1. Muldi YUHENDRI[1], 2. Risfendra[1], 3. Emilham MIRSHAD[1], 4. Adam R. SIDIQI[1]

Universitas Negeri Padang (1)
ORCID: 1. 0000-0002-2123-2981; 2. 0000-0002-8614-3947; 3. 0000-0002-1206-9465; 4. 0000-0002-8061-6985

doi:10.15199/48.2024.10.22

Real-time Control of Separately Excited DC Motor Based on Fuzzy PI System Using Arduino

***Abstract**. This paper presents the design and implementation of a speed control system for a separately excited Direct Current (DC) motor using a four-quadrant DC chopper. DC motor speed control is designed using the Fuzzy PI method. The proposed system is implemented using an Arduino Mega 2560. The reliability of the speed control system is validated through experiments with varying both rotor speeds and loads. The experimental results show that the proposed Fuzzy PI controller has provided results in accordance with the objectives. All experiments carried out show that the Fuzzy PI controller has successfully controlled the motor speed according to the reference speed.*

Streszczenie.** W artykule przedstawiono projekt i realizację układu sterowania prędkością silnika prądu stałego (DC) z oddzielnym wzbudzeniem, wykorzystującego czterokwadrantowy przerywacz prądu stałego. Sterowanie prędkością silnika prądu stałego zostało zaprojektowane przy użyciu metody Fuzzy PI. Proponowany system jest realizowany przy użyciu Arduino Mega 2560. Niezawodność systemu kontroli prędkości jest sprawdzana poprzez eksperymenty ze zmieniającymi się zarówno prędkościami, jak i obciążeniami wirnika. Wyniki eksperymentów pokazują, że proponowany sterownik Fuzzy PI zapewnił wyniki zgodne z założonymi celami. Wszystkie przeprowadzone eksperymenty pokazują, że sterownik Fuzzy PI skutecznie steruje prędkością silnika zgodnie z prędkością odniesienia. (Sterowanie w czasie rzeczywistym silnikiem prądu stałego o oddzielnym wzbudzeniu w oparciu o układ Fuzzy PI z wykorzystaniem Arduino**)

Keywords: Fuzzy PI controller, separately excited DC motor, four-quadran DC chopper, Arduino.
Słowa kluczowe: Kontroler Fuzzy PI, silnik prądu stałego wzbudzony oddzielnie, czterokwadrantowy przerywacz prądu stałego, Arduino.

Introduction

Production equipment in industry that works in rotation is generally driven by an electric motor. The electric motor as the driving force must be able to work according to the needs of the production process. Therefore, the electric motor must be controlled. Several control techniques on electric motors are applied to obtain machine operations that suit the needs of the production process, such as regulating speed, direction of rotation, braking and starting current on large power motors [1].

DC motors are a type of electric motor that is widely used in industry, because they are easier to control than AC motors [2]. Based on the excitation voltage source, DC motors are categorized into self excited and separately excited [3]. In this paper, speed control for a separate excited DC motor is proposed. This motor is easy to control, where speed regulation through flux regulation in the field coil and torque regulation in the armature can be done separately. DC motor speed control is usually carried out using a power converter. The power converter can control speed, direction of rotation, braking and soft starting in one unified control system [4]. The type of power converters are determined by the voltage source operating range of the machine. DC machine operation can be divided into four quadrants, namely forward motoring, reverse motoring, forward generating and reverse generating [5]. The types of converters commonly used are controlled rectifiers for AC sources and DC-DC converters or DC choppers for DC sources. DC motor control is proposed using a four quadrant DC chopper, which allows the motor to operate on all quadrants of the DC machine.

