

Improving Neural Network Based on Seagull Optimization Algorithm for Controlling DC Motor

Widi Aribowo *, Supari Muslim, Fendi Achmad, Aditya Chandra Hermawan

*Department of Electrical Engineering
Faculty of Engineering
Universitas Negeri Surabaya
Unesa Kampus Ketintang
Surabaya, Indonesia*

Abstract

This article presents a direct current (DC) motor control approach using a hybrid Seagull Optimization Algorithm (SOA) and Neural Network (NN) method. SOA method is a nature-inspired algorithm. DC motor speed control is very important to maintain the stability of motor operation. The SOA method is an algorithm that duplicates the life of the seagull in nature. Neural network algorithms will be improved using the SOA method. The neural network used in this study is a feed-forward neural network (FFNN). This research will focus on controlling DC motor speed. The efficacy of the proposed method is compared with the Proportional Integral Derivative (PID) method, the Feed Forward Neural Network (FFNN), and the Cascade Forward Backpropagation Neural Network (CFBNN). From the results of the study, the proposed control method has good capabilities compared to standard neural methods, namely FFNN and CFBNN. Integral Time Absolute Error and Square Error (ITAE and ITSE) values from the proposed method are on average of 0.96% and 0.2% better than the FFNN and CFBNN methods.

Keywords: Seagull Optimization Algorithm, metaheuristic, DC motor, neural network.

I. INTRODUCTION

The technological development of the electric power system originating from renewable energy is growing rapidly. Renewable energy will produce a DC power source. This encourages the use of equipment with a higher DC power source. DC motor is equipment that converts electrical energy into mechanical energy [1]. DC motors use a dc power supply as input. Voltages from different polarities will produce mechanical energy. DC motors are very popular in use from household to industrial appliances. DC motors have advantages over AC motors, which are good speed control for acceleration and braking [1]. In addition, the DC motor can adjust in use so that it has a longer service life

Rotation speed is one of the important parameters that must be controlled by a DC motor. In DC motor speed regulation, PID control techniques are generally used with various methods. PID (Proportional Integral Derivative) is a control technique that is often used in control engineering. PID control consists of three types of controllers that are combined, namely Proportional, Integral, and Derivative. Parameters can determine the quality of the response of a control.

PID control is working with a feedback mechanism to correct errors between the error value of the measurement and the deviation value. In general, the PID control system can be used together or separately because each control has its advantages such as accelerating the rise time, minimizing steady-state errors, and reducing

overshoot or undershoot. Technological developments are beginning to shift towards an automation process using computers as the control center. Machine learning algorithms can drive significant advances in automatic control [2]. Conventional control methods have lacked the speed tracking requirement. This is influenced by sudden disturbances and variations in parameters [3].

Several studies have discussed DC motor control using artificial intelligence, such as a combination of Fuzzy Logic Controller (FLC) and Proportional-Integral (PI) controllers. DC motor is controlled with PI linear control theory into the fuzzy control structure [4]-[6]. The combination of FLC and PID controllers has also been described, the PID control method combined with the fuzzy method [7]- [9]. In addition, neural network-based control has also been developed. A popular neural network method is the feed-forward neural network [10]-[12].

Some researchers have combined intelligent control with some Nature-Inspired Algorithms. Nature-Inspired Algorithms have grown enormously in recent years. Several Nature-Inspired Algorithms have been applied to control DC motor, namely ant colony optimization (ACO) [13]-[16], grey wolf optimization (GWO) [17]-[20], Big Bang - Big Crunch [21], Grasshopper Optimization Algorithm [22], Atom Search Optimization Algorithm [23], Henry gas solubility optimization [24], and Flower Pollination Algorithm [25]-[28].

The purpose of this research is to control a DC motor based on a neural network which is enhanced using the Seagull Optimization Algorithm (SOA). To measure the performance of the proposed method, it will be compared with the PID method, feed-forward neural network

* Corresponding Author.

Email: widiaribowo@unesa.ac.id

Received: January 14, 2021 ; Revised: June 15, 2021

Accepted: July 05, 2021 ; Published: August 31, 2021

(FFNN), and cascade forward backpropagation neural network (CFBNN).

