Engineering Science and Technology, an International Journal 24 (2021) 331–342

HOSTED BY

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

journal homepage: www.elsevier.com/locate/jestch

Full Length Article

Opposition based Henry gas solubility optimization as a novel algorithm for PID control of DC motor

Serdar Ekinci [a,*], Baran Hekimoğlu [b], Davut Izci [c]

[a] Department of Computer Engineering, Batman University, Batman 72100, Turkey
[b] Department of Electrical & Electronics Engineering, Batman University, Batman 72100, Turkey
[c] Vocational School of Technical Sciences, Batman University, Batman 72060, Turkey

ARTICLE INFO

Article history:
Received 14 February 2020
Revised 22 June 2020
Accepted 18 August 2020
Available online 25 September 2020

Keywords:
Opposition-based learning
Henry gas solubility optimization algorithm
DC motor speed control
PID controller tuning

ABSTRACT

This paper proposes Henry gas solubility optimization with opposition-based learning (OBL/HGO) as a novel optimization approach for DC motor speed regulation. The proposed approach was used to obtain the best parameters (proportional, integral, derivative) of PID controller by minimizing the integral of time multiplied absolute error (ITAE) as the objective function. The optimized controller was used to regulate the speed of a DC motor. The analysis of statistical tests, convergence profile, performance index, robustness, and disturbance rejection along with transient and frequency responses were all conducted in order to validate the effectiveness of the proposed approach. Also, the performance of the proposed OBL/HGSO tuned PID (OBL/HGSO-PID) controller was not only compared with the PID controller tuned by the original HGSO algorithm but also with other controllers that were tuned by the state-of-the art meta-heuristic algorithms such as atom search optimization (ASO), stochastic fractal search (SFS), grey wolf optimization (GWO) and sine–cosine algorithm (SCA). The conducted simulation results and comparisons with the proposed HGSO-PID controller and other existing controllers have showed that the proposed OBL/HGO-PID controller has superior control performance and excellent robustness even under the conditions of both system uncertainties and load disturbances.

© 2020 Karabuk University. Publishing services by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

Direct current (DC) motor speed control has been an application area for many meta-heuristics algorithms as a real-world engineering application within the last decade since it provides an observable test bed for performance evaluations and comparisons. Basic requirements to demonstrate such a real-world engineering application are a suitable controller, an objective function and an optimization algorithm. The controller can be of any type such as PID [1], fractional order PID (FOPID) [2,3], neural network (NN) [4], fuzzy logic (FL) [5] or adaptive network-based fuzzy inference system (ANFIS) [6]. Likewise, the objective function can also be of any type such as integral of time multiplied absolute error (ITAE) [7], integral of time multiplied square error (ITSE) [8], and time domain performance indices based objective function (ZLG) [9]. However, various options are available even for the same controllers in terms of the optimization algorithms. For example, artificial bee colony (ABC) [10], atom search (ASO) and its chaotic version (ChASO) [3], bat (BA) [11], improved beetle antennae search (BAS) [12], differential evolution (DE) [11,13], improved DE [14], firefly (FFA) [11], flower pollination (FPA) [15], genetic (GA) [16], gravitational search (GSA) [17], grey wolf (GWO) [2,18,19], invasive weed (IWO) [20], Jaya (JOA) [21], kidney (KA) [8], particle swarm (PSO) [1,11,22], constrained PSO (CPSO) [23], dynamic PSO (dPSO) [24], salp swarm (SSA) [25], sine–cosine (SCA) [26], improved SCA (ISCA) [27], stochastic fractal search (SFS) [7,28], and wolf search (WSA) [11] are optimization algorithms that have been proposed to tune the controller parameters of a DC motor for speed control. Further examples of the intelligent methods to achieve optimal parameters in different systems can be found in [29–31].

The common goal of those listed studies was to utilize an optimally designed controller via an optimization algorithm in order to minimize the transient and steady state criteria such as maximum overshoot, settling time, rise time and steady state error, consequently, to achieve an improved speed control performance of a DC motor. It is important to acquire the best gain parameters of a controller to achieve the stated goal, however, algorithms proposed for improving the performance may not find the optimal controller gains due to weaknesses they present. For example, ABC and PSO have great promise to address many optimization

* Corresponding author.
E-mail address: serdar.ekinci@batman.edu.tr (S. Ekinci).
Peer review under responsibility of Karabuk University.

https://doi.org/10.1016/j.jestch.2020.08.011
2215-0986/© 2020 Karabuk University. Publishing services by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

problems, yet, they are problematic in memory capacity and computational load. Achieving better results with other optimization approaches is feasible, however, early convergence, local minimum stagnation, difficulty in parameter selection, and increased computational time can cause issues in those algorithms [3,32]. One of the examples is GWO [33] which was proposed as a competitive optimizer for global optimization problems, however, the performance and the convergence are greatly affected from various parameters. For example, it cannot converge to optimal solution [34] if the initial population is not chosen conveniently. Another example is a recently developed algorithm, known as SCA [35], to solve optimization problems. Its exploitation is quite poor although it is rich in terms of exploration which causes an imbalance between exploration and exploitation [36]. Similar issues can be observed for SFS [37] and ASO [38] algorithms. The SFS has a premature convergence and stagnation problem, similar to other evolutionary algorithms, due to its evolutionary and stochastic nature [39] although it has been proved to be a promising algorithm compared to other evolutionary counterparts. Similarly, ASO suffers from local minima stagnation and slow convergence rate [40] despite its simple structure and easier implementation. Moreover, there is no precise algorithm capable of optimal design although the current meta-heuristic tuning techniques have made significant contributions to the design of the controller for speed regulation. Therefore, testing the performance of new algorithms on speed regulation and comparing it with previous studies through various analyzes is important to see if a better option can be achieved.

Not all the meta-heuristic algorithms are superior in terms of optimization capability due to issues affecting their performance and accuracy. The opposition-based learning (OBL) was introduced [41] to avoid such issues. Basically, the opposite position of a candidate solution is created via OBL. This allows simultaneous two-way exploration of the search area. Considering the objective function values, OBL selects the best components from a set that is accepted by candidate solutions and opposing solutions. The OBL rule can either be used at the start of a meta-heuristic algorithm or at any part of a solution that is modified within a meta-heuristic algorithm. By using OBL, different meta-heuristic algorithms have been improved and yielded better results than their standard versions. There are varieties of examples showing better performance parameters whilst integrated with OBL. Those examples can be listed as; increasing the convergence rate of SCA using the opposition-based sine cosine algorithm (OBSCA) [42], avoiding stagnation in local minima using modified SCA (m-SCA) [43], and improving grasshopper optimization (GOA) [44] along with crow search (CSA) [45] algorithms. The OBL combination has also been extended for engineering applications where it was used alongside with whale optimization algorithm (OBWOA) [46] and SCA [47] for parameter estimation of solar cell diode models. It was also used with GWO (OBGWO) to study the economic load distribution problem [48]. Another application of this mechanism was its application to hybrid coral reefs optimization (OHCRO) in order to determine the parameters of an adaptive infinite impulse response (IIR) filter [49]. The latter is a system identification example of the OBL mechanism. Feature selection with social spider optimization (SSO) [50] and SSA [51] algorithms are other examples of implementation of OBL mechanism. In addition, opposition-based dragonfly algorithm (OBDA) has recently been introduced as an alternative approach for various engineering problems [52]. All these studies show that OBL is a capable mechanism in order to achieve better results in optimization problems. This is also the motivation of this paper, which proposes the opposition-based learning with Henry gas solubility optimization (OBL/HGSO) algorithm for speed control of a DC motor.

HGSO algorithm is a recently proposed physics-inspired population-based algorithm, mimicking gas behavior governed by Henry's law that was successfully tested on several benchmark functions, test suites and three real-world optimization problems [53]. It was also used for solving motif discovery problem [54]. The HGSO has also been used in automotive industry to optimize the shape of a brake pedal [55] and it was proved to be effective in such a real-world optimization problem. HGSO has found an application area in the field of machine learning by being adopted in support vector regression (SVR) [56]. It was used to update the parameters, thus, helped the proposed approach to reach optimum performance. The classification is another application area of HGSO where it was used for feature selection and enhancement of classification accuracy [57]. Further example of HGSO can be found in controller design where it was used to find optimum parameters of a fractional order proportional–integral–derivative (FOPID) controller in an automatic voltage regulator (AVR) system [58]. The aim of this paper is to introduce a modified version of HGSO which is named as opposition-based learning Henry gas solubility optimization (OBL/HGSO). The use of OBL in combination with the optimization features of HGSO improves the accuracy and performance of the standard HGSO considerably. Therefore, the disadvantages of the standard HGSO is eliminated with such improvement whilst preserving the good optimization capabilities. To the best of authors' knowledge, this paper is the first attempt to propose HGSO and OBL/HGSO algorithms for optimal design of a PID controller in a DC motor speed control system. The specific contributions of this study can be summarized as follows:

- The OBL-based HGSO (OBL/HGSO) algorithm was proposed for the first time.
- HGSO and OBL/HGSO algorithms were used for the first time ever in PID design for speed regulation of DC motor. In addition, this is the first application of OBL/HGSO and HGSO algorithms in electrical engineering field.
- The efficiencies of optimally designed HGSO-PID and OBL/HGSO-PID controllers for improving the performance of DC motor were verified in terms of statistical criteria and convergence profile.
- The performance of the proposed HGSO-PID and OBL/HGSO-PID approaches were compared in detail with the ASO-PID [3], SFS-PID [28], GWO-PID [2], and SCA-PID [26] through various analyzes since those controllers with specified algorithms are the most recent approaches that have found the best controller gains.
- The comparative results of transient and frequency responses, robustness and load disturbance rejection analyzes were all clearly confirmed the effectiveness of the proposed opposition-based design, namely the OBL/HGSO-PID controller, and its superiority over other algorithms.

