09/13/23

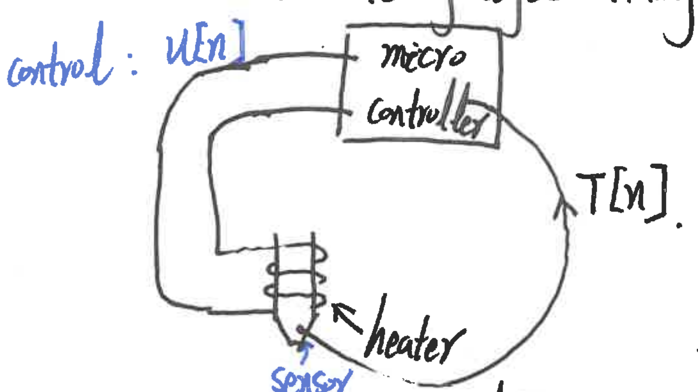
* Review of 1st order Difference Equation
* Feedforward Control
* Feedback control.

1. Last time, we learned about models represented by 1st order Linear Constant Coefficient Ordinary Difference Equation

$$\begin{cases} y[n] = \lambda y[n-1] + \gamma_0 u[n-1] \\ y[0] \end{cases}$$

solution: $$y[n] = \underbrace{\lambda^n y[0]}_{\substack{\text{Zero Input Response} \\ \text{(homogeneous)}}} + \underbrace{\gamma \sum_{m=0}^{n-1} \lambda^{n-1-m} u[m]}_{\substack{\text{Zero State Response} \\ \text{(one particular solution)}}}.$$

We talk about one specific example: 3D printer nozzle.



Temperature change: $\frac{dT}{dt} \sim \underset{\gamma u[n]}{\text{heat input}} - \underset{\beta T[n]}{\text{heat loss}}.$ The higher T, the faster heat gets dissipated.

In discrete time form: $$\frac{T[n]-T[n-1]}{\Delta T} = -\beta T[n-1] + \gamma u[n-1].$$

ΔT: intervals for sampling temperature.

Today, we added the heat loss term compared with last lecture, to be ~~more~~ closer to real situation (and closer to your Lab this week).

Re-arrange: $T[n] = (1 - \Delta T \beta) T[n-1] + \Delta T \gamma u[n-1]$.

There are two methods for generating input $u[n]$ to reach the desired T_d.

2. Feed Forward Control.

To reach the target temp. $T_d[n] = T_d$ (a constant), set $u[n] = K_{ff} T_d$.

(Higher target T_d, more power input onto the heater)

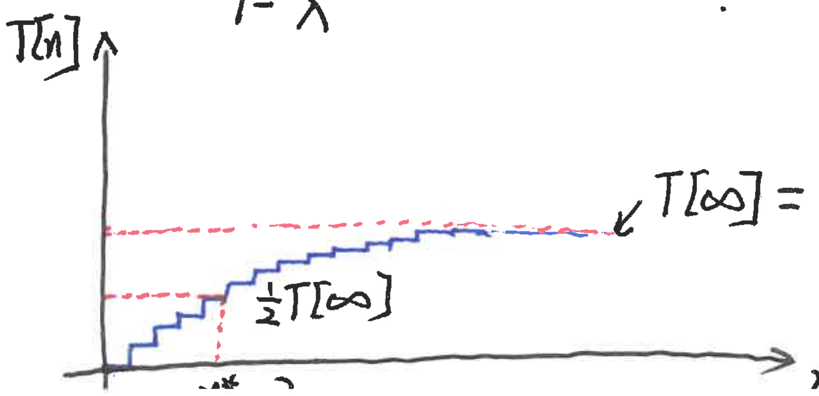
$$T[n] = \underbrace{(1 - \Delta T \beta)}_{\lambda} T[n-1] + \gamma \Delta T K_{ff} T_d[n-1].$$

We learned $|\lambda| < 1$ to be a stable system.

