energies

Article

Hybrid Gray Wolf Optimization–Proportional Integral Based Speed Controllers for Brush-Less DC Motor

Shukri Mahmood Younus Younus [1], Uğurhan Kutbay [1], Javad Rahebi [2,*] and Fırat Hardalaç [1]

[1] Electrical & Electronics Department, Gazi University, Ankara 06570, Turkey
[2] Software Engineering Department, Istanbul Topkapi University, Istanbul 34087, Turkey
* Correspondence: cevatrahebi@topkapi.edu.tr

Abstract: For Brush-less DC motors to function better under various operating settings, such as constant load situations, variable loading situations, and variable set speed situations, speed controller design is essential. Conventional controllers including proportional integral controllers, frequently fall short of efficiency expectations and this is mostly because the characteristics of a Brush-less DC motor drive exhibit non linearity. This work proposes a hybrid gray wolf optimization and proportional integral controller for management of the speed in Brush-less DC motors to address this issue. For constant load conditions, varying load situations and varying set speed situations, the proposed controller's efficiency is evaluated and contrasted with that of PID controller, PSO-PI controller, and ANFIS. In this study, two PI controller are used to get the more stability of the system based on tuning of their coefficients with meta heuristic method. The simulation findings show that Hybrid GWO-PI-based controllers are in every way superior to other controllers under consideration. In this study, four case studies are presented, and the best-case study was obtained 0.18619, 0.01928, 0.00030, and 0.01233 for RMSE, IAE, ITAE, and ISE respectively.

Keywords: brushless DC motor; PID; GWO-PI; hybrid controller

Citation: Younus, S.M.Y.; Kutbay, U.; Rahebi, J.; Hardalaç, F. Hybrid Gray Wolf Optimization–Proportional Integral Based Speed Controllers for Brush-Less DC Motor. *Energies* **2023**, *16*, 1640. https://doi.org/10.3390/en16041640

Academic Editor: Lorand Szabo

Received: 30 November 2022
Revised: 9 January 2023
Accepted: 12 January 2023
Published: 7 February 2023

1. Introduction

Every market sector, including appliances, industrial control, automation, aviation, and others, uses brush-less DC motors. The main benefits of brush-less DC motors include their high levels of efficiency, extended useful lives, reduced noise levels, and changeable high-speed ranges [1,2]. Three main categories of brush-less DC motor operation control exist positioning applications, variable loads, and constant loads [3]. Variable speed is more crucial for constant load applications than maintaining the precision of speed at a constant speed. In these applications the load is directly connected to motor shaft. Blower, fan and pump applications are a few examples of this type. Such applications call for low-cost, primarily open-loop controllers [4]. The load placed on the motor changes depending on the applications' changing speed requirements. High-speed control precision and strong dynamic responses can be necessary for these applications. Washing machines, dryers, and compressors are a few examples of household appliances [5]. Most industrial and automation applications fall under the category of positioning applications. Applications falling under this category all use some form of power transmission, such as basic belt-driven systems, mechanical gears, timer belts, or timing belts [6]. Different types of controllers have been designed to satisfy the diverse application needs. The following section contains a literature review of various brush-less DC motor speed controller types.

To drive the fan, Akkaya et al. (2007) developed a Brush-less DC motor driving system using a proportional integral (PI) speed controller [7]. Both the intended starting response and the continuous time response are used by the process to calculate the proportional gain and the integral gain. Steady state and overshoot inaccuracy are significant features of the speed response, according to the simulation results. A proportional-integral (PI) controller used in a brush-less DC motor was implemented by Singh et al. (2010), Jin Gao & Yuwen Hu (2010), and others [8,9]. The PI controller creates a system slowdown and uncertainty issue when the Brush-less DC motor is operating in particular specific status. The application of proportional integral current regulation for Brush-less DC motors was explained by Karthikeyan (2011) [10]. The gain of the proportional integral controller was adjusted through examination and error. However, this strategy needs much time and in unable to yield the excellent value of the controller's gain. Joice et al. (2013) have demonstrated that a proportional integral (PI) speed controller can be used with brush-less DC motors [11]. The PI controller's gain was selected using a trial-and-error methodology. Essentially, the examination and error approach is a typed of technique that required so much and is also unable to provide correct gain values in PI controllers. The following methods are effective in resolving the PI controller tuning issue. Metin Demirtas (2011) [12] provided a description of the offline tuning of a PI speed controller in Brush-less DC motor by utilizing a genetic algorithm. However, there are greater fluctuations during the steady state phase and a higher overshoot in the speed response during the transient period. Ibrahim et al. (2014) performed the PID speed controller tuning used in a Brush-less DC motor using PSO and Bacterial Foraging [13]. However, the PID controller has been tailored to a certain motor's speed and torque status. The parameters referring to time domain and indices referring to performance fluctuate substantially and the motor's performance suffers if operating conditions exceed the predetermined level.

For brush-less DC motors, Rubaai et al. (2008) created a hybrid fuzzy PID and traditional PID speed controller [14]. The fuzzy PID controller is activated whenever there is an abrupt change in the set speed or load. Under typical operational circumstances, the PID controller acts independently. For this control method to detect how the Brush-less DC motor's working circumstances were changing, an external component was required. Tan Chee Siong et al. in 2011 [15] created a controller based on fuzzy proportional derivative used in the Brush-less DC motor. When there is a fast shift in load, the speed response increases undershoot and overshoot values. Moreover, the speed response contains a higher steady state error under steady state conditions. The PI speed controller based on fuzzy in a Brush-less DC motor was described by Shyam & Daya (2013) [16]. The fuzzy PI controllers and the PI controller's efficacy were investigated. The fuzzy PI controller, on the other hand, has caused an uncertainty problem as a result of sudden shifts in set speed and load. Abdullah et al. (2014) created a PID controller with high sensitivity by utilizing a fuzzy logic controller and also a radial based on function network [17]. The PID controller with fuzzy logic tuning outperforms the PID controller with radial function network tuning in terms of performance. However, due to unexpected load variations, a problem of uncertainty was created by the fuzzy tuned PID controller. The time domain specification parameters were also been weakened by it. Binod Kumar Sahu et al. (2015) [18] created a fuzzy-PID controller using an optimization method which is based on teaching-learning. The multi-area power system's load frequency was controlled using the created controller. However, the designed controller only reduces the transient performance characteristics while improving the steady state performance. The result is greater rise time, undershoot, and overshoot.

The speed controller for Brush-less DC motors based on Radial based on Function Neural Networks (RBFNN) has been described by Yingfa Wang (2007) [19]. The RBFNN's hidden layer's structure and parameters are tweaked offline using a genetic algorithm. Therefore, the controller requires a high number of representative data pairs, and the training process takes longer. Due to a sudden load disruption, the controller also has an uncertainty issue. Zhiqiang Cheng (2009) [20] created a brush-less DC motor sliding mode for controlling speed based on neural networks. The RBFNN controller and sliding mode controller were combined to create the controller. Sliding mode controllers frequently experience chattering issues, whereas RBFNN controllers require training data pairs. Additionally, this caused the Brush-less DC motor to respond slowly. Sinanc et al. (2014) [21] presented a speed controller which uses artificial neural networks. An offline training method is utilized to train the neural network. Generally, training offline algorithms is time consuming and takes much training data.

For identifying Brush-less DC motors, Faieghi et al. (2010) described an intelligent agent-based Adaptive Neuro-Fuzzy Inference System (ANFIS), which performs Non-linear Auto-Regressive Moving Average with Exogenous Input (NARMAX) system [22]. Using particle swarm optimization, PID speed controller is optimized. The created controller was evaluated under conditions of constant loading. Nevertheless, a Brush-less DC motor's speed response has a greater overshoot, undershoot, and steady state inaccuracy. The description of a speed controller based on ANFIS for Brush-less DC motor was provided a by Varatharaju et al. (2011) [23]. The offline training mode was used to teach the ANFIS controller. ANFIS controller and traditional proportional integral controller were contrasted. However, the ANFIS controller causes a great steady state inaccuracy in Brush-less DC motor speed. An ANFIS controller based with online supervised proportional derivative was developed to manage the speed of Brush-less DC motors [24]. The proportional derivative controller was used to modify the Neuro fuzzy controller's output layer gain. The system performance has been negatively impacted by excessive overshoot, a long settling time and high steady state error because of tweaking the proportional and derivative gains. A speed controller based on fuzzy neural networks was created by Gu drying et al. (2014) for the Brush-less DC motor [25]. The designed controller and a traditional PID controller were compared. However, this controller has generated uncertainty issues because of changes in set speed and the load. According to several research on the controller for Brush-less DC motor cited and already published, it is obvious that there is much room for further investigation. As a result, the stated work on creating controllers to improve Brush-less DC motor performance has been taken up.