2. HGSO algorithm

Henry's law was formulated in 1800s [59] as a gas law and explains the state of a gas which dissolves in a liquid at a constant temperature. Basically, this law describes the relationship between gas and liquid in terms of the dissolving property of gas and states that the amount of any gas which dissolves in any liquid with any volume is proportional to partial pressure of the given gas and liquid in an equilibrium state. Therefore, temperature is an important parameter that Henry's law depends on [60]. According to Henry's law, the relationship between the solubility and the partial pressure of a gas is expressed as:

$$S_g = H \times P_g \tag{1}$$

where S_g, P_g, and H are gas solubility, partial pressure and Henry's constant, respectively. The Henry's constant is a specific value (at each specific temperature) for each gas-solvent mixture.

The temperature variations of any gas–liquid system cause a change of Henry's constant since this law is highly dependent on temperature and it requires to be considered. This change can be characterized with the help of Eq. (2):

$$\frac{d\ln H}{d(1/T)} = \frac{-\nabla_{sol}E}{R} \tag{2}$$

In Eq. (2), $\nabla_{sol}E$, R and T represent the dissolution enthalpy, the constant of gas, and the temperature dependence, respectively. Accommodating Eq. (1) can be as:

$$H(T) = e^{B/T} \times A \tag{3}$$

where A and B are parameters of T. An alternative expression for temperature of *298.15 K* can be obtained as:

$$H(T) = H^{\theta} \times e^{(-\frac{\nabla_{sol}E}{R}(1/T-1/T^{\theta}))} \tag{4}$$

The Eq. (2) has credibility for constant $\nabla_{sol}E$ values which helps reformulation of Eq. (4) as:

$$H(T) = e^{(-C\times(1/T-1/T^{\theta}))} \times H^{\theta} \tag{5}$$

The Eqs. (1) - (5) can be used to determine the gas solubility in liquid medium and emphasizes that the pressure and the temperature are two important parameters that has effect on solubility. This is the fundamental behavior of Henry's law and it has inspired the HGSO [53]. The HGSO is considered as a global optimization algorithm since it has exploration and exploitation stages. The mathematical model for HGSO was reported in 8 steps [53]. The following equation provides the initialization process (Step 1) for the number (population size N) and positions of gases where t is iteration time and r represents a random number that is between 0 and 1:

$$X_i(t+1) = X_{min} + r \times (X_{max} - X_{min}) \tag{6}$$

The bounds of the problem are denoted by X_{max} and X_{min}. X_i is the notation for the position of ith gas in population N. Values for Henry's constant of type $j(H_j(t))$, the number of gas i, partial pressure of $P_{i,j}$ of gas i in cluster j, and $\nabla_{sol}E/R$ constant value of $j(C_i)$ are initialized with the Eq. (7) where l_1, l_2, l_3 are constants with values of 5×10^{-2}, 100, and 10^{-2}, respectively:

$$\begin{aligned} H_j(t) &= l_1 \times rand(0,1) \\ P_{i,j} &= l_2 \times rand(0,1) \\ C_j &= l_3 \times rand(0,1) \end{aligned} \tag{7}$$

The second step is the clustering step where the population agents are being divided to the number of gas types to obtain equal clusters. Henry's constant value of H_j is the same for each specific cluster since similar gases forms each of them. The evaluation process is performed in third step where the best gas, reaching the highest equilibrium, in each cluster j is identified. In order to obtain the optimal gas, a ranking stage is also performed in the entire swarm as part of this step. Updating the Henry's coefficient is performed in the fourth step using Eq (8):

$$\begin{aligned} H_j(t+1) &= H_j(t) \times e^{(-C_j\times(1/T(t)-1/T^{\theta}))} \\ T(t) &= e^{(-t/iter)} \end{aligned} \tag{8}$$

In Eq. (8), the temperature, the Henry's coefficient for cluster j, and the total number of iterations are denoted by T, H_j, and *iter*, respectively. T^{θ} is a constant that equals to 298.15. In step 5, updating the solubility is performed using the following equation:

$$S_{i,j}(t) = K \times H_j(t+1) \times P_{i,j}(t) \tag{9}$$

where K is a constant, $P_{i,j}$ and $S_{i,j}$ are the partial pressure and the solubility of gas i in cluster j. The position update is carried out in step 6 as demonstrated by Eq. (10) where $X_{i,j}$ is the gas position in the cluster, r is a random constant, t is the iteration time, $X_{i,best}$ is the best gas in the cluster (gas i, cluster j):

$$\begin{aligned} X_{i,j}(t+1) &= X_{i,j}(t) + F \times r \times \gamma \times \left(X_{i,best}(t) - X_{i,j}(t)\right) \\ &+ F \times r \times \alpha \times \left(S_{i,j}(t) \times X_{best}(t) - X_{i,j}(t)\right) \\ \gamma &= \beta \times e^{-\left(\frac{F_{best}(t)+\varepsilon}{F_{i,j}(t)+\varepsilon}\right)}, \ \varepsilon = 0.05 \end{aligned} \tag{10}$$

The best gas in the swarm is denoted by X_{best}. In addition, γ represents the ability of gas (gas j, cluster i), α is the influence of other gas i in cluster j, and β is a constant. $F_{i,j}$ and F_{best} parameters are the fitness of of gas i in cluster j and the best gas in the swarm. In step 7, number of worst agents (N_w) are ranked and selected with the help of Eq. (11) in order to escape from local optimum.

$$\begin{aligned} &N_w = N \times (rand(c_2 - c_1) + c_1), \\ &c_1 = 0.1 \ and \ c_2 = 0.2 \end{aligned} \tag{11}$$

In final step (Step 8), the position of the worst agents is updated as follows:

$$G_{i,j} = G_{\min(i,j)} + r \times (G_{\max(i,j)} - G_{\min(i,j)}) \tag{12}$$

In Eq. (12), $G_{i,j}$ represents the position of gas i in cluster j, and r is a random number. G_{min} and G_{max} are the bounds of the problem. Fig. 1 illustrates the principle of Henry gas solubility. As can be seen, the volume of the gas, which is in equilibrium, decreases whilst the pressure is increased.

3. Proposed OBL/HGSO algorithm

The basic HGSO has few drawbacks such as having slow convergence and being stuck in a local best solution. These drawbacks arise from updating some solutions towards the local best solution although there are better and available but far away solutions which HGSO cannot manage to find. Therefore, the solutions in the opposite directions are required to be considered in order to avoid those drawbacks. The latter requirement is achieved with the algorithm proposed in this paper, which uses OBL alongside the HGSO. The proposed OBL/HGSO improves the exploration efficiency of the search space by combining the search capabilities of original HGSO with OBL. Compared to similar algorithms the proposed approach requires fewer parameters to set. Besides, the configuration of HGSO algorithm is not affected by inclusion of the OBL, on the contrary, it helps increasing the accuracy of the best

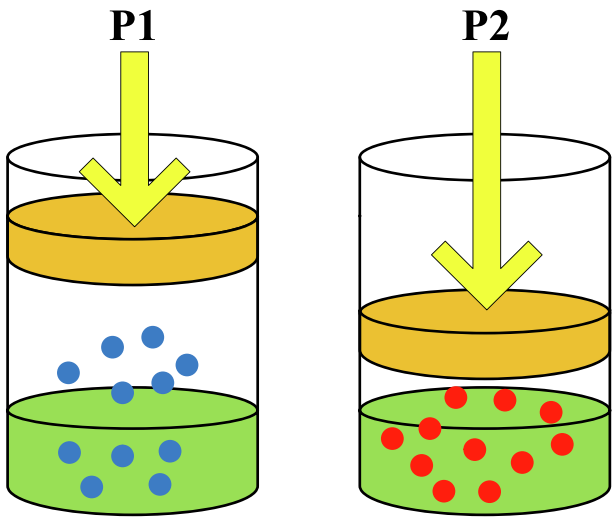


Fig. 1. The principle of Henry gas solubility [53].

solution. Thus, reducing the size of the initial population, required for improvement in terms of converging to the optimal solution, is possible since a broader search space can be explored by OBL/HGSO.

The proposed algorithm helps improving the HGSO in two steps. Firstly, the OBL is used to initialize the population by speeding up the convergence and exploring all the search space for all solutions in order to avoid stagnating in a local best solution. In addition, it is also used for controlling the updated solutions in the opposite direction and comparing those solutions with the current one in order to check if there is another best solution. By doing so, it avoids from local best minimum. The following subsections explain the stages utilized for the proposed algorithm.

3.1. Initialization stage

The proposed method starts with initializing a random population of X which has a size of N such that the position vector for the initial solution is described as $x_i = [x_{i1}, x_{i2}, \cdots, x_{in}]$ where $i = 1, 2, \cdots, N$. The OBL is then used to calculate the solutions in the opposite directions of each solutions and generates an opposite population of $\overline{X}$. Using both populations of X and $\overline{X}$, the best N number of solutions are chosen. The steps in this stage are as follows:

- Randomly start the solutions for population of X.
- Calculate the opposite population of $\overline{X}$ as:

$\overline{x}_{ij} = u_i + l_i - x_{ij}$ where $i = 1, 2, \cdots, N$ and $j = 1, 2, \cdots, n$. In here, l and u are the lower and upper limits for the search space, respectively. x_{ij} and $\overline{x}_{ij}$ represent the jth solution of the position i for the population of X and the corresponding opposite population of $\overline{X}$, respectively.

- Choose N number of best solutions from the union of $X \cup \overline{X}$ in order to form a new population.

