Lecture 6: Experiment 5
EE380 (Control Systems)

Manavaalan Gunasekaran
PhD student
manvaal@iitk.ac.in

Arun Kant Singh
PhD student
arunkant@iitk.ac.in

Ramprasad Potluri
Associate Professor
potluri@iitk.ac.in

Department of Electrical Engineering
Indian Institute of Technology Kanpur

September 09, 2011

Contents

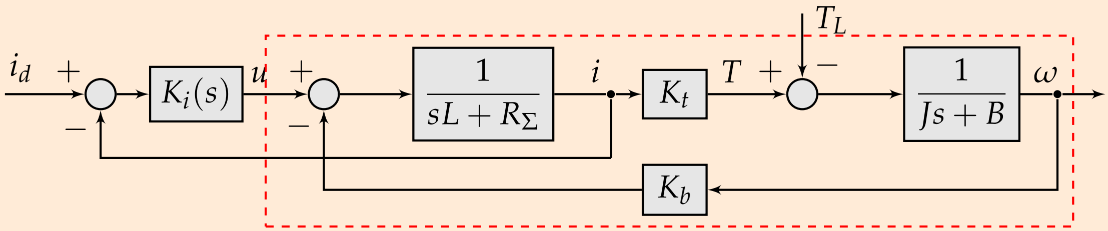
Announcements

- Before doing an experiment, download latest versions of supporting documents from Brihaspati.
- Latest version of program listings are on Brihaspati.
- Turn off power supply to board when not programming dsPIC or taking readings.
- After completion of experiment
  - Shut down PC, FG, PS.
  - Remove PICkit 2 from dsPIC board.

Back
Forward
Close

Outline of the experiment

- Design $K_i(s)$ such that i tracks i_d.



Plant: The part outside dsPIC30F4012

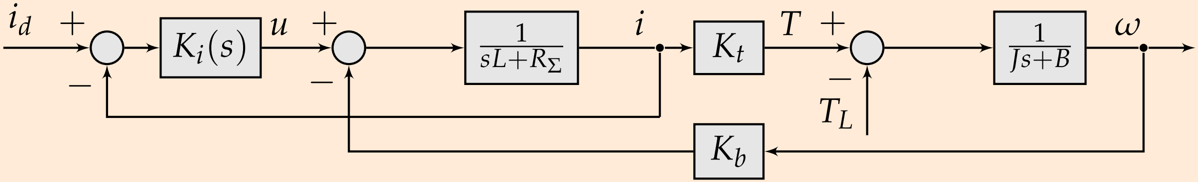
- Use $\widehat{i} \approx i_{\text{sens}}/1.8 - 1/30$. Is this a good approximation?
- Determine a trackable i_d using

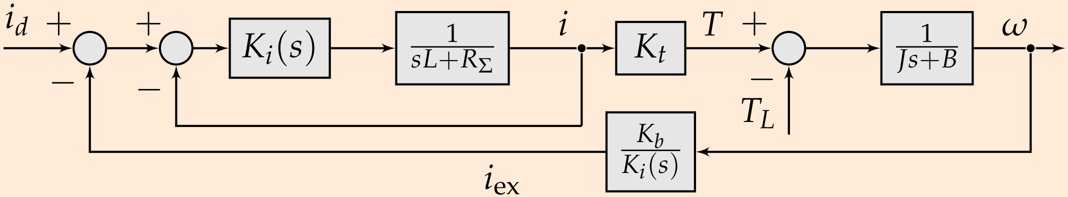
$$\text{fundamental torque equation}\quad J\frac{\mathrm{d}\omega}{\mathrm{d}t} = -B\omega + T - T_L$$

$$\text{voltage equation}\quad V = L\frac{\mathrm{d}i}{\mathrm{d}t} + Ri + E$$

How to design a $K_i(s)$?

