.8%) and steady-state error minimization (0.0005), making it highly suitable for applications requiring precise voltage control. On the other hand, the PSO-tuned PID controller excelled in reducing settling time (2.7 seconds) and improving rise time (1.2 seconds), making it ideal for systems requiring rapid stabilization. Both metaheuristic approaches showed substantial enhancements. The study highlights the importance of selecting the appropriate optimization technique based on specific system requirements, whether the priority is minimizing overshoot, reducing settling time, or achieving near-zero steady-state error.

1. INTRODUCTION

The stability and efficiency of electrical power systems are critical for modern society, and Automatic Voltage Regulators (AVRs) play a vital role in maintaining voltage stability in synchronous generators. However, the performance of AVR systems is often compromised by challenges such as non-linear dynamics, external disturbances, and time delays, which can lead to voltage fluctuations, instability, and poor transient response [1][2]. Traditional Proportional-Integral-Derivative (PID) controllers, widely used in AVR systems, are typically tuned using manual methods like the Ziegler-Nichols approach [3]. These methods, while simple, often result in suboptimal performance, characterized by excessive overshoot, slow settling times, and significant steady-state errors. As power systems become more complex and the demand for precise voltage control increases, there is a pressing need for more advanced and automated tuning methods to address these limitations and improve the overall performance of AVR systems [4].

Recent advancements in control systems have seen the application of metaheuristic optimization techniques for tuning PID controllers [5]. Studies such as those by Nayak and Singh (2015) and Govindan (2020) have demonstrated the effectiveness of Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) in optimizing PID parameters for various control systems. For instance, Adel and Abdelkader (2020) applied PSO to tune PID controllers for non-linear systems, achieving significant improvements in transient response. Similarly, Jayachitra and Vinodha (2014) used GA to optimize PID controllers for industrial processes, showcasing its ability to handle complex, non-linear systems. However, while these studies highlight the potential of metaheuristic techniques, there is limited research comparing the performance of PSO and GA specifically for AVR systems. Furthermore, most existing studies focus on single-objective optimization, neglecting the need for a balanced approach that simultaneously considers multiple performance metrics such as overshoot, settling time, and steady-state error. This study aims to fill this gap by comprehensively

*Corresponding Author
Email: mbawa695@gmail.com

comparing PSO and GA for PID tuning in AVR systems, focusing on their ability to optimize multiple performance criteria.

This study contributes to the existing body of knowledge in several ways.

1) It compares two prominent metaheuristic optimization techniques, PSO and GA, for tuning PID controllers in AVR systems, highlighting their strengths and weaknesses.

2) It introduces a multi-objective optimization approach that minimizes overshoot, settling time, and steady-state error, offering a more balanced and practical solution for real-world AVR systems.

3) The study demonstrates the superiority of metaheuristic-tuned PID controllers over traditional manual tuning methods, providing empirical evidence of their effectiveness.

4) The findings of this study have practical implications for the design and implementation of AVR systems, offering engineers and researchers a more efficient and automated approach to PID tuning that can enhance the stability and performance of power systems.

The remainder of this paper is organized as follows. Section 2 describes the components of an AVR system and the mathematical model, which is presented as a transfer function. Section 3 explains the concept of PID control, detailing how proportional (Kp), integral (Ki), and derivative (Kd) gains influence system behavior. Sections 4 and 5 introduce PSO and GA as metaheuristic techniques. Section 6 presents the performance comparison of PSO and GA-tuned PID controllers based on key metrics such as overshoot, settling time, steady-state error, and rise time. Finally, Section 7 concludes the paper by summarizing the key findings and suggesting directions for future research.

2. METHOD

2.1 The AVR System Model

The AVR system is designed to regulate the voltage output of a synchronous generator by controlling the excitation voltage [6]. The system consists of four main components:

I. Amplifier: Amplifies the control signal.

II. Exciter: Provides the necessary field current to the generator.

III. Generator: A generator converts mechanical energy into electrical energy.

IV. Sensor: Measures the output voltage and provides feedback to the controller.

An Automatic Voltage Regulator (AVR) is a device that keeps the output voltage of an alternator or synchronous generator steady despite changes in operating conditions or load. Initially, the AVR uses a sensor, typically a voltage transformer circuit, to measure the output voltage. This estimated value is then compared to a reference voltage representing the desired output level. Any discrepancy between the reference and measured voltages produces an error signal, which the system then amplifies. The electrical current that flows through the generator's field winding, known as the excitation current, is managed by this amplified error signal. A magnetic field is produced by the rotor's field winding [7]. The stator's side experiences induced voltage due to this magnetic field's interaction with the stator windings. Ultimately, the generator's output voltage is the induced voltage in the stator windings. However, the AVR system could find it difficult to react appropriately to changes in input voltage, load demand, or other outside disruptions. Without dynamic adjustment capabilities and real-time feedback, the system might operate inefficiently or with voltage variations. Furthermore, operators may need to manually recalibrate the AVR system or adjust its settings, which could cause delays and potential power supply interruptions [8].