3.2. Updating stage

The best solution (x_p) is determined (from the first stage) after choosing the best N solutions. The agents in population of X are updated using HGSO algorithm and the fitness functions are calculated. In addition, the opposite population of $\overline{X}$ is calculated according to OBL and the fitness functions for each $\overline{x}$ are determined. The next step in OBL/HGSO is to choose N number of best solutions from the union of both populations $(X \cup \overline{X})$. All the steps are repeating until stopping condition is reached. The flowchart of proposed OBL/HGSO algorithm is shown in Fig. 2.

4. DC motor speed control with PID

The speed regulation was controlled via the armature voltage. The closed loop of the system using PID controller is demonstrated in Fig. 3. The PID transfer function is given in Eq. (13) where K_p, K_i, and K_d are gains of PID (proportional, integral and derivative, respectively). ω and ω_{ref} are actual motor speed and reference speed, respectively. R_a is armature resistance equal to 0.4Ω, L_a is armature inductance equal to 2.7H, J is inertia torque of motor equal to $0.0004\text{kg}\cdot\text{m}^2$, B is motor friction constant equal to $0.0022\text{N}\cdot\text{m}\cdot\text{s/rad}$, K_m is motor torque constant equal to $0.015\text{N}\cdot\text{m/A}$, and K_b is electromotive force constant equal to $0.05\text{V}\cdot\text{s}$.

$$G_{PID}(s) = K_p + \frac{K_i}{s} + K_d s \tag{13}$$

Open-loop transfer function of DC motor is obtained as follows. The induced voltage $e_b(t)$ for a constant flux is proportional to the angular velocity $\omega(t) = d\theta/dt$ and given as:

$$e_b(t) = K_b \frac{d\theta(t)}{dt} = K_b \omega(t) \tag{14}$$

The armature voltage $e_a(t)$ is used to govern the speed of an armature-controlled DC servo motor. The armature circuit's differential equation is given in Eq. (15).

$$e_a(t) = L_a \frac{di_a(t)}{dt} + R_a i_a(t) + e_b(t) \tag{15}$$

Assuming the load torque is zero, a corresponding torque (sum of inertia and friction torques) is produced by the armature current as provided in Eq. (16).

$$T(t) = J\frac{d\omega(t)}{dt} + B\omega(t) = K_m i_a(t) \tag{16}$$

The Laplace transforms of Eqs. (14) - (16) with zero initial conditions are given as follows:

$$E_b(s) = K_b \Omega(s) \tag{17}$$

$$E_a(s) = (L_a s + R_a) I_a(s) + E_b(s) \tag{18}$$

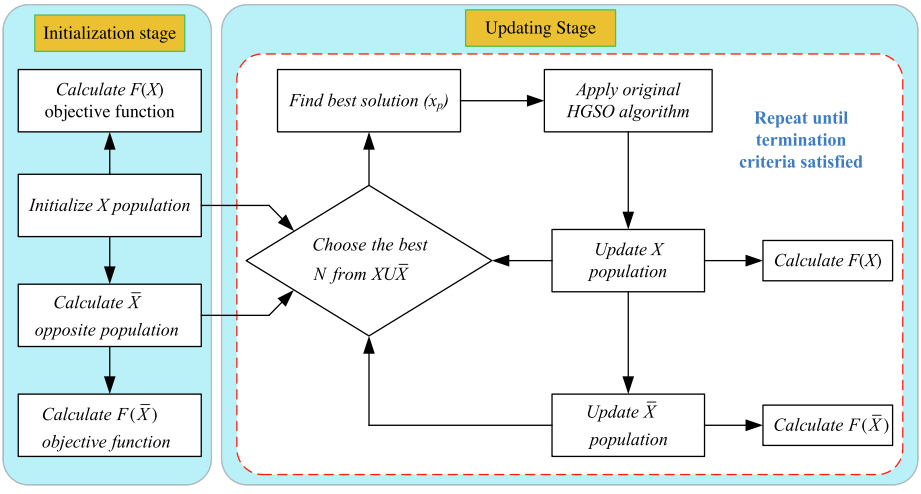


Fig. 2. The proposed OBL/HGSO algorithm.

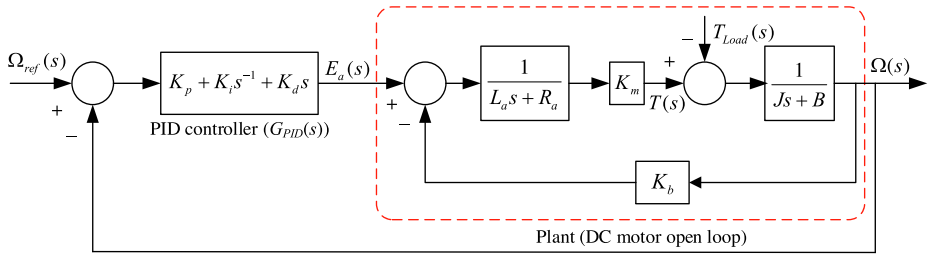


Fig. 3. A PID controlled DC motor speed system.

$$T(s) = (Js + B)\Omega(s) = K_m I_a(s) \tag{19}$$

Hence, the open-loop transfer function describing the relationship between input voltage and the output speed of DC motor can be written as in Eq. (20).

$$\frac{\Omega(s)}{E_a(s)} = \frac{K_m}{(L_a s + R_a)(Js + B) + K_b K_m},\ for\ T_{Load} = 0 \tag{20}$$

Furthermore, the relationship between motor speed (ω) and load torque (T_{Load}) when the input voltage (E_a) is zero can also be given by the following transfer function.

$$\frac{\Omega(s)}{T_{Load}(s)} = -\frac{(L_a s + R_a)}{(L_a s + R_a)(Js + B) + K_b K_m},\ for\ E_a = 0 \tag{21}$$

Substituting DC motor parameter values in Eqs. (20) and (21), the following open-loop transfer function is obtained:

$$G_{open-loop}(s) = \begin{cases} \frac{\Omega(s)}{E_a(s)} = \frac{15}{1.08s^2+6.1s+1.63},\ for\ T_{Load} = 0 \\ \frac{\Omega(s)}{T_{Load}(s)} = -\frac{2700s+400}{1.08s^2+6.1s+1.63},\ for\ E_a = 0 \end{cases} \tag{22}$$

Finally, DC motor closed-loop transfer function ($T_{closed-loop}(s)$) with a PID speed controller is given in Eq. (23).

$$= \begin{cases} \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{15\left(K_d s^2+K_p s+K_i\right)}{1.08s^3+6.1s^2+1.63s+15\left(K_d s^2+K_p s+K_i\right)},\ for\ T_{Load} = 0 \\ \frac{\Omega(s)}{T_{Load}(s)} = -\frac{(2700s+400)s}{1.08s^3+6.1s^2+1.63s+15\left(K_d s^2+K_p s+K_i\right)},\ for\ \Omega_{ref} = 0 \end{cases} \tag{23}$$

5. Optimization problem and application of OBL/HGSO algorithm

The ITAE function was adopted in this study for a fair comparison with refs [2,3,26,28]. The ITAE objective function is given as presented in Eq. (24). The proposed OBL/HGSO algorithm's parameters are listed in Table 1 and the block diagram of OBL/HGSO-based PID (OBL/HGSO-PID) controller for DC motor is shown in Fig. 4.

$$ITAE = \int_0^{tsim} t \cdot |e(t)| dt \tag{24}$$

Table 1
Parameters of OBL/HGSO algorithm for solving optimization problem.

Parameter	Value
Number of gas particles	40
Iteration number	50
Independent run number	20
$[I_1; I_2; I_3]$	[0.05; 100; 0.01]
$[c_1; c_2; \beta; \alpha; K]$	[0.1; 0.2; 1; 1; 1]
Lower bound for$[K_p; K_i; K_d]$	[0.001; 0.001; 0.001]
Upper bound for$[K_p; K_i; K_d]$	[20; 20; 20]
Dimension for optimization problem	3
Simulation time (t_{sim})	1 *s*

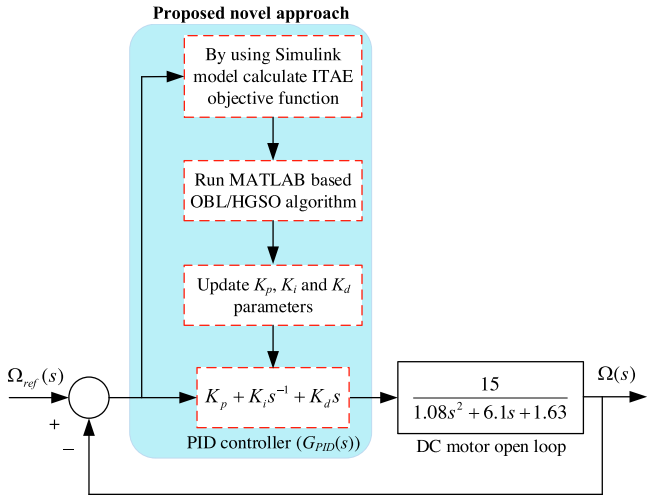


Fig. 4. The proposed OBL/HGSO-PID approach for DC motor speed control.

The procedure of finding the optimal gains of PID controller with the OBL/HGSO algorithm started with initialization stage where a developed MATLAB/Simulink model for DC motor speed control was integrated with OBL/HGSO. PID controller gains, which required to be optimized, were assigned to a vector of real numbers such as $P = (K_p; K_i; K_d)$ representing each gas particle in the population. The population was consisted of randomly generated N number of gas particles and their associated oppositions. Then, the time domain simulation of the speed control system of the DC motor with the proposed PID controller and unity feedback were carried out for each gas particle and the speed response curves of the system were obtained together with the ITAE value.

