MF2007: Dynamic & Motion Control
Workshop 1

Kilian DEMEULEMEESTER, Jeremy POUECH
{kiliande,pouech}@kth.se

March 27, 2017

Contents

Abstract

This paper summarizes our results of workshop 1, about parameter identification and control of a DC-motor. The first part contains informations about parameter identification for two different models of the motor: linear and non linear frictions models. The second part deals with controling the DC-motor, in velocity and in position.

Parameter identification

Level 1

In this section, we need to design the linear friction d_1 in the friction model $M_f = d_1\dot{\varphi}$ and the total inertia J.

In order to do so, we add our motor model to ControlDesk. We then fit the theoretical curve to the real one (using sliders to modify d_1 and J).
The fitted curves for motor 1 and motor 2 are on Figure 1.
We have the following values:

	Motor 1	Motor 2
J	$4.2e-7$	$4.6e-6$
d_1	$4.8e-6$	$5.8e-6$

Level 2

In this section, we identify the parameters for a second model with a nonlinear friction model $M_f = d_2\varphi + F_c\,\mathrm{sgn}(\varphi)$, which includes static friction, using the Karnop friction model.
To do so, we add our motor model to ControlDesk. We the fit the theoretical curve to the real one (using sliders to modify d_2 and F_c).
The fitted curves for motor 1 and motor 2 are on Figure 2 (from a two steps with different amplitudes). Figure 3 shows the velocity signal with a sinewave input with low amplitude.
We have the following values:

	Motor 1	Motor 2
d_2	$1.0e-8$	$1.0e-8$
F_c	$1.0e-4$	$1.0e-4$

Considering the non linear friction model improves the quality of our model.

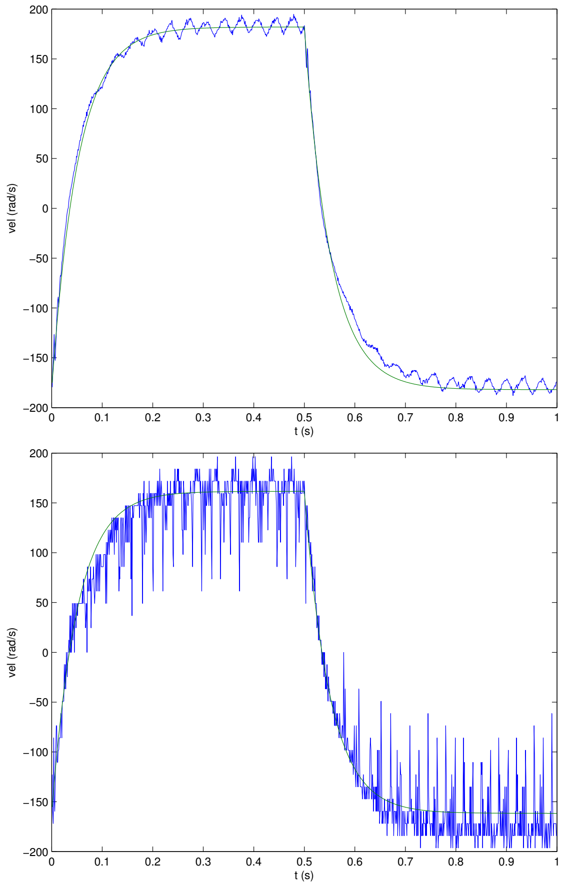

Figure 1: Fitted curves for motor 1 & 2 – green
Real curves for motor 1 & 2 – blue