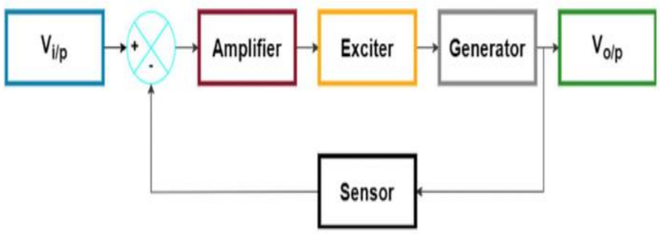

Figure 1. The block diagram of an AVR system without a controller

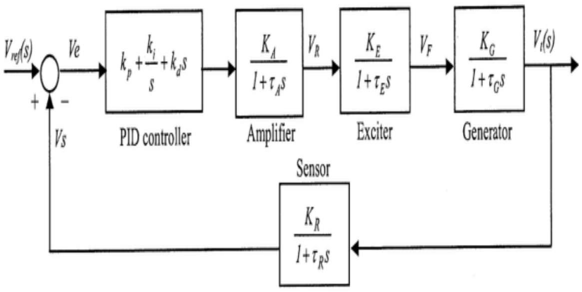

Figure 2. The Simulink-based block diagram of an AVR system with a PID controller

The transfer function of the AVR system can be represented as:

$$G(s) = \frac{K}{(1+TAs)(1+TEs)(1+TGs)(1+Ts)}$$

Where:
K is the system gain,
TA represents the time constants of the amplifier, TE represents the time constants of the exciter, TG represents the time constants of the generator, and Ts represents the time constants of the sensor. This transfer function serves as the foundation for developing the control system [9][10]

2.2 The PID Controller in AVR

Over the years, researchers have developed several traditional and reliable control algorithms for load frequency management applications, including PID control, intelligent control, adaptive control, reliable control, and MPC control[6], [9]. To achieve the intended control result, industries typically utilize a sort of feedback control system called a proportional, integral, and derivative (PID) controller [11][12]. PID controllers are utilized in around 90% of industrial loops because of their straightforward design, ease of usage, resilience, and limited number of tuning parameters. A well-thought-out PID controller can guarantee the stable operation of the plant and smooth process operation [13][14]. The difference between the reference signal and the actual output determines the proportional control signals they provide [15][16]. PID controllers are often designed to provide stability, reference tracking, and disturbance rejection, all requirements for the steady response domain [17]. Figure 3 below displays a simplified block schematic of a plant managed by a PID controller [18].

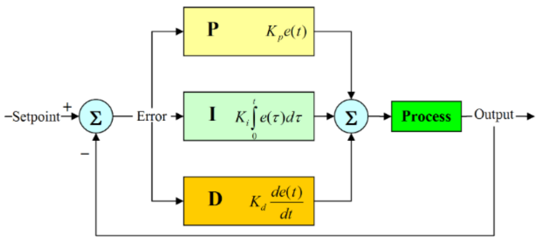

Figure 3. PID controller block diagram

Conventional PID controllers often have difficulty managing non-linear, higher-order linear, and systems with time delays [19]. Over the years, numerous methods for tuning PID controllers have been proposed. For instance, Ziegler and Nichols introduced techniques based on frequency and time domains. However, with advancements in computer technology and Artificial Intelligence (AI), intelligent algorithms for optimal PID tuning have emerged, including Genetic Algorithms and Particle Swarm Optimization. Standard error metrics used to optimize PID parameters include Integral Square Error (ISE), Integral Absolute Error (IAE), and Integrated Time Absolute Error (ITAE) [19]. These methods often involve fitting the process's frequency response to a specific second-order plus dead time model, which can represent both monotonic and oscillatory process behaviors [20].