It is typical to obtain different speed output curves and ITAE values with different particles. Therefore, N number of solutions (gas particles) with the best ITAE values was chosen to be updated for the next iteration before returning to the HGSO algorithm. During the latter process, the oppositions were generated whilst each gas particle was being updated. This process was a bidirectional flow between control system and OBL/HGSO algorithm. This process was maintained until reaching maximum iteration. Finally, the best gas particle with the lowest ITAE value was displayed as the optimal PID controller gain set at the end of optimization process. The detailed flow chart of the proposed design procedure is shown in Fig. 5.

6. Comparative simulation results and discussion

The programming codes of the OBL/HGSO algorithm were run on MATLAB/Simulink software package that was installed on a computer with Intel® i7 2.50 GHz processor and 16.00 GB RAM. The simulations of the transient response, frequency response, robustness and load disturbance rejection analyses were also

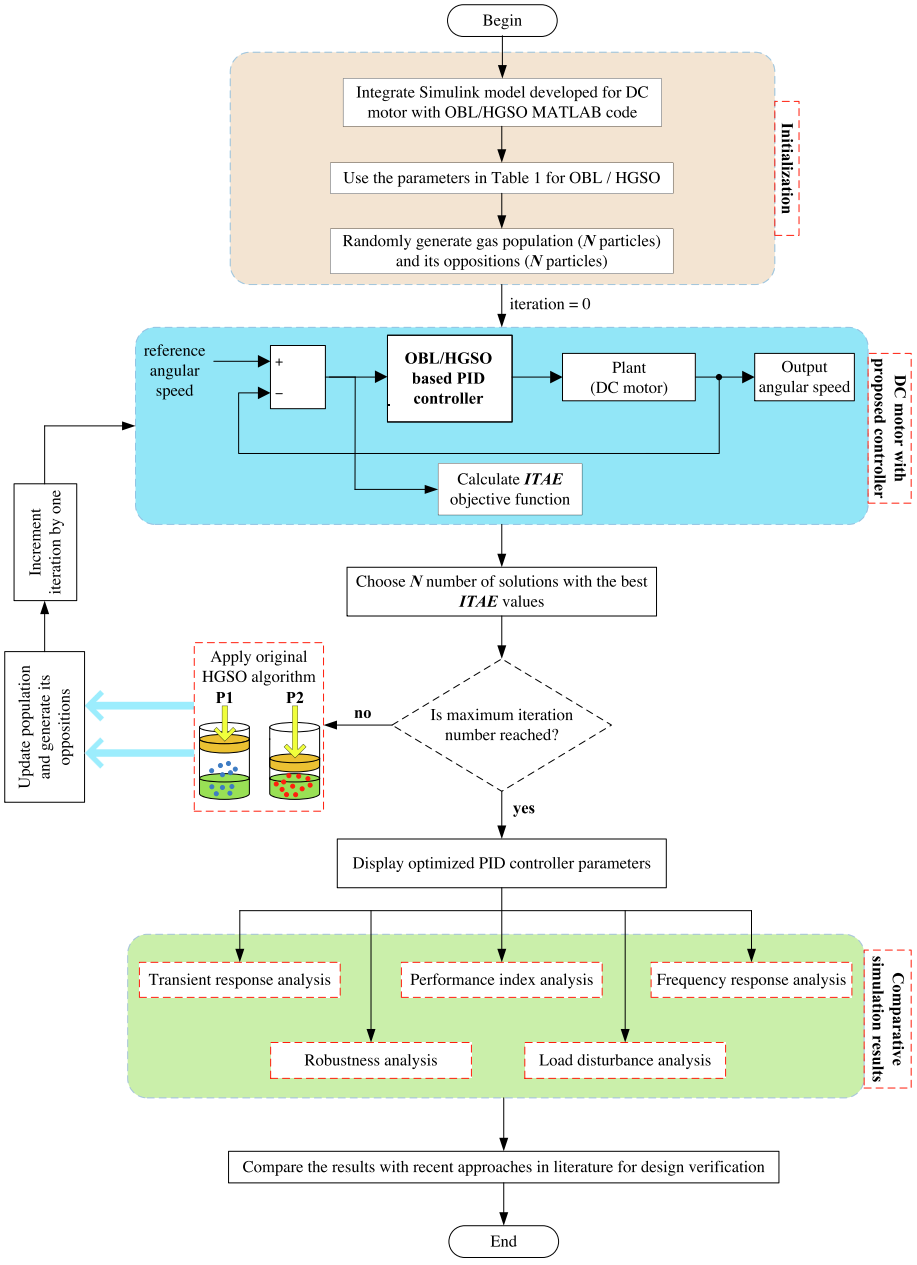


Fig. 5. Flowchart of the proposed design procedure for DC motor speed control.

carried out using the stated environment. The simulation results of various analyses obtained by using the optimal gain set were compared with other recent approaches for the verification of the design as explained in the following sub-sections.

6.1. Statistical test analysis

HGSO and the proposed OBL/HGSO algorithms were run independently for 20 times. The obtained values of ITAE objective function for all runs are shown in Fig. 6. The statistical values of objective function, such as worst, best, average, median and standard deviation values are given in Table 2 and the boxplot is given in Fig. 7. From both figures and the table, it can be concluded that OBL/HGSO algorithm has quite good statistical performance such that even the worst value obtained by OBL/HGSO algorithm is well below the best value obtained by the basic HGSO algorithm.

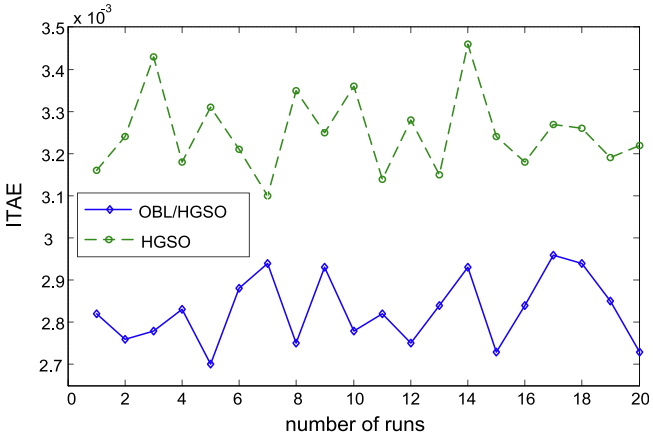


Fig. 6. The obtained ITAE values from all runs.

Table 2
Statistical values of ITAE objective function.

Statistical index	HGSO	OBL/HGSO
Worst (Maximum)	0.0035	0.0030
Best (Minimum)	0.0031	0.0027
Average	0.0032	0.0028
Median	0.0032	0.0028
Standard deviation	9.5250E-05	8.0760E-05

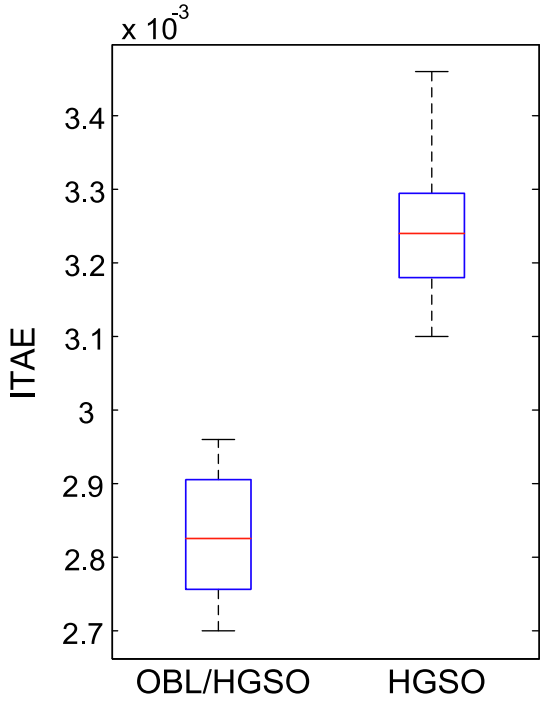


Fig. 7. Boxplot for HGSO and OBL/HGSO.

6.2. Convergence profile

The convergences of the best run of HGSO and the proposed OBL/HGSO algorithms are demonstrated in Fig. 8. From this figure, it can be concluded that OBL/HGSO algorithm can provide a lowest ITAE value with a high convergence speed since it took only 13 iterations to find the optimal values for the best run. The parameters of HGSO-PID and OBL/HGSO-PID controllers, found after the optimization process, are given in Table 3. The obtained transfer functions using these gains are provided in Eqs. (25) - (26).

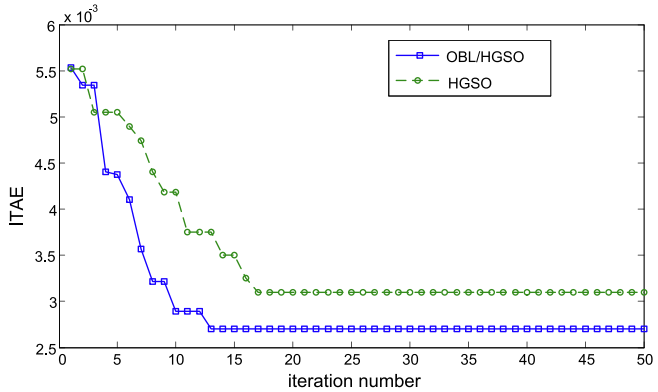


Fig. 8. Comparative convergence curves.

Table 3
Gain parameters of the proposed controllers and other compared controllers.

