Objectives

The objective of the lab is to become familiar with Matlab/Simulink by implementing a simple DC motor simulation with a proportional/integral control law. The lab also provides an opportunity to learn about saving figures for reports.

1 Introduction

Videos available on the main web page show the steps needed to perform the experiments of this lab. Other tutorials may also be consulted on-line.

2 Experiments

(a) Create a Simulink block diagram to simulate the response of a brush DC motor described by

$$\frac{d\omega(t)}{dt} = -100\omega(t) + 1000v(t),$$

where \( \omega \) (rad/s) is the angular velocity and \( v \) (V) is the voltage applied. Fig. 1 shows the Simulink diagram of a possible implementation.

![Simulink Diagram](image)

Figure 1: Diagram for the simulation of a first-order system

Simulate the response for 0.2 s, applying a step of voltage of 10 V at \( t = 0.1 \) s. Plot the speed as a function of time.
(b) Augment the previous simulation with a proportional-integral controller

\[ v(t) = k_P (\omega_{REF}(t) - \omega(t)) + k_I \int (\omega_{REF}(\tau) - \omega(\tau)) \, d\tau. \]  

(2)

Let \( k_P = 0.02 \) and \( k_I = 4 \). Fig. 2 shows the Simulink diagram of a possible implementation.

![Simulink Diagram](image)

**Figure 2:** Diagram for the simulation of a first-order system with a PI controller

Simulate the response for 0.5 s, applying a step of reference velocity of 100 rad/s at \( t = 0.1 \) s and another step of 200 rad/s at \( t = 0.3 \) s. Plot the speed and the voltage as functions of time. Also include a picture of your final Simulink diagram and of any sub-blocks or embedded m-files.

### 3 Report at a glance

Be sure to include:

- Plot of open-loop velocity response.
- Plot of closed-loop velocity response.
- Plot of closed-loop voltage response.
- Picture of final Simulink diagram and of any sub-blocks or embedded m-files.

In addition, the report should include comments and explanations for the work performed and the results observed/produced. Be sure to label the axes of all your plots, and to use line widths and font sizes that provide good readability. Also include units for all of your values.
Lab 2 – Open and closed-loop control of a single-phase motor

Objectives

The objective of this lab is to observe the responses of a single-phase motor controlled using a Hall effect sensor and supplied by a DC source. Speed and torque ripples are estimated. The experiments also show that the motor can be operated under very limited conditions using an open-loop controller.

1 Introduction

A brushless DC fan motor is considered. The model of Example 1 in Chapter 1 is modified for an arbitrary number of pole pairs \( n_P \), yielding

\[
L \frac{di_A}{dt} = v_A - Ri_A + K \omega \sin(n_P \theta) \\
J \frac{d\omega}{dt} = -K \sin(n_P \theta) i_A - \tau_{LF} \\
\frac{d\theta}{dt} = \omega,
\]

(3)

where \( v_A (V) \) and \( i_A (A) \) are the voltage and the current applied to the motor winding, \( \omega \) (rad/s) is the angular velocity of the motor, \( \theta \) (rad) is the angular position, and \( \tau_{LF} \) (Nm) is the load and friction torque. The torque is assumed to be due to aerodynamic friction and be of the form

\[
\tau_{LF} = B_2 \omega^2 \text{sign}(\omega).
\]

(4)

The parameters are:

- \( R (\Omega) \), the resistance of the phase winding.
- \( L (H) \), the inductance of the phase winding.
- \( K \) (N m/A or V s), the torque constant, or back-emf constant, with \( K = n_P \psi_0 \) and \( \psi_0 \) is the peak value of the total flux linkage due to the permanent magnet.
- \( J \) (kg m\(^2\)), the rotor inertia.
- \( B_2 \) (N m s\(^2\)), the coefficient of the friction torque.

The motor is controlled using a single Hall effect sensor and a DC supply, so that

\[
v_A = -V_{dc} \text{sign}(\sin(n_P \theta)),
\]

(5)

where \( V_{dc} \) is the DC supply voltage.
2 Experiments

(a) Implement the model of the motor and control law in Simulink. Fig. 3 shows the Simulink diagram of a possible implementation.

Figure 3: Diagram for the simulation of a single-phase motor

Let \( R = 26 \ \Omega \), \( L = 5 \ \text{mH} \), \( K = 0.024 \ \text{N m/A} \), \( J = 2.8 \times 10^{-5} \ \text{kg m}^2 \), \( B_2 = 4.6 \times 10^{-8} \ \text{N m s}^2 \), \( n_P = 2 \), and \( V_{dc} = 12 \ \text{V} \). Let all initial conditions be zero, except for \( \theta(0) = \pi/6 \). Plot the speed in rpm and the torque in N m as functions of time for 10 s. Estimate the average speed and the average torque in steady-state, and the percent ripples (approximated as half of the peak-to-peak value divided by the average value).

(b) Repeat the simulations for \( V_{dc} = 3 \ \text{V}, 6 \ \text{V}, 9 \ \text{V}, \) and \( 12 \ \text{V} \) and plot the steady-state speed as a function of the DC voltage.

(c) Replace the control law by the open-loop control law

\[
\nu_A = -V_{dc} \ \text{sign}(\sin(n_P \theta_d)). \tag{6}
\]

The angle \( \theta \) of (5) was replaced by a desired angle

\[
\theta_d = \omega_d t, \tag{7}
\]

and \( \omega_d \) is some desired rotational speed. Simulate the responses of the motor with this open-loop control law. The objective is to show that the motor will rotate at the speed \( \omega_d \), but only for a very restricted set of frequencies. Find a constant speed reference \( \omega_d \) for which the motor position \( \theta \) follows the reference position \( \theta_d \) (search in the range from 10 to 50 rad/s). Plot \( \theta \) and \( \theta_d \) as functions of time. Comment on the speed ripple, and on what happens when \( \theta(0) = -\pi/6 \).
3 Report at a glance

Be sure to include:

• Plot of the speed and torque, and estimates of average speed and torque, and values of ripple.
• Plot of the steady-state speed as a function of the DC voltage.
• Plot $\theta$ and $\theta_d$ for a speed $\omega_d$ such that the motor stays synchronized, and value of $\omega_d$.

In addition, the report should include comments and explanations for the work performed and the results observed/produced. Be sure to label the axes of all your plots, and to use line widths and font sizes that provide good readability. Also include units for all of your values.
Lab 3 – Modeling of the brush DC motor

Objectives

The objective of the lab is to observe the characteristics of a brush DC motor. The responses of the DC motor to steps of voltage are measured, and the data is used to estimate its parameters.

1 Introduction

A model for the brush DC motor with Coulomb and viscous friction is

\[
\begin{align*}
L \frac{di}{dt} &= v - Ri - K\omega \\
J \frac{d\omega}{dt} &= Ki - C \text{sign}(\omega) - B\omega,
\end{align*}
\]

where \( v \) (V) is the voltage applied to the motor, \( i \) (A) is the current, and \( \omega \) (rad/s) is the angular velocity of the motor. The parameters are:
- \( R \) (Ω), the armature resistance.
- \( L \) (H), the armature inductance.
- \( K \) (Nm/A or V s), the torque constant, also called the back-emf constant.
- \( J \) (kg m²), the rotor inertia.
- \( C \) (N m), the coefficient of Coulomb friction.
- \( B \) (N m s), the coefficient of viscous friction.

Assuming that the inductance is negligible, \( v = Ri + K\omega \) and the following approximate model is obtained

\[
J \frac{d\omega}{dt} = - \left( \frac{K^2}{R} + B \right) \omega + \frac{K}{R} v - C \text{sign}(\omega).
\]

With positive direction of motion, the Coulomb friction can be viewed as a constant disturbance, and the transfer function from the voltage to the angular velocity is a first-order linear time-invariant system.