The PID controller provides the control input based on the error between the desired and actual voltage. The control law is given by:

$$u(t) = Kpe(t) + Ki \int e(t)dt + Kd\frac{de(t)}{dt} \quad (1)$$

where:
u(t) is the control signal,
e(t) is the error between the reference and output voltage; Kp, Ki, and Kd are the proportional, integral, and derivative gains, respectively.
Tuning these parameters is crucial to achieving desired system performance characteristics like minimal overshoot, fast settling time, and low steady-state error.

2.3 The Particle Swarm Optimization (PSO)

Mr. James Kennedy and Russell C. Eberhart developed the particle swarm optimization algorithm in 1995 [21][22]. PSO is a metaheuristic optimization technique inspired by the social behavior of birds flocking or fish schooling [23][24]. Each particle represents a potential solution (in this case, a set of PID parameters). The particles explore the search space by updating their velocities and positions based on their best-known positions and the best global position discovered by the swarm [21][22][25].

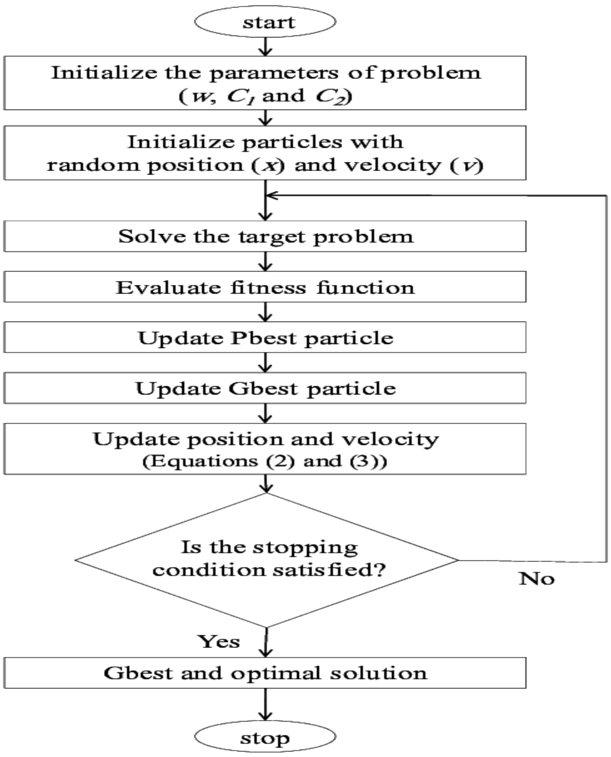

Figure 4. Particle Swarm Optimization algorithm flowchart

The position update rules for each particle **i** are given by:

$$vi(t+1) = wvi(t) + c1r1[pbest,i - xi(t)] + c2r2[gbest - xi(t)] \quad (2)$$

$$xi(t+1) = xi(t) + vi(t+1) \quad (3)$$

Here:

- vi(t) is the velocity of particle i,
- xi(t) is the position of particle i,
- pbest i is the best position found by particle i,
- gbest is the global best position found by the swarm,
- w is the inertia weight,
- c1 and c2 are acceleration coefficients,
- r1 and r2 are random values between 0 and 1.

The optimization process uses the Integral of Time-weighted Absolute Error (ITAE) as its objective function.

$$ITAE = \int_0^{\infty} t|e(t)|dt \quad (4)$$

PSO minimizes this objective function to find the optimal Kp, Ki, and Kd values.

2.4 Genetic Algorithm (GA) Technique

Genetic Algorithms (GA) are a type of random search method used for solving non-linear systems of equations and optimizing complex problems [26]. GA employs probabilistic transition rules rather than deterministic ones and operates on a population of potential solutions, individuals, or chromosomes, which evolve over iterations [27][28]. Each iteration of the algorithm is called a generation. The algorithm simulates the evolution of solutions using a fitness function along with genetic operators such as reproduction, crossover, and mutation[29]-[31]. As shown in Figure 4, a Genetic Algorithm typically begins with a randomly initialized population. This population, or mating pool, is often represented by a real-valued number or a binary string known as a chromosome [32]-[34]. The performance of each individual is measured and evaluated by an objective function, which assigns a corresponding number to each, termed its fitness [28][35][36].

The system evaluates the fitness of each chromosome and implements the survival of the fittest strategy. This study determines the fitness of each chromosome based on the error value. A genetic algorithm has three primary operations: reproduction, crossover, and mutation [37]-[39]. Figure 4 illustrates the sequence of operations involved in the GA.

