

Application of Neural Networks for the Control and Speed Regulation of Separately Excited DC Motor

Abdoulaye KEBE¹, Cheikh Ahmadou Bamba NIAN², Moustapha DIOP³ and Ibrahima GUEYE⁴

¹⁻⁴ENSETP/STI-Université Cheikh Anta DIOP/, Dakar, Senegal

Email: abdoulaye.kebe@ucad.edu.sn, ng7ahmada@gmail.com, moustapha17@ucad.edu.sn, ibrahima64.gueye@ucad.edu.sn

Abstract—Our study focuses on the application of neural networks for the control and regulation of the speed of a separately excited DC machine. To do this, we conducted a comparative study of our intelligent control with a traditional control method such as PID. The objective of this comparison is to better evaluate the performance criteria of our intelligent system in terms of speed, accuracy, and robustness against load variations. The work includes modeling of the separately excited DC machine, tuning of PID parameters, learning and implementation of the neural block on our system, and simulation of both control methods using the MATLAB/Simulink environment.

The simulation results have demonstrated that the neural control exhibits significantly better speed regulation performance compared to the traditional PID control.

Index Terms— Artificial Neural Network, Neural Control, PID Controller, Separately Excited DC Machine.

I. INTRODUCTION

In the context of electrical engineering, separately excited direct current motors (SE-DCM) are distinguished by their ability to provide precise speed control, making them essential machines in many industrial fields. Specifically studying the control and regulation methods for these motors is therefore of paramount importance to ensure optimum machine performance.

Two approaches are widely used in the literature: Proportional Integral Derivative (PID) control and the use of artificial neural networks (ANN) for the control and speed regulation of these machines. [2] [5][7]

PID control is a well-established traditional control method for DCM speed regulation. Based on three terms; proportional, integral and derivative; this approach aims to minimize the static error of the speed. The PID tuner functionality available on matlab will be used as a method for tuning the PID parameters [1]. Although widely used, PID control may have limitations in terms of dynamic accuracy and robustness, especially under variable operating conditions [1], [2], [3], [4] [5], [7].

With the development of artificial intelligence, ANNs are becoming a promising alternative for DCM speed regulation [2], [3]. These networks are able to model complex relationships between inputs and outputs, making them attractive candidates for dynamic system control such as DCM. Neural network architectures, such as multi-layer networks, can be trained to learn to control the SE-DCMs speed based on the provided inputs. This approach can provide better adaptation to load variations and superior performance under dynamic operating conditions.

Many studies have compared the performance of PID control with that of RNA for the speed regulation of DCM

[2], [5], [6], [7]. These studies generally evaluate stability criteria such as speed, robustness, dynamic precision characterized by overshoots, etc.

II. LITERATURE REVIEW

HS Dakheel et al in their paper "Simulation Model of Servo Motor by Using Matlab" [2], have offered an in-depth analysis of the application of ANNs over PID controllers to improve the performance of DCM. The results highlight the superiority of ANNs in terms of accuracy, responsiveness, and overshoot reduction over conventional PID controllers. However, the limitations such as handling the complexity of neural networks and robustness of AI-based approaches highlight the need for future research for practical and reliable application of these techniques in real industrial environments. Furthermore, the identified research opportunities such as exploring more advanced ANN configurations and studying the integration of deep learning techniques offer promising avenues to further improve the performance of DCM.

The paper "Implementing optimization of PID controller for DC motor speed control" [1], published by Yasir Ghazi and Ahmed Aladely presents a novel approach for optimizing PID parameters in DC motor speed control. Using a genetic algorithm, the authors have demonstrated significant improvement in the performance of the PID controller, including faster response time, reduced overshoot, minimized steady-state error, reduced oscillations, and better ability to reject output disturbances. This study also compares the results obtained with the PID genetic algorithm with those of the MATLAB PID Tuner software and the modified Ziegler-Nichols (MZN) method. The findings of this research highlight the effectiveness of the proposed approach in improving the performance of DC motor speed control systems, which could have significant implications in various industries and applications requiring precise and efficient motor speed control.

