EECS 16B Designing Information Devices and Systems II
Fall 2020 UC Berkeley
Note 10

1 Overview

In the next module of the course, we will be learning about **dynamical systems** and how we can control them. Some examples are an airplane's flight, the air inside a building, network traffic on the internet, or even a circuit. We will learn how to develop controllers for these systems to regulate particular quantities that we care about, like an autopilot to level an airplane's flight, a thermostat to keep a building at a comfortable temperature, internet congestion control to manage data rates, or a voltage regulator to limit the power consumption. Other dynamical systems and controllers can be found in nature, like the biochemical systems that regulate conditions inside a living cell.

When we want to study or control a dynamical system, our first step is usually to write out equations that describe its physics. These equations are called a *model*, and they predict what a system will do over time. We will study systems that change continuously in time like electrical circuits, and systems that evolve in discrete time steps, like the yearly number of professors in EECS.

State variables are a set of variables that fully represent the state of a dynamical system at a given time, like capacitor voltages and inductor currents in electrical circuits. In a mechanical system, they could be the positions and velocities of masses. The state variables can be written together in a **state vector** $\vec{x}(t) \in \mathbb{R}^n$ where n is the number of state variables that describe the system. The majority of our models will involve vector differential equations but our emphasis is no longer on the solutions to these differential equations, rather we are interested in the overall behavior of the system and how we can control it.

2 Continuous-Time Systems

We will start by looking at continuous time systems. A **continuous-time system** is one in which all of the input, output, and state-variables are continuous time functions. A continuous-time system with n state variables represented in a state vector $\vec{x}(t) \in \mathbb{R}^n$ can often be represented by the following differential equation.

$$\frac{d}{dt}\vec{x}(t) = f(\vec{x}(t)) \tag{1}$$

where $f(\vec{x}(t))$ is an length n-vector that dictates the derivatives of each state variable according to the current value of the states. The form of f depends on the system we are modeling as we will see in examples.

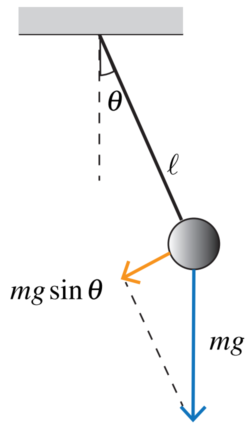
If the system has input variables we can manipulate (*e.g.* voltage and current sources in a circuit, or force and torque delivered to a mechanical system), we represent the system as

$$\frac{d}{dt}\vec{x}(t) = f(\vec{x}(t), \vec{u}(t)) \tag{2}$$

where we refer to $\vec{u}(t)$ as the *control input*, since we can manipulate this input to influence the behavior of the system. Most of our examples will contain a single control input, but we write $\vec{u}(t)$ as a vector to allow for multiple control inputs.

2.1 Example

The motion of the pendulum depicted on the right is governed by the differential equation



$$m\ell\frac{d^2\theta(t)}{dt^2} = -k\ell\frac{d\theta(t)}{dt} - mg\sin\theta(t) \tag{3}$$

where the left hand side is mass×accelaration in the tangential direction and the right hand side is total force acting in that direction.[1] The constant k represents the damping constant of the pendulum due to drag forces.

To bring this second order differential equation to state space form we define the state variables

$$x_1(t) \triangleq \theta(t) \qquad x_2(t) \triangleq \frac{d\theta(t)}{dt}$$

and note that they satisfy

$$\begin{aligned} \frac{dx_1(t)}{dt} &= x_2(t) \\ \frac{dx_2(t)}{dt} &= -\frac{k}{m}x_2(t) - \frac{g}{\ell}\sin x_1(t). \end{aligned} \tag{4}$$

The first equation here follows from the definition of $x_2(t)$, and the second equation follows from (3). In this state representation we have two first order differential equations, one for each state variable, instead of the second order differential equation (3) for one variable.

Here we did not consider disturbances or control inputs that could be applied (say, to balance the pendulum in the upright position) so the equations (4) have the form (2) with

$$f(\vec{x}(t)) = \begin{bmatrix} x_2(t) \\ -\frac{k}{m}x_2(t) - \frac{g}{\ell}\sin x_1(t) \end{bmatrix}.$$

Note how we are unable to write a matrix-differential equation $\frac{d}{dt}\vec{x}(t) = A\vec{x}(t)$. This is because the system is *non-linear.* We will define what it means to be a linear-system in a later section.

[1] This equation can be derived using techniques from Physics 7A by writing out an $F = ma$ equation. Often times, we will give you the dynamics of a system since we don't assume physics as a prerequisite to this class.

3 Discrete-Time Systems

Another important type of system to look at is are discrete-time systems. A **discrete-time system** is one in which all of the input, output, and state-variables are functions in discrete time. This means that the all of our functions will map natural numbers $\mathbb{N} = (0, 1, 2, \ldots)$ to the real numbers $\mathbb{R}$. Similar to how we described continuous-time systems through differential equations, we can describe discrete-time systems through **difference equations**.

$$\vec{x}[n+1] = f(\vec{x}[n], \vec{u}[n]) \tag{5}$$

where $f(\vec{x}[t]$ is an length n-vector that dictates the derivatives of each state variable according to the current value of the states. The form of f depends on the system we are modeling as we will see in examples.