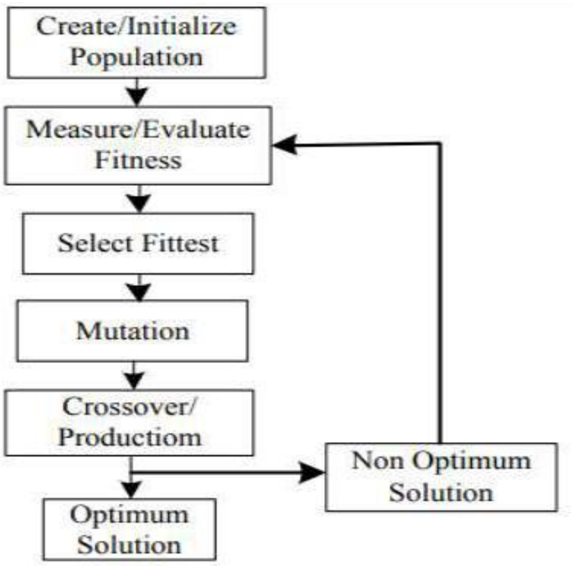

Figure 5. The flow chart of GA

Step 1. Initialize the parameters with a population of random solutions, including crossover rate, mutation rate, number of clusters, and number of generations. Determine the coding mode.
Step 2. Compute and evaluate the fitness function's value.
Step 3. Perform crossover and mutation operations to form a new cluster.
Step 4. Continue repeating Step 2 until the process achieves the best value [36].

Table 1. System Parameters

Parameter	**Gain**	**Time constant**
Amplifier gain	KA=10	τA=0.1
Exciter	$KE=1$	τE=0.4
Generator	KG=1	τG=1.0
Sensor	$KR=1$	τR=0.05

Substituting the system parameters in the AVR equation;

$$KG(s)H(s) = \frac{K_A}{(1+TAs)(1+TEs)(1+TGs)(1+Ts)} \quad (5)$$

The open loop transfer function of the system is

$$KG(s)H(s) = \frac{K_A}{(1+0.1s)(1+0.4s)(1+s)(1+0.05s)} \quad (6)$$

$$KG(s)H(s) = \frac{500K_A}{(s+10)(s+2.5)(s+1)(s+20)} \quad (7)$$

$$KG(s)H(s) = \frac{500K_A}{s^4+33.5s^3+307.5s^2+775s+500} \quad (8)$$

The characteristics polynomial equation is thus

$$s^4 + 33.5s^3 + 307.5s^2 + 775s + 500 + 500K_A = 0 \quad (9)$$

Therefore, the closed-loop transfer function of the system is

$$\frac{V_t(s)}{V_{ref}(s)} = \frac{25K_A(s+20)}{s^4+33.5s^3+307.5s^2+775s+500+500K_A} \quad (10)$$

While the steady-state response is

$$Vt_{ss} = \lim s\, V_t(s) = \frac{K_A}{1+K_A} = \frac{10}{1+10} = 0.909 \quad (11)$$

The steady-state error is

$$Ve_{ss} = 1.0 - 0.909 = 0.091 \quad (12)$$

To reduce the steady state error and increase the system stability, the PID controller with transfer function:

$$G_c(s) = K_p + \frac{K_I}{s} + K_D s \quad (13)$$

Is thus introduced;
Thus, the block diagram of the PID-compensated AVR system will be the result.

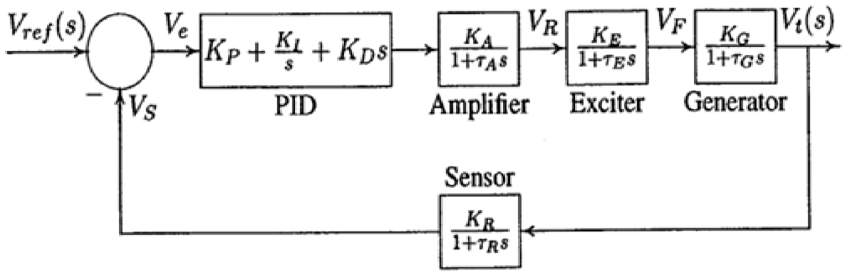

Figure 6. The block diagram of PID compensated AVR system

The simulation block diagram is thus.