2. Materials and Methods

The benefits of a proportional integral (PI) controller are a straightforward formation as well as an affordable controller. In contrast, the PID controller cannot deliver the optimal performance when used with a non-linear system because of the non-linear feature of the Brush-less DC motor. For industrial applications, it is crucial to design high-performance brush-less DC motor drives. Dynamic speed command tracking and load regulating responsiveness are essential for these drives. The drive should also have excellent integral performance indices and time domain parameters. In order to improve the performance of the Brush-less DC motor, a Hybrid GWO-PI-based speed controller is investigated in this paper for speed control of a Brush-less DC motor. The objectives of the paper are as follows:

- To create a Brush-less DC motor simulation model in MATLAB/Simulink Toolbox for various control strategies.
- To create various hybrid GWO-PI speed controllers for Brushless DC motors.
- To make a recommendation for the most efficient controller based on the parameter contents considered and acquired features of speed response for a variety of operating cases for a Brush-less DC motor.

The following include the contributions to this paper:

- To design the PID controller to be able to control speed in the Brush-less DC motor.
- To design speed controller based on PSO-PI utilized in a Brush-less DC motor.
- To design controller based on Adaptive Neuro Fuzzy Inference system for Brush-less DC motor.
- To design Hybrid GWO-PI-based speed control of Brush-less DC motor.

In this study the three-phase permanent magnet synchronous machine with sinusoidal model is used.

2.1. Speed Control in Brush-Less DC Motors

Figure 1 depicts the scheme of Brush-less DC motor's speed control. A three-phase star-connected Brush-less DC motor can be described using the five equations listed below, numbered from Equations (1)–(5):

$$v_{ab} = R(i_a - i_b) + L\frac{d}{dt}(i_a - i_b) + e_a - e_b \tag{1}$$

$$v_{bc} = R(i_b - i_c) + L\frac{d}{dt}(i_b - i_c) + e_b - e_c \tag{2}$$

$$v_{ca} = R(i_c - i_a) + L\frac{d}{dt}(i_c - i_a) + e_c - e_a \tag{3}$$

$$T_e = k_f\omega_m + J\frac{d\omega_m}{dt} + T_L \tag{4}$$

$$\omega_r = \frac{d\theta_r}{dt} \tag{5}$$

where v_{ab}, v_{bc}, and v_{ca} are the volts representing the phase-to-phase voltage. i_a, i_b, and i_c in amperes are used to represent the stator winding's phase currents. In Henry, L stands for the motor's self-inductance. The volt symbols e_a, e_b, and e_c stand for the back electromagnetic force. The terms T_e $(N - m)$ and T_L $(N - m)$ refer to the motor's magnetic torque and load torque, respectively. J defines the inertia of the rotor, the frictional constant is given by k_f, the motor's rotor speed is given by r (rad/s) and r (rad) provides the rotor's position. Figure 1 indicates a system including two loops: (1) the inner loop, (2) the outer loop. The inner loop synchronizes the rotating position (situation) by the gate signal of the PWM inverter with utilizing a switching logic circuit and a hall sensor. The motor's real speed is sensed by the outer loop, which also utilized to create speed errors by comparing it to the reference speed. Once the speed error has been analysed by the controller, it is used to regulate the switching logic, PWM inverter, and DC bus voltage that controls the speed of the Brush-less DC motor [26].

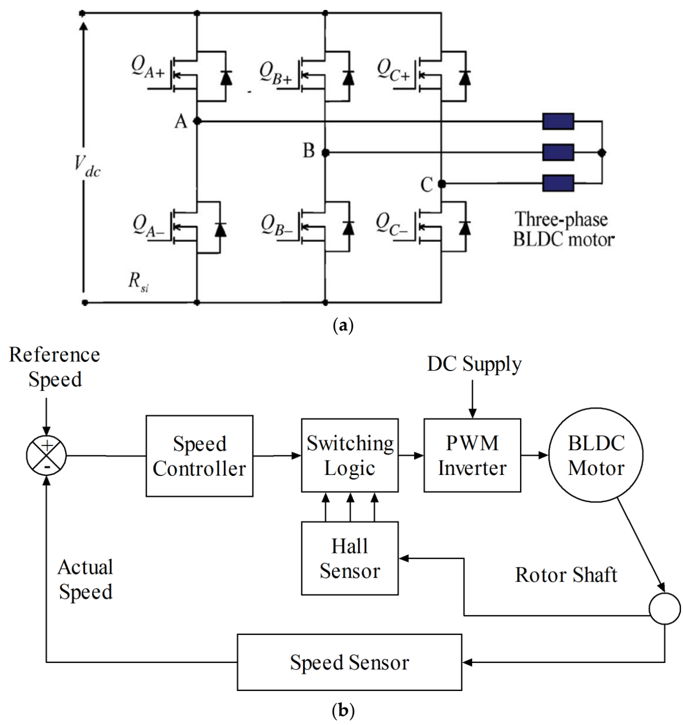

Figure 1. (**a**) Typical inverter systems for a BLDC motor, (**b**) Brushless DC motor's speed control system.

2.2. Controllers Used in Brush-Less DC Motor

For the speed control of Brush-less DC motors, four types of controllers have been investigated, i.e., PID controller, PSO-PI controller, controller based on Adaptive Neuro Fuzzy Inference System, and suggested Hybrid GWO-PI controller. Brief reviews of the above controllers are described in this section.

Conventional PID Controller

Figure 2 depicts the PID controller for the structure. Equation (6) expresses the PID controller's control output in time-domain as,

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de}{dt} \tag{6}$$

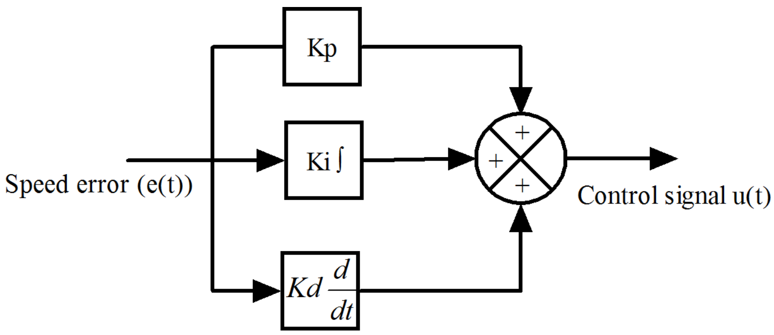

Figure 2. Formation of a conventional PID controller.

Here, the tracking speed error is given by $e(t)$, the control signal to the plant is given by $u(t)$, the proportional gain is defined by Kp, the integral gain is given by Ki, and the derivative gain of the PID controller is given by Kd. The tracking speed error, or the difference between the requested input material (ref) and the actual output, is shown by the equation $E(t)$ (act). The Simulink model for the PID controller is shown in Figure 3.

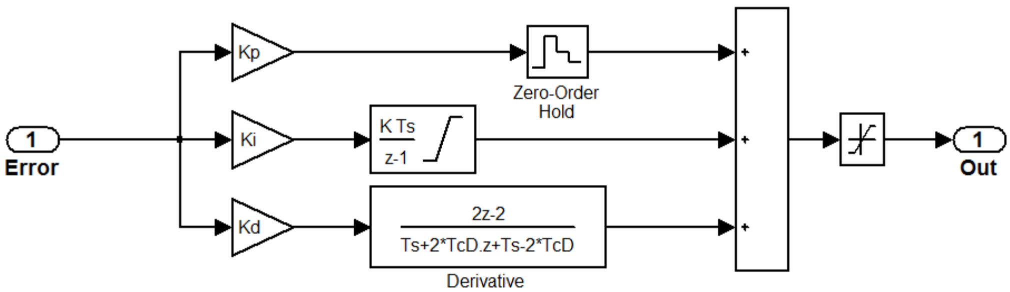

Figure 3. Simulink model of a conventional PID controller.

2.3. Proposed Gray Wolf PI Controller

GWO imitates the directorial hierarchies and also the hunting mechanisms that gray wolves exhibit in their own environment, essentially. Each pack is made up of alpha, beta, delta, and omega grey wolves. In addition, there are three steps in their hunting procedure: scouting, encircling, and attacking prey. All of these procedures are carried out concurrently with the optimization operation. GWO, a new and effective meta-heuristic technique, is offered by Mirjalili [27]. Since it is based on animals and nature, it is comprehended and practical to execute. The primary advantage of GWO is its adaptability, simplicity, and clarity. When compared to other well-known and effective meta-heuristic conceptions, a few recent research indicate that GWO may offer gratifying outcomes. For instance, this occurred when Mirjalili used 29 test functions in order to compare of GWO with the

Gravitational Search Algorithm (GSA), Differential Evolution (DE), PSO, Evolution Strategy, and Evolutionary Programming.