For simplicity, let's assume $T[0] = 0$ (not necessarily $0^\circ C$, but a chosen reference)

$$T[n] = \gamma \Delta T K_{ff} \sum_{m=0}^{n-1} \lambda^{n-m-1} T_d = \gamma \Delta T K_{ff} T_d \frac{1 - \lambda^n}{1 - \lambda} \quad \text{(geometric series)}$$

$$T[\infty] = \frac{\gamma \Delta T K_{ff} T_d}{1 - \lambda} \quad (\lambda^n \to 0).$$



$$T[\infty] = \frac{\gamma \Delta T K_{ff} T_d}{\underset{\uparrow\,1-\Delta T\beta}{1 - \lambda}} = \frac{\gamma}{\beta} K_{ff} T_d.$$

we noted: ① need to calibrate K_{ff} such that $T[\infty] = \frac{\gamma}{\beta} K_{ff} T_d \to T_d$.

② How fast does $T[n]$ approaches T_d?

Half time: $\lambda^{n^*} = \frac{1}{2}$ $\quad n^* = \log_\lambda \frac{1}{2}$

Or if from experiment, we get n^* $\quad \lambda = \left(\frac{1}{2}\right)^{\frac{1}{n^*}} = e^{\frac{1}{n^*}\log\frac{1}{2}}$

Replace $\lambda = 1 - \Delta T \beta$, $\Rightarrow$ $\beta = \frac{1}{\Delta T}\left(1 - e^{\frac{1}{n^*}\log\frac{1}{2}}\right)$

Note β and γ can be determined through this "step response"

(You will use it this Friday).

From point 1: there's no guarantee that $T[\infty] \to T_d$, need a better method.

---

3. Feedback control.

Let the input heat be proportional to the difference between T_d and measured $T[n]$

$$u[n] = K_p \left(T_d[n] - T[n]\right).$$

$$T[n] = (1 - \Delta T \beta) T[n-1] + \Delta T \gamma K_p \left(T_d[n-1] - T[n-1]\right).$$

Rearrange $\Rightarrow$ $T[n] = \underbrace{(1 - \Delta T \beta - \Delta T \gamma K_p)}_{\lambda^*} T[n-1] + \Delta T \gamma K_p T_d[n-1]$.

a. What is the final temperature? —— steady state error.

One can fit in the formula of $T[n]$, Or use the trick:

Assuming stable system $(|\lambda^*| < 1)$ $\lim_{n\to\infty} T[n] = \lim_{n\to\infty} T[n-1] = T[\infty]$.

$$T[\infty] = \lambda^* T[\infty] + \Delta T \gamma K_p T_d.$$

$$T[\infty] = \frac{1}{1-\lambda^*} \Delta T \gamma K_p T_d.$$

Replace $\lambda^* = 1 - \Delta T \beta - \Delta T \gamma K_p$: $\boxed{T_{FB}[\infty] = \frac{\cancel{\beta}\gamma K_p}{\gamma K_p + \beta} T_d.}$

How does this compare with Feed Forward case $T_{FF}[\infty] = \frac{\gamma K_{ff}}{\beta} T_d$?

K_p appears both in the numerator and denominator,

If $\gamma K_p >> \beta$, $T_{FB}[\infty] \approx T_d$. $\Longrightarrow$ want K_p as high as possible.

Except for the restraint $|\lambda^*| < 1$. (Is the range of valid K_p related to ΔT?)

b. What happens if there is disturbance ? —— Disturbance Rejection.

$$T[n] = \lambda^* T[n-1] + \Delta T \gamma K_p T_d[n-1] + \underbrace{\Delta T \cdot \chi[n-1]}_{\text{disturbance.}}$$

Assume a constant disturbance $\chi[n] = \chi$ starting at some n_0.