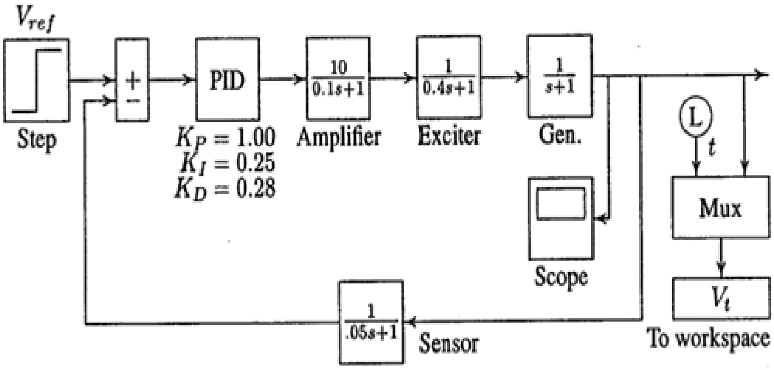

Figure 7. Simulated block diagram of PID compensated AVR system

The PID controller (Proportional-Integral-Derivative controller) was tuned using the Particle Swarm Optimization (PSO) technique to ensure voltage stability of the system, reduce steady-state error, minimize overshoot, and decrease settling time.

3. RESULTS AND DISCUSSIONS

Table 2. PSO and GA Parameters

Parameter	**Particle Swarm Optimization (PSO**	**Genetic Algorithm (GA)**
Population Size	50 particles	50 individuals
Number of Iterations	100 iterations	100 generations
Inertia Weight (w)	0.9 (initial), linearly decreasing to 0.4	N/A
Acceleration Coefficients (c1, c2)	c1 = 2.0, c2 = 2.0	N/A
Crossover Rate	N/A	0.8 (80%)
Mutation Rate	N/A	0.1 (10%)
Selection Method	N/A	Roulette Wheel Selection
Crossover Method	N/A	Single-point crossover
Mutation Method	N/A	Gaussian mutation
Search Space	Kp: [0, 10], Ki: [0, 10], Kd: [0, 10]	Kp: [0, 10], Ki: [0, 10], Kd: [0, 10]
Fitness Function	ITAE	ITAE
Stopping Criteria	Maximum iterations or convergence	Maximum generations or convergence

The system's performance was assessed using key metrics such as overshoot, settling time, and steady-state error. The results are summarized in Table 3 below.

Table 3. Performance Metrics Comparison

Controller Type	Overshoot (%)	Settling Time (s)	Steady-State Error	Rise Time (s)
PSO-Tuned PID	4.2	2.7	0.001	1.2
GA-Tuned PID	0.8	3.2	0.0005	1.4

