Shameem HASAN[1], Golam SAROWAR[2]

University of Liberal Arts Bangladesh (ULAB) (1), Islamic University of Technology (IUT) (1,2), Bangladesh

doi:10.15199/48.2021.04.06

Amelioration of Ship Control with Improved Dynamic Response of Motor Controller

***Abstract**. During a sea voyage, it is significantly important to fix the speed and angle of movement of the ship precisely depending on the distance of the destination and also in icebreaking purpose. So, enhancement of transient response of commutator motor used in the ship is essential. Due to their better reliabilities as well as low costing, nowadays, Direct Current (commutator) motors are utilized in icebreaking ships, in industrial near and far applications, robotic manipulators, and also for home appliances. Thus, it is vital to introduce a suitable controller for managing the speed and transient behavior of a Direct Current. In this study, for the enhancement of dynamic response, various DC commutator motor controllers have been simulated. At first, the DC commutator motor parameters are selected and four optimized controllers: Ziegler-Nichols (ZN)-based conventional Proportional Integral Derivative (PID) controller, Genetic Algorithm (GA)-based PID controller (GA-PID), as well as Flower Pollination Algorithm (FPA)-based PID (FPA-PID) are designed and simulated to manage the angular speed together with dynamic response of shaft of a Direct Current commutator motor actuator. The electric actuator response for every controller is ascertained as well as compared after applying step input that is necessary for the simulation of transient response of the motor. The performance analysis shows that the FPA-PID controller is adequate for the steering task in the ship, which requires precision and associated with transient response properties.*

***Streszczenie.** W artykule analizowano różne metody sterowania silnikiem komutacyjnym DC. Szczególną uwagę poświęcono sterownikom PID , PID wspomaganym algorytmami genetycznymi oraz Flower Pollination FPA-PID. . Analizowano możliwości sterowania prędkością przy odpowiednich parametrach dynamicznych. (odpowiedź na wymuszenie skokowe). Stwierdzono, że najlepsze właściwości do sterowani silnikiem statku ma algorytm FPA-PID. (**Poprawa możliwości sterowanie silnikiem statku z uwzględnieniem odpowiedzi dynamicznej**)*

Słowa kluczowe: sterowanie silnikiem komutatorowym, silnik statku, algorytm FPA-PID, odpowiedź dynamiczna
Keywords: Flower Pollination Algorithm, Genetic Algorithm, Proportional-Integral-Derivative, Transient response

Introduction

This paper starts with briefly describing the classification and applications of DC machines. A Direct Current motor converts electrical energy into mechanical energy [1,2]. A Direct Current motor's speed and dynamic response can vary over a wide range. Small size Direct Current motors are used to prepare tools, toys, along with other appliances. Large Direct Current motors are utilized in electric vehicles, elevator, or drives for steel rolling mills. DC commutator motor has also been used in submarines [2,3].

One of the major applications of DC motor is ship control. In this paper, the Direct Current (Commutator) motor mounted in the ship to control the steering along with speed is the main discussion topic. Induction and synchronous motors are sometimes used in ships but DC commutator motor is preferred because of its simplicity and flexibility. The power rating of DC motor is 3.70 kW, rated voltage is 240 V and speed is 1750 rpm [4]. To ensure the steering operation and safe movement of the ship, amelioration of dynamic response is necessary. Initially, with selection of appropriate parameters, a Direct Current commutator motor is modeled. From that, transfer function is derived [5].

A moving mechanical device included in the control system of a ship can be called actuator. Actuators are the heart of a mechatronic systems like ships. Electrical energy available on ship is used to actuate the mechanical system using magnetic field (i.e. emf). Speed control of DC commutator motor require armature voltage and field current variation. Electric actuators require less energy and does not need auxiliary equipments. Electric actuator of the ship is driven by DC commutator motor. To achieve higher performance of Direct Current commutator motor [6], the need of suitable controller design is realized. The optimum design criteria [7] are selected. Ziegler- Nichols 1st and 2nd methods are simulated for finding out the gains of conventional PID controller. This conventional PID controller is designed with Ziegler-Nichols sustained oscillation method (2nd method), for closed loop DC motor drive [8].

Genetic Algorithm popularly known as GA is regarded as a random global search algorithm and optimization method that is similar to the method of natural evolution. John Holland introduced GA in 1970. Simultaneous improvement in performance for computational type systems has made them lucrative for applications related to optimization. GA is used for the selected motor drive's dynamic response improvement.

Several incidents in nature have possess characteristics that can be used. Over the recent years, researchers have invented several algorithms which are nature-inspired, to deduct the best possible optimum solutions for various problems. Among them, notable examples are Artificial Bee Colony (ABC) Algorithm [9], Genetic Algorithm (GA) [10], Particle Swarm Optimization (PSO) [11], Gray Wolf Algorithm (GWA) [12,13], Firefly Algorithm (FA) [14], Bat Algorithm (BA) [15] and Cuckoo Search Algorithm (CSA) [16]. Besides, nature still contains various other features which can be utilized to make solution of different types of optimization problems. One such event or example is reproduction strategy of flowering plant by pollination, popularly called as the Flower Pollination Algorithm (FPA) [17]. In 2012, Yang proposed this algorithm. In this paper, aspects of FPA as a robust and user-friendly metaheuristic optimization algorithm. A. Y. Al Maliki and K. Iqbal designed a PID without Kalman Filter resulting in settling time of 1.07 s and overshoot of 23.7%. They also designed a PID controller with Kalman Filter where settling time of 0.635 s as well as overshoot of 8.014% was achieved. After that they designed a Fuzzy logic controller with Kalman filter where settling time was 0.257 s and overshoot was 5% [18].

Wahyudi and M. Rosaliana M. et al. designed a PID controller resulting in rise time of 1.625 s, settling time of 2.234 sec along with overshoot of 4.4%. They also designed a Fuzzy PID controller where rise time was 1.828 s, settling time was 3.453 s and overshoot was 3.42% [19].

In case of GA controller, J. Pongfai and W. Assawinchaichote found rise time 1.088 s, settling time 2.802 s together with overshoot 21.7%. They also observed rise time of 0.44 s, settling time of 1.629 s and overshoot of 7.6% in case of Neural Network. When Neural Network

(NN) was applied with GA, it was found out that rise time is 0.41 s, settling time is 1.54 s and overshoot is 6.9% [20].

H. S. Hameed designed a PID controller resulting in rise time of 0.0202 s, settling time of 0.35 s and overshoot of 16.53% [21].

H. Chaudhary, S. Khatoon and R. Singh designed a PID controller resulting in rise time of 0.1 s, settling time of 0.5 s and overshoot of 14.2%. They also designed a Fuzzy Logic Controller with rise time of 0.86 s, settling time of 0.3 s and overshoot of 18.9%. They also simulated Adaptive Neuro Fuzzy Logic controller from where rise time of 0.01 s, settling time of 0.02 s and overshoot of 0.5% was obtained [22].

