

# EE3501, Experiment 11

## Permanent Magnet DC Machine Simulation

Jonathan Kimball, August 31, 2020

### Abstract

This lab uses Simulink to explore the behavior of a permanent magnet dc (PMDC) machine. PMDC machines are commonly used for position control applications (sometimes termed “servos”) and as small actuators. Because their field is provided by a permanent magnet (and therefore is essentially fixed), they have some limits in flexibility but advantages in simplicity and cost. This simulation gives you an opportunity to explore their behavior.

## 1 Introduction

The equivalent circuit of a PMDC is given in Figure 1. As compared to other dc machine types, the circuit is simpler because there is no field winding. The constant  $K_m$  represents flux linkage, determined by  $K_a\Phi_d$  in machines with a field winding. The electrical and mechanical dynamic model is given by

$$L_{aa} \frac{di_a}{dt} = v_t - i_a R_a - K_m \omega_m \quad (1)$$

$$T_e = K_m i_a \quad (2)$$

$$J \frac{d\omega_m}{dt} = T_e - T_L - B_m \omega_m \quad (3)$$

where variables are defined in Table 1. Notation:  $v_t$  and  $i_a$  are assumed to be time-varying;  $V_t$  is a dc, constant value.

If you purchase a PMDC motor, its datasheet will often report two different constants: one that relates speed and voltage and another that relates torque and current. For example,

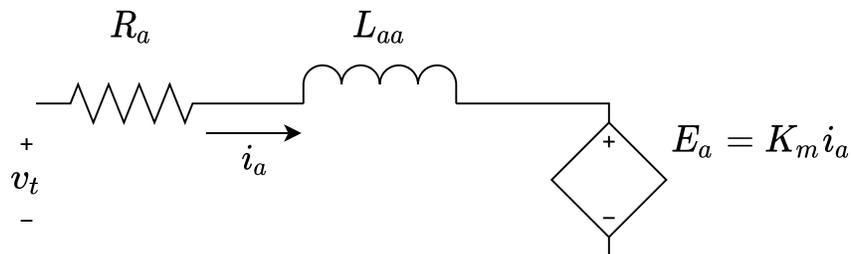


Figure 1: Equivalent circuit of PMDC machine.

Table 1: PMDC Parameters and Variables

Variable	Description	Value for This Experiment
$V_t$	Terminal voltage	6 V
$i_a$	Armature current	
$R_a$	Armature resistance	7 $\Omega$
$L_{aa}$	Armature (self) inductance	120 mH
$K_m$	Torque constant	14.1 mV $\cdot$ s/rad
$\omega_m$	Motor speed in rad/s	
$T_e$	Electromagnetic torque in N $\cdot$ m	
$T_L$	Load torque in N $\cdot$ m	
$B_m$	Viscous friction constant	6.04 $\mu$ N $\cdot$ s
$J$	Rotational inertia	$1.06 \times 10^{-6}$ kg $\cdot$ s <sup>2</sup>

the Portescap 08GS61107 has a back-EMF constant of 280  $\mu$ V/RPM and a torque constant of 2.63 mN  $\cdot$  m/A. The difference is the units:

$$0.28 \frac{\text{mV}}{\text{RPM}} = 0.28 \frac{\text{mV} \cdot \text{min}}{\text{rev}} \times \frac{\text{rev}}{2\pi \text{rad}} \times \frac{60 \text{ s}}{\text{min}} = 2.674 \text{ V} \cdot \text{s/rad} \approx 2.63 \text{ mN} \cdot \text{m/A} \quad (4)$$

Or converting the other direction,

$$2.63 \text{ mN} \cdot \text{m/A} \times \frac{2\pi}{60} = 275 \mu\text{V/RPM} \approx 280 \mu\text{V/RPM} \quad (5)$$

Which is to say that any differences may be ascribed to rounding.

## 2 Simulating a Nonlinear Dynamical System

MATLAB<sup>®</sup>/Simulink<sup>®</sup> is a development system particularly suited to simulating dynamical systems. A dynamical system is defined by a set of state variables, each of which evolves according to a differential equation. In the present case, the state variables are  $i$ ,  $x$ , and  $v$ . The most straightforward way to simulate the behavior of a dynamical system is to compute the derivatives of the state variables, then use integrators to compute the actual state variables. This system has two state equations, repeated here for convenience, with the derivatives isolated:

$$\frac{di_a}{dt} = \frac{1}{L_{aa}} (v_t - i_a R_a - K_m \omega_m) \quad (6)$$

$$\frac{d\omega_m}{dt} = \frac{1}{J} (T_e - T_L - B_m \omega_m) \quad (7)$$

(Time derivatives arise often enough that there is special notation using a dot, e.g.  $\dot{x} = v$ . When one state variable is  $i$ , though, this can become confusing.) So a simulation may be built to compute the right-hand sides of (6)-(7) and integrate them to find the state variables.