Same trick for steady state:

$$T[\infty] = \lambda^* T[\infty] + \Delta T \gamma K_p T_d + \Delta T \chi$$

$$T[\infty] = \frac{\Delta T \gamma K_p}{1-\lambda^*} T_d + \frac{\Delta T}{1-\lambda^*} \chi$$

Define error : $e[n] = T_d[n] - T[n]$, replacing $\lambda^* = 1 - \Delta T \beta - \Delta T \gamma K_p$.

$$\boxed{e[\infty] = \frac{\cancel{\gamma K_p}\beta}{\gamma K_p + \beta} T_d \cancel{+} - \underbrace{\frac{1}{\gamma K_p + \beta}}\chi}$$ $\Longrightarrow$ want a larger K_p, to be more robust to disturb[ance]

February 15, 2023

6.3100 Lecture 4 Notes – Spring 2023

Experimental characterization of first order systems, and simulation tools

Dennis Freeman and Kevin Chen

Outline:

1. First order system: estimation of system properties
2. MATLAB tools for analyzing a first order system
3. Review of complex numbers – interpretation using the complex plane

1. First order system: estimation of system properties

In the previous lectures, we discussed how to solve first order systems. If I give you the mathematical model of a system, then you can implement a proportional controller and find the optimal Kp. However, when you design a control system, how do you know the properties of the system in the first place? If I give you a robot, how do you figure out key parameters in your model? System identification is an area of study in which researchers measure the system properties through running simple experiments and observing system response.

We will introduce a very simple technique that is helpful for solving lab 1. First, a first order system is given by the equation:

$$y[n] = y[n-1] + \Delta T(\beta y[n-1] + \gamma c[n-1])$$

Here the parameters β and γ are system properties that we want to measure. We need to design a reasonable control input signal $c[n]$. Note that our goal is to measure β and γ, not stably control the system or closely follow a trajectory. Let's try two different options:

Feedback control:

$$u[n] = K_p(y_d[n] - y[n])$$

Given this feedback controller, the system equation becomes:

$$y[n] = y[n-1] + \Delta T(\beta y[n-1] + \gamma K_p(y_d[n-1] - y[n-1]))$$

$$y[n] = y[n-1] + \Delta T\big(\beta y[n-1] - \gamma K_p y[n-1]\big) + \gamma \Delta T K_p y_d[n-1]$$

$$y[n] - y[n-1] - \Delta T\big(\beta y[n-1] - \gamma K_p y[n-1]\big) = \gamma \Delta T K_p y_d[n-1]$$

$$y[n] = y[n-1]\big(1 + \Delta T\beta - \Delta T\gamma K_p\big) + \gamma \Delta T K_p y_d[n-1]$$

Here the natural frequency is given by:

$$\lambda = 1 + \Delta T\beta - \Delta T\gamma K_p$$

This is a little bit problematic. The natural frequency changes as we change K_p. So it is not easy to measure the system β and γ. Let's try another controller.

Feedforward control:

$$u[n] = K_{ff} y_d[n]$$

Given this feedforward controller, the system equation becomes:

$$y[n] = y[n-1] + \Delta T(\beta y[n-1] + \gamma K_{ff} y_d[n-1])$$

$$y[n] = y[n-1] + \Delta T(\beta y[n-1]) + \gamma \Delta T K_{ff} y_d[n-1]$$

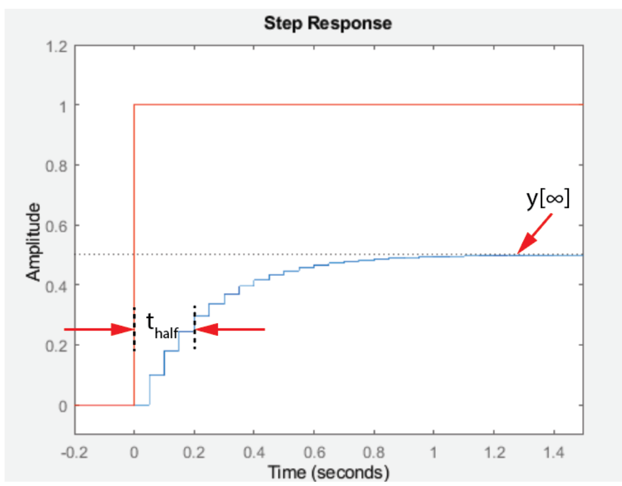
$$y[n] - y[n-1] - \Delta T(\beta y[n-1]) = \gamma \Delta T K_{ff} y_d[n-1]$$