M. K. Rout and D. Sain et al. designed a PID controller with rise time of 5.5 s, settling time of 1.7 s and overshoot of 12%. They also simulated state space controller with rise time of 5 s, settling time of 1.38 sec and overshoot of 10%. After that they designed a Fuzzy logic controller with where rise time was 3.37 s, settling time was 2.21 s and overshoot was 9.1% [23].

Z. Has, A. H. Muslim et al. designed a PID controller resulting in rise time of 1.1153 s, settling time of 2.53 s and overshoot of 6.84%. They also designed an Adaptive Fuzzy PID controller where rise time of 0.4533 s, settling time of 0.02 s and overshoot of 7.23% was achieved [24].

In case of GA based controller design, N. P. Adhikari, M. Choubey et al. found rise time of 0.161 sec, settling time of 0.7 sec and overshoot of 10.1%. They also observed rise time of 0.18 s, settling time of 2.41 s and overshoot of 10.8% in case of Ziegler-Nichols method for PID controller [25].

P. M. Meshram and R. G. Kanojiya simulated traditional ZN based PID controller and found out rise time of 0.377 s, settling time of 5.68 s and overshoot of 51.8%. They also found out that in case of CHR based controller, rise time is 0.074 s, settling time is 0.439 s and overshoot is 14.5% [26].

Y. A. Almatheel and A. Abdelrahman designed a PID controller where rise time was 0.8727 s, settling time was 2.9782 s in addition to overshoot was 8%. They also designed a Fuzzy logic controller where rise time was obtained as 0.76 s, settling time as 2.62 s and overshoot as 1.2% [27].

Flower Pollination Algorithm (FPA) is considered a biologically inspired algorithm to solve different problems related to optimization [13]. The standard version of FPA works well in case of many applications, but there is still scope of improvement. Considering the complex nature of optimization problems in real life, the basic FPA has been modified to improve the performance.

M. Abdel Basset and L. A. Shawky [28,29] presented a hybrid FPA method for optimization problems. In this hybrid method, FPA and PSO are combined together to improve the search accuracy. Results proved that this combined approach contained more accuracy as well as efficiency to find out optimal solutions compared to other methods discussed in that literature.

M. Abdel-Baset and I. M. Hezam [30] proposed a hybrid version of FPA along with GA, called FPA-GA, to provide solution of problems like PID controller optimization. In their proposed method, the GA was implemented after initial FPA loop. The authors performed tests on seven well-known benchmarking design problems. The result indicated that the performance of the proposed FPA-GA was better than the conventional algorithms.

This paper mainly focuses on the improvement of transient response of DC commutator motor mounted as the electric actuator component in a ship using optimized controller so that improved steering and speed control can be ensured with safe movement. Dynamic response of DC commutator motor with ZN-PID, GA-PID and FPA-PID controllers are simulated in MATLAB and Simulink [31] and compared to find out the optimized controller for controlling the speed and propulsion of the motor situated the ship.

DC Motor Modelling

An equivalent circuit of a Direct Current (commutator) motor is displayed in Fig. 1 [11]. In case of a dc commutator motor, ‘‘commutator’’ does the operation of rectification of voltage which is induced in the conductors. It also makes the coils of the moving armature switch so that the coils always remain under the influence of the magnetic field provided by the poles.

By applying Kirchhoff's Voltage Law (KVL), the equation of the Direct Current motor becomes [32]

(1) $$i_a(t)R_a + v_b(t) + \frac{di_a(t)}{dt}L_a = v_s(t)$$

Here, $i_a(t)$ indicates the current flowing through armature, $v_b(t)$ stands for back emf and $v_s(t)$ symbolizes the source voltage [1].

(2) $$v_b(t) = K_e\omega(t)$$

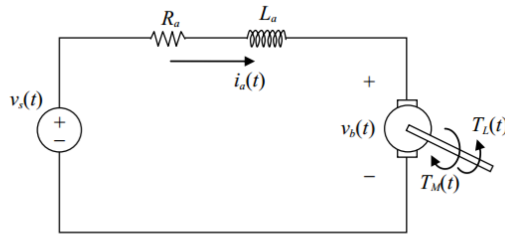

Fig. 1. An equivalent circuit of Direct Current (commutator) motor

The rotor generates a torque indicated by T_M which is proportional to the current of the armature.

(3) $$T_M(t) = K_T i_a(t)$$

Considering $v_s(t) = V_s$ is a constant, $i_a(t) = I_a$, $\omega(t) = \Omega$ and $T_M(t) = T$ are also constant in the steady-state. From (1) to (3), the equations can be rewritten as [33]

(4) $$R_a I_a + K_e\Omega = V_s$$

(5) $$T = I_a K_T$$

(6) $$V_s I_a = T\Omega + I_a^2 R_a$$

(7) $$T = K_e I_a$$

From (5) and (7), it can be understood that, K_e and K_T are same. Besides, to drive an external torque, $T_L(t)$, the mechanical equivalent equation of Direct Current commutator motor is described as (8) [26,32].

(8) $$\omega(t)B_M + \frac{d\omega(t)}{dt}J_M = -T_L(t) + T_M(t)$$

Based on (1), (2) and (8),

(9) $$R_a i_a(t) + L_a\frac{di_a(t)}{dt} + \mathrm{k}\omega(t) = v_s(t)$$

(10) $$J_M\frac{d\omega(t)}{dt} + B_M\omega(t) - ki_a(t) = -T_L(t)$$

Electrical time constant (L_a/R_a) is sometimes ignored since it has one order smaller magnitude than the mechanical time constant (J_M/B_M).

(11) $$i_a(t) = \frac{1}{R_a}v_s(t) - \frac{k}{R_a}\omega(t)$$

Substituting (11) into (10),

(12) $$\frac{d\omega(t)}{dt} + (\frac{k^2}{R_a J_M} + \frac{B_M}{J_M})\omega(t) = \frac{k}{R_a J_M}v_s(t) - T_L(t)\frac{1}{J_M}$$

The Direct Current motor experiences two types of external sources, $v_s(t)$ and $T_L(t)$ [32,33]. Now, a Direct Current motor model in state-space needs to be derived [21,32].

At first, a case is considered where the Direct Current motor needs to rotate at a constant speed and $\omega(t)$ is chosen as output.