The application GWO algorithm to control of power system and machines are as follows, GWO optimization used for optimal power flow control in HVDC system [28], GWO applied overhead transmission line for parameter calculation of the transmission line [29], GWO was used in the tuning of PI controller of multi are load frequency control [30]. Figure 4 depicts the gray wolf's hierarchies.

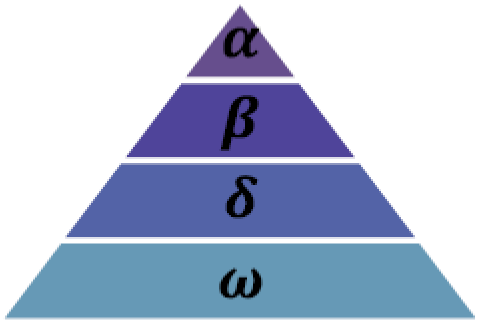

Figure 4. Hierarchies of the gray wolf [27].

Any optimization problem that postulates the best solution, or alpha (α), can be solved mathematically by illustrating the wolf's social hierarchy. The expressions "beta" (β) and "delta" (δ) refer to the second- and third-best solutions, respectively, while "omega" (ω) refers to other options.

2.3.1. Method for Hunting Gray Wolves

In explaining and teaching the gray wolf algorithm, we can say that this algorithm consists of 3 main steps:

Track, search, chase, and approach the potential prey.
Harass and encircle a prey until the prey no longer moves.
Attack the prey.

Figure 5a shows the effects of Equations (1) and (2), some possible neighbors, and a 2D positional vector. The mentioned figure also illustrates the wolf's position (X, Y) that changes/updates with respect to the prey's location (X*, Y*). The contents of $\vec{A}$ and $\vec{C}$ vectors can be adjusted to attain places close to the best agent. Here, (X* − X, Y*) serves as an example that we can reach by adjusting $\vec{A} = (1,0)$ and $\vec{C} = (1,1)$. In Figure 5b, a grey wolf's possible updated positions are depicted in a 3D space. It is important to understand that wolves can access any location among the points because of the accidental vectors $\vec{r_1}$ and, $\vec{r_2}$ as Figure 5 illustrates. Consequently, a gray wolf may update its location in any random location in the space surrounding the prey using Equations (1) and (2).

The attack is commanded by the alpha wolf as the prey is surrounded by wolves and do not move anymore. The reduction of the vector an is used to model this process. The coefficient vector A reduces as (a) lowers since it is an accidental vector in the range [−2a, 2a]. The wolf alpha will approach the prey (and the other wolves) if $|A| < 1$, and the wolf will avoid the prey (and the rest of the wolves) if $|A| > 1$. All wolves must update their positions in accordance with the positions of the alpha, beta, and delta wolves according to the gray wolf algorithm.

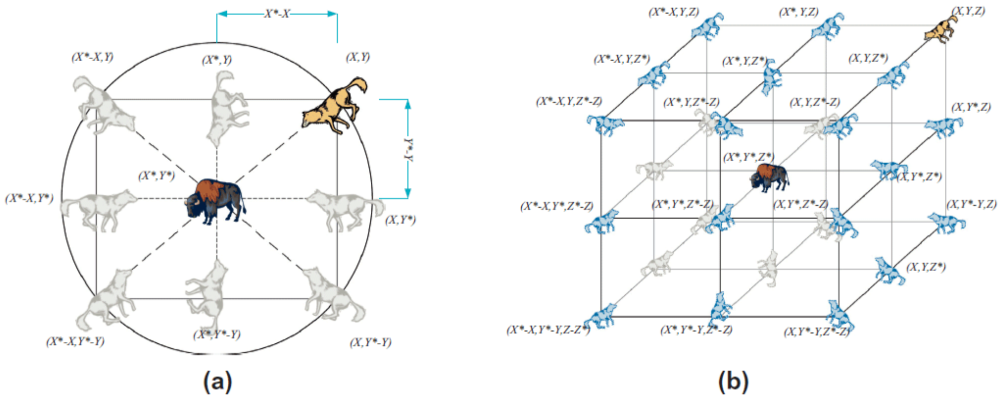

Figure 5. Gray wolf's structures to get prey, (**a**) 2D and (**b**) 3D position vectors [27].

During the hunt, gray wolves surround the predation. The equations that follow provide a mathematical representation of the siege behavior. In the relations below the current iteration t, A, and C are coefficient vectors, X_p is the prey position vector and X is the position vector of the gray wolf.

$$\vec{D} = \left|\vec{C}.\vec{X}_p(t) - \vec{X}_n(t)\right| \tag{7}$$

$$\vec{X}_n(t+1) = \vec{X}_p(t) - \vec{A}.\vec{D} \tag{8}$$

Vectors A and C are calculated as follows:

$$\vec{A} = 2\vec{a}.\vec{r}_1 - \vec{a} \tag{9}$$

$$\vec{C} = 2\vec{r}_2 \tag{10}$$

Here, r_1 and r_2 are the accidental vectors [0, 1] and after iterations, $\vec{a}$ components have been linearly reduced from 2 to 0.

Searching and excavation operations will be typically directed by alpha. Beta and delta

$$\vec{D}_\alpha = \left|\vec{C}_1.\vec{X}_\alpha - \vec{X}\right|,\ \vec{D}_\beta = \left|\vec{C}_2.\vec{X}_\beta - \vec{X}\right|,\ \vec{D}_\delta = \left|\vec{C}_3.\vec{X}_\delta - \vec{X}\right| \tag{11}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1.\left(\vec{D}_\alpha\right),\ \vec{X}_2 = \vec{X}_\beta - \vec{A}_2.\left(\vec{D}_\beta\right),\ \vec{X}_3 = \vec{X}_\delta - \vec{A}_3.\left(\vec{D}_\delta\right) \tag{12}$$

$$\vec{X}_n(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{13}$$

The search steps are shown in Figure 6.

To conclude, the search for an optimized solution, or in other words, prey, initiates with creating random grey wolves' population (or candidate solutions) using the GWO algorithm. After a considerable number of iterations, three top wolves (alpha, beta, and delta) estimate the prey's position. Based on that, every candidate solution (wolf) changes (updates) its distance from the selected prey. To explore and exploit further, we reduced the parameter a from 2 to 0. As $|\vec{A}| > 1$, the candidate solutions diverge from the prey, however when $|\vec{A}| < 1$ is true, the candidate solutions converge toward the prey. At the end, when the end criterion is met, we terminate the GWO.

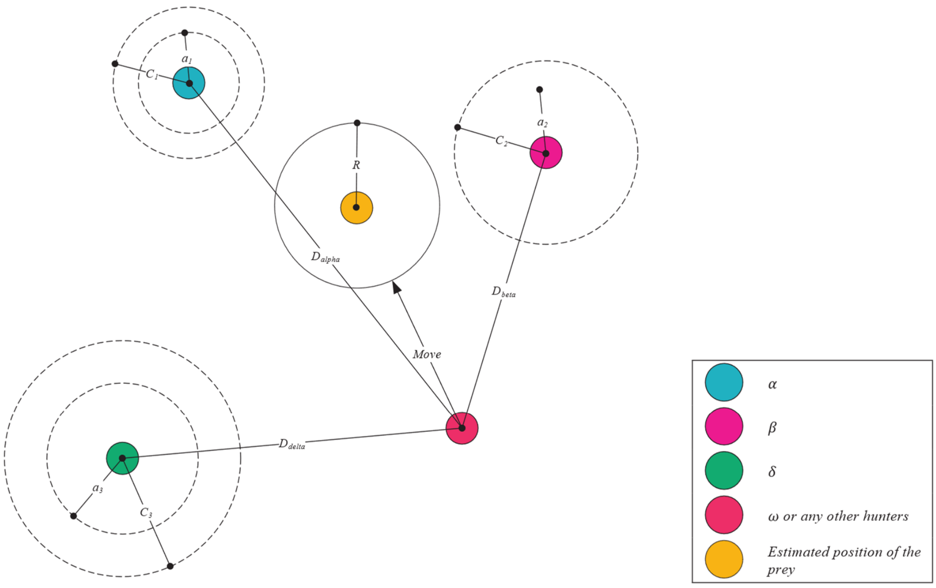

Figure 6. Search steps [27].

According to the contents of the flowchart, the gray wolf algorithm can be considered as follows. This flowchart will only work by specifying the values of vectors A and C. Explaining this flowchart is very simple by studying the above-mentioned steps. Figure 7 indicated the Flowchart of the Gray Wolf Algorithm.