II. METHOD

A. A Seagull Optimization Algorithm

The Seagull Optimization Algorithm is a metaheuristic method that mimics the life of a seagull [29]. The seagull, scientifically named *Laridae*, is an omnivore. It feeds on reptiles, earthworms, insects, fish, and so on. Seagulls have intelligence and live in groups. The seagull has a special organ, which has a pair of glands that can remove salt from its body through a hole in the beak. This allows the seagull to drink from both salt and fresh water. Seagull uses his intelligence to find and attack his prey. Seagulls have migratory and hunting behavior. Migrating of the seagull is to find abundant food sources. This can be explained as follows

- Seagull migrates in groups by making formations. This is to avoid collisions between seagulls
- Seagulls follow the individuals who have the best survival in group travel.
- The seagull can update the starting position with the base of the strongest seagull.

The process of exploration and exploitation from a seagull can be mathematically modeled as follows,

1) Migration (exploration)

In the migration process, the seagulls will move from one place to another. In this process, several criteria must be met, namely:

a) Reduce crashes;

To reduce crashes between seagulls, variable A is used to calculate the position of the new search agent. Formation to reduce collisions can be seen in Figure 1. This process can be formulated as (1), (2), and (3).

$$\vec{C}_s = A \times \vec{P}_s(x) \quad (1)$$

$$A = f_c - \left(x \times \left(\frac{f_c}{Max_{iteration}} \right) \right) \quad (2)$$

$$x = 0, 1, 2, \dots, Max_{iteration} \quad (3)$$

Where the position for avoiding crash search agent with another search agent is \vec{C}_s . The recent position from search agent is \vec{P}_s . The signifier of the current iteration is x . The movement behavior of the search agent in a given search space is A . f_c is to tune the frequency of attaching variable A which is linearly lower to 0. Normally, f_c is tuned to 2.

b) Following the best seagull direction

After completing the first phase aimed at collision avoidance, search agents will follow the best individuals. This phase can be formulated as (4) and (5).

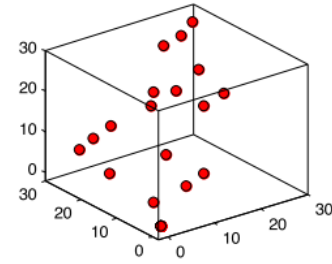


Figure 1. Reduce crashes between search agents [29].

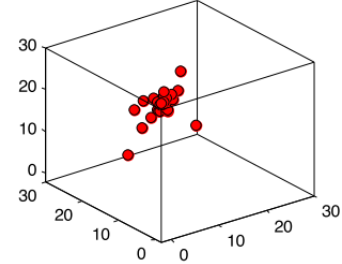


Figure 2. The Phase of Following the best seagull direction [29].

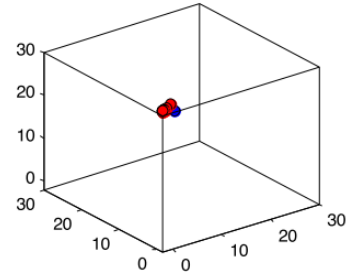


Figure 3. Convergence follows the best seagull [29].

$$\vec{M}_s = B \times (\vec{P}_{bs}(x) - \vec{P}_s(x)) \quad (4)$$

$$B = 2 \times A^2 \times rd \quad (5)$$

Where the position of seagull \vec{P}_s follows the best seagull \vec{P}_{bs} is \vec{M}_s . The parameter that regulates the balance between exploration and exploitation is B . rd is a random value with range [0, 1]. Figure 2 is an illustration of the phase following the best seagull direction.

c) Stay tight to the best seagull

Finally, the seagull can reform the position following to best seagull. it can be shown in Figure 3 and formulated as (6).

$$\vec{D}_s = |\vec{C}_s + \vec{M}_s| \quad (6)$$

Where the range between the seagull and best seagull is \vec{D}_s .

2) Attacking (exploitation)

In the exploitation process, seagulls can vary the attack and speed during migration. Seagulls use their weight and wings to adjust their height. Seagull behavior has a spiral motion when attacking its prey.

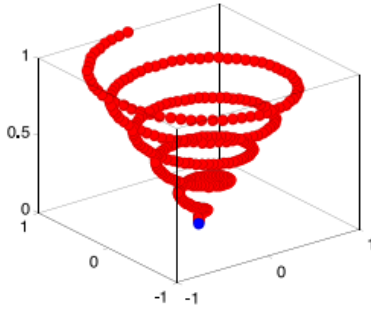


Figure 4. The basic assaulting attitude of seagull [29].