2 Experiments

2.1 Parameter estimation

The simulation model \( \text{Lab3.mdl} \) applies a sequence of voltages to the motor at specified times. Fig. 4 shows the diagram of the simulation. The \textit{Command generator} applies six increasing steps of voltages to the motor.
Figure 4: Diagram of the brush DC motor simulation

Make yourself familiar with the model, then run the simulation. In Matlab’s workspace, plot $v(t)$, $\omega(t)$ and $i(t)$. Note that, after a transient period starting when $v(t)$ changes, $\omega(t)$ and $i(t)$ reach constant values. These values are such that

\[
\begin{align*}
    v &= R \, i + K \omega \\
    i &= \frac{C}{K} + \frac{B}{K} \, \omega.
\end{align*}
\]  

Proceed with the following tasks:
(a) Using Matlab’s data cursor, measure the steady-state values, and create a table with the six values of voltage, velocity, and current.
(b) Plot $i(\omega)$ as well as your best fit of the line $i = a + b\omega$. The parameters $a$ and $b$ are your estimates of $C/K$ and $B/K$.
(c) Repeat (b) with a plot of $v/i$ as a function of $\omega/i$, and deduce estimates of $R$ and $K$. Using the results of part (b), deduce values of $C$ and $B$.
(d) Plot $\omega(t)$ and, from the response to the first step of voltage, estimate the time constant of the system. Based on the first-order model (9), the time constant is equal to

\[
\tau = \frac{J}{K^2/R + B}.
\]

Deduce an estimate of $J$.
(e) Plot $i(t)$ and, from the response to the first step of voltage, estimate $di/dt$ at the initial time of transition. Deduce an estimate of $L$.
(f) Observe that the variable defining the simulation step size is $T_S$ using Simulation/Model Configuration Parameters. Then, find the value of $T_S$ in the initialization function by clicking:
View, Model Explorer, Model Explorer, Lab2*, Callbacks, InitFcn*. The initialization function is a useful technique to initialize parameters that appear in several places in the simulation model. Verify that the delay observed in the response of part (e) matches the simulation step size.

(g) Look inside the simulation of the brush DC motor and compare the estimated values to the values used in the model. Observe that some parameters are estimated more reliably than others with the method proposed.

2.2 Sensor noise

The simulation model includes a model of the encoder that is used in practice to measure the angular position. The model is a quantizer, performing an operation similar to the one performed in A/D converters. The measured velocity is obtained from the measured position using

\[ \omega_{\text{meas}}(t) = \frac{\theta_{\text{meas}}(t) - \theta_{\text{meas}}(t - T_S)}{T_S}, \]

where \( T_S \) is the sampling period. The velocity is also filtered by a block that averages the latest \( N_{\text{avg}} \) values of the estimates. Differentiation of the quantized position produces significant noise on the velocity estimate. Averaging of the velocity reduces the noise, and is equivalent to choosing a period \( N_{\text{avg}} T_S \) in the velocity estimate. Proceed with the following tasks:

(a) Find the value of \( N_{\text{avg}} \) in the initialization function of the simulation model.

(b) Find the quantization level \( \Delta \theta \) from the quantizer block of the model. Also observe the effect of quantization on \( \theta_{\text{meas}}(t) \). Because the data includes periods of low velocity, the quantization level can be determined from the minimum non-zero value of the signal obtained by applying the Matlab function \text{diff( )} \) to \( \theta_{\text{meas}}(t) \).

(c) Plot \( \omega_{\text{meas}}(t) \) and zoom in on the steady-state response after the first step to obtain an estimate of the peak-to-peak velocity noise on the measurement. Repeat for the last step of velocity. How does the noise compare at low and high velocities? Observe that the level of noise \( \Delta \omega \) on the velocity can be related to the level of quantization noise \( \Delta \theta \) obtained in part (b) using

\[ \Delta \omega = \frac{\Delta \theta}{N_{\text{avg}} T_S} \]

Note that averaging of the velocity estimate reduces the noise, but also delays the velocity estimate.

3 Report at a glance

Be sure to include:

- Table of values of \( \omega, v, i \).
- Plots of \( i(\omega), v/i (\omega/i), \omega(t), i(t) \) and values of the estimated parameters.
• Comparison of the estimated parameters to the values of the model.
• Values of $T_S$, $N_{avg}$, $\Delta \theta$, and $\Delta \omega$ and calculation for $\Delta \omega$.

In addition, the report should include comments and explanations for the work performed and the results observed/produced. Be sure to label the axes of all your plots, and to use line widths and font sizes that provide good readability. Also include units for all of your values.
Lab 4 – Control of a brush DC motor

Objectives
The objective of the lab is to experiment with a proportional-integral (PI) control law for the tracking of velocity references by a brush DC motor. A proportional-integral-derivative (PID) control law is also tested for positioning. Controller parameters are selected using a pole placement procedure, and modifications are applied to the control law to avoid overshoot in the responses.

1 Introduction
In Lab 3, (9) gave an approximate model of a brush DC motor under voltage command
\[ \frac{d\omega}{dt} = -a \omega + k_1 v, \] (14)
where
\[ a = \frac{K^2}{R J}, \quad k_1 = \frac{K}{R J}. \] (15)
Friction was neglected, or treated as a disturbance. In the Laplace domain, the transfer function of the system is
\[ \frac{\omega(s)}{v(s)} = \frac{k_1}{s + a}. \] (16)
In this lab, the first-order system representation of the motor is used to design control laws for velocity and position.

2 PI control of velocity
A proportional-integral (PI) control law for the velocity consists in choosing a voltage
\[ v = k_P (\omega_{REF} - \omega) + k_I \int (\omega_{REF} - \omega) dt, \] (17)
where \( \omega_{REF} \) is the reference velocity, and \( k_P \) and \( k_I \) are the proportional and integral controller gains. The transfer function of the compensator is
\[ C(s) = k_P + \frac{k_I}{s}, \] (18)
so that the closed-loop transfer function is given by

\[
\frac{\omega(s)}{\omega_{REF}(s)} = \frac{k_1(k_Ps + k_I)}{s^2 + (a + k_I k_P)s + k_1 k_I}.
\] (19)

The closed-loop poles are placed at some desired value \( s = -a_d \) if the denominator of the transfer function is equal to \( s^2 + 2a_d s + a_d^2 \). This property leads to the values of the PID gains

\[
k_P = \frac{2a_d - a}{k_1}, \quad k_I = \frac{a_d^2}{k_1}.
\] (20)

The zero of the compensator is then equal to

\[
z = -\frac{k_I}{k_P} = -\frac{a_d^2}{2a_d - a}. \tag{21}
\]

An alternative option used here consists in choosing the PI gains so that the zero of the compensator is at

\[
z = -\frac{k_I}{k_P} = -a. \tag{22}
\]

This choice results in a pole/zero cancellation, which is acceptable given that \( a > 0 \), and \( a \) is relatively large and well-known. The closed-loop transfer function is then

\[
\frac{\omega(s)}{\omega_{REF}(s)} = \frac{k_1 k_P(s + a)}{s(s + a) + k_1 k_P(s + a)} = \frac{k_1 k_P}{s + k_1 k_P}, \tag{23}
\]

which is a first-order system with pole at \( s = -k_1 k_P \). For a desired pole at \( s = -a_d \), the PI gains are given by

\[
k_P = \frac{a_d}{k_1}, \quad k_I = \frac{a a_d}{k_1}. \tag{24}
\]

### 3 PID control of position

For the control of position, the transfer function of the system becomes

\[
\frac{\theta(s)}{v(s)} = \frac{k_1}{s(s + a)}. \tag{25}
\]

A proportional-integral-derivative (PID) control law is used and is given by

\[
v = k_P(\theta_{REF} - \theta) + k_I \int (\theta_{REF} - \theta) dt + k_D \frac{d}{dt} (\theta_{REF} - \theta), \tag{26}
\]

where \( \theta_{REF} \) is the reference position of the motor, and \( k_P \), \( k_I \), and \( k_D \) are the proportional, integral, and derivative gains, respectively. Heuristically, one finds that, for increasing values of the parameters, \( k_P \) accelerates the speed of the response, \( k_D \) improves the damping, and
$k_I$ forces the return of the steady-state error to zero but slows down the system. However, the response of the closed-loop system depends on all three parameters in a complex manner, making manual tuning difficult. To simplify the selection of the controller parameters, a method is used to compute the gains as functions of a single parameter, namely the desired location of the closed-loop poles.