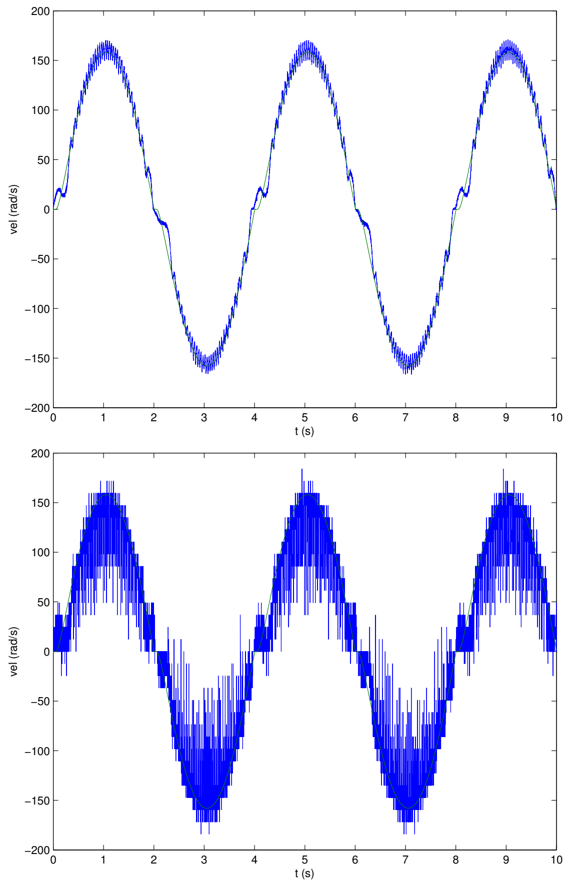

Figure 2: Fitted curves for motor 1 & 2 – green
Real curves for motor 1 & 2 – blue

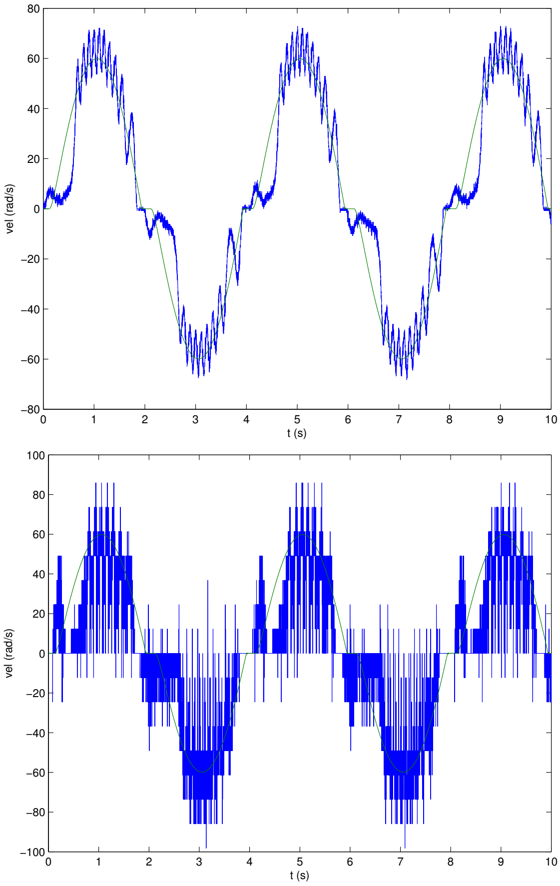

Figure 3: Motor response for a sinewave of low amplitude
Theoretical curves for motor 1 & 2 – green
Real curves for motor 1 & 2 – blue

Controlling the motor

Velocity control

Level 1

We need to design a discrete time PI-controller which can be used for the following two reference signals.

1. Step response to 300 rad/s :
   - As fast as possible;
   - No overshoot;
   - Maximum error at steady state 5 rad/s.
2. Sine wave reference signal:
   $vel_{ref} = 300\sin(0.2t)$ rad/s.

In this subsection, $T_s = 1$ ms.
In order to design the PI-controller, we use the following method:

- Add a continuous PI-controller to the model: $C_{PI}(p) = K(1 + \frac{1}{T_i p})$;
- Compute the close loop transfer function of the global system;
- Design K, T_i such as there is no overshoot and a fast step response;
- Discretize the controller;
- Redesign K, T_i in order to fulfill the criteria with the discrete controller.

