Speed and Position Control of a DC Motor

Rami Awar, Yahya Khaled Al Ali, Youssef Al Jrab
American University of Beirut
Beirut, Lebanon
rba13@mail.aub.edu, yka09@mail.aub.edu, yya04@mail.aub.edu

***Abstract*—This paper will present multiple controller designs for both speed and position control of a direct-current motor, while abiding to requirements set in order to control an engraving laser. For both speed and position control, proportional, PID, lag-lead, and full-state feedback controllers are investigated, and experimental results from a Quanser Qube Servo that nearly matches the mathematical model are compared with simulation results from that model. The PID controller design in both systems meets the requirements with the least error and is thus chosen as the final design.**

I. Introduction

The Direct Current motor is one of the most widely used motors in industries worldwide. Even though they are more costly to maintain over induction motors, they are still preferred due to excellent speed control characteristics[NT09]. In addition to this, speed and position controllers are widely available for DC motors of all sizes and power, from toy motors to industrial grade motors. But even with all these favorable qualities, simply applying constant power to a DC motor will not maintain the desired speed[Awa10]. Disturbances taking place at any point inside the motor, whether a varying load torque on the shaft or electrical noise on the input, will guarantee that the motor speed drifts away from that which is desired. There exists several types of DC motors such as a stepper motor, a permanent magnet DC motor, PCB motors, ... Each has its advantages and disadvantages over the others, but for the purpose of this laser engraver application, a brushed DC motor will be used. DC motor shaft rotation speed and position (angle) are mainly measured using encoders, which could be of many types (optical, rotary, magnetic, ...). Then encoders issue a stream of pulses with variable frequency according to the motor speed, hence have a unit of Pulses-Per-Revolution (PPR). [ySEAH03] The encoder in use in the experimental section of this paper is an optical encoder mounted on the shaft of the brushed motor.

II. Literature Review

The main purpose of this paper is to investigate the fitness of different controllers meeting set requirements for DC motor speed and position control. The most widespread DC motor driving method is PWM or Pulse Width Modulation, which works by alternating the applied voltage from high to low and vice-versa. This delivery of energy through a succession of pulses allows for speed control by varying the duty cycle of this alternating signal[SSU16]. The fast switching is seen by the motor as a $V_{rms} < V_{supply}$, which allows for voltage variation digitally, without the use of varying resistances. Most of the literature focuses on controlling DC motors with simple feedback loops, using tunable compensators, and modern research investigates advanced tuning techniques relating to artificial intelligence and optimization algorithms. Other interesting control techniques are independent of sensors (i.e. encoders of all kinds), and were based on estimating the required states.

Thomas et al. worked on designing a position controller of a DC motor by tuning a PID controller using a genetic algorithm. The DC motor model is considered a third order model, thus no approximations have been made to have more DOFs. Aside from tuning the PID controller using a genetic algorithm, the paper also discussed designing the controller using the Ziegler-Nichols method. The second method has shown to be inferior to the genetic algorithm tuner, which is expected since ZN provides a first guess of the parameters needed, and the stochastic nature of genetic algorithms works on minimizing the error between the actual response and desired response. Another advantage of the genetic algorithm tuning technique is that is could be applied to higher order systems as well, making it a modular multi-purpose algorithm.

Praesomboon et al. [SP09] proposed a sensorless DC motor speed controller using a Kalman filter. With the end goal being a system output of estimated speed, this was achieved by designing the Kalman filter to reject noise. The error between the estimated speed and reference speed is then fed back to the system and driven to zero by a linear amplifier (proportional gain). This method of senseless control based on estimating states seems to have been popular since the 1970s.

In 1978, Rajaram et al. wrote about a new approach to sensing a DC motor speed without the use of any additional circuitry, cutting costs and increasing compactness without affecting weight. They aimed to avoid traditional sensing methods that involve external additional hardware along with mechanical coupling by basing their measurement on the back EMF of the motor, in addition to a sensing resistor in series with the armature and some simple circuitry. That results in an amplifier output that is proportional to the speed of the motor, and thus can be used as a sensor measurement [Raj78].

Another intelligent DC motor position controller designed by Ohishi et al. is a load insensitive controller, which works by estimating the load torque [ea02]. By estimating this torque the controller can adjust the torque needed to drive the system to the required position without any steady state error. This is similar to having a variable parameter mathematical model, but instead of the controller being affected by each parameter change, a state estimation is invoked to take away that complexity and overhead in processing.

III. Model

The DC Motor dynamics used are based on the motor model in Fig. [1]. The derivation

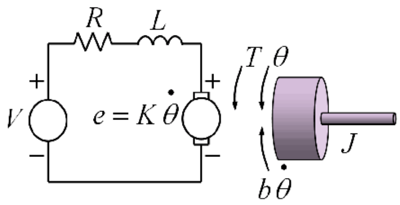

Figure 1. Circuit model of a DC Motor.

The equations for the internal generated voltage and the induced torque developed by the DC Motor are given by: [J.C12]

$$T_{ind} = K_\phi I_A$$

$$e_a = K_\phi \omega_m$$

Applying Kirchoff's law and Newton's law to system in Fig.[1]. The following is obtained:

$$J\frac{d^2\theta}{dt^2} + b\frac{d\theta}{dt} = Ki$$

$$L\frac{di}{dt} + Ri = V - K\frac{d\theta}{dt}$$

Applying the Laplace transform, the transfer function $G_{speed}(s)$ is obtained, assuming $K_b = K_m$.

$$G_{speed}(s) = \frac{K}{JLs^2 + (JR + Lb)s + (Rb + K^2)}$$

Introducing an integrator gives the position θ as an output, and the transfer function becomes $G_{position}(s)$, assuming $K_b = K_m$.

$$G_{position}(s) = \frac{K}{s[JLs^2 + (JR + Lb)s + (Rb + K^2)]}$$

The motor block becomes that in Fig.[2], where K_m, K_b, J, b, R_a, L_a are the motor parameters provided.

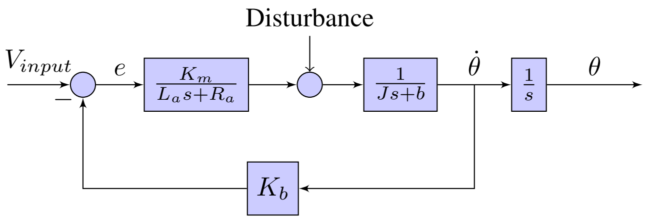

Figure 2. Block diagram for the DC motor model.