The closed-loop transfer function is

$$\frac{\theta(s)}{\theta_{REF}(s)} = \frac{(k_D s^2 + k_P s + k_I)k_1}{s^3 + (a + k_D k_1)s^2 + k_P k_1 s + k_I k_1}.$$  \hspace{1cm} (27)

The closed-loop poles are placed at some desired value $s = -a_d$ if the denominator of the transfer function is equal to $s^3 + 3a_d s^2 + 3a_d^2 s + a_d^3$. This property leads to the values of the PID gains

$$k_D = \frac{3a_d - a}{k_1}, \quad k_P = \frac{3a_d^2}{k_1}, \quad k_I = \frac{a_d^3}{k_1}.$$  \hspace{1cm} (28)

The zero of the compensator is equal to

$$z = -\frac{k_I}{k_P} = -\frac{a_d}{3}.$$  \hspace{1cm} (29)

4 Improvements

Some modifications are brought to the control law to improve the responses:

(a) **Anti-windup**: for large steps of the reference input, the response of the system exhibits an overshoot, even if no overshoot is observed for small steps. This problem occurs because the control input reaches a limit and the response of the system deviates from the linear response. The integrator value becomes large, but the output does not produce the negative feedback needed to stabilize the integrator state. To fix the problem, integration may be stopped when a limit is reached.

(b) **Modified gains**: in the PID control law, the derivative action is only applied to $\theta$, so that large control values do not result from step changes in the reference input. The proportional term is also modified to reduce the gain applied to $\theta_{REF}$. Then, the control law becomes

$$v = k_F k_P \theta_{REF} - k_P \theta + k_I \int (\theta_{REF} - \theta)dt - k_D \frac{d\theta}{dt},$$  \hspace{1cm} (30)

where the feedforward gain $k_F \leq 1$. With the modification, the transfer function becomes

$$\frac{\theta(s)}{\theta_{REF}(s)} = \frac{(k_F k_P s + k_I)k_1}{s^3 + (a + k_D k_1)s^2 + k_P k_1 s + k_I k_1}.$$  \hspace{1cm} (31)

The controller zero is moved to

$$z = -\frac{k_I}{k_F k_P} = -\frac{a_d}{3k_F}.$$  \hspace{1cm} (32)
A value of $k_F < 1$ is used if an overshoot of the response is observed due to the zero having a smaller magnitude than the closed-loop poles. Possible choices include $k_F = 0$, which removes the controller zero altogether, and $k_F = 1/3$, which results in the cancellation of a closed-loop pole.

5 Experiments

5.1 PI control of velocity

(a) Implement the PI control law using the brush DC motor simulation file Lab3.mdl. Use the $\omega_{\text{meas}}$ output from the velocity estimator as the feedback to the controller. For $a_d$, use 120 rad/s (this value is approximately equal to $2a$, so this choice represents an acceleration of the open-loop response by a factor of 2). Include anti-windup protection to account for the voltage limit of 25 V.

Simulate the response of the system for a sequence of reference values equal to 0 rpm, 1000 rpm, 2000 rpm, 3000 rpm, 2000 rpm, and 0 rpm, separated by 1 s from each other. Remember, though, that the speed in the simulation is expressed in rad/s. Plot $\omega_{\text{meas}}$ and $\omega_{\text{REF}}$ (in rpm) as functions of time and discuss the results. In particular, explain why the motor is not capable of reaching the highest speed.

A suggestion for the PI controller is to implement the algorithm as the combination of an integrator and a Matlab m-function, as shown on Fig. 5. All the equations of the algorithm can then rapidly be coded in the PI controller block. Compute exact values of the controller gains and check that the values rounded to the first significant digit are $k_P = 0.2$ and $k_I = 10$. Fig. 6 shows a possible organization of the overall simulation, including the PI controller as a sub-block.

(b) Repeat part (a) with a larger value of $a_d$ (800 rad/s), and discuss the results.
5.2 PID control of position

(a) Implement the PID control law with $a_d = 60$ rad/s. Check that your values of the controller gains rounded to the first significant digit are $k_P = 20$, $k_I = 400$, and $k_D = 0.2$. Use $\omega_{meas}$ for $d\theta/dt$ and $\theta_{meas}$ for $\theta$. Include anti-windup protection to account for the voltage limit of 25 V and implement the modifications of (30) with $k_F = 0$.

Simulate the response of the system for a sequence of reference values equal to 0 deg, 90 deg, 3600 deg, and 0 deg, separated by 1 s from each other. Remember, though, that the position in the simulation is expressed in rad. Plot $\theta_{meas}$ and $\theta_{REF}$ (in degrees) as functions of time and discuss the results. Also plot the responses focusing on the first 90 degree step response.

(b) Test the system of part (a) without the anti-windup protection and with $k_F = 0$, and observe the windup phenomenon.

(c) Test the system of part (a) with anti-windup protection and with $k_F = 1$, and observe the overshoot on the response to the 90° step. When the voltage limit is not reached, the behavior is linear so that this overshoot is different than the one observed in part (b).

6 Report at a glance

Be sure to include:

- Values of the PI gains for the velocity controller.
- Plot of $\omega_{meas}$ and $\omega_{REF}$ as functions of time for $a_d = 120$ rad/s.
- Plot of $\omega_{meas}$ and $\omega_{REF}$ as functions of time for $a_d = 800$ rad/s.
- Values of the PID gains for the position controller.
- Plot of $\theta_{\text{meas}}$ and $\theta_{\text{REF}}$ as functions of time when $k_F = 0$ and with anti-windup protection. Additional plot focused on the 90° step.
- Plot of $\theta_{\text{meas}}$ and $\theta_{\text{REF}}$ with $k_F = 0$ and anti-windup protection disabled.
- Plot of $\theta_{\text{meas}}$ and $\theta_{\text{REF}}$ when $k_F = 1$ focused on the 90° step.

In addition, the report should include comments and explanations for the work performed and the results observed/produced. Be sure to label the axes of all your plots, and to use line widths and font sizes that provide good readability. Also include units for all of your values.
Lab 5 – Open-loop control of a stepper motor

Objectives

The objective of the lab is to experiment with different methods of open-loop control for stepper motors. Sinusoidal and four-step commutation methods are implemented. The maximum speed attainable with these methods is investigated. The oscillatory response to a step in four-step commutation is also observed. A DQ method of control is implemented, and the significant increase of speed attainable using field weakening is observed.

1 Introduction

1.1 Stepper motor model

The model of a two-phase stepper motor is given by

\[
\begin{align*}
L \frac{d i_A}{dt} &= v_A - R i_A + K \omega \sin(n_p \theta) \\
L \frac{d i_B}{dt} &= v_B - R i_B - K \omega \cos(n_p \theta) \\
J \frac{d \omega}{dt} &= -K i_A \sin(n_p \theta) + K i_B \cos(n_p \theta) - \tau_{LF} \\
\frac{d \theta}{dt} &= \omega,
\end{align*}
\]

where \( v_A \) and \( v_B \) (V) are the voltages applied to the two phases of the motor, \( i_A \) and \( i_B \) (A) are the phase currents, \( \omega \) (rad/s) is the angular velocity of the motor, \( \theta \) (rad) is the angular position, and \( \tau_{LF} \) (N m) is the load and friction torque. The parameters are:

- \( R \) (\( \Omega \)), the resistance of a phase winding.
- \( L \) (H), the inductance of a phase winding.
- \( K \) (N m/A or V s), the torque constant, also called the back-emf constant.
- \( J \) (kg m\(^2\)), the rotor inertia.
- \( n_p \), the number of pole pairs.
1.2 Open-loop control

Methods of open-loop control include:

- sinusoidal commutation where the stator voltages are two-phase sinusoidal voltages

\[
\begin{align*}
    v_A &= V_{pk} \cos(\omega_e t) \\
    v_B &= V_{pk} \sin(\omega_e t),
\end{align*}
\]

(34)

with \( \omega_e = 2\pi f_e \) (in rad/s) and \( f_e \) is the frequency (in Hz).

- four-step commutation methods (one-phase-on and two-phase-on) where the sinusoidal voltages are replaced by square wave approximations. The two-phase-on method corresponds to

\[
\begin{align*}
    v_A &= V_{pk} \text{sign}(\cos(\omega_e t)) \\
    v_B &= V_{pk} \text{sign}(\sin(\omega_e t)),
\end{align*}
\]

(35)

where the sign function is such that \( \text{sign}(x) = 1 \) for \( x > 0 \), \( \text{sign}(0) = 0 \), and \( \text{sign}(x) = -1 \) for \( x < 0 \).

