

DC Motor Speed Control Using Machine Learning Algorithm

Jeen Ann Abraham

Department of Electronics and Communication.
RKDF College of Engineering
Bhopal, India.

Sanjeev Shrivastava

Department of Electronics and Communication.
RKDF College of Engineering
Bhopal, India.

Abstract—The control and automation has been accelerated to its next generation with the introduction of machine learning algorithms and advanced computing capability of modern processors, in control systems for various applications. Many researches are undergoing in the field of controller design and tuning using the deep learning algorithms. This paper addresses a comparison study on effectiveness and performance between a proportional integral derivative controller and a neural network controller which was designed using the existing tools and libraries from Matlab/Simulink package. Initially, the proportional integral derivative controller was tuned for a DC motor model with a balance between robustness and faster response. Later, the neural network controller was trained for the same model followed by optimization of the controller. Then both controllers were experimented using the same model of DC motor by simulation in Matlab/Simulink. Followed to this, another experimentation was done by changing the values for moment of inertia of rotor followed by simulating the controller response without re-tuning the controller for newer model. Result analysis was done by studying the response for both controller from the plotted graph and root mean square error calculation. Simulation experiments and comparison between the controllers demonstrate that the controller with machine learning algorithms is better in control systems for a longer period.

Keywords— Machine Learning; Controller; Tuning; Neural Network; Simulation

I. INTRODUCTION

ANN (Artificial Neural Networks) are found in most of the modern applications since last decade. Even though it was identified in 1943 by Warren S. McCulloch and Walter Pitts [1], the inception of deep learning algorithms and advanced computing capability makes suitable environment for the ANN to find its place in modern world applications such as pattern recognition system, speech recognition system, diagnosing disease in medical applications and stock market prediction. Nowadays, the application of neural networks starts to evade the control and automation field with the help of AI (Artificial Intelligence) and machine learning algorithms. An intelligent control of wind turbine proposed by Mahmoud et al. in [2] explains the use of AI in wind turbine control applications. Behind the mask, the machine learning maps the inputs to outputs by exploring the correlation between both, followed by approximation. Despite several advanced techniques and algorithms, the Proportional-Integral-Derivative (PID) control still has its solid space in control systems used in majority of industrial applications because of its efficiency and performance [3] [4]. However, the performance for nonlinear systems varies and depends on the tuning of controllers.

The paper is structured as follows. Background and methodology is explained in Section II and Section III followed by Tuning of PID controller in Section IV. The Section V discusses about neural network predictive controller and simulation model followed by the comparison of results obtained from both the controllers in Section VI. Finally, Section VII summarizes the paper and outlines future work.

II. BACKGROUND

The purpose of this research is to improve the tolerance of controller output for changes in system parameters with respect to time. The benefits of increasing the tolerance is that the controller can mitigate the disturbances in plant process to certain extent. For example, in case of motor operated valve used in polymer application, the valve friction increases if it's not operating for a certain period due to deposit of particles between seating area. This results in change in moment of inertia of rotor and the controller will not be able to give proper output because the tuned controller parameters correspond to the system without any friction. The PID controllers may produce overshoots as the performance of system deteriorates due to change in various system parameters. Self-tuning control strategies well suits for such system with time varying characteristics. Deep learning controllers find its applications in controlling complex nonlinear processes because of the approximation properties to predict precise output from the sample inputs. DC motors are widely used in various control applications as final control element. These applications vary from robot arm control used in automation industry to feed pump driver in rocket engines. In literature [7] the authors proposed a deep learning controller for speed control of DC motor which was designed by learning the PID controller. In this paper, a study was conducted for improving the performance and effectiveness of neural network controller to that proposed in [7] using MATLAB/ Simulink.