DC motor speed control can be done by regulating torque via armature current or regulating flux via field current. [6]. In this paper, DC motor speed control is proposed by regulating torque through armature current, while the motor flux is maintained constant. Several control methods have been applied to regulate this torque, such as the PI controller [7], Sliding mode control [8], linear quadratic regulator [9], robust control [10] and several artificial intelligence methods, including fuzzy logic controllers [11], artificial neural networks [12] and various other algorithms. Each method has advantages and disadvantages. The PI controller method has the advantage of being easy to implement, but has the disadvantage of not being able to withstand parameter changes due to external disturbances. This is because both the proportional and integral gains are constant. To overcome this problem, various gain values have been developed by combining the PI controller with other methods, such as the Fuzzy PI controller [13], neural network PI controller [14] and so on. This makes the controller more adaptive and robust to parameter uncertainty. Therefore, in this paper a DC motor speed control based on a fuzzy PI controller is proposed. The DC motor speed control system is implemented with a four quadrant DC chopper, making it compatible for all DC motor operating quadrants. The Fuzzy PI controller system is used to control motor speed by regulating torque through armature current control. The control system is implemented with Arduino Mega 2560. The correctness of the proposed control system is verified through experiments by varying both motor speed and load.

Separately Excited DC Motor

The Separately Excited DC Motor (SEDCM) is a type of DC motor that uses two separate voltage sources, namely for the field coil and for the armature coil. Figure 1 shows the equivalent circuit of SEDCM

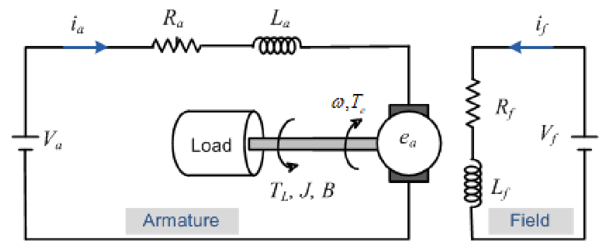

Fig.1. SEDCM equivalent circuit

The SEDCM dynamics and its load are represented by the following differential equation:

$$V_a(t) = e_a(t) + R_a i_a(t) + L_a \frac{di_a(t)}{dt} \quad (1)$$

(2) $$e_a(t) = K_f \ \omega(t) \ \phi(t)$$

(3) $$\phi(t) = \frac{N \ i_f(t)}{\phi_m}$$

(4) $$T_e(t) = K_t i_a(t) \ \phi(t) = T_L(t) + B\omega(t) + J\frac{d\omega(t)}{dt}$$

where : V_a – Applied voltage to the armature, I_a – armature current, L_a, R_a – Inductance and resistance of the armature coil, e_a - the back electromotive force, T_e – The electromagnetic torque, T_L – Load torque, B – Friction coefficient, J - moment of inertia, K_f, K_t - constants, ϕ - stator flux, ϕ_m - resultant of armature flux with stator flux, ω - rotor speed. Based on (2), (3) and (4) can be concluded that the SEDCM speed can be regulated by adjusting I_a on the armature side and adjusting I_f on the field side. If the speed control are only carried out on the armature side, then the SEDCM speed can be represented by the following Laplace transform :

(5) $$\omega(s) = \frac{T_e(s) - T_L(s)}{(Js + B)} = \frac{K_t}{(Js + B)} I_a(s)$$

(6) $$I_a(s) = \frac{V_a(s) - K_f\omega(s)}{(L_a s + R_a)}$$

Speed Control of SEDCM

SEDCM speed control is carried out only on the armature side by adjust I_a with the stator flux maintained constant. The armature current is regulated by adjusting the armature voltage using a four-quadrant DC chopper, as shown in Figure 2. Arrangement of armature current is proposed using the Fuzzy PI method. The Fuzzy PI controller control system is implemented with an Arduino Mega 2560 programmed with Simulink Matlab.