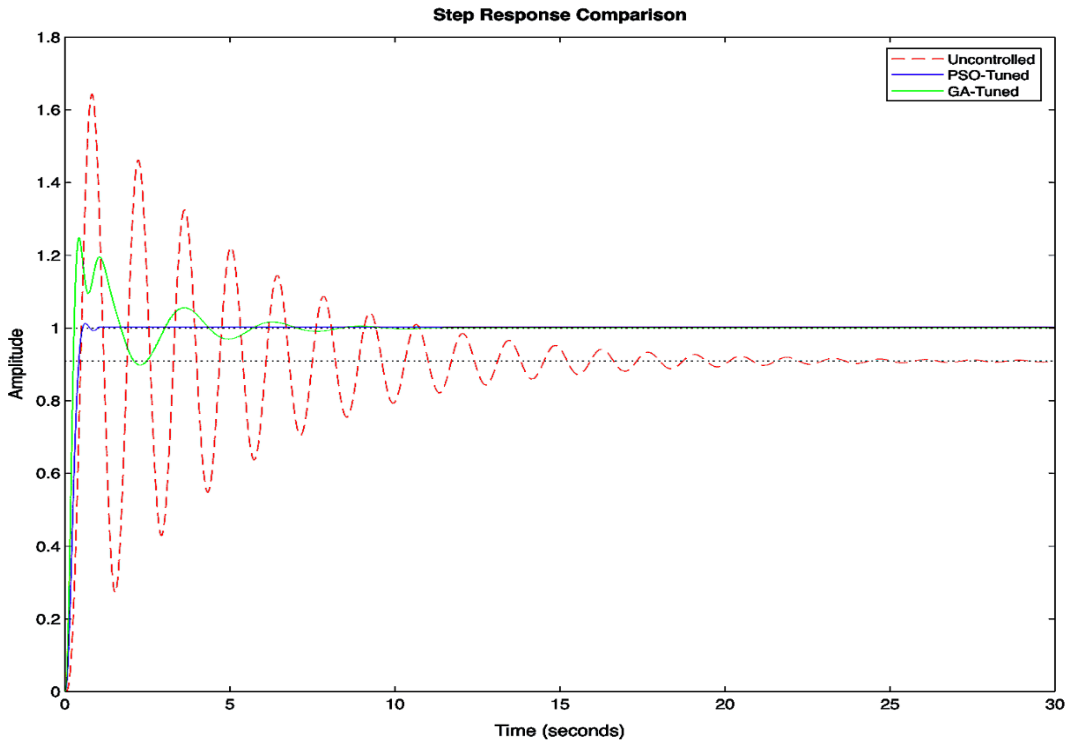

Figure 8. Step responses for Uncontrolled, PSO, and GA-tuned Systems

Table 4. Gains Values for GA and PSO

Algorithm	Kp	Ki	Kd
GA	0.92	0.65	0.08
PSO	1.25	0.28	0.18

4. CONCLUSION

This study has demonstrated the effectiveness of Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) in tuning PID controllers for Automatic Voltage Regulator (AVR) systems, significantly enhancing their dynamic performance. Traditional tuning methods, such as Ziegler-Nichols, often fail to deliver optimal results due to their heuristic nature and lack of adaptability to system variations. In contrast, metaheuristic techniques like PSO and GA offer superior performance by systematically optimizing PID parameters based on well-defined objective functions, such as the Integral of Time-weighted Absolute Error (ITAE). The comparative analysis revealed that PSO excels in achieving a faster response, with a settling time of 2.7 seconds and a rise time of 1.2 seconds, making it particularly well-suited for applications that demand rapid stabilization.

However, PSO exhibited a higher overshoot (4.2%) than GA. On the other hand, GA demonstrated superior accuracy by minimizing overshoot to just 0.8% and reducing steady-state error to 0.0005, making it the preferred choice for applications where precision and stability are paramount. This trade-off between response speed and accuracy suggests that the selection of an optimization method should be application-specific, with PSO being ideal for time-sensitive systems and GA for highly stable voltage regulation.

Moreover, the findings underscore the potential benefits of hybrid optimization techniques that combine the strengths of both PSO and GA. A hybrid PSO-GA approach could leverage the fast convergence of PSO and the high accuracy of GA to deliver an optimal balance between response speed and precision. Future research should explore hybrid models and validate their effectiveness through real-time hardware implementations. Additionally, investigating alternative metaheuristic techniques, such as Artificial Bee Colony (ABC) and Differential Evolution (DE), could provide further improvements in AVR system performance. In conclusion, this study highlights the advantages of using metaheuristic optimization for PID tuning in AVR systems. Engineers can achieve improved voltage regulation, enhanced transient response, and greater overall system stability by selecting the appropriate optimization method based on system requirements. The results of this research provide a strong foundation for further advancements in intelligent control strategies, paving the way for more efficient and reliable power system operations.

REFERENCES:

[1] C. Kaya, Z. H. Kilimci, M. Uysal, and M. Kaya, “The Metaheuristic Optimization Techniques in Text Classification,” *International Journal of Computational and Experimental Science and Engineering*, vol. 10, no. 2, pp. 126–132, 2024. https://doi.org/10.22399/ijcesen.295

[2] S. Dahal, G. J. Hegglid, J. K. Nøland, B. B. Chhetri, S. Mishra, and T. Øyvang, “Integrating Multiple Slack Bus Operations and Metaheuristic Techniques for Power Flow Optimization,” *Scientific Reports*, vol. 15, 2024. https://doi.org/10.21203/rs.3.rs-5360850/v1

[3] A. Nayak and M. Singh, “Study of Tuning of PID Controller by Using Particle Swarm Optimization,” *International Journal of Advanced Engineering Research and Studies*, 2015.

[4] A. H. M. S. Ula and A. R. Hasan, “Design and Implementation of a Personal Computer Based Automatic Voltage Regulator for a Synchronous Generator,” *IEEE Transactions on Energy Conversion*, vol. 7, no. 1, pp. 125–131, 1992. https://doi.org/10.1109/60.124551.

[5] T. Nagunwa, “Comparative Analysis of Nature-Inspired Metaheuristic Techniques for Optimizing Phishing Website Detection,” *Analytics*, vol. 3, no. 3, pp. 344–367, 2024. https://doi.org/10.3390/analytics3030019.

[6] P. Govindan, “Evolutionary Algorithms-Based Tuning of PID controller for an AVR system,” *International Journal of Electrical and Computer Engineering*, vol. 10, no. 3, pp. 3047–3056, 2020. https://doi.org/10.11591/ijece.v10i3.pp3047-3056.