III. METHODOLOGY

With graphical user interface, Neural network tool box and several libraries, simulation models for deep learning applications can be developed using MATLAB/Simulink software package. The case study for control application was done by modelling a DC motor with voltage as input and speed as output followed by simulating the same using different controllers in MATLAB/Simulink. The dynamic equations for a DC motor is as follows:

$$\frac{dw}{dt} = \frac{1}{J}(K_t i - bw) \quad (1)$$

$$\frac{di}{dt} = \frac{1}{L}(V - Ri - K_e w) \quad (2)$$

Where, J is the moment of inertia of the rotor, K_t is motor torque constant, i is the armature current, b is motor viscous friction constant, L is electric inductance, R is electric resistance, and K_e is electromotive force constant [7]. The model is then simulated with a PID controller and Neural Network Predictive Controller (NNPC). Before simulation the PID controller was tuned and NNPC was trained for the designed DC motor. Levenberg-Marquardt backpropagation algorithm is faster and have good performance that makes it well suitable for training neural networks compared to resilient back propagation algorithms [5][6]. Finally, the study on tolerance of controller response for change in friction of motor was done by increasing the moment of inertia and simulating the model without re-tuning the controllers. The data analysis was performed by calculation of Root Mean Square Error (RMSE) between set speed and motor output speed by both controllers.

IV. PID CONTROLLER AND TUNING

PID controllers has only three parameters, but it is not easy to find accurate values to achieve the system stability. There is a general trade-off between the robustness and performance of the control systems [8-9]. Although PID controller can be implemented both parallel and series forms, the parallel forms can achieve better performance for oscillatory processes [10]. Initially the DC motor is simulated with PID controller for experimenting the performance versus robustness.

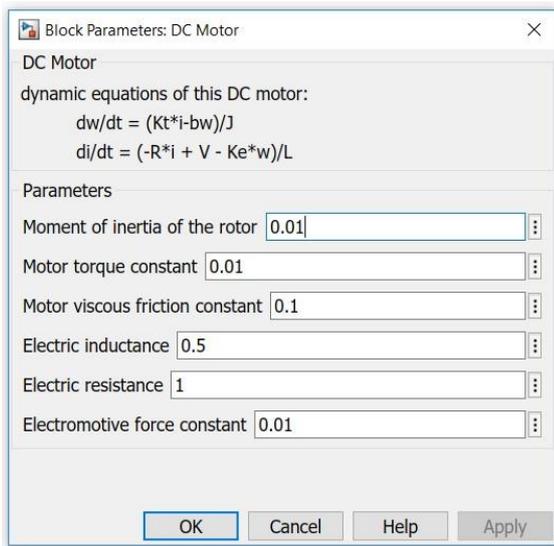


Fig. 1. DC motor parameters.

The values for the DC motor parameters are shown in Figure 1. These values are kept same as discussed in literature [7] for a comparison study on performance between both controllers. The equation for continuous time PID controller with first order derivative filter is as follows:

$$C = K_p c \left(1 + \frac{1}{T_i s} + \frac{T_d s}{\frac{T_d}{N} s + 1} \right) \quad (3)$$

Where, K_p is the proportional gain, T_i and T_d are the integral and derivative times and N is the first-order derivative filter divisor. The PID controller is tuned for the DC motor and results shows that an increase in robustness degrades the performance and vice versa. The system was tuned for better performance, but the controller output starts oscillating for a transient period as shown in Figure 2. In Figure 3 the system was tuned with increased robustness, but the result was a sluggish response. As a result, the plant was re-tuned to achieve a trade-off between robustness and performance.

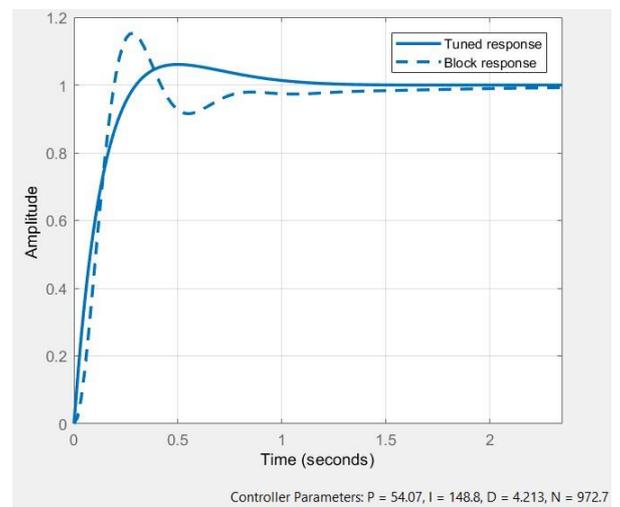


Fig. 2. PID controller output with rapid response.

