SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Dynamic Particle Swarm Optimization for PID Controller Design of AVR Systems

Manajer Kumar Rana[1]*, Raghav Prasad Parouha[2]

[1]Department of Mathematics, Indira Gandhi National Tribal University, Amarkantak, 484887, Madhya Pradesh, India.
E- mail: ressch.manajer.rana@igntu.ac.in
[2] Department of Mathematics, Indira Gandhi National Tribal University, Amarkantak, 484887, Madhya Pradesh, India.
E- mail: rparouha@igntu.ac.in

Abstract
Particle swarm optimization (PSO) is a robust technique in optimization for challenging tasks. Several state-of-the-art PSO variants have emerged in the last two decades. Despite their potential performance, the premature convergence issue is still alive due to insufficient population diversity in the later phase. Consequently, performance is degraded and misses the opportunity of a promising solution. Therefore, to rectify premature convergence issues and build overall optimization performance, Dynamic PSO (DPSO) is introduced in this work for optimal tuning of proportional-integral-derivative (PID) controller parameters in an automatic voltage regulator (AVR) system. Introduced DPSO is based on dynamic parameter adjustment in the velocity update equation, as well as the trajectory modulation factor (TMF) in the position update equation. DPSO consist of (i) constant inertia weight, to reduce redundant momentum, (ii) acceleration coefficients (time-varying), to endorse better transition from exploration to exploitation, and (iii) TMF, to dynamically shrink the particle trajectory and attain proper diversity. The optimal values of the PID controller parameters are evaluated using DPSO according to the objective function ITAE. Compared results illustrate that DPSO delivered better control outcomes for the PID controller parameter than competing approaches in measures of settling time, overshoot, and statistical performance.

Keywords: Metaheuristic Algorithms, Particle Swarm Optimization, AVR System, PID-Controller.

1. Introduction

Automatic voltage regulators (AVRs) act as a key factor in the modern power system. As it manages the terminal voltage across a synchronous generator within the desired limits despite load changes dynamically and parameters uncertainties [1]. Stable voltage regulation is crucial for enhancing voltage stability in power systems and avoiding an undesirable operating environment [2]. Therefore, researchers and engineers are giving more attention to designing effective controller strategies for the AVR system. Even though numerous controller strategies have been reported in the literature. However, PID controllers are the most widely used controllers in industrial and power system applications. Because of its simple framework and principles, ease of understanding and implementation, and reliable control performance. A generic PID controller consists of three gain parameters and the appropriate selection of these parameters is crucial to achieving satisfactory performance. Namely, these parameters are: "proportional gain (K_P), integral gain (K_I), and derivative gain (K_D)". Inappropriate choices of these gain parameters may lead to excessive overshoot, higher settling time, slower rise time, larger error, and reduced system stability. The canonical PID tuning techniques often fail to offer satisfactory performance for highly nonlinear and complicated systems. These techniques include Cohen Coon (CC), Ziegler–Nichols, and trial-and-error [3], [4]. To address such inconvenience, researchers deployed several optimizations techniques for PID parameter tuning. Currently, employed metaheuristic algorithms (MAs) are PSO [5], biogeography-based optimization (BBO) [6], artificial bee colony (ABC) [7], novel improved slime mould algorithm (SMA) [8], and arithmetic optimization algorithm (AOA) [9]. All these optimizers are employed to deliver optimal tuning of the PID controller within an AVR system. They encounter, despite their successful performance, some shortcomings, including premature convergence, minimal local stagnation, and challenges in selecting the controlling parameter [10]. Such shortcomings may be unable to determine the best values for the PID controller coefficients, consequently causing undesired overshoots and oscillatory responses in the system. Therefore, exploring novel robust MAs remains an active challenge research area. This work introduced a novel PSO variant, dynamic PSO (DPSO), for optimal tuning of PID controller parameters in an AVR system.

The main contribution of the current study is:

- A dynamic PSO (DPSO) algorithm is developed that includes constant inertia weight, time-varying acceleration coefficient, and an increasing-decreasing tendency trajectory modulation factor (TMF).
- The developed DPSO applied to PID controller for the AVR system.