The synchronous speed corresponding to the two-phase voltages is

\[
\omega_s = \frac{\omega_e}{n_P}
\]

(36)

In general, the motor speed \( \omega \) will converge to the synchronous speed if the motor stays synchronized.

1.3 Response to a single step

The dynamics of the system (33) linearized around \( n_p \theta = 90^\circ \) (i.e., \( \pi/2 \)), \( \omega = 0 \), \( i_A = 0 \), and \( i_B = I \) satisfy

\[
\frac{d}{dt} \begin{pmatrix}
    \delta i_A \\
    \delta i_B \\
    \delta \omega \\
    \delta \theta
\end{pmatrix} = \begin{pmatrix}
    -R/L & 0 & K/L & 0 \\
    0 & -R/L & 0 & 0 \\
    -K/J & 0 & 0 & -KIn_p/J \\
    0 & 0 & 1 & 0
\end{pmatrix} \begin{pmatrix}
    \delta i_A \\
    \delta i_B \\
    \delta \omega \\
    \delta \theta
\end{pmatrix} + \begin{pmatrix}
    1/L & 0 \\
    0 & 1/L \\
    0 & 0 \\
    0 & 0
\end{pmatrix} \begin{pmatrix}
    \delta v_A \\
    \delta v_B
\end{pmatrix},
\]

(37)

where the load torque is assumed to be zero. Two eigenvalues of the system matrix are usually lightly damped for practical values of motor parameters. For \( \delta i_A = 0 \), the natural motion of the system is described by

\[
\frac{d^2}{dt^2} (\delta \theta) = -\frac{KIn_p}{J} \delta \theta.
\]

(38)
An estimate of the resonant frequency (in rad/s) is

\[ \omega_n = \sqrt{\frac{K I n_p}{J}}, \]  

where \( I = V/R \) and \( V \) is the amplitude of the voltages. This frequency of oscillation is observed when the motor is operated in one-phase-on four-step commutation mode. In two-phase-on, the magnitude of the current vector is replaced by \( I = \sqrt{2} V/R \).

### 1.4 DQ transformation and DQ model

A useful representation of the stepper motor is obtained through the DQ transformation

\[
\begin{align*}
v_d &= \cos(n_p \theta) \ v_A + \sin(n_p \theta) \ v_B \\
v_q &= -\sin(n_p \theta) \ v_A + \cos(n_p \theta) \ v_B,
\end{align*}
\]

whose inverse is given by

\[
\begin{align*}
v_A &= \cos(n_p \theta) \ v_d - \sin(n_p \theta) \ v_q \\
v_B &= \sin(n_p \theta) \ v_d + \cos(n_p \theta) \ v_q.
\end{align*}
\]

In the DQ coordinates, the stepper motor model becomes

\[
\begin{align*}
L \frac{di_d}{dt} &= v_d - Ri_d + n_p \omega Li_q \\
L \frac{di_q}{dt} &= v_q - Ri_q - n_p \omega Li_d - K \omega \\
J \frac{d\omega}{dt} &= Ki_q - \tau_{LF} \\
\frac{d\theta}{dt} &= \omega.
\end{align*}
\]

For negligible inductance \( L \)

\[ v_q = Ri_q + K \omega \]  

and the response from \( v_q \) to \( \omega \) is similar to the response of a voltage-controlled brush DC motor with negligible inductance, namely

\[ J \frac{d\omega}{dt} = -\frac{K^2}{R} \omega + \frac{K}{R} v_q - \tau_{LF}. \]

### 1.5 Field weakening

At low speeds, it is reasonable to set \( v_d = 0 \), resulting in \( i_q \approx v_q/R \). At high speeds, however, the current \( i_q \) rapidly decreases for a given voltage \( v_q \) because of inductive effects and because
of the back-emf $K\omega$. In order to increase the torque available, the current $i_q$ may be maximized for a bounded amplifier voltage by setting a relationship between the DQ voltages, such as

$$v_d = -\frac{n_p\omega L}{R} v_q.$$  \hspace{1cm} (45)

(45) is an optimal solution to torque maximization under voltage constraints called field weakening.

## 2 Experiments

### 2.1 Open-loop response with sinusoidal commutation

The file *Lab5.mdl* contains a simulation of a stepper motor. Fig. 7 shows the Simulink diagram of the simulation. The *Command generator* applies two-phase sinusoidal voltages of constant magnitude and with increasing frequency, so that the motor progressively accelerates.

Run the simulation and plot the speed (in rpm) as a function of time, together with the synchronous speed using (36) (the number of pole pairs of the motor is $n_P = 12$). Observe that the response becomes increasingly oscillatory as the speed increases.

### 2.2 Open-loop response with increasing frequency

Replace the contents of the command generator so that the frequency (variable *freq*) increases linearly from $f_e = 0$ to $f_e = f_{MAX}$, where $f_{MAX}$ is a variable set in the command generator.
The frequency should increase over 20 s, then stay at $f_{MAX}$ for the remainder of the simulation time, which is 40 s. Set the value of the peak voltage to a constant 25 V.

Simulations of the motor show that:

- for $f_{MAX} \leq 54$ Hz, the motor velocity and the angle track their references.
- for $54$ Hz $< f_{MAX} \leq 64$ Hz, the motor velocity oscillates significantly, but the angle continues to track the reference (at least approximately).
- for $f_{MAX} > 64$ Hz, the motor velocity drops and the motor either stops or rotates at a slower speed than the reference.

Verify these claims by plotting the speed together with the synchronous speed (in rpm) for $f_{MAX} = 54$ Hz, 55 Hz, and 65 Hz. Also plot the angular position $\theta$ (in rad) together with the corresponding angular reference $\theta_{REF} = \omega_e t / n_P$ (in rad), where $\omega_e t$ is the variable alpha found inside the Two-phase voltages block. Finally, plot the voltages over a time period corresponding to about three cycles of the sinusoidal commutation sequence, choosing a period at the end of the experiment where the frequency is constant.

### 2.3 Open-loop response with two-phase-on four-step commutation

Replace the two-phase voltages block by a commutation block as shown on the figure below.

An input of the block (labelled com on the figure) should give you the option of implementing two-phase-on four-step commutation in addition to sinusoidal commutation.

![Diagram for the simulation of four-step commutation](image)

For $f_{MAX} = 72$ Hz and $f_{MAX} = 73$ Hz, plot the speed together with the synchronous speed.
(in rpm), and plot the angular position $\theta$ together with the angular reference $\theta_{REF}$. Also plot the voltages over a time period representing two cycles of the four-step commutation sequence.

Observe that the motor exhibits complicated resonant behavior, in addition to loss of synchronism. Resonant peaks are visible in the velocity plots at low and high frequencies. Give the synchronous speeds at which resonances occur in the two-phase-on mode and the corresponding frequencies of the stator voltages (in Hz).

### 2.4 Stepping mode

Replace the linearly increasing frequency command by a fixed frequency of 6 Hz and let the simulation time be 0.15 s. Plot the angular position of the motor (in rad) for the two-phase-on mode. Observe the oscillatory response in stepping mode, and verify that the equilibrium positions are at $n_P \theta = 45^\circ$, $135^\circ$, ... Verify that $n_P = 12$ and note that the first step is smaller because the motor is initialized at the zero position.

Measure the magnitude of a step from the plot, and compare to the expected value of $\pi/2n_P$ using $n_P = 12$. From the response, estimate the natural frequency of oscillations. Compare the value of the frequency to the one predicted by

$$f_n = \frac{1}{2\pi} \sqrt{\frac{K I n_p}{J}}, \quad (46)$$

using the parameters found inside the simulation model and $I = \sqrt{2}V/R$.

### 2.5 Open-loop DQ control

Replace the commutation block in the simulation by a DQ control block with $v_d$ and $v_q$ as inputs, as shown in the figure below. Modify the command generator to apply voltages $v_d = 0$ and $v_q = V_{pk}$, where $V_{pk}$ increases linearly from 0 to 25 V over 20 s, then stays at 25 V. Simulate the responses for 30 s and plot the speed in rpm.