It can be described as shown in Figure 4. This can be modeled in mathematics as (7), (8), (9), and (10).

$$x' = r \times \cos(k) \quad (7)$$

$$y' = r \times \sin(k) \quad (8)$$

$$z' = r \times k \quad (9)$$

$$r = u \times e^{kv} \quad (10)$$

The radius of each spin of the spiral is r , the variable in the range $[0 \leq k \leq 2\pi]$ is k . The parameters for forming the spiral are u and v . e is the primary of the logarithm. The updated seagull position can be modeled as (11).

$$\vec{P}_s(x) = (\vec{D}_s \times x' \times y' \times z') + \vec{P}_{bs}(x) \quad (11)$$

Where $\vec{P}_s(x)$ holds the best solution and updates the position of other seagulls.

B. Feed-Forward Neural Network (FFNN)

An artificial neural network (ANN) is a mathematical model that duplicates the structure and functional aspects of biological networks [30]. ANN consists of groups of interrelated artificial neurons and processes data using computational methods. The main items in FFNN are neurons managed with the inputs, outputs, and hidden layers. Input layers translate data into the network. The signal is passed to a weighted connection on the hidden layer. At this layer, each neuron receives weighted data and included bias. Next, The data flows to the output layer. FFNN can be formulated in (12)-(15).

$$U_1(t) = \sum_{i=1}^j W_{ij} A_n(t) + b_1 \quad (12)$$

$$U_2(t) = f(U_1(t)) = \frac{1}{1+\exp^{-U_1}} \quad (13)$$

$$U_3(t) = \sum_{j=1}^k W_{jk} U_2(t) + b_2 \quad (14)$$

$$U_4(t) = f(U_3(t)) = \frac{1}{1+\exp^{-U_3}} \quad (15)$$

C. DC Motor

DC motors have the function to convert electrical energy into mechanical energy. The torque produced by the DC motor is using a DC power supply. DC motors are classified as external and self-exciting types.

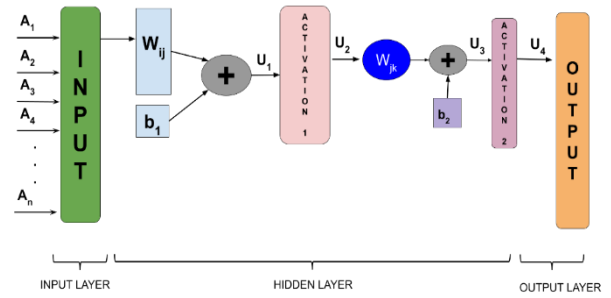


Figure 5. The FFNN Structure [32].

TABLE 1.
DC MOTOR PARAMETERS

Parameter	Value
Back emf constant (K_b)	0.01 N-m/Amp
Armature resistance (R)	2 Ω
Armature inductance (L)	0.2 H
Mechanical inertia (J)	0.02 Kg/m ²
Friction coefficient (B)	0.5 Nm/rmp

The basic circuit of motor the DC is illustrated in Figure 6. The detail of DC Motor can be seen in Table 1. Where I_f is Field current (A). Armature voltage (V_a) has a function to control the speed of a dc motor [31]. The mathematical equation is as in (16).

$$V_a(s) = (R_a + L_a \cdot s) \cdot I_a(s) + E_b(s) \quad (16)$$

The induced voltage (E_b) is proportional to the angular velocity (ω) for constant flux as in (17).

$$E_b(s) = K_b \omega(s) \quad (17)$$

The torque generated by the armature current is the sum of the inertia and friction torque. The mathematical equation is (8).

$$T_m(s) = J s \omega(s) + B \omega(s) \quad (18)$$

D. The Proposed SOA-NN Model

SOA methods have a good global search [29]. So, it is used to find the best weight using a neural network. Neural network methods identify and map the incoming signal. Next, the neural network will be configured to get a random weight. This will be processed using the SOA method until the optimal weight. Details of the SOA-NN method can be seen in Figure 7. The Pseudo-code of the proposed SOA is described in Table 2.

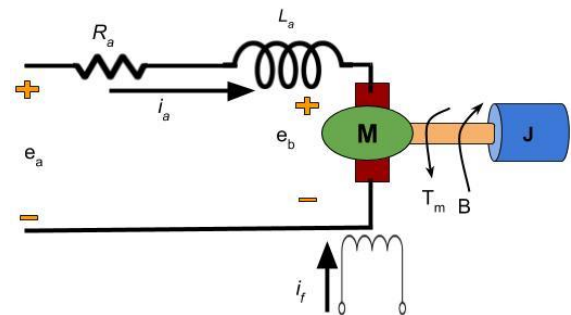


Figure 6. DC Motor Equivalent Circuit [31].

