

Simulation Analysis of DC Motor Speed Control Using PID Controller

Pirah Peerzada^a, Wasi Hyder Larik^a

^aDepartment of Electrical Engineering, Mehran University of Engineering and Technology, Jamshoro, 76062, Pakistan

Corresponding author e-mail: (pirhpeerzada@gmail.com, wasilarik@gmail.com)

[Received on: 25-04-2021 Accepted on: 20-05-2021 Published on: 05-06-2021]

Abstract—Emerging technologies such as electric vehicles require high efficiency electric drives. DC motor in this case has greater efficiency, attractive speed-torque characteristics, and precise control. In this study, DC motor is modelled, and its speed control analysis is done under various loading conditions. The speed control is achieved through PID controller.

I. INTRODUCTION

DC motors have always been a great demand for various industries due to their high efficiency, precise speed control, and good torque speed characteristics. The DC motor is an energy conversion actuator, converting electrical energy (from the source) to mechanical energy. Due to the ease with which the speed can be adjusted to offer the desired performance characteristics under various conditions, the use of DC drives in different applications has increased in recent years [2]. The speed-controlled methods for DC motors have not been completely analyzed yet. Numerous approaches have been utilized for DC motor speed control. In EVs, DC motors experience disturbances due to variable loads which affects the performances and stability of system, However DC motor needs to be tested under various loading conditions. The ideal driving motors have linear characteristics of voltage to speed. In contrary, practical motors have nonlinear characteristics of voltage to speed. Practically, it is very difficult to analyze and control nonlinear responses. For maintaining the performance and stability, efficient controller is required.

Mathematical model for basic closed loop dc motor control and loop control with proportional integral and derivative (PID) controllers has been developed this modelling approach is useful for developing intelligent control method such as fuzzy control, neural network, and so on. In [4-5] the voltage and speed of DC motor is controlled using PID controller on MATLAB Simulink and Arduino Microcontroller. In [6-9] the authors have presented their research project on controlling the speed of DC motors with different suitable tuning methods for enhancing the control techniques. Nowadays many new techniques have been explored for the control of speed response of DC motors, among others the PID controller have been emerged to applicable due to its simple structure and robustness

to modelling the error. In [10], The speed of a DC motor is controlled with a MATLAB-based PID controller with the Variations in parameters and their comparison which is common in industrial processes. The MATLAB program is used to calculate and simulate a PID controller to supervise and control the speed response of a DC motor. For the test system, the PID-based approach algorithm worked well. The table below summarizes the effects of the PID controller parameters **kp**, **ki**, and **kd** on a closed loop system.

This paper is mainly concerned to improve the performance of DC motor's non-linear response. For its compensation, it is mandatory to develop a closed loop algorithm based on PID controller. The prime objective of this study is to propose PID controller based on closed loop algorithm for speed control of DC motor.

II. MODELLING OF DC MOTOR

A. MATHEMATICAL MODEL

The mathematical modelling of DC motor is done by applying Kirchhoff's Voltage Law to armature circuit of DC motor and Newton's second law of motion to mechanical portion of DC motor. For mathematical model of DC motor, we assume that

- The system input is source voltage supplied to the terminals of motor.
- Output of the system is angular speed (ω) at shaft
- Furthermore, viscous friction model is assumed, that is, the friction torque is proportional to the output of motor.
- Moreover, the current of armature circuit and magnetic field strength are proportional to DC motor's produced torque
- Motor torque is K_t times the armature current (i).

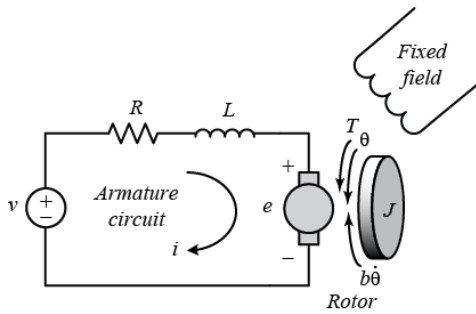


Figure 1. Circuit diagram of DC motor

As assumed above, DC motor's produced torque is proportional to the armature current and the strength of the magnetic field. In this example we will keep the magnetic field unchanged and, thus, that the motor torque becomes proportional to only the armature current i by a constant factor K_t as shown in the equation below.

$$T = K_t \cdot i \rightarrow$$

The back emf, e , is K_e times the angular velocity of the shaft.

$$e = K_e \dot{\theta} \rightarrow$$

In SI units, the motor torque and back emf constants are equal, that is, $K_t = K_e$; therefore, we will use K to represent both the motor torque constant and the back emf constant.

By applying second law of motion on shaft, shown in figure, we get

$$J\ddot{\theta} + b\dot{\theta} = K i \quad (i)$$

And applying KVL on RL circuit of fig.1, we get:

$$L \frac{di}{dt} + Ri = V - K \dot{\theta} \quad (ii)$$

Equation (i) and (ii) are modelled equations for a DC motor

B. TRANSFER FUNCTION

Transfer function of DC motor is derived by applying Laplace transform to equation (i) and equation (ii). Here, input of the system will be taken as applied voltage (V) to the DC motor while output is angular speed ($\dot{\theta}$) of the shaft of the DC motor. Thus, the obtained transfer function is:

$$\frac{\dot{\theta}(s)}{V(s)} = \frac{K}{(Ls+R)(Js+b)+K^2} \quad (iii)$$

The values of parameters are taken from www.ctms.engin.umich.edu, which are as following:

$$J = 0.01 \text{ kg.m}^2$$

$$b = 0.1 \text{ N.m.s}$$

$$K_e = 0.01 \text{ V/rad/sec}$$

$$K_t = 0.01 \text{ N.m/Amp}$$

$$R = 1 \text{ Ohm}$$

$$L = 0.5 \text{ H}$$

By putting the values of our parameters, transfer function becomes:

$$\frac{\dot{\theta}(s)}{V(s)} = \frac{0.01}{(0.5s+1)(0.01s+0.1)+0.0001} \quad (iv)$$

C. SIMULATION MODEL

The general block diagram of DC motor can be seen in fig. 2

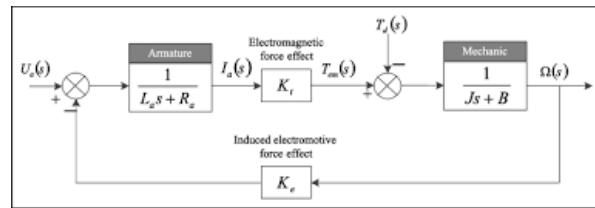


Figure 2. Block diagram of DC Motor

On the basis of this general block diagram, DC motor is modelled, without any external load, in Simulink/MATLAB that can be seen in fig 3. All the values of considered parameters are employed in this model with a constant input voltage of 12v. Output response, that is angular speed vs time graph, can be observed through the scope connected to the model of DC motor.

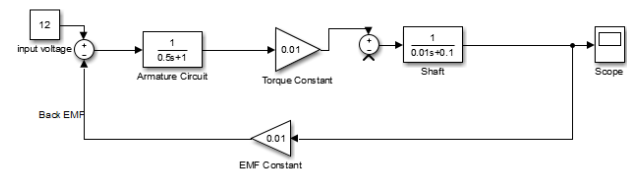


Figure 3. Simulink model of DC motor without load

After connecting load, i.e., represented by Load Torque, simulated model looks like

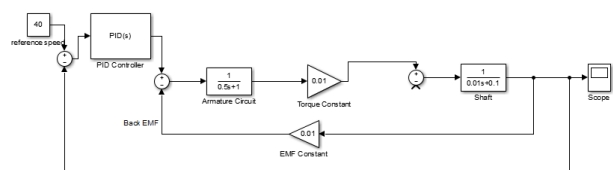


Figure 4. Simulink model of DC motor with load

III. SYSTEM DESCRIPTION AND WORKING

In this section, designed controller is tested to investigate the angular velocity of DC motor under different loads. The designed controller contains control loops for speed and armature current. A Proportional-Integral, Derivative (PID) controller is coupled with DC motor to obtain required speed of motor and to sustain its speed under different loading conditions.

Here, the velocity measured at shaft of the DC motor is known as actual speed. This real speed of the motor is sensed by the feedback path of the controller and is compared with its original preset speed. In result, an error is generated which is the difference of reference and actual speeds. The PID controller takes that error as input, determines the required motor's armature current and supplies that to the motor accordingly. In this study, the controller has been tested in three different cases. In the first one, actual speed is compared with reference speed when motor is at no load condition, fig 5.

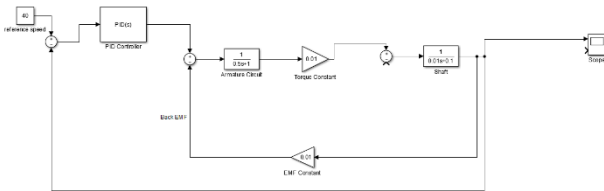


Figure 5. Simulink model of DC motor with PID controller without load

In second case, actual speed is compared with reference speed when some light load is connected with DC motor, that is shown in Fig. 6.

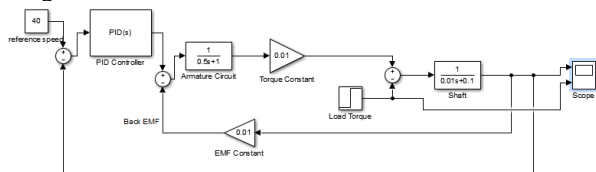


Figure 6. Simulink model of DC motor with PID controller with load

In last case, same comparison is done but a heavy load is connected with DC motor, shown in Fig. 6.

The PID controller is a very common but effective approach for designing control systems. It is a controller that take numerous values of error into consideration. In addition, it is also a method of control loop feedback. The effect of all its controllers (P, I and D) are summed together to produce an optimal output. However, tuning of PID controller is of prime concern. There are various methods for tuning PID controller such as Ziegler-Nichols Method, manual tuning method and MATLAB tuning method. PID controller is a suitable choice for control systems because of its zero steady state error, quick response, and maximum stability free of overshoots. In this paper, MATLAB tuning method has been used first to obtain the values of all gains of PID controller to reduce the high settling time of the DC motor. In the next step, manual tuning is accomplished to compensate the overshoot and steady state error. The output of

the PID controller is input voltage to the DC motor. Ultimately, the gain values of PID controller after tuning are:

| Kp | | Ki | Kd |
|----|--|-----|----|
| 80 | | 120 | 10 |

IV. RESULTS AND DISCUSSION

Results of the proposed system model simulation, that has been developed in Simulink/MATLAB, have been discussed here in this segment. All the three cases under which the proposed controller has been tested, are presented here step by step. Initially, the model of DC motor, without controller, was developed in Simulink/MATLAB on the basis of its block diagram. Its output response is investigated at the input voltage of 12 v in Fig.7.

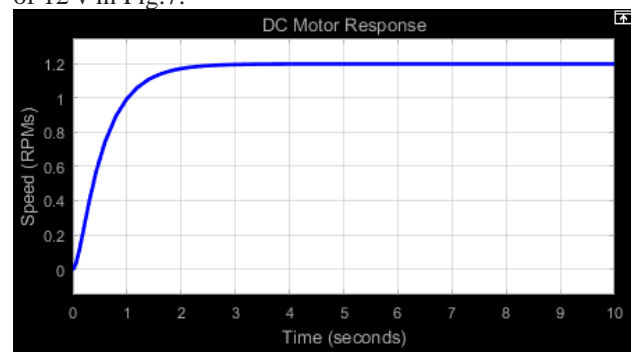


Figure 7. output response (angular speed) of DC motor without PID controller

Here, it can be seen that settling time is 2 seconds while the output reaches at 1.2, i.e., 10% of its input. Therefore, the steady state error here is 90%.

Now, the controller has been coupled with the system and its response at case 1 (without load) can be seen in Fig.8.

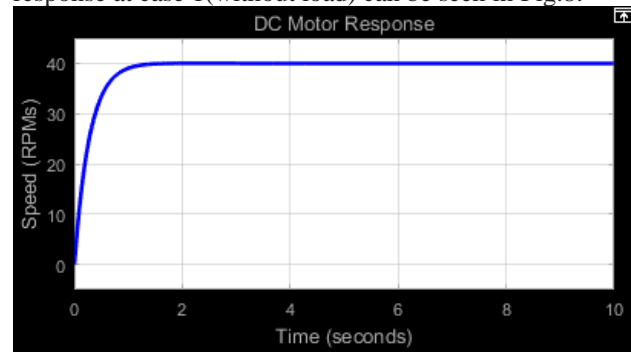


Figure 8. output response (angular speed) of DC motor with PID controller without load

The reference speed has been set to 40 RPMs. It is clear in Fig.8. that the settling time has been significantly reduced to 0.5s, while the actual speed of the DC motor is same as its reference speed. Thus, controller has mitigated the steady state error of DC motor.

In the next step, response of controlled DC motor can be seen on small load, as shown in Fig.9.

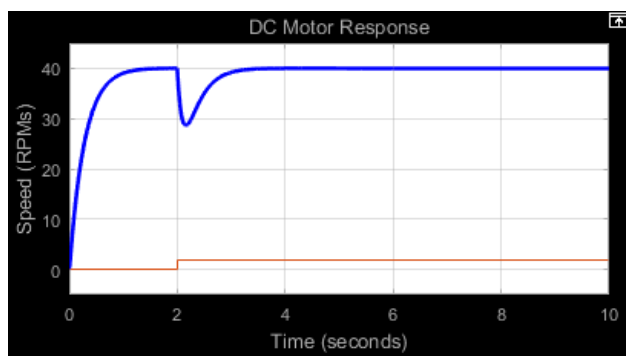


Figure 9. output response (angular speed) of DC motor with PID controller with small load

Load torque is supplied to the shaft of the motor after two seconds and consequently, speed of the DC motor has dropped from its reference value. A disturbance can be observed in the graph that represents that load has been connected to the DC motor. After a span of 0.5s i.e., the settling time of controller too, speed of the DC motor returns to its preset reference value that shows the proper working of controller.

Finally, in the third case, we have connected a heavy load and observed its impact on the speed of DC motor that can be seen in Fig.10.

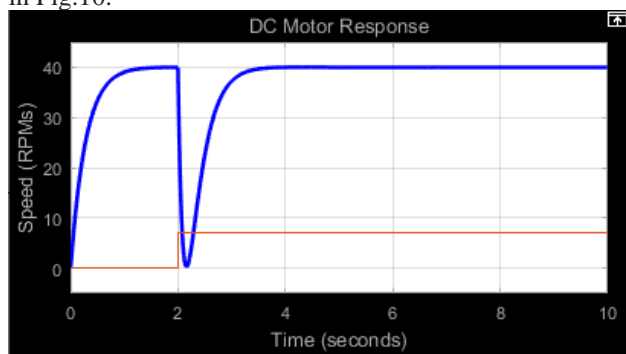


Figure 10. output response (angular speed) of DC motor with PID controller with heavy load

Since the magnitude of current load is higher, the speed of DC motor drops to zero. After 0.5s, controller has returned the speed to its original reference value, i.e., 40 RPMs. Thus, controller is working properly even under the effect of heavy load.

V. CONCLUSION

This study proposed the DC motor speed control method employing PID controller under shifting load. DC motor model was developed with the PID controller to control its speed. In this project, speed is maintained constant while variable load torques. The proposed system was successfully tested. From the above discussed results, it is obvious that PID controller is an effective approach to control the speed of DC motor under various loads. Moreover, successful outcomes that are mentioned above, proved that the PID controller is suitable in controlling and sustaining the speed of DC motor.

VI. REFERENCES

- [1] Sabir, M. M., & Khan, J. A. (2014). Optimal design of PID controller for the speed control of DC motor by using metaheuristic techniques. *Advances in artificial neural systems*, 2014.
- [2] Hameed, W. I., & Mohamad, K. A. (2012). Speed control of separately excited dc motor using fuzzy neural model reference controller. *International Journal of Instrumentation and Control Systems (IJICS)*, 2(4), 27-39.
- [3] Venu, K., Rushikesh, C., & Rajasekhar, V. (2013). Design and analysis of DC motor with PID controller—a state space approach. *Transac Electr Electron Eng (ITSI-TEEE)*, 1, 11-14.
- [4] Nawi, B., Sulaini, B., Mohd, Z. A., Shamsul, A. Z., & Zairi, I. R. (2015). PID voltage control for DC motor using MATLAB Simulink and Arduino microcontroller. *Journal of Applied Environmental and Biological Sciences*, 5(9), 166-173.
- [5] Patil, M. M. (2014). Modelling and simulation of dc drive using PI and PID controller. *International journal of innovative research in electrical, electronics, instrumentation, and control engineering*, 2(12).
- [6] Aborisade, D. O. (2014). DC motor with load coupled by gears speed control using modified Ziegler-Nichols based PID tunings. *Control Theory and Informatics*, 4(5).
- [7] Elsgogy, W. M., Fkirin, M. A., & Hassan, M. M. (2013, May). Speed control of DC motor using PID controller based on artificial intelligence techniques. In *2013 International Conference on Control, Decision and Information Technologies (CoDIT)* (pp. 196-201). IEEE.
- [8] Rout, M. K., Sain, D., Swain, S. K., & Mishra, S. K. (2016, March). PID controller design for cruise control system using genetic algorithm. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* (pp. 4170-4174). IEEE.
- [9] Loucif, F. (2005). DC motor speed control using PID controller. *제어로봇시스템학회: 학술대회논문집*, 2557-2561.
- [10] Singh, A. P., Narayan, U., & Verma, A. (2013). Speed control of DC motor using PID controller based on matlab. *Innovative Systems Design and Engineering*, 4(6), 22-28.