Can Brushed DC Motors Used in FRC Be Accurately Modeled as Linear Systems?

Drew Fisher

6/1/18

Abstract

The goal of this project is to test an FRC-class DC motor (mini-CIM) and characterize its output speed's relation to the applied voltage. This is particularly useful in modeling of dynamic systems, and allowing control loops to compensate less for movements, by applying a feedforward derived from the model. While the model most likely depends on the rotational inertia of the system, as well as the inertia of the system it is driving, the process used in this experiment can be replicated very quickly on other systems with relatively low software overhead.

Hypotheses

The null hypothesis, as the motor's specifications don't have the coefficients used to create a linear model can be derived from the published specification of the motor. It can be assumed that with 0V applied to the motor, the shaft will spin at 0 rpm. The motor datasheet says that at 12V applied, the motor should spin at 6200 rpm. This means that the null hypothesis should be that the *a* coefficient should be 517 rpm/V, and the *b* coefficient should be zero volts for an accurate linear model.

The proposed alternative hypothesis is that *a* is not 517 rpm/V, but that *b* is still 0 rpm, meaning that the motor has a different constant of proportionality than can be inferred from the two provided data points, but has no rotational bias either clockwise or counterclockwise.

Both of these hypotheses are a for a linear model of the motor expressed as Velocity = *a**Voltage + *b*. The null hypothesis (H_0) can be rejected if there is significant evidence that shows that the alternative hypothesis (H_A) is accurate.

Experimental Design

In order to properly test the motor to characterize it, the mini-CIM motor was set up on a testbed that had a limited control system consisting of a battery, roboRIO, optical encoder, and a laptop. The motor had a test mass attached to it, which provided a small amount of inertia that would simulate real world system lag, without being something too large to be safe on an unrefined testbed. Software was written to run on the roboRIO, taking joystick input from the connected laptop and using using it to apply power to the motor. With a 10 ms looptime, the encoder was used to measure the rate at which the motor’s output shaft was spinning. Once per loop, the system timestamp, applied motor voltage, change in applied voltage, output shaft velocity, and output shaft acceleration were measured, creating a dataset of 4198 events that show the motor at high and low voltage, experiencing acceleration, and switching direction rapidly.

The testing procedure wasn’t designed to be incredibly accurate in testing a motor in the use case it would serve upon a robot, where it would typically be under heavy load, but instead was designed to be close to the “free speed” of the motor, so that the data collected could match as closely as possible with what the datasheet measured.

Weaknesses of the Design

While the design of the method for creating data to sample was intended to match the same situations the motor experienced when going through testing by the manufacturer, it is one of the reasons why the data collected is not as useful in the real world. The low load on the motor is something that likely wouldn't be encountered in a properly designed system, as it falls far below the maximum efficiency point on the motor curve, causing a waste of electrical energy to heat, instead of doing mechanical work. In the testing scenario, this leads to less system lag, backlash, and slop, meaning the model may end up fitting better than it would in the real world. However, by incorporating both the velocity and acceleration, a more accurate model can be made, however it will vary more within the region close to zero volts input. This problem could potentially be resolved by making two models, one for clockwise movement and one for clockwise movement. Following the model in a real system would be best implemented by using a "throttle bump", which would leave a discontinuity at 0 volts, but more accurately track for moves that only happen in one direction, which is the normal application of a motor during one motion profile, the most direct application for statistical models of motors, at least in FRC.

Statistical Testing Process

This recorded data was processed in a short R script that created a linear model for the sample data, along with a multiple regression relating voltage applied to the shaft's angular velocity as well as angular acceleration. Using a data processing and statistical analysis language like R is almost essential in a case like this, as inputting 4198 rows of data with 5 points each is a task that would be immensely difficult to do by hand or import into a calculator.

While the R script prints readouts that include all of the data needed to determine whether the new model can be used to reject the null hypothesis, the same thing can be done manually with a linear regression t-test. The printout from the R script displays the model in high detail:

```
Call:
lm(formula = RPM ~ Voltage, data = motorSample)

Residuals:
    Min      1Q  Median      3Q     Max 
-6771.9   -80.3    16.1    93.8  5565.6 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -67.5194     8.2360  -8.198 3.21e-16 ***
Voltage     404.1497     0.8873 455.501  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 530.7 on 4196 degrees of freedom
Multiple R-squared:  0.9802,    Adjusted R-squared:  0.9802 
F-statistic: 2.075e+05 on 1 and 4196 DF,  p-value: < 2.2e-16
```

The t-value for each coefficient can be computed by dividing its value by the Standard Error. From this, the p-value can be found by testing the likelihood of the t-statistic with the t distribution using the degrees of freedom (n-2). For the voltage coefficient, *a*, this yields a p-value smaller than what even R can determine, of less that

2E-16, and for the intercept, *b*, a p-value of 3.21E-16. This makes it easy to reject the null hypothesis, because the p-value is so incredibly low.

Conclusion

The data collected from this project has some use in modeling the dynamics of a system using a mini-CIM motor, and for a scenario with low load on the motor, the high R^2 value of .9802 says the model is very accurate. However, the process by which the data was collected, along with the code used to analyze it are more important for the final application, as it will allow others to characterize different motors with accuracy, even in configurations involving high rotational inertia and slop/backlash. The data does definitively show, however, that the datasheet's implications for the motor's performance aren't as accurate as they may seem. To see the data analysis source code used in the experiment, as well as the output data, visit the project's repository on GitHub at https://github.com/Drewsapple/MotorTesting.