The study conducted by Jeen Ann Abraham and Sanjeev Shrivastava [5], focuses on DC motor speed control using machine learning algorithm. The comparative study between proportional integral derivative controller and neural network controller demonstrated the superior efficiency of neural network controller in control systems, providing significant benefits over an extended period. This research highlights the growing importance of neural networks and machine learning in modern control applications, emphasizing the potential of machine learning algorithms to improve the accuracy and robustness of controllers.

The study conducted by Smith et al. (2023), titled "Application of Neural Networks for Speed Control of Separately Excited DC Machines" [7], explores the use of artificial neural networks for precise and fast speed control of separately excited DC machines. They compared the performance of their intelligent control based on neural networks with a traditional control using a Proportional Integral (PI) controller. The study used MATLAB's ANN toolbox and Simulink library to simulate both control methods. The simulation results showed that the neural controller exhibits a faster response to speed fluctuations and more precise control compared to the traditional PI controller. The authors managed to achieve a moderate error of $1e-7$ and 8 after 5000 cycles of neural network training.

III. MODELING OF THE DC MOTOR

Modeling the DC motor consists of determining from the model of our system (figure1), the electrical, mechanical and electromechanical equations of operation. [1]

By using the mesh law at the armature level, we will have the following electrical equations:

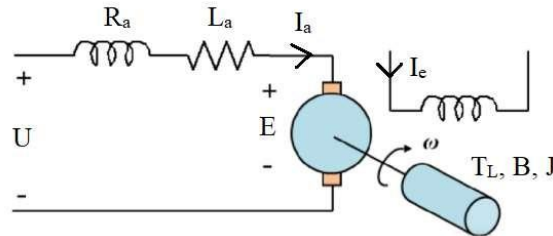


Figure 1: Separately Excitation DC Motor Model

$$u(t) - e(t) = \varepsilon(t) = R \cdot i(t) + L \frac{di(t)}{dt} \quad (1)$$

We will have the mechanical equations: $T_{em}(t) - T_r(t) = J \frac{d\Omega(t)}{dt} + f \cdot \Omega(t)$ (2)

The electromechanical equations will be: $e(t) = K \cdot \Omega(t)$ (3) $T_{em}(t) = K \cdot i(t)$ (4)

The Laplace transformation [2] of these equations gives us:

$$(1) \rightarrow U(p) - E(p) = \mathcal{E}(P) = (R + L_s).I(p) \quad (5)$$

$$\text{Either } H_1(p) = \frac{I(p)}{\mathcal{E}(p)} \rightarrow H_1(p) = \frac{1}{R+L_p} \quad (6) \text{ Equation(4) gives : } T_{em}(P) = K.I(p) \quad (7)$$

$$\text{Either } H_2(p) = \frac{T_{em}(p)}{I(p)} \rightarrow H_2(p) = K(8) T_{em}(p) - T_r(p) = Js\Omega(p) + f.\Omega(p) \quad (9)$$

The mechanical equation will be (2):

$$\text{Either } H_3(p) = \frac{\Omega(p)}{T(p)} \rightarrow H_3(p) = \frac{1}{J_p + f} \quad (10) \text{ Equation(3) gives : } E(p) = K.\Omega(p) \quad (11)$$

$$H(p) = \frac{E(p)}{\Omega(p)} \rightarrow H_4(p) = K \quad (12)$$

The transfer function of the MCC-ES is: $G(p) = \frac{K}{L_j.p^2 + (R_j + L_f).p + (R_f + K)}$ (13)

With a friction constant that is very low we can make the following approximations:

$$R_j + L_f \approx R_j \text{ and } R_f + K^2 \approx K^2 \quad (14)$$

$$G(p) = \frac{G}{\tau_e \cdot \tau_m \cdot p^2 + \tau_m \cdot p + 1} \quad (15) \quad G_0 = \frac{1}{K} ; K_e = K_c = K ; \tau_e = \frac{L}{R} ; \tau_m = \frac{RJ}{K_e \cdot K_c} \quad (16)$$

Consider the following physical parameters:

TABLE I: IMPLEMENTATION PARAMETERS OF DC MOTOR [1]