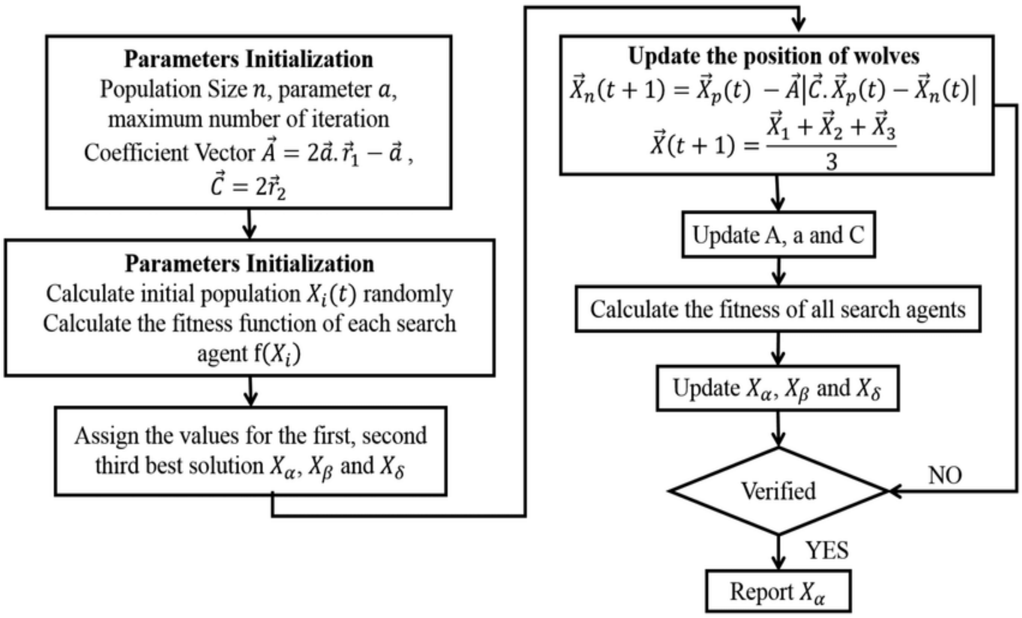

Figure 7. Flowchart of Gray Wolf Algorithm.

This looks and introduces the grey wolf optimizer into a unique set of rules to generate the global seek vector for decreasing the characteristic numbers from the fault indicators that created by guide enforcing in Simulink to attain the capabilities for global exploration. Numerous steps are contained in the gray wolf optimizer:

(1) Social hierarchy mechanism

According to their fitness values, the wolves are separated into 4 groups (W_1, W_2, W_3, and W_4). The first three groups have the ability to control the wolves and are made up of excellent adaptable gray wolves.

(2) Surround the prey

The grey wolves must surround their prey while predation. A mathematical model can be found here:

$$\vec{S} = \left| \vec{Q_2} \times \vec{Z_P}(x) - \vec{Z}(x) \right| \tag{14}$$

$$\vec{Z}(x+1) = \vec{Z_P}(x) - \vec{Q_1} \times \vec{S} \tag{15}$$

In which x represents the cutting-edge new release number. $\vec{S}$ is the distance and route of the wolf from its prey. $\vec{Z_P}$ is the location of the prey. $\vec{Z}$ gives the placement of the wolves. $\vec{Q_1}$ and $\vec{Q_2}$ are coefficient vectors and can be described as follows:

$$\vec{Q_1} = a(2 \times rand(0, 1) - 1) \tag{16}$$

$$\vec{Q_2} = 2 \times rand(0, 1) \tag{17}$$

In this equation a is factor of attenuation.

When the range of iterations rises, the coefficient experiences a linear decrease from 2 to zero.

(3) Hunting

The wolf W_1 instructs wolves W_2 and W_3 to reduce the prey's surrounding circle (domain) in order to gain the motive for hunting. The mathematical model is given as:

$$\begin{cases} \vec{S_{W_1}} = \left| \vec{Q_2} \times \vec{Z_{W_1}}(x) - \vec{Z}(x) \right| \\ \vec{S_{W_2}} = \left| \vec{Q_2} \times \vec{Z_{W_2}}(x) - \vec{Z}(x) \right| \\ \vec{S_{W_3}} = \left| \vec{Q_2} \times \vec{Z_{W_3}}(x) - \vec{Z}(x) \right| \end{cases} \tag{18}$$

$$\begin{cases} \vec{Z_1} = \vec{Z_{W_1}} - \vec{Q_1} \times \vec{S_{W_1}} \\ \vec{Z_2} = \vec{Z_{W_2}} - \vec{Q_1} \times \vec{S_{W_2}} \\ \vec{Z_3} = \vec{Z_{W_3}} - \vec{Q_1} \times \vec{S_{W_3}} \end{cases} \tag{19}$$

where, $\vec{Z_{W_1}}$ $\vec{Z_{W_2}}$, $\vec{Z_{W_3}}$ indicate the positions of wolf W_1, wolf W_2 and wolf W_3, respectively. $\vec{Z}$ gives the location and position information belonging to the remaining wolves. $\vec{S_{W_1}}$, $\vec{S_{W_2}}$, $\vec{S_{W_3}}$ indicate the rout and step size (step length) of wolf W_4 which moves toward wolf W_1, wolf W_2 and wolf W_3, respectively.

(4) Generation a global search vector

The global search vector is defined as the following:

$$\vec{V} = \vec{V} + rand \times (Z_1 + Z_2 + Z_3 - 3P_P(x)) \tag{20}$$

The improved position update formula is as follows:

$$P_p(\vec{x+1}) = \frac{P_p(\vec{x}) + P_p(\vec{x+1}) + \vec{V}}{3} \tag{21}$$

In Equation (15) the features from the created data matrix can be obtained. These features will be used in the training of the neural network.

2.3.2. Hybrid GWO-PI Controller

In this section, the development of H-GWO-PI controller is presented. Figure 8 shows the Simulink model for H-GWO-PI controller. In this figure, GWO tuner provides the gain contents for the PI controller regarding to the reference value and the actual value of error.

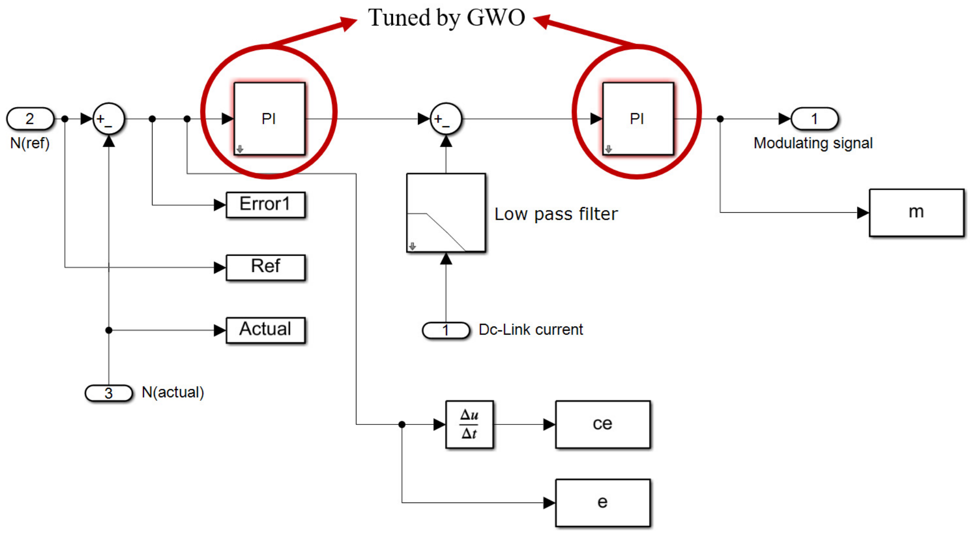

Figure 8. Simulink model for H-GWO-PI controller.

The speed response of the contemplated Brush-less DC motor is examined for fixed load situations, varying set speed situations and varying load situations. Time domain parameters and performance indices for the hybrid GWO-PI are obtained and compared, including recovery time, settling time, rising time, steady state error, undershoot, overshoot, RMSE, ITAE, IAE, and ISE PID controller, PSO-PI controller, ANFIS controller, hybrid GWO-PI. The BLDC motor's characteristics have been extracted from [26].

As seen in Figure 8, the Error1 is the input of the first controller (speed PI controller), and the difference between output of first Speed PI controller and the filtered current value is the input of the second current PI controller. The red colored circle and direction shows the PI controller tuned by GWO algorithm. The objective function is the error between reference and the actual value. The aim of this objective function is reducing the error value.

$$Objective\ Function$$
$$=abs(Referencespeed - actual\ Speed\ of\ the\ motor) + abs(Reference\ current - actual\ current\ of\ the\ bldc\ motor) \tag{22}$$

The absolute value of the error speed and error current is measured from the simulation for different set of values of the PI controller's parameter from the GWO algorithm. After completion of maximum iteration, GWO provide optimal parameter results for the PI controllers such as Kp1, Ki1, Kp2 and Ki2. The limits or constraints for the control variables such as Kp1, Ki1, Kp2, and Ki2 are between 0.01 (Lower bound) to 3 (Upper bound). The GWO provide the values for two PI controllers such as Kp1, Ki1, Kp2, and Ki2. For each set of population, measuring objective function value stated in the Equation (22). For each iteration, best results are stored separately, and end of the final iteration, GWO provide the optimal parameters for the two PI controllers.

