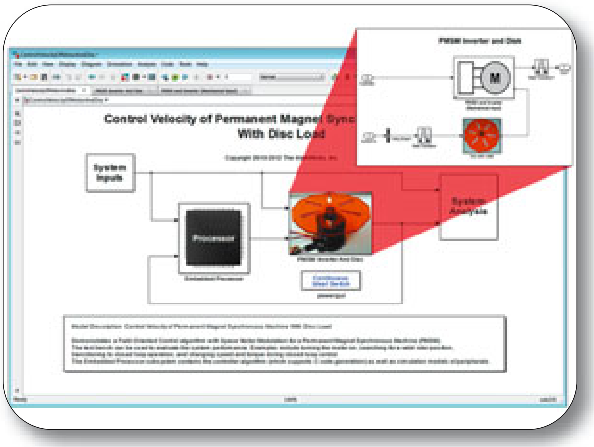
High-fidelity model of electric motor aids control system design

By Brad Hieb, MathWorks

This article describes a workflow for creating a permanent magnet synchronous machine (PMSM) plant model using Matlab and Simulink and commonly available lab equipment. The workflow involves three steps: execute tests, identify model parameters from test data, and verify parameters via simulation.



■ An accurate plant model is the linchpin of control system development using model-based design. With a well-constructed plant model, engineers can verify the functionality of their control system, conduct closed-loop model-in-the-loop tests, tune gains via simulation, optimize the design, and run what-if analyses that would be difficult or risky to do on the actual plant. Despite these advantages, engineers are sometimes reluctant to commit the time and resources required to create and validate a plant model. Concerns include how much time it will take to run a simulation, how much domain and tool knowledge will be required to build and validate the model, and what type of equipment will be needed to acquire hardware test data for building and validating the model.

We used the plant model to build and tune a closed-loop PMSM control system model. We ran step response and coast-down tests using the controller model in simulation and on hardware using an xPC Target turnkey real-time testing system. We found close agreement between the simulation and hardware results, with normalized root-mean-square deviation (NRMSD) below 2% for key signals such as rotor velocity and motor phase currents. The PMSM plant model, developed with SimPowerSystems, includes the motor and a load - in this example, an acrylic disc. The model has nine parameters that define its behaviour: one (disc inertia) associated with the load and eight associated with the motor.

We conducted five tests to characterize these parameters: the bifilar pendulum test, the back-EMF test, the friction test, the coast-down test, and the DC voltage step test. In this article, we will focus on the coast-down test and the DC voltage step tests. These tests demonstrate progressively more sophisticated methods of parameter identification, and illustrate extracting parameter values via curve fitting and parameter estimation, respectively. For each test, we describe the test setup and then explain how we conducted the test, acquired the data, extracted the parameter value, and verified it.

To characterize the rotor inertia (H) we spin the rotor up to an initial speed (ω_{r0}) and measure the rotational speed (ω) as the rotor coasts to a stop. Using this measured result, the rotor inertia can be identified by curve fitting the equation for ω_r to the measured rotational speed during the period of time when the motor is coasting to a stop. The differential equation [1] describes the mechanical behaviour of the motor. The coast-down test is set up so that the load torque (T_{load}) is always 0. Once the motor is up to an initial, steady-state speed, the motor is turned off, so that the electromagnetic driving torque (T_{em}) is also 0. Under these conditions the solution to [1] is given by the equation for ω_r [2], where r is the rotational speed of the rotor shaft, ω_{r0} is the initial rotational speed of the rotor shaft, J_0 and b are the Coulomb friction and viscous damping coefficients, respectively, characterized from a separate friction test, Tem is the electromagnetic driving torque (0 during this test), and Tload is the load torque (0 during this test).

$$\frac{d\omega_r}{dt} = \frac{1}{H}\left(T_{em} - b\omega_r - J_0 - T_{load}\right) \qquad (1)$$

$$\begin{aligned} &If \\ &T_{load} = 0 \\ &T_{em} = 0 \end{aligned}$$

Then

$$\omega_r = \left(\omega_{r0} + \frac{J_0}{b}\right) e^{-\frac{b}{H}t} - \frac{J_0}{b} \qquad (2)$$

In the lab we created an open-loop Simulink test model to drive the motor to an initial speed of 150 radians per second, at which time the motor drive was turned off and the rotor coasted to a stop. Throughout the test the model captured the output of the rotational speed sensor. Using Simulink Coder and xPC Target, we deployed this model to an xPC Target turnkey real-time system. We executed the model using xPC Target, and imported the rotor speed data into Matlab for analysis.