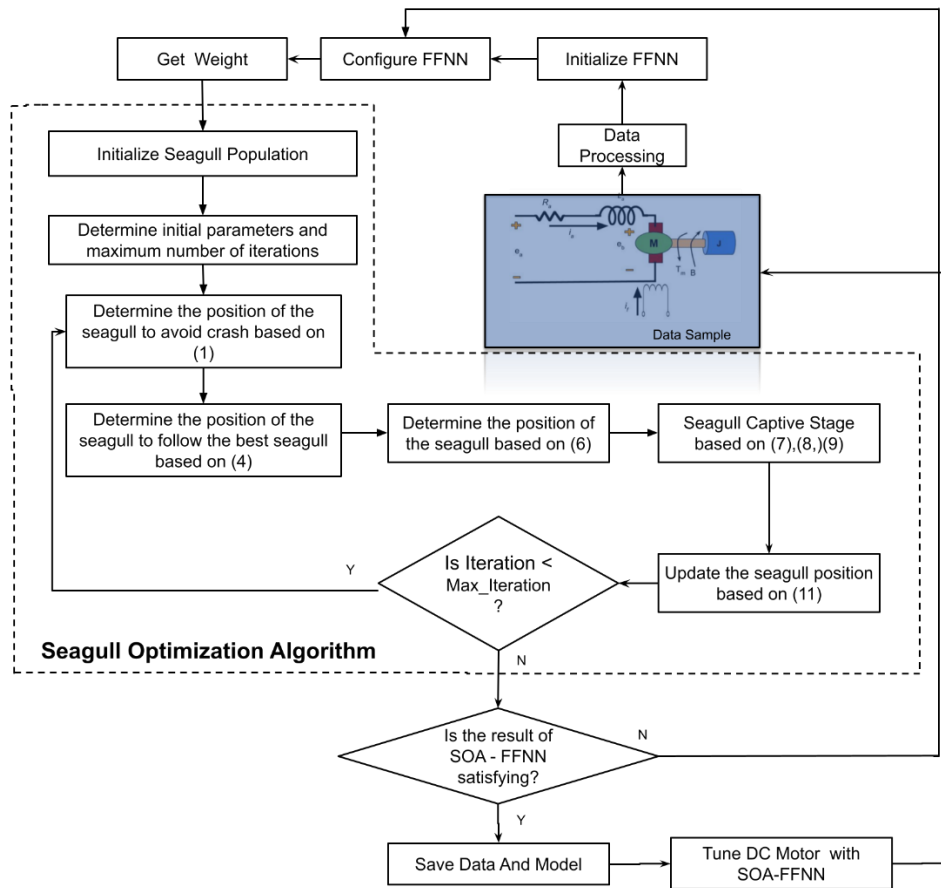


Figure 7. The SOA-NN Flowchart.

III. RESULTS AND DISCUSSIONS

In this section, the experimental data is simulated with MATLAB 2015. The performance validation of the proposed SOA-NN method is compared with three other methods, namely PID, FFNN and CFBNN. The diagram of DC motor control using SOA-NN can be seen in Figure 8. For the FFNN, and CFBNN methods, this is using 4 hidden layers, Levenberg-Marquardt, and 1000 iterations. Benchmark function has two categories, namely unimodal and multimodal. In this research, upper bound and lower Bound are used and the most optimal is using function 7 of unimodal benchmark functions.

Meanwhile, the proposed SOA-NN method duplicates training such as the FFNN method by adding the SOA algorithm method to improve neural network performance. Details of the SOA-NN method can be seen in Table 3. The convergence curve of the SOA-NN method can be seen in Figure 9. The curve approaches the value of 0 at 50 seconds. The installation and testing of the SOA-NN method are applied to determine the training data and targets with the proposed method. An illustration of a closed-loop system with various controllers to a DC motor can be seen in Figure 10. Table 4 is a detailed output of the various controllers at reference speed 1.

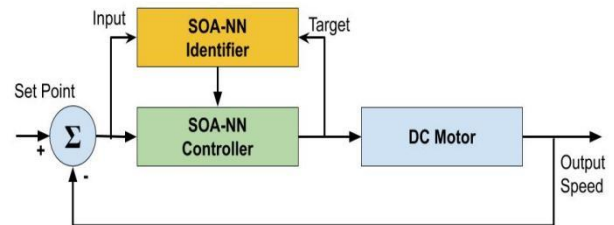


Figure 8. SOA-NN Controller For DC Motor.