3. Results and Discussion

A. Results for Constant Load Case

The specification BLDC motor are as follows, rated power is 10 KW, rated voltage is 48 V, rated speed is 1500 rpm and rated torque is 27 nm. A brush-less DC motor's speed response is simulated for both full load and no load conditions and the findings are provided in this section. For a set speed of 1500 rpm and no load case, Figure 9 depicts the curve of speed response. Table 1 lists the corresponding performance metrics.

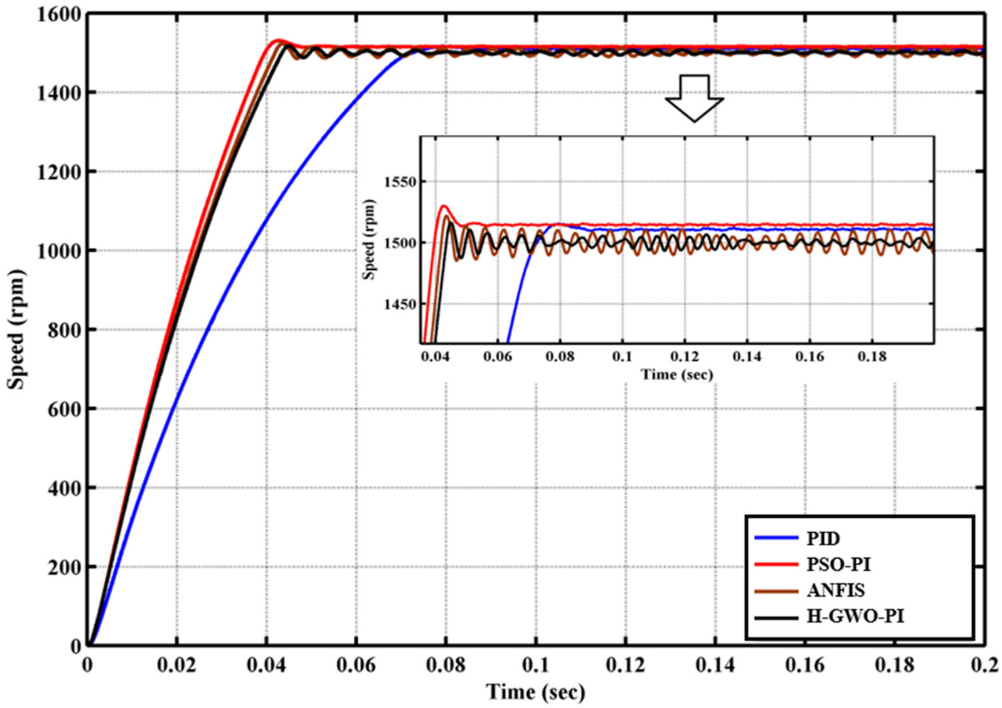

Figure 9. The speed response of a brushless DC motor for No-load.

Table 1. No-load condition's performance parameters.

Controller	Rise Time (s)	Overshoot (%)	Settling Time (s)	Steady State Error (rpm)	RMSE	IAE	ITAE	ISE
PID	0.0522	1.0227	0.0678	10.8886	0.2949	0.0256	0.00052	0.01618
PSO-PI	0.0299	1.9995	0.0383	14.4856	0.2432	0.0176	0.00029	0.01131
ANFIS	0.0323	1.0122	0.0210	5.13863	0.1932	0.0180	0.00026	0.01099
H-GWO-PI	0.0290	1.0086	0.0342	2.40466	0.1815	0.0174	0.00025	0.01031
Maximum margin for comparisons	0.0522	1.9995	0.0678	14.4856	0.2949	0.0256	0.00052	0.01618

In Table 1, maximum value of each column parameter is taken as the base value for comparisons. From this table, PID controller is worst in terms of rise time, Settling Time, ITAE RMSE, ISE and IAE. The PSO-PI controller has the worst overshoot and steady state error. ANFIS controller performs well than PID and PSO-PI controller. But the proposed controller is the only one that benefits from all performance factors i.e., Hybrid GWO-PI controller. With respect to the comparison outcomes, it is obvious that Hybrid GWO-PI controller is the best controller among other considered controllers. Figure 10 indicates the curve of speed response for full load situation (15 nm). Table 2 provides the controllers' performance indices and the corresponding time domain specifications.

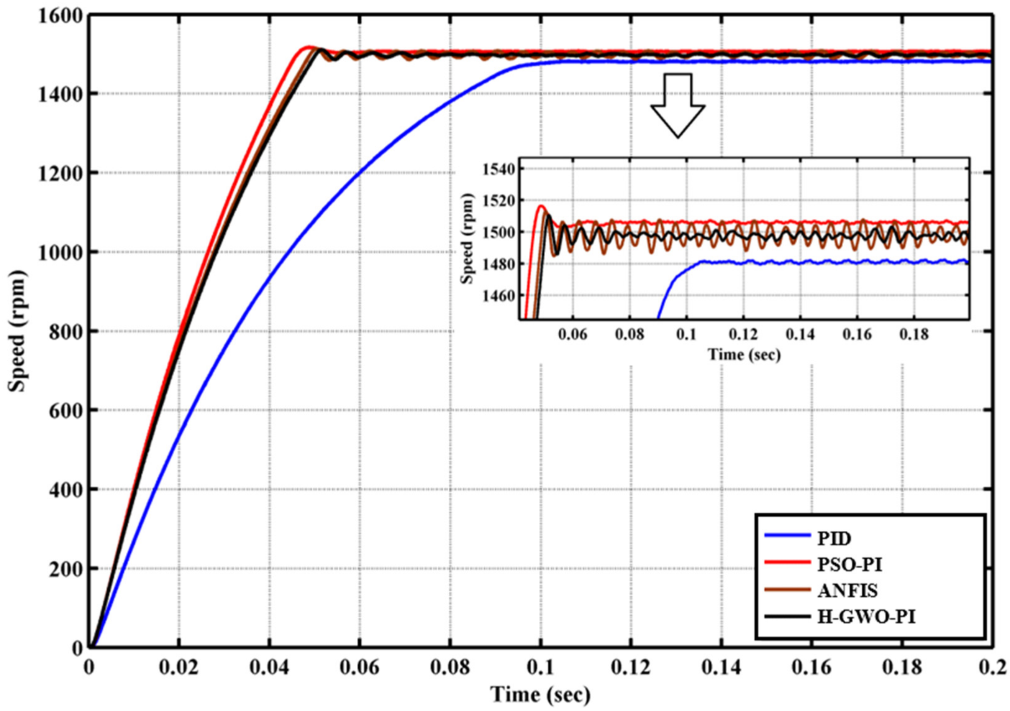

Figure 10. Speed response of the brushless dc motor under full load condition.

Table 2. Full load condition's performance parameters.

Controller	Rise Time (s)	Overshoot (%)	Undershoot (%)	Settling Time (s)	Steady State Error (rpm)	RMSE	IAE	ITAE	ISE
PID	0.0701	0.0000	0.0004	0.0959	18.8356	0.3249	0.0312	0.0008	0.0192
PSO-PI	0.0378	1.0850	0.0005	0.0475	6.1273	0.2571	0.0293	0.0003	0.0133
ANFIS	0.0342	0.8700	0.0002	0.0445	3.9891	0.1886	0.0193	0.0003	0.0126
H-GWO-PI	0.0344	0.7031	0.0002	0.0443	2.4532	0.1827	0.0192	0.0003	0.0124
Maximum Margin for Comparisons	0.0701	1.0850	0.0040	0.0959	18.8356	0.3249	0.0312	0.0008	0.0192

In Table 2, maximum value of each column parameter is taken as the base value for comparisons. From this table, PID controller is the poorest in regard to all parameters except overshoot. Regarding steady state error, undershoot and RMSE, the PSO-PI controller exhibits insignificant behavior. ANFIS controller performs well in terms all parameters than PID and PSO-PI controller. But Hybrid GWO-PI controller produces better performance in all vital parameters compared to other controllers considered.

B. Results for Varying Load Conditions

The drive is consistently exposed to differing load situations for various applications in the industry. In order to assess the advantage of the proposed controller, an abrupt alternation in load conditions is used to operate the closed loop system of the Brush-less DC motor. In this section, two examples' speed reactions under several load situations are discussed. For instance in case A, the speed is set at 1500 rpm and the load is altered starting from zero to maximum (15 nm) in 0.1 s. In case B, the load is reduced starting from full load situation (15 nm) to No load (zero nm) situation in 0.1 s while the speed of the BLDC motor is equal to 1500 rpm.