Next, replace $v_d$ by the value specified by the field weakening formula, and adjust $v_q$ such that $v_d^2 + v_q^2 = V_{pk}^2$. Plot the speed in rpm and compare the maximum speed reached with both options of DQ control, and compare the results to those obtained with the sinusoidal and four-step commutation modes. Overall, the results highlight the considerable differences of behavior observed with various open-loop control methods.

### 3 Report at a glance

Be sure to include:

- Plot of open-loop speed response and synchronous speed (in rpm) with increasing steps of frequency.
Figure 9: Diagram for the simulation of open-loop DQ control

- Plot of open-loop speed response and synchronous speed (in rpm) with sinusoidal commutation and linearly increasing frequency for the three frequencies $f_{MAX}$ specified. Plots of angular position and reference (in rad) and plot of voltages over about three cycles.
- Plot of open-loop speed response and synchronous speed (in rpm) with two-phase-on commutation and linearly increasing frequency for the two frequencies $f_{MAX}$ specified. Plots of angular position and reference (in rad) and estimate of frequencies for the resonant peaks at low speeds. Plot of voltages over about three cycles.
- Plot of step response with two-phase-on commutation. Verification of the step size and estimation of the frequency of oscillation. Comparison with the formula.
- Plot of velocity (in rpm) with DQ control and $v_d = 0$. Value of the maximum speed achieved.
- Plot of velocity (in rpm) with DQ control and field weakening. Value of the maximum speed achieved.

In addition, the report should include comments and explanations for the work performed and the results observed/produced. Be sure to label the axes of all your plots, and to use line widths and font sizes that provide good readability. Also include units for all of your values.
Lab 6 – Brushless DC motor control
with six-step commutation

Objectives
The objective of the lab is to control the speed of a brushless DC motor using a six-step commutation scheme. The commutation method is first tested, followed by a current control loop, and then by a velocity control loop.

1 Introduction
A standard circuit for the control of a brushless DC motor is the three-phase inverter shown on Fig. 10. A resistor with value $R_{\text{SENSE}}$ is placed at the bottom of the circuit as a means of sensing the motor current, with $I_{\text{SENSE}} = V_{\text{SENSE}} / R_{\text{SENSE}}$.

Transistors are turned on based on the following commutation table. Other transistors are turned off and all transistors are turned off for zero torque command (discontinuous conduction...
mode).

<table>
<thead>
<tr>
<th>Sector</th>
<th>$n_p\theta$</th>
<th>$\tau_M &gt; 0$</th>
<th>$\tau_M &lt; 0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>$-30^\circ \rightarrow 30^\circ$</td>
<td>T2, T6</td>
<td>T3, T5</td>
</tr>
<tr>
<td>S2</td>
<td>$30^\circ \rightarrow 90^\circ$</td>
<td>T2, T4</td>
<td>T1, T5</td>
</tr>
<tr>
<td>S3</td>
<td>$90^\circ \rightarrow 150^\circ$</td>
<td>T3, T4</td>
<td>T1, T6</td>
</tr>
<tr>
<td>S4</td>
<td>$150^\circ \rightarrow 210^\circ$</td>
<td>T3, T5</td>
<td>T2, T6</td>
</tr>
<tr>
<td>S5</td>
<td>$210^\circ \rightarrow 270^\circ$</td>
<td>T1, T5</td>
<td>T2, T4</td>
</tr>
<tr>
<td>S6</td>
<td>$270^\circ \rightarrow 330^\circ$</td>
<td>T1, T6</td>
<td>T3, T4</td>
</tr>
</tbody>
</table>

## 2 Experiments

### 2.1 Open-loop stepping

The file *Lab6.mdl* contains a simulation of the brushless DC motor. Fig. 11 shows the diagram of the simulation.

Figure 11: Diagram of the simulation of a brushless DC motor with six-step model

The inputs and outputs of the BLDC model are:

- $vol$, the DC supply voltage (in V).
• $t_1$ to $t_6$, the six transistor commands (0 for off, 1 for on).
• $th$, the position of the motor (in rad).
• $om$, the velocity of the motor (in rad/s).
• $v_1$, $v_2$, $v_3$, the line voltages applied to the motor windings (in V).
• $v_{sense}$, the voltage on the sensing resistor (in V).
• $ha$, $hb$, $hc$, the Hall effect sensor outputs (0 for off, 1 for on).

The simulation also includes a block that implements the six steps (or sectors) of the commutation logic and a command generator that moves from step to step twice across the table. Run the simulation and observe the response of the motor, which is reminiscent of the response of the stepper motor.

2.2 Quadrature control using six-step commutation

Modify the open-loop commutation block to implement six-step commutation using the Hall effect sensors. Fig. 12 shows the diagram of a possible implementation.

![Figure 12: Implementation of six-step commutation](image)

Figure 12: Implementation of six-step commutation
The decoding of the Hall effect sensors can be performed using the table below.

<table>
<thead>
<tr>
<th>$h_A$</th>
<th>$h_B$</th>
<th>$h_C$</th>
<th>$\text{dir} \geq 0$</th>
<th>$\text{dir} &lt; 0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>$b_1$</td>
<td>$c_1$</td>
<td>$T_2, T_6$</td>
<td>$T_3, T_5$</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$b_2$</td>
<td>$c_2$</td>
<td>$T_2, T_4$</td>
<td>$T_1, T_5$</td>
</tr>
<tr>
<td>$a_3$</td>
<td>$b_3$</td>
<td>$c_3$</td>
<td>$T_3, T_4$</td>
<td>$T_1, T_6$</td>
</tr>
<tr>
<td>$a_4$</td>
<td>$b_4$</td>
<td>$c_4$</td>
<td>$T_3, T_5$</td>
<td>$T_2, T_6$</td>
</tr>
<tr>
<td>$a_5$</td>
<td>$b_5$</td>
<td>$c_5$</td>
<td>$T_1, T_5$</td>
<td>$T_2, T_4$</td>
</tr>
<tr>
<td>$a_6$</td>
<td>$b_6$</td>
<td>$c_6$</td>
<td>$T_1, T_6$</td>
<td>$T_3, T_4$</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

The modified commutation block should have as inputs the three Hall effect sensors, a variable $\text{dir}$ defining the sign of the torque (positive torque for $\text{dir} = 1$, negative torque otherwise), and a variable $\text{on}$ defining whether to turn on the transistors at all (all transistors are off for $\text{on} = 0$). Blocks should also be added in Simulink to process the voltage from the 0.5 Ω sensing resistor as follows:

- scale the voltage measured on the resistor to read a value in A.
- take the absolute value of the current.
- filter the signal with a first-order filter having unity gain and a pole at $s = -5000 \text{ rad/s}$.

Finally, generate commands so that: $\text{dir} = 1$ with $\text{on} = 1$ (for 0.03 s), $\text{dir} = 1$ with $\text{on} = 0$ (for 0.03 s), $\text{dir} = -1$ with $\text{on} = 1$ (for 0.03 s), $\text{dir} = -1$ with $\text{on} = 0$ (for 0.02 s). Let the DC supply voltage be 6 V.

Run the simulation for 0.11 s. Plot the transient responses of the velocity and of the filtered current. Next, run another simulation with $\text{dir} = 1$ and $\text{on} = 1$ for the whole period, and plot the responses of the velocity and of the filtered current, as well as the variables $t_1$ to $t_6$, $h_a$, $h_b$, $h_c$, and $v_1$, $v_2$, $v_3$. Keep the presentation compact without sacrificing clarity by plotting 2 to 3 variables in a row using the function $\text{subplot}$ in Matlab. Also plot the complementary transistors (e.g., $t_1$ and $t_4$) on the same graph. Comment on the responses, in particular the different segments of the voltages $v_1$, $v_2$, $v_3$.

### 2.3 Current control

Having completed the six-step commutation block, augment the controller to achieve current regulation. Fig. 13 shows the block diagram of a possible implementation.

The sign of the reference current determines the $\text{dir}$ variable of the six-step commutation block (a positive reference current implies a positive torque command, and a negative reference current implies a negative torque command). The current loop then compares the absolute value of the current reference to the filtered measurement of the ground current (also taken in absolute value). The error drives the on/off switch of the commutation block through a hysteresis controller. The block, called $\text{Relay}$, can be found in the Simulink library. Set its
Figure 13: Implementation of current control
parameters so that the switch on point is 0.001, the switch off point is -0.001, the output when on is 1, and the output when off is 0. Raise the supply voltage to 12 V.