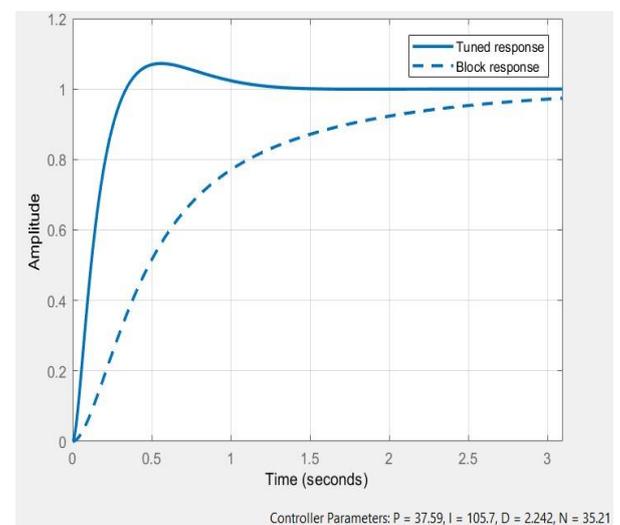


Fig. 3. PID controller output for better stability.

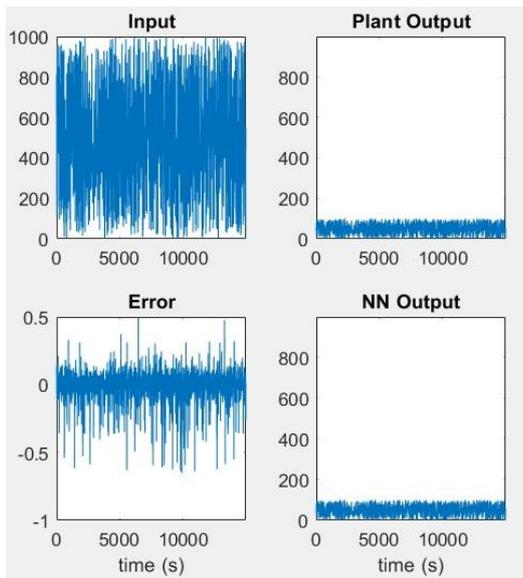


Fig. 4. Training data.

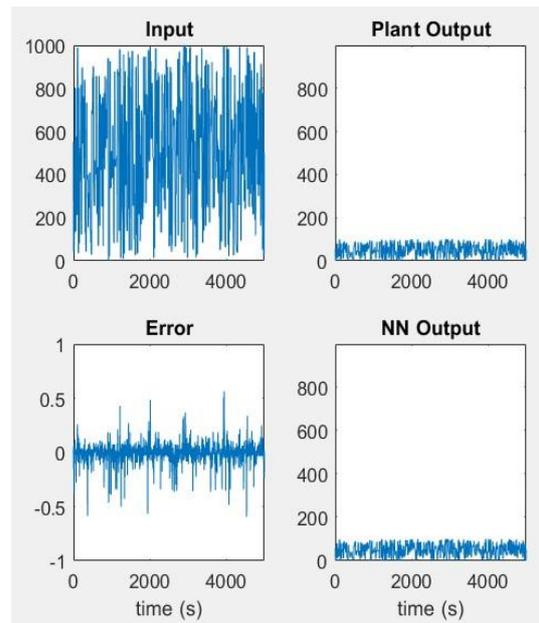


Fig. 5. Validation data

V. NEURAL NETWORK PREDICTIVE CONTROLLER

The algorithm for NNPC is based on minimizing the cost function over a finite period. The cost function is minimized to generate a control input that meets the constraints to track the reference trajectory by the plant within certain tolerance [11]. The equation for cost function is as per equation 4.

$$J = \sum_{j=N_1}^{N_2} (y_r(t+j) - y_m(t+j))^2 + \rho \sum_{j=1}^{N_u} (u'(t+j-1) - u'(t+j-2))^2 \quad (4)$$

Where N_1 and N_2 are the minimum and maximum costing horizon, N_u is the control horizon, y_r is the reference trajectory,

y_m is the predicted output of the neural network and ρ is the control weighing factor. The performance optimization is done by reducing the size of the interval of uncertainty using golden section search method discussed in literature [12]. The plant model must be developed to predict the future outputs using neural networks. The neural network plant model has a total of 14 hidden layers with 2 delayed plant inputs and outputs each at a sampling rate of 0.2. Finally, the controller is trained using Levenberg-Marquardt backpropagation algorithm with an iteration of 2000 times. These predictions are then used by the optimization algorithm to predict controller output. Figure 4 and Figure 5 shows the interpretation of data obtained from training and validation of the Neural network plant model. The control of DC motor with voltage V as input and angular speed w as output based on PID controller and neural network predictive controller in MATLAB environment is shown in Figure 5.