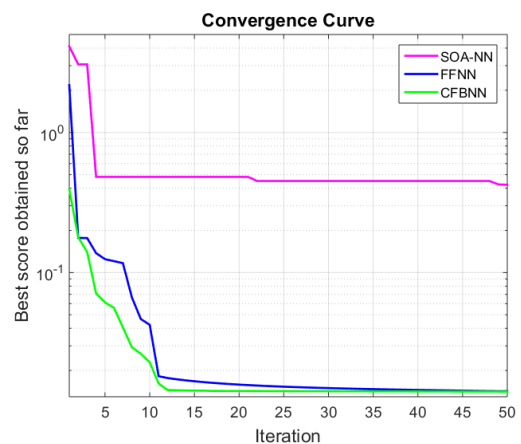


Figure 9. The convergence curve of SOA.

TABLE 2.
PSEUDO CODE OF SEAGULL OPTIMIZATION ALGORITHM

Algorithm 1 Pseudo Code of Seagull Optimization Algorithm	
Input: Seagull population \vec{P}_s	
Output: Optimal search agent \vec{P}_{bs}	
1: procedure SOA	
2: Initialize the parameters A , B , and $Max_{iteration}$	
3: Set $f_c \leftarrow 1$	
4: Set $u \leftarrow 1$	
5: Set $v \leftarrow 1$	
6: while ($x < Max_{iteration}$) do	
7: $\vec{P}_{bs} \leftarrow \text{ComputeFitness}(\vec{P}_{bs})$	
/* Migration behavior */	
8: $rd \leftarrow \text{Rand}(0,1)$	
9: $k \leftarrow \text{Rand}(0, 2\pi)$	
/* Attacking behavior */	
10: $r \leftarrow r = u \times e^{kv}$ using (10)	
11: Calculate the distance \vec{D}_s using (6)	
12: $P \leftarrow x' \times y' \times z'$ using (7)–(9)	
13: $\vec{P}_s(x) \leftarrow (\vec{D}_s \times p) + \vec{P}_{bs}(x)$	
14: $x \leftarrow x + 1$	
15: end while	
16: return \vec{P}_{bs}	
17: end procedure	
1: procedure ComputeFitness(\vec{P}_s)	
2: for $x \leftarrow 1$ to n do	
3: $\text{FIT}_s[i] \leftarrow \text{FitnessFunction}(\vec{P}_s(i,:))$	
4: end for	
5: $\text{FIT}_{sbest}[i] \leftarrow \text{BEST}(\text{FIT}_s[i])$	
6: return FIT_{sbest}	
7: end procedure	
1: procedure BEST($\text{FIT}_s[i]$)	
2: Best $\leftarrow \text{FIT}_s[0]$	
3: for $i \leftarrow 1$ to n do	
4: if ($\text{FIT}_s[i] < \text{Best}$) then	
5: Best $\leftarrow \text{FIT}_s[i]$	
6: end if	
7: end for	
8: return Best	
9: end procedure	

Total weighted absolute value error (ITAE) and Total time-weighted square of error (ITSE) are applied to evaluate SOA-NN performance. The ITAE and ITSE fitness functions are as follows.

$$ITAE = \int_0^\infty t \cdot e(t) \cdot dt \quad (19)$$

$$ITSE = \int_0^\infty t \cdot e^2(t) \cdot dt \quad (20)$$

The comparison of the ITAE and ITSE with the four controllers can be seen in Table 5. The ITAE value of the PID has a value of 0.0428. Meanwhile, the lowest ITAE value is owned by the SOA-NN method, which is 0.0352. The ITSE value for the SOA-NN method is 0.0227. Meanwhile, the highest value of ITSE is owned by the PID method, namely 0.0249.