Controller type	K_p	K_i	K_d
OBL/HGSO-PID (proposed)	16.9327	0.9508	2.8512
HGSO-PID (proposed)	13.4430	1.2059	2.2707
ASO-PID [3]	11.9437	2.0521	2.4358
SFS-PID [28]	1.6315	0.2798	0.2395
GWO-PID [2]	6.8984	0.5626	0.9293
SCA-PID [26]	4.5012	0.5260	0.5302

$$T_{OBL/HGSO-PID}(s) = \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{42.77s^2 + 254s + 14.26}{1.08s^3 + 48.87s^2 + 255.6s + 14.26} \quad (25)$$

$$T_{HGSO-PID}(s) = \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{34.06s^2 + 201.6s + 18.09}{1.08s^3 + 40.16s^2 + 203.3s + 18.09} \quad (26)$$

6.3. Comparison of overshoot, rise time and settling time

The gain parameters for other controllers that were chosen for comparison purposes are listed in Table 3. The transfer functions of these controllers are provided in Eqs. (27) - (30). The comparative step response results and transient response analysis results for speed control system designed by different approaches are presented in Figs. 9 – 12 where Fig. 9 presents the step response comparisons, Figs. 10, 11, and 12 present bar plot comparisons of maximum percentage overshoot, rise time (for 10%→90%), and settling time (for ± 2% tolerance), respectively.

$$T_{ASO-PID}(s) = \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{36.54s^2 + 179.2s + 30.78}{1.08s^3 + 42.64s^2 + 180.8s + 30.78} \quad (27)$$

$$T_{SFS-PID}(s) = \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{3.592s^2 + 24.47s + 4.197}{1.08s^3 + 9.693s^2 + 26.1s + 4.197} \quad (28)$$

$$T_{GWO-PID}(s) = \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{13.94s^2 + 103.5s + 8.439}{1.08s^3 + 20.04s^2 + 105.1s + 8.439} \quad (29)$$

$$T_{SCA-PID}(s) = \frac{\Omega(s)}{\Omega_{ref}(s)} = \frac{7.953s^2 + 67.52s + 7.89}{1.08s^3 + 14.05s^2 + 69.15s + 7.89} \quad (30)$$

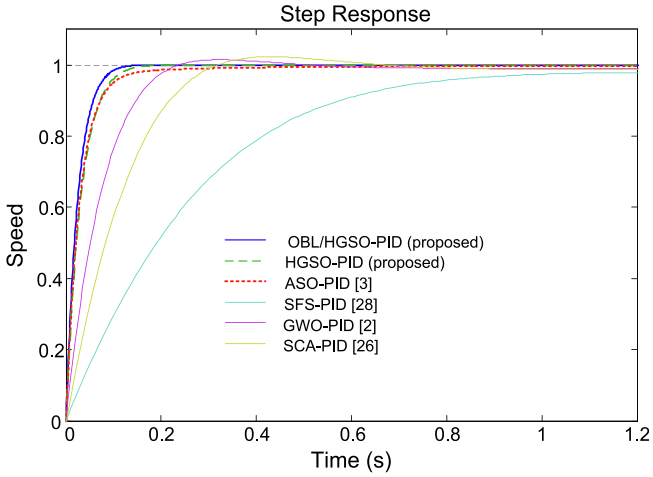


Fig. 9. Speed comparison with existing approaches.

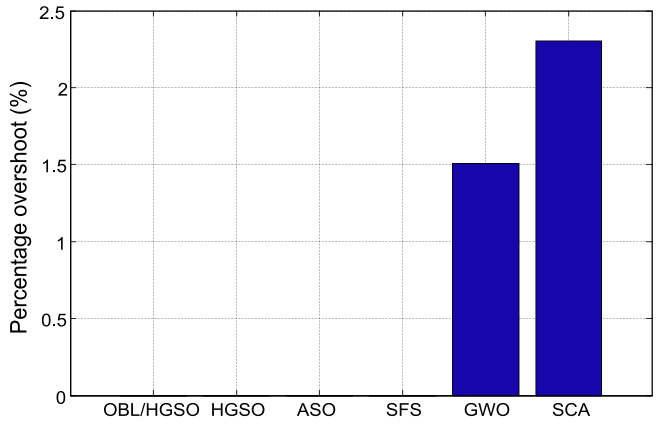


Fig. 10. Percentage overshoots for different approaches.

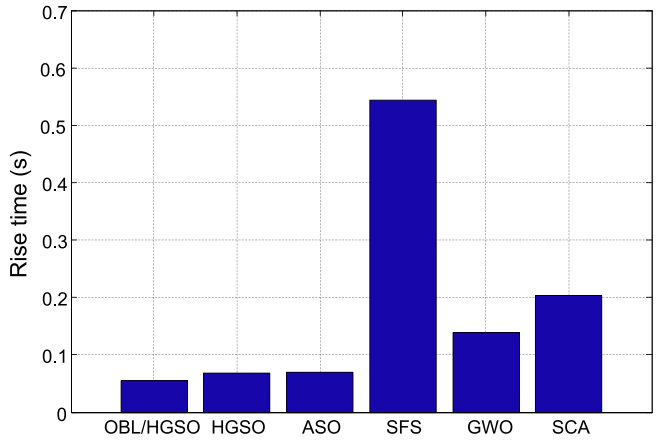


Fig. 11. Rise times for different approaches.

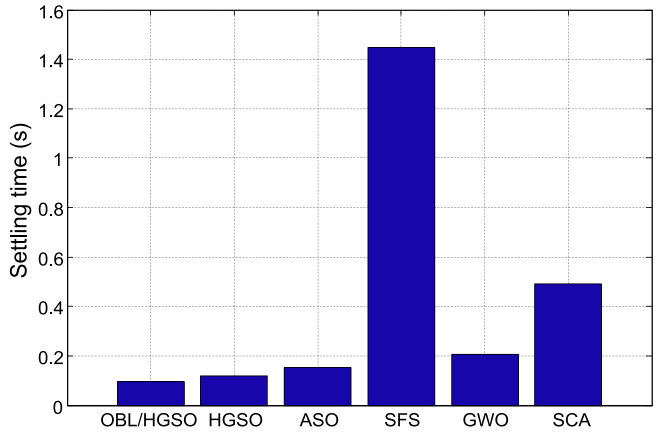


Fig. 12. Settling times for different approaches.

As seen from these figures, OBL/HGSO-PID controller clearly has better time response than others including HGSO-PID controller, as well. Thus, the proposed controller design approach is superior not only to HGSO based design approach but also to other controller design approaches such as ASO [3], SFS [28], GWO [2] and SCA [26], with better transient stability, fast damping characteristics and minimum overshoot.

6.4. Comparison of performance index

Due to its widespread use, W objective function was also adopted for performance index comparison. The formula of the W is given in Eq. (31) [9] where E_{ss},M_p, T_r and T_s stand for steady-state error, maximum overshoot, rise and settling times, respectively. The weighting parameter ρ is usually chosen as 1.0 [61]. The performance of a system is maximum for the lowest value of W in terms of transient response.

$$W(K_p, K_i, K_d) = (1 - e^{-\rho})(M_p + E_{ss}) + e^{-\rho}(T_s - T_r) \tag{31}$$

The comparative bar plots of W performance index values obtained from different approaches is given in Fig. 13. As seen from the figure, the proposed OBL/HGSO-PID controller has the lowest W value. This result shows the superiority of OBL/HGSO over other approaches including the original HGSO.

6.5. Comparison of frequency domain analyses

The comparative Bode plots of DC motor speed control system with different controller designs are shown in Fig. 14 and the comparative frequency response performance analysis results such as gain margin (in decibel), phase margin (in degrees), and bandwidth (in Hertz) are presented in Table 4. The table clearly demonstrates that the most stable system in terms of frequency response criteria is the proposed OBL/HGSO-PID controller.

6.6. Comparison of robustness analysis

A robust controller is desired in order to maintain the system response in acceptable ranges. Therefore, robustness analysis was performed to determine how stable the proposed system is, in case of uncertainties. To do so, the system behavior was observed by separately changing the electrical resistance (R_a) and torque constant (K_m) of DC motor with ±25% and ±20%, respectively. Thus, this action created four possible operating scenarios. Those scenarios are presented in Table 5.

The comparative simulation results of transient response analysis for all scenarios are shown in Tables 6–9. Likewise, the comparative speed step response plots are depicted in Figs. 15–18. Despite the changes occurring in DC motor parameters, the proposed OBL/HGSO-PID controller had the least values for both settling and rise times with no overshoot, except the Scenario I and II. The latter two scenarios only presented an insignificant overshoot whilst compared to other controllers. The robustness of the proposed OBL/HGSO-PID controller was confirmed via these results in terms of regulating the speed of a DC motor.

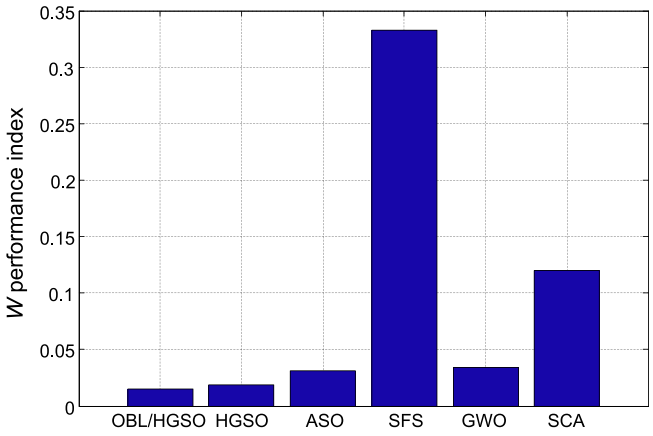


Fig. 13. W performance index values for different approaches.

