Lecture 5: Experiment 4
EE380 (Control Systems)

Ramprasad Potluri
Associate Professor
potluri@iitk.ac.in

Manavaalan Gunasekaran
PhD student
manvaal@iitk.ac.in

Department of Electrical Engineering
Indian Institute of Technology Kanpur

August 19, 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.

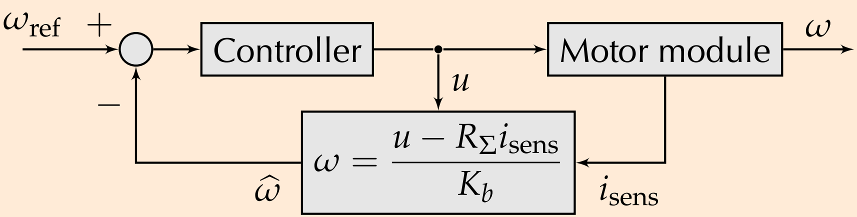
Part I

Procedure of Exp.4

Outline of the experiment

Feedback of ω assumed absent. Want ω to track ω_{ref}. Steps:

- Obtain estimate $\widehat{\omega}$ of ω using u and i.
- Use feedback of $\widehat{\omega}$ to track ω_{ref} with controller from Exp.1.



- Repeat control using feedback of ω with controller from Exp.1.
- Is $\widehat{\omega}$ an adequate replacement for ω?

Tasks common to all 6 experiments

Simulation

- Perform PC-based simulation of CL system using GNU Octave.
- Perform PC-based simulation of digital control of a continuous-time system using GNU Octave.

Realization on hardware

- Utilize the various components of an integrated development environment (IDE): editor, compiler, linker, debugger, and programmer to program a μC.
- Program controller using C language into μC.
- Monitoring: read data into PC from μC using UART modules.

Analysis

- Compare actual performance with predicted performance.

Homework (HW) vs. Lab work (LW)

HW

Recall $\frac{\omega(s)}{V(s)} = \frac{K_m}{\tau_m s+1}$ & $C(s)$ from Exp.1

↓

Find R_Σ & B from
$$\begin{aligned} K_m &= \frac{K_T}{R_\Sigma B + K_T K_b}, \\ \tau_m &= \frac{R_\Sigma J}{R_\Sigma B + K_T K_b} \end{aligned}$$

↓

Simulate CL sys using feedback of i & `easysim.m`

↓

Simulate CL sys using feedback of ω & `easysim.m`

↓

Is $\widehat{\omega}$ adequate substitute for ω?

LW

Identify K_m & τ_m using step input

↓

If K_m, τ_m different from Exp.1, then calculate R_Σ & B. Else, use R_Σ from HW.

↓

Control using feedback of i

↓

Control using feedback of ω

↓

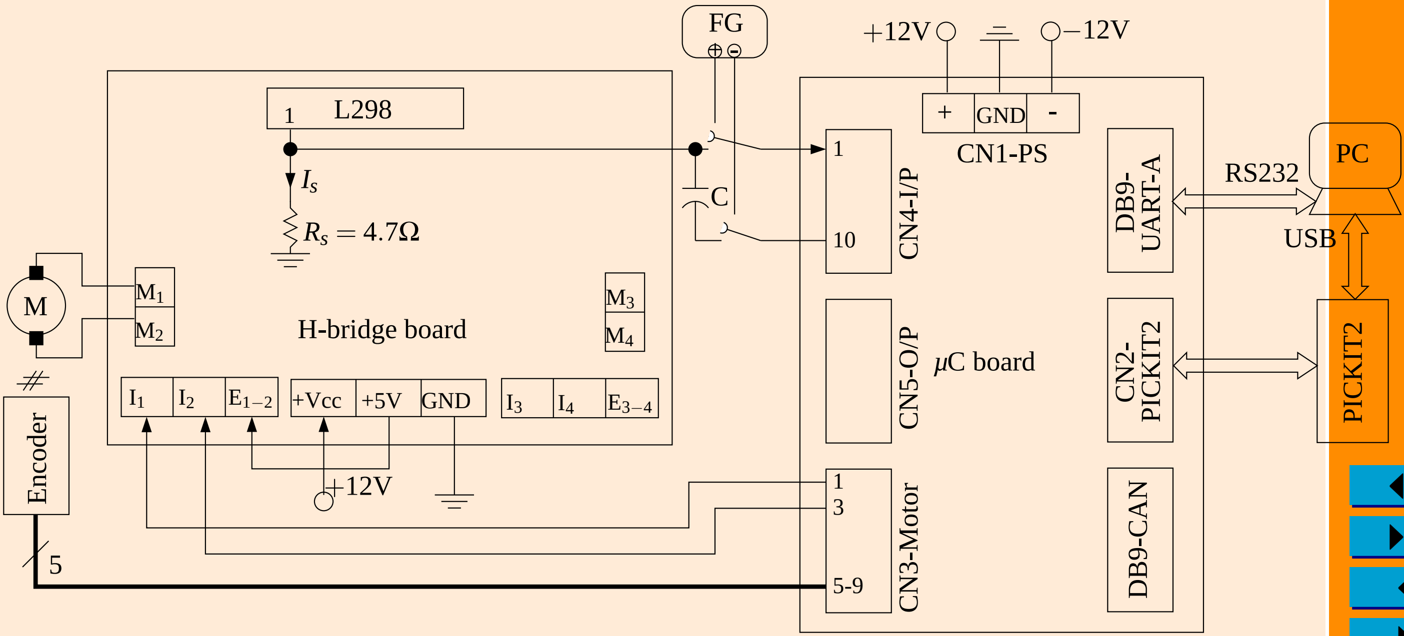
Plot ω from both controls on same figure