TABLE 3.
PARAMETER OF SOA-NN

Parameter	Value
Hidden Layer	4
Training	Levenberg-Marquardt
Maximum Iteration Number	50
Number of Seagull	50
Lower Bound; Upper Bound	-1.28;1.28

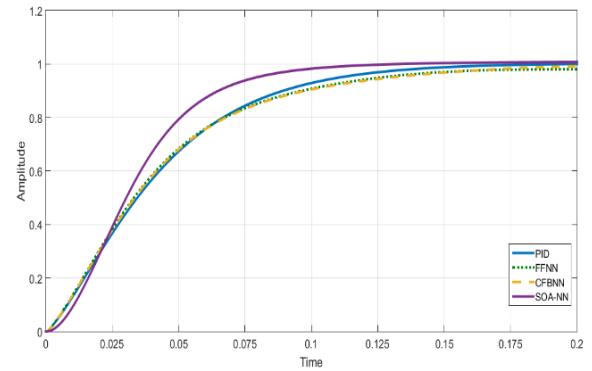


Figure 10. Speed Output Of The DC Motor at reference of speed 1.

TABLE 4. TIME DOMAIN PERFORMANCE COMPARISON FOR EACH CONTROLLER AT REFERENCE SPEED OF 1.

Controller	Overshoot	Settling Time (s)	Rise Time (s)
PID	1.003	0.164	0.0010
FFNN	No Overshoot	0.19	0.0005
CFBNN	No Overshoot	0.198	0.0025
SOA-NN	1.008	0.1397	0.0003

TABLE 5. COMPARISON OF THE FITNESS FUNCTION OF THE EACH CONTROLLER AT REFERENCE SPEED OF 1

Controller	ITAE	ITSE
PID	0.0428	0.0241
FFNN	0.0446	0.0244
CFBNN	0.0448	0.0247
SOA-NN	0.0352	0.0227

Table 6 is a detailed output of the various controllers at reference speed 0.6. The comparison of the ITAE and ITSE with the four controllers at reference speed 0.6 can be seen in Table 7. The ITAE value of the PID has a value of 0.02528. This value is the lowest value at the reference speed of 0.6.

TABLE 6.
TIME DOMAIN PERFORMANCE COMPARISON FOR EACH CONTROLLER AT REFERENCE SPEED OF 0.6

Controller	Overshoot	Settling Time (s)	Rise Time (s)
PID	0.6002	0.175	0.0018
FFNN	0.6089	0.180	0.0015
CFBNN	0.6076	0.179	0.0015
SOA-NN	0.6057	0.177	0.0015

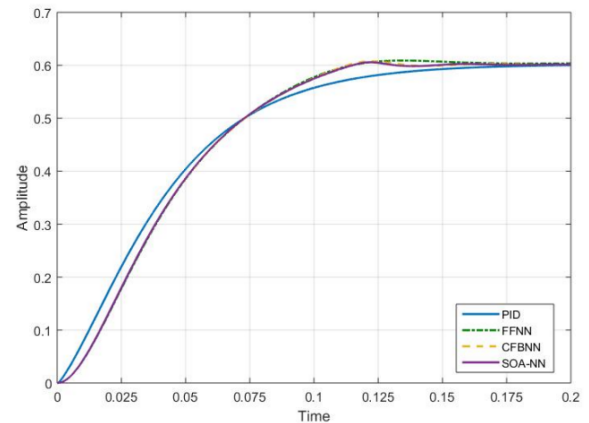


Figure 11. Speed Output Of The DC Motor at reference of speed 0.6.

TABLE 7. COMPARISON OF THE FITNESS FUNCTION OF THE EACH CONTROLLER AT A REFERENCE SPEED OF 0.6

Controller	ITAE	ITSE
PID	0.02528	0.04451
FFNN	0.026199	0.04518
CFBNN	0.025987	0.045
SOA-NN	0.025938	0.04496

The ITAE value for the SOA-NN method is 0.025938 and the ITSE value of SOA-NN is 0.04496. Meanwhile, the highest value of ITSE is owned by the FFNN method, which is 0.04518. The comparison graph with reference speed 0.6 can be seen in Figure 11.

CONCLUSION

This study aims to control a DC motor using the Seagull Optimization Algorithm and Neural Network (SOA-NN) method. DC motor control is very popular research. From the research, it can be concluded that the proposed method has a good performance. By using two-speed references, which are 1 and 0.6, the performance of SOA-NN is better than the FFNN and CFBNN methods. The ITAE and ITSE values for the SOA-NN method with a reference speed of 1 are 0.0352 and 0.0227. On the other hand, with a reference speed of 1, the worst ITAE and ITSE values are owned by the CFBNN method. ITAE and ITSE of the CFBNN method are 0.448 and 0.247. With a reference speed of 0.6, the ITAE and ITSE values of the SOA-NN method are 0.025938 and 0.04496. this value is better than the FFNN and CFBNN methods. The proposed method, namely SOA-NN, still needs to be improved by adding new methods. Testing is also required with more complex case studies. Several neural network models that have been found need to be tested to determine the performance of the proposed method.