The State Space representation of the speed system has been derived to show two states that are expressed in terms of the given parameters:

$$\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} \frac{-JR-Lb}{JL} & 1 \\ \frac{-Rb-K^2}{JL} & 0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K}{JL} \end{bmatrix} * u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

IV. Analysis

The speed control system is a second order electromechanical system having two poles at $s_1 = -7399$ and $s_2 = -16.2$. It is seen that the pole s_2, which corresponds to the mechanical system dominates the electro-mechanical system's response because of a difference by a factor of 456 from the pole s_1, which corresponds to the electrical system.Henceforth, since the factor of difference is greater than five, the system could be approximated by a first order mechanical system, since the mechanical response and second order response are exactly alike as seen in the following figure.

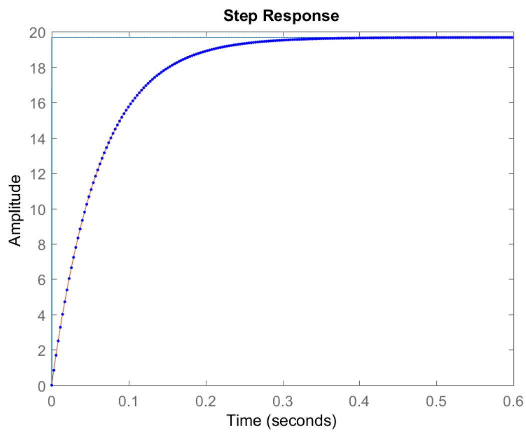

Figure 3. Comparison between electrical system, mechanical system, and electromechanical system.

The red curve represents the mechanical system, the light blue curve represents the electrical system and the blue dotted curve represents the second order electromechanical system.
In this figure, it is seen that the mechanical and electromechanical systems overlap however, **the second order system is used in both analysis and design since it is intended to have similar responses to the responses of the Quanser Qube that are achieved experimentally**.

In the first analysis iteration, it is found that the damping friction coefficient is unrealistic for several reasons listed below:

- $10N \cdot m$ are exerted by the motor shaft at a speed of 1rd/s which is similar to hanging 10kg at the end of a 1 meter long rod, and spinning it at 1rd/s which is 60 degrees per second. This torque obviously cannot be achieved by a motor running at 10-24V with around 2A of current and without a gearbox.

- Quanser Qube Servo parameters are similar to the provided parameters in all aspects but the viscous friction coefficient, which differs by 6 orders of magnitude.
- The proportional gain, which translates into voltage required to drive the motor, needed to track a step response (i.e. 1rd/s) is over 7000. A 7000V input to the motor is impossible knowing that it operates on 10-24V.

A more reasonable value has been derived by referring to the Quanser Qube Servo parameters posted on their website, by using the formula

$$T_{nom} = b \cdot \omega_{nom}$$

. It is found that $b = 6.88 * 10^{-5} N \cdot m \cdot s/rad$ and the derived value is used throughout the analysis, simulation, and experimentation.

A. Speed Control System

1) Time Domain Analysis: The time domain analysis of the speed control system was done via MATLAB software, `ltiview` and `stepinfo` commands. The following figure shows both open and closed loop response of the speed system in response to a step input.

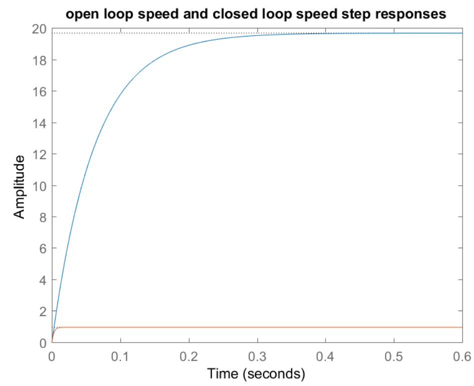

Figure 4. Comparison between open loop speed response and closed loop speed response.

It is seen that the closed loop system is better to meet the speed requirements since it is near 1 at steady state whereas the open loop is near 20 at steady state.
For open loop, the peak is 19.7 so it is an undesired response. However for the closed loop, the overshoot is 0, the steady state error is 4.8%, and the settling time is 0.0113 sec.

2) Frequency Domain Analysis: It was verified upon simulation via MATLAB that the system's speed is stable. This is verified with infinite gain margin and a safe phase margin in both the open loop and closed loop systems. The bandwidth of the closed loop system exceeds that of the open loop speed system by a factor of 21.6; which clarifies that the closed loop response is of a better response.

Speed open loop: Phase margin: 90.4648 degrees
Gain Margin: Inf dB
Bandwidth: 16.1278 rad/s

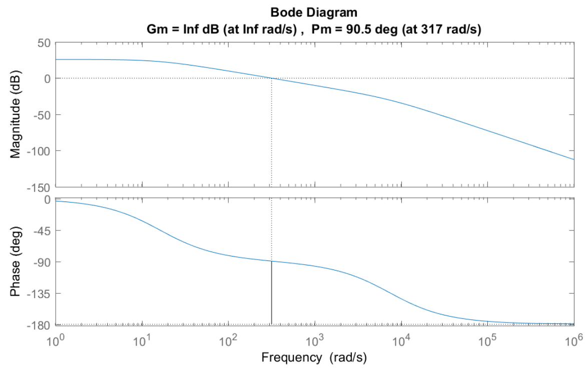

Figure 5. Bode plot of the open loop speed system.

Speed closed loop: Phase margin: Inf dB
Gain Margin: Inf
Bandwidth: 348.210 rad/s

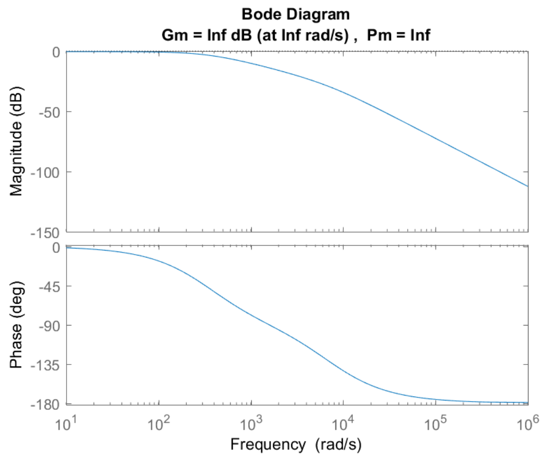

Figure 6. Bode plot of the closed loop speed system.