The speed response curves for case A are shown in Figure 11. When the load changes from zero to full, the motor speed should decrease. However, a motor controlled by a PID and PSO-PI controllers has a significant departure from its set speed. Motor speed fluctuates by up to 5 rpm around the programmed speed when using a PSO-PI controller.

However, a Hybrid GWO-PI controller causes the motor speed to fluctuate around 1.8 rpm. Overshoot is another crucial characteristic; it should be kept to a minimum as much as is practical. Any abrupt change in the load raises the case. For motors using PID and PSO-PI controllers, overshoot is particularly high. For a motor using a ANFIS controller, overshoot is mild. However, the motor with the suggested controller only overshot by 0.14%. Table 3 presents performance parameters including time domain requirements and performance indices. The maximum value for each column parameter in this table is used as the baseline for comparisons. The PID controller does not benefit from the performance indices shown in this table. Both the PSO-PI and ANFIS controllers have performance criteria that are moderated. But compared to other controllers under consideration, the proposed controller performs better.

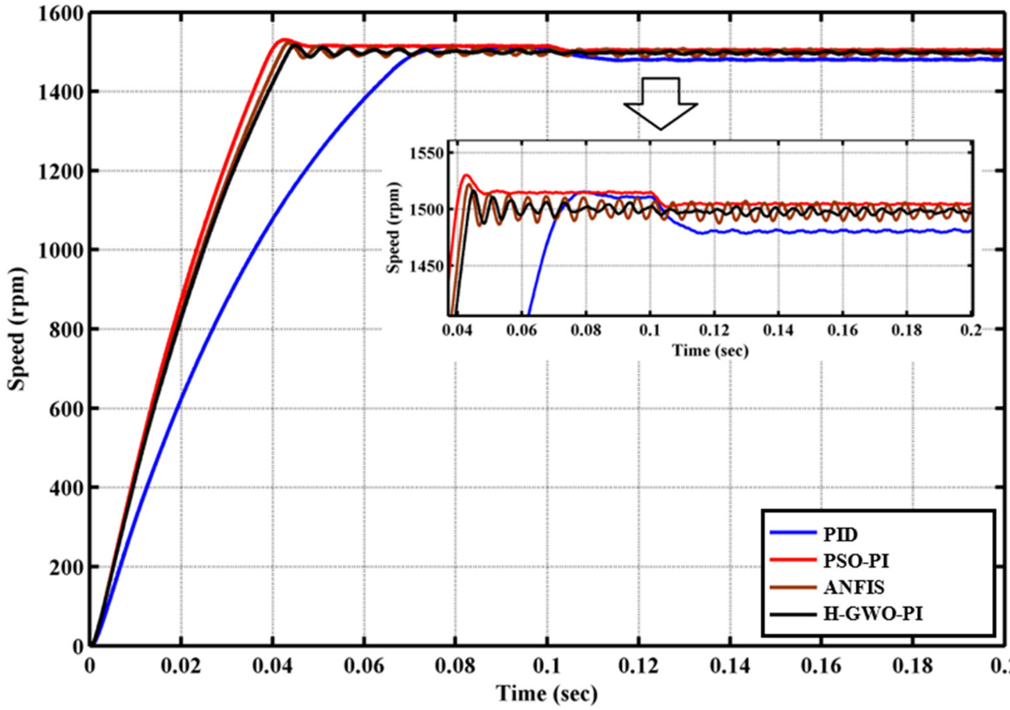

Figure 11. Response of the Brush-less DC motor in terms of speed under Case A.

Table 3. Performance parameters for Case A condition.

Controller	Overshoot (%)	Recovery Time (s)	Steady State Error (rpm)	RMSE	IAE	ITAE	ISE
PID	0.69834	0.20119	18.9933	0.30123	0.03209	0.00042	0.01701
PSO-PI	0.99244	0.15992	4.30241	0.30221	0.01802	0.00026	0.01191
ANFIS	0.47754	0.12970	4.79506	0.18270	0.01856	0.00024	0.01163
H-GWO-PI	0.14115	0.12690	1.86081	0.18249	0.01735	0.00024	0.01130
Maximum Margin for Comparisons	0.99553	0.16002	19.85259	0.29505	0.02577	0.00054	0.01619

A brush-less DC motor's speed response is shown in Figure 12 as the load changes from being fully loaded to being unloaded and Table 4 gives the related control system parameters. Speed will increase when a rapid load rejection occurs, and the speed error should be as small as possible. The steady state error of a motor with PID and PSO-PI controllers is greater. Moderate steady state error for motor using ANFIS controller. However, the Hybrid GWO-PI controller motor only has a stable inaccuracy of 1.12 rpm. Important parameters like speed overshoot, should be kept to a minimum. Compared to alternative controllers examined, the suggested has the least overshoot. The maximum value for each column parameter in Table 4 is used as the comparison standard. Based on this information, a motor using a PID controller has the poorest RMSE, ITAE, and ISE. The motor equipped with the PSO-PI controller performs the poorest in terms of steady state error, overshoot and RMSE. In comparison to the proposed controller and the PSO-PI controller, the motor with the ANFIS controller has a modest performance parameter. But compared to other controllers under consideration, the motor containing the suggested controller exhibits better performance parameters.

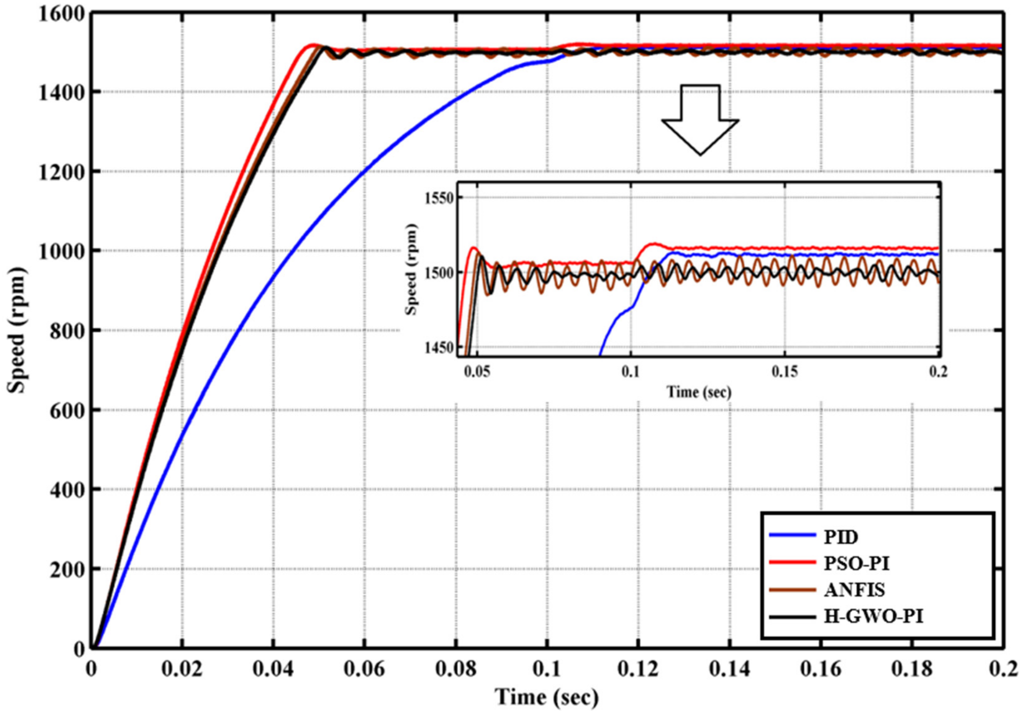

Figure 12. Response of the Brushless DC motor in terms of speed under Case B.

Table 4. Performance parameters for Case B condition.

Controller	Overshoot (%)	Recovery Time (s)	Steady State Error (rpm)	RMSE	IAE	ITAE	ISE
PID	0.79923	0.19536	11.99223	0.32481	0.02934	0.00081	0.01923
PSO-PI	1.30223	0.12012	16.0923	0.25716	0.02033	0.00029	0.01256
ANFIS	0.73044	0.15238	5.16432	0.18945	0.02088	0.00033	0.01243
H-GWO-PI	0.36210	0.11484	1.12620	0.18619	0.01928	0.00030	0.01233
Maximum Margin for Comparisons	1.26840	0.19536	15.81742	0.32481	0.03102	0.00079	0.01923

C. Results Under Various Set Speed Conditions

The drives set speed is altered in process industry in accordance with process demands. To confirm the effectiveness of the recommended controllers for those settings, two conditions for operating with varying set speeds are hypothesized and simulated. In cases C and D, the set speed is first altered from 1500 to 1000 rpm and later, from 1000 to 1500 rpm, respectively. The load is set to zero in both situations.