REFERENCES

- [1] A. Lotfy, M. Kaveh, M.R. Mosavi, and A.R. Rahmati, "An enhanced fuzzy controller based on improved genetic algorithm for speed control of DC motors," *Analog Integrated Circuits and Signal Processing*, vol. 105, pp. 141-155, 2020.
- [2] M. Aamir, "On Replacing PID Controller with ANN Controller for DC Motor Position Control," *International Journal of Research Studies in Computing*, vol 2(1), pp. 21-29, 2013.
- [3] A.A. El-Samahy, and M.A. Shamseldin, "Brushless DC motor tracking control using self-tuning fuzzy PID control and model reference adaptive control," *Ain Shams Engineering Journal*, Vol. 9, Issue 3, pp 341-352, 2018.
- [4] N. N. Baharudin and S. M. Ayob, "Brushless DC motor drive control using Single Input Fuzzy PI Controller (SIFPIC)," *2015 IEEE Conference on Energy Conversion (CENCON)*, Johor Bahru, 2015, pp. 13-18.
- [5] B. N. Kommula and V. R. Kota, "Performance evaluation of Hybrid Fuzzy PI speed controller for Brushless DC motor for Electric vehicle application," *2015 Conference on Power, Control, Communication and Computational Technologies for Sustainable Growth (PCCCTSG)*, Kurnool, 2015, pp. 266-270.
- [6] N. Tiwary, A. Rathinam and S. Ajitha, "Design of Hybrid Fuzzy-PI controller for speed control of Brushless DC motor," *International Conference on Electronics, Communication and Instrumentation (ICECI)*, Kolkata, 2014, pp. 1-4.
- [7] E. Flores-Morán, W. Yáñez-Pazmiño and J. Barzola-Monteses, "Genetic algorithm and fuzzy self-tuning PID for DC motor position controllers," *2018 19th International Carpathian Control Conference (ICCC)*, Szilvasvarad, 2018, pp. 162-168.
- [8] A. Y. Al-Maliki and K. Iqbal, "FLC-based PID controller tuning for sensorless speed control of DC motor," *2018 IEEE International Conference on Industrial Technology (ICIT)*, Lyon, 2018, pp. 169-174.
- [9] R. Tudoroiu, M. Zaheeruddin, N. Tudoroiu and D. D. Burdescu, "Fuzzy Logic PID Control of a PMDCM Speed Connected to a 10-kW DC PV Array Microgrid—Case Study," *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, Leipzig, Germany, 2019, pp. 359-362.
- [10] B. M. Zaineb, A. Aicha, B. H. Mouna and S. Lassaad, "Speed control of DC motor based on an adaptive feed forward neural IMC controller," *2017 International Conference on Green Energy Conversion Systems (GECS)*, Hammamet, 2017, pp. 1-7.
- [11] R. Jacob and S. Murugan, "Implementation of neural network based PID controller," *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, Chennai, 2016, pp. 2769-2771.
- [12] N. Pimkumwong and M. Wang, "An Online Artificial Neural Network Speed Estimator for Sensorless Speed Control of Separately Excited DC Motor," *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Chiang Rai, Thailand, 2018, pp. 37-40.
- [13] R. Singh, A. Kumar and R. Sharma, "Fractional Order PID Control using Ant Colony Optimization," *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, Delhi, 2016, pp. 1-6.
- [14] D. Sandoval, I. Soto and P. Adasme, "Control of direct current motor using Ant Colony optimization," *2015 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, Santiago, 2015, pp. 79-82.
- [15] E. C. Şimşek, A. Köse, M. Şahin and E. Irmak, "Optimization of PID Parameters Using Ant Colony Algorithm for Position Control of DC Motor," *2019 8th International Conference on Renewable Energy Research and Applications (ICRERA)*, Brasov, Romania, 2019, pp. 1047-1051.
- [16] B. A. Kouassi, Y. Zhang, S. Ouattara and M. J. Mbyamm Kiki, "PID Tuning of Chopper Fed Speed Control of DC Motor Based on Ant Colony Optimization Algorithm," *2019 IEEE 3rd International Electrical and Energy Conference (CIEEC)*, Beijing, China, 2019, pp. 407-412.
- [17] K. R. Das, D. Das and J. Das, "Optimal tuning of PID controller using GWO algorithm for speed control in DC motor," *2015 International Conference on Soft Computing Techniques and Implementations (ICSCTI)*, Faridabad, 2015, pp. 108-112.
- [18] B. M. Kumar and R. B. Ashok, "Soft Computing Using GWO (Grey Wolf Optimization) for the Performance Improvement of High-Speed Brushless DC Motor," *2018 International Conference on Emerging Trends and Innovations in Engineering and Technological Research (ICETIETR)*, Ernakulam, 2018, pp. 1-6.
- [19] A. Ahmed, R. Gupta and G. Parmar, "GWO/PID Approach for Optimal Control of DC Motor," *2018 5th International Conference on Signal Processing and Integrated Networks (SPIN)*, Noida, 2018, pp. 181-186.
- [20] M. Muniraj, R. Arulmozhiyal, D. Kesavan, "An Improved Self-Tuning Control Mechanism for BLDC Motor Using Grey Wolf Optimization Algorithm," in *Lect. Notes Electr. Eng.*, V. Bindhu, J. Chen, J. Tavares, eds., Singapore: Springer Singapore, 2020, pp. 315-323.
- [21] H. K. Verma and C. Jain, "Big-bang big-crunch based optimization of PID controller for DC motor," *2015 International Conference on Computer, Communication and Control (IC4)*, Indore, 2015, pp. 1-7.
- [22] D. Potnuru and A.S. Tummala, "Implementation of Grasshopper Optimization Algorithm for Controlling a BLDC Motor Drive," *Soft Computing in Data Analytics. Advances in Intelligent Systems and Computing*, vol. 758, 2019.
- [23] B. Hekimoğlu, "Optimal Tuning of Fractional Order PID Controller for DC Motor Speed Control via Chaotic Atom Search Optimization Algorithm," in *IEEE Access*, vol. 7, pp. 38100-38114, 2019.
- [24] Serdar Ekcinci, Baran Hekimoğlu and Davut Izci, "Opposition based Henry gas solubility optimization as a novel algorithm for PID control of DC motor," *Engineering Science and Technology, an International Journal*, 2020.
- [25] P. Khlubwannarat, A. Nawikavatan and D. Puangdownreong, "Application of Parallel Flower Pollination Algorithm to Fractional-Order Model Identification of BLDC Motor," *2020*