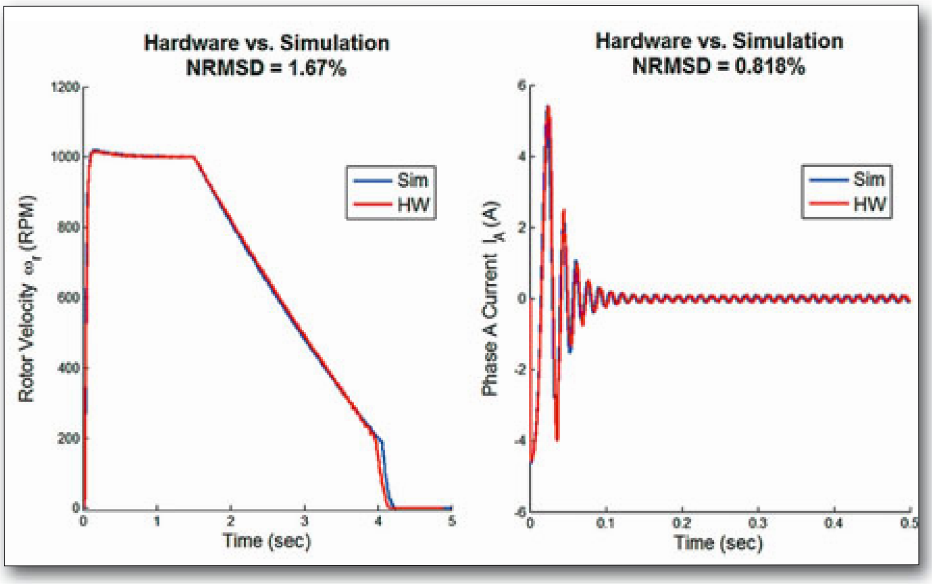


Figure 1. Comparison of simulation results (blue) with hardware results (red) for rotor velocity (left) and phase current (right)

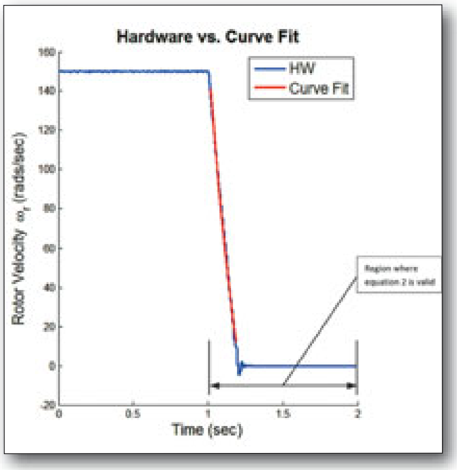


Figure 2. Plot of rotor velocity during the coast-down test. Blue = hardware test results; red = curve fitting results.

After running the tests, we plotted the measured speed data in Matlab and used Curve Fitting Toolbox to fit equation [2] for the rotor angular velocity (ω_r) to the measured speed data while the rotor was coasting to a halt. Using the value of H from the curve fit, we evaluated equation [2] from the point at which the motor started coasting and plotted the results with the original test data. The equation equation [2] with the value of H from the curve fit closely predicts the motor speed during the coast-down test.

We used a model to verify our parameter identification result. Using the rotor inertia value obtained from the coast-down test (3.2177e-06 Kg m^2 in our PMSM model), we ran a simulation of the coast-down test in Simulink. We then compared and plotted the simulated results with the measured results. The results matched closely, with a normalized root-mean-square deviation (NRMSD) of about 2%. In the DC voltage step test a DC voltage is applied across the motor phase A and phase B connections and the resulting current is measured. Electrically, under these conditions, a three-phase PMSM behaves like a circuit with two series resistors and two series inductors. The measured current (i) is used to find the resistance and inductance parameter values. During the test the rotor is held motionless to avoid complicating the analysis with back-EMF waveforms, which tend to oppose the current flow. To avoid burning out the motor with the rotor motionless, a current limiting resistor (R_{limit}) is added and a step pulse rather than a steady DC voltage is used.

We again used xPC Target and an xPC Target turnkey real-time system to conduct the test. In Simulink we developed a model that produced a series of 24-volt pulses roughly 2.5 milliseconds in duration. We deployed this model to our xPC Target system using Simulink Coder, and applied the voltage pulse across the phase A and phase B terminals of the PMSM. We measured the applied voltage and the current flowing through the motor using an oscilloscope, and using Instrument Control Toolbox we read the measured data into Matlab, where we plotted the results. Extracting the phase resistance from the measured data required only the application of Ohm's law (R = V/I) using the steady-state values for voltage and current. For the PMSM we calculated the resistance as 23.26 volts / 2.01 amps = 11.60 ohms. By subtracting 10 ohms (the value of the current limiting resistor), and dividing the result by 2 to account for the two-phase resistances in series, we calculated the motor phase resistance to be 0.8 ohms.