[7] O. F. Nami *et al.*, “Performance Comparison of PID, FOPID, and NN-PID Controller for AUV Steering Problem,” *Jurnal Elektronika dan Telekomunikasi*, vol. 24, no. 1, p. 72, Aug. 2024. https://doi.org/10.55981/jet.596.

[8] O. M. Hesham, M. A. Attia, and S. F. Mekhamer, “Enhancement of AVR system performance by using Hybrid Harmony Search and Dwarf Mongoose Optimization Algorithms,” *Science Report*, vol. 14, no. 1, Dec. 2024. https://doi.org/10.1038/s41598-024-77120-3.

[9] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol.4, pp. 1942-1948, 1995. https://doi.org/10.1109/ICNN.1995.488968.

[10] V. Chopra, S.K Singla, and L. Dewan, “Comparative Analysis of Tuning a PID Controller Using Intelligent Methods”, *Acta Polytechnica Hungarica*, vol. 11, pp. 235-249, 2014.

[11] A. Jayachitra and R. Vinodha, “Genetic Algorithm Based PID Controller Tuning Approach for Continuous Stirred Tank Reactor,” *Advances in Artificial Intelligence*, vol. 2014, pp. 1–8, Dec. 2014. https://doi.org/10.1155/2014/791230.

[12] P. Govindan, “Evolutionary Algorithms-based Tuning of PID Controller for an AVR System,” *International Journal of Electrical and Computer Engineering*, vol. 10, no. 3, pp. 3047–3056, 2020. https://doi.org/10.11591/ijece.v10i3.pp3047-3056.

[13] T. A. Faris, K. Yusoff, M. Farid Atan, N. A. Rahman, S. F. Salleh, and N. A. Wahab, “Optimization of PID Tuning Using Genetic Algorithm,” *Journal of Applied Science & Process Engineering*, vol. 2, no. 2, 2015. https://doi.org/10.33736/jaspe.168.2015.

[14] V. Chopra, S.K Singla, and L. Dewan, “Comparative Analysis of Tuning a PID Controller Using Intelligent Methods”, *Acta Polytechnica Hungarica*, vol. 11, pp. 235-249, 2014.

[15] J. C. Basilio and S. R. Matos, “Design of PI and PID Controllers with Transient Performance Specification,” *IEEE Transactions on Education*, vol. 45, no. 4, pp. 364–370, 2002. https://doi.org/10.1109/TE.2002.804399.

[16] O. M. Hesham, M. A. Attia, and S. F. Mekhamer, “Enhancement of AVR System Performance by using Hybrid Harmony Search and Dwarf Mongoose Optimization Algorithms,” *Science Report*, vol. 14, no. 1, Dec. 2024. https://doi.org/10.1038/s41598-024-77120-3.

[17] M. Abachizadeh, M. R. H. Yazdi and A. Yousefi-Koma, "Optimal Tuning of PID Controllers using Artificial Bee Colony Algorithm," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 379-384, 2020. https://doi.org/10.1109/AIM.2010.5695861.

[18] M. Rayid Hasan Mojumder and N. Kumar Roy, "Review of Meta-Heuristic Optimization Algorithms to Tune The PID Controller Parameters for Automatic Voltage Regulator," *arXiv*, 2024. https://doi.org/10.48550/arXiv.2409.00538.

[19] K. S. Tang, Kim Fung Man, Guanrong Chen and S. Kwong, "An Optimal Fuzzy PID controller," *IEEE Transactions on Industrial Electronics*, vol. 48, no. 4, pp. 757-765, Aug. 2001. https://doi.org/10.1109/41.937407.

[20] Qing-Guo Wang, Tong-Heng Lee, Ho-Wang Fung, Qiang Bi and Yu Zhang, "PID tuning for improved performance," *IEEE Transactions on Control Systems Technology*, vol. 7, no. 4, pp. 457-465, July 1999. https://doi.org/10.1109/41.937407.

[21] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol.4, pp. 1942-1948, 1995. https://doi.org/10.1109/ICNN.1995.488968.

[22] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 1995. https://doi.org/10.1109/ICNN.1995.488968.

[23] A. Jones, "Particle Swarm Optimization," 2023. https://doi.org/10.13140/RG.2.2.32162.20163.

[24] S. Liu et al., "An Improved Particle Swarm Optimization Method for Nonlinear Optimization," *International Journal of Intelligent Systems*, vol. 2024, no. 1, 2024. https://doi.org/10.1155/2024/6628110.