This article is structured in several folds: Sec. 2 discusses the methodology, which includes the simple PSO and the proposed DPSO algorithm. Sec. 3 outlines the AVR mathematical model and PID tuning expression. Experimental results and comparative analysis are offered in Sec. 4. Sec. 5 concludes the study with future research ideas.

2. Methodology

This section firstly briefly discusses simple PSO and then proposes dynamic PSO (DPSO), describe in detail.

2.1. Simple PSO

In 1995, two scientists, Kennedy and Eberhart, presented the PSO [11]. It falls into the family of swarm-based algorithms enthused by bird flocking and fish schooling. PSO renew the particle (candidate solution) using the velocity- and position-updating equations. Over the past two decades, PSO has garnered considerable interest among researchers due to its simple framework, few operations and parameters, and robust performance.

2.2. Proposed Dynamic PSO algorithm

The DPSO is developed to enhance convergence performance and local refinement while sustaining population diversity. The design algorithm differs from simple PSO in the parameter directives for the velocity equation and the trajectory modulation factor (TMF) in the position update equation. The proposed DPSO is presented below in detail.

2.2.1.1. The dynamic control parameters

Inertia weight coefficient ω

It exponentially set to "0.1". This value is compatible with other designed parameters and helps improve algorithm performance. Due to the constant value, it consistently maintains the flight particles' exploration and exploitation capabilities throughout the optimization. As a result, the optimizer can optimize consistently with a stronger global searching capability and convergence pace.

Acceleration coefficients c_1 and c_2

Acceleration coefficients may be designed as:

$$c_1 = e^{-16\left(\frac{t}{T}\right)^2}, c_2 = 1 - c_1 \quad (1)$$

The cognitive coefficients $c_1 \in (0, 1]$ slightly decreases, and at the same time, the social coefficients $c_2 \in [0, 1)$ slightly increases over time. This process gently pulls the particles into a wider feasible region and then rapidly shifts them towards the optimal solution.

The trajectory modulation factor (TMF) α

This factor newly introduced and can be formulated as:

$$\alpha = \left(\frac{2.5 - 2*k}{0.5 + 2*k}\right) * (k)^2; k = \frac{t}{T} \quad (2)$$

The TMF increases gradually, then progressively decreases over time. This TMF feature expands the particle trajectory memory and allows broader movement; as a result, it will not rely solely on particle velocity. This promotes broader search and helps avoid suboptimum entrapment and particle stagnation. Later, progressively narrow the particle trajectory with enhancing convergence performance in the vicinity of the global solution. Incorporating all the introduced parameters into DPSO, the velocity and position updated rule become:

$$v_i^j(t+1) = \omega * v_i^j(t) + c_1 * r_1 * \left({p_{best}}_i^j(t) - x_i^j(t)\right) + c_2 * r_2 * \left({g_{best}}^j(t) - x_i^j(t)\right) \quad (3)$$

$$x_i^j(t+1) = \alpha * x_i^j(t) + v_i^j(t+1) \quad (4)$$

The behavior of parameters (c_1, c_2, and α) during the optimization of DPSO is represented in Figure 1.

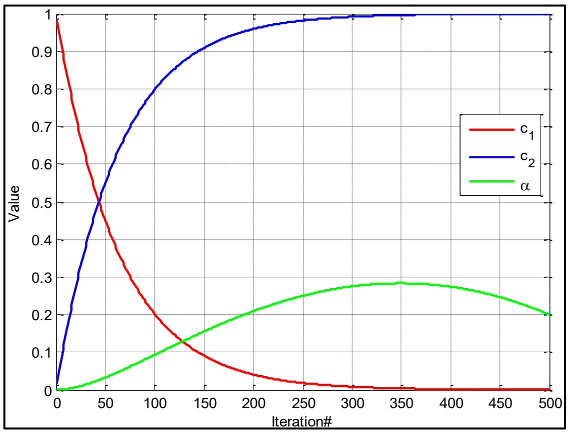

Figure 1. Behavior of parameters (c_1, c_2, and α) during the optimization of DPSO

3. The AVR mathematical model and PID tuning expression

Automatic Voltage Regulator (AVR) System

The AVR System [12], [13], [14], [15] is employed to preserve the terminal voltage of a synchronous generator in power systems constant within the designed reference range. Amplifier, exciter, generator, and sensor are the essential components of the AVR system. Table 1 presents the mathematical representation of the system's transfer function.