Settings	Values
Inductance armature	L = 0.1215H
Armature resistance	R = 11.2 Ω
Rotor inertia	J _m = 0.02215 kg-m ²
Viscous friction coefficient	B _m = 0.002953 Nm-s/rad
Constant torque motor	K _t = 1.28 Nm/A
Back emf constant	K _b = 1.28 V s/rad
Speed	1500 rpm

Substituting the motor data (Table 1) into (15), we will have the open loop transfer function of the system:

$$G(p) = \frac{1.28}{0.0027p^2 + 0.2481.p + 1.671} \quad (17)$$

IV. PID IMPLEMENTATION

To determine the parameters of our corrector, we will use the Transfer Function Based functionality (PID Tuner App). To do this, we must simulate the diagram in figure 2 on Matlab/Simulink.

We adjust the parameters until we have a system that is not only fast but also has less overshoot and good precision. By playing on the response time and on the robustness, we obtained the final performances of our corrected system (see figure3) and the response after correction is clearly visible on the diagram of Figure4.

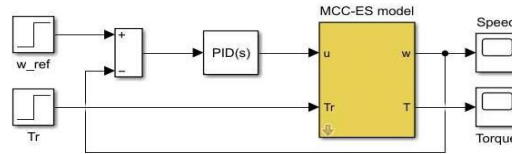


Figure 2: MCC Diagram With PID

Performance and Robustness		
	Tuned	Block
Rise time	0.113 seconds	3.84 seconds
Settling time	0.416 seconds	7.44 seconds
Overshoot	7.57 %	0 %
Peak	1.08	0.999
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	69 deg @ 13.4 rad/s	129 deg @ 1.16 rad/s
Closed-loop stability	Stable	Stable

Figure3: Corrected system performance and PID settings



Figure 4: System response after correction(Tuner response)

Our PID parameters found from the PID-Tuner functionality are:

$$K_p = 2.08745139954927 \quad (18) \quad K_i = 25.9627704034806 \quad (19)$$

$$K_d = 0.0107675813686319 \quad (20) \quad N = 1527.66309381817 \quad (21)$$

V. IMPLEMENTATION OF THE NEURAL CORRECTOR ON THE SYSTEM

A neural network is a system of interconnected nonlinear operators, receiving signals from the outside through its inputs, and delivering output signals, which are the activities of certain neurons [2], [3]. For our case, we will learn from the data collected during the simulation with the PID corrector. We record the data at the input and output of the PID from workspaces available on matlab.

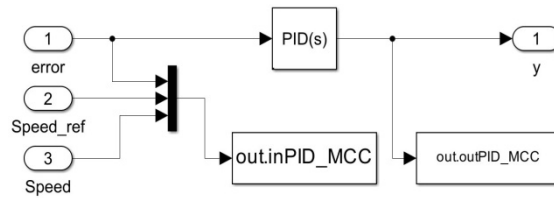


Figure 5: Workspace Data Records

So we record on the inPID_MCC the speed setpoint, the error and the speed response. And the corrected output of the PID on the outPID_MCC.

With the step time of the engine torque (T), we will simulate an idle speed up to time $t=1$ s then we will subject the engine to a load torque of the order of 50 Nm. The recording was done with a stop time of 10 and a sample time of the powergui at 5e-6s. O collected:

$$3 \times 1000008 \text{ input data points} \quad (22) \quad 3 \times 1000008 \text{ output data points} \quad (23)$$

We do neural learning with the nftool function on matlab and we collect the following results:

The learning progression is given in the diagram of Figure 7. A performance of 7.42×10^{-10} tending towards 0 a gradient of 3.86×10^{-5} and a Mu of 1×10^{-6} .