[25] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, pp. 39-43, 1995. https://doi.org/10.1109/ICNN.1995.488968.

[26] C. R. Reeves, C.C Wright, "Genetic Algorithms and the Design of Experiments*". The IMA Volumes in Mathematics and its Applications Springer*, vol 111. Springer, 1999. https://doi.org/10.1007/978-1-4612-1542-4_12

[27] A. Jayachitra and R. Vinodha, "Genetic Algorithm Based PID Controller Tuning Approach for Continuous Stirred Tank Reactor," *Advances in Artificial Intelligence*, vol. 2014, pp. 1–8, 2014. https://doi.org/10.1155/2014/791230

[28] S. S. Patil, U. B. Deshannavar, S. N. Gadekar-Shinde, A. H. Gadagi, and S. A. Kadapure, "Optimization Studies on Batch Extraction of Phenolic Compounds from Azadirachta Indica using Genetic Algorithm and Machine Learning Techniques," *Heliyon*, vol. 9, no. 11, Nov. 2023. https://doi.org/10.1016/j.heliyon.2023.e21991

[29] T. A. Faris, K. Yusoff, M. Farid Atan, N. A. Rahman, S. F. Salleh, and N. A. Wahab, "Optimization of PID Tuning Using Genetic Algorithm," *Journal of Applied Science & Process Engineering*, vol. 2, no. 2, 2015. https://doi.org/10.33736/jaspe.168.2015

[30] P. Aivaliotis-Apostolopoulos and D. Loukidis, "Swarming Genetic Algorithm: A Nested Fully Coupled Hybrid of Genetic Algorithm and Particle Swarm Optimization," *PLoS One*, vol. 17, no. 9, 2022. https://doi.org/10.1371/journal.pone.0275094

[31] R. De Figueiredo, B. Toso, and J. Schmith, "Auto-Tuning PID Controller Based on Genetic Algorithm," *IntechOpen*, 2023. https://doi.org/10.5772/intechopen.110143

[32] A. H. Fadhil and J. R. Lina, "Particle Swarm Optimization and Genetic Algorithm for Tuning PID Controller of Synchronous Generator AVR System," *Engineering and Technology Journal*, vol. 29, no. 16, 2011.

[33] A. Hasibuan, F. Akbar, S. Meliala, Rosdiana, R. Putri, and I. M. Ari Nrartha, "Control and Monitoring of 1 Phase Generator Automatic Voltage Regulator Internet of Things," *International Journal of Electrical and Computer Engineering*, vol. 14, no. 6, pp. 7158–7168, Dec. 2024. https://doi.org/10.11591/ijece.v14i6.pp7158-7168.

[34] T. A. Faris, K. Yusoff, M. Farid Atan, N. A. Rahman, S. F. Salleh, and N. A. Wahab, "Optimization of PID Tuning Using Genetic Algorithm," *Journal of Applied Science & Process Engineering*, vol. 2, no. 2, 2015. https://doi.org/10.33736/jaspe.168.2015.

[35] M. Abdurrahman, M. A. Omer Nourain, M. O. Abdulsalam Muhieeddeen, O. H. ALnour Mahjoub, and M. A. Hassan Ahmed, "Optimizing the PID Controller by Using the Genetic Algorithm," *FES Journal of Engineering Sciences*, vol. 11, no. 2, pp. 1–7, 2022. https://doi.org/10.52981/fjes.v11i2.1929.

[36] G. Renner and A. Ekart, "Genetic Algorithms in Computer-Aided Design," *Computer-Aided Design*, vol. 35, no. 8, pp. 709-726, 2003. https://doi.org/10.1016/S0010-4485(03)00003-4.

[37] A. Kumar and R. Gupta, "Tuning of PID Controller Using PSO Algorithm and Compare Results of Integral Errors for AVR System," *International Journal Corner*, 2013.

[38] M. A. Ibrahim, A. K. Mahmood, and N. S. Sultan, "Optimal PID Controller of a Brushless DC motor using Genetic Algorithm," *International Journal of Power Electronics and Drive Systems*, vol. 10, no. 2, pp. 822–830, Jun. 2019. https://doi.org/10.11591/ijpeds.v10.i2.pp822-830.

[39] A. Jayachitra and R. Vinodha, "Genetic Algorithm Based PID Controller Tuning Approach for Continuous Stirred Tank Reactor," *Advances in Artificial Intelligence*, vol. 2014, pp. 1–8, Dec. 2014. https://doi.org/10.1155/2014/791230.

.