Characterizing the inductance required a more sophisticated approach. At first glance, it looks as if we could have used curve fitting, as we did when characterizing the rotor inertia. However, due to the internal resistance of the DC supply, the measured DC voltage decays from an initial value of 24 volts at the start of the test, when the current into the circuit is 0, to a steady-state value of 23.26 volts after the current is flowing in the circuit. Because the input voltage is not a pure step signal, the results from curve fitting the solution to the series RL circuit equation would not be accurate.

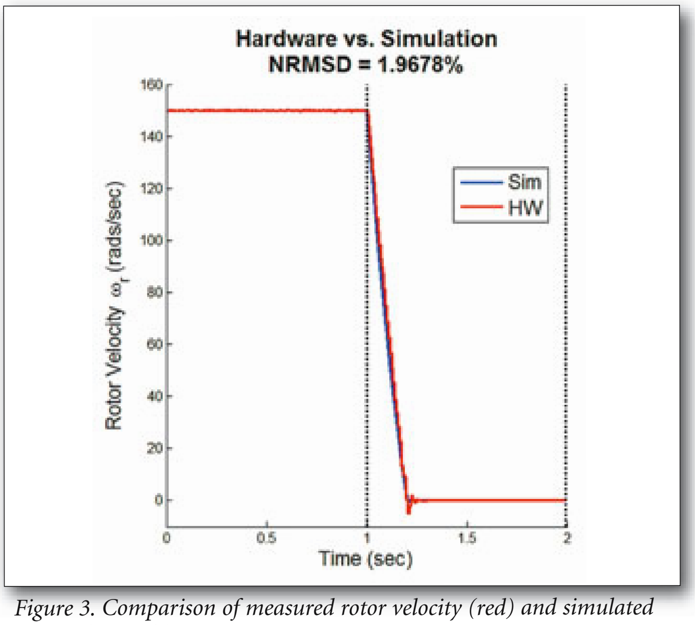


Figure 3. Comparison of measured rotor velocity (red) and simulated rotor velocity (blue)

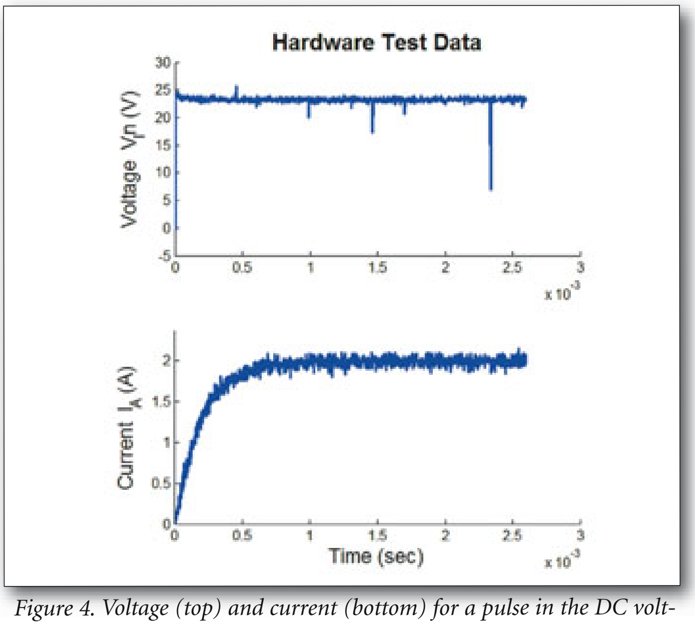


Figure 4. Voltage (top) and current (bottom) for a pulse in the DC voltage step test

To overcome this difficulty we opted for a more robust approach using parameter estimation and Simulink Design Optimization. The advantage of this approach is that it requires neither a pure step input nor curve fitting. We modelled the motor equivalent series RL circuit with Simulink and Simscape. Simulink Design Optimization applied the measured voltage as an input to the model, and with the value of the limiting resistor (R_{limit}) and the motor phase resistance (R_hat) already known, estimated the value of the inductance (L_hat) to make the current predicted by the model match the measured current data as closely as possible. To verify the values that we had obtained for phase resistance (0.8 ohms) and inductance (1.15 millihenries), we plugged the values into our PMSM model and stimulated the model with the same input that we used to stimulate the actual motor. We compared the simulation results with our measured results. The results matched closely, with an NRMSD of about 3%.

After identifying and verifying all key parameters, our PMSM plant model was ready to use in the development of the motor controller. We used Simulink Design Optimization to tune the proportional and integral gains of the controller outer loop, the velocity regulator. We ran closed-loop simulations to verify the functionality of the controller model, and used Simulink Coder to generate code from the model, which we deployed to an xPC Target turnkey real-time target machine. As a final controller verification step, we ran step response and coast-down simulations in Simulink and hardware tests using the deployed controller code on an xPC Target turnkey real-time system. We compared simulation and hardware test results for rotor velocity and phase current, and once again found close agreement between the model and the hardware, with NRMSD below 2% in both cases. ■