Universidad Complutense de Madrid

Sistemas Dinámicos y Realimentación

Juan Jiménez
Héctor García de Marina

15 de noviembre de 2020

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Prefacio

Capítulo 2

Modelado de sistemas dinámicos

Warning: Section in progress

2.1. State-space systems

We will focus on systems that can be described by quantifiable characteristics or states, e.g., temperature, velocity or voltage. These states might change over time. One might interact with a system via a quantifiable input, and one might measure some information from the states of the system via a quantifiable output.

Let us define $x(t) \in \mathbb{R}^n$, $y(t) \in \mathbb{R}^m$ and $u(t) \in \mathbb{R}^k$ as the stacked (signal) vector of states, the output, and the input of a system Σ respectively. In particular, we will describe them as continuous signals over time, e.g., $x : [0, \infty) \to \mathbb{R}^n$.

The system Σ is a model that predicts the value of the states and the output over time. This prediction incorporates the impact of the input on the states and the output. We utilize differential equations as a tool to predict the evolution of the states of the system Σ over time as follows

$$\Sigma := \begin{cases} \dot{x}(t) = & f(x(t), u(t)) \\ y(t) = & g(x(t), u(t)) \end{cases}, \tag{2.1}$$

where $\dot{x} := \frac{\mathrm{d}}{\mathrm{dt}}(x(t))$ is the short notation for total derivative with respect to time, and $f : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}^n$ and $g : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}^m$ are functions.

We can represent the system Σ as a block with input/output ports as in figure 2.1.

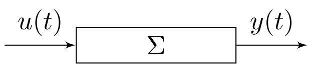

Figura 2.1: Input/output block diagram of system Σ.

2.2. Ejemplos

2.2.1. Inverted pendulum

We are going to derive f and g for the inverted pendulum.

First, we derive the equations of motion of the mass in the inverted pendulum system in figure 2.2 as a first step to figure out the system's functions f and g. Consider that we can interact with the system with a torque T applied on the base, the mass is under a friction force proportional to its speed, and we can only measure the angle θ from the system.

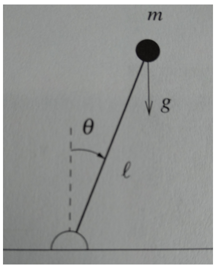

Figura 2.2: Inverted pendulum

We choose the angle θ with respect to the vertical to derive the dynamics of the inverted pendulum. Define m, l, g, and $I \in \mathbb{R}$ as the mass, pendulum's lenght, gravity acceleration and inertia moment respectively. In particular, we have that $I = ml^2$, and we will exploit that $I\ddot{\theta} =$ sum of torques. We have to consider three torques; namely: 1. Torque T applied by us; 2. Torque $-b\dot{\theta}$ applied by the friction; 3. Torque $mgl\sin\theta$ applied by the gravitational forces, i.e., mg is the force on the mass, times $\sin\theta$ since the force only applies perpendicular to the bar, and times l to compute the resultant torque. Therefore the dynamics of the inverted pendulum are given by

$$\ddot{\theta} = \frac{1}{ml^2}\left(mgl\sin\theta - b\dot{\theta} + T\right). \tag{2.2}$$

It looks reasonable to choose θ as one of our states to construct $x(t)$. In fact, since we have derived a second-order system, we will need $\dot{\theta}$ as a second state since we have the differential equation for its derivative $\ddot{\theta}$. Thus, let us define the state vector

$$x := \begin{bmatrix}\theta \\ \dot{\theta}\end{bmatrix}, \tag{2.3}$$

and since the torque T is how we interact with the system, we choose the input $u(t) = T(t)$.

Now we are ready to construct the functions f and g in (2.1) for the inverted pendulum. In particular, we know that f and g only accepts as inputs/arguments the state space vector x and the input u. On the left side of (2.1) we have the time derivative of $x(t)$, therefore

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\begin{bmatrix}\theta \\ \dot{\theta}\end{bmatrix}\right) = f(x(t), u(t)) = \begin{bmatrix}f_1(x(t), u(t)) \\ f_2(x(t), u(t))\end{bmatrix}, \tag{2.4}$$

where $f_1 = \dot{\theta}$. Note that for the first row in (2.4), on the left we have $\frac{\mathrm{d}}{\mathrm{dt}}\theta$, and on the right $f_1 = \dot{\theta}$ because $\dot{\theta}$ is actually a state, roughly speaking we can say that $\dot{\theta}(t) \in x(t)$. Unfortunatelly, we cannot say that $f_2 = \ddot{\theta}(t)$ because (roughly speaking) $\ddot{\theta}(t) \notin x(t)$. Nevertheless, we have that f_2 is given by the differential equation (2.2). Then, let me write explicitly f as follows