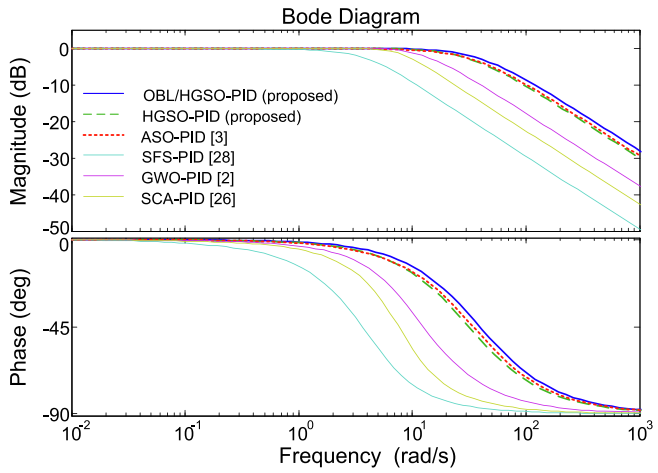


Fig. 14. Comparative Bode plots for different controller designs.

Table 4
Comparative frequency response performance analysis results.

Controller type	Gain margin (dB)	Phase margin (deg.)	Bandwidth (Hz)
OBL/HGSO-PID (proposed)	∞	180°	**39.8561**
HGSO-PID (proposed)	∞	180°	31.7975
ASO-PID [3]	∞	180°	32.9113
SFS-PID [28]	∞	180°	4.1183
GWO-PID [2]	∞	180°	14.9018
SCA-PID [26]	∞	180°	10.1347

Table 5
Four possible operating point scenarios with changing DC motor parameters.

Motor parameter	Scenario I	Scenario II	Scenario III	Scenario IV
R_a	0.30	0.30	0.50	0.50
K_m	0.012	0.018	0.012	0.018

Table 6
Transient response results for Scenario I.

Controller type	$M_p(\%)$	$T_r(s)$	$T_s(s)$
OBL/HGSO-PID (proposed)	0.0560	**0.0678**	**0.1163**
HGSO-PID (proposed)	0.0127	0.0849	0.1455
ASO-PID [3]	**0.0000**	0.0872	0.1936
SFS-PID [28]	**0.0000**	0.6569	1.2031
GWO-PID [2]	1.5195	0.1683	0.2471
SCA-PID [26]	2.1514	0.2447	0.5618

Table 7
Transient response results for Scenario II.

Controller type	$M_p(\%)$	$T_r(s)$	$T_s(s)$
OBL/HGSO-PID (proposed)	0.0959	**0.0455**	**0.0785**
HGSO-PID (proposed)	0.0659	0.0570	0.0982
ASO-PID [3]	**0.0000**	0.0569	0.1198
SFS-PID [28]	**0.0000**	0.4452	0.7986
GWO-PID [2]	1.8415	0.1173	0.1731
SCA-PID [26]	2.9696	0.1733	0.4787

6.7. Comparison of load disturbance analysis

The ability of OBL/HGSO-PID controller to respond to unexpected disturbance effects occurring in the speed control system is discussed in this sub-section. To do so, a disturbing signal for the torque load ($T_L = 0.01$) was created at $t = 0$ s. The disturbance needs to be suppressed quickly by the designed controller in order to minimize its effects on the performance of the system. The speed response of DC motor due to the disturbing load torque effect is depicted in Fig. 19. As clearly seen the OBL/HGSO-PID provides faster and better suppression to the disturbing load torque effect with minimum undershoot and settling time compared to HGSO-PID, ASO-PID [3], SFS-PID [28], GWO-PID [2], and SCA-PID [26] controllers.

Table 8
Transient response results for Scenario III.

Controller type	$M_p(\%)$	$T_r(s)$	$T_s(s)$
OBL/HGSO-PID (proposed)	**0.0000**	**0.0682**	**0.1191**
HGSO-PID (proposed)	**0.0000**	0.0856	0.1499
ASO-PID [3]	**0.0000**	0.0881	0.2107
SFS-PID [28]	**0.0000**	0.7027	3.7214
GWO-PID [2]	0.9547	0.1706	0.2547
SCA-PID [26]	1.3036	0.2492	0.3573

Table 9
Transient response results for Scenario IV.

Controller type	$M_p(\%)$	$T_r(s)$	$T_s(s)$
OBL/HGSO-PID (proposed)	**0.0000**	**0.0457**	**0.0798**
HGSO-PID (proposed)	**0.0000**	0.0573	0.1002
ASO-PID [3]	**0.0000**	0.0573	0.1257
SFS-PID [28]	**0.0000**	0.4649	3.3544
GWO-PID [2]	1.4479	0.1185	0.1767
SCA-PID [26]	2.3664	0.1755	0.4326

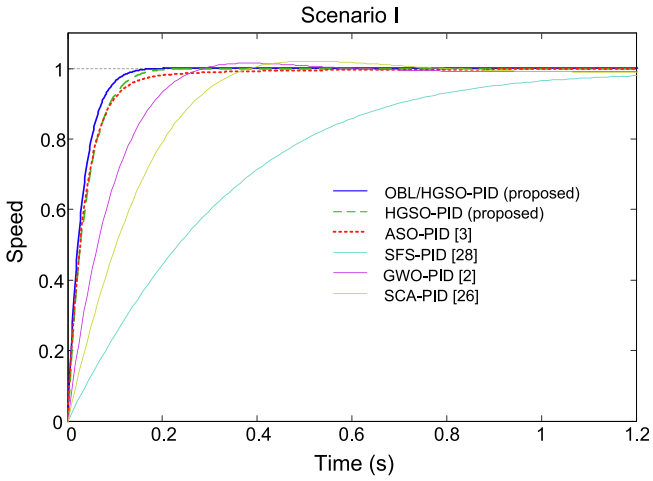


Fig. 15. Speed step responses of DC motor for Scenario I.

6.8. Comparison of energy and maximum control signal of controllers

Since MATLAB/Simulink implementation requires a non-ideal transfer function for the adopted controller, the proposed OBL/HGSO-PID controller was implemented with a first-order filter and tested in Simulink in order to obtain the controller efforts. The transfer function of the PID controller using a first-order filter is given as follows

$$G_{PID_filter}(s) = K_p + \frac{K_i}{s} + \frac{sK_d}{T_f s + 1} \tag{32}$$

where T_f is the time constant of the first-order filter and set to 0.001. Comparative performance of the controller outputs tuned

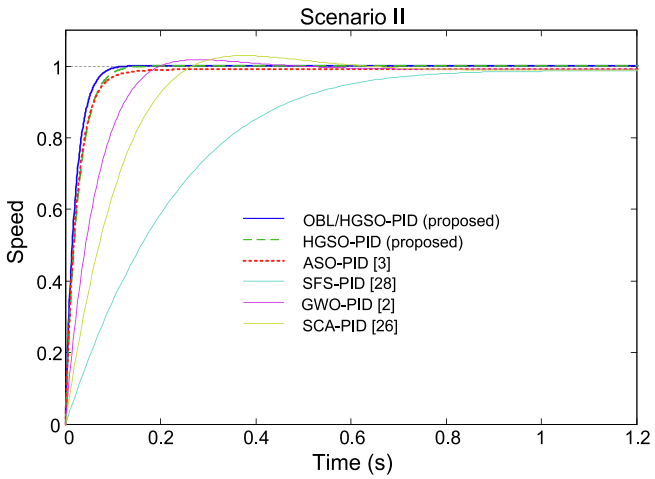


Fig. 16. Speed step responses of DC motor for Scenario II.

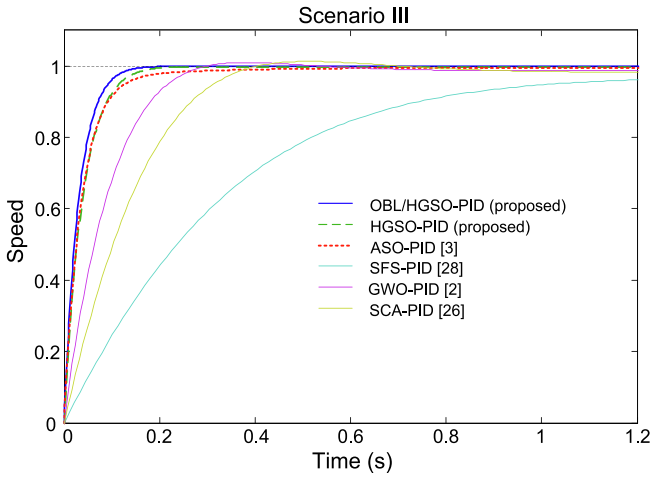


Fig. 17. Speed step responses of DC motor for Scenario III.

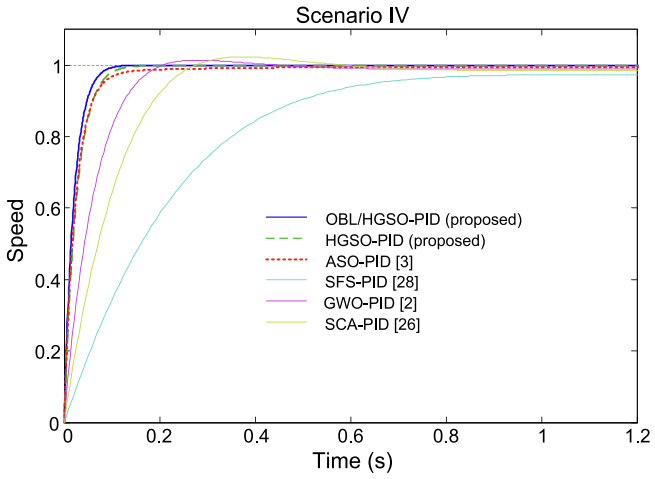


Fig. 18. Speed step responses of DC motor for Scenario IV.

by various approaches in terms of the controller efforts is given in Table 10. It can be seen that the controller with the highest control effort is the proposed OBL/HGSO-PID controller.