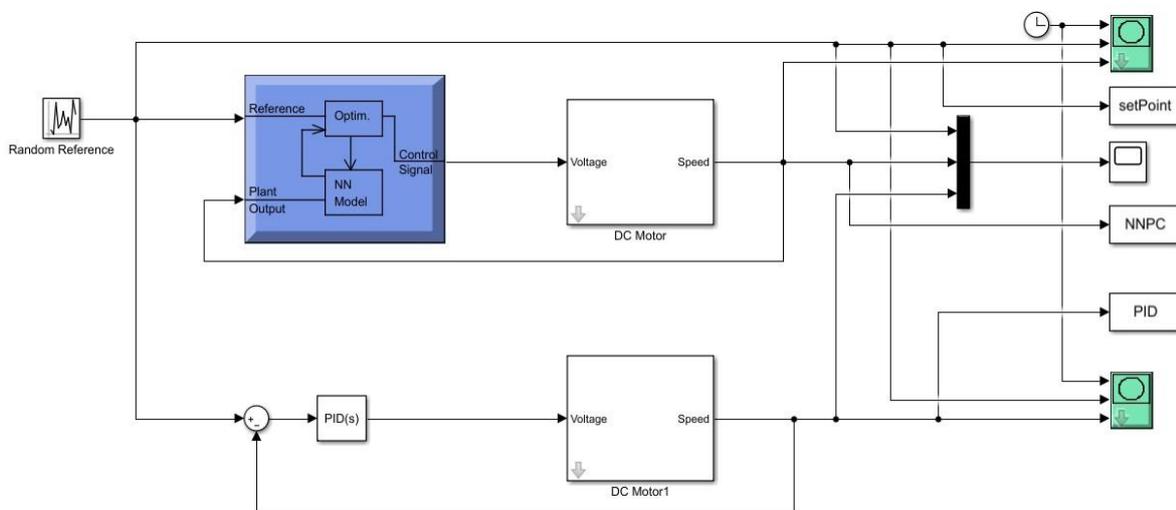


Fig. 6. Simulink model for feedback control of DC motor based on neural network controller.