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\begin{bmatrix}\theta \\ \dot{\theta}\end{bmatrix}\right) = f(x(t), u(t)) = \begin{bmatrix}\dot{\theta} \\ \frac{1}{ml^2}\left(mgl\sin\theta - b\dot{\theta} + T\right)\end{bmatrix}, \tag{2.5}$$

The calculation of g is more straightforward. We have established that we can only measure the angle, therefore $y(t) = \theta(t)$, i.e.,

$$g(x(t), u(t)) = \theta(t). \tag{2.6}$$

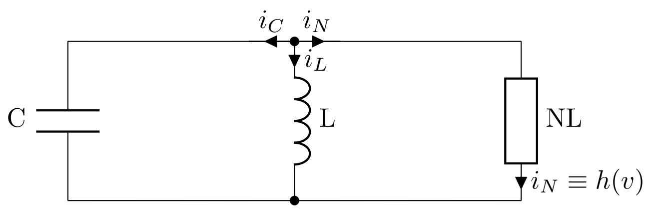

Figura 2.3: Circuito eléctrico no lineal

A Python simulation of this dynamics can be found at https://github.com/noether/aut_course.

2.2.2. El oscilador de Van der Pol

Se trata de un oscilador, propuesto por primera vez por Balthasar Van der Pol, cuando trabajaba en Philips, para explicar las oscilaciones observadas en tubos de vacío. Podemos obtener la ecuación del oscilador, empleando el circuito de la figura 2.2.2

Donde el elemento no lineal NL, presenta una relación entre voltaje e intensidad caracterizada por la función $h(v)$.

El voltaje v en los tres componentes del circuito debe ser igual; además,

$$v = L\frac{di_L}{dt} \tag{2.7}$$

$$i_C = C\frac{dv}{dt} \tag{2.8}$$

Si aplicamos la primera ley de Kirchohff al nodo superior del circuito,

$$i_C + i_L + i_N = 0 \tag{2.9}$$

$$C\frac{dv}{dt} + \frac{1}{L}\int_{-\infty}^{t} v(s)ds + h(v) = 0 \tag{2.10}$$

Si derivamos 2.10 con respecto al tiempo, dividimos por C y reordenamos,

$$\frac{d^2v}{dt^2} + \frac{1}{C}\frac{dh(v)}{dv}\frac{dv}{dt} + \frac{1}{LC}\cdot v = 0 \tag{2.11}$$

Se trata de un caso particular de la ecuación de Liénard,

$$\ddot{v} + f(v)\dot{v} + g(v) = 0 \tag{2.12}$$

Si definimos ahora, $h(v)$,

$$h(v) = m(\frac{1}{3}v^3 - 1) \tag{2.13}$$

$$\frac{dh}{dv} = m(v^2 - 1) \tag{2.14}$$

y sustituyendo en 2.11,

$$\ddot{v} + m\frac{1}{C}(v^2 - 1)\dot{v} + \frac{1}{LC}v = 0 \tag{2.15}$$

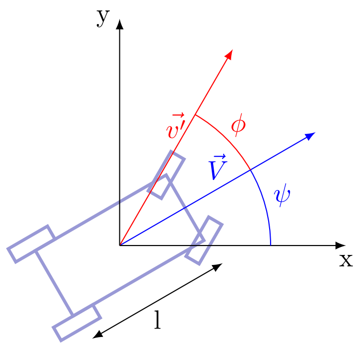

Figura 2.4: Esquema de un vehículo terrestre de 4 ruedas

Podemos representarla finalmente en variables de estado, tomando $x_1 = v$ y $x_2 = \dot{v}$,

$$\dot{x}_1 = x_2 \tag{2.16}$$
$$\dot{x}_2 = -\frac{1}{LC}x_1 - m\frac{1}{C}(x_1^2 - 1)x_2 \tag{2.17}$$

Podemos ahora hacer un primer análisis cualitativo. El primer término a la derecha del igual en la ecuación 2.17 representa una fuerza recuperadora proporcional al desplazamiento, el segundo término, crecerá con la velocidad para $x_1 < 1$, alejando así al sistema del origen y representará un término disipativo para $x_1 > 1$ acercándolo por tanto de nuevo al origen. Es por tanto esperable, que se alcance algún tipo de situación de equilibrio. Más adelante definiremos esta situación rigurosamente como un ciclo límite.