- 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Phuket, Thailand, 2020, pp. 131-134.
- [26] D. Potnuru, K.A. Mary and C.S. Babu, "Experimental implementation of Flower Pollination Algorithm for speed controller of a BLDC motor," *Ain Shams Engineering Journal*, vol. 10, no. 2, pp. 287-295, 2019.
- [27] T. Tarczewski, L. J. Niewiara and L. M. Grzesiak, "An Application of Flower Pollination Algorithm to Auto-Tuning of Linear-Quadratic Regulator for DC-DC Power Converter," *2018 20th European Conference on Power Electronics and Applications (EPE'18 ECCE Europe)*, Riga, 2018, pp. 1-8.
- [28] P. S. R. Nayak and T. A. Rufzal, "Flower Pollination Algorithm Based PI Controller Design for Induction Motor Scheme of Soft-Starting," *2018 20th National Power Systems Conference (NPSC)*, Tiruchirappalli, India, 2018, pp. 1-6.
- [29] G. Dhiman and V. Kumar, "Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems," *Knowl-Based Syst*, vol.165, pp. 169–196, 2019.
- [30] W. Aribowo, S. Muslim, munoto, B. Suprianto, U. T. Kartini and I. G. P. Asto Buditjahjanto, "Tuning of Power System Stabilizer Using Cascade Forward Backpropagation," *2020 Third International Conference on Vocational Education and Electrical Engineering (ICVEE)*, Surabaya, Indonesia, 2020, pp. 1-5.
- [31] B.A. Obaid, A.L. Saleh and A.K. Kadhim, "Resolving of optimal fractional PID controller for DC motor drive based on anti-windup by invasive weed optimization technique," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 15, no. 1, pp. 95-103. 2019.
- [32] W. Aribowo, B. Suprianto, and Joko, "Improving neural network using a sine tree-seed algorithm for tuning motor DC," *International Journal of Power Electronics and Drive System (IJPEDS)*, vol. 12, no. 2, pp. 1196-1204, 2021, doi: <http://doi.org/10.11591/ijpeds.v12.i2.pp1196-1204>.