Table 1. Transfer function component

Component	Expression	Parameters
Amplifier	$G_A(s) = \frac{K_A}{(1+sT_A)}$	K_A and T_A, respectively, stand for amplifier gain and time constant.
Exciter	$G_E(s) = \frac{K_E}{(1+sT_E)}$	K_E and T_E, respectively, specify the exciter gain and time constant.
Generator	$G_G(s) = \frac{K_G}{(1+sT_G)}$	K_G and T_G, respectively, represents the generator gain and time constant.
Sensor	$G_S(s) = \frac{K_S}{(1+sT_S)}$	K_S and T_S, respectively, stand for sensor gain and time constant.

Combining all subsystem transfer functions, the complete AVR System transfer function is formulated as:

$$G_{AVR}(s) = \frac{K_A * K_E * K_G * K_S}{(1+sT_A)*(1+sT_E)*(1+sT_G)*(1+sT_S)} \quad (5)$$

Table 2 summarizes the standard parameters and design values used in the AVR system [16], [17].

Table 2. AVR system parameters design values

Component	**Selected parameter values**	
	Gain	**Time constant**
Amplifier	$K_A = 10$	$T_A = 0.1$
Exciter	$K_E = 1$	$T_E = 0.4$
Generator	$K_G = 1$	$T_G = 1$
Sensor	$K_S = 1$	$T_S = 0.01$

Including these parameter values, the AVR system transfer function becomes

$$G_{AVR}(s) = \frac{10}{(0.1s+1)*(0.4s+1)*(s+1)*(0.01s+1)} \quad (6)$$

Proportional-Integral-Derivative (PID) controller

The PID controller used for the AVR tuning is expressed as equation (7).

$$G_C(s) = K_P + \frac{K_I}{s} + sK_D \quad (7)$$

where, K_P, K_I and K_D, respectively, implies the proportional, integral, and derivative gain.

The transfer function in the closed-loop

The closed-loop AVR system with a PID controller is given by equation (8).

$$T(s) = \frac{G_C(s)*G_{AVR}(s)}{1+G_C(s)*G_{AVR}(s)} \quad (8)$$

The objective function

The PID controller tuning problem is formulated as optimization problem by the Integral Time Absolute Error (ITAE) measure:

$$\text{ITAE} = \int_0^t t|e(t)|dt \quad (9)$$

The optimization function is:

$$\min f(K_P, K_I, K_D) = \int_0^t t|e(t)|dt \quad (10)$$

subject to constrained is:

$0.01 \leq K_P, K_I, K_D \leq 2.$

4. Simulation results and comparative analysis

The primary objective of the experiments in the current study is to acquire optimal PID controller parameters tuning using projected DPSO in an AVR system. The experimental results and performance were compared with well-recognized and recently presented techniques, including PSO [11], Dung Beetle Optimizer (DBO) [18], Golf Optimization Algorithm (GOA) [19], and Divine Religions Algorithm (DRA) [20]. All the parameter settings of these optimizers are same as their original work. For fair comparison, the experimental setting is adopted:

- N (number of populations) = 50.
- T (maximum iteration count) = 100.

All outcomes for objective function are presented and evaluated based on the average performance of approaches across continuous 30 independent trials. The bold print signifies either a better outcome or the best-known outcome. All optimizers were coded in the "MATLAB" platform, student version, on a "DESKTOP (Windows 11 (64-bit)) equipped with an Intel(R) Core (TM) i5-7200U CPU @2.70 GHz and 12 GB of RAM". Table 3 presents the obtained PID controller parameters using several optimizers.

Table 3. Optimized PID parameters

Algorithm	PID parameters		
	K_P	K_I	K_D
PSO	1.4783	1.0154	0.4765
DBO	1.3433	0.9252	0.4330
GOA	1.2670	0.8716	0.4063
DRA	1.2544	0.8671	0.3952
DPSO	1.04576	0.73106	0.32475

Table 4 illustrates the time-domain analysis and shows that DPSO performs better at controlling performance than the rival. Furthermore, DPSO obtained the lower setting time and overshoot among all the optimizers. The minimum setting time specifies the faster stability of terminal voltages after load changes. The minimum overshoot represents optimization of the unwanted voltage deviation and ensure more reliable operation. This performance shows that the DPSO has better optimization capability for computing PID parameter settings, thereby improving voltage regulation, ensuring faster system stability, and reducing oscillations in the AVR system.

Table 4: Time-domain analysis

Algorithm	Time-domain analysis				
	Rise Time (sec)	Settling Time (sec)	Overshoot (%)	Peak Time (sec)	E_{SS}
PSO	**0.137197**	0.803009	21.950634	**0.310644**	1.289366e-07
DBO	0.145199	0.798496	19.910387	0.326020	1.330238e-07
GOA	0.163107	0.853230	16.376125	0.355569	1.341553e-07
DRA	0.1409	0.7884	21.4603	0.3240	**7.07779e-08**
DPSO	0.164036	**0.760835**	**7.609371**	0.43088	1.303185e-07

After time-domain analysis, the statistical performance is summarized in Table 5. This table includes the performance metrics: best, mean, worst, and standard deviation (Std dev) of the objective function values. Table 5 displays that DPSO produced a better or comparable solution to the rival optimizers across all performance metrics. Additionally, the DPSO delivered the lowest best value, "0.0328", which is similar to the PSO, DBO, and GOA. The least mean and Std dev values of DPSO are the same as GOA, whereas highly competitive worst values are obtained with GOA. Therefore, advised DPSO perform superior in performance metrics, best, mean, and Std dev compared to all optimizers.

Table 5. Statistical analysis

Algorithm	Best	Mean	Worst	Std dev
PSO	**0.0328**	0.0333	0.0348	0.0007
DBO	**0.0328**	0.0330	0.0335	0.0002
GOA	**0.0328**	**0.0329**	**0.0333**	**0.0001**
DRA	0.0329	0.0330	0.0334	0.0002
DPSO	**0.0328**	**0.0329**	0.0334	**0.0001**

Furthermore, DPSO performance was assessed through visualizations. For this purpose, Figure 2 represents the step response of the AVR system, including all controllers. Visualize Figure 2, which indicates that from 1 second onward, the reference value is 1 per-unit (p.u.), and the system reaches complete balance thereafter. The error variation over time is represented in Figure 3 and reflects that the error variation nearly becomes "0" at 1 second after the commencement.

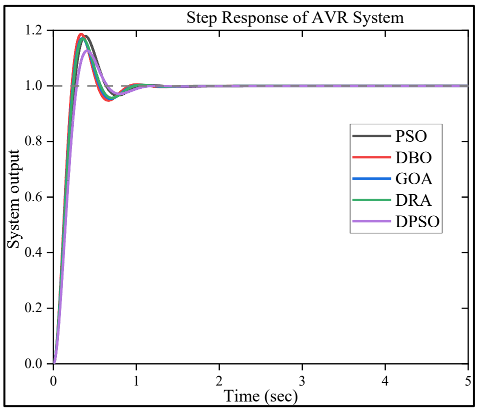

Figure 2. System output and reference

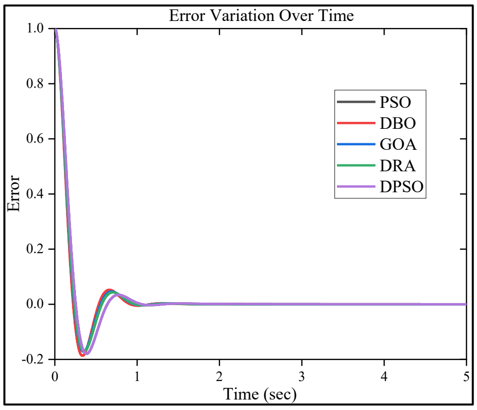

Figure 3. Error variation over time

The convergence performance of the DPSO and its rival optimizers is demonstrated in Figure 4. In Figure 4, the horizontal axis and vertical axis, respectively, represent the iteration count and the objective function's fitness values. Clearly, this figure shows that DPSO converges with more agility than its rival. Also, DPSO performs superior to others as the iteration count rises. Overall, based on numerical assessments and graphical representations, DPSO outperforms other methods for tuning PID controller parameters in an AVR system.