Let the transfer function for the motor be:

$$H_m(p) = \frac{K_m}{1 + \tau_m p}$$

The transfer function for the continuous PI-controller is:

$$C_{PI}(p) = K\frac{1 + T_i p}{T_i p}$$

Thus, the close loop transfer function is:

$$H_{cl}(p) = \frac{C_{PI}(p)H_m(p)}{1 + C_{PI}(p)H_m(p)} \tag{1}$$

Simplifying equation (1) leads to:

$$H_{cl}(p) = \frac{1 + T_i p}{(1 + T_i p)(1 + \frac{\tau_m}{KK_m}p) + (\frac{T_i}{KK_m} - \frac{\tau_m}{KK_m})p} \tag{2}$$

Let

$$T_i = \tau_m \tag{3}$$

This value leads to a *first order transfer function in closed loop*.
We then have the following transfer function :

$$H_{cl}(p) = \frac{1}{1 + \frac{\tau_m}{KK_m}p} \tag{4}$$

Let $\tau_s = \frac{\tau_m}{KK_m}$ be the new time constant of the system.

Remark : It seems that the bigger K is, the smaller τ_s is. However, we will see that due to saturation limitation that we can not pick whatever value we want.

The continuous transfer function is discretize using Tustin method $\left(p \leftrightarrow \frac{2}{T_s}\frac{1-z^{-1}}{1+z^{-1}}\right)$ leads to the following transfer function:

$$\frac{K}{2T_i}\frac{(2T_i + T_s) + (T_s - 2T_i)z^{-1}}{1 - z^{-1}} \tag{5}$$

K has to be tuned now. Using Simulink, we create a model of the system including the discrete PI-controller.
We have the following restriction on U : $U(t) \leq 12$ V.
Thus, if K gets too big, the output of the PI-controller is saturated.
We have the following behavior with K too big:

K too big
$\downarrow$
The PI-controller output is bigger than 12 V.
$\downarrow$
$U(t)$ is saturated
$\downarrow$
The PI-controller input error can not decrease
$\downarrow$
The integral part of the controller grows to infinity (windup effect)
$\downarrow$
Once the motor has almost reach its final value, the saturation disapear and the big integral part leads to an overshoot

In order to avoid these behavior, we use the two following solutions:

- K is tuned using an iterative method (the biggest value without any overshoot)
- The integral part in the PI-controller is saturated ($|U| \leq 7$ V) in order to avoid its growing to infinity in case of a saturation of the output.

This analysis leads to the following understanding : Even if our system is modelized by a first order system which can theoreticaly has a step response as fast as we want, the saturation in the motor input command leads to response velocity limitations.

This behavior is visible on Figure 4: for each motor input, the rising velocity is the same.
Figure 4 and 5 display also that our controller fulfill the criteria for the two reference signals – with a rise time $t_r \leq 50$ ms.
Figure 6 shows the real step response of the motor.
Using Matlab, we find the following parameters:

K	T_i (s)
0.13	0.8

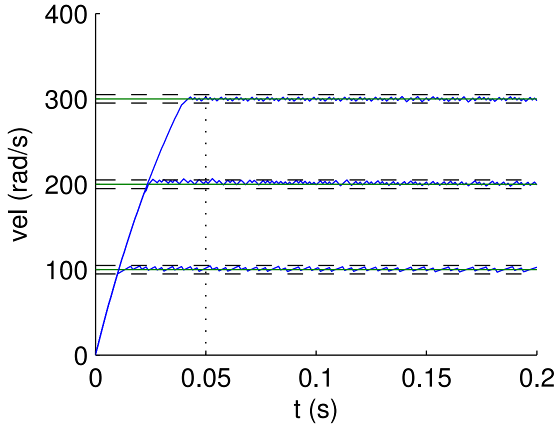

Figure 4: Simulated motor step responses for 3 step input of amplitude (100, 200, 300) rad/s