(13) $y(t)=\omega(t)$

Now, assuming state variables as $x_1(t) = i_a(t)$ and $x_2(t) = \omega(t)$, the equations become

(14) $$\frac{dx_1(t)}{dt}L_a + x_1(t)R_a + x_2(t)k = v_s(t)$$

(15) $$\frac{dx_2(t)}{dt}J_M - kx_1(t) + x_2(t)B_M = -T_L(t)$$

Putting the values of state variables in (14) and (15),

(16) $$\frac{dx_1(t)}{dt} = -x_1(t)\frac{R_a}{L_a} - x_2(t)\frac{k}{L_a} + v_s(t)\frac{1}{L_a}$$

(17) $$\frac{dx_2(t)}{dt} = x_1(t)\frac{k}{J_M} - x_2(t)\frac{B_M}{J_M} - T_L(t)\frac{1}{J_M}$$

Output equation, (18) $y(t) = x_2(t)$

Standard State equation: (19) $\frac{dx_1(t)}{dt} = x(t)A + u(t)B$

Output equation: (20) $cx(t) = y(t)$

Now, state vectors expressed by $x(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$, input vector,

$u(t)=\begin{bmatrix} v_s(t) \\ T_L(t) \end{bmatrix}$, $A=\begin{bmatrix} -\frac{R_a}{L_a} & -\frac{k}{L_a} \\ \frac{k}{J_M} & -\frac{B_M}{J_M} \end{bmatrix}$, $B=\begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{1}{J_M} \end{bmatrix}$ and $c=\begin{bmatrix} 0 & 1 \end{bmatrix}$

Again, rearranging (19) provides,

(21) $$\frac{dx_1(t)}{dt} = x(t)A + u_1(t)b_1 + u_2(t)b_2$$

Here, $b_1=\begin{bmatrix} \frac{1}{L_a} \\ 0 \end{bmatrix}$, $b_2=\begin{bmatrix} 0 \\ -\frac{1}{J_M} \end{bmatrix}$, $u_1(t) = v_s(t)$ and $u_2(t) = T_L(t)$

When the Direct Current motor is operating without any payload, $u_2(t) = T_L(t) = 0$.

If the aim is to manage the desired angle of the Direct Current motor, then the output is [33]

(22) $$y(t) = \int_0^t \omega(\tau)d\tau = \theta(t)$$

Choosing $x_3(t) = \theta(t)$, state equations of the total system is

(23) $$\frac{dx_1(t)}{dt} = \frac{-R_a}{L_a}x_1(t) - \frac{k}{L_a}x_2(t) + \frac{1}{L_a}v_s(t)$$

(24) $$\frac{dx_2(t)}{dt} = \frac{k}{J_M}x_1(t) - \frac{B_M}{J_M}x_2(t) - \frac{1}{J_M}T_L(t)$$

(25) $$\frac{dx_3(t)}{dt} = x_2(t)$$

Output equation:

(26) $y(t) = x_3(t)$

Table 1. DC motor dynamics [4]

Symbol	Parameters	Selected values
J_M	Moment of inertia	0.02215 Kgm^2
B_M	Coefficient of viscous friction	0.002953 Nms
K_T	Torque constant	1.28 Nm/A
K_e	Back emf constant	1.28 Vs/rad
R_a	Armature resistance	11.2 Ω
L_a	Inductance of armature	0.1215 H

In matrix form, we have state equation:

(27) $$\frac{dx(t)}{dt} = Ax(t) + Bu(t)$$

Output equation: (28) $y(t) = cx(t)$

where $x(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}$, $u(t)=\begin{bmatrix} v_s(t) \\ T_L(t) \end{bmatrix}$,

$A=\begin{bmatrix} -\frac{R_a}{L_a} & -\frac{k}{L_a} & 0 \\ \frac{k}{J_M} & -\frac{B_M}{J_M} & 0 \\ 0 & 1 & 0 \end{bmatrix}$, $B=\begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{1}{J_M} \\ 0 & 0 \end{bmatrix}$ and $c=\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$

Without any payload $T_L(t)$,

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} -92.18 & -10.535 & 0 \\ 57.788 & -0.133 & 0 \\ 0 & 1 & 0 \end{bmatrix} . \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 8.23 \\ 0 \\ 0 \end{bmatrix} u(t)$$

(28)

where $i_a(t) = x_1(t)$, $\omega(t) = x_2(t)$, $\theta(t) = x_3(t)$ and $v_s(t) = u(t)$.

If the output is $\theta(t) = x_3(t)$ then

$$y(t)=\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

(29)

The characteristic polynomial of Direct Current motor can be obtained as

$$|sI - A| = \begin{vmatrix} s+92.18 & 10.535 & 0 \\ -57.788 & s+0.133 & 0 \\ 0 & -1 & s \end{vmatrix} = s^3 + 92.3s^2 + 621.05s$$

(30)

Transfer function of Direct Current is

$$H(s) = \frac{Y(s)}{U(s)} = \frac{475.84}{s^3 + 92.3s^2 + 621.05s}$$

(31)

$$H(s) = \frac{Y(s)}{U(s)} = \frac{475.84}{s(s+85)(s+7.31)}$$

(32)

The equation describing the relation between the reference angular speed symbolized by $\dot{\theta}_{ref}$ as well as the output angular speed symbolized by $\dot{\theta}_{out}$ is indicated by transfer function of the closed-loop system displayed in (33) where K is considered the constant gain [2].

$$\frac{\dot{\theta}_{out}}{\dot{\theta}_{ref}} = \frac{\frac{K.K_T}{J_M L_a}}{s^2 + \frac{(J_M R_a + B_M L_a)}{J_M L_a}s + \frac{K.K_T + B_M R_a + K_e K_T}{J_M L_a}}$$

(33)

Design Criteria

There are three important parameters of transient response of a system which are mentioned below [5]:

Rise Time (t_r): Time needed for a response of a system to rise from 10% to 90% (overdamped); 5% to 95% (critically damped); 0% to 100% (under damped) value of steady state response is called rise time.

$$t_r = \frac{\pi - \tan^{-1}\frac{\sqrt{1-\xi^2}}{\xi}}{\omega\sqrt{1-\xi^2}}$$

(34)

Settling Time (t_s): The time needed for a dynamic response to attain a specific range of approximately (2% - 5%) of its final value, is known as settling time [2].

(35) $t_s = \frac{4}{\omega\varsigma}$

Maximum Overshoot (*%OS*): Maximum positive deviation of the response of a system from its desired value. Here the desired amount is the steady-state value [5,7].

(36) $\%OS = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}} \times 100$

The design criteria of the controller for improved transient response can be mentioned as [5]:

- Rise time < 1 sec.
- Overshoot < 5%.
- Settling time < 1 sec.
-

Conventional PID Controller

Proportional-Integral-Derivative controllers are utilized all over the world [1]. There are some specific rules utilized in case of PID tuning. Ziegler and Nichols proposed an example of PID controller tuning in 1940 [8]. These rules are mainly based on specific assumed models. The adjustment of a Proportional-Integral-Derivative controller consists of choosing gains (K_P, K_I, and K_D) [34]. These gains are important to ensure the satisfaction of performance completely. By applying Ziegler– Nichols method, those gains are found by Laplace transformation.