3.1 Notation

To distinguish between a discrete-time and continuous time system, we will use square brackets $\vec{x}[n]$ for discrete-time systems whereas continuous time systems will use the normal round brackets, $\vec{x}(t)$. In addition to this, we will index our discrete time variable with the letter n to denote the natural numbers.

3.2 Difference Equations

The discrete-time analog of differentiation $\frac{d}{dt}$ is a time delay $\vec{x}[n+1]$. Therefore, we can write out a first-order scalar difference equation of the form

$$x[n+1] = \alpha x[n] + \beta u[n] \tag{6}$$

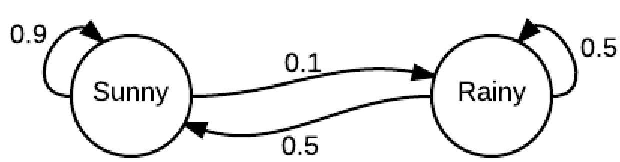
where $u[n]$ is a discrete time input. There is an entire study of solving difference equations that is connected to solving differential equations, but we will not be emphasizing this in this course. However, we will note that when $u[n] = 0, x[n] = \alpha^n$ is a solution. Try to verify this on your own.

3.3 Motivation

Since the universe runs in continuous time, one might ask what the motivation behind discrete-time systems is. While many systems are represented by continuous-time systems, computers are fundamentally discrete. This means if we would like to provide inputs into our system, we would like to work with discrete time systems. If the original system is continuous, we will have to *sample* data from this continuous time system to create a discrete-time system. We will explore how to do this in a later note.

3.4 Example

Suppose we had a model that could predict the whether the weather will be sunny or rainy depending on the current day's weather.



The arrows on the graph represent the probability of transitioning from a sunny to rainy day and vice-versa.

We could define the states $x_1[n]$ and $x_2[n]$ which represent the probability that it is sunny and rainy on a given day n. Then based on the graph, we can write the following recurrence relation

$$x_1[n+1] = 0.9x_1[n] + 0.5x_2[n] \tag{7}$$
$$x_2[n+1] = 0.1x_1[n] + 0.5x_2[n] \tag{8}$$

To express this in the form of Equation , we can express the function f as

$$f(\vec{x}[n]) = \begin{bmatrix} 0.9x_1[n] + 0.5x_2[n] \\ 0.1x_1[n] + 0.5x_2[n] \end{bmatrix} = \begin{bmatrix} 0.9 & 0.5 \\ 0.1 & 0.5 \end{bmatrix} \vec{x}[n] \tag{9}$$

Note how we are able to write a matrix equation $\vec{x}[n+1] = A\vec{x}[n]$. This is unrelated to the fact that the system is discrete, rather it is because the system is *linear.*

4 Linear Systems

A function f is linear if it satisfies the two properties:

$$\textbf{1. Scaling: } f(\alpha x) = \alpha f(x) \tag{10}$$
$$\textbf{2. Additivity: } f(x+y) = f(x) + f(y) \tag{11}$$

So far, we have spent a great deal of time studying linear systems, but we've never bothered to call them linear. In fact, any system represented by a matrix $f(\vec{x}) = A\vec{x}$ is a linear system since f satisfies the two properties above.

4.1 Examples

We will give some examples of linear and nonlinear functions.

Function	Linear	Non-linear
$f(x) = 2x$	Yes	No
$f(x) = 3x+5$	No	Yes
$f(x) = \sin(x)$	No	Yes
$f(x) = x^2$	No	Yes
$f(x,u) = \alpha x + \beta u$	Yes	No
$f(\vec{x}) = A\vec{x}$	Yes	No
$f(\vec{x}) = A\vec{x} + \vec{b}$	No	Yes
$f(\vec{x},\vec{u}) = A\vec{x} + B\vec{u}$	Yes	No

For a continuous or discrete-time system represented by the equation $\frac{d}{dt}\vec{x} = f(\vec{x},\vec{u})$ or $\vec{x}[n+1] = f(\vec{x},\vec{u})$, we say that the system is linear if f is a linear function.

4.2 Motivation

Why do we care about linear systems and what do we do if our system is non-linear? Linear systems are very nice since they often have predictable behavior. We will be developing a large toolbox on how to analyze and control these linear systems. On the other hand, nonlinear systems often have unpredictable behavior and will be more difficult to control.

Note that many systems in the real world are non-linear often due to factors such as noise, disturbances, or internal resistive forces. If a system is non-linear, then we will try to find a small region where the system is in fact linear. This technique is called **linearization** and will be the focus of the next note.

Contributors:

- Murat Arcak.
- Taejin Hwang.
- Justin Yim.