2.2.3. Un vehículo de cuatro ruedas.

La figura 2.2.3 muestra un esquema de un vehículo terrestre de cuatro ruedas, visto desde arriba. Si consideramos que se mueve en el plano x, y, y que su velocidad instántanea $\vec{V}$ está siempre orientada en la dirección de avance del vehículo *psi*–asumimos que no derrapa, ni se mueve lateralmente–, podemos entoces definir el velocidad, en el sistema de referencia x, y como,

$$\dot{x} = V_x(t) = V\cos(\psi(t)) \tag{2.18}$$
$$\dot{y} = V_y(t) = V\sin(\psi(t)) \tag{2.19}$$
$$\tag{2.20}$$

Además el vehículo girará, siempre que las ruedas delanteras no estén alineadas con las ruedas traseras, cambiando así su dirección de avance . Podemos relacionar la velocidad de giro del vehículo $\dot{\psi}$ con el ángulo de orientación de las ruedas delanteras ϕ, y la velocidad a la que avanzan $\vec{v'}$. podemos obtener las componetes de dicha velocidad en ejes cuerpo (paralela y perpendicular a la dirección de avance del vehículo),

$$v_P = v'\cos(\phi(t)) \tag{2.21}$$
$$v_T = v'\sin(\phi(t)) \tag{2.22}$$

Pero la rueda esta unida al vehículo así que su velocidad en la direccíon de avance debe ser la misma que la del vehículo: $v_P \equiv V$. A partir de esta relación podemos obtener la velocidad tangencial de las ruedas como,

$$v_T = V\frac{\sin(\phi)}{\cos(\phi)} = V\tan(\phi) \tag{2.23}$$

Si tomamos como centro de giro del vehículo el centro de su eje trasero, y la batalla (distancia entre ejes) es l, obtenemos una expresión para su velocidad de giro,

$$\dot{\psi} = \frac{V}{l}\tan(\phi) \tag{2.24}$$

En resumen, podemos describir el sistema mediante tres ecuaciones de estado $x_1 \equiv x, x_2 \equiv y, x_3 \equiv \psi$:

$$\dot{x_1} = V\cos(x_3) \tag{2.25}$$
$$\dot{x_2} = V\sin(x_3) \tag{2.26}$$
$$\dot{x_3} = \frac{V}{l}\tan(\phi) \tag{2.27}$$

Si consideramos $V = cte$, la única entrada al sistema sería el ángulo de giro de las ruedas $u(t) = \phi(t)$, controlando su valor, podemos hacer girar al vehículo en la dirección deseada.

2.3. Simulations / Numerical solutions

We can still calculate numerical solutions (also known as simulations) for Σ given a starting point $x(0)$. The *Euler integration* is an easy numerical method that can give us some information about Σ. The following algorithm is what you can use in your Python/Matlab simulations

Algorithm 1. *1. Set step time ΔT*

2. *Set $x = x(0)$*
3. *Set $y = g(x, u)$*
4. *Log x and y, so you can plot them later*
5. *Set $t = 0$*
6. *Set final time T^**
7. *While $t \leq T^*$ then:*
   a) *Set $x_{new} = x_{old} + f(x_{old}, u)\Delta T$*
   b) *Set $y_{new} = g(x_{new}, u)$*
   c) *Draw x*
   d) *Log x and y, so you can plot them later*
   e) *Set $t = t + \Delta t$*
8. *Plot the log for the elements of x and y over time t*

This algorithm performs okei when ΔT is sufficiently small. How small? It always depends on the system Σ, in particular, of f. Check https://en.wikipedia.org/wiki/Euler_method for more details, and of course, for more accurate methods. There are always compromises, typically good accuracy entails more computational cost per iteration.

For now, in the simulation we will leave $u = 0$, i.e., no control action over the system Σ. Once we know how to design u, we will calculate u before the step 7.a in Algorithm 1.

Capítulo 3

Comportamiento dinámico y estabilidad

Capítulo 4

Sistema Lineales

4.1. Linear maps

In this chapter, we focus on a particular class of state-space systems called *state-space linear systems*. First, we need the notion of *linear map*.