3) Controllability and Observability: The system is composed of two controllable and observable states which means that the speed system is both completely controllable and completely observable .Therefore, all states in the system can be modified to reach a certain desired response, and can all be sensed, measured, using sensors to know the estimators for the input of a system.

Controllability matrix of the speed system:

$$\begin{bmatrix} 1 & -7.4176e^3 \\ 0 & 1 \end{bmatrix}$$

Uncontrollable states = 0

Observability matrix of the speed system:

$$\begin{bmatrix} 0 & 2.3529e^6 \\ 2.3529e^6 & 0 \end{bmatrix}$$

Unobservable states = 0

B. Position Control System

1) Time Domain Analysis: The time domain analysis of the position control system was done via MATLAB software, LTIVIEW and stepinfo commands. The following figure shows both open and closed loop response of the speed system in response to a step input.

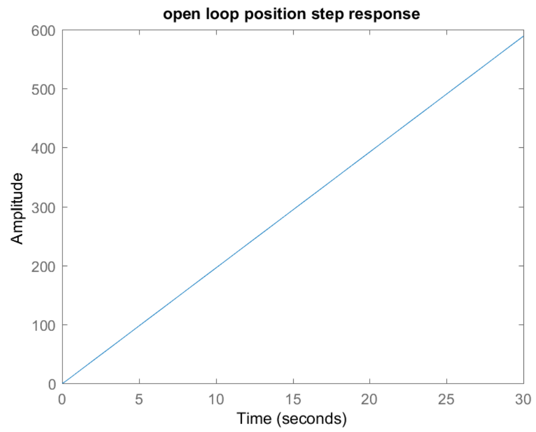

Figure 7. Open loop position step response.

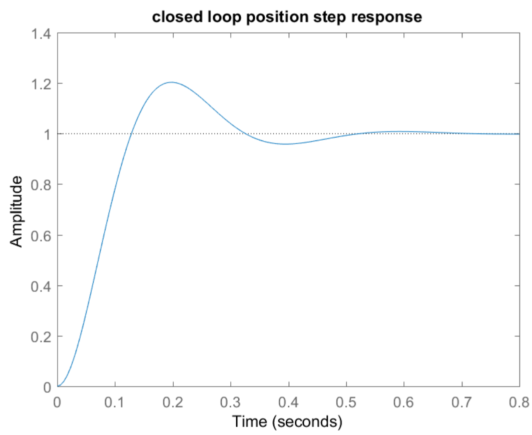

Figure 8. Closed loop position step response.

Applying the step for both open loop and closed loop position systems, it is seen that the open loop system increases as a ramp nonstop, which seems logical since an integrator was added which aggravated stability, so it is definitely not desired as a system response. However, the feedback in the closed loop system made the response settle to 1 instead of increasing infinitely which is desired. For the closed loop, the maximum overshoot is 20%, the settling time is 0.468 seconds, and a steady state error of 0.

2) Frequency Domain Analysis: It was verified upon simulation via Matlab that the system's speed is stable. This is verified with 51.5dB gain margin and 48 degrees phase margin in the open loop and closed loop systems. The bandwidth of the closed loop system exceeds that of the open loop speed system by a factor of 1.3; which clarifies that the closed loop response is of a better response.

Position open loop: Phase margin: 47.8028 degrees
Gain Margin: 51.5259 dB
Bandwidth: 18.4 rad/s

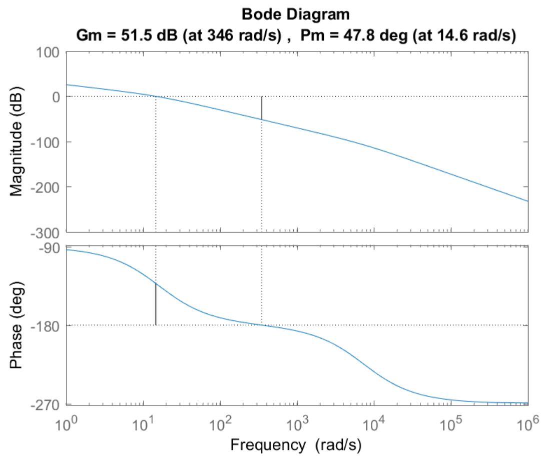

Figure 9. Bode Plot of the open loop position system.

Position closed loop: Phase margin: 49.3 Degrees
Gain Margin: 51.5 dB
Bandwidth: 23.5968 rad/s

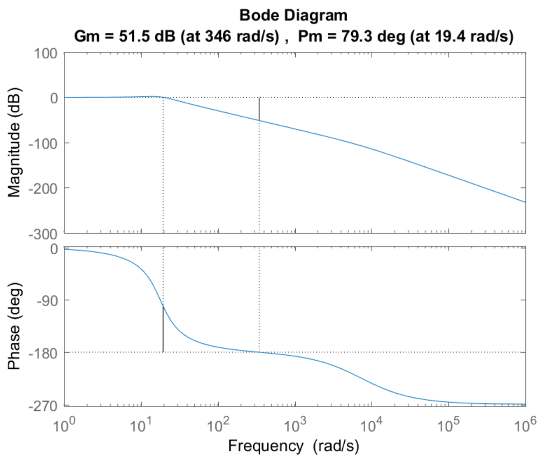

Figure 10. Bode Plot of the closed loop position system.

3) Controllability and Observability: The system is composed of three controllable and observable states which means that the position system is both completely controllable and completely observable .Therefore, all states in the system can be modified to reach a certain desired response, and can all be sensed, measured, using sensors to know the estimators for the input of a system. This statement, is clarified in both the controllability and observability matrices respectively. Controllability matrix of the position system:

$$\begin{bmatrix} 1 & -7.4176e^3 & 5.4902e^7 \\ 0 & 1 & -7.4176e^3 \\ 0 & 0 & 1 \end{bmatrix}$$

Uncontrollable states = 0

Observability matrix of the position system:

$$\begin{bmatrix} 0 & 0 & 2.3529e^6 \\ 0 & 2.3529e^6 & 0 \\ 2.529e^6 & 0 & 0 \end{bmatrix}$$

Unobservable states = 0

V. Design

A. Speed Controller

In speed control the transfer function $G_{speed}(s)$ is used in designing the speed controller. The four controllers that will be designed are a proportional controller, a proportional-integral-derivative controller, a lead-lag controller using root-locus, and a full state feedback controller. The requirements to be met are the following:

- $\pm 100rpm$ forces the motor speed to stay within this range of the desired input speed.