(37) $C(s) = K_p + \frac{K_I}{s} + K_D s$

(38) $C(s) = K_P(1 + \frac{1}{T_i s} + T_d s)$

Here, K_P = Proportional Gain of controllers
K_I = Integral gain of controllers
K_D = Controllers Derivative gain
T_i = Reset time of the controller = $\frac{K_P}{K_I}$
T_d = Rate time or derivative time of the controller

Table 2. Impact of PID controller in closed loop system

Action	Rise time	Overshoot	Settling time
Increase K_P	Decrease	Increase	Small change
Increase K_i	Decrease	Increase	Increase
Increase K_d	Minor decrease	Minor decrease	Minor Decrease

Ziegler-Nichols 1st method for tuning

Two parameters characterize the transient response in this method, *L* and *T* [8]. Here, *L* indicates the “delay time” and *T* is considered as the “time constant”. The equation of plant model is shown in (39)

(39) $G(s) = \frac{Ke^{-sL}}{Ts+1}$

Table 3. Controller gains obtained from Ziegler-Nichols 1st method

PID type	K_P	$T_i = \frac{K_p}{K_i}$	$T_d = \frac{K_d}{K_p}$
P	*T/L*	∞	*0*
PI	*0.9T/L*	*L/0.3*	*0*
PID	*1.2 T/L*	*2L*	*L/2*

Here, *T*= 0.07 sec and *L*= 0.04. From that K_p = 2.1, T_i = 0.08 and T_d = 0.005. Ziegler-Nichols 1st method is more useful for an open-loop response. But we need a closed-loop response for better tuning. For that, Ziegler Nichols 2nd method of tuning for sustained oscillation is introduced [35].

Ziegler-Nichols Tuning method for sustained oscillation

In Ziegler-Nichols 2nd method which is also called sustained oscillation method, critical gain K_{cr} and critical period P_{cr} is obtained with proper tuning. The transfer function of PID controller is shown in (40)

(40) $G_C(s) = K_P(1 + \frac{1}{sT_i} + sT_d)$

The second method is especially helpful for motors or plants which can be unstable in case of proportional control [28]. The steps required for tuning Proportional-Integral-Derivative (PID) controller with the help of 2nd method is as follows [25,36]:

- At first, K_I as well as K_D are reduced to zero and K_P is increased from zero to some value which can be considered as critical value when K_P = K_{cr}. For this condition, sustained oscillation takes place.
- The value of critical gain (K_{cr}) and the corresponding period of sustained oscillation, (P_{cr}) is noted.
- Initially, integral time (T_i) was fixed as ∞ and derivative time (T_d) was set to 0 [37].

Table 4. Controller gains for Ziegler-Nichols 2nd method for sustained oscillation

PID Type	K_P	$T_i = \frac{K_p}{K_i}$	$T_d = \frac{K_d}{K_p}$
P	*0.5*K_{cr}	∞	*0*
PI	*0.45*K_{cr}	$\frac{P_{cr}}{1.2}$	*0*
PID	*0.6*K_{cr}	$\frac{P_{cr}}{2}$	$\frac{P_{cr}}{8}$

Simulation diagram for Ziegler-Nichols 2nd method is shown in Fig. 2. Here, for our system, Critical gain, K_{cr} = 3.479 and Ultimate period, P_{cr} = 0.16 sec.

Table 5. Rules of thumb for tuning PID controller gains with Ziegler Nichols method of sustained oscillation [38]

Action	Rise time	Over-shoot	Settling time	Stability
Increase K_P	Faster	Increase	Small change	Gets worse
Increase T_d	Slower	Decrease	Decrease	Improves
Increase $1/T_i$	Faster	Increase	Increase	Gets worse

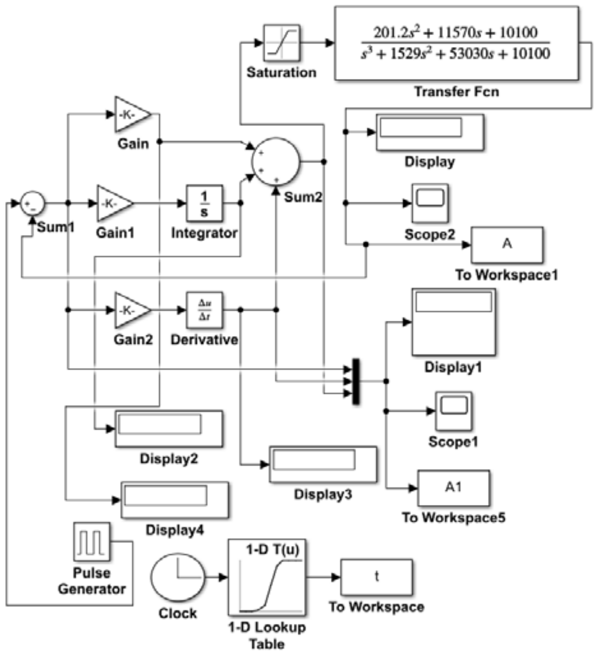

Fig. 2. Simulation diagram of Ziegler-Nichols tuning method for conventional Proportional-Integral-Derivative controller

By applying this tuning method, the parameters for this motor control are found as K_p = 2.08, K_i =26.09 sec and K_d = 0.042 sec.

Genetic Algorithm

In the Genetic Algorithm (GA), there are some evolution parameters available such as reproduction, crossover, along with mutation in order to find out the best possible solution. GA ignores local minima points of solution and converges to sub-optimal solutions. By adopting this method, GA is able to locate areas with high-performance in complex domains. The significant differences of GA with conventional methods are [39]:

1. GA looks for a population comprising of several parallel points.
2. GA does not provide information regarding derivative or other required auxiliary knowledge. Objective function along with similar stages of fitness influence the searching direction.
3. GA utilizes rules related to probabilistic transition. It does not utilize deterministic rules.
4. GA can sometime produce several potential probable solutions of a particular problem. User makes the final choice.
5. GA influences the encoding of a set of parameters but does not affect that particular parameter set.

Reproduction

During period of reproduction, fitness value of every chromosome is calculated. Fitness is necessary in the procedure of selection to produce fitter individuals. Chromosome which is fit contains a greater opportunity of reproduction. A standard selection technique called “Roulette Wheel” selection method where each of the individual is assigned a portion of the wheel called roulette wheel [42]. The individual gets fitter with the bigger size of the particle. At first, a pointer is rotated, and indicated individual is selected. This process keeps on going until the criteria regarding the reproduction or selection is satisfied.

Different versions of a particular string can be nominated for reproduction. The stings which are fitter, dominate over other stings. There remain four standard methods of selection:

1. Roulette Wheel selection
2. Stochastic Universal sampling
3. Tournament selection
4. Normalized geometric selection

Due to complexities associated in other three methods, roulette wheel selection method is considered.

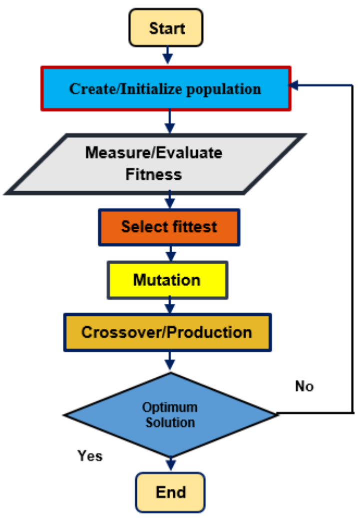

Fig. 3. Flowchart of Genetic Algorithm (GA) process

Crossover

After selection or reproduction procedure, crossover algorithm is applied. Crossover focuses on swapping specific portions of selected strings to retain certain portions of old chromosomes to create better offspring. Characters of a chromosome is influenced directly by operators. The probability of crossover indicates the frequency of crossover. For example, probability of 10% indicates “offspring” will receive 90% characteristics of “parents” and 100% probability indicates that every generation will be comprised of completely fresh new offspring.