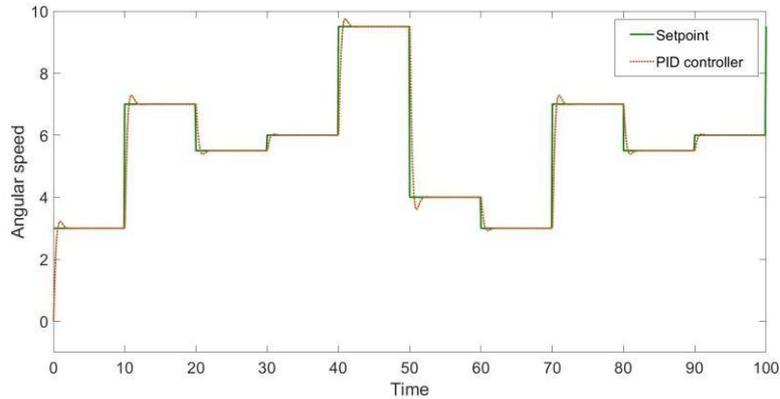


Fig. 7. Response from PID controller

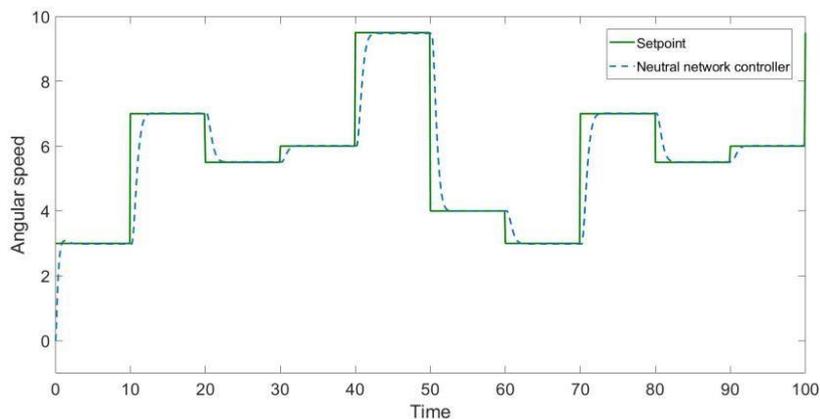


Fig. 8. Response from neural network predictive controller

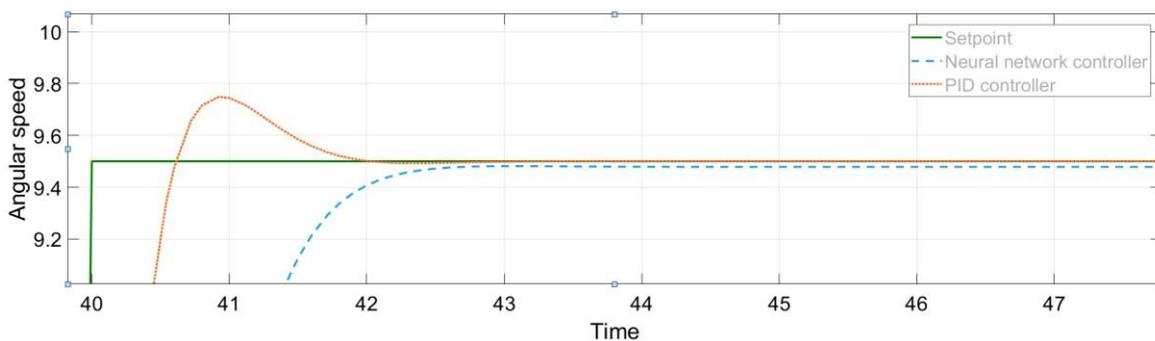


Fig. 9. Magnified view of output from both controllers

VI. SIMULATION RESULTS

A random reference block which generates normally distributed random numbers serves as the setpoint variable for simulation. The model plots setpoint and output from the DC motor with reference to time for each controller. The scope helps to compare the variation between the two controllers output. The simulation was done in MATLAB/ Simulink environment for a period of 100 seconds. Performance study was conducted by comparing both the controller output for overshoot and time to achieve system stability. Also, the RMSE is calculated on output from the DC motor for each controller and shown in Table 1. Another experiment was done by changing the moment of inertia for the rotor by increasing and decreasing the value by 10 percent after training the controller to study the response of DC motor for each controller. This was

done because minor change may happen in motor parameters eventually.

Figure 7 and Figure 8 shows the DC motor output for the PID controller and the neural network predictive controller.

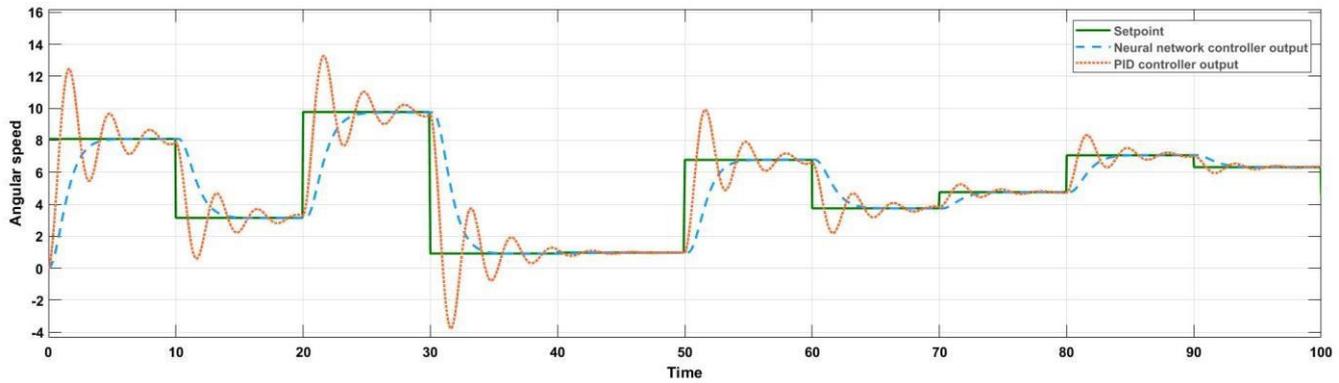


Fig. 10. Output from both controllers for increase in moment of inertia of rotor.