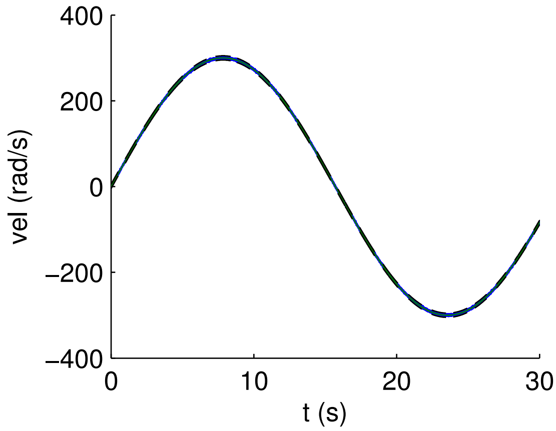

Figure 5: Simulated motor response for the sine wave reference

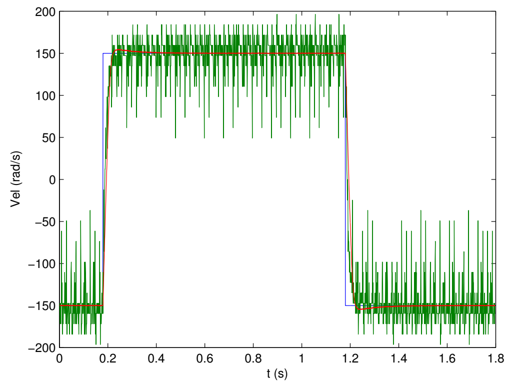

Figure 6: Real motor step response

Position control

In the two following section, we design a discrete controller that controls the position of the motor, with the following specifications:

- Step response to one revolution (2π) with rise time $t_r = 0.07$ s ± 0.005 s.
- Maximum overshoot 2% of the step (2π)
- Steady state error, maximum 0.5%

Level 1

In this subsection, any sampling period can be used. Here, $T_s = 1$ ms.
We designed to different controller: a *proportional controller* additional to the PI-controller design in section and a *phase-lead controller*.
Transforming the temporal criteria of the controller specification to frequency criteria leads to:

- Cutoff frequency: $w_c t_r \approx 3 \Leftrightarrow w_c \approx 42.8$ rad/s
- Phase margin: $D\% \leq 2\% \Leftrightarrow \Delta\Phi \geq 70°$ (see Figure 7 for this criteria)

Moreover, since the position loop include a integrator, the steady state error will be zero.

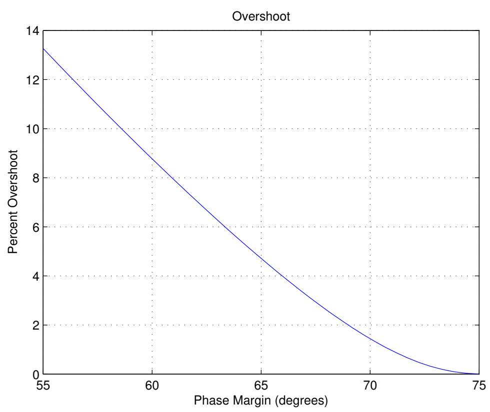

Figure 7: Overshoot as a function of the phase margin

Proportional controller

With the PI-controller from the previous section, the "continuous" close loop transfer function of the system is:

$$H_{cl,vel}(p) = \frac{1}{1 + \frac{\tau_m}{KK_m} p} \tag{6}$$

Therefore, outputting the position leads to the addition of an integrator block (integration of the velocity). The new transfer function is:

$$H_{cl,pos}(p) = \frac{1}{p} \frac{1}{1 + \frac{\tau_m}{KK_m} p} \tag{7}$$

The maximum rise time for the velocity controller verify $t_{r,vel} \leq 5O$ ms $\Leftrightarrow w_c = 125.6$ rad/s. Therefore, in the worst case scenario, the transfer function for the position is:

$$H_{cl,pos}(p) = \frac{1}{p} \frac{1}{1 + \frac{p}{125.6}} \tag{8}$$

Figure 8 shows the bode diagram of this function.

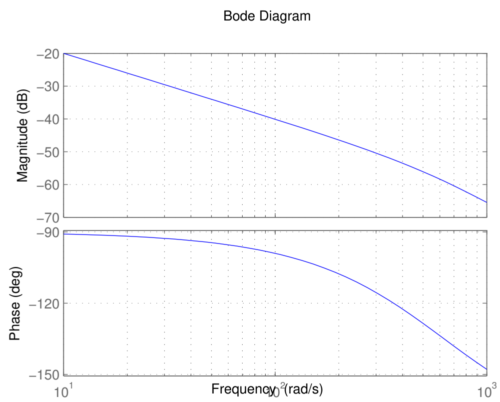

Figure 8: Bode diagram of $H_{cl,pos}$

According to the bode diagram on Figure 8, we have:

- $G_{dB,pos}(\omega = 42.8 \text{ rad/s}) = -32$ dB
- $\Delta\Phi = 86.3°$

Therefore, we only need to translate vertically the gain curves in order to fulfill the criteria.
Using a proportional controller $H_P(p) = K_P$ leads to:

$$G_{dB,cont}(\omega = 42.8 \text{ rad/s}) = 32 \text{ dB} \Leftrightarrow K_P = 10^{\frac{32}{20}} = 40$$

Our theoretical proportional controller is:

$$C_P(s) = 40$$

Figure 9 shows the simulated step response of the system and Figure 10 the real step response. *All of the criteria are met.*

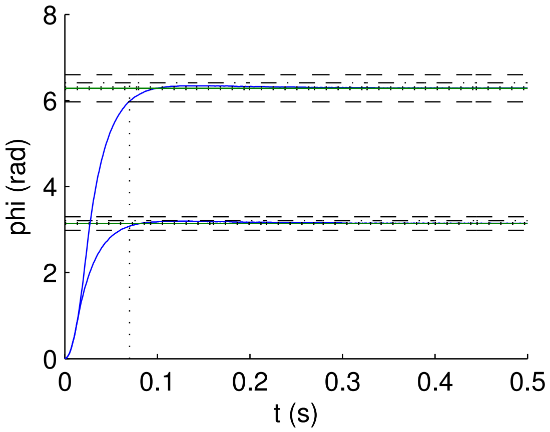

Figure 9: PI-controller – Simulated steps responses of the system controled in position for two differents amplitudes

Phase-lead controller

Whithout any controller, the continuous close loop function of the motor is:

$$H_{m,pos}(s) = \frac{\frac{K_m}{Rd+K_m}}{1+\frac{RJ}{Rd+K_M K_E}s}\frac{1}{s}$$

Figure 11 shows the bode diagram of this transfer function.

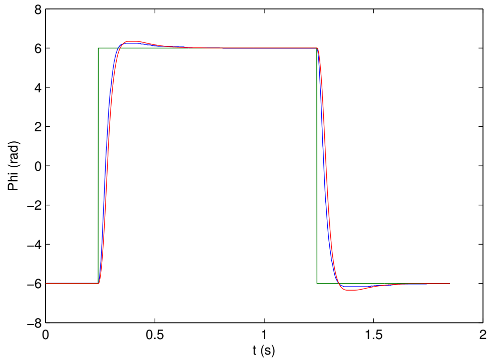

Figure 10: PI-controller – Real step response of the system controled in position

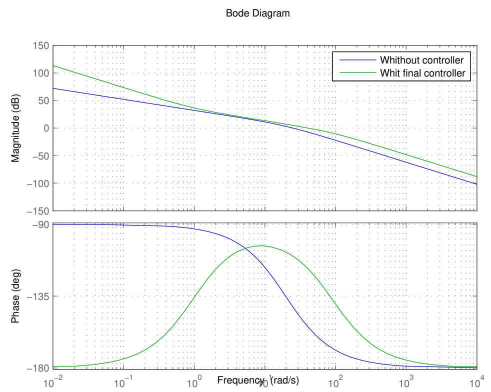

Figure 11: Bode diagram
Blue – Without controller
Green – With the final controller

We want to design a phase-lead controller:

$$C_\Phi(s) = K_\Phi \frac{1+T_\Phi s}{1+aT_\Phi s}, 0 < a < 1$$

We want a phase margin equal to 70° at $\omega_c = 42.8$ rad/s. Therefore, since $\Phi_{H_{m,pos}} = 15°$, we need $\varphi_m = 45°$.
With this kind of controller, a fulfill the following equation:

$$\sin\varphi_m = \frac{1-a}{1+a} \Leftrightarrow a = \frac{1-\sin\varphi_m}{1+\sin\varphi_m} = 0.17$$

T_Φ fulfill the following equation:

$$\frac{1}{T_\Phi\sqrt{a}} = \omega_c \Leftrightarrow T_\Phi = \frac{1}{\omega_c\sqrt{a}} \approx 56 \text{ ms}$$

And K_Φ fulfill the following equation ($\tau_m = \frac{RJ}{Rd+K_M K_E}$):

$$20\log\left(\frac{K}{\sqrt{a}}\right) + |H_{m,pos}(j\omega)|_{\omega=\omega_c} = 0$$

$$\Leftrightarrow K_\Phi = \frac{\sqrt{a}}{\sqrt{1+\tau_m \omega_c}} \approx 0.99$$

Our theoretical phase controller is:

$$C_\Phi(s) = 0.99\frac{1+0.056s}{1+0.0097s}$$

Using this controller, we observed that the steady state error was almost equal to 20%. Indeed, the static error is not integrated. In order to correct that, we add an integral action in our controller. Tuning the values of those leads to the following controller:

a	=	0.2379
T	=	0.0478
K	=	1.1725
Ti	=	1

Thus:

$$C(s) = 0.99\left(\frac{1+0.049s}{1+0.011s}\right)\left(1+\frac{1}{s}\right)$$

Figure 12 shows the theoretical step response of the system and Figure 13 shows the simulated step response (with simulink). Finally, Figure 14 shows the real step response.

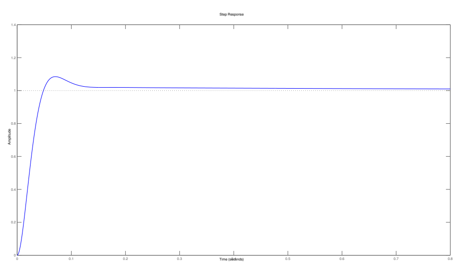

Figure 12: Phase-lead controller – Theoretical step response of the system with the controller

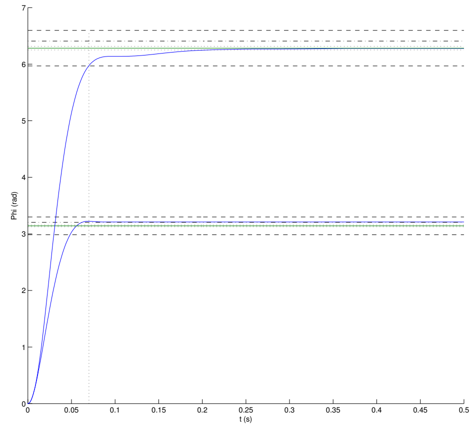

Figure 13: Phase-lead controller – Simulated step response of the system with the controller

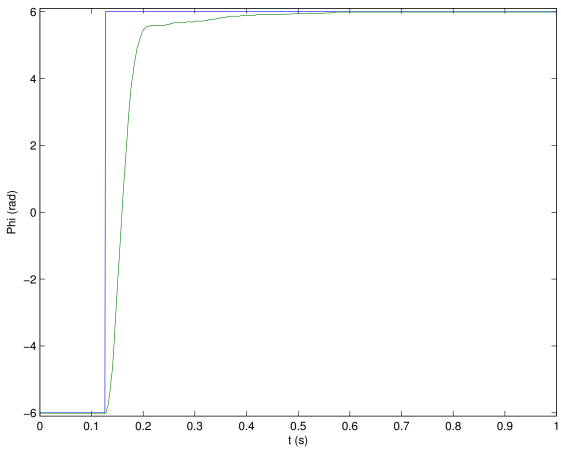

Figure 14: Phase-lead controller – Real step response of the system with the controller