There are 4 types of crossover.

1. Single point crossover
2. Multi-point crossover
3. Uniform crossover
4. Arithmetic crossover

In case of arithmetic crossover, integer representations are mostly used. Arithmetic crossover operates by considering the weighted average of the two available parents. Following formulae are used for the arithmetic crossover.

$Child_1 = \alpha.\ x + (1-\alpha).\ Y$. $Child_2 = \alpha.\ x + (1-\alpha).\ y$

For α = 0.5, both children will become indistinguishable and the procedure is shown in the Fig. 4. Arithmetic crossover is used in this paper.

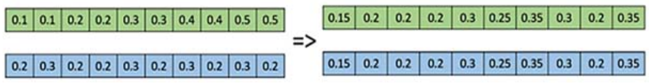

Fig. 4. Illustration of arithmetic crossover

Mutation

With the assistance of both reproduction and crossover, a significant amount of various strings is generated. But after reproduction and crossover, two fundamental problems are still present in the GA.

1. Based on the initially selected population, it should be ensured that there is enough amount of diversity present in the initially considered strings to ensure that GA should look for the optimum solution in the entire problem space.
2. On sub-optimum string points, GA may converge owing to choice of poor initially considered population.

Mutation can solve these problems of GA. Occasional random alteration of a particular value in a position of the string is called mutation. Mutation probability is usually kept at some low value because a rising rate of mutation would be destructive for fit strings which are fit and would not allow GA a random search [10].

Using mutation probability values of 0.1% or 1% is a common practice. After a specific string is chosen for mutation process, a randomly selected element of that string is altered, or can be called “mutated”. If bit position 4 is determined for mutation in a binary string value of 10000, the resulting new offspring string value becomes 10010 because of the fourth bit position flipping. In this paper, ITAE has been applied for the betterment of the dynamic response.

(41) $$ITAE = \int_0^\infty |e(t)|\, t.dt$$

Table 6. Selected parameters of Genetic Algorithm

GA Property	Value/Method
Size of population	50
Fitness function	ITAE
Process of selection	Roulette Wheel
Method of crossover	Arithmetic
Mutation Probability	0.01
Generation	100

Results of the Implemented GA-PID Controller

GA begins with initial population consisting several chromosomes. Every chromosome produces a solution. The aim is to develop new future individuals who can be able to provide better performance than previous generation. GA is repeated for several generations. GA stops when the algorithm produces individual that provides optimum solution [30]. To evaluate performance of GA, fitness function is used.

In this paper, initially, GA-PID controller with a population of 20 is simulated. After that, simulations are performed with population size of 25,30,35,40,45,50 and 60 and compared among each other. The analyzing parameter for the response of the GA-PID are the reduction of overshoot, improvement of rise time, and reduction of settling time. Plot of fitness function of GA based PID controller considering the population size as 50 is shown in Fig. 5.

Optimum measurements of controller gains are found as K_P = 1.995, K_I = 26.9 and K_D = 0.03.

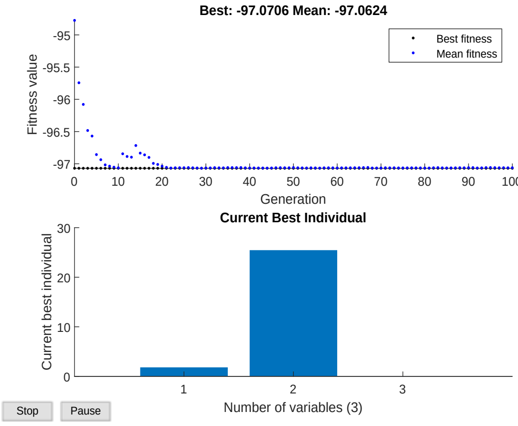

Fig. 5. Plot of fitness function with respect to generation and optimum gains of GA-PID controller

Flower Pollination Algorithm

Biological Inspirations

The angiosperms are considered as the biggest species of plants which could exist since the period of late Cretaceous (around 150 million years ago). The cause of this survival is optimized pollination process. Flower is a structure which holds the organs of reproduction. Flowers which are unisexual or imperfect contain both stamens (male) as well as carpels (female) in separate flowers. On the other hand, flowers which are bisexual or perfect bears both organs in that particular flower. Pollination is a procedure where pollen is transferred to the stigma so that fertilization of the female gamete can be influenced [17]. Flowers can possess attractive attributes to attract various animals, different insects, or birds. A few flowers attract pollinators to visit them frequently with the above-mentioned attributes and keep flower constancy intact.

Based on the pollination types, pollination can be divided into two groups.

i. Cross-pollination: Pollen is shifted from one flower to another flower of a different plant.

ii. Self-pollination: Pollen is shifted from one flower to different flower of that particular plant.

Based on pollinators, pollination can be classified as well.

i. Biotic-pollination: At the time of animal or insect visiting flower in order to eat pollen or sip nectar, biotic pollination takes place. While eating or sipping, pollen gets connected to the body of the visitor. If the animal or insect provides a visit to another flower, pollen gets transferred to the stigma of the flower and fertilization can occur. Around 90% flowering plants adopt this pollination. Because of the movement and flying of the pollinators with different speeds, pollen can shift to quite long distance. This kind of pollination can be regarded as the global pollination considering the effect of Levy flights [10].

ii. Abiotic-pollination: Abiotic-pollination contains a property of limited occurrence because pollen is shifted from a flower to another with the assistance of wind, water diffusion or gravity. About 10% of the flowers utilize this kind of pollination. The distance covered by the pollen is typically short. Abiotic-pollination lacks in precision.

Rules of Flower Pollination Algorithm

- Rule 1: Global Pollination occurs if both biotic pollination as well as cross-pollination takes place. For global pollination, pollinators follow Levy flights.
- Rule 2: Local Pollination is considered to occur if abiotic as well as self-pollination takes place.
- Rule 3: Probability of reproduction which is proportional to the resemblance between any two flowers is called flower constancy.
- Rule 4: Probability of switching is indicated by $p \in [0, 1]$. Switching probability can be regulated between local as well as global type pollination

because of few externally affecting factors, for example water, wind etc. [40].

