EECS 16B Designing Information Devices and Systems II
Spring 2019 A. Sahai, J. Roychowdhury, K. Pister

HW 5

This optional homework is "due" on Wednesday, March 6, 2019, at 11:59PM. Self-grades are "due" on Saturday, March 9, 2019, at 11:59PM.

1. Lecture notes and feedback

Staying up to date with lectures is an important part of the learning process in this course. This question is worth as much as the rest of the homework. Fortunately, it is also really easy.

(a) **Please attach your notes (handwritten or typed is fine) for the lecture from the Tuesday of the week before this HW is due *to the separate Gradescope assignment*.**

(b) **What did you think was the most important lesson of the Tuesday lecture?**

(c) **Please attach your notes (handwritten or typed is fine) for the lecture from the Thursday of the week before this HW is due *to the separate Gradescope assignment*.**

(d) **What did you think was the most important lesson of the Thursday lecture?**

(e) **Do you have any feedback on these lectures? How could they be improved in the future to better support students taking 16B?**

2. Identifying systems from their responses to known inputs

In many problems, we have an unknown system, and would like to characterize it. One of the ways of doing so is to observe the system response with different initial conditions (or inputs). This problem is also called system identification. It is a prototypical example of a problem that today is called machine learning — inferring an underlying pattern from data, and doing so well enough to be able to exploit that pattern in some practical setting.

Because you have an exam coming up, we have decided to simply release this problem as a demonstration using NumPy instead of making you work through it. You will have to do system identification by least squares in the lab later, but here, the only thing you need to do is play with the included Jupyter notebook and note down your observations.

(a) **Please share your observations on Example 2.**

(b) **Please share your observations on Example 3.**

(c) **Please share your observations on Example 4.**

(d) **Please share your observations on Example 5.**

3. Design for controllability and observability

We are given a discrete-time system

$$\vec{x}(i+1) = \begin{bmatrix} -2 & 1 \\ \gamma & -2 \end{bmatrix} \vec{x}(i) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u(i) \tag{1}$$

$$y(i) = \begin{bmatrix} 0 & 1 \end{bmatrix} \vec{x}(i) \tag{2}$$

with tuneable parameter γ.

(a) **How could we tune γ to make the system controllable but not observable?**

(b) **How could we tune γ to make the system observable but not controllable?**

(c) We are given a new discrete-time system

$$\vec{x}(i+1) = \begin{bmatrix} -1 & 1 \\ 0 & -1 \end{bmatrix} \vec{x}(i). \tag{3}$$

along with only one sensor and one actuator to control and observe the system.
Which state should we control with the actuator to make the system controllable?
In other words, you can design a $\vec{b}$ to add a $\vec{b}u[i]$ to the dynamics given in (3), however this $\vec{b}$ can only contain a single 1 and the rest of the entries must be 0. Where do you place this 1?

(d) In the above part, **which state should we measure with the sensor to make the system observable?**
In other words, you can design a $\vec{c}$ to add a scalar output $y[i] = \vec{c}^T\vec{x}[i]$ to the dynamics given in (3). However this $\vec{c}$ can only contain a single 1 and the rest of the entries must be 0. Where do you place this 1?

4. Observations of discrete systems' orbits

The concept of observability exists to tell us whether or not we can eventually infer the state of a system given any (unknown) initial condition assuming perfect knowledge of the system, no controls or disturbances, and noiseless observations. What makes this nontrivial is that we don't know the initial condition — this is what must be inferred.

For a discrete-time system with n-dimensional state $\vec{x}$ with neither inputs nor disturbances

$$\vec{x}(i+1) = A\vec{x}(i) \tag{4}$$

and with linear observations

$$\vec{y}(i) = C\vec{x}(i) \tag{5}$$

this can be checked by seeing whether the observability matrix

$$\mathscr{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{6}$$

has enough linearly independent rows. In other words, $\text{span}(\mathscr{O}^T) = \mathbb{R}^n$. If it does span the whole space, then the system (4) is observable using (**??**). If it does not, then the system is not observable.

The goal of this problem is to help you understand why this is true, and thereby to better understand both this and the controllability test proven in lecture.

(a) Consider the case of $C = \vec{c}^T$ — a single row (a scalar observation per unit time). Suppose that you find a particular k so that $\vec{c}^TA^k = \sum_{i=0}^{k-1} \alpha_i \vec{c}^TA^i$ — in other words, that $\vec{c}^TA^k$ lies in the span of earlier rows that have been found. **Show that there exist β_i so that $\vec{c}^TA^{k+1} = \sum_{i=0}^{k-1} \beta_i \vec{c}^TA^i$.**
(HINT: This is like lecture except sideways.)