↓

Is $\widehat{\omega}$ adequate substitute for ω?

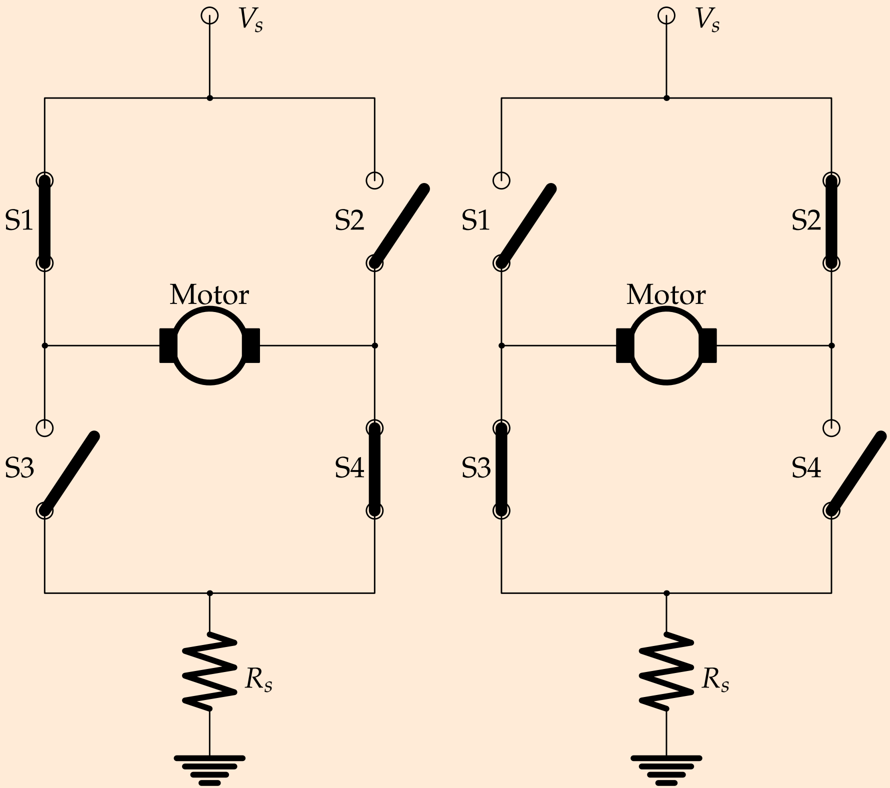
Hardware connections

The setup



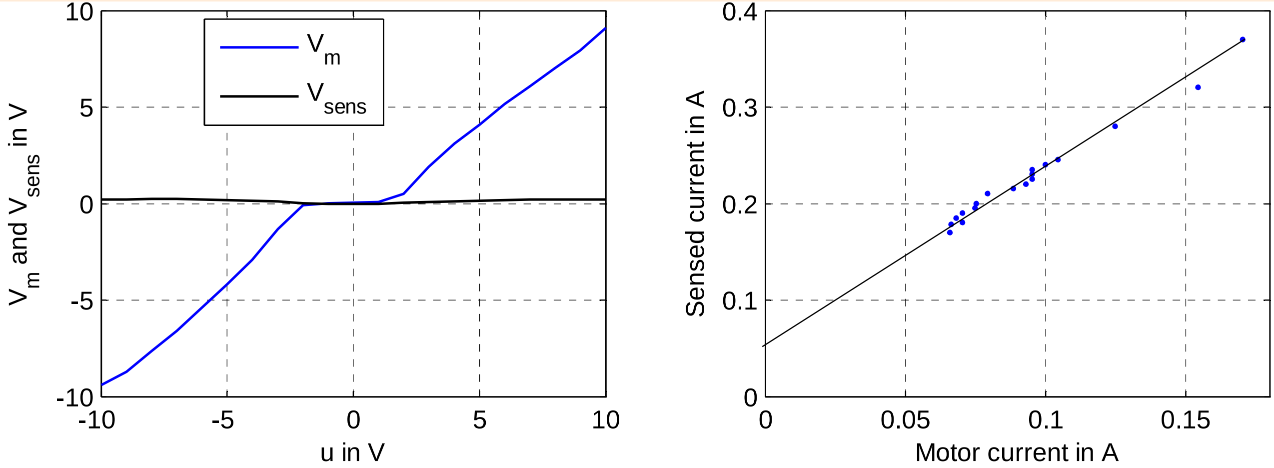
Pin 1 of L298 is connected to Pin 1 of CN4-I/P of μC board.