Global Search of FPA (Biotic)

Global search of FPA begins with the generation of random initial population to find out current best solution. In order to find out a new solution of global search FPA, at first, the type of pollination needs to be obtained according to a predetermined probability p (Rule 4). For example, a random number is taken into consideration indicated by $r \in$ [0, 1]. If $r < p$, global pollination occurs. If $r > p$, local pollination takes

place. The flower constancy which is related to Rule 1 as well as Rule 3 are expressed in (42):

(42) $$x_i^{t+1} = x_i^t + \gamma L(\lambda)(x_i^t - g^*)$$

x_i^t indicates a solution i at time t and g* is considered the current best solution. scaling factor is indicated by γ , and step size is symbolized by L which is derived from Levy flight according to the following Levy distribution with the condition L> 0:

(43) $$L(s,a) = \frac{\lambda\Gamma(\lambda)\sin(\pi\lambda/2)}{\pi}\frac{a}{S^{1+\lambda}}, |s| \to \infty$$

$\Gamma(\lambda)$ is a standard function of gamma function along with index λ. Control parameter of distribution is indicated by "a" which stands for the tail amplitude and a = 1 is considered in the proposed FPA [3]. In accordance with nonlinear transformation popularly known as Mantegna algorithm, a large step ($s >> s_0 > 0$) is generated:

(44) $$S = \frac{U}{|V|^{\lambda-1}}$$

In (49), U along with V are regarded as two Gaussian random values achieved from a Gaussian distribution where mean is equal to zero. Standard deviations are denoted by σ_u and σ_v. These standard deviations cannot be chosen with independence for any random value. The value of σ_v is set to 1.

(45) $$U \sim (0,\sigma_u^2), V \sim (0,\sigma_v^2)$$

(46) $$\sigma_u = \left[\frac{\Gamma(1+\lambda)}{\lambda\Gamma((1+\lambda)/2)}\frac{\sin(\pi\lambda/2)}{2^{\frac{(\lambda-1)}{2}}}\right]^{\frac{1}{\lambda}}, \sigma_v = 1$$

In the simulations of this paper, λ = 1.5 is used.

Local Search of FPA (Abiotic)

Constancy of flower has emerged from this particular concept. Specific set of flower types are only visited by pollinators. In this way, utilization of energy is performed because they do not have to explore new types of flower. The plants with flowers provide sufficient amount of reward in the form of nectar to the pollinators which increases the visiting frequency of pollinators and thus reproduction success is maximized [41].

For $r > p$, local pollination takes place. The local pollination which is described in Rule 2 along with Rule 3 can be represented by (52):

(47) $$x_i^{t+1} = x_i^t + \varepsilon(x_j^t - x_k^t)$$

Here, x_k^t and x_j^t are two randomly selected solutions such as two pollens from different flowers of a particular plant. x_k^t and x_j^t appears from identical species or chosen from alike population. This is accurately considered a local pollination if derived from a uniform distribution in [0,1]. Flower pollination can be local or global but adjacent flowers positioned in the closest neighbors have better chances to be pollinated by the pollen of local flower pollen than those which are situated at distance. To elaborate this statement, switching probability mentioned in Rule 4 or the proximity probability, *p* responsible for switching between global and local pollination can be used. At the beginning, initial value of *p* = 0.5 was considered. Further changing the value of *p*, it can be observed that *p* = 0.8 works better.

Table 7. Selected Flower Pollination Algorithm parameters

FPA Parameters	Value
Size of population	50
Switching probability	0.8
Dimension	3
Iteration	100

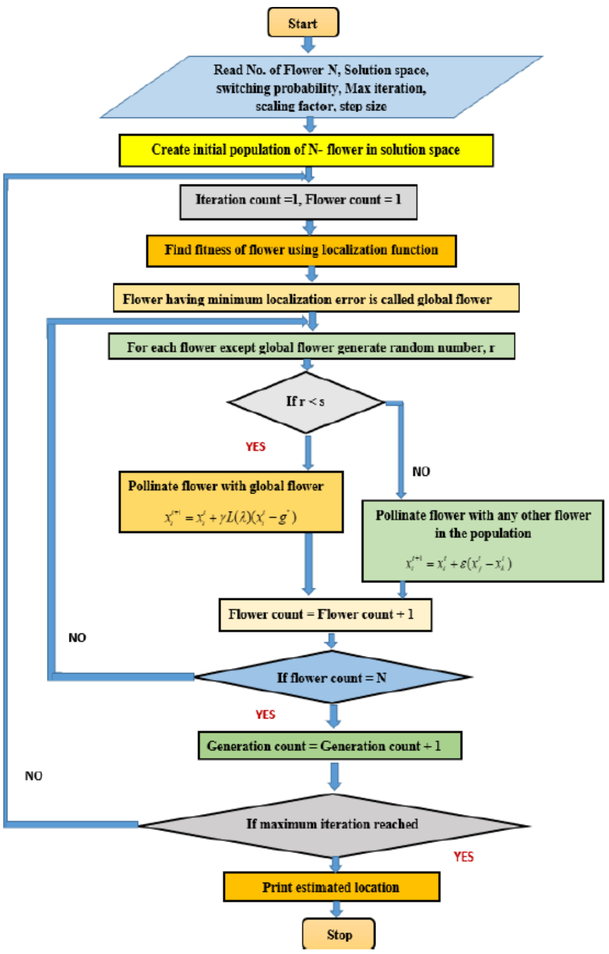

Fig. 6. Flowchart of Flower Pollination Algorithm

Table 8. Performance of Controllers

Controllers	Rise time (s)	Settling time (s)	Overshoot (%)
ZN-PID	0.187	0.862	15.6
GA-PID	0.117	0.431	9.65
FPA-PID	0.11	0.391	4.87

In this paper, flower pollination algorithm based PID controller is designed to obtain improved dynamic response.

Simulation Results

From the results mentioned in Table 8, gained from simulation as well as detailed discussion in this paper, the following result comparison points can be stated.

- The Genetic Algorithm based PID controller (GA- PID) provides 1.6 times better rise time, 2 times better settling time and 1.62 times better overshoot than the Ziegler-Nichols sustained oscillation based PID controller (ZN-PID)
- Flower Pollination Algorithm based PID controller (FPA-PID) produces 1.7 times improved rise time, 2.21 times improved settling time and 3.2 times better overshoot than Ziegler-Nichols sustained oscillation based PID controller (ZN-PID).
- FPA-PID provides 1.064 times better rise time, 1.1 times improved settling time and 1.98 times improved overshoot than Genetic Algorithm based PID controller (GA-PID).