From this requirement, it can be deduced that the maximum overshoot is $100rpm/\omega_{max} = 100rpm/2000rpm = 0.05$ which translates to 5%.

1) Proportional Controller:

From these requirements it is found that $\zeta = 0.4765$ from the 5% maximum overshoot criterion. A 0% overshoot corresponds to $\zeta = 1.0$ and so the accepted range of zeta would be $0.4765 < \zeta < 1$. Plotting the root locus and choosing a gain such that the chosen roots meet the criteria gives a gain $C = 8.34$. A comparison of the proportional controller step response and the closed loop step response is found in Fig.[11].

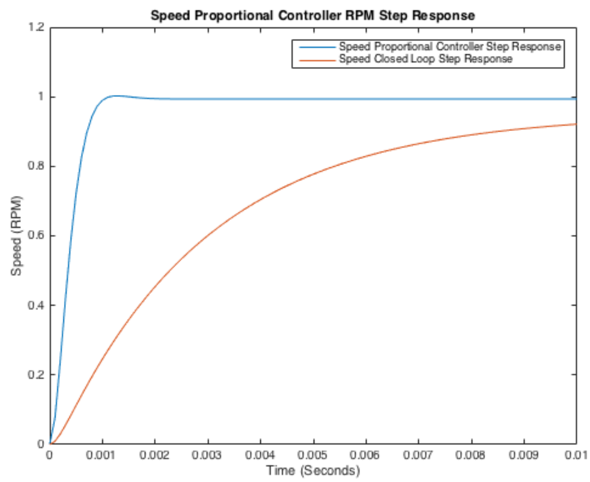

Figure 11. Comparing Speed Closed Loop and Proportional Step Responses

It is seen that the maximum overshoot is 0.3% which lies well within the required range.

2) PID Controller:

In designing the PID controller, MATLAB's `pidTuner` command is used. This allows to customize the transient response speed and controller robustness to get the desired response much quicker than manual tuning using the Ziegler-Nichols method. A comparison of the closed loop step response and the proportional-integral-derivative controller step response is shown in Fig.[12]. A PID controller is chosen over a PI controller due to the quicker transient response. A PI controller will have less of an overshoot at the cost of lost speed. Given that an overshoot of 3% lies well within the allowed range (100rpm), the focus was chosen to be on speeding up the response.

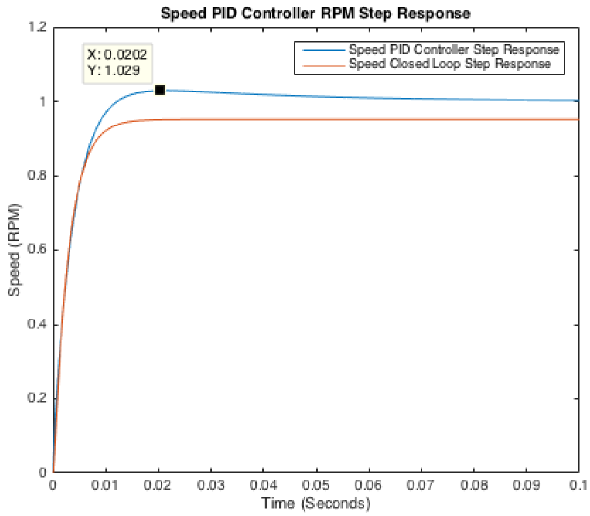

Figure 12. Comparing Speed Closed Loop and PID Step Responses

3) Compensator Design with Root Locus:

It is seen that by simply adjusting the gain (as done in the preceding proportional controller), the requirements can be met and thus no lag or lead compensator is necessary. To further prove this point consider the following:
The transient response of the system settles after 11.3 ms at a value of 0.952 rad/s when it responds to a unit step response, which means that the system response is quick enough. Hence, the transient response does not need modification. This is verified by the two real closed loop poles that yield an angle deficiency of zero degrees. Therefore, any pole-zero placement would not have any effect on the transient response. Moreover, because it lacks an integrator, it is a type 0 system. This means that for a given step input, the response will yields a steady state error that decreases with an increase in K, proportional gain. Therefore, with the current system, a lag compensator would not decrease the steady state error because of the absence of complex closed loop poles. In addition to this, the response has two closed loop real poles that are large enough to cancel the effect of the near origin pole-zero. Hence, the overall angle that would result is zero which means that we don't have a lagging compensator. With such criteria, the design of a lag-lead system would not give the desired response. Hence, the better solution is to use a proportional gain that decreases the system's steady state error, or a PID compensator that drives this error to zero.

4) Full state feedback controller: In designing the full state feedback controller, it is assumed that all states of this system are measurable and available for feedback. In order to design the full state feedback controller, it is crucial at first to check if the system is fully state controllable. As checked before, the speed system is fully state controllable. The speed system is known to have the following state space representation:

$$\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} -7415.2 & 1 \\ -119614.1 & 0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 2352941.176 \end{bmatrix} * u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

The state controller will be placed such that to make the control signal u=-[K]x+k_1r. The gain matrix [K] will be decided with respect to the location of the desired closed loop poles of the system. The specifications for the speed system is to track the input without exceeding ± 100 rpm, so the desired closed loop pole locations will be chosen to be $s_1 = -189 + j169$ and $s_2 = -189 - j169$ in order to meet these specifications. Using the function place in Matlab we get the gain matrix to be:

$$[K] = \begin{bmatrix} 22.16 & -0.003 \end{bmatrix}$$

$$A - BK = \begin{bmatrix} -7415.2 & 1 \\ -5.225e^7 & 7038.2 \end{bmatrix}$$

Using u= -[K]x + k_1r and x'= Ax + Bu so x'= (A-BK)x + Bk_1r which will become:

$$\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} -7415.2 & 1 \\ -5.225e^7 & 7038.2 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 5.214e^7 \end{bmatrix} * r$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

B. Position Controller

In designing the position controller, the transfer function $G_{position}(s)$ is used. The four controllers that will be designed are a proportional controller, a proportional-integral-derivative controller, a lead-lag controller using root-locus, and a full state feedback controller. The requirements to be met are the following:

- Maximum overshoot and steady state error of 1° in order for the laser engraving to be legible
- Ability to travel 50° in 0.5 seconds

The first requirement is clear, while the second must be met with simulation/implementation and according adjustments since the relationship between the degrees turned and the time taken by the motor to settle is not linear and is slightly complex to derive.