$$y[n] = y[n-1](1 + \Delta T\beta) + \gamma \Delta T K_{ff} y_d[n-1]$$

Now this is much better because the natural frequency is given by:

$$\lambda = 1 + \Delta T\beta$$

To measure β and γ, we need to derive 2 relationships. A common approach is to look at the step response of a system. Here we let $y_d[n] = 1$ for n>0, and y[n]=0 for n=0. Suppose we run this experiment and obtain the following graph:



Graphically, we need to get two equations.

First, we can calculate λ by measuring the time y[n] uses to reach half of y[∞]. We have:

$$\lambda^{n^*} = 0.5$$

For the plot above, we measure $n^* = 3$. To calculate λ, we can use:

$$\lambda = \exp\left(\frac{1}{n^*}\log(0.5)\right)$$

We can back-calculate β:

$$\beta = \frac{\exp\left(\frac{1}{n^*}\log(0.5)\right) - 1}{\Delta T} = -4.1$$

Next, we need to solve for γ. We can solve for γ using the steady state condition. We measure y[∞] = 0.5. For large time, the equation becomes:

$$y[\infty] = y[\infty](1 + \Delta T\beta) + \gamma\Delta T K_{ff}$$

We obtain the equation:

$$\gamma = -\frac{y[\infty]\beta}{K_{ff}} = 2.1$$

You will practice this technique in lab 1.

2. MATLAB tools for analyzing a first order system

Most 1st order systems are simple enough that we can solve them manually. However, as we will see in the next few weeks, the algebra becomes tedious for higher order system. Numerical tools are useful for solving control systems. We are going to use MATLAB in this course. The reason that we use MATLAB is that it has very convenient packages, and many physical systems are controlled by MATLAB/Simulink.

The code below generates the system identification plot on page 2.

```
%define variables
Kff = 1;
beta = -4;
gamma = 2;
dt = 1/20;

%define the numerator and denomination of the transfer function
%will be covered in detail next week
den = [1, -(1+dt*beta)];
num = [0 dt*gamma*Kff];

%open a new figure
close all; figure(2); hold on

%form a discrete time control system
sys = tf(num,den,dt,'variable','z^-1');
step(sys,1.5) %simulate for 1.5 second

%plot reference data
axis([-0.2 1.5 -0.1 1.2])
plot([-0.2 0],[0 0],'r-')
plot([0 0],[0 1],'r-')
plot([0 1.5],[1 1],'r-')
```

The lines of code that may look puzzling relate to the definition of a transfer function (tf) with a numerator and a denominator. We will explain the transform techniques in the next 2 weeks of class. For now, it is sufficient to use this function. It comes from "pattern matching".

$$y[n] = y[n-1](1 + \Delta T\beta) + \gamma\Delta T K_{ff} y_d[n-1]$$

$$y[n] - y[n-1](1 + \Delta T\beta) = \gamma\Delta T K_{ff} y_d[n-1]$$

The denominator relates to the coefficients in front of y[n] and y[n-1]. Here they are given by :

$$den = [1, \qquad -\Delta T\beta - 1]$$

The numerator relates to the coefficients in front of $y_d[n]$ and $y_d[n-1]$. Here they are given by:

$$num = [0, \qquad \Delta T\gamma K_{ff}]$$

We can easily modify this code to study proportional feedback control. Under feedback control, the system equation is given by:

$$y[n] - y[n-1]\left(1 + \Delta T\beta - \Delta T\gamma K_p\right) = \gamma\Delta T K_p y_d[n-1]$$

Now we only need to change the denominator and numerator lines:

```
%change to proportional controller
den = [1, -(1+dt*beta)+dt*gamma*Kp];
num = [0 dt*gamma*Kp];
```

We can re-run the simulation and get the following graph: