İSTANBUL UNIVERSITY CERRAHPAŞA

RESEARCH ARTICLE

Electrica 2021; 21(1): 151-159

Comparative Performance Analysis of Slime Mould Algorithm For Efficient Design of Proportional–Integral–Derivative Controller

Davut İzci[1], Serdar Ekinci[2]

[1]Department of Electronics and Automation, Vocational School of Technical Sciences, Batman University, Batman, Turkey
[2]Department of Computer Engineering, Faculty of Engineering and Architecture, Batman University, Batman, Turkey

Cite this article as: İzci D, Ekinci S. Comparative Performance Analysis of Slime Mould Algorithm For Efficient Design of Proportional–Integral–Derivative Controller. *Electrica,* 2021; 21(1): 151-159.

ABSTRACT

This paper deals with the performance analysis of a recently proposed metaheuristic algorithm known as the slime mould algorithm (SMA). This algorithm has been proved to be effective on several benchmark functions and constraint problems. This study further demonstrates its ability based on optimizing real-life engineering problems. Thus, the optimization ability of the SMA has been assessed by adopting proportional-integral-derivative (PID) controllers to regulate the speed of a direct current (DC) motor and maintaining the terminal output of an automatic voltage regulator (AVR) system. The obtained results were compared with the controller performances designed by other competitive metaheuristic algorithms, such as Harris hawks optimization (HHO), atom search optimization (ASO), and grey wolf optimization (GWO) algorithms for DC motor and symbiotic organisms search (SOS), local unimodal sampling (LUS), and many optimizing liaisons (MOL) algorithms for AVR system. The results showed that the PID controllers tuned by the SMA technique have superior performance compared to other counterparts.

Keywords: Slime mould algorithm, PID controller, automatic voltage regulator, speed control

Corresponding Author:
Davut İzci

E-mail:
davut.izci@batman.edu.tr

Received: 03.09.2020

Accepted: 27.10.2020

DOI: 10.5152/electrica.2021.20077

Content of this journal is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Introduction

Proportional-integral-derivative (PID) controllers have a wide range of application areas, such as power plants, chemical processing, automatic control, and industrial processes, despite many other available controllers [1]. The demand for PID controllers arises from their simple and yet effective structure. However, the effectiveness of PID controllers depends on the fine-tuning of their parameters [2]. Traditional methods such as Ziegler-Nichols, Cohen-Coon, or manual tuning are available for the latter task; however, these methods cannot provide the desired stability and good transient response in increased complexity [3]. Metaheuristic algorithms are methods used to tune PID parameters to achieve enhanced capabilities and deal with complex problems [4]. Unlike traditional methods, metaheuristic algorithms have stochastic nature with gradient-free mechanisms. Therefore, metaheuristic algorithms require minimal mathematical analysis since they consider and solve optimization problems using only the inputs and outputs. Thus, they assume an optimization problem as a black box without calculating the search space's derivative. This is one of the essential advantages of metaheuristic algorithms, making them highly flexible for solving a diverse range of problems. Hence, it is the reason for their widespread use.

Direct current (DC) motor and automatic voltage regulator (AVR) systems have been application areas for many metaheuristic algorithms as real-life engineering problems since they provide an observable test bed for performance evaluations and comparisons. Some of the examples of metaheuristic optimization methods for PID controlled DC motors are Harris hawks optimization (HHO) [5], atom search optimization (ASO) [6], grey wolf optimization (GWO) [7], particle swarm optimization (PSO) [8, 9], flower pollination (FPA) [10], stochastic fractal search (SFS) [11], teaching-learning based optimization (TLBO) [12], swarm learning process (SLP) [1], and improved genetic [13] algorithms. However, examples of PID controlled AVR systems are symbiotic organisms search (SOS) [14], local unimodal sampling (LUS) [15], many optimizing li-

aisons (MOL) [16], stochastic fractal search (SFS) [17], improved kidney-inspired (IKA) [18], improved spotted hyena optimization (ISHO) [4], tree seed (TSA) [19], artificial ecosystem-based optimization (AEO) [20], enhanced crow search (ECSA) [3], and water cycle (WCA) [21] algorithms.

Despite various metaheuristic algorithms, there has been an ongoing study to develop new methods. The need to develop new approaches can be described by the No Free Lunch (NFL) theorem [22]. According to this theorem, there is no optimization algorithm capable of finding the optimal solution for every optimization problem. Therefore, no algorithm can solve all optimization problems, with different types and nature, effectively than any other option. Instead, each of them can be quite successful for a specific set of problems.

Slime mould algorithm (SMA) [23] has been proposed as a novel-metaheuristic optimization algorithm as part of the NFL theorem's assumptions. It has already been proven successful on several benchmark functions and classical engineering problems [23] and estimates solar photovoltaic cell parameters [24, 25]. However, many real-life engineering problems are still available for SMA performance evaluation. In this study, DC motor speed regulation and AVR system control were adopted as two new real-life engineering problems to assess the SMA algorithm's performance.

Providing efficient solutions to the problems of DC motor speed regulation and AVR system control are essential contributions of this study based on the newly developed algorithm's performance evaluation. To present a thorough evaluation, the obtained results of the optimized systems using SMA were compared with the acquired results using other metaheuristic algorithms, such as HHO [5], ASO [6], and GWO [7] algorithms for DC motor and SOS [14], LUS [15] and MOL [16] algorithms for AVR system.

Slime mould algorithm

SMA is one of the recently proposed stochastic optimization methods that mimics the foraging behavior of Physarum Polycephalum [23] instead of its entire life cycle. The mathematical model can be explained under three subsections.

Approaching food

In this behavior, the odor in the air attracts the slime mould to approach the food. This can be modeled as in (1):

$$\overrightarrow{X(t+1)} = \begin{cases} \overrightarrow{X_b(t)} + \overrightarrow{vb} \cdot \left(\vec{W} \cdot \overrightarrow{X_A(t)} - \overrightarrow{X_B(t)}\right), r < p \\ \overrightarrow{vc} \cdot \overrightarrow{X(t)}, r \geq p \end{cases} \quad (1)$$

where $\vec{X}$ denotes the slime mould's location, $\vec{X_b}$ is the location with the highest odor concentration that has been found so far. $\vec{X_A}$ and $\vec{X_B}$ are randomly chosen individuals from slime mould. $\vec{vb}$ is a parameter having a range of $[-a, a]$. The value of a is given as follows:

$$a = arctanh\left(-\left(\frac{t}{t_{max}}\right) + 1\right) \quad (2)$$

where t_{max} is the maximum iteration. $\vec{vc}$ represents a parameter that decreases from one to zero. The term shown by p is defined as in (3):

$$p = tanh|S(i) - DF| \quad (3)$$

where the fitness $\vec{X}$ of is denoted by $s(i)$ and $i \in 1,2,\ldots n$. DF is the representation of the best fitness acquired in all iterations. The weight of slime mould is represented by $\vec{W}$ and is calculated using (4).

$$\vec{W}(SmellIndex(i)) = \begin{cases} 1 + r \cdot \log\left(\frac{bF - S(i)}{bF - wF} + 1\right), condition \\ 1 - r \cdot \log\left(\frac{bF - S(i)}{bF - wF} + 1\right), others \end{cases} \quad (4)$$

r represents a random value in a range of $[0, 1]$ and $condition$ is an indication of first half of the population that are ranked by $S(i)$. The optimal and worst fitness achieved in the current iteration are represented by bF and wF, respectively. The sequence of fitness values is represented by $SmellIndex$ and sorted as follows:

$$SmellIndex = sort(S) \quad (5)$$

Wrapping food

In this approach, the contraction mode of slime mould is simulated mathematically. A strong wave is generated by slime mould when a high concentration of food is contracted by the vein, causing a fast cytoplasmic flow and a thick vein. The relationship between the food concentration and the vein width is simulated mathematically by (4). Thus, the weight near the region is bigger when the food concentration is satisfying. The region's weight reduces for lower concentrations, thereby allowing other regions to be explored. The slime mould's location is updated using the above principle and is given as:

$$\overrightarrow{X^*} = \begin{cases} rand \cdot (UB - LB) + LB, rand < z \\ \overrightarrow{X_b(t)} + \overrightarrow{vb} \cdot \left(\vec{W} \cdot \overrightarrow{X_A(t)} - \overrightarrow{X_B(t)}\right), r < p \\ \overrightarrow{vc} \cdot \overrightarrow{X(t)}, r \geq p \end{cases} \quad (6)$$

where the lower and upper boundaries of search space is denoted by LB and UB, respectively. r and $rand$ are random values within $[0, 1]$ and z is a parameter within a range of $[0, 0.1]$.

Oscillation

The cytoplasmic flow in veins is changed by the biological oscillator, which produces a propagation wave. Slime mould mainly depends on this produced wave for finding a food position with a better concentration. $\vec{W}$, $\vec{vb}$ and $\vec{vc}$ are used to simulate different venous width.

The slime mould's oscillation frequency is simulated mathematically using $\vec{W}$. Slime mould can approach the location of food with higher concentration more quickly. However, in the

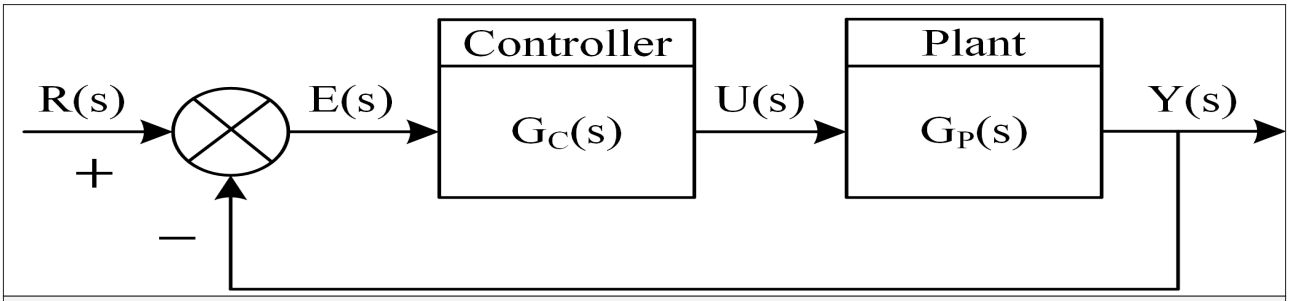


Figure 1. Block diagram of feedback control system with PID controller

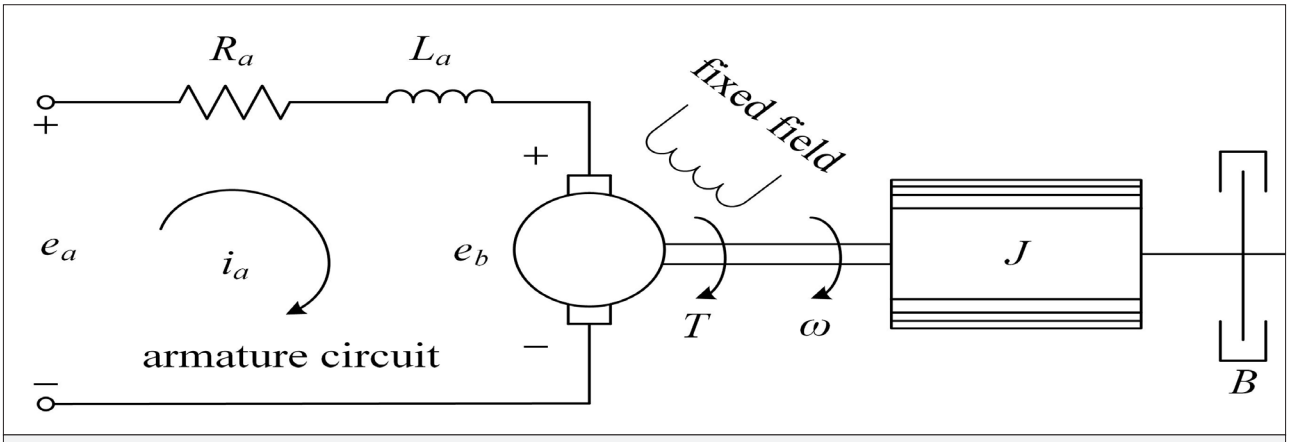


Figure 2. DC motor equivalent circuit

case of locations with lower food concentrations, it approaches slowly. In this way, slime mould efficiency is improved by selecting the optimal food source.

The slime mould's selective behavior is mimicked by the synergistic interaction between $\vec{vb}$ and $\vec{vc}$. Slime mould separates some organic matter to explore different areas. This behavior explores a food source with higher quality, even if it has found a better food source. The latter case also prevents slime mould from investigating only one source. Moreover, the decision of whether approaching the food source or finding other sources is simulated by the oscillation process of $\vec{vb}$. There may be some restrictions, such as dry environment and light, which do not allow slime mould calculation. However, the possibility of finding a food source with higher quality is improved without trapping in a local optimum. The pseudo-code of SMA is given in Algorithm 1.

PID controllers

Due to their effectiveness and simplicity, PID controllers are mostly adopted controllers in industrial processes. A PID controller is used to remove or decrease steady-state error and enhance the system's dynamic performance. The transfer function of a PID controller is provided in (7), where K_p, K_i and K_d are the coefficients to be tuned for the proportional, integral, and derivative terms, respectively.

Algorithm 1. Pseudo-code of slime mould algorithm

Initialize population size, maximum iteration;

Initialize slime mould positions $X_i\ (i=1,2,\ldots n)$;

while ($t \leq$*maximum iteration*)

calculate all slime mould's fitness;

update best fitness, X_b;

calculate W using (4);

for each position of search

update vb,vc,p;

update positions using (6);

end for

t++;

end while

return best fitness, X_b;

$$G_C(s) = \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s \quad (7)$$

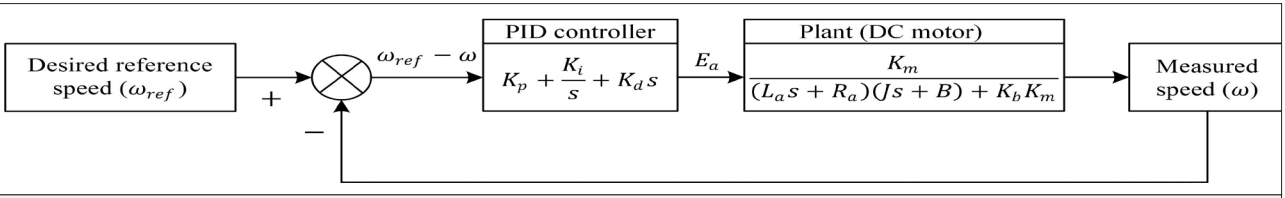


Figure 3. DC motor system with PID controller

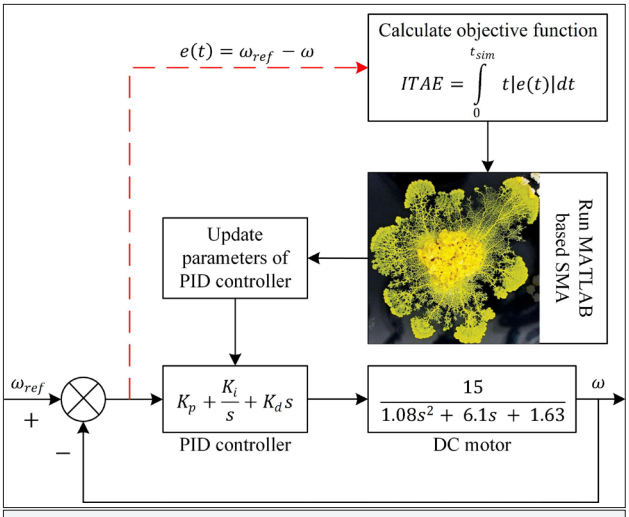


Figure 4. Proposed SMA based PID tuning procedure in DC motor system

Figure 1 shows a basic PID control system, where $R(s)$ is the reference signal, $Y(s)$ is the actual output, $E(s)$ is the tracking error signal, $U(s)$ is the control signal, $Gc(s)$ is the controller, and $Gp(s)$ is plant.

PID controller design for speed control

Modeling of DC motor control system

Regulating the speed of a DC motor is an essential task to carry out specific work. In principle, a DC motor operates by converting electrical energy into mechanical form. Figure 2 illustrates the equivalent circuit of the DC motor. The open-loop transfer function that demonstrates the relationship of the output speed ($\Omega(s)$) and the input voltage ($E_a(s)$) of the DC motor can be written as given in (8).

$$G_{DC-motor}(s) = \frac{\Omega(s)}{E_a(s)} = \frac{K_m}{(L_a s + R_a)(Js + B) + K_b K_m} \quad (8)$$

The parameters of DC motor were chosen to be $0.0004\ kg \cdot m^2$ for moment of inertia (J), $0.0022\ N \cdot m \cdot s/rad$ for friction coefficient (B), $0.4\ \Omega$ for armature resistance (R_a) 2.7 H, for armature inductance (L_a), $0.05\text{V} \cdot s/rad$ for back EMF constant (K_b), $0.015\ N \cdot m/A$ for motor torque constant (K_m) [5,6,7]. The transfer function given in (9) is obtained by substituting the above parameters in (8).

$$G_{DC-motor}(s) = \frac{\Omega(s)}{E_a(s)} = \frac{15}{1.08s^2 + 6.1s + 1.63} \quad (9)$$

DC motor speed control with PID

Considering the previously proposed DC motor and PID controller models, the block diagram of the closed-loop DC motor speed control system with the PID controller can be demonstrated as in Figure 3 along with the transfer function as given in (10).

$$G_{DC-motor+PID}(s) = \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{15(K_d s^2 + K_p s + K_i)}{1.08s^3 + 6.1s^2 + 1.63s + 15(K_d s^2 + K_p s + K_i)} \quad (10)$$

SMA based PID controller for DC motor

To optimize the system's dynamic response and stability of the system, the integral of time multiplied absolute error $(ITAE)$ was adopted as the objective function. The mathematical expression of $ITAE$ is given in (11), where t_{sim} is the simulation time and $e(t)$ is the error signal. The error signal is represented as the arithmetic difference between the reference angular speed (ω_{ref}) and the measured angular speed (ω).

$$ITAE = \int_0^{tsim} t|e(t)|dt \quad (11)$$

In this study, the gain parameters were tuned via SMA by considering the PID controller's lower and upper bounds to be 0.001 and 20, respectively. These are commonly used values in the literature for respective parameters [5,6,7]. Minimizing the fitness function $(ITAE)$ improves the system's response and stability. A DC motor with a PID controller tuned by the SMA is demonstrated in Figure 4.

PID controller design for AVR system

Modeling of AVR control system

Maintaining the terminal voltage of a generator at the desired magnitude is the primary function of an AVR system. There are four main components of the AVR system: amplifier, exciter, generator, and sensor. Figure 5 shows the block diagram of a simple AVR system.

In this work, the parameters of AVR system were chosen to be 10.00 for amplifier gain (K_a), $0.10\ s$ for amplifier time constant

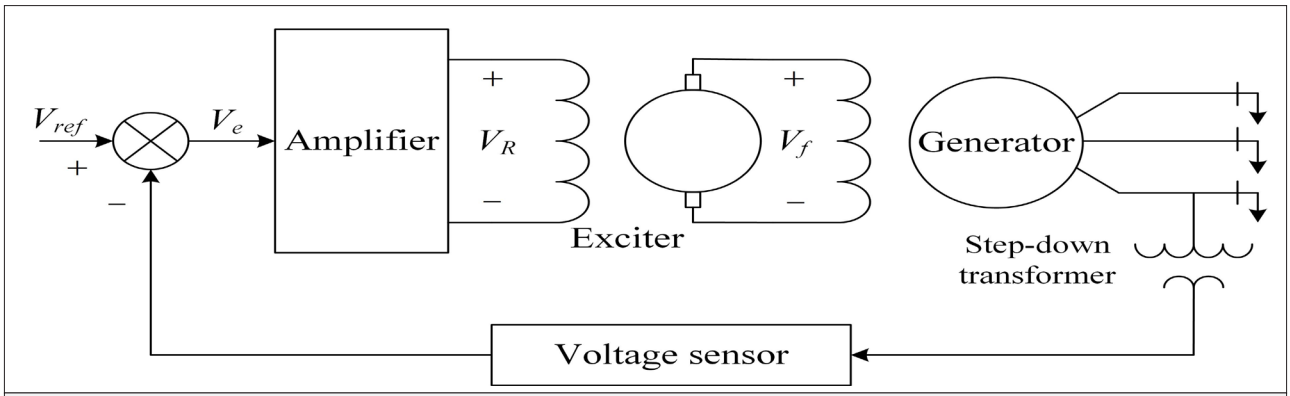


Figure 5. Schematic diagram of a simple AVR

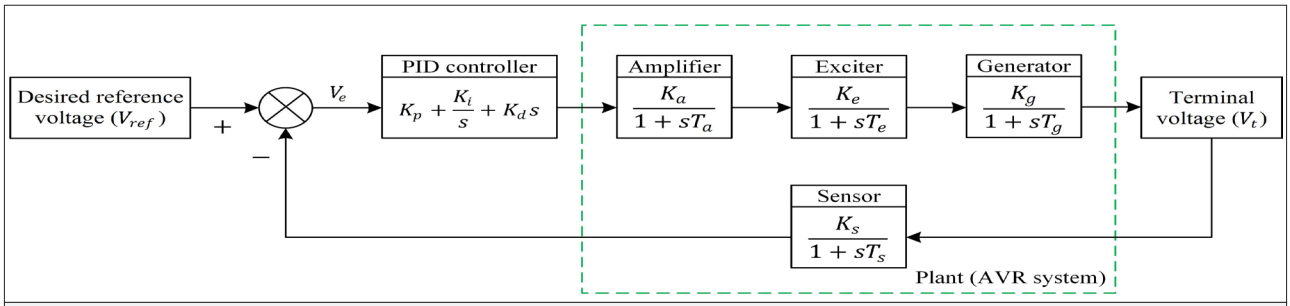


Figure 6. AVR system with PID controller

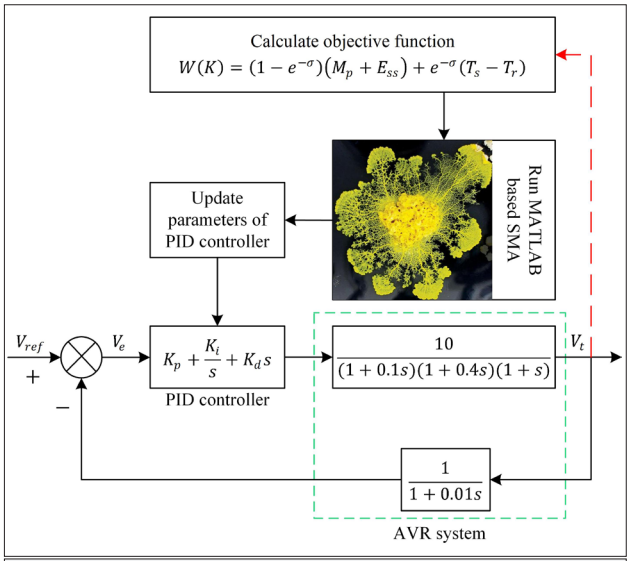


Figure 7. Proposed SMA based PID tuning procedure in AVR system

(T_a), 1.00 for exciter gain (K_e), 0.40 s for exciter time constant (T_e), 1.00 for generator gain (K_g), 1.00 s for generator time constant (T_g), 1.00 for sensor gain (K_s), and 0.01 s for sensor time constant (T_s) for a fair comparison with refs. [14–16]. The AVR system's transfer function with the above parameters is obtained as given in (12).

$$G_{AVR}(s) = \frac{V_t(s)}{V_{ref}(s)} = \frac{0.1\,s + 10}{0.0004s^4 + 0.0454s^3 + 0.555s^2 + 1.51s + 11} \quad (12)$$

AVR system with PID

As shown in Figure 6, a PID controller was added to the AVR system to maintain the terminal voltage at the desired level and improve the dynamic response. The AVR system's transfer function with the PID controller is provided in (13).

$$G_{AVR+PID}(s) = \frac{V_t(s)}{V_{ref}(s)} = \frac{(0.1\,s + 10)(K_d s^2 + K_p s + K_i)}{0.0004\,s^5 + 0.0454\,s^4 + 0.555\,s^3 + 1.51\,s^2 + s + 10(K_d s^2 + K_p s + K_i)} \quad (13)$$

SMA based PID controller for AVR system

The SMA was used to tune the gain parameters of a PID controller to achieve a desired dynamic response of an AVR system. In this study, the range of PID parameters for the AVR system was chosen to be [0.0, 2.0] for the lower and the upper gain bounds of PID to reach a wider initial search space and obtain better-optimized gains. To obtain optimal K_p, K_i, and K_d values, a time-domain performance index $(W(K))$, provided in (14), was considered in this study since it includes time response specifications.

$$W(K) = (1 - e^{-\sigma})(M_p + E_{ss}) + e^{-\sigma}(T_s - T_r) \quad (14)$$

Here, $K = [K_p, K_i, K_d]$, M_p is peak overshoot, E_{ss} is a steady-state error, T_s is settling time, and T_r is rise time. σ is a weighting coef-

Table 1. PID parameters obtained with different algorithms for DC motor system

Algorithm	K_p	K_i	K_d
SMA (proposed)	18.6458	2.5235	3.1921
HHO [5]	15.8581	3.6963	2.7732
ASO [6]	11.9437	2.0521	2.4358
GWO [7]	6.8984	0.5626	0.9293

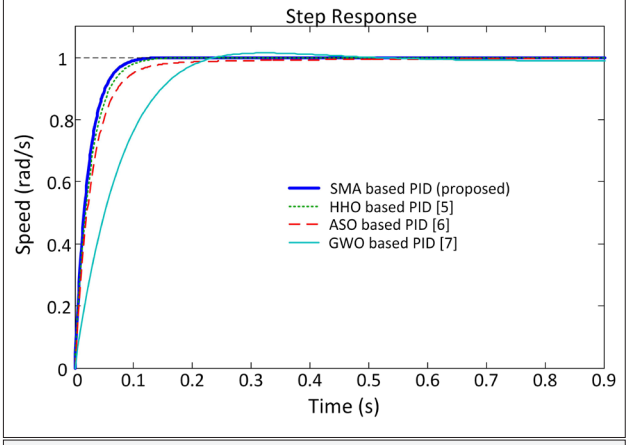


Figure 8. Speed comparison of the proposed tuning method and other methods used recently in the literature

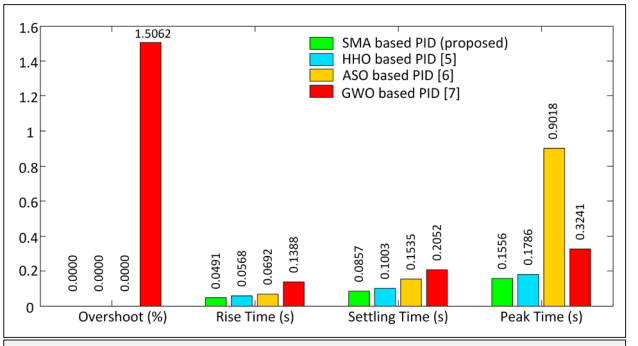


Figure 9. Comparison of maximum percentage overshoot, rise, settling and peak times for various tuning methods

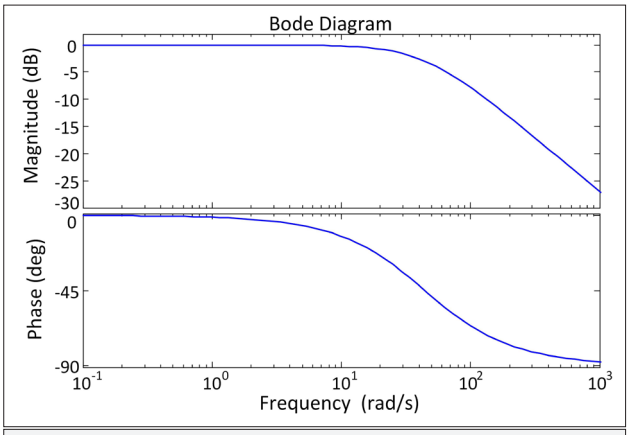


Figure 10. Bode diagram of DC motor with SMA based PID controller

ficient, which is set as 1.0. Figure 7 shows the block diagram of the proposed SMA based PID controlled AVR system.

Simulation results and discussion

The proposed method simulation was conducted using MATLAB/Simulink (Version: 2014a) software installed on an Intel CoreTM i7 computer with a 2.50 GHz processor. The maximum number of iterations and population size (search agents) of the SMA was 50 and 40, respectively, for both DC motor and AVR control systems. The obtained statistical results for DC motor and AVR control systems were averaged over 30 runs since the adopted optimization has a stochastic nature.

Comparative results for DC motor

The successful completion of the optimization process by adopting SMA produced the PID controller parameters of $K_p = 18.6458$, $K_i = 2.5235$ and $K_d = 3.1921$. Substituting those gains in (10) provides the transfer function given in (15).

$$G_{DC-motor+PID+SMA}(s) = \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{47.88s^2 + 279.7s + 37.85}{1.08s^3 + 53.98s^2 + 281.3s + 37.85} \quad (15)$$

Table 1 contains the gain parameters obtained using other recent metaheuristic algorithms. The listed ones were the appropriate algorithms that were chosen for comparison. The comparative step and transient response analysis for the speed control system designed by different approaches are presented in figures 8 and 9.

Figure 8 presents the step response comparison. Figure 9 shows the bar plot comparisons of maximum percentage overshoot, rise time (for 10% → 90%), settling time (for a tolerance band of ±2), and peak time, respectively. From these results, the SMA based PID controller has a better time response than others. Therefore, the proposed design approach of the PID controller for DC motor speed regulation is superior to other design approaches, such as HHO [5], ASO [6], and GWO [7], with better transient stability, fast damping characteristics, and no overshoot.

Figure 10 shows the SMA based PID controller's magnitude and phase plot in a DC motor system. The comparison of frequency-domain parameters, such as gain margin (in decibel), phase margin (in degrees), and bandwidth (in Hertz) for SMA and other algorithms based PID controllers are given in Table 2. The table shows that the system tuned by SMA is the most stable in terms of frequency response criteria.

Comparative results for AVR system

The optimization process adopting SMA produced the PID parameters of $K_p = 0.6173$, $K_i = 0.4166$ and $K_d = 0.2035$. Substi-

Table 2. Frequency response results obtained for different algorithms

Algorithm	Gain margin (dB)	Phase margin (deg.)	Bandwidth (Hz)
SMA (proposed)	Infinite	180	44.4579
HHO [5]	Infinite	180	38.5081
ASO [6]	Infinite	180	32.9113
GWO [7]	Infinite	180	14.9018

Table 3. PID parameters obtained with different algorithms for AVR system

Algorithm	K_p	K_i	K_d
SMA (proposed)	0.6173	0.4166	0.2035
SOS [14]	0.5693	0.4097	0.1750
LUS [15]	0.5878	0.4062	0.1843
MOL [16]	0.5857	0.4189	0.1772

Table 4. Frequency response results obtained for different algorithms

Algorithm	Gain margin (dB)	Phase margin (deg.)	Bandwidth (Hz)
SMA (proposed)	Infinite	180	7.1061
SOS [14]	Infinite	180	6.1894
LUS [15]	Infinite	180	6.5047
MOL [16]	Infinite	180	6.3391

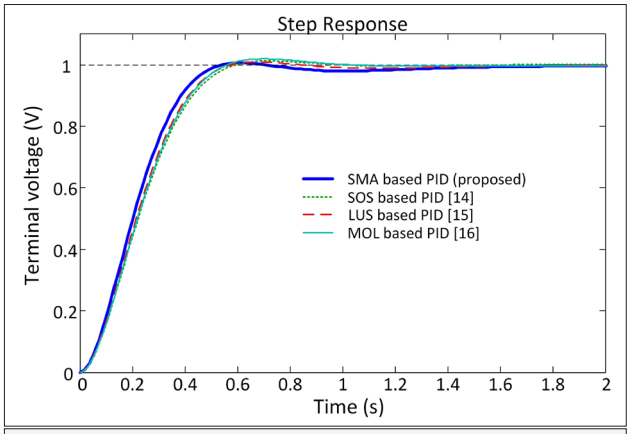


Figure 11. Terminal voltage comparison of the proposed tuning method and other methods used recently in the literature