Where R_s is in the H-bridge



Armature resistance includes all the resistance in the path of the armature: R_H (sum of resistance of S1 & S2 or S3 & S4, whichever pair is conducting), R_m (resistance of motor's armature), and R_s.

Correct current to use for feedback



- Instead of i_{sens}, more accurate to use $i_m \approx \frac{1}{1.8} i_{\text{sens}} - \frac{1}{30}$, which fits the straight line in the right hand figure.
- If time permits, replace i_{sens} with i_m and redo experiment.

Discretization

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

Simulation diagram

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

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

$$\begin{aligned} x_1(k+1) &= a_{11}x_1(k) + a_{12}x_2(k) + b_1 u(k) \\ x_2(k+1) &= a_{21}x_1(k) + a_{22}x_2(k) + b_2 u(k) \\ y(k) &= c_1 x_1(k) + c_2 x_2(k) + d u(k) \end{aligned}$$

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

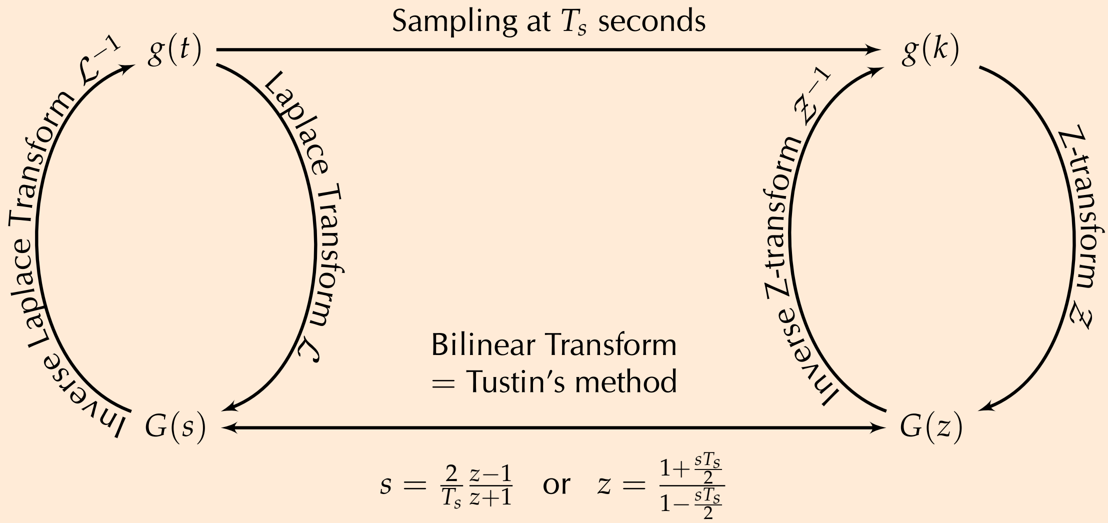
```
x1[1] = a11 * x1[0] + a12 * x2[0] + b1 * u;
x2[1] = a21 * x1[0] + a22 * x2[0] + b2 * u;
    y = c1 * x1[0] + c2 * x2[0] + d * u;
x1[0] = x1[1];
x2[0] = x2[1];
```

Part II

Review of Exp.2

Least squares sys-id theory

Bilinear transform and Z-transform



- Both s-domain & z-domain are fictitious domains.
- They simplify working with differential equations & difference equations respectively.
- Bilinear transform is not the only way to go $G(s) \leftrightarrow G(z)$.
- T_s constrained by Nqyquist sampling rate.