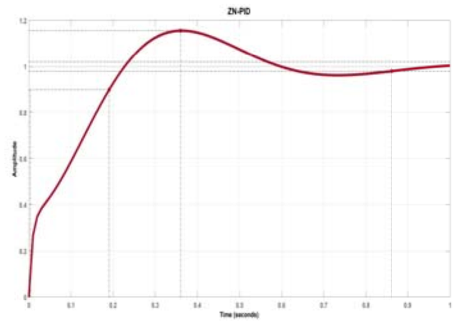

Fig. 7. Transient response of Ziegler-Nichols sustained oscillation based PID controller

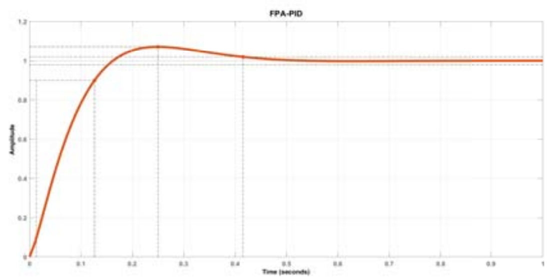

Fig. 8. Transient response of GA-PID controller

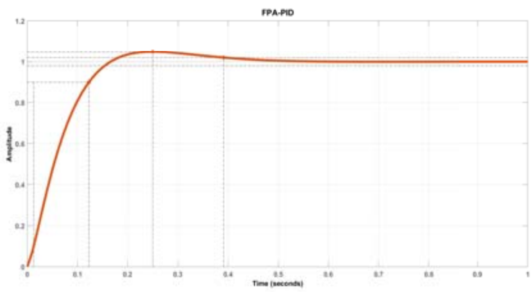

Fig. 9. Transient response of FPA-PID controller

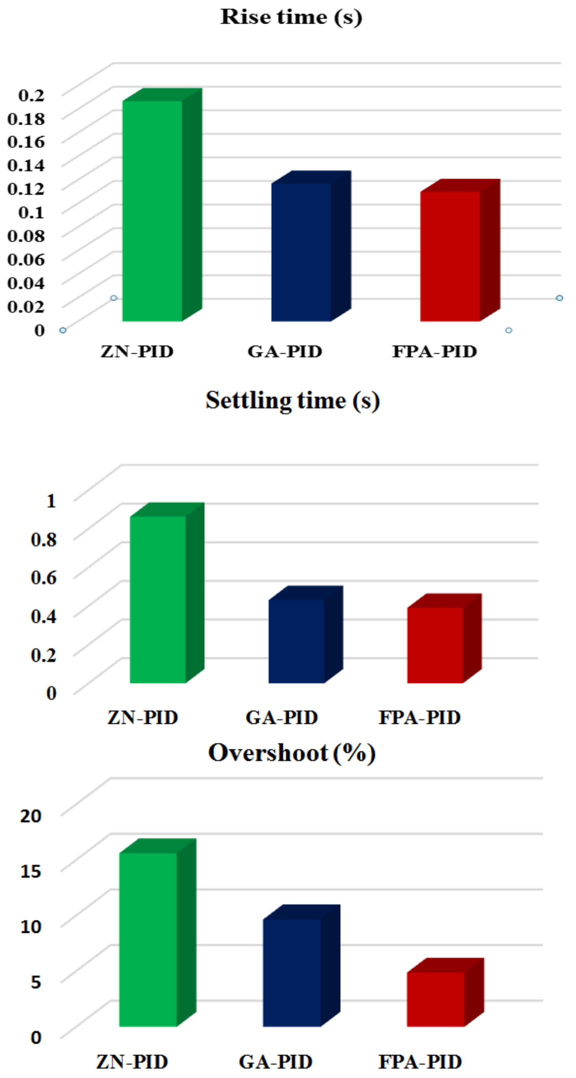

Figure 10. Graphical Representation of performance of controllers.

Conclusion

DC commutator motor is used for the propulsion of icebreaking ships because they are flexible and less complex. For analyzing improved performance of the motor mounted in a ship, designing appropriate controllers to satisfy design criteria for dynamic response is necessary so that the ship can be controlled in a better way. For the better transient response using the optimized value of the controller, improved speed and steering control along with safe movement of the ship can be ensured. Ziegler-Nichols 1st method and sustained oscillation method are utilized to find out optimized parameters of gain for the conventional PID controller. In order to achieve a better dynamic response, Genetic Algorithm based tuning for PID controller (GA-PID) is also applied. After that, one of the algorithms inspired from nature, known as Flower Pollination Algorithm (FPA) is implemented for tuning PID controller to improve the transient or dynamic response of the motor. Research tool for this work is MATLAB and Simulink, where simulations are done, and appropriate behaviors regarding each controller are observed. In fine, the controllers are compared among themselves depending on their transient response improvement performance. The simulation results show that, for step input which is essential for steering task in the ship, Flower Pollination Algorithm based on PID controller (FPA-PID) has better performance over other controllers producing desired and improved transient response. This means that the FPA-PID is adequate for the steering task of the Direct Current commutator motor actuator mounted in the ship.

Authors: *Shameem Hasan, University of Liberal Arts Bangladesh (ULAB)House - 56, Road - 4/A, Satmasjid Road, Dhaka 1209, Bangladesh.*
Golam Sarowar, Islamic University of Technology (IUT), Boardbazar, Gazipur - 1704, Bangladesh.

REFERENCES

[1] A. Andrzejewski "Time-Optimal Position Control of DC Motor Servo Drive" *Przegląd Elektrotechniczny*, 12/2019, R. 95, pp. 85-88.
[2] K. Ogata K., "Modern Control Engineering", 4th Edition, 2015, Dorling Kindersley Pvt. Ltd., India.
[3] R. C. Dorf, R.C. and R.H. Bishop, "Modern Control Systems", 9th edition, Prentice-Hall, Inc., Upper Saddle River, NJ, 2001.
[4] Manafeddin Namazov-Onur Basturk, "DC motor position control using fuzzy proportional-derivative controllers with different defuzzification methods", Turkish Journal of Fuzzy Systems, Vol. 1 (2010), No.1, pp. 36-54.
[5] N. S. Nise, "Control System Engineering", John Wiley and Sons Inc., New York, 2011.
[6] Md. M. Gani, Md. S. Isalm, Muhammad A. Ullah, "Modeling and Designing a Genetically Optimized PID Controller for Separately Excited DC Motor," *International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 2019.
[7] B. Kada, A. Elkebir, K. Negadi, Hafida Belhadj and D. E. Chaouch "Application of Adaptive Controller Neural Network Based on RBF NN for Temperature Control Electrical Resistance Furnace", *Przegląd Elektrotechniczny*, 6/2020, R. 96, pp. 33-38.
[8] J. G. Ziegler and N. B. Nichols, "Optimum Settings for Automatic Controllers", *Transactions on ASME,* Vol. 64 (1942), No. 11, pp. 759–768.
[9] D. Karaboga, B. Gorkemli, C. Ozturk and N. Karaboga "A comprehensive survey: artificial bee colony (ABC) algorithm and applications", *Artificial Intelligence review, Springer*, vol. 42 (2014), pp. 21-57.
[10] J. M. Herrero, X. Blasco, M. Martinez and J. V. Salcedo, "Optimal PID Tuning with Genetic Algorithm for Non Linear Process Models." *15th Triennial World Congress*, *Elsevier*, pp. 31-36, 2002.
[11] J. Kennedy, "Particle swarm optimization", *Encyclopedia of Machine Learning, Springer*, pp. 760–766, 2011.
[12] S. Mirjalili, S. M. Mirjalili, A. Lewis, "Grey wolf optimizer", *Advances in Engineering Software*, vol. 69 (2014), pp. 46-61.
[13] S. Arora, H. Singh, M. Sharma, S. K. Sharma, P. Anand, "A New Hybrid Algorithm Based on Grey Wolf Optimization and Crow Search Algorithm for Unconstrained Function Optimization and Feature Selection," *IEEE Access*, Vol. 7 (2019), pp. 26343-26361.
[14] S. Kalra and S. Arora, "Firefly algorithm hybridized with flower pollination algorithm for multimodal functions." *Proceedings of the International Congress on Information and Communication Technology*, *Springer*, pp. 207–219, 2016.
[15] X. S. Yang, "A new metaheuristic bat-inspired algorithm", *Nature Inspired Cooperative Strategies for Optimization*, *Springer*, pp. 65–74, 2010.
[16] A. H. Gandomi, X. S. Yang and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems", *Engineering with Computers*, Vol. 29 (2013), No. 1, pp. 17–35.
[17] X. S. Yang, "Flower pollination algorithm for global optimization", International Conference on Unconventional Computing and Natural Computation, *Springer*, pp. 240–249, 2012.
[18] A. Y. Al-Maliki and K. Iqbal, "PID-Type FLC Controller Design and Tuning for Sensorless Speed Control of DC Motor", *Advances in Science, Technology and Engineering Systems Journal,* Vol. 3 (2018), No. 6, pp. 515-522.
[19] Wahyudi, M. Rosalina, A. Ajulian, B. Winardi, "Self-Tuning Fuzzy PID Design for BLDC Speed Control" *GRD Journals – Global Research and Development Journal for Engineering*, Vol. 3 (2018), pp. 4-11.