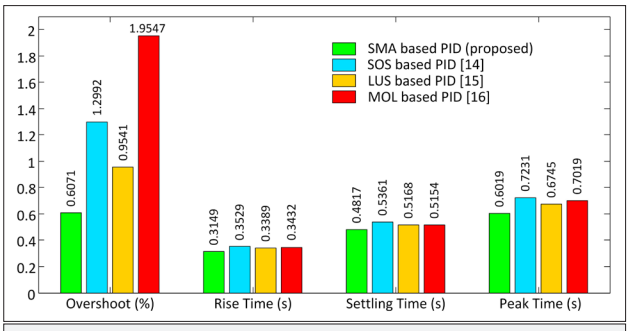


Figure 12. Comparison of maximum percentage overshoot, rise, settling, and peak times for various tuning methods

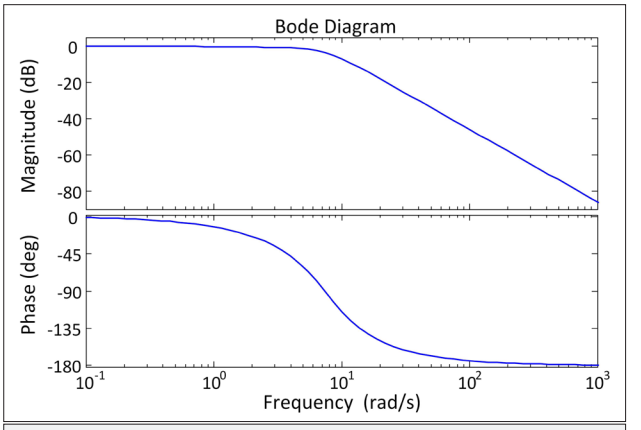


Figure 13. Bode diagram of AVR system with SMA based PID controller

tuting these gains in (13) provides the transfer function given in (16).

$$G_{AVR+PID+SMA}(s) = \frac{V_t(s)}{V_{ref}(s)} = \frac{0.02035s^3 + 2.097s^2 + 6.215s + 4.166}{0.0004s^5 + 0.0454s^4 + 0.555s^3 + 3.545s^2 + 7.173s + 4.166} \quad (16)$$

Table 3 contains the gain parameters obtained using other recent metaheuristic algorithms. The listed ones were the appropriate algorithms chosen for comparison. The AVR system's time responses with controllers based on various algorithms, including the proposed SMA, are illustrated in Figure 11. Comparative results based on overshoot, rise, settling, and peak times, are presented in Figure 12 since they are essential transient response criteria. From figures 11 and 12, the proposed SMA based PID controller perfectly improves the AVR system's transient response compared to SOS [14], LUS [15], and MOL [16] based controllers.

Figure 13 shows the AVR control system's magnitude and phase curves designed with the recommended approach. Moreover, relative frequency response results are listed in Table 4. From the table, the frequency response performance of the controlled AVR system using the SMA based PID controller is the best compared to the other approaches.

Conclusions

In this study, the SMA capability has been further assessed on two real-world engineering problems, such as regulating a DC motor's speed and controlling the output voltage of an AVR system by adopting a PID controller. The performances of the PID controlled DC motor and AVR systems were evaluated by transient and frequency responses. To validate the efficiency, the achieved performances were compared with the same systems tuned by recent and effective metaheuristic algorithms, such as SOS [14], LUS [15], and MOL [16] for AVR system and HHO [5], ASO [6], and GWO [7] for DC motor. The comparative results showed that SMA is an effective algorithm for solving the aforementioned real-world engineering problems.

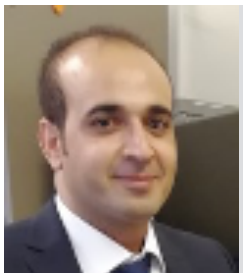
Peer-review: Externally peer-reviewed.

Conflict of Interest: The authors have no conflicts of interest to declare.

Financial Disclosure: The authors declared that the study has received no financial support.

References

1. J. Pongfai, X. Su, H. Zhang, W. Assawinchaichote, "A novel optimal PID controller autotuning design based on the SLP algorithm," Expert Syst., vol. 37, no. 2, p. e12489, 2020. [Crossref]
2. B. Hekimoğlu, S. Ekinci, "Optimally Designed PID Controller for a DC-DC Buck Converter via a Hybrid Whale Optimization Algorithm with Simulated Annealing," Electrica, vol. 20, no. 1, pp. 19–27, 2020. [Crossref]
3. A. K. Bhullar, R. Kaur, S. Sondhi, "Enhanced crow search algorithm for AVR optimization," Soft Comput., 2020. [Crossref]
4. G. Zhou, J. Li, Z. Tang, Q. Luo, Y. Zhou, "An improved spotted hyena optimizer for PID parameters in an AVR system," Math. Biosci. Eng., vol. 17, no. 4, pp. 3767–3783, 2020. [Crossref]
5. S. Ekinci, D. Izci, B. Hekimoğlu, "PID Speed Control of DC Motor Using Harris Hawks Optimization Algorithm," 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE), Istanbul, Turkey, pp. 1-6, 2020. [Crossref]
6. B. Hekimoğlu, "Optimal Tuning of Fractional Order PID Controller for DC Motor Speed Control via Chaotic Atom Search Optimization Algorithm," IEEE Access, vol. 7, pp. 38100–38114, 2019. [Crossref]
7. J. Agarwal, G. Parmar, R. Gupta, A. Sikander, "Analysis of grey wolf optimizer based fractional order PID controller in speed control of DC motor," Microsyst. Technol., vol. 24, no. 12, pp. 4997–5006, 2018. [Crossref]
8. Z. Qi, Q. Shi, H. Zhang, "Tuning of digital PID controllers using particle swarm optimization algorithm for a CAN-Based DC motor subject to stochastic delays," IEEE Trans. Ind. Electron., vol. 67, no. 7, pp. 5637–5646, 2020. [Crossref]
9. A. Rodríguez-Molina, M. G. Villarreal-Cervantes, M. Aldape-Pérez, "An adaptive control study for a DC motor using meta-heuristic algorithms," IFAC-PapersOnLine, vol. 50, no. 1, pp. 13114–13120, 2017. [Crossref]
10. D. Puangdownreong, "Fractional order PID controller design for DC motor speed control system via flower pollination algorithm," Trans. Electr. Eng. Electron. Commun., vol. 17, no. 1, pp. 14–23, 2019. [Crossref]
11. R. Bhatt, G. Parmar, R. Gupta, A. Sikander, "Application of stochastic fractal search in approximation and control of LTI systems," Microsyst. Technol., vol. 25, no. 1, pp. 105–114, 2019. [Crossref]
12. A. Mishra, N. Singh, S. Yadav, "Design of Optimal PID Controller for Varied System Using Teaching–Learning-Based Optimization," in Advances in Computing and Intelligent Systems, Springer, pp. 153–163, 2020. [Crossref]
13. A. Lotfy, M. Kaveh, M. R. Mosavi, A. R. Rahmati, "An enhanced fuzzy controller based on improved genetic algorithm for speed control of DC motors," Analog Integr. Circuits Signal Process., 2020. [Crossref]
14. E. Çelik, R. Durgut, "Performance enhancement of automatic voltage regulator by modified cost function and symbiotic organisms search algorithm," Eng. Sci. Technol. an Int. J., vol. 21, no. 5, pp. 1104–1111, 2018. [Crossref]
15. P. K. Mohanty, B. K. Sahu, S. Panda, "Tuning and assessment of proportional-integral-derivative controller for an automatic voltage regulator system employing local unimodal sampling algorithm," Electr. Power Components Syst., vol. 42, no. 9, pp. 959–969, 2014. [Crossref]
16. S. Panda, B. K. Sahu, P. K. Mohanty, "Design and performance analysis of PID controller for an automatic voltage regulator system using simplified particle swarm optimization," J. Franklin Inst., vol. 349, no. 8, pp. 2609–2625, 2012. [Crossref]
17. E. Celik, "Incorporation of stochastic fractal search algorithm into efficient design of PID controller for an automatic voltage regulator system," Neural Comput. Appl., vol. 30, no. 6, pp. 1991–2002, Sep. 2018. [Crossref]
18. S. Ekinci, B. Hekimoğlu, "Improved Kidney-Inspired Algorithm Approach for Tuning of PID Controller in AVR System," IEEE Access, vol. 7, pp. 39935–39947, 2019. [Crossref]
19. E. Kose, "Optimal Control of AVR System with Tree Seed Algorithm-Based PID Controller," IEEE Access, vol. 8, pp. 89457–89467, 2020. [Crossref]
20. M. Ćalasan, M. Micev, Ž. Djurovic, H. M. A. Mageed, "Artificial ecosystem-based optimization for optimal tuning of robust PID controllers in AVR systems with limited value of excitation voltage," Int. J. Electr. Eng. Educ., p. 0020720920940605, Jul. 2020. [Crossref]
21. N. Pachauri, "Water cycle algorithm-based PID controller for AVR," COMPEL - Int. J. Comput. Math. Electr. Electron. Eng., vol. 39, no. 3, pp. 551–567, 2020. [Crossref]
22. D. H. Wolpert, W. G. Macready, "No free lunch theorems for optimization," IEEE Trans. Evol. Comput., vol. 1, no. 1, pp. 67–82, 1997. [Crossref]
23. S. Li, H. Chen, M. Wang, A. A. Heidari, S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," Futur. Gener. Comput. Syst., vol. 111, pp. 300–323, 2020. [Crossref]
24. C. Kumar, T. D. Raj, M. Premkumar, T. D. Raj, "A New Stochastic Slime Mould Optimization Algorithm for the Estimation of Solar Photovoltaic Cell Parameters," Optik (Stuttg)., p. 165277, 2020. [Crossref]
25. D. Yousri, M. Abd Elaziz, D. Oliva, L. Abualigah, M. A. A. Al-qaness, and A. A. Ewees, "Reliable applied objective for identifying simple and detailed photovoltaic models using modern metaheuristics: Comparative study," Energy Convers. Manag., vol. 223, p. 113279, 2020. [Crossref]



Davut İzci received his BSc degree from Dicle University, Turkey, in Electrical and Electronic Engineering and his MSc and PhD degrees from Newcastle University, England - UK, in Mechatronics and Microsystems, respectively. He is currently an Assistant Professor working at Batman University, Turkey. His research interests are in microsystems development, sensing applications, robotics and instrumentation along with control systems and metaheuristic optimization techniques.



Serdar Ekinci was born in Diyarbakir, Turkey, in 1984. He received the B.S. degree in control engineering, and the M.S. and Ph.D. degrees in electrical engineering from Istanbul Technical University (ITU), in 2007, 2010, and 2015, respectively. Since 2016, he has been an Assistant Professor with the Computer Engineering Department, Batman University, Batman, Turkey. His areas of interests are electrical power systems, stability, control technology, and the applications of metaheuristic optimization to power system control.