1) Proportional Controller: From these requirements, it is found that $\zeta = 0.7796$ if we consider the 1° relative to 50° which would be an acceptable choice. A safer choice would be to consider the 1° relative to 360° since any rotation greater than this limit can be broken down into multiple rotations with a larger modulo. This gives us $0.7796 < \zeta < 0.8831$, which would translate into an area to place the poles in on the root locus. Plotting the root locus using MATLAB and using the command `sgrid`, then choosing any point within the area formed by the two lines forming angles 27.98° and 38.83° with the real axis, we get Fig.[13]. Decreasing the obtained gain from 0.344 to 0.3 moves us towards the safer choice of $\zeta = 0.8831$.

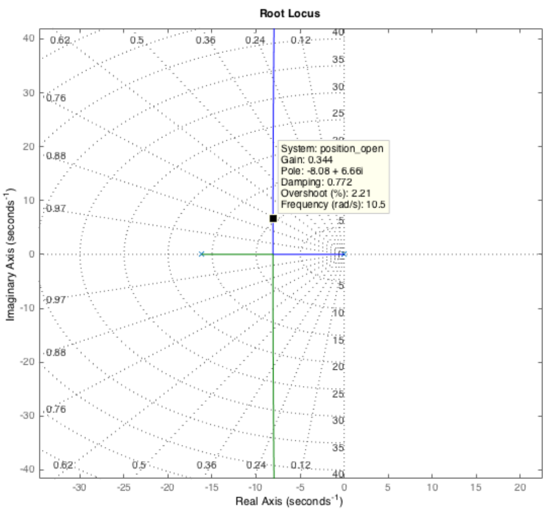

Figure 13. Root Locus of Position Control System

The choice of $K_p = 0.3$ produces the response to a 50° step input in Fig.[14]. We see that the maximum overshoot is less than 0.5° and the response is within 2% of its final value in less than 0.5 seconds, thus meeting our requirements.

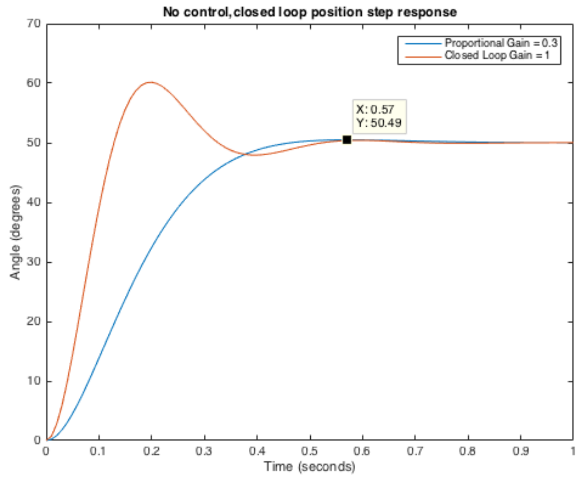

Figure 14. 50° Step Response of Position Proportional Controller

2) PID Controller: The PID controller tuning was done using MATLAB's `pidTuner` command. The obtained results meet the 1% overshoot requirement, as well as the 50° in 0.5 seconds requirement. Fig.[15] shows the step response of this new system along with the derived $K_p, K_i, and K_d$ values.

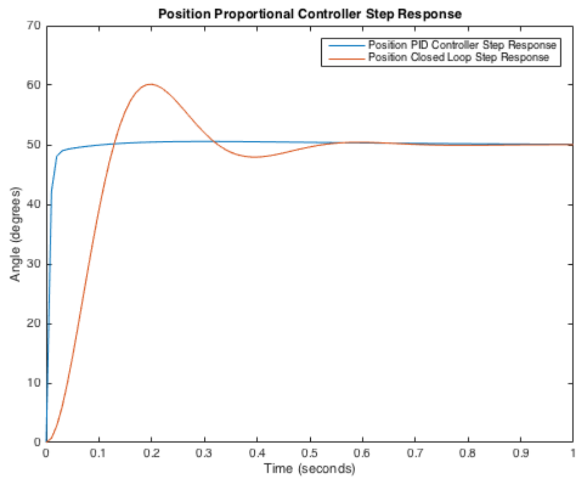

Figure 15. Position PID Controller 45° Step Response w/ $K_p = 6.7498, K_i = 15.8366, K_d = 0.6035$

3) Compensator designed via root locus: Upon simulation via Matlab, the response to a unit step has 0.11 s rise time and a steady state error of zero. A lead compensator was designed with the intent of changing the rise time to speed up the transient response. The closed loop poles of the original system initially were at:

$$s = -8.06 + j15.9$$

$$s = -8.06 - j15.9$$

$$s = -7401.53$$

Because the complex conjugate poles are complex conjugates, therefore, the zeta and the natural frequency were found as:

$$\zeta = 0.452$$

$$\omega_n = 17.83 rad/s$$

Since, the system's zeta is not in the desired range [0.7796-0.8831] that corresponds to the intended 50 degrees and a safe margin of 360 degrees respectively; The average of both was selected, 0.83. Therefore the new complex conjugate poles change to:

$$s = -14.82 + j9.9$$

$$s = -14.82 - j9.9$$

The compensator was designed using the angle deficiency method with an angle of deficiency of 64.26 degrees that relies on one lead compensator because the angle is strictly less than 65 degrees. This method yields the chosen zero and pole of the compensator as follows:

$$z = -11.85; p = -25.8$$

Moreover, the gain was calculated to be $K = 47.03e6$ which is unrealistic and drives the response to instability as shown in Fig.[16].

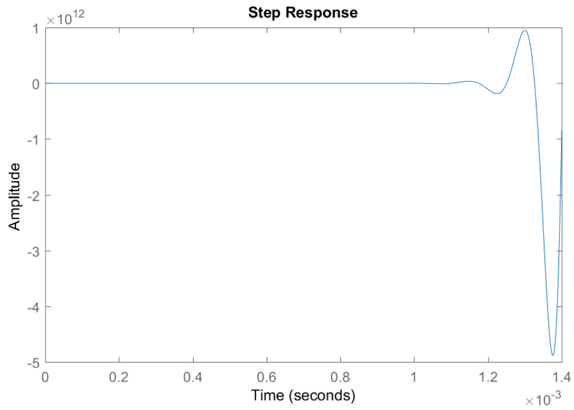

Figure 16. Position system response to a lead compensator.

Furthermore, because the lag system tends to increase the maximum overshoot beyond the desired limitation, and yield a long tail, the lag compensator was not used.

Hence, with these two types of compensator designs yielding no improvement to the system's response, the lag-lead compensator design would not be an improvement either. As a result, no lagor lead compensator is recommended.

4) Full state feedback controller: In designing the full state feedback controller, it is assumed that all states of this system are measurable and available for feedback. In order to design the full state feedback controller, it is crucial at first to check if the system is fully state controllable. As checked before, the position system is fully state controllable. The position system is known to have the following state space representation:

$$\begin{bmatrix} x_1' \\ x_2' \\ x_3' \end{bmatrix} = \begin{bmatrix} -214370 & 1 & 0 \\ -345684.8 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 6.8e^6 \end{bmatrix} * u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

The state controller will be placed such that to make the control signal u=-[K]x+k_1r. The gain matrix [K] will be decided with respect to the location of the desired closed loop poles of the system. The specifications for the position system is to track the input of 50° in 0.5 sec without exceeding $\pm 1^\circ$, so the desired closed loop pole locations will be chosen to be $s_1 = -10 + j6.7$, $s_2 = -10 - j6.7$, and $s_3 = -100$. A second order approximation was used using the fact that the complex conjugate poles are dominant with a factor of 10 difference from the third pole, to meet the required specifications. Using the function place in Matlab we get the gain matrix to be:

$$[K] = \begin{bmatrix} -1.44e^9 & 6754 & 3.151e^-3 \end{bmatrix}$$

$$A - BK = \begin{bmatrix} -2.14370e^5 & 1 & 0 \\ -3.456e^5 & 0 & 1 \\ 9.84e^15 & 4.59e^10 & 2.14e^5 \end{bmatrix}$$

Using u= -[K]x + k_1r and x'= Ax + Bu so x'= (A-BK)x + Bk_1r which will become:

$$\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} -2.14370e^5 & 1 & 0 \\ -3.456e^5 & 0 & 1 \\ 9.84e^15 & 4.59e^10 & 2.14e^5 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -9.845e \end{bmatrix}$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

According to those results, the full state feedback controller shows non-linearity because of the high gain values. This issue, is caused by the large gain the control signals carry that require larger, more expensive, actuators.

VI. Final Design

Relying on the experimental validation, the PID controller is chosen to be the final controller design. For both speed and position, the PID controller showed rejection to any time of noise applied while meeting the specified criteria for both speed and position control systems. The PID compensated system, is distinguished by a fast transient with a small settling time while maintaining the specified maximum overshoot. While on the contrary, any other controller as P, PI, full state feedback controllers, reach the desired specifications but the difference is best shown in disturbance rejection. The P controller, minimizes the error to zero if the proportional gain was large enough but doesn't respond fast to any disturbance. The PI controller rejects quite well, but compared to the PID, the PID controller is better in increasing stiffness against noise. The lead compensator showed nonlinearity that corresponds to a large gain in position control while on the other hand, it was not used in speed control because, the system consists of two real roots. The lag compensator increased overshoot beyond the limit and increased the position's response's settling time while on the other hand, the lag compensator was not applied in speed control because of the two real roots that broke the rule of thumb that relates to the angle criterion. Full state feedback controllers, responded well in speed control, reached specifications, and competed with the PID controller a lot. But, in position control, the full state feedback controller showed nonlinearity because, the system had massive gains within its state space representation. The thing that makes such system more expensive than a PID controller whose value is not as high as large actuators. Henceforth, PID controller met all desired specifications in both speed and position control systems.

A. Disturbance Rejection

The disturbance to output transfer function can be represented as follows:

$$G_{disturbance}(s) = \frac{Ls + R}{(Ls + R)(Js + b) + C * K + K^2}$$

where C stands for the compensator's transfer function.

From the transfer function, it can be noticed that if we have an infinite value in the denominator then, the output due to the disturbance would converge to zero in steady state. To meet such specification, it was found that the PID controller induces a pole at the origin in its transfer function which results in the desired infinite value in the denominator that drives the system's disturbance to zero. The difference between both PI and PID controllers was the time each controller's system would need to eliminate the error. PID controllers' settling time was less than that of the PI controller by a factor that increased as we increase the magnitude of the disturbance as MATLAB verifies such argument. The maximum value of the response to a unit step disturbance is:

- Speed Response: 7.57rd/s at t=0.00196s. This value decays exponentially until zero at 0.12s.
- Position Response : 394rd at t=0.201s. This value decays exponentially until zero at 2.5s.

These values are found by observing the step response to the disturbance transfer function in MATLAB Fig.[17].

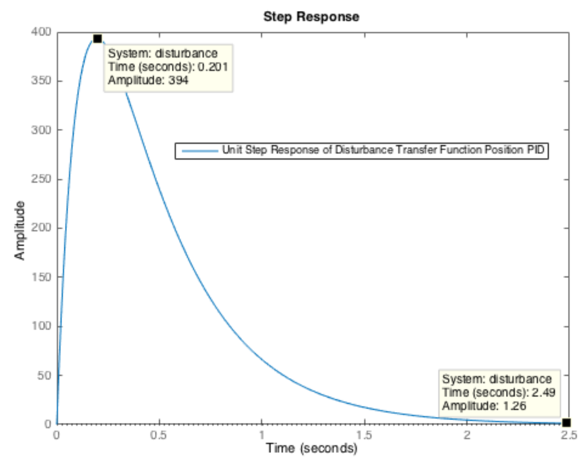

Figure 17. Position System Response to a Step Disturbance

B. Tracking

The PID corrected system lags the the ramp input by a magnitude of 0.03 and tends to correct its behavior as time increases to ideally track the input response. The PID controller was designed to surpass the given specifications, as the experimental validation proved. No oscillations are shown versus a ramp input. The system is sufficiently damped so that both control systems meet the maximum overshoot criteria that limits zeta to a certain range to assure a margin for safety purposes.

VII. Experimental Results

To simulate the numerical model, MATLAB's Simulink is used. Important things to note are the following:

- The process of converting the input from degrees to radians (in the case of position) and from RPM to radians per second (in the case of speed) is done by inserting a scaling gain block `RPM2RDS` right after the input, and another `RDS2RPM` right before the output. This way, the system isn't affected by what the input unit is, and runs according to its base SI units which are radians and radians per second. The conversion provides an interface for the user to input more familiar units, and observe more familiar outputs.

By trial and error, in an attempt to minimize the difference between the mathematical model and the physical DC motor setup, it is derived that the damping friction coefficient is b = 20.88e-5 $N{\cdot}m{\cdot}s/rad$. Using this damping friction coefficient allows the mathematical model responses to most precisely match the physical model responses. This is found after analyzing and designing the above controllers, so this change in b is not made. All results will be based on b=6.88e-5$N \cdot m \cdot s/rad$.

A. Speed Control

In speed control, it is important to note that even though the step inputs are speeds of 1 rd/s, the error requirement of 5% will result in this requirement being met for any speed from 0 rpm to 2000 rpm, which is the range of operation of the motor. Thus step responses of 1 rd/s meeting the 5% error criterion are sure to meeting the requirement for higher speeds; the unscaled step response is enough to portray the behavior of the motor for all operating speeds.

1) Proportional Controller:

Implementing the proportional controller with a gain $K_p = 8.381t$ results in a very oscillatory response caused by the variance of the damping friction coefficient b from its claimed value by Quanser. This is caused by overuse, lack of lubricant, wearing over time, etc. Decreasing the gain to $K_p = 3$ enhances the response by decreasing the oscillations, but the steady state error increases significantly. This shows that a gain of 8.381 is better than a gain of 3 at tracking the step input, but at the cost of a much larger overshoot and more sustained oscillations. This is shown in Fig.[18], and the requirement of $\pm 100rpm$ cannot be met using a proportional controller without changing the damping friction coefficient. In this case, the actual response differs greatly from that expected from the mathematical model. Perhaps motor lubrication can help meet the requirement.

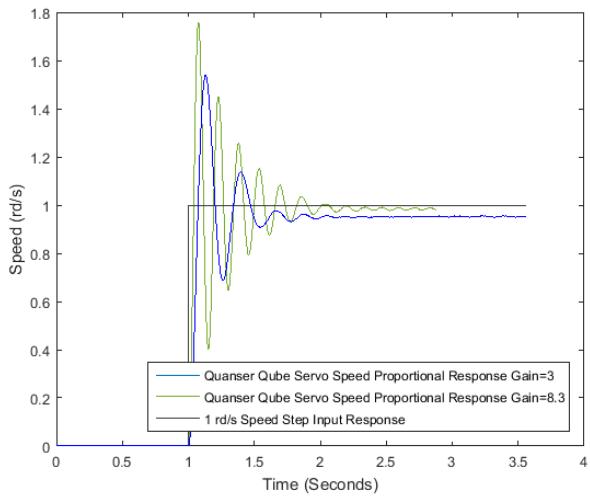

Figure 18. $1rd/s$ Step Response of Speed Proportional Controller w/ $K_p = 8.381$

2) PID Controller:

Using the parameters derived for the numerical model, the Quanser Qube produces very fast speed changes, which nearly damages the motor. This is caused by the K_d 0.0006, so it is increased to 0.5 to stop this destructive behavior. $K_p and K_i$ are the same as the ones derived in our numerical model. This results in the step response observed in Fig.[19]. This response doesn't meet the requirement either, but with some different tuning definitely could, at the cost of response time. As is seen in the figure, the motor takes around 0.5 seconds to settle, which is arguably large in speed controlling this motor compared to the responses acquired with MATLAB using the mathematical model.

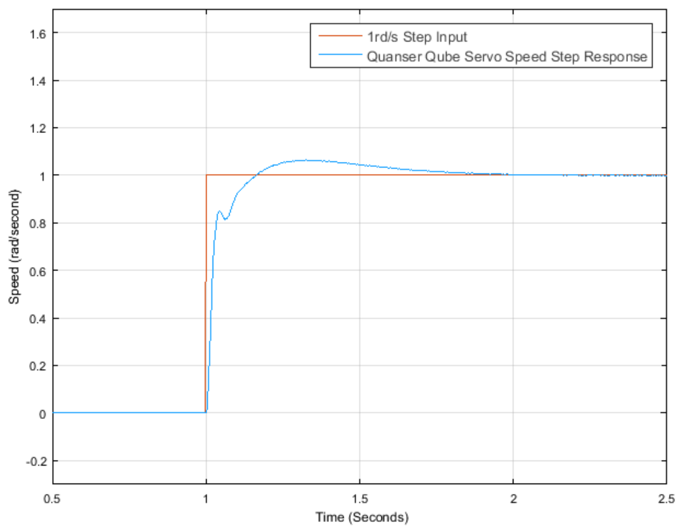

Figure 19. $1rd/s$ Step Response of Speed PID Controller w/ $K_p = 6, K_d = 0.5, K_i = 20$

The PID controller performs much better than the proportional controller, mainly due to the fact that it contains an integral term and can drive the error to zero. Perturbing the shaft with external objects, constant pressure is applied, and the motor increases its torque to drive its speed towards the desired speed. This error minimization is not possible using a simple gain factor.

B. Position Control

In position control, it is important to note that even if the responses were for a position of 1 degree, the maximum overshoot and steady state error of 1 degrees is still met, as well as the 50 degrees in 0.5 seconds criterion, since the error requirement is derived as a percentage. As long as the error is within this percentage the requirement is satisfied for any input less than 360 degrees.

1) Proportional Controller: Implementing the proportional controller with gain $K_p = 0.3$ on the Quanser Qube Servo gives us a significant and visible steady state error seen in Fig.[20].

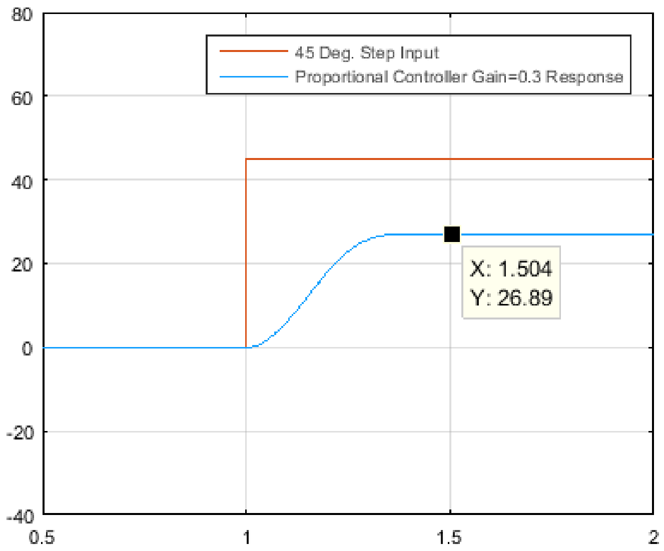

Figure 20. 45° Step Response of Position Proportional Controller w/ $K_p = 0.3$

This error is corrected by increasing the gain to 0.5 to account for the different viscous friction coefficient of the Quanser Qube Fig.[21]. A proportional controller cannot drive the steady state error to zero on its own, and this is seen in the experimental results.