The speed response for the Case C scenario is represented in Figure 13. Table 5 displays the pertinent performance metrics. For comparisons, the maximum value for each column was used as the base value. According to this table, the PID controller has the poorest RMSE, IAE, ITAE, and ISE values. In terms of Undershoot, Recovery Time (s), and Steady state error (rpm), PSO-PI performs the poorest. Regarding steady state error, recovery time, ITAE, ISE, RMSE and IAE, Hybrid GWO-PI performs better than other controllers.

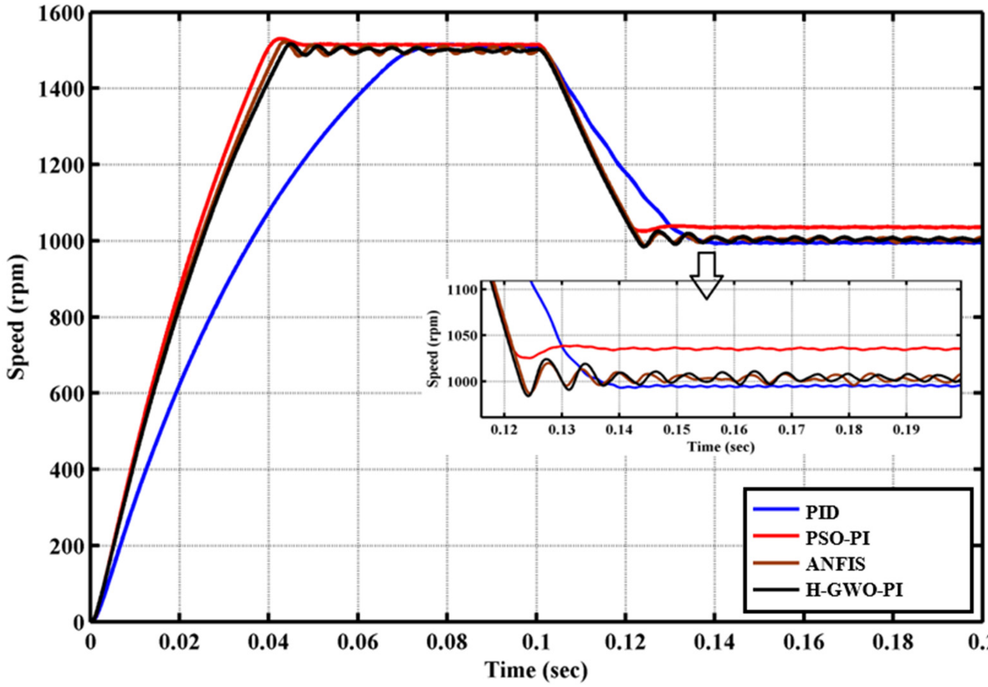

Figure 13. Response of the Brushless DC motor in terms of speed under Case C.

Table 5. Performance parameters for Case C condition.

Controller	Overshoot (%)	Undershoot (%)	Recovery Time (s)	Steady State Error (rpm)	RMSE	IAE	ITAE	ISE
PID	1.4009	2.1398	0.17003	7.893322	0.31764	0.02917	0.00091	0.01747
PSO-PI	1.3129	2.9884	0.21000	52.99004	0.26559	0.02070	0.00061	0.01236
ANFIS	2.6065	2.1398	0.21000	4.59215	0.20568	0.02164	0.00060	0.01304
H-GWO-PI	1.2040	0.4398	0.16397	4.40948	0.20420	0.02011	0.00055	0.01232
Maximum Margin for Comparisons	2.6065	2.9884	0.2100	53.3055	0.3176	0.0291	0.00091	0.01747

The speed response of the Brush-less DC motor in the Case D scenario is represented in Figure 14 and the corresponding performance metrics are shown in Table 6. For comparisons, the maximum value of each column parameter is used as the baseline. Except for the motor with the proposed controller, the system responds oscillatory to all other controllers. Additionally, compared to the other controllers, this one created a reduced steady state error. The steady-state error for the Hybrid GWO-PI controller is 0.74963 rpm. Only the proposed controller and the ANFIS controller benefit from an overshoot. PSO-PI controllers perform worse than other controllers on performance indicators. Compared with PID controllers, ANFIS and PSO-PI controllers perform best. However, compared to other controllers, the Hybrid GWO-PI controller provides better performance metrics. The Hybrid GWO-PI controller has demonstrated improved performance in all operating circumstances compared to other proposed controllers for Brush-less DC motor.

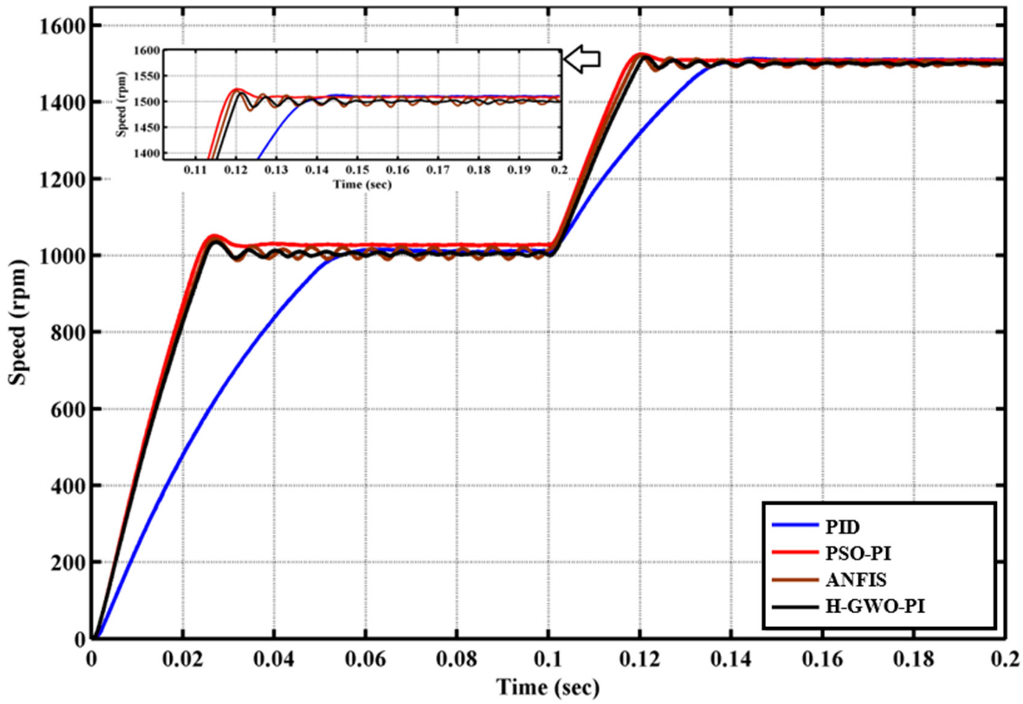

Figure 14. Response of the Brushless DC motor in terms of speed under Case D.

Table 6. Performance parameters for Case D condition.

Controller	Overshoot (%)	Recovery Time (s)	Steady State Error (rpm)	RMSE	IAE	ITAE	ISE
PID	0.90223	0.19902	9.9231	0.30128	0.0425	0.00059	0.02031
PSO-PI	1.70230	0.14024	7.99124	0.19920	0.02009	0.00033	0.00835
ANFIS	0.78184	0.21000	4.07081	0.17757	0.01423	0.00041	0.00816
H-GWO-PI	0.77103	0.13596	0.74963	0.17677	0.01384	0.00030	0.00804
Maximum Margin for Comparisons	1.61221	0.21000	10.00541	0.28130	0.02385	0.00062	0.01438

The Kp and Ki values for each PI controller is shown in Table 7.

Table 7. PI controller parameter for each Method.

	Kp1	Ki1	Kd1	Kp2	Ki2	Kd2
PID	0.4553	0.6221	0.22	0.0272	0.0228	0.21
PSO-PI	0.2401	1.9494	-	0.2609	0.9623	-
ANFIS PI	2.5184	2.0793	-	1.0073	1.1988	-
H-GWO-PI	1.4209	2.9329	-	0.0100	0.0100	-

As seen in Figure 8, there are two PI controller that tuned by the GWO, PSO and ANFIS. For the left side controller, the coefficients were Kp1 and Ki1. For the right-side controller, the coefficients were Kp2 and Ki2. The value of the coefficient for PI controller are selected between 0.01 to 3. These values are obtained after ten times running of each method. For conventional PID controller parameters Kp1, Ki1, Kd1, Kp2, Ki2, and Kd2 are tuned by ZN method and it is reported in Table 7.