- Redraw block diagram



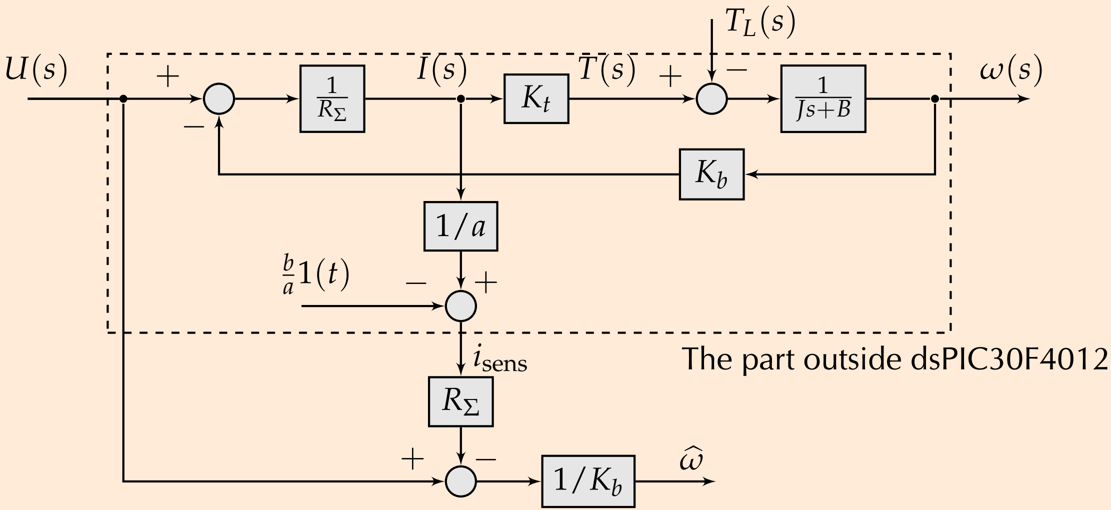


- i tracks $i_d - i_{\mathrm{ex}}$, not i_d alone. i is called well-regulated if it tracks i_d nicely. Nice tracking won't happen while i_{ex} dominates. $\therefore$ choose $K_i(s)$ to supress i_{ex}. Two choices for $K_i(s)$ are P and PI.
- P gives same gain in transient and SS $\Rightarrow$ large demand on u. PI places small demand on u in transient and acceptable demand in SS. Also, $\exists$ current filter to reject noise in i_{sens}. $\Rightarrow$ Choose PI.

Back
Forward
Close

Is $\widehat{i} \approx \frac{i_{\text{sens}}}{1.8} - \frac{1}{30}$ a good approximation?

- From Exp-t 4 saw that don't have an accurate relationship $i = f(i_{\text{sens}})$.
- Need to review the derivation that leads to $\begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} R_\Sigma & 7 - K_b\widehat{\omega}_{ss7} \\ R_\Sigma & 9 - K_b\widehat{\omega}_{ss9} \end{bmatrix}^{-1} \begin{bmatrix} 7 \\ 9 \end{bmatrix} \times \alpha$



- With an accurate relationship $i = f(i_{\text{sens}})$ can have $\omega \approx \widehat{\omega}$ in Exp-t 4.
- For now, let us live with $\widehat{i} \approx \frac{i_{\text{sens}}}{1.8} - \frac{1}{30}$ or whatever gave you ω closest to $\widehat{\omega}$ in Exp-t 4. Consequences to be seen in experiment on DOB.

What is a trackable i_d?

We define a trackable i_d as one that does not need u to go into saturation. In our case u saturates at about 9 – 10 V. So, let us find i_d that needs $u \leq 9$ V for the applied T_L.

- $L \approx 0$ (see, e.g., Exp-t 4) $\Rightarrow u = R_\Sigma i + E \Rightarrow \boxed{u = R_\Sigma i + K_b \omega}$.
- At ω_{ss}, $\dot{\omega} = 0 \Rightarrow T = K_t i = T_L + B\omega \Rightarrow \boxed{\omega = \frac{K_t i - T_L}{B}}$.
- From the two boxed equations, write an equation in terms of u, i, T_L by eliminating ω:

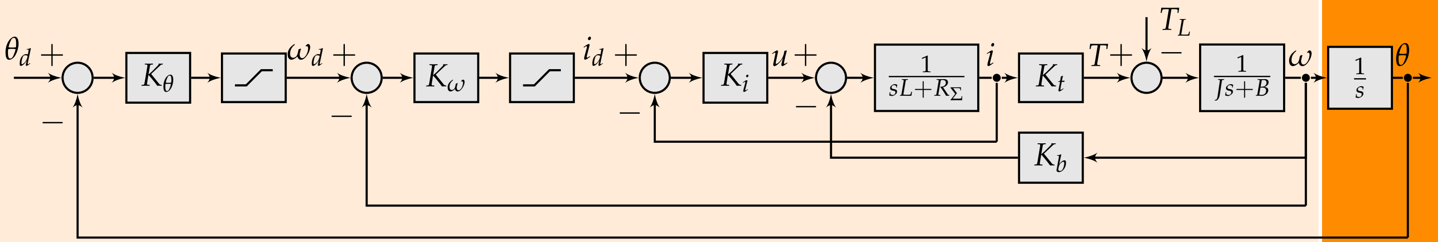
$$u + \frac{K_b}{B} T_L = \left(R_\Sigma + \frac{K_b K_t}{B} \right) i$$

- For $u = 9$ V, $T_L = 0$, find i_{d1}. For $u = 9$ V, $T_L = 0.003$ Nm, find i_{d2}. These are max. trackable currents.
- The min. trackable currents are obtained from $i = T_L / K_t$.