Definition 1. *Consider the mapping $H : V \to W$. If H preserves the operations of addition and scalar multiplication, i.e.,*

$$H(v_1 + v_2) = H(v_1) + H(v_2), \quad v_1, v_2 \in \mathbb{V}$$
$$H(\alpha v_1) = \alpha H(v_1), \quad \alpha \in \mathbb{K},$$

then H is a linear map.

4.1.1. Exercise: Check whether the following maps are linear or not

1. $H_1(v) := Av, A \in \mathbb{R}^{n\times n}, \quad v \in \mathbb{R}^n$
2. $H_2(v) := \frac{\mathrm{d}}{\mathrm{dt}}(v(t)), \quad v \in \mathcal{C}^1$
3. $H_3(v) := \int_0^T v(t)dt, \quad v \in \mathcal{C}^1, T \in \mathbb{R}_{\geq 0}$
4. $H_4(v) := D(v) := v(t-T), \quad v \in \mathcal{C}^1, T \in \mathbb{R}_{\geq 0}$
5. $H_5(v) := Av + b, \quad A \in \mathbb{R}^{n\times n}, v, b \in \mathbb{R}^n$

4.2. Continuous state-space linear systems

The following system defines a continuous state-space linear system

$$\Sigma := \begin{cases} \dot{x}(t) & = A(t)x(t) + B(t)u(t), \quad x \in \mathbb{R}^n, u \in \mathbb{R}^k \\ \dot{y}(t) & = C(t)x(t) + D(t)u(t), \quad y \in \mathbb{R}^m \end{cases} \tag{4.1}$$

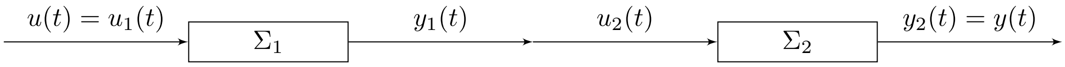

Figura 4.1: Example of series interconnection.

4.2.1. Exercise: Write as a block diagram the continuous state-space linear system and check that consists only of linear maps

4.2.2. Exercise: Interconnections of continuous state-space linear systems

Rewrite as a single system, i.e., as in (4.1):

1. the series (or cascade) interconnection of two continuous state-space linear systems, i.e., $y_1(t) = u_2(t)$.
2. the parallel interconnection of two continuous state-space linear systems, i.e., $y(t) = y_1(t) + y_2(t)$.
3. the feedback interconnection, i.e., $u_1(t) = u(t) - y(t)$, assuming $u, y, \in \mathbb{R}^k$.

4.3. Solution to Linear State-Space systems

The solution to an *ordinary differential equation* (ODE) is given by the addition of two terms: the solution to the homogeneous part, and a particular solution to the non-homogeneous.

$$\dot{x}(t) = \underbrace{A(t)x(t)}_{\text{homogeneous}} + \underbrace{B(t)u(t)}_{\text{non-homogeneous}} \tag{4.2}$$

Theorem 1. *Peano-Barker series The unique solution to the homogeneous $\dot{x} = Ax$ is given by*

$$x(t) = \Phi(t, t_0)x(t_0), \quad x(t_0) \in \mathbb{R}^n, t \geq 0, \tag{4.3}$$

where

$$\begin{aligned}\Phi(t, t_0) := I &+ \int_{t_0}^{t} A(s_1)ds_1 + \int_{t_0}^{t} A(s_1) \int_{t_0}^{s_1} A(s_2)ds_2 ds_1 \\ &+ \int_{t_0}^{t} A(s_1) \int_{t_0}^{s_1} A(s_2) \int_{t_0}^{s_2} A(s_3)ds_3 ds_2 ds_1 + \dots .\end{aligned} \tag{4.4}$$

Sketch of the proof: First we calculate the following time derivative

$$\begin{aligned}\frac{d}{dt}\Phi(t, t_0) &= A(t) + A(t)\int_{t_0}^{t} A(s_2)ds_2 \\ &+ A(t)\int_{t_0}^{t} A(s_2)\int_{t_0}^{s_2} A(s_3)ds_3 ds_2 + \dots \\ &= A(t)\Phi(t, t_0).\end{aligned} \tag{4.5}$$

We claim that the solution to the homogenenous part of (4.2) is $x(t) = \Phi(t, t_0)x_0$ (x_0 is the short notation for $x(t_0)$), whose time derivative is given by

$$\begin{aligned} \frac{d}{dt}x &= \frac{d}{dt}\Phi(t, t_0)x_0 \\ &= A(t)\Phi(t, t_0)x_0 \\ &= A(t)x(t), \end{aligned} \tag{4.6}$$

which is proving the identity $\dot{x} = A(t)x(t)$ given that $x(t) = \Phi(t, t_0)x_0$. In order to make this proof complete, we would need to prove that the series (4.4) converges for $t \geq t_0$. That material should be covered in a standard course on differential equations.

The matrix $\Phi(t, t_0)$ is called the ***state transition matrix***. Given an initial condition x_0, we can predict $x(t)$ in (4.2) by *iterating* over and over with $\Phi(t, t_0)$ given that we do not interact with the system, i.e., $u(t) = 0, t \geq t_0$.

4.3.1. Exercise

Check that

$$\begin{aligned} x(t) &= \Phi(t, t_0)x_0 + \int_{t_0}^{t} \Phi(t, \tau)B(\tau)u(\tau)d\tau \\ y(t) &= C(t)\phi(t, t_0)x_0 + \int_{t_0}^{t} C(t)\Phi(t, \tau)B(\tau)u(\tau)d\tau + D(t)u(t) \end{aligned}$$

are the solutions to

$$\begin{aligned} \dot{x}(t) &= A(t)x(t) + B(t)u(t) \\ \dot{y}(t) &= C(t)x(t) + D(t)u(t) \end{aligned}$$

4.4. Solution to Linear Time Invariant Systems

The matrix $\Phi(t, t_0)$ can be calculated analytically when A is a matrix with constant coefficients. If A is constant, we can take it out from the integrals in (4.4)

$$\begin{aligned} \Phi(t, t_0) := I + A\int_{t_0}^{t} ds_1 + A^2 \int_{t_0}^{t}\int_{t_0}^{s_1} ds_2 ds_1 \\ + A^3 \int_{t_0}^{t}\int_{t_0}^{s_1}\int_{t_0}^{s_2} ds_3 ds_2 ds_1 + \dots, \end{aligned} \tag{4.7}$$

and noting that the following integrals can be easily solved

$$\begin{aligned} \int_{t_0}^{t} ds_1 &= (t - t_0) \\ \int_{t_0}^{t}\int_{t_0}^{s_1} ds_2 ds_1 &= \frac{(t-t_0)^2}{2} \\ &\vdots \\ \int_{t_0}^{t}\int_{t_0}^{s_1}\cdots\int_{t_0}^{s_{k-2}}\int_{t_0}^{s_{k-1}} ds_k ds_{k-1}\cdots ds_2 ds_1 &= \frac{(t-t_0)^k}{k!}, \end{aligned}$$

then we have that (4.7) can be calculated by

$$\Phi(t,t_0) = \sum_{k=0}^{\infty} \frac{(t-t_0)^k}{k!} A^k, \tag{4.8}$$

which resembles to the power series of the scalar exponential function, i.e., $e^x := \sum_{k=0}^{\infty} \frac{1}{k!} x^k = 1 + x + \frac{x^2}{2} + \frac{x^3}{3!} + \dots$. In fact, the definition of the *exponential of a matrix* is

$$exp(A) = I + A + \frac{1}{2}A^2 + \frac{1}{3!}A^3 + \dots \tag{4.9}$$

Let us set $t_0 = 0$ for the sake of convinience, then

$$\begin{aligned} \Phi(t,0) &= I + tA + \frac{t^2}{2}A^2 + \frac{t^3}{3!}A^3 + \dots \\ &= exp(At), \end{aligned} \tag{4.10}$$

therefore the solution to the homogeneous (4.2) with A constant and setting $t_0 = 0$ is

$$x(t) = exp(At)x_0, \quad t \geq 0. \tag{4.11}$$

To continue further, we need the following result from Linear Algebra.