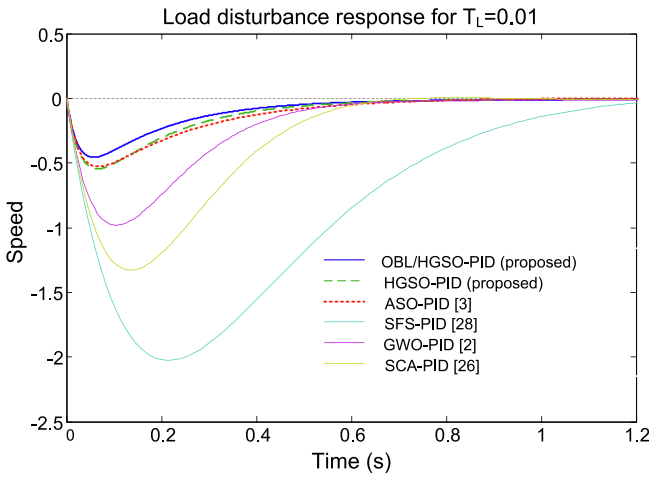


Fig. 19. Load disturbance responses of DC motor.

Table 10
Energy and maximum control signal of controllers.

Controller type	Energy	U_{max}
OBL/HGSO-PID (proposed)	4277.5070	2868.1327
HGSO-PID	2714.9616	2284.1430
ASO-PID [3]	3113.8590	2447.7437
SFS-PID [28]	30.6680	241.1315
GWO-PID [2]	458.3019	936.1984
SCA-PID [26]	150.2166	534.7012

7. Conclusions

The HGSO and OBL/HGSO algorithms were proposed with this study for PID controller design in order to regulate the speed of a DC motor. The efficiency of original HGSO algorithm was enhanced by using the OBL mechanism with it, consequently, the algorithm become capable of finding better solutions. Therefore, the optimal gain parameters of the designed PID controller were obtained without stagnating in local optimal solutions and this was confirmed by conducting statistical test and convergence profile analyzes. The superiority of the proposed algorithm was further verified through comparing it with the-state-of-the-art design approaches [2,3,26,28] in terms of transient and frequency responses, performance index along with robustness and load disturbance suppression analyzes. Therefore, OBL/HGSO algorithm was proven to be more efficient and superior to the compared algorithms in terms of finding the best optimal parameters of PID controller for DC motor speed control system. Concerning future studies, the proposed gain tuning method can be applied to tune the controllers for more complex control systems such as FOPID controller, fuzzy PID and fuzzy FOPID controllers.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] 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. 67 (2020) 5637–5646, https://doi.org/10.1109/TIE.2019.2934030.

[2] 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. 24 (2018) 4997–5006, https://doi.org/10.1007/s00542-018-3920-4.
[3] B. Hekimoğlu, Optimal Tuning of Fractional Order PID Controller for DC Motor Speed Control via Chaotic Atom Search Optimization Algorithm, IEEE Access. 7 (2019) 38100–38114, https://doi.org/10.1109/ACCESS.2019.2905961.
[4] N. Leena, R. Shanmugasundaram, Artificial neural network controller for improved performance of brushless DC motor, in: 2014 Int. Conf. Power Signals Control Comput. EPSCICON 2014, 2014: pp. 1–6. https://doi.org/10.1109/EPSCICON.2014.6887513.
[5] M.A.H. Azman, J.M. Aris, Z. Hussain, A.A.A. Samat, A.M. Nazelan, A comparative study of fuzzy logic controller and artificial neural network in speed control of separately excited DC motor, in: Proc. - 7th IEEE Int. Conf. Control Syst. Comput. Eng. ICCSCE 2017, 2017: pp. 336–341. https://doi.org/10.1109/ICCSCE.2017.8284430.
[6] M.S. Wang, S.C. Chen, C.H. Shih, Speed control of brushless DC motor by adaptive network-based fuzzy inference, Microsyst. Technol. 24 (2018) 33–39, https://doi.org/10.1007/s00542-016-3148-0.
[7] I. Khanam, G. Parmar, Application of SFS algorithm in control of DC motor and comparative analysis, in: 2017 4th IEEE Uttar Pradesh Sect. Int. Conf. Electr. Comput. Electron. UPCON 2017, 2017: pp. 256–261. https://doi.org/10.1109/UPCON.2017.8251057.
[8] B. Hekimoğlu Böbrek-ilhamlı Algoritma Yoluyla Ayarlanan PID Kontrolör Kullanarak DC Motor Hız Kontrolü Bitlis Eren Üniversitesi Fen Bilim. Derg. 8 2019 652 663 https://doi.org/10.17798/bitlisfen.496782
[9] Z.L. Gaing, A particle swarm optimization approach for optimum design of PID controller in AVR system, IEEE Trans. Energy Convers. 19 (2004) 384–391, https://doi.org/10.1109/TEC.2003.821821.
[10] A.K. Mishra, V.K. Tiwari, R. Kumar, T. Verma, Speed control of dc motor using artificial bee colony optimization technique, in: CARE 2013 - 2013 IEEE Int. Conf. Control. Autom. Robot. Embed. Syst. Proc., 2013: pp. 1–6. https://doi.org/10.1109/CARE.2013.6733772.
[11] A. Rodríguez-Molina, M.G. Villarreal-Cervantes, M. Aldape-Pérez, An adaptive control study for a DC motor using meta-heuristic algorithms, IFAC-PapersOnLine. 50 (2017) 13114–13120, https://doi.org/10.1016/j.ifacol.2017.08.2164.
[12] X. Lin, Y. Liu, Y. Wang, Design and research of DC motor speed control system based on improved BAS, in: Proc. 2018 Chinese Autom. Congr. CAC 2018, 2018: pp. 3701–3705. https://doi.org/10.1109/CAC.2018.8623171.
[13] L. Syafaah, Widianto, I. Pakaya, D. Suhardi, M. Irfan, PID designs using DE and PSO algorithms for damping oscillations in a DC motor speed, in: 2017 4th Int. Conf. Electr. Eng. Comput. Sci. Informatics, 2017: pp. 1–5. https://doi.org/10.1109/eecsi.2017.8239138.
[14] A. Rodríguez-Molina, M.G. Villarreal-Cervantes, J. Álvarez-Gallegos, M. Aldape-Pérez, Bio-inspired adaptive control strategy for the highly efficient speed regulation of the DC motor under parametric uncertainty, Appl. Soft Comput. J. 75 (2019) 29–45, https://doi.org/10.1016/j.asoc.2018.11.002.
[15] D. Potnuru, K. Alice Mary, C. Sai Babu, Experimental implementation of Flower Pollination Algorithm for speed controller of a BLDC motor, Ain Shams Eng. J. 10 (2019) 287–295, https://doi.org/10.1016/j.asej.2018.07.005.
[16] A.T. El-Deen, A.A. Hakim Mahmoud, A.R. El-Sawi, Optimal PID tuning for DC motor speed controller based on genetic algorithm, Int. Rev. Autom. Control. 8 (2015) 80–85.
[17] S. Duman, D. Maden, U. Güvenç, Determination of the PID controller parameters for speed and position control of DC motor using gravitational search algorithm, in: ELECO 2011 - 7th Int. Conf. Electr. Electron. Eng., 2011: pp. I-225-I–229.
[18] A. Madadi, M.M. Motlagh, Optimal Control of DC motor using Grey Wolf Optimizer Algorithm, Tech. J. Eng. Appl. Sci. 4 (2014) 373–379.
[19] M. Muniraj, R. Arulmozhiyal, D. Kesavan, An Improved Self-tuning Control Mechanism for BLDC Motor Using Grey Wolf Optimization Algorithm, in: V. Bindhu, J. Chen, J.M.R.S. Tavares (Eds.), Lect. Notes Electr. Eng., Springer Singapore, Singapore, 2020: pp. 315–323. https://doi.org/10.1007/978-981-15-2612-1_30.
[20] M. Khalilpour, H.H. Razmjooy, N., P. Moallem, Optimal Control of DC motor using Invasive Weed Optimization (IWO) Algorithm, in: Majlesi Conf. Electr. Eng., 2011.
[21] R.K. Achanta, V.K. Pamula, DC motor speed control using PID controller tuned by jaya optimization algorithm, in: 2017 IEEE Int. Conf. Power, Control. Signals Instrum. Eng., 2017: pp. 983–987. https://doi.org/10.1109/ICPCSI.2017.8391856.
[22] R. V Jain, M. V Aware, A.S. Junghare, Tuning of Fractional Order PID controller using particle swarm optimization technique for DC motor speed control, in: 2016 IEEE 1st Int. Conf. Power Electron. Intell. Control Energy Syst., 2016: pp. 1–4. https://doi.org/10.1109/ICPEICES.2016.7853070.
[23] A. Roy, S. Srivastava, Design of optimal PIλDδ controller for speed control of DC motor using constrained particle swarm optimization, in: Proc. IEEE Int. Conf. Circuit, Power Comput. Technol. ICCPCT 2016, 2016: pp. 1–6. https://doi.org/10.1109/ICCPCT.2016.7530150.
[24] S. Khubalkar, A. Junghare, M. Aware, S. Das, Modeling and control of a permanent-magnet brushless DC motor drive using a fractional order proportional-integral-derivative controller, Turkish J. Electr. Eng. Comput. Sci. 25 (2017) 4223–4241, https://doi.org/10.3906/elk-1612-277.
[25] B. Hekimoğlu, S. Ekinci, V. Demiray, R. Doğurici, A. Yıldırım, Speed Control of DC Motor Using PID Controller Tuned By Salp Swarm Algorithm, in: 2018 Int. Eng. Nat. Sci. Conf. (IENSC 2018), 2018: pp. 1878–1889.
[26] J. Agarwal, G. Parmar, R. Gupta, Application of sine cosine algorithm in optimal control of DC motor and robustness analysis, Wulfenia J. 24 (2017).
[27] S. Ekinci, B. Hekimoğlu, A. Demirören, E. Eker, Speed Control of DC Motor Using Improved Sine Cosine Algorithm Based PID Controller, in: 2019 3rd Int. Symp. Multidiscip. Stud. Innov. Technol., 2019: pp. 1–7. https://doi.org/10.1109/ISMSIT.2019.8932907.
[28] R. Bhatt, G. Parmar, R. Gupta, A. Sikander, Application of stochastic fractal search in approximation and control of LTI systems, Microsyst. Technol. 25 (2019) 105–114, https://doi.org/10.1007/s00542-018-3939-6.
[29] Z. Han, H. Qi, L. Wang, M.I. Menhas, M. Fei, Water level control of nuclear power plant steam generator based on intelligent virtual reference feedback tuning, in: Commun. Comput. Inf. Sci., 2018: pp. 14–23. https://doi.org/10.1007/978-981-13-2381-2_2.
[30] Y. Wen, L. Wang, W. Peng, M.I. Menhas, L. Qian, Application of Intelligent Virtual Reference Feedback Tuning to Temperature Control in a Heat Exchanger, in: K. Li, M. Fei, D. Du, Z. Yang, D. Yang (Eds.), Commun. Comput. Inf. Sci., Springer Singapore, Singapore, 2018: pp. 311–320. https://doi.org/10.1007/978-981-13-2384-3_29.
[31] M.I. Menhas, L. Wang, N. Ayesha, N. Qadeer, M. Waris, S. Manzoor, M. Fei, Continuous Human Learning Optimizer based PID Controller Design of an Automatic Voltage Regulator System, in: 2018 Aust. New Zeal. Control Conf., 2018: pp. 148–153. https://doi.org/10.1109/ANZCC.2018.8606577.
[32] 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. 21 (2018) 1104–1111, https://doi.org/10.1016/j.jestch.2018.08.006.
[33] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey Wolf Optimizer, Adv. Eng. Softw. 69 (2014) 46–61, https://doi.org/10.1016/j.advengsoft.2013.12.007.
[34] R.A. Ibrahim M.A. Elaziz S. Lu Chaotic opposition-based grey-wolf optimization algorithm based on differential evolution and disruption operator for global optimization Expert Syst. Appl. 108 2018 1 27 https://doi.org/https://doi.org/10.1016/j.eswa.2018.04.028.
[35] S. Mirjalili, SCA: A Sine Cosine Algorithm for solving optimization problems, Knowledge-Based Syst. 96 (2016) 120–133, https://doi.org/10.1016/j.knosys.2015.12.022.
[36] S. Gupta, K. Deep, A novel hybrid sine cosine algorithm for global optimization and its application to train multilayer perceptrons, Appl. Intell. 50 (2020) 993–1026, https://doi.org/10.1007/s10489-019-01570-w.
[37] H. Salimi, Stochastic Fractal Search: A powerful metaheuristic algorithm, Knowledge-Based Syst. 75 (2015) 1–18, https://doi.org/10.1016/j.knosys.2014.07.025.
[38] W. Zhao, L. Wang, Z. Zhang, Atom search optimization and its application to solve a hydrogeologic parameter estimation problem, Knowledge-Based Syst. 163 (2019) 283–304, https://doi.org/10.1016/j.knosys.2018.08.030.
[39] J. Lin, Z.J. Wang, Parameter identification for fractional-order chaotic systems using a hybrid stochastic fractal search algorithm, Nonlinear Dyn. 90 (2017) 1243–1255, https://doi.org/10.1007/s11071-017-3723-7.
[40] S. Ekinci A. Demiroren H. Zeynelgil B. Hekimoğlu An opposition-based atom search optimization algorithm for automatic voltage regulator system J. Fac. Eng. Archit. Gazi Univ. 35 2020 1141 1158 https://doi.org/10.17341/gazimmfd.598576.
[41] H.R. Tizhoosh, Opposition-Based Learning: A New Scheme for Machine Intelligence, in: Int. Conf. Comput. Intell. Model. Control Autom. Int. Conf. Intell. Agents, Web Technol. Internet Commer., 2005: pp. 695–701. https://doi.org/10.1109/CIMCA.2005.1631345.
[42] M. Abd Elaziz, D. Oliva, S. Xiong, An improved Opposition-Based Sine Cosine Algorithm for global optimization, Expert Syst. Appl. 90 (2017) 484–500, https://doi.org/10.1016/j.eswa.2017.07.043.
[43] S. Gupta, K. Deep, A hybrid self-adaptive sine cosine algorithm with opposition based learning, Expert Syst. Appl. 119 (2019) 210–230, https://doi.org/10.1016/j.eswa.2018.10.050.
[44] A.A. Ewees, M. Abd Elaziz, E.H. Houssein, Improved grasshopper optimization algorithm using opposition-based learning, Expert Syst. Appl. 112 (2018) 156–172, https://doi.org/10.1016/j.eswa.2018.06.023.
[45] S. Shekhawat, A. Saxena, Development and applications of an intelligent crow search algorithm based on opposition based learning, ISA Trans. (2019), https://doi.org/10.1016/j.isatra.2019.09.004.
[46] M. Abd Elaziz, D. Oliva, Parameter estimation of solar cells diode models by an improved opposition-based whale optimization algorithm, Energy Convers. Manag. 171 (2018) 1843–1859, https://doi.org/10.1016/j.enconman.2018.05.062.
[47] H. Chen, S. Jiao, A.A. Heidari, M. Wang, X. Chen, X. Zhao, An opposition-based sine cosine approach with local search for parameter estimation of photovoltaic models, Energy Convers. Manag. 195 (2019) 927–942, https://doi.org/10.1016/j.enconman.2019.05.057.
[48] M. Pradhan, P.K. Roy, T. Pal, Oppositional based grey wolf optimization algorithm for economic dispatch problem of power system, Ain Shams Eng. J. 9 (2018) 2015–2025, https://doi.org/10.1016/j.asej.2016.08.023.
[49] Y. Yang, B. Yang, M. Niu, Adaptive infinite impulse response system identification using opposition based hybrid coral reefs optimization algorithm, Appl. Intell. 48 (2018) 1689–1706, https://doi.org/10.1007/s10489-017-1034-9.
[50] R.A. Ibrahim, M.A. Elaziz, D. Oliva, E. Cuevas, S. Lu, An opposition-based social spider optimization for feature selection, Soft Comput. 23 (2019) 13547–13567, https://doi.org/10.1007/s00500-019-03891-x.