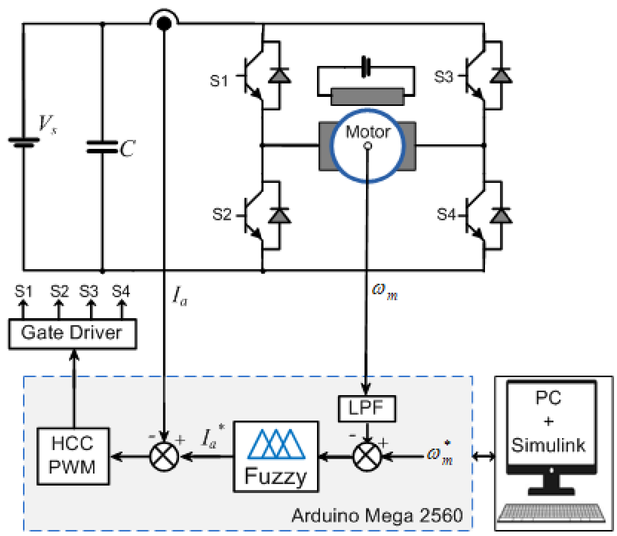

Fig.2. Proposed SEDCM speed control

Fuzzy PI system design is proposed to improve the performance of conventional PI controllers. If using a conventional PI controller, the reference armature current as controller output based on Figure 2 is :

(7) $$I_a^*(t) = K_p \ e(t) + K_i \int_0^t e(t)dt$$

(8) $$e(t) = \omega^*(t) - \omega(t)$$

where : K_p, K_i – Proportional gain and integral gain of PI controller with constant values, e – error of rotor speed. Mamdani's fuzzy inference system is proposed to obtain varying K_p and K_i gains, so it is expected to increase controller reliability. Figure 3 shows the scheme of the proposed Fuzzy PI controller.

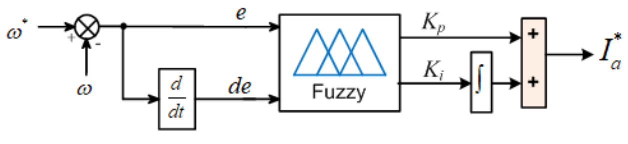

Fig.3. Scheme of Fuzzy PI controller

Figure 3 shows that the Fuzzy Logic Controller (FLC) is designed to be multi input multi output (MIMO). The FLC input are the speed error e and change error de, while the output are a proportional gain and an integral gain. The process of determining the output on FLC goes through several stages, namely fuzzification, an inference system based on a rule base and defuzzification [15]. In the fuzzification process, the input numerical data are mapped into a fuzzy membership functions. Figures 4(a) and 4(b) shows the membership functions of FLC inputs. Input 1 in Figure 4(a) represents the speed error e with five fuzzy membership functions, namely Negative Big (NB), Negative Small (NS), Zero (Z), Positive Small (PS) and Positive Big (PB). Input 2 in Figure 4(b) represents the error de with three fuzzy membership functions, namely Negative (N), Zero (Z) and Positive (P).

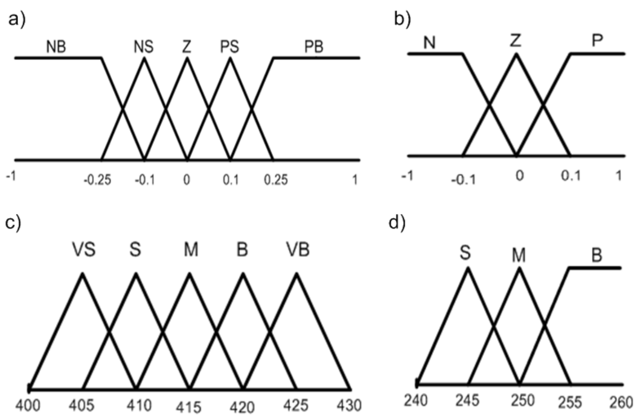

Fig.4. Member fucntions of Fuzzy Logic Controller.
(a) Input 1: speed error e, (b) Input 2: Error change de,
(c) Output 1: Integral gain K_i , (d) output 2: Proportional gain K_p

FLC with the Mamdani inference system presents output values with a fuzzy membership function. Figures 4(c) and 4(d) present the output membership functions, which consists of the output for K_i and the output for K_p . The K_i output are presented with five fuzzy membership functions, namely Very Small (VS), Small (S), Medium (M), Big (B) and Very Big (VB), as shown in Figure 4(c). The second output K_p are presented with three fuzzy membership functions, namely Small (S), Medium (M) and Big (B), as shown in Figure 4(d). If the first input is represented by x_1, second input is x_2, the first output is y_1 and the second output is y_2, then the determination of the FLC output is formulated by the following rule:

(9) If x_1 is e_n and x_2 is de_n then y_1 is K_p^n and y_2 is K_i^n

where e_n and de_n are the fuzzy sets of the n^{th} antecedent pair. K_p^n and K_i^n are the fuzzy sets of the n^{th} consequent pair. Based on Mamdani inference system with the max-min composition, the aggregate for each output are formulated by [16]:

(10)
$$\mu_{Kp^n} = \max_n \left[\min\left[\mu_{e^n}(input(i)), \mu_{de^n}(input(j)) \right] \right]$$
$$\mu_{Ki^n} = \max_n \left[\min\left[\mu_{e^n}(input(i)), \mu_{de^n}(input(j)) \right] \right]$$
$$n = 1,2,..,r$$

The rules for all input and output membership functions are shown by the rule base table in Figure 5.

a)

de \ e	NB	NS	Z	PS	PB
N	M	M	S	B	B
Z	M	M	S	B	B
P	M	M	S	B	B

b)

de \ e	NB	NS	Z	PS	PB
N	S	M	B	VB	VB
Z	VS	M	M	B	VB
P	VS	S	M	M	VB

Fig.5. Rule Base of Fuzzy Logic Controller.
(a) Rule base for output K_p, (b) Rule base for output K_i

After the output aggregate is obtained through the Mamdani fuzzy inference mechanism, defuzzification is then carried out, namely converting the output value from the linguistic fuzzy membership function into numerical data. Several defuzzification methods are possible. In this research, the defuzzification process was carried out using the center of area (COA) method. After the two FLC outputs are obtained, the reference armature current value I_a* as the Fuzzy PI controller output can be written as follows :

(11)
$$I_a^*(t) = K_p(t) + \int_0^t K_i(t)dt$$

The reference armature current obtained from the Fuzzy PI controller output will be compared with the feedback armature current from the sensor. The resulting current error will determine the power semiconductor modulation pulse in the DC chopper. This modulation pulse is determined using the concept of Hysteresis current control Pulse Widht Modulation (HCC PWM), as shown in Figure 2 The modulation pulses produced by HCCPWM are made in two patterns, where the first pattern is opposite to the second modulation pulse pattern. The first pulse is used to modulate semiconductors S1 and S4, while the second pulse is used for semiconductors S2 and S3, as shown in Figure 2.

Results and Discussion

The SEDCM speed control system with the Fuzzy PI controller was validated through laboratory experiments. The proposed Fuzzy PI controller system was tested on Terco SEDCM with a DC generator as load. Figure 6(a) shows the block diagram of the experiment. The experimental components consist of a 2 kW SEDCM equipped with a tachogenerator, 1 kW DC generator connected to the SEDCM shaft, load resistor connected to a DC generator, power supply, PC with matlab software,DC chopper and gate driver. The DC chopper is designed using four IRF460 MOSFETs with an IR2110 gate driver. The Fuzzy PI controller was designed with Matlab software and implemented on the Arduino Mega 2560 board as a control device. Figure 6(b) shows the hardware installation of the experiment.

The parameters of each component used in the experiment are described in Table 1 below

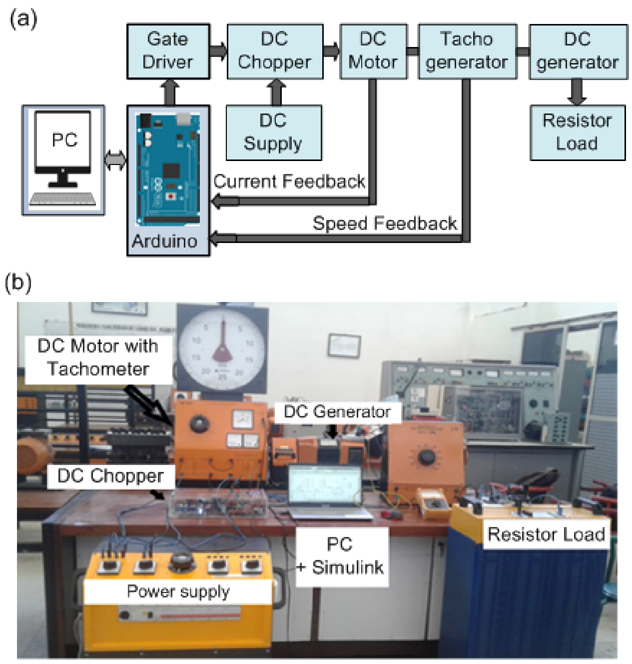

Fig.6. Experimental setup.
(a) The block diagram of the experiment, (b) Hardware setup

.Table 1. The experimental parameters

Component	Parameters	Value
DC Motor	Type	MV1036-225
	Power	2 kW
	Armature voltage	220 V
	Armature current	12 A
	Excitation voltage	220 V
	Excitation current	0.8 A
	Speed	1400 rpm
DC Generator	Type	MV1006
	Power	1.2 kW
	Armature voltage	220 V
	Armature current	5.5 A
	Excitation voltage	220 V
	Excitation current	0.55 A
	Speed	1400 rpm
Resistor Load	Power	3.3 kW
	DC Voltage	220 V
	DC current	15 A
Tachogenerator	Sensitivity	14 V / 1000 rpm

The experiment was carried out in two stages. The first experiment was carried out with varying motor speed and the DC generator load was made constant without connecting a load resistor. Firstly, the motor reference speed ω_m* a is set at 500 rpm, then increased to 700 rpm at 10 seconds and increased again to 800 rpm at 15 seconds, then reduced to 600 rpm at 20 seconds, as shown in Figure 7(a). The experimental results show that the motor speed can follow the reference speed even though the value is varied. These results show that the Fuzzy PI controller design for controlling SEDCM speed has worked well, as shown in Figure 7(a). These results confirm that the Fuzzy PI controller has successfully controlled the armature current according to the value required to control motor speed. Figure 7(b) shows the response of the reference armature current I_a* as the output of the Fuzzy PI controller in controlling motor speed. Based on (5), the armature current is directly proportional to the motor speed. Figure 7(b) shows that when the motor reference speed is increased, the Fuzzy PI controller also produces a greater reference armature current, so that the motor speed remains controlled even though it is varied. Figure 7(b) also shows that the HCCPWM parameter design has also provided a smooth feedback armature current response that matches the reference armature current produced by the Fuzzy PI controller. The combination of reliable HCCPWM action and the Fuzzy PI controller provides a motor speed response in accordance with the reference speed.

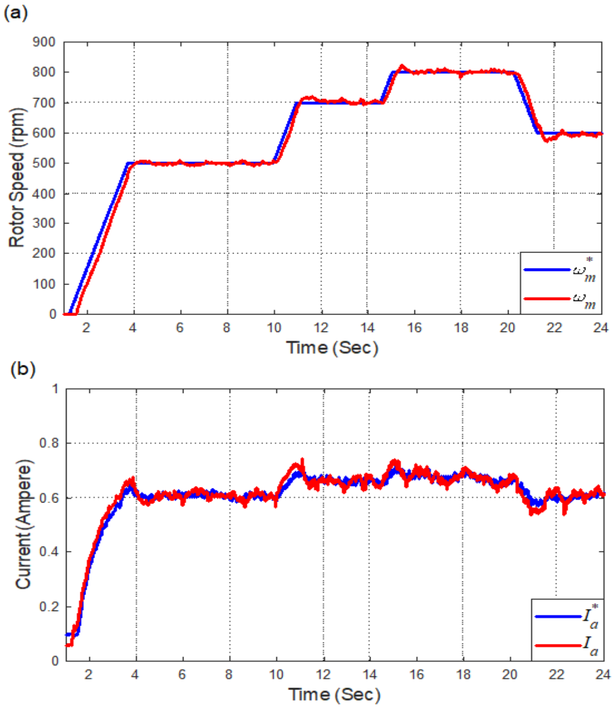

Fig.7. The first experimental results.
(a) Rotor speed, (b) Armature current

After the first experiment with constant load was completed, the second experiment was carried out with constant speed and varying load. The motor speed is kept constant at 500 rpm, while the load is varied via a variable resistor connected to a DC generator. In the initial condition, the motor load is set at 1.4 Ampere, then increased to 3 Ampere at 12 seconds. Figure 8(a) shows the motor speed response in the second experiment. The experimental results show that the motor speed can still follow the reference speed even though the load is varied. When the load is increased, the motor speed drops from the reference speed, but the motor speed can return to the reference value after some time, as shown in Figure 8(a). This shows that the proposed Fuzzy PI controller design has been successful in controlling motor speed even though the load is varied. If the motor is not controlled or the control system is less reliable, the motor speed will drop when the load is increased. The fuzzy PI controller design proposed in this paper has provided reliable control action, where the Fuzzy PI has successfully controlled motor speed both when the speed varies and when the load varies. The Fuzzy PI controller has succeeded in regulating the armature current so that the motor speed can be controlled according to the reference. Figure 8(b) shows the armature current response in the second experiment. Figure 8(b) shows that the armature current of the motor becomes greater than the reference current at the initial time. This arises because the motor requires greater torque for acceleration when the motor is loaded in its initial condition. , but this does not exceed the starting current surge. This can be reduced because the motor reference speed starts from zero, so that the motor's soft starting is maintained even though the motor has been loaded from the start.

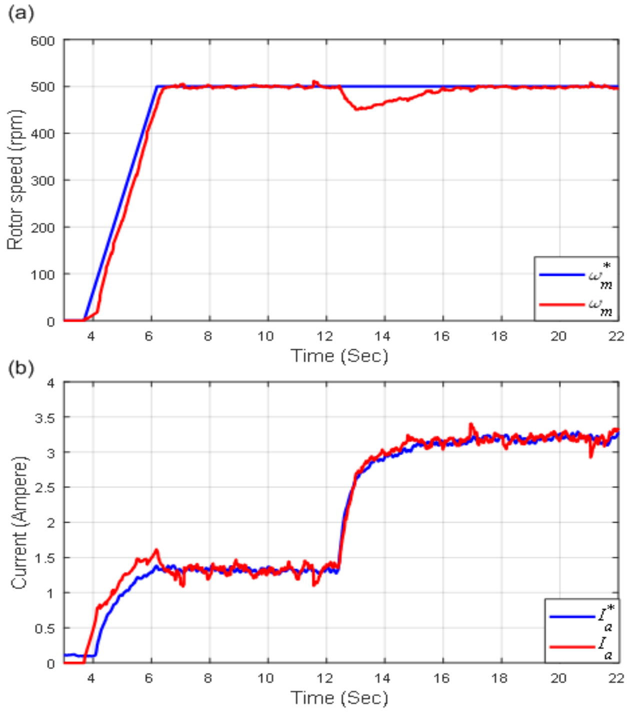

Fig.8. The first experimental results.
(a) Rotor speed, (b) Armature current

All experimental results show that the Fuzzy PI controller design has been successful in controlling motor speed in all conditions. This shows that the proposed control system has provided results as planned.

Conclusion

SEDCM speed control using four quadrant DC Chopper is proposed using a Fuzzy PI controller implemented with Arduino. The proposed control system is validated through experiments with various treatments. The experimental results show that the Fuzzy PI controller implemented with the Arduino Mega 2560 has successfully controlled motor speed, both under varying speed conditions and under varying load conditions.

***Authors**: Dr. Muldi Yuhendri. Instrumentation Control and Automation Research Group (ICARG), Department of Electrical Engineering, Universitas Negeri Padang, muldiy@ft.unp.ac.id; Risfendra. Ph.D. Department of Electrical Engineering, Universitas Negeri Padang, risfendra@ft.unp.ac.id; Emilham Mirshad. MT. Department of Electrical Engineering, Universitas Negeri Padang, emilhammirshad@ft.unp.ac.id; Adam R. Sidiqi. M.Sc. Department of Electrical Engineering, Universitas Negeri Padang, adamsidiqi@ft.unp.ac.id.*

REFERENCES

[1] Luigi A., Diego T., Design of Electric Motors and Power Drive Systems According to Efficiency Standards, *IEEE Trans. Ind. Electron.*, 68(2021), No. 10, 9287–9296, doi: 10.1109/TIE.2020.3020028.
[2] Ali N. H., Rasha A. M., Safwan A. H., Automatic Speed Control of DC Series Motor by Using Arduino, *Prz. Elektrotechniczny*, 99(2023), No. 3, 137–141.
[3] Mohamed M. R. A., Atef A. E., Zuhair M. A., Abdallah M. H., Torque control of a 5 KW, 220 V separately excited DC motor using microcomputer, *Int. J. Power Electron. Drive Syst.*, 14(2023), No. 2, 727–740.
[4] Jun Y., Han W., Liang H., Shihua L., Robust Predictive Speed Regulation of Converter-Driven DC Motors via a Discrete-Time Reduced-Order GPIO, *IEEE Trans. Ind. Electron.*, 66(2019), No. 10, 7893–7903
[5] Ramon S. O., Alfredo R. C., Eduardo H. M., Hierarchical Flatness-Based Control for Velocity Trajectory Tracking of the

‘DC/DC Boost Converter-DC Motor’ System Powered by Renewable Energy, *IEEE Access*, 11(2023), No. 4, 32464–32475.
[6] Sedar E., Davut I., Musa Y., Efficient Speed Control for DC Motors Using Novel Gazelle Simplex Optimizer, *IEEE Access*, 11(2023), No. 9, 105830–105842.
[7] Sariga S., Vinod B. R., MRAS Based Speed Control of DC Motor with Conventional PI Control — A Comparative Study, *Int. J. Control. Autom. Syst.*, 20(2022), 1–12.
[8] Jagannath S., Sohom C, Advanced Nonswitching Discrete Time Sliding Mode Control for Speed Tracking of a DC Motor, *IEEE Trans. Ind. Appl.*, 58(2022), No. 5, 6217–6226.
[9] Adel B., Mohamed A., Abdellah D., Fateh M., Ahmed B., Improvement of the linear quadratic regulator control applied to a DC-DC boost converter driving a permanent magnet direct current motor, *Int. J. Electr. Comput. Eng.*, 13(2023), No. 6, 6131–6140.
[10] Ihechiluru S. O., Clinton O. E., Robust control of a DC motor, *Heliyon*, 6(2020), No. 12.
[11] Ahmad M. Z., Mohammad E., Embedded two level direct adaptive fuzzy controller for DC motor speed control, *Ain Shams Eng. J.*, 9(2018), No. 1, 65–75.
[12] Trong T. N., The neural network-based control system of direct current motor driver, *Int. J. Electr. Comput. Eng.*, 9(2019), No. 2, 1445–1452.
[13] Rezoug N., Zerikat M., Chekroun S., An Efficient Fuzzy PI Approach to Real-time Control of a ROS Based Mobile Robot, *Prz. Elektrotechniczny*, 98(2022), No. 2, 1–5.
[14] Ye N., Schagin A., Ye H. L., Speed control of DC motor by using neural network parameter tuner for PI-controller, *Proc. 2019 IEEE Conf. Russ. Young Res. Electr. Electron. Eng. ElConRus 2019*, 2152–2156.
[15] Zhao H. L., Jie N., Hua L. W., Lie C., Xiao H. L., Ming Y. L., Switched PI Control Based MRAS for Sensorless Control of PMSM Drives Using Fuzzy-Logic-Controller, *IEEE Open J. Power Electron.*, 3(2022), No. 6, 368–381.
[16] Nelson L. M., Nihat İ., Murat L., “Control and performance analyses of a DC motor using optimized PIDs and fuzzy logic controller,” *Results Control Optim.*, 13(2023), No. 5, 1-27.

.