4. Conclusions

To improve the performance of Brush-less DC motors, this research has given systematic methods using artificial intelligence techniques. MATLAB/Simulink software is utilized in order to design and execute the suggested controllers. The efficiency of the controllers has been examined and evaluated for different operation circumstances of Brush-less DC motor. Under all operational scenarios, the Hybrid GWO-PI controller outperforms other contenders regarding the enhanced time domain specifications and enhanced performance indices. The controller proposed, can resolve the indeterminacy problem brought on by load and set speed fluctuations. Outstanding behavior of the controller makes it the perfect choice for use in the processing industry. In the best scenario, the results have been obtained for case B, and it was 1.26840%, 0.19536, 15.81742, 0.32481, 0.03102, 0.00079, and 0.01923 for overshoot, recovery Time (s), steady state error (rpm), RMSE, IAE, ITAE, and ISE respectively.

Author Contributions: Methodology, S.M.Y.Y.; Software, J.R.; Validation, U.K.; Formal analysis, F.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Krishnan, R. *Switched Reluctance Motor Drives: Modeling, Simulation, Analysis, Design, and Applications*; CRC Press: Boca Raton, FL, USA, 2017; ISBN 1315220067.
2. Miller, T.J.E. *Brushless Permanent-Magnet and Reluctance Motor Drives*; Clarendon Press: Oxford, UK, 1989.
3. Sathyan, A.; Milivojevic, N.; Lee, Y.-J.; Krishnamurthy, M.; Emadi, A. An FPGA-Based Novel Digital PWM Control Scheme for BLDC Motor Drives. *IEEE Trans. Ind. Electron.* **2009**, *56*, 3040–3049. [CrossRef]
4. Bharatkar, S.S.; Yanamshetti, R.; Chatterjee, D.; Ganguli, A.K. Dual-Mode Switching Technique for Reduction of Commutation Torque Ripple of Brushless Dc Motor. *IET Electr. Power Appl.* **2011**, *5*, 193–202. [CrossRef]
5. Lai, Y.-S.; Lin, Y.-K. A Unified Approach to Zero-Crossing Point Detection of Back EMF for Brushless DC Motor Drives without Current and Hall Sensors. *IEEE Trans. Power Electron.* **2010**, *26*, 1704–1713. [CrossRef]
6. Gambetta, D.; Ahfock, A. New Sensorless Commutation Technique for Brushless DC Motors. *IET Electr. Power Appl.* **2009**, *3*, 40–49. [CrossRef]
7. Akkaya, R.; Kulaksız, A.A.; Aydoğdu, Ö. DSP Implementation of a PV System with GA-MLP-NN Based MPPT Controller Supplying BLDC Motor Drive. *Energy Convers. Manag.* **2007**, *48*, 210–218. [CrossRef]
8. Singh, B.; Singh, S. Single-Phase Power Factor Controller Topologies for Permanent Magnet Brushless DC Motor Drives. *IET Power Electron.* **2010**, *3*, 147–175. [CrossRef]
9. Gao, J.; Hu, Y. Direct Self-Control for BLDC Motor Drives Based on Three-Dimensional Coordinate System. *IEEE Trans. Ind. Electron.* **2009**, *57*, 2836–2844.
10. Karthikeyan, J.; Sekaran, R.D. Current Control of Brushless Dc Motor Based on a Common Dc Signal for Space Operated Vehicles. *Int. J. Electr. Power Energy Syst.* **2011**, *33*, 1721–1727. [CrossRef]
11. Joice, C.S.; Paranjothi, S.R.; Kumar, V.J.S. Digital Control Strategy for Four Quadrant Operation of Three Phase BLDC Motor with Load Variations. *IEEE Trans. Ind. Inform.* **2012**, *9*, 974–982. [CrossRef]
12. Demirtas, M. Off-Line Tuning of a PI Speed Controller for a Permanent Magnet Brushless DC Motor Using DSP. *Energy Convers. Manag.* **2011**, *52*, 264–273. [CrossRef]
13. Ibrahim, H.E.A.; Hassan, F.N.; Shomer, A.O. Optimal PID Control of a Brushless DC Motor Using PSO and BF Techniques. *Ain Shams Eng. J.* **2014**, *5*, 391–398. [CrossRef]
14. Rubaai, A.; Castro-Sitiriche, M.J.; Ofoli, A.R. DSP-Based Laboratory Implementation of Hybrid Fuzzy-PID Controller Using Genetic Optimization for High-Performance Motor Drives. *IEEE Trans. Ind. Appl.* **2008**, *44*, 1977–1986. [CrossRef]
15. Siong, T.C.; Ismail, B.; Siraj, S.F.; Mohammed, M.F. Fuzzy Logic Controller for BLDC Permanent Magnet Motor Drives. *Int. J. Electr. Comput. Sci.* **2011**, *11*, 13–18.
16. Shyam, A.; Febin Daya, J.L. A Comparative Study on the Speed Response of BLDC Motor Using Conventional PI Controller, Anti-Windup PI Controller and Fuzzy Controller. In Proceedings of the 2013 International Conference on Control Communication and Computing (ICCC), Thiruvananthapuram, India, 13–15 December 2013; pp. 68–73.
17. Al Gizi, A.J.H.; Mustafa, M.W.; Jebur, H.H. A Novel Design of High-Sensitive Fuzzy PID Controller. *Appl. Soft Comput.* **2014**, *24*, 794–805. [CrossRef]
18. Sahu, B.K.; Pati, S.; Mohanty, P.K.; Panda, S. Teaching–Learning Based Optimization Algorithm Based Fuzzy-PID Controller for Automatic Generation Control of Multi-Area Power System. *Appl. Soft Comput.* **2015**, *27*, 240–249. [CrossRef]
19. Wang, Y.; Xia, C.; Zhang, M.; Liu, D. Adaptive Speed Control for Brushless DC Motors Based on Genetic Algorithm and RBF Neural Network. In Proceedings of the 2007 IEEE International Conference on Control and Automation, Guangzhou, China, 30 May 2007–1 June 2007; pp. 1219–1222.

20. Cheng, Z.; Hou, C.; Wu, X. Global Sliding Mode Control for Brushless DC Motors by Neural Networks. In Proceedings of the 2009 International Conference on Artificial Intelligence and Computational Intelligence, Shanghai, China, 7–8 November 2009; Volume 4, pp. 3–6.
21. Sinanc, D.; Sahin, M.; Esen, Z.; Yavanoglu, U.; Sagiroglu, S. An Intelligent Feedback Control Mechanism for Brushless DC Motors. In Proceedings of the 2014 16th International Power Electronics and Motion Control Conference and Exposition, Antalya, Turkey, 21–24 September 2014; pp. 939–944.
22. Faieghi, M.R.; Azimi, S.M. Design an Optimized PID Controller for Brushless DC Motor by Using PSO and Based on NARMAX Identified Model with ANFIS. In Proceedings of the 2010 12th International Conference on Computer Modelling and Simulation, Cambridge, UK, 24–26 March 2010; pp. 16–21.
23. Varatharaju, V.M.; Mathur, B. Adaptive Controllers for Permanent Magnet Brushless DC Motor Drive System Using Adaptive-Network-Based Fuzzy Interference System. *Am. J. Appl. Sci.* **2011**, *8*, 810. [CrossRef]
24. Mosavi, M.-R.; Rahmati, A.; Khoshsaadat, A.; Elektrotechniczny, P. Design of Efficient Adaptive Neuro-Fuzzy Controller Based on Supervisory Learning Capable for Speed and Torque Control of BLDC Motor. *Przegląd Elektrotechniczny* **2012**, *88*, 238–246.
25. Deying, G.; Jinquan, Z. Speed Control of BLDCM Based on Compensated Fuzzy Neural Network. In Proceedings of the The 26th Chinese Control and Decision Conference (2014 CCDC), Changsha, China, 31 May–2 June 2014; pp. 4541–4544.
26. Premkumar, K.; Manikandan, B. V Adaptive Neuro-Fuzzy Inference System Based Speed Controller for Brushless DC Motor. *Neurocomputing* **2014**, *138*, 260–270. [CrossRef]
27. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
28. Hassan, H.A.; Zellagui, M. Application of Grey Wolf Optimizer Algorithm for Optimal Power Flow of Two-Terminal HVDC Transmission System. *Adv. Electr. Electron. Eng.* **2018**, *15*, 701–712. [CrossRef]
29. Shaikh, M.S.; Hua, C.; Jatoi, M.A.; Ansari, M.M.; Qader, A.A. Application of Grey Wolf Optimisation Algorithm in Parameter Calculation of Overhead Transmission Line System. *IET Sci. Meas. Technol.* **2021**, *15*, 218–231. [CrossRef]
30. Guha, D.; Roy, P.K.; Banerjee, S. Load Frequency Control of Large Scale Power System Using Quasi-Oppositional Grey Wolf Optimization Algorithm. *Eng. Sci. Technol. Int. J.* **2016**, *19*, 1693–1713. [CrossRef]