Further magnified view for the response of DC motor for both controllers is shown in Figure 9. It was observed that the PID controller output overshoots for each step change in input while the overshoot is negligible for the output from NNPC. On the other hand, accuracy for PID controller is higher than that of neural network controller. When the moment of inertia of rotor reduced by ten percent there was no large effect on controller output, but the same for PID controller starts oscillating for ten percent increase in the moment of inertia and as a result system becomes unstable. The NNPC shows the same output with minor delay. Figure 16 shows the response of both controllers with ten percent increase in moment of inertia.

TABLE I. Root Mean Square Error for both controllers.

Method	PID Controller	Neural Network Controller
RMSE	8.64×10^{-5}	7.21×10^{-4}

VII. CONCLUSION

The MATLAB based neural network controller and a PID controller was designed for the control of DC motor model. A comparison study was done on performance between both controllers and effect of change in inertia of rotor on the controller output to verify the application of machine learning in control systems. From the simulation results, it has observed that accuracy of NNPC is slightly less compared to PID controller. At the same time NNPC has better control on dc motor with minor delay for ten percent increase in moment of inertia of rotor, where the PID controller fails and thereby, explains that, there will not be any large change in controller output due to reduction in performance of equipment for a neural network controller. The results demonstrate the feasibility and advantage of using machine learning algorithms in control systems. The future work will be to improve the preciseness of the above neural network controller by more stable, robust machine learning algorithm with less time complexity and a real-time implementation of the controller in a DC motor speed control system.

REFERENCES

- [1] Warren S. McCulloch and Walter Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, Vol. 5, 1943, pp. 115-133.
- [2] Tawfek K. Mahmoud, Zhao Yang Dong and Jin Ma, "A developed integrated scheme based approach for wind turbine intelligent control," *IEEE transactions on sustainable energy*, Vol. 8, No. 3, July 2017.
- [3] G. K. McMillan and D. M. Considine, "Process/ industrial instruments and controls handbook," McGraw-Hill, New York, 1999.
- [4] Ang K.H., Chong G., Li Y., PID Control System Analysis, Design, and Technology. *IEEE Trans. Control Syst. Technol.* 2005, 13, pp. 559-576.
- [5] Arnold Reynaldi, Samuel Lukas and Helena Margaretha, "Backpropagation and levenberg-marquardt algorithm for training finite element neural network," *Proceedings of the 2012 sixth UKSIM/AMSS european symposium on computer modeling and simulation*, pp. 89-94.
- [6] Kişi Özgür and Uncuoğlu Erdal, "Comparison of three back-propagation training algorithms for two case studies," *Indian journal of engineering and material sciences*, vol 12 october 2005, pp. 434-442.
- [7] Kangbeom Cheon, Jaehoon Kim, Moussa Hamadache, and Dongik Lee "On replacing pid controller with deep learning controller for dc motor system," *Journal of automation and control engineering* Vol. 3, No. 6, December 2015, pp. 452-456.
- [8] Emre Sariyildiz, Haoyong Yu and Kouhei Ohnishi "A practical tuning method for the robust pid controller with velocity feedback," *Machines* Vol. 3, 2015, pp. 208-222.
- [9] K. Åström, K.J. and T. Hägglund, "PID controllers: theory, design, and tuning," *Instrument society of America, Research triangle park*, 1995.
- [10] Skogestad S., "Simple analytical rules for model reduction and pid controller tuning," *Journal of process control*, Vol. 13, 2003, pp. 291309.
- [11] Soloway D., and P. J. Haley, "Neural generalized predictive control," *Proceedings of the 1996 IEEE international symposium on intelligent control*, 1996, pp. 277-281.
- [12] Hagan, M.T., H.B. Demuth, and M.H. Beale, *Neural Network Design*, Boston, MA: PWS Publishing, 1996.