(b) Consider an arbitrary $\ell > k$. Assume that $\vec{c}^T A^\ell = \sum_{i=0}^{k-1} \gamma_i \vec{c}^T A^i$ — in other words, that $\vec{c}^T A^\ell$ lies in the span of the first k rows that had been found. **Show that there exist δ_i so that $\vec{c}^T A^{\ell+1} = \sum_{i=0}^{k-1} \delta_i \vec{c}^T A^i$.**

(HINT: This should feel almost entirely repetitive)

From the base case of k and the inductive step established here, you can conclude that the statement holds for all $\ell \geq k$. Once you see that the span of the rows has stopped increasing, you can safely stop checking.

(c) You know from the Gaussian elimination that you saw in detail in 16A that you cannot have more than n linearly independent rows of length n. This is because Gaussian elimination on linearly independent rows always makes one step of progress and there are only n steps of progress maximum possible. **Argue why this and the argument in earlier parts implies that checking $\mathscr{O}$ as defined above is enough to determine observability.**

(HINT: Consider first the case when it is not observable. Why is $\mathscr{O}$ sufficient for checking then? Next consider the case when it is observable. Why is $\mathscr{O}$ sufficient then?)

5. Sampling a continuous-time control system to get a discrete-time control system

Recall from lecture that a continuous-time system

$$\begin{aligned} \frac{d}{dt}\vec{x}(t) &= A\vec{x}(t) \\ y(t) &= C\vec{x}(t) \end{aligned} \tag{7}$$

can be represented with a discrete-time model

$$\begin{aligned} \vec{x}_d(i+1) &= A_d\vec{x}(i) \\ y_d(i) &= C\vec{x}_d(i) \end{aligned} \tag{8}$$

where $\vec{x}_d(i)$ and $y_d(i)$ are the values of the state $\vec{x}(t)$ and output $y(t)$ at time instants $t = i\Delta$, $i = 0,1,2,3,\ldots$ In this problem we will see that the observability of (8) might depend on the sampling period Δ — some periods Δ may be bad. Since observability depends on only A and C alone we have omitted the inputs in the equations above.

(a) Suppose A is diagonalizable — has a full complement of eigenvectos. So there exists a matrix V such that

$$V^{-1}AV = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}.$$

Show that

$$A_d = V \begin{bmatrix} e^{\lambda_1\Delta} & & \\ & \ddots & \\ & & e^{\lambda_n\Delta} \end{bmatrix} V^{-1}.$$

(HINT: To do so, you can change coordinates to a new state vector $\tilde{\vec{x}} = V^{-1}\vec{x}$ and then get the discrete-time model for $\tilde{\vec{x}}_d(i)$. You would then return to the original state with $\vec{x}_d(i) = V\tilde{\vec{x}}_d(i)$ to obtain (8).)

(b) Use the result of part (a) to **calculate A_d as a function of the sampling period Δ** for the specific case of

$$A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}.$$

(c) **BONUS:** Let $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$ so that (A, C) is an observable pair. **Show that there exist values of Δ for which (A_d, C) is *not* observable.**

The fact that such "counterexamples" exist only for very specific values of Δ tells us that indeed it should be possible to meaningfully extend the concept of observability to continuous time, even though we have not done so yet.

6. Write Your Own Question And Provide a Thorough Solution.

Writing your own problems is a very important way to really learn material. The famous "Bloom's Taxonomy" that lists the levels of learning is: Remember, Understand, Apply, Analyze, Evaluate, and Create. Using what you know to create is the top level. We rarely ask you any homework questions about the lowest level of straight-up remembering, expecting you to be able to do that yourself (e.g. making flashcards). But we don't want the same to be true about the highest level. As a practical matter, having some practice at trying to create problems helps you study for exams much better than simply counting on solving existing practice problems. This is because thinking about how to create an interesting problem forces you to really look at the material from the perspective of those who are going to create the exams. Besides, this is fun. If you want to make a boring problem, go ahead. That is your prerogative. But it is more fun to really engage with the material, discover something interesting, and then come up with a problem that walks others down a journey that lets them share your discovery. You don't have to achieve this every week. But unless you try every week, it probably won't ever happen.

7. Homework Process and Study Group

Citing sources and collaborators are an important part of life, including being a student! We also want to understand what resources you find helpful and how much time homework is taking, so we can change things in the future if possible.

(a) **What sources (if any) did you use as you worked through the homework?**

(b) **Who did you work on this homework with?** List names and student ID's. (In case of homework party, you can also just describe the group.)

(c) **How did you work on this homework?** (For example, *I first worked by myself for 2 hours, but got stuck on problem 3, so I went to office hours. Then I went to homework party for a few hours, where I finished the homework.*)

(d) **Roughly how many total hours did you work on this homework?**

Contributors:

- Alex Devonport.
- John Maidens.
- Anant Sahai.
- Murat Arcak.