$G(s) \longleftrightarrow G(z)$

- Consider definitions of $\mathcal{L}$ and $\mathcal{Z}$

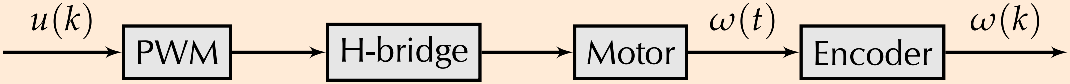
$$Y(s) = \mathcal{L}\{y(t)\} \triangleq \int_{t=0}^{\infty} y(t)e^{-st}dt$$

$$Y(z) = \mathcal{Z}\{y(k)\} \triangleq \sum_{k=0}^{\infty} y(k)z^{-k}$$

- Comparison suggests $z = e^{sT_s}$.
- To convert $G(s)$ to $G(z)$, can substitute $s = \frac{\ln z}{T_s}$.
- Easier to work with an approximation

$$z = e^{sT_s} = e^{\frac{sT_s}{2}}e^{\frac{sT_s}{2}} = \frac{e^{\frac{sT_s}{2}}}{e^{-\frac{sT_s}{2}}} = \frac{1 + \frac{\left(\frac{sT_s}{2}\right)}{1!} + \frac{\left(\frac{sT_s}{2}\right)^2}{2!} + \cdots}{1 + \frac{\left(-\frac{sT_s}{2}\right)}{1!} + \frac{\left(-\frac{sT_s}{2}\right)^2}{2!} + \cdots} \approx \frac{1 + \frac{sT_s}{2}}{1 - \frac{sT_s}{2}}$$

How Z-transform used in our sys-id



- $u(k)$ denotes sample of $u(t)$ at sampling instant $t = kT_s$.
- Let $u(k) \to \omega(k)$ TF be $G(z)$.
- Use $u(k), \omega(k)$ pairs to build $G(z)$.
- Use bilinear transform to go from $G(z)$ to $G(s)$.

Important property of Z-transform used:

$$z^{-l}X(z) \leftrightarrow x(k-l) \text{ given } X(z) \leftrightarrow x(k).$$

Back
Forward
Close

What is least squares sys-id? (1/2)

- Let $G(z) = \dfrac{b_1 z^2 + b_2 z + b_3}{z^3 + a_1 z^2 + a_2 z + a_3} = \dfrac{Y(z)}{U(z)}$.
- Cross multiply:

$$b_1 z^2 U(z) + b_2 z U(z) + b_3 U(z) = z^3 Y(z) + a_1 z^2 Y(z) + a_2 z Y(z) + a_3 Y(z).$$

- Multiply throughout by z^{-3}:

$$\begin{aligned} b_1 z^{-1} U(z) + b_2 z^{-2} U(z) + b_3 z^{-3} U(z) = \\ Y(z) + a_1 z^{-1} Y(z) + a_2 z^{-2} Y(z) + a_3 z^{-3} Y(z). \end{aligned}$$

- Take $\mathcal{Z}^{-1}$ to obtain difference equation

$$\begin{aligned} b_1 u(k-1) + b_2 u(k-2) + b_3 u(k-3) = \\ y(k) + a_1 y(k-1) + a_2 y(k-2) + a_3 y(k-3). \end{aligned}$$

What is least squares sys-id? (2/2)

Consider $$b_1u(k-1)+b_2u(k-2)+b_3u(k-3) = y(k)+a_1y(k-1)+a_2y(k-2)+a_3y(k-3). \quad (1)$$

- Let $\sigma = [b_1 \quad b_2 \quad b_3 \quad -a_1 \quad -a_2 \quad -a_3]^\mathsf{T}$.
- Suppose we have data of $u(k)$ and $y(k)$ for $k = 0, 1, \ldots, N$.
- Problem: Find σ such that (1) holds for this data.

  I.E., find parameters of a TF that fits to input-output data.
- Let error in the fit be

$$\varepsilon(k,\sigma) = b_1u(k-1)+b_2u(k-2)+b_3u(k-3)-y(k) - a_1y(k-1)-a_2y(k-2)-a_3y(k-3).$$

- Modified problem: Find σ to minimize $\mathcal{J}(\sigma) \triangleq \sum_{k=0}^{N} \varepsilon^2(k,\sigma)$.
- If $\mathcal{J}(\sigma = \sigma_0) = 0$, then find best estimate $\hat{\sigma}$ of σ_0.

What the experiment taught

- Sys-id techinques from Exp.1 & Exp.2 give different results.
- Likely cause is not only the dead zone nonlinearity in the plant, but also the input signals the sys-id technique uses.

  E.g., the step input ($u = 7$) in Exp.1 does not keep plant in dead zone, while the low-frequency ($5 - 10$ Hz) triangular input makes the plant go into dead zone twich every cycle.
- Will using rectangular waveform instead of triangular waveform (TW) give a different model with least squares sys-id (LSS)?
- If plant behaves as 1st order even with TW, LSS will say that plant has one LHP pole that is $10 - 20$ times deeper than the other.