Dynamics of a Brushed DC Motor
ENG ME 302 Engineering Mechanics II

1. Introduction.

A DC motor is a device that transforms electrical energy to rotational mechanical energy. In order for a DC motor to be used effectively as a component of a dynamic system, the transient relation between the electrical input and the mechanical output must be quantified. In this laboratory exercise, a dynamic model of a brushed DC motor is investigated. Parameters of the model are measured, and the transient response predicted by the model is compared with experimental data.

2. Modeling assumptions.

A model of a brushed DC motor is shown in Fig. 1. The model is phenomenological; the details of the internal mechanical and magnetic operation of the motor are not considered here. The essential element of the model is an electromechanical transducer that converts electrical energy to rotational mechanical energy. The constitutive equations that relate the electrical variables of the transducer to the mechanical variables of the transducer are

$$v_T = T\dot{\phi} \tag{1}$$

and

$$\tau = Ti, \tag{2}$$

where v_T is the voltage across the electrical ports of the transducer, $\dot{\phi}$ is the angular velocity of shaft of the transducer, i is the current into the transducer, τ is the torque on the shaft of the transducer, and T is the transducer constant. It follows from eqns. (1) and (2) that

$$v_T i = \tau\dot{\phi}. \tag{3}$$

With the senses of v_T, i, $\dot{\phi}$, and τ defined in Fig. 1, eqn. (3) states that the electrical power into the transducer is equal to the mechanical power out of the transducer. The transducer is thus a *conservative* electromechanical device.

The electrical ports of the transducer are connected in Fig. 1 to a resistor R that represents the electrical resistance of the motor windings. The voltage v_e in Fig. 1 is the voltage across the actual electrical terminals of the motor. Kirchhoff's voltage law for the loop that contains the external voltage, the resistor, and the electrical ports of the transducer is

$$v_T + iR = v_e. \tag{4}$$

The output shaft of the transducer in Fig. 1 is connected to a rotor that represents the mechanical load of the motor and a rotational dashpot that

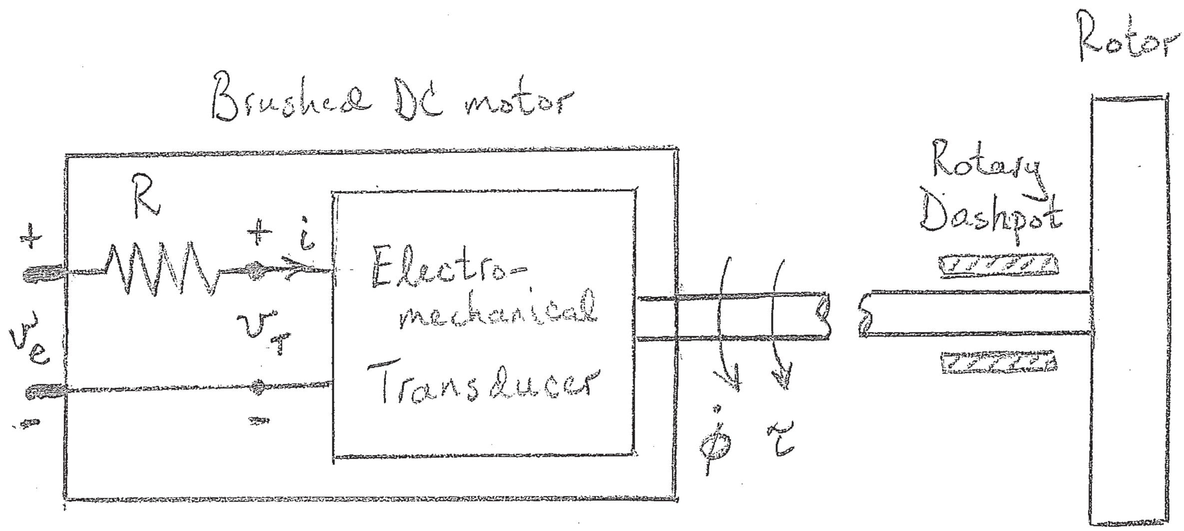

Fig. 1. Phenomenological model of a brushed DC motor.

represents the friction in the bearings of the motor. The rotary dashpot produces a torque $b\dot{\phi}$ on the rotor that opposes its rotation. The parameter b is the rotary dashpot constant. The angular momentum equation for the rotor is

$$I\ddot{\phi} + b\dot{\phi} = \tau, \tag{5}$$

where I is the moment of inertia of the rotor about its axis of rotation.

Exercise 1. Use eqns. (1)–(5) to derive the differential equation

$$I\ddot{\phi} + \left(b + \frac{T^2}{R}\right)\dot{\phi} = \frac{T}{R}v_e \tag{6}$$

that relates the external voltage v_e to the angular position ϕ of the motor shaft. Since the angular velocity ω of the motor is equal to $\dot{\phi}$, eqn. (6) can also be written as

$$I\dot{\omega} + \left(b + \frac{T^2}{R}\right)\omega = \frac{T}{R}v_e. \tag{7}$$

2. Motor driver system.

Fig. 2 is a schematic of the Arduino microprocessor and motor driver board (Pololu MC33926) that are used in this lab to drive the DC motor. The motor is provided with a rotary encoder that senses changes in the angular position of the motor shaft. One revolution of the motor shaft corresponds to 64 counts of the encoder index. The output of the encoder is connected to the Arduino board for processing and display. The sample Arduino program in Appendix 1 illustrates the commands for driving the motor and collecting the data from the encoder. The voltage that is supplied to the

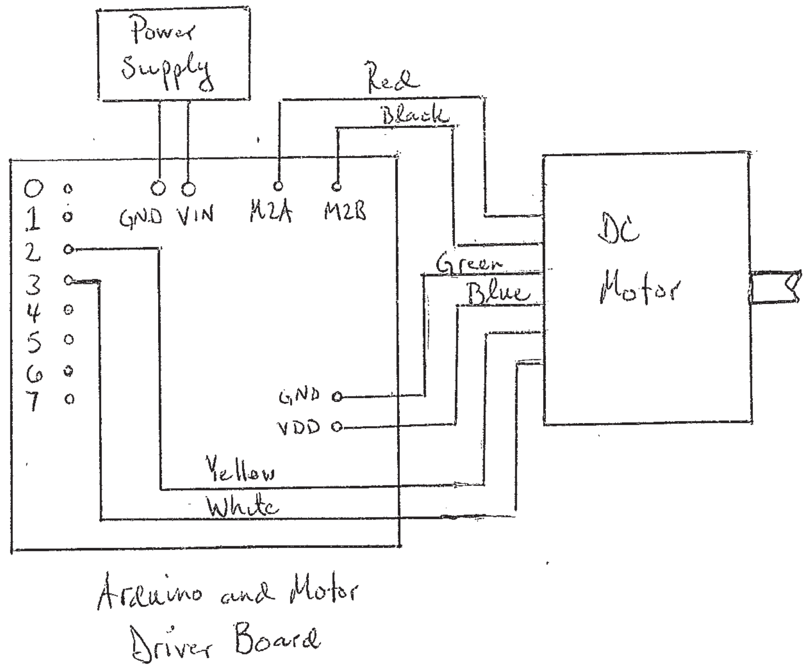

Fig. 2. Arduino microprocessor and motor driver board.

motor by the motor driver board is specified in the Arduino program by the integer argument of the `md.setM2Speed` command. This integer can have a value between −400 and 400. The integer argument of the `md.setM2Speed` command is *not* the speed of the motor; it is an integer that determines the *voltage* that is supplied to the motor by the motor driver board.

Before coming to the lab, download the Arduino software package from `https://www.arduino.cc/en/Main/Software`, and install the software on your laptop computer. Also, download the motor driver and encoder libraries from the course blackboard site, and install these libraries in your Arduino directory.

3. Parameter identification.

In order to use the differential equation in Exercise 1 to predict the performance of the motor, numerical values for the motor parameters, T, b, and R must be obtained. This can be accomplished by the following procedure.

1. Use a voltmeter to determine the relationship between the voltage input v_e to the motor and the integer argument of the `md.setM2Speed` command in the Arduino program that is used to drive the motor.

2. To determine the transducer constant T, couple two motors as shown in Fig. 3. Use the Arduino to run the driving motor at a constant speed, and use the encoder output of the driving motor to calculate the angular velocity $\dot{\phi}$ of the driving (and therefore of the driven) motor.

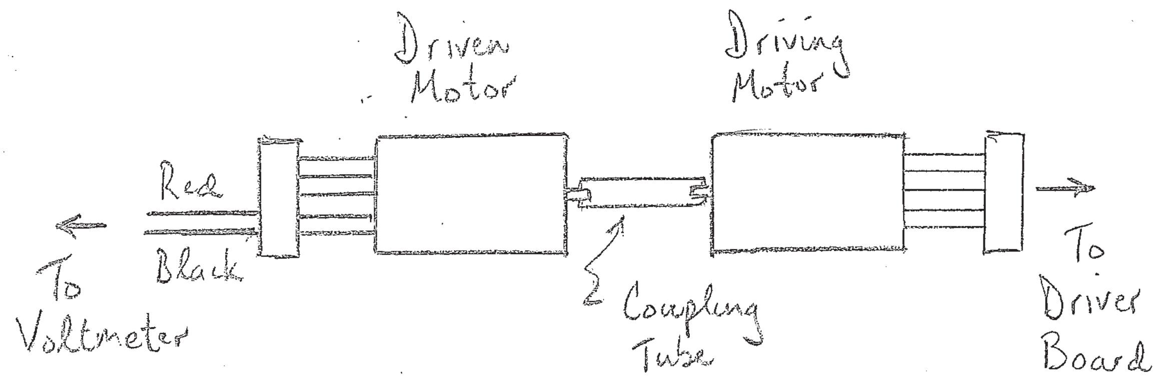

Fig. 3. Configuration for measurement of transducer constant T.

Use a voltmeter to measure the voltage v_e across the terminals of the driven motor while it is spinning. (The driven motor is operated as a *generator* in this experiment.) Since the voltmeter draws very little current, the relation between the angular velocity $\dot{\phi}$ and the external voltage v_e of the driven motor in this configuration is given by eqns. (1) and (4) as

$$v_e = T\dot{\phi}. \tag{8}$$

Measure the external voltage v_e of the driven motor for several different values of the angular velocity $\dot{\phi}$, and plot v_e vs. $\dot{\phi}$. Is the relation between v_e and $\dot{\phi}$ linear? If it is, the transducer constant T is the slope of the line that relates v_e and $\dot{\phi}$. Interchange the driving and driven motor, and repeat the determination of T. Are the transducer constants of the two supposedly identical motors the same?

3. Calculate the moment of inertia I of the rotor from measurements of the mass and the dimensions of the rotor.

4. Determine the dashpot constant b by an experiment in which the rotor is given an rapid initial spin by hand while the input terminals to the motor are open. Use the encoder output to plot the subsequent decay of the angular velocity of the rotor. Since the motor current i is zero in this experiment, eqns. (2) and (5) for this experiment become

$$I\ddot{\phi} + b\dot{\phi} = Ti = 0, \tag{9}$$

or

$$I\dot{\omega} + b\omega = 0, \tag{10}$$

Solve this differential equation for ω analytically, and compute the dashpot constant b by fitting the experimental data for ω to the analytic solution of the differential equation.

5. To measure the electrical resistance R of the motor, apply a voltage v_e of 2 or 3 volts to the motor while holding the rotor fixed. (Don't

hold the rotor fixed too long, or the motor may overheat.) Measure the current (this is the "stall current") to the motor, and use Ohm's law to calculate the resistance. Repeat the measurement for several angular positions of the rotor, since the contact between the internal brushes and the rotor may change slightly with angular position. The data sheet for the motor used in this lab (https://www.pololu.com/product/1440) lists a stall current of 5 A at an input voltage of 12 V. Are these values consistent with your measurements?

4. Model verification.

To test the predictive value of the motor model, use eqn. (7) to predict the angular velocity of the motor as a function of time when a step change in the external voltage v_e is applied. Use the encoder output to measure the actual angular velocity of the motor as a function of time when a step change in the external voltage v_e is applied via the Arduino program. Compare the measured angular velocity with angular velocity predicted by the differential equation.

Appendix A. Sample Arduino program

```
#include "DualMC33926MotorShield.h"
#include <Encoder.h>

Encoder motor(2,3);
DualMC33926MotorShield md;

int newtime = 0;
int oldtime = 0;
int newphi = 0;
int oldphi = 0;

void stopIfFault()
{
if (md.getFault())
{
Serial.println("fault");
while(1);
}
}

void setup()
{
Serial.begin(9600);
md.init();
}

void loop()
{
md.setM2Speed(200);
newtime = millis();
newphi = motor.read();
Serial.print(oldtime);
Serial.print(" ");
Serial.print(newtime);
Serial.print(" ");
Serial.print(oldphi);
Serial.print(" ");
Serial.print(newphi);
Serial.print(" ");
Serial.println(md.getM2CurrentMilliamps());
oldtime = newtime;
oldphi = newphi;
}
```