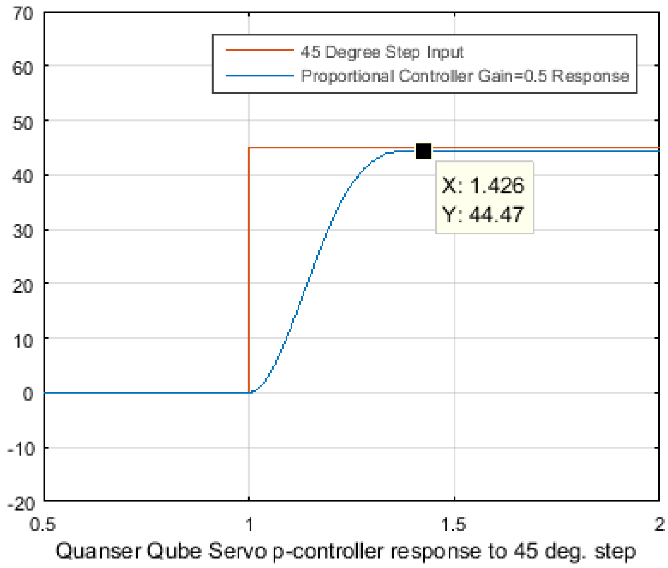

Figure 21. 45° Step Response of Position Proportional Controller w/ $K_p = 0.5$

2) PID Controller:

Using a PID controller, the controller exhibits much higher disturbance rejection and response speed, as well as less sustained oscillations. By perturbing the shaft with external objects, the motor does not fail to return to its preset position without any error or sustained oscillations. This is seen in Fig.[22].

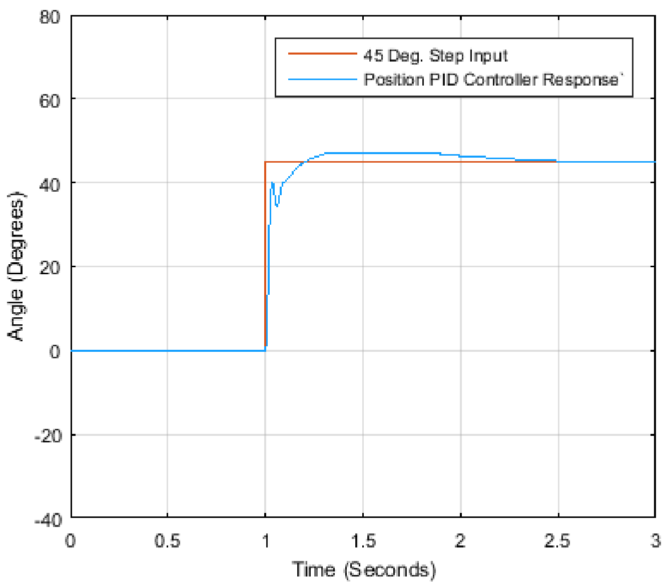

Figure 22. 45° Step Response of Position PID Controller w/ $K_p = 6.7498, K_i = 15.8366, K_d = 0.6035$

VIII. Conclusion

Six controller designs for both speed and position control systems were designed, simulated, experimented and evaluated against the given requirements that limit the transient and steady state responses to maximum overshoot and a desired settling time for phase, angle shift and speed control. The PID, PI, and full state feedback controllers competed against each other's characteristics, as they all drove the error to zero, met the required specifications, and met the desired responses upon simulation and experimentation. The PID controller was a better design because of its immediate response to any kind of disturbance, faster transient response even though by a difference of 0.05 seconds from its closest competitor,full state feedback, and relative cost compared to other controllers that rely on a big budget to satisfy the large gained control signals by means of large actuators.

NOMENCLATURE

$\dot{\theta}$	Angular Speed in $rd \cdot s^{-1}$
ω	Angular Speed in $rd \cdot s^{-1}$
ω_{nom}	Motor Nominal Angular Speed in $rd \cdot s^{-1}$
θ	Angle in rd
b	Motor Viscous Friction Coefficient in $N \cdot m \cdot s \cdot rd^{-1}$
J	Moment of Inertia in $kg \cdot m^2$
K_m, K_b	Motor Constant in $N \cdot m \cdot A^{-1}$
L_a	Armature Electric Inductance in H
R_a	Armature Electric Resistance in ω
T	Torque in $N \cdot m$
T_{nom}	Motor Nominal Torque in $N \cdot m$
V_{rms}	Root Mean Square Voltage in V

REFERENCES

[Awa10] Mohd Amir Fikri Bin Awang. *DC Motor Speed Controller*. BS dissertation, Universiti Malaysia Pahang, 2010.

[ea02] Kiyoshi Ohishi et al. Microprocessor-Controlled DC Motor for Load-Insensitive Position Servo System. *IEEE Transactions on Industrial Electronics*, 49(2):462 – 473, 2002.

[J.C12] Stephen J.Chapman. *Electric Machinery Fundamentals*. McGraw Hill, 1221 Avenue of the Americas, New York, NY 10020, 2012.

[NT09] Dr. P. Poongodi Neenu Thomas. Position Control of DC Motor using Genetic Algorithm Based PID Controller. *Proceedings of the World Congress on Engineering*, 2, 2009.

[Raj78] S. Rajaram. A New Method for Speed Measurement/Control of DC Motors. *IEEE Transactions on Instrumentation and Measurement*, 27(1):99 – 102, 1978.

[SP09] et al. S. Praesomboon, S. Athaphaisal. Sensorless speed control of DC servo motor using Kalman filter. *7th International Conference on Information, Communications and Signal Processing*, pages 1–5, 2009.

[SSU16] Dinesh Patil Sarita S. Umadi1. DC Motor Speed Control Using Microcontroller. *International Journal of Engineering and Techniques*, 2(6), 2016.

[ySEAH03] S. M. Uashi y. S. E. Ali, S. B. M. Noor and M. K Hassan. Microcontroller Performance for DC Motor Speed Control System. *National Power and Energy Conference Proceedings*, pages 104–109, 2003.