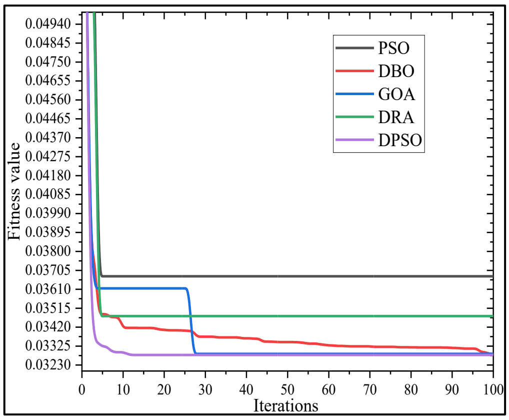

Figure 4. Convergence performance

5. Conclusion and future research ideas

Dynamic PSO (DPSO) is suggested for the optimal PID controller parameters tuning in an AVR system. DPSO is based on dynamic parameter adjustment in the velocity update equation and the trajectory modulation factor (TMF) in the position update equation. DPSO uses- (i) constant inertia weight that assists in reducing redundant momentum, (ii) acceleration coefficients (time-varying) which help in endorsing better transition from exploration to exploitation, and (iii) TMF that facilitates dynamically shrinking particle trajectory and helps in properly maintaining diversity. Simulation results prove that DPSO delivers better control outcomes for PID controller parameters than rival approaches in measures of settling time, overshoot, and statistical performance. Future research: DPSO applied to complex scenarios to further assess its robustness in the power system.

References

[1] M. N. Ali, M. Soliman, K. Mahmoud, J. M. Guerrero, M. Lehtonen, and M. M. F. Darwish, "Resilient Design of Robust Multi-Objectives PID Controllers for Automatic Voltage Regulators: D-Decomposition Approach," *IEEE Access*, vol. 9, pp. 106589–106605, 2021, doi: 10.1109/ACCESS.2021.3100415.

[2] L. dos Santos Coelho, "Tuning of PID controller for an automatic regulator voltage system using chaotic optimization approach," *Chaos Solitons Fractals*, vol. 39, no. 4, pp. 1504–1514, Feb. 2009, doi: 10.1016/j.chaos.2007.06.018.

[3] B. Hekimoglu, S. Ekinci, and S. Kaya, "Optimal PID Controller Design of DC-DC Buck Converter using Whale Optimization Algorithm," in *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, IEEE, Sep. 2018, pp. 1–6. doi: 10.1109/IDAP.2018.8620833.

[4] A. V. R. L. A. Md. S. Tripathi, "Design of Ziegler Nichols Tuning Controller for AVR System," *International Journal of Research in Electronics & Communication Technology* , vol. 1, no. 2, pp. 154–158, Oct. 2013.

[5] Z.-L. Gaing, "A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System," *IEEE Transactions on Energy Conversion*, vol. 19, no. 2, pp. 384–391, Jun. 2004, doi: 10.1109/TEC.2003.821821.

[6] U. GÜVENÇ, T. YİĞİT, A. H. IŞIK, and İ. AKKAYA, "Performance analysis of biogeography-based optimization for automatic voltage regulator system," *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, vol. 24, pp. 1150–1162, 2016, doi: 10.3906/elk-1311-111.

[7] H. Gozde and M. C. Taplamacioglu, "Comparative performance analysis of artificial bee colony algorithm for automatic voltage regulator (AVR) system," *J. Franklin Inst.*, vol. 348, no. 8, pp. 1927–1946, Oct. 2011, doi: 10.1016/j.jfranklin.2011.05.012.