The complete system is described by the state variables, the inputs  $v_t$  and  $T_L$ , and outputs. In our simulation, outputs will be the two state variables, the back-EMF  $E_a = K_m \omega_m$ , and the torque  $T_e = K_m i_a$ .

A Simulink diagram is illustrated in Figure 2. Parameters are given in Table 1.

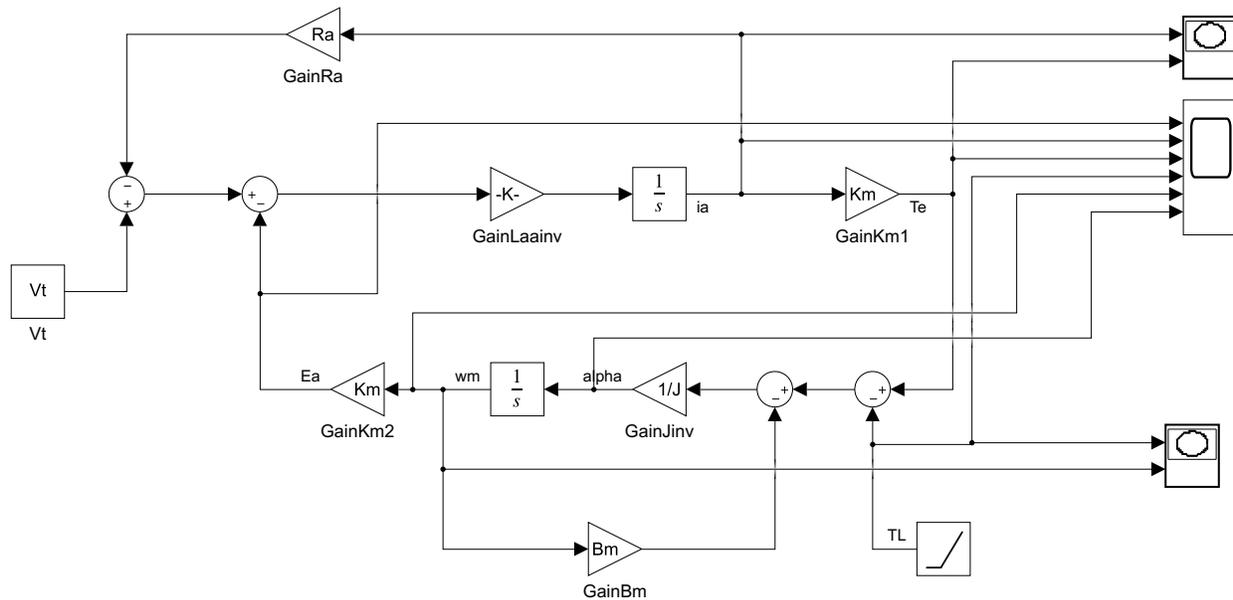


Figure 2: Simulink diagram.

### 3 Laboratory Experiment (Simulation)

Obtain the Simulink file and open it. On the Modeling tab, open the Model Explorer and navigate within the tree to the Model Workspace. You should see all of the parameters listed there. Verify that the parameters match those given in Table 1. Back on the Modeling tab, click on the Model Settings. Ensure that the start time is 0.0, stop time is 1.0, and the solver is Variable-step, auto.

There are two inputs,  $V_t$  and  $T_L$ .  $V_t$  is set to a constant 6 V.  $T_L$  is set to zero (no-load) for the first 500 ms, and then to ramp up. Run the simulation and observe the behavior of the system. Save the Scope view by copying it to the clipboard and pasting it into a Word doc, or by exporting it to the workspace and processing it as you desire. Do similarly for the two XY graphs.

Repeat this process for variations in each of the parameters in Table 1. Parameters may be edited within the Model Explorer or Model Data Editor. Each parameter available has a defined minimum and maximum; vary one parameter at a time to its limits, then return it to nominal.

### 4 Calculations and Question

1. For each of the variations in parameters, discuss the impact on static and dynamic behavior.
2. Suppose that your objective is for the motor to reach an equilibrium point as quickly as possible, with minimum overshoot and minimum settling time. Which parameters have the most significant effect, and what values should they have?