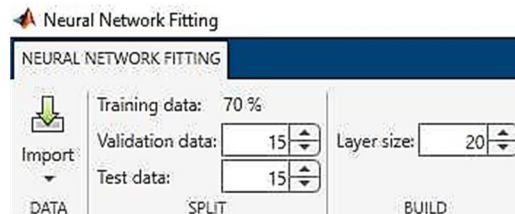


Figure 6: Neural Network Parameterization

Unit	Initial Value	Stopped Value	Target Value
Epoch	0	1000	1000
Elapsed Time	-	00:22:34	-
Performance	8.68×10^3	7.42×10^{-10}	0
Gradient	3.86×10^4	3.86×10^{-5}	1×10^{-7}
Mu	0.001	1×10^{-6}	1×10^{-10}
Validation Checks	0	0	6

Figure 7: Progression of Neural Learning

Training Results			
Training start time: 09-May-2024 18:02:28			
Layer size: 10			
	Observations	MSE	R
Training	700006	7.4154e-10	1.0000
Validation	150001	7.1591e-10	1.0000
Test	150001	6.0771e-10	1.0000

Figure 8: Learning Outcomes

We have a mean square error (MSE) of the order of 10^{-10} (see Figure8), which tends towards 0 and a regression perfectly equal to 1 as desired (see Figure 9).

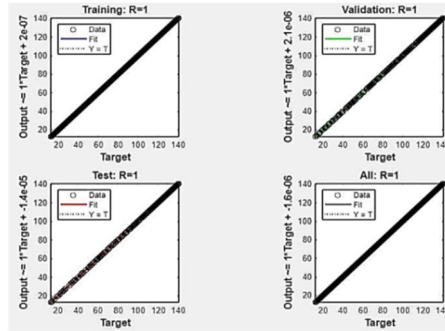


Figure 9 : Regression Curves

The obtained neural block is then implemented on the system with the introduction of a limiter.

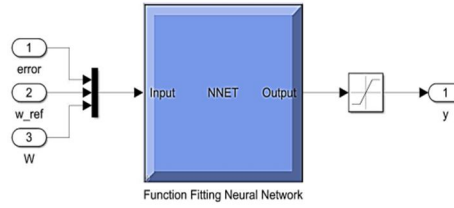


Figure 10 : Neural block implemented on the system

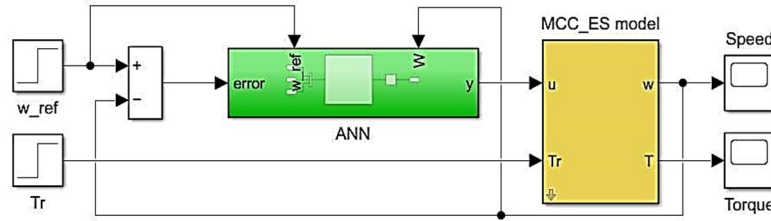


Figure 11: BF diagram with the neural corrector

VI. SIMULATION, RESULT AND DISCUSSIONS

For the simulation, we subjected the SE-DCM to no-load operation for the first second and then to nominal operation with a torque $Tr = 50$ Nm at time $t = 1$ s until $t = 2$ s. The block diagrams of the two systems (DCM with the PID and DCM with the neural block) are shown in figure 12.

After simulating our system (composed of the PID and the ANN), we obtained on the same curve the speed responses of the system with the PID, of the system with the ANN and of the reference speed on the diagram in figure 13.

From the speed responses obtained with the figure13, we can calculate for each system (with PID and with ANN), the parameters like the first two overshoots and the response time at 5% for a comparison in terms of performances. (See table 2)

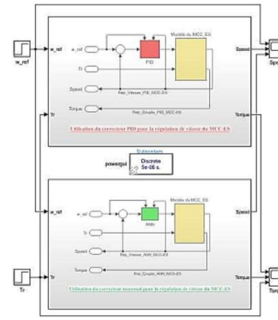


Figure 12: Block Diagrams of Both Systems on Simulink

TABLE II: PERFORMANCE CRITERIA OF PID AND ANN SPEED RESPONSES WITH BOTH REGIMES

Speeds	Empty		In charge	
	PID	ANN	PID	ANN
1 st Overshoot (%)	7.6	0.467	6.2	0.933
2 nd Overshoot (%)	0.267	0.333	0.2	0.067
T _{r5%} (ms)	336	199	151	0