Test the current controller with a reference $I_{REF} = 0 \text{ A}, 0.5 \text{ A}, 1 \text{ A}, 0.5 \text{ A}, 0 \text{ A}, -0.5 \text{ A}, -1 \text{ A}, -0.5 \text{ A}$ (each for 0.01 s), and 0 A (for 0.03 s). Simulate the responses for 0.11 s. Plot the velocity and the filtered current as functions of time. Comment on the responses, and observe that the acceleration is approximately proportional to the current, so that a model

$$\omega(s) \frac{k_0}{I_{REF}(s)} = \frac{1}{s}$$

(47)

can be assumed for control design. Give the value of $k_0$ estimated from the responses with its units.

### 2.4 Velocity control

Based on the result of the previous section, a proportional control law with gain $k_P$ for the velocity using $I_{REF}$ as a control input yields a stable first-order system with pole at $s = -k_0k_P$. Fig. 14 shows the diagram of a possible implementation.

Compute the gain $k_P$ needed to obtain in a response with a time constant equal to 3 ms. Implement the proportional velocity control loop in the simulation and test it with a reference velocity equal to 0 rad/s for 0.01 s, followed by 40 rad/s, 80 rad/s, 40 rad/s, 0 rad/s, -40 rad/s (each for 0.02 s), followed by 0 rad/s (for 0.03 s). Run the simulation for 0.14 s and plot the velocity and the filtered current as functions of time. Comment on the responses.

### 3 Report at a glance

Be sure to include:

- Six-step commutation
  - Plot of velocity and filtered current for varying commands.
  - Plot of velocity and filtered current for fixed commands.
  - Plot of transistor commands, Hall effect sensors, and line voltages.
- Current control
  - Plot of velocity and filtered current for the profile with varying current.
  - Computation of the constant $k_0$.
- Velocity control
  - Computation of the proportional gain.
  - Plot of velocity and filtered current for the profile with varying velocity.

In addition, the report should include comments and explanations for the work performed and the results observed/produced. Be sure to label the axes of all your plots, and to use line widths and font sizes that provide good readability. Also include units for all of your values.
Figure 14: Implementation of velocity control
Lab 7 – Position control of a brushless DC motor with sinusoidal commutation

Objectives

The objective of this lab is to implement a position control law for a brushless DC motor. A three-phase DQ transformation is used. An inner PI control loop regulates the currents, and an outer PID control loop regulates the position. Modifications are applied to the control laws to ensure good performance while enforcing limits on the voltages, currents, and velocity.

1 Introduction

The model of a $Y$-connected three-phase brushless DC motor was found to be given by

\[
\begin{align*}
L \frac{di_A}{dt} &= v_A - Ri_A + K \omega \sin(n_P \theta) \\
L \frac{di_B}{dt} &= v_B - Ri_B + K \omega \sin(n_P \theta - 120^\circ) \\
L \frac{di_C}{dt} &= v_C - Ri_C + K \omega \sin(n_P \theta - 240^\circ) \\
J \frac{d\omega}{dt} &= -K \left(i_A \sin(n_P \theta) + i_B \sin(n_P \theta - 120^\circ) + i_C \sin(n_P \theta - 240^\circ)\right) - \tau_{LF},
\end{align*}
\]

where $v_A, v_B, v_C$ (V) are the line-to-neutral voltages applied to the three phases of the motor, $i_A, i_B, i_C$ (A) are the line currents, $\omega$ (rad/s) is the angular velocity of the motor, $\theta$ (rad) is the angular position, and $\tau_{LF}$ (N m) is the load and friction torque. Note that, to be precise or to implement equations in Matlab, 120° and 240° should be replaced by $2\pi/3$ and $4\pi/3$, respectively.

The parameters are:

- $R$ (Ω), the resistance of a phase winding.
- $L$ (H), the inductance of a phase winding.
- $K$ (N m/A or V s), the torque constant, also called the back-emf constant.
- $J$ (kg m$^2$), the rotor inertia.
- $n_P$, the number of pole pairs.
An equal magnitude 3-phase DQ transformation is given by
\[
\begin{pmatrix}
  i_d \\
  i_q
\end{pmatrix} = \frac{2}{3} \begin{pmatrix}
  \cos(n_P \theta) & \cos(n_P \theta - 120^\circ) & \cos(n_P \theta - 240^\circ) \\
  -\sin(n_P \theta) & -\sin(n_P \theta - 120^\circ) & -\sin(n_P \theta - 240^\circ)
\end{pmatrix}
\begin{pmatrix}
  i_A \\
  i_B \\
  i_C
\end{pmatrix},
\]
(49)

with the inverse
\[
\begin{pmatrix}
  i_A \\
  i_B \\
  i_C
\end{pmatrix} = \begin{pmatrix}
  \cos(n_P \theta) & -\sin(n_P \theta) \\
  \cos(n_P \theta - 120^\circ) & -\sin(n_P \theta - 120^\circ) \\
  \cos(n_P \theta - 240^\circ) & -\sin(n_P \theta - 240^\circ)
\end{pmatrix}
\begin{pmatrix}
  i_d \\
  i_q
\end{pmatrix}.
\]
(50)

The following DQ model results
\[
\begin{align*}
L \frac{di_d}{dt} &= v_d - Ri_d + n_P \omega L i_q \\
L \frac{di_q}{dt} &= v_q - Ri_q - n_P \omega L i_d - K \omega \\
J \frac{d\omega}{dt} &= \frac{3}{2} K i_q - \tau_{LF} \\
\frac{d\theta}{dt} &= \omega.
\end{align*}
\]
(51)

Note that the transfer function from \( i_q \) to \( \theta \) is the standard response
\[
\frac{\theta}{i_q} = \frac{k_0}{s^2}, \quad \text{with} \quad k_0 = \frac{3}{2} \frac{K}{J}.
\]
(52)

DQ control consists in implementing current control loops for the currents \( i_d \) and \( i_q \), and a position/velocity control loop using the current reference \( i_{q,REF} \) assuming tracking by the current \( i_q \). In quadrature control, the reference value for \( i_d \) is set to zero. The overall organization of the components of the controller are shown on Fig. 15.

## 2 Current controller

The control loops for \( i_d \) and \( i_q \) are given by
\[
\begin{align*}
v_d &= k_{PC}(i_{d,REF} - i_d) + k_{IC} \int (i_{d,REF} - i_d)d\tau \\
v_q &= k_{PC}(i_{q,REF} - i_q) + k_{IC} \int (i_{q,REF} - i_q)d\tau.
\end{align*}
\]
(53)

For the design, the inductive and back-emf terms on the right-hand side of the first two rows of the DQ model (52) are neglected (i.e., the terms are treated as disturbances) Then, the response of \( i_d \) in the Laplace domain is given by
\[
L s i_d + R i_d = v_d = \left(k_{PC} + \frac{k_{IC}}{s}\right)(i_{d,REF} - i_d),
\]
(54)
and

\[ i_d = \frac{k_{PC}}{Ls + R} \frac{1}{s} (i_{d,REF} - i_d). \]  

Choosing

[\text{\[ k_{IC} = \frac{R}{L} k_{PC}, \] (56)}]

it follows that

\[ i_d = \frac{k_{PC}}{Ls} (i_{d,REF} - i_d). \] \[(57)\]

The transfer function from \( i_{d,REF} \) to \( i_d \) is then

\[ \frac{i_d}{i_{d,REF}} = \frac{k_{PC}}{Ls + k_{PC}}. \]  

(58)

The same transfer function applies to the current \( i_q \). The responses of the two current control loops correspond to the same first-order system with a DC gain equal to one and with a pole at \( s = -k_{PC}/L \). The pole can be placed at \( s = -a_{dc} \) for

\[ k_{PC} = L a_{dc}, \quad k_{IC} = R a_{dc}. \] \[(59)\]

### 3 Position controller

Having developed the current control loops, a outer control loop can added for position control, using the same tools as were applied for brush DC motors. A PID controller with quadrature
control consists in
\[
\begin{align*}
    i_d,_{REF} & = 0 \\
    i_q,_{REF} & = k_P(\theta_{REF} - \theta) + k_I \int (\theta_{REF} - \theta) d\tau - k_D \omega.
\end{align*}
\] (60)

Neglecting friction and assuming that the currents track the references, (52) and (60) give the closed-loop transfer function
\[
\frac{\theta}{\theta_{REF}} = \frac{k_P k_0 s + k_I k_0}{s^3 + k_D k_0 s^2 + k_P k_0 s + k_I k_0}.
\] (61)

A possible design choice is to place the three poles at some desired value \( s = -a_d \). This choice corresponds to
\[
\begin{align*}
    k_D & = \frac{3 a_d}{k_0}, & k_P & = \frac{3 a_d^2}{k_0}, & k_I & = \frac{a_d^3}{k_0}.
\end{align*}
\] (62)

4 Improvements

Several modifications can be applied to the control law to improve the performance:

• a feedforward gain can be inserted in the position controller so that
\[
k_P(\theta_{REF} - \theta) \rightarrow k_P(k_F \theta_{REF} - \theta).
\] (63)

This modification shifts the zero of the transfer function without moving the closed-loop poles. A possible overshoot in the response can be avoided. If the poles are placed at \( s = -a_d \), the zero is located as
\[
z = -\frac{k_I}{k_F k_P} = -\frac{a_d}{3k_F}.
\] (64)

• the PID position controller can be represented as the combination of a PI position controller and a proportional-integral velocity controller. Specifically, the reference for \( i_q \) in (60) is rewritten as
\[
\begin{align*}
    i_q,_{REF} & = k_D (\omega_{REF} - \omega) \\
    \omega_{REF} & = \frac{k_P}{k_D} (k_F \theta_{REF} - \theta) + \frac{k_I}{k_D} \int (\theta_{REF} - \theta) d\tau.
\end{align*}
\] (65)

This implementation makes it possible to limit the speed reference by capping \( \omega_{REF} \) to \( \pm \omega_{MAX} \).

5 Command limiting

Limiting of the commands is useful to satisfy hardware constraints. Specifically,
• **voltage limiting:** in steady-state,
\[
V_{pk} = \sqrt{v_d^2 + v_q^2},
\]
where \( V_{pk} \) is the peak line-neutral voltage. If \( V_{pk} > V_{MAX} \), \( v_d \) and \( v_q \) are multiplied by \( V_{MAX}/V_{pk} \). When limiting occurs, integration also should be stopped in (53).

• **velocity limiting:** \( \omega_{REF} \) should be capped to \( \pm \omega_{MAX} \).

• **current limiting:** in steady-state
\[
I_{pk} = \sqrt{i_d^2 + i_q^2} = i_q,
\]
where \( I_{pk} \) is the peak line current. The variable \( i_q,REF \) should be capped to \( \pm I_{MAX} \). Integration should also be stopped in (65) when the magnitudes of \( i_q,REF \) or \( \omega_{REF} \) reach their limit.

6 Experiments

6.1 Open-loop response

The file Lab7.mdl contains a simulation of a brushless DC motor. Fig. 15 shows the diagram of the simulation. The Command generator applies three-phase voltages of increasing magnitude and frequency. Stepwise changes are applied so that the motor progressively accelerates, but the response becomes increasingly oscillatory. Run the simulation and plot the speed (in rpm) as a function of time, together with the synchronous speed (the number of pole pairs is \( n_P = 2 \)).

6.2 Design of the position controller

Find the parameters of the motor in the simulation model, and deduce:

• the PI gains \( k_{PC} \) and \( k_{IC} \) of the current control loop with \( \omega_{dc} = 2\pi \times 1000 \) rad/s.

• the PIDF gains \( k_P, k_I, k_D, k_F \) of the position control loop such that \( \omega_d = 2\pi \times 20 \) rad/s and one closed-loop pole is cancelled by a zero.

Verify that values of your controller parameters rounded to the first significant digit are \( k_{PC} = 10 \), \( k_{IC} = 6000 \), \( k_P = 10 \), \( k_I = 600 \), \( k_D = 0.1 \), and \( k_F = 0.3 \).

6.3 Testing of the position controller

A possible implementation of the simulation is shown on Fig. 17. Fig. 18 shows the contents of the Position controller block. The inner block named DQ_control is an m-file that contains all the equations of the controller. The variable \( ie \) is the integrator for the velocity controller, while the variables \( ieid \) and \( ieiq \) are the integrators for the current controllers.
Figure 16: Diagram of the simulation of a BLDC motor

Figure 17: Diagram for the simulation of DQ control
Figure 18: Contents of the position controller block
In the code, obtain \( i_C \) from \( i_C = -i_A - i_B \). Let \( \omega_{MAX} \) be 4000 rpm (convert to rad/s), \( V_{MAX} = 30 \) V, \( I_{MAX} = 7 \) A. Test the position control system with a reference input having the following profile.

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>( \theta_{REF} ) (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-1</td>
<td>0</td>
</tr>
<tr>
<td>1-2</td>
<td>10</td>
</tr>
<tr>
<td>2-3</td>
<td>0</td>
</tr>
<tr>
<td>3-5</td>
<td>90</td>
</tr>
<tr>
<td>5-7</td>
<td>0</td>
</tr>
<tr>
<td>7-10</td>
<td>3600</td>
</tr>
<tr>
<td>10-12</td>
<td>0</td>
</tr>
</tbody>
</table>

Plot the position and reference in degrees as functions of time from 0 to 6 s and separately from 6 to 12 s. Over the full period from 0 to 12 s, plot the line current \( i_A \) and the line current \( i_B \), the line-to-line voltage \( v_A - v_B \), the line-to-line voltage \( v_B - v_C \), and the speed (in rpm). In these plots, include horizontal lines delineating the positive and negative limits of voltage, current, and speed on the plots. For the voltages, the line-to-neutral limit \( V_{MAX} \) should be converted to an equivalent line-to-line limit.

### 7 Report at a glance

Be sure to include:

- Plot of the open-loop speed response with the synchronous speed.
- Values of the controller parameters.
- Plot of \( \theta \) and \( \theta_{REF} \) from 0 to 6 s.
- Plot of \( \theta \) and \( \theta_{REF} \) from 6 to 12 s.
- Plots of \( i_A \) and \( i_B \) with their limits.
- Plots of \( v_A - v_B \) and \( v_B - v_C \) with their limits.
- Plot of \( \omega \) (in rpm) with its limits.

In addition, the report should include comments and explanations for the work performed and the results observed/produced. Be sure to label the axes of all your plots, and to use line widths and font sizes that provide good readability. Also include units for all of your values.
Lab 8 – Field-oriented control of induction motors

Objectives
The objective of the lab is to implement a field-oriented control method for the tracking of speed references by a squirrel-cage induction motor. For simplicity, current command is assumed. The current in the direct axis maintains a desired rotor flux magnitude, while the current in the quadrature axis controls the torque. Commands are limited to satisfy constraints.

1 Introduction
The model of a two-phase squirrel-cage induction motor with current command expressed in AB variables and using rotor fluxes is given by

\[
\begin{align*}
\frac{d\psi_{RA}}{dt} &= \frac{1}{T_R} \psi_{RA} + \frac{M}{T_R} i_{SA} - n_P \omega \psi_{RB} \\
\frac{d\psi_{RB}}{dt} &= \frac{1}{T_R} \psi_{RB} + \frac{M}{T_R} i_{SB} + n_P \omega \psi_{RA} \\
J \frac{d\omega}{dt} &= \frac{n_P M}{L_R} (i_{SB} \psi_{RA} - i_{SA} \psi_{RB}) - \tau_{LF},
\end{align*}
\]

where \(i_{SA}, i_{SB}\) (A) are the stator currents, \(\psi_{RA}, \psi_{RB}\) (Wb) are the components of the rotor flux, \(\omega\) (rad/s) is the angular velocity of the motor, and \(\tau_{LF}\) (N m) is the load and friction torque. The parameters are:

- \(T_R\) (s), the rotor time constant.
- \(L_R\) (H), the self-inductance of a rotor winding.
- \(R_R\) (Ω), the resistance of a rotor winding.
- \(M\) (H), the mutual inductance between a stator and a rotor winding when aligned.
- \(J\) (kg m\(^2\)), the rotor inertia.
- \(n_P\), the number of pole pairs.
A field-oriented control scheme is obtained through a DQ transformation

\[
\begin{pmatrix}
    i_{sd} \\
    i_{sq}
\end{pmatrix} = \begin{pmatrix}
    \cos(\rho) & \sin(\rho) \\
    -\sin(\rho) & \cos(\rho)
\end{pmatrix} \begin{pmatrix}
    i_{SA} \\
    i_{SB}
\end{pmatrix},
\]

with inverse

\[
\begin{pmatrix}
    i_{SA} \\
    i_{SB}
\end{pmatrix} = \begin{pmatrix}
    \cos(\rho) & -\sin(\rho) \\
    \sin(\rho) & \cos(\rho)
\end{pmatrix} \begin{pmatrix}
    i_{sd} \\
    i_{sq}
\end{pmatrix}.
\]

The angle \( \rho \) is given by

\[
\rho = \angle (\psi_{RA} + j\psi_{RB}).
\]

Then, the DQ rotor fluxes become

\[
\psi_d = \sqrt{\psi^2_{RA} + \psi^2_{RB}} = \psi, \quad \psi_q = 0.
\]

and the equations of the motor in the DQ variables are

\[
\begin{align*}
\frac{d\psi}{dt} & = -\frac{1}{T_R} \psi + \frac{M}{T_R} i_{sd} \\
\frac{d\rho}{dt} & = n_P \omega + \frac{M}{T_R} i_{sq} \\
\frac{d\omega}{dt} & = \frac{n_P M}{J L_R} \psi i_{sq} - \frac{\tau_{LF}}{J}.
\end{align*}
\]

Note that the use of the model requires that \( \psi \) be bounded away from 0. A current \( i_{sd} > 0 \) must be applied and \( \psi = M i_{sd} > 0 \) in steady-state.

2 Field-oriented control

The transfer function from \( i_{sq} \) to \( \omega \) is

\[
\frac{\omega}{i_{sq}} = \frac{k_0}{s}, \quad \text{with} \quad k_0 = \frac{n_P M}{J L_R} \psi.
\]

Assume that a PI controller is used to regulate the velocity

\[
i_{sq} = k_P (\omega_{REF} - \omega) + k_I \int (\omega_{REF} - \omega) d\tau,
\]

where \( \omega_{REF} \) (rad/s) is the speed reference and \( k_P, k_I \) are proportional and integral gains. The closed-loop transfer function from \( \omega_{REF} \) to \( \omega \) is

\[
\frac{\omega(s)}{\omega_{REF}(s)} = \frac{sk_0 k_P + k_0 k_I}{s^2 + sk_0 k_P + k_0 k_I}.
\]
The two closed-loop poles are located at $s = -a_d$ if

$$k_P = \frac{2a_d}{k_0}, \quad k_I = \frac{a_d^2}{k_0}. \quad (77)$$

Note that the controller gains are automatically adjusted as functions of the flux level, since $k_0$ is proportional to $\psi$.

Because the rotor flux variables are not easily measured, they are obtained through some type of estimator, in general. A simple open-loop observer consists in duplicating the differential equations

$$\frac{d\psi}{dt} = -\frac{1}{T_R} \psi + \frac{M}{T_R} i_{sd}$$

$$\frac{d\rho}{dt} = n_p \omega + \frac{M}{T_R} i_{sq} \psi. \quad (78)$$

The differential equation for $\rho$ is only integrated when $\psi \geq \psi_{\text{min}}$, for some $\psi_{\text{min}} > 0$. When $\psi < \psi_{\text{min}}$, $i_{sq}$ is set to zero and the differential equation for $\rho$ is replaced by

$$\frac{d\rho}{dt} = n_p \omega. \quad (79)$$

### 3 Improvements

Several modifications can be applied to the control law to improve the performance:

- to ensure that $|i_{SA}| \leq I_{\text{MAX}}$ and $|i_{SB}| \leq I_{\text{MAX}}$, the DQ currents are limited so that

$$i_{sd}^2 + i_{sq}^2 \leq I_{\text{MAX}}^2. \quad (80)$$

Given that $i_{sd}$ is independently specified to achieve a certain flux level, the following limit is applied

$$|i_{sq}| \leq \sqrt{I_{\text{MAX}}^2 - i_{sd}^2}. \quad (81)$$

- a feedforward gain is inserted in the velocity controller so that (75) is replaced by

$$i_{sq} = k_P (k_F \omega_{\text{REF}} - \omega) + k_I \int (\omega_{\text{REF}} - \omega) d\tau. \quad (82)$$

This modification shifts the zero of the closed-loop transfer function without moving the poles. A possible overshoot in the response can be avoided. Integration is stopped in (82) when the current $i_{sq}$ is capped to satisfy (81), or when (79) is applied.
4 Three-phase motor

A three-phase motor can be controlled by applying $3 - 2$ and $2 - 3$ transformations. If an equal magnitude transformation is used, the peak magnitudes of the three-phase and two-phase variables are the same in steady-state. The only adjustment to be made is a factor of $3/2$ in the mechanical equation, so that $k_0$ is replaced by

$$k_0 = \frac{3}{2} \frac{n_P M}{J L_R} \psi.$$

(83)

5 Experiments

(a) The file *Lab8.mdl* contains the two-phase equivalent model of a three-phase induction motor under current command. Fig. 19 shows the diagram of the simulation. A block generates two-phase currents of constant magnitude and frequency. Note that the motor is a 50 Hz motor with two poles pairs, so that the synchronous speed is 1500 rpm.

![Diagram of the simulation of a squirrel-cage induction motor under current command](image)

Figure 19: Diagram of the simulation of a squirrel-cage induction motor under current command

Run the simulation and plot the speed as a function of time. Also plot the rotor flux magnitude $\psi$ as a function of time using (72) and $\psi_{RA}$ and $\psi_{RB}$ from the simulation model. Observe that the speed and the flux increase slowly at first, but the rate grows when the motor gets close the synchronous speed. Speed oscillations are observed when the synchronous speed.
Implement the field-oriented control algorithm in the simulation. The figure below shows the Simulink diagram of a possible implementation.

Figure 20: Implementation of field-oriented control

Find the motor parameters from the simulation model and compute $k_0$ from (83). For the controller, let $a_d = 50 \text{ rad/s}$, $k_F = 2/3$, $I_{MAX} = 8.6 \text{ A}$, and

$$\psi_{min} = \frac{1}{4} M i_{sd}.$$  \hfill (84)

For the input signals, let $i_{sd} = 2.5 \text{ A}$ at first, and increase $i_{sd}$ to $3 \text{ A}$ after $3 \text{ s}$. Let the speed reference be zero for $1 \text{ s}$, in order to let the flux build up. After $1 \text{ s}$, increase the speed reference to 1500 rpm (convert to rad/s). Let the load torque be zero at first and increase to $15.1 \text{ N m}$ after $2 \text{ s}$.

Plot as functions of time the speed in rpm, the rotor flux magnitude, and the peak stator current

$$I_{pk} = \sqrt{i_{SA}^2 + i_{SB}^2}.$$  \hfill (85)

The speed response should be consistent with the choice of poles and zero, with a small dip when the load torque is suddenly applied. The stator current should reach the limit during the acceleration, return to the value of $i_{sd}$ when the speed reference is reached, and increase again when the load torque is applied. The change in flux reference (through $i_{sd}$) does not impact the speed response. Even though $i_{sd}$ increases, the overall stator current decreases because a smaller current $i_{sq}$ is needed to achieve the desired torque.
Impressive results are obtained, although conditions are idealized. The control parameters are chosen using the exact values of the simulation model. Also, the induction motor model does not include non-ideal effects such as magnetic saturation or rotor resistance variations with heating. Although the results may not be as good in practice, the simulation illustrates the elegant solution provided by field-oriented control.

6 Report at a glance

Be sure to include:

- Plots of speed and rotor flux magnitude for currents of fixed magnitude and frequency.
- Plots of speed, rotor flux magnitude, and peak stator current for the field-oriented controller.

In addition, the report should include comments and explanations for the work performed and the results observed/produced. Be sure to label the axes of all your plots, and to use line widths and font sizes that provide good readability. Also include units on all of your values.