[8] D. Izci, S. Ekinci, H. L. Zeynelgil, and J. Hedley, "Performance evaluation of a novel improved slime mould algorithm for direct current motor and automatic voltage regulator systems," *Transactions of the Institute of Measurement and Control*, vol. 44, no. 2, pp. 435–456, Jan. 2022, doi: 10.1177/01423312211037967.

[9] D. Izci, S. Ekinci, and H. Cetin, "Arithmetic Optimization Algorithm based Controller Design for Automatic Voltage Regulator System," in *2022 Innovations in Intelligent Systems and Applications Conference (ASYU)*, IEEE, Sep. 2022, pp. 1–5. doi: 10.1109/ASYU56188.2022.9925431.

[10]S. Ekinci, A. Demiroren, and B. Hekimoglu, "Parameter optimization of power system stabilizers via kidney-inspired algorithm," *Transactions of the Institute of Measurement and Control*, vol. 41, no. 5, pp. 1405–1417, Mar. 2019, doi: 10.1177/0142331218780947.

[11]J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, IEEE, pp. 1942–1948. doi: 10.1109/ICNN.1995.488968.

[12]A. F. Güven, O. Özdal Mengi, and M. M. Samy, "Adjustment of PID Controller Parameters in AVR Using the Spider Wasp Optimizer," in *2024 25th International Middle East Power System Conference (MEPCON)*, IEEE, Dec. 2024, pp. 1–5. doi: 10.1109/MEPCON63025.2024.10850298.

[13]V. Mukherjee and S. P. Ghoshal, "Intelligent particle swarm optimized fuzzy PID controller for AVR system," *Electric Power Systems Research*, vol. 77, no. 12, pp. 1689–1698, Oct. 2007, doi: 10.1016/j.epsr.2006.12.004.

[14]A. J. Sultan, "Optimal AVR Control System using Particle Swarm Optimization," *IJARCCE*, vol. 6, no. 1, pp. 139–142, Jan. 2017, doi: 10.17148/IJARCCE.2017.6127.

[15]A. Bouaddi, R. Rabeh, and M. Ferfra, "Optimal control of automatic voltage regulator system using hybrid PSO-GWO algorithm-based PID controller," *Bulletin of Electrical Engineering and Informatics*, vol. 13, no. 5, pp. 3070–3080, Oct. 2024, doi: 10.11591/eei.v13i5.8186.

[16]S. M. A. Altbawi, A. S. Bin Mokhtar, T. A. Jumani, I. Khan, N. N. Hamadneh, and A. Khan, "Optimal design of Fractional order PID controller based Automatic voltage regulator system using gradient-based optimization algorithm," *Journal of King Saud University - Engineering Sciences*, vol. 36, no. 1, pp. 32–44, Jan. 2024, doi: 10.1016/j.jksues.2021.07.009.

[17]S. Ekinci and B. Hekimoglu, "Improved Kidney-Inspired Algorithm Approach for Tuning of PID Controller in AVR System," *IEEE Access*, vol. 7, pp. 39935–39947, 2019, doi: 10.1109/ACCESS.2019.2906980.

[18]J. Xue and B. Shen, "Dung beetle optimizer: a new meta-heuristic algorithm for global optimization," *J. Supercomput.*, vol. 79, no. 7, pp. 7305–7336, May 2023, doi: 10.1007/s11227-022-04959-6.

[19]Z. Montazeri, T. Niknam, J. Aghaei, O. P. Malik, M. Dehghani, and G. Dhiman, "Golf Optimization Algorithm: A New Game-Based Metaheuristic Algorithm and Its Application to Energy Commitment Problem Considering Resilience," *Biomimetics*, vol. 8, no. 5, p. 386, Aug. 2023, doi: 10.3390/biomimetics8050386.

[20]A. T. Mozhdehi *et al.*, "Divine Religions Algorithm: a novel social-inspired metaheuristic algorithm for engineering and continuous optimization problems," *Cluster Comput.*, vol. 28, no. 4, p. 253, Aug. 2025, doi: 10.1007/s10586-024-04954-x.