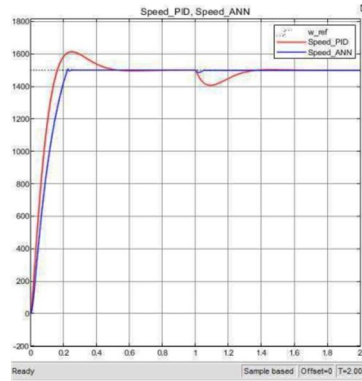


Figure 13: System speed responses with PID and with ANN

For speed control, in both scenarios, neural control shows more promising results than PID. Indeed, in the no-load regime, ANN shows a lower first overshoot, an almost negligible second overshoot and a much shorter 5% response time compared to PID. Similarly, under load, ANN shows a significant reduction in the first two overshoots and the 5% response time compared to PID.

For the couples, we obtained on the same curve the responses of the system with the PID and of the system with the ANN on the diagram in Figure14.

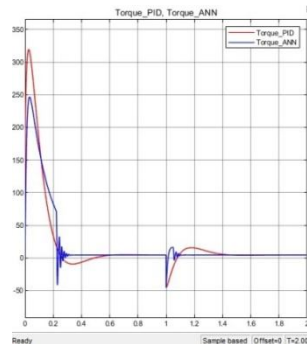


Figure 14: Couple Responses With PID and ANN

From the couples responses obtained with the Figure14, we can calculate for each system (with PID and with

ANN), the parameters like the peak at start-up and the response time at 5% for a better comparison in terms of performances of the two systems.(See Table3)

TABLE III: PERFORMANCE CRITERIA OF PID AND ANN COUPLE RESPONSES WITH THE TWO REGIMES

Couples	Empty		In charge	
	PID	ANN	PID	ANN
Peak start	319.5	246.8	6.2	0.933
T _{r5%} (ms)	607	344	649	137

For torque control, the ANN also outperforms PID in both regimes. ANN exhibits lower starting peaks and 5% shorter response times, both at no-load and under load, compared to PID.

The comparative results between PID control and neural control (ANN) provide valuable information on their respective performances in both operating regimes, no-load and under load.

VII. CONCLUSION

The results indicate that neural control generally provides better performance than PID control in terms of speed and torque regulation, both at no-load and under load. These advantages include significantly reduced overshoots and faster response times, suggesting that ANN is more effective in regulating load variations and providing a more accurate and faster response under these conditions.

REFERENCES

- [1] YG Rashid and AMA Hussain, "Implementation of PID Controller Optimization for DC Motor Speed Control". Indonesian Journal of Electrical Engineering and Computer Science, vol. 2, no2, page 657-664. (2021).
- [2] HS Dakheel, ZB Abdullah, NS Jasim, and , SW Shneen, "Simulation model of ANN and PID controller for DC servo motor using Matlab/Simulink."TELKOMNIKA (Telecommunication, Informatics, Electronics and Control), vol. 20, no. 4, page 922932. (2022).DOI: 10.12928/TELKOMNIKA.v20i4.23248
- [3] S. Yogesh, M. Gupta, and M. Garg."DC motor speed control using artificial neural network."International Journal of Modern Communication Technologies and Research, vol.2, no.2, page 19-24. (2014).
- [4] M. Alhanjouri, "DC Motor Speed Control Using Artificial Neural Network." International Journal of Science and Research (IJSR), vol.7, no3, page 2140-2148. (2017).
- [5] J. A. Abraham, and S. Shrivastava, "DC Motor Speed Control Using Machine Learning Algorithm." Int. J. Eng. Res. Technol, vol.7, no4, page 456-470. (2018).
- [6] Y. Guo, and M. E. A. Mohamed, "DC motor speed control using an ANFIS-based hybrid PID configuration controller." IEEE Access, vol.8, page 125638-125647. (2020).
- [7] MR Zaidan,"Impact of artificial neural network for DC motor speed control compared to conventional controller."Journal of Engineering and Applied Sciences, vol.13, no21, page 9156-9163. (2018).
- [8] S. Oladipo, Y. Sun, & Z. Wang, "Optimization of PID controller with metaheuristic algorithms for DC motor drives." Int. Rev. Electr. Eng, vol.15, no5, p 352-381. (2020)