[20] J. Pongfai and W. Assawinchaichote, "Self-tuning PID parameters using NN-GA for Brush DC motor control system" *14th International IEEE Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 111-114, 2017.
[21] H. S. Hameed, "Brushless DC Motor Controller Design Using Matlab Applications", *1st International Scientific Conference of Engineering Sciences – 3rd Scientific Conference of Engineering Science (ISCES)*, pp. 44-49, 2018.
[22] H. Chaudhary, S. Khatoon and R. Singh, "ANFIS Based Speed Control of DC Motor", *IEEE conference on Innovative Applications of Computational Intelligence on Power, Energy and Controls with their Impact on Humanity*, pp. 63-67. 2016.
[23] M. K. Rout, D. Sain, S. K. Swain and S. K. Mishra, "PID Controller Design for Cruise Control System using Genetic Algorithm", *IEEE International Conference on Electrical, Electronics, and Optimization Techniques*, pp. 4170 - 4175, 2016.
[24] Z. Has, A. H. Muslim and N. A. Mardiyah, "Adaptive-Fuzzy-PID Controller Based Disturbance Observer for DC Motor Speed Control", *Proceedings of IEECSI*, 2017.
[25] N. P. Adhikari, M. Choubey and R. Singh, "DC Motor Control Using Ziegler Nichols and Genetic Algorithm Technique", *International Journal of Electrical, Electronics and Computer Engineering*, Vol. 1 (2012), No. 1, pp. 33-36.
[26] P. M. Meshram and R. G. Kanojiya, "Tuning of PID Controller using Ziegler-Nichols Method for Speed Control of DC Motor", *IEEE- International Conference On Advances in Engineering, Science and Management*, pp. 117-122, 2012.
[27] Y. A. Almatheel and A. Abdelrahman, "Speed Control of DC Motor Using Fuzzy Logic Controller", *IEEE International Conference on Communication, Control, Computing and Electronics Engineering*, pp. 1-8, 2017.
[28] M. Abdel-Basset, L. A. Shawky, "Flower Pollination algorithm: A comprehensive review", *Artificial Intelligence Review, a part of Springer Nature*, 2018.
[29] H. Bouaziza, M. Berghidab, A. Lemouaric, "Solving the generalized cubic cell formation problem using discrete flower pollination algorithm," *Expert Systems with Applications, Elsevier*, Vol. 150 (2020), pp. 1-13.
[30] M. Abdel-Baset and I. M. Hezam, "An Effective Hybrid Flower Pollination and Genetic Algorithm for Constrained Optimization Problems", *Advanced Engineering Technology and Application*, No. 4, pp. 27–34, 2015.
[31] P. Kielan, D. Mazur, A. Szklarz, "BLDC motor control in HiL configuration with the use of Matlab/Simulink software and PLC", *Przegląd Elektrotechniczny*, 06/2017, R. 93, pp. 85-88, 5-8.
[32] Modeling of DC Motor - National Chiao Tung University,http://ocw.nctu.edu.tw/course/dssi032/DSSI_2.pdf.
[33] Y-P Chen, "Modeling of DC motor", Spring Course on Dynamic System Simulation and Implementation, Department of Electrical and Computer Engineering, NCTU, 2015.
[34] Has Z., Habibidin A. Muslim and Mardiyah N. A., "Adaptive-Fuzzy-PID Controller Based Disturbance Observer for DC Motor Speed Control", *Proceedings of IEECSI*, 2017.
[35] S. Hasan, "Improvement of Transient Response of a DC Motor Controller Based on Non-Linear Optimization Techniques", *International Journal of Engineering and Technical Research*, Vol. 8 (2018), No. 11, pp. 08-13.
[36] G. Sarowar and S. Hasan, "Enhancement of Cruise Control System by Improving Transient Response of Motor Controller using Non-Linear Optimization Techniques", *International Journal of Computer Science and Network Security (IJCSNS)*, Vol. 18 (2018), No. 12, pp. 145-152.
[37] D. Yousri, A. Mabrouk, Lobna A. Said, A. G. Radwan, "Biological Inspired Optimization Algorithms for Cole-Impedance Parameters Identification", *International Journal of Electronics and Communications,* Vol. 78 (2017), pp. 1-33.
[38] W. M. Elsrogy, M. A. Fkirin, M. A. M. Hassan. "Speed control of DC motor using PID controller based on artificial intelligence techniques", *IEEE International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 196-201, 2013.
[39] D. C. Meena and A. Devanshu. "Genetic algorithm tuned PID controller for process control", *IEEE International Conference on Inventive Systems and Control (ICISC),* pp. 1-6, 2017.
[40] M. Abdel Basset, I. El Henawy and N. Diab. "Optimal orbital elements of binary stars based on flower pollination algorithm", *International Journal of Mathematical Modelling and Numerical Optimisation*, Vol. 9 (2019), No. 1, pp. 56-69.
[41] C. Jin, K. Gu, Silviu-I. Niculescu, I. Boussaada, "Stability Analysis of Systems with Delay-Dependent Coefficients: An Overview," *IEEE Access*, Vol. 6 (2018), pp. 27392 – 27407.
[42] Godem Ali M. Ismeal, K. Kyslan and V. Fedák, "CAD of Cascade Controllers for DC Drives Using Genetic Algorithm Methods", *Procedia Engineering*, Vol. 96 (2014), No. 96, pp. 182-189.