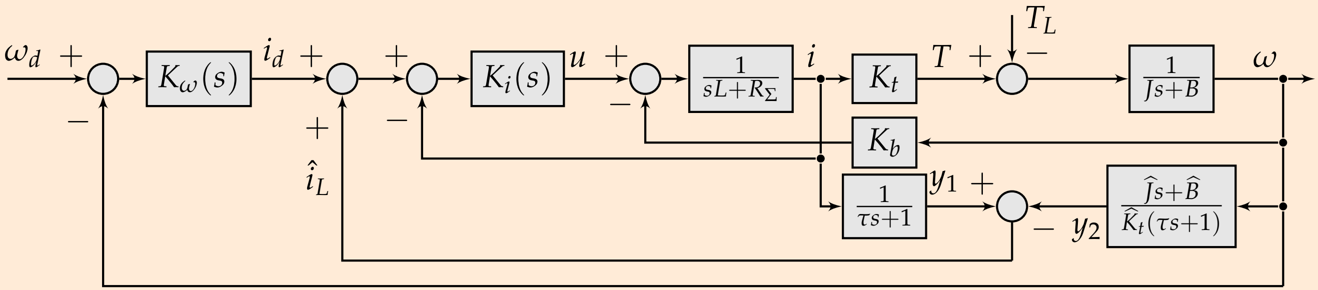
Where is i tracking i_d useful?

Abbreviation: $K_\theta \equiv K_\theta(s)$, $K_\omega \equiv K_\omega(s)$, $K_i \equiv K_i(s)$.

Position control of pmdc motor while restricting speed and current.



Disturbance observer (DOB): Exp-t 6.



Homework (HW) vs. Lab work (LW)

HW

Determine TF from u to i for our pmdc motor module. Use R_Σ & B found experimentally.

↓

Choose PI controller for $t_s = 0.5$ s

↓

Determine max. values of i_{d1} & i_{d2}

↓

Simulate CL sys using modified `easysim.m` for i_{d1} & i_{d2}

↓

Plot i_1 vs. t & i_2 vs. t on one figure. Plot u_1 vs. t & u_2 vs. t one one figure.

LW

In `main-prog.c`: code PI controller; implement control of i; use $i = f(i_{\text{sens}})$ that gave best results in Exp-t 4.

↓

Track 90%i_{d1} and 90%i_{d2}. i_{d2} provided by 1.5 kg weight hanging from pulley.

↓

While pulley rotating, in both cases stall and release pulley

↓

Take plots in both cases before and after application of additional T_L

↓

Comment on dist. rej. & tracking.

Discretization

$$\frac{a_1 s + a_0}{s^2 + b_1 s + b_0}$$

$\downarrow$ Simulation diagram

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -b_0 x_1 - b_1 x_2 + u \\ y &= a_0 x_1 + a_1 x_2 \end{aligned}$$

$\downarrow$ Euler's approximation

$$\begin{aligned} x_{1(k+1)} &= x_{1(k)} + \Delta t\ x_{2(k)} \\ x_{2(k+1)} &= -b_0 \Delta t\ x_{1(k)} + (1 - b_1 \Delta t) x_{2(k)} + \Delta t\ u_k \\ y_k &= a_0 x_{1(k)} + a_1 x_{2(k)} \end{aligned}$$

Simulate; LW: C code, Implement, Analyze

- Simulation: `easysim.m`
- Discretized controller
  $\longrightarrow$ C code:

- Implement: As in demo slides
- Analyze: Compare results

$$
\begin{aligned}
x_1(k+1) &= a_{11}x_1(k) + a_{12}x_2(k) + b_1u(k) \\
x_2(k+1) &= a_{21}x_1(k) + a_{22}x_2(k) + b_2u(k) \\
y(k) &= c_1x_1(k) + c_2x_2(k) + du(k)
\end{aligned}
$$

In `main-prog.c` before `main()` insert `float x1[2],x2[2];`
In `main()` insert `x1[0] = x2[0] = 0;`

```
x1[1] = a11 * x1[0] + a12 * x2[0] + b1 * u;
x2[1] = a21 * x1[0] + a22 * x2[0] + b2 * u;
    y = c1 * x1[0] + c2 * x2[0] + d * u;
x1[0] = x1[1];
x2[0] = x2[1];
```