[51] M. Tubishat, N. Idris, L. Shuib, M.A.M. Abushariah, S. Mirjalili, Improved Salp Swarm Algorithm based on opposition based learning and novel local search algorithm for feature selection, Expert Syst. Appl. 145 (2020), https://doi.org/10.1016/j.eswa.2019.113122 113122.
[52] X. Bao, H. Jia, C. Lang, Dragonfly algorithm with Opposition-based learning for multilevel thresholding color image segmentation, Symmetry (Basel). 11 (2019), https://doi.org/10.3390/sym11050716.
[53] F.A. Hashim, E.H. Houssein, M.S. Mabrouk, W. Al-Atabany, S. Mirjalili, Henry gas solubility optimization: A novel physics-based algorithm, Futur. Gener. Comput. Syst. 101 (2019) 646–667, https://doi.org/10.1016/j.future.2019.07.015.
[54] F.A. Hashim, E.H. Houssein, K. Hussain, M.S. Mabrouk, W. Al-Atabany, A modified Henry gas solubility optimization for solving motif discovery problem, Neural Comput. Appl. (2019), https://doi.org/10.1007/s00521-019-04611-0.
[55] B.S. Yıldız, A.R. Yıldız, N. Pholdee, S. Bureerat, S.M. Sait, V. Patel, The Henry gas solubility optimization algorithm for optimum structural design of automobile brake components, Mater. Test. 62 (2020) 261–264, https://doi.org/10.3139/120.111479.
[56] W. Cao, X. Liu, J. Ni, Parameter Optimization of Support Vector Regression Using Henry Gas Solubility Optimization Algorithm, IEEE Access. 8 (2020) 88633–88642, https://doi.org/10.1109/ACCESS.2020.2993267.
[57] N. Neggaz, E.H. Houssein, K. Hussain, An efficient henry gas solubility optimization for feature selection, Expert Syst. Appl. 152 (2020), https://doi.org/10.1016/j.eswa.2020.113364 113364.
[58] S. Ekinci, D. Izci, B. Hekimoğlu, Henry Gas Solubility Optimization Algorithm Based FOPID Controller Design for Automatic Voltage Regulator, in: 2nd Int. Conf. Electr. Commun. Comput. Eng., IEEE, Istanbul, 2020.
[59] W. Henry, J. Banks III, Experiments on the quantity of gases absorbed by water, at different temperatures, and under different pressures, Philos. Trans. R. Soc. London. 93 (1803) 29–274, https://doi.org/10.1098/rstl.1803.0004.
[60] J. Staudinger, P.V. Roberts, A critical review of Henry's law constants for environmental applications, Crit. Rev. Environ. Sci. Technol. 26 (1996) 205–297, https://doi.org/10.1080/10643389609388492.
[61] M.A. Sahib, A novel optimal PID plus second order derivative controller for AVR system, Eng. Sci. Technol. an Int. J. 18 (2015) 194–206, https://doi.org/10.1016/j.jestch.2014.11.006.