Theorem 2. ***Jordan Form***. *For every square matrix $A \in \mathbb{C}^{n\times n}$, there exists a non-singular change of basis matrix $P \in \mathbb{C}^{n\times n}$ that transform A into*

$$J = PAP^{-1} = \begin{bmatrix} J_1 & 0 & 0 & \dots & 0 \\ 0 & J_2 & 0 & \dots & 0 \\ 0 & 0 & J_3 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \cdots & J_l \end{bmatrix}, \tag{4.12}$$

where each J_i is a Jordan block of the form

$$J_i = \begin{bmatrix} \lambda_i & 1 & 0 & \dots & 0 \\ 0 & \lambda_i & 1 & \dots & 0 \\ 0 & 0 & \lambda_i & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_i \end{bmatrix}_{n_i \times n_i}, \tag{4.13}$$

*where each λ_i is an eigenvalue of A, and the number l of Jordan blocks is equal to the total number of independent eigenvectors of A. The matrix J is unique up to a reordering of the Jordan blocks and is called the **Jordan normal form** of A.*

Note that $A = P^{-1}JP$ as well, and we leave as an exercise to prove that

$$A^k = P^{-1}J^kP, \tag{4.14}$$

so we can calculate

$$
\begin{aligned}
exp(At) &= P^{-1}\left(\sum_{k=1}^{\infty}\frac{t^k}{k!}\begin{bmatrix} J_1^k & 0 & \cdots & 0 \\ 0 & J_2^k & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & J_l^k \end{bmatrix}\right)P \\
&= P^{-1}\begin{bmatrix} exp(J_1 t) & 0 & \cdots & 0 \\ 0 & exp(J_2 t) & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & exp(J_l t) \end{bmatrix}P
\end{aligned} \tag{4.15}
$$

Therefore if J is just a diagonal matrix with the eigenvalues of A, i.e., $J_l = \lambda_l \in \mathbb{C}$, then $exp(J_l t) = e^{\lambda_l t} \in \mathbb{C}$ is a trivial calculation.

Now, let us check the consequencues on the following two conditions

1. J is diagonal.

2. All the eigenvalues of A have negative real part.

Knowing that $\text{lím}_{t\to\infty} e^{\lambda t} \to 0$ if $\lambda \in \mathbb{R}_{<0}$, then we will have that $exp(At) \to 0$ as $t \to \infty$ if the previous two conditions are satisfied! So if we take a look at (4.11), we can conclude that

$$
\operatorname*{lím}_{t\to\infty} x(t) \to 0, \tag{4.16}
$$

therefore we can make a prediction on the evolution of $x(t)$ by just checking the eigenvalues of A. If J is not diagonal we can also conclude similar results, but we will not cover them here. We will talk about stability in the next lecture, and how to design a controller such that we can guarantee (4.16).

4.5. Linearization of state-space systems

Unfortunately, it is really (really) hard to calculate the analytic solution of $x(t)$ and $y(t)$ for a generic system Σ. Nevertheless, we will see that we can find the analytic solution for a state-space linear system.

The question then is whether we can relate a generic Σ to a state-space linear system.

If $f(x,t)$ and $g(x,t)$ are real analytic around a specific point (x^*, u^*), then we can approximate them around (x^*, u^*) by a Taylor series expansion. This approximation is what we call *linearization* if we stop at order one in the Taylor series

$$
\Sigma := \left.\begin{cases} \dot{x}(t) = & f(x(t), u(t)) \\ y(t) = & g(x(t), u(t)) \end{cases}\right|_{x\approx x^*, u\approx u^*} \approx \begin{cases} x(t) & = x^* + \delta x(t) \\ u(t) & = u^* + \delta u(t) \\ \delta\dot{x}(t) & = A(t)\delta x(t) + B(t)\delta u(t) \\ \delta y(t) & = C(t)\delta x(t) + D(t)\delta u(t) \end{cases},
$$

where

$$A(t) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{|x=x^*,u=u^*} \qquad B(t) = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_k} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_k}{\partial u_1} & \cdots & \frac{\partial f_k}{\partial u_k} \end{bmatrix}_{|x=x^*,u=u^*}$$
$$C(t) = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_n} \\ \vdots & \vdots & \vdots \\ \frac{\partial g_m}{\partial x_1} & \cdots & \frac{\partial g_m}{\partial x_n} \end{bmatrix}_{|x=x^*,u=u^*} \qquad D(t) = \begin{bmatrix} \frac{\partial g_1}{\partial u_1} & \cdots & \frac{\partial g_1}{\partial u_k} \\ \vdots & \vdots & \vdots \\ \frac{\partial g_m}{\partial u_1} & \cdots & \frac{\partial g_m}{\partial u_k} \end{bmatrix}_{|x=x^*,u=u^*}.$$

Roughly speaking, we calculate the sensitivity (up to first order) of f and g when we make a small variation on x and u around (x^*, u^*). How close (x, u) must be to (x^*, u^*) depends on the particular system Σ. Later in the course, we will provide bounds for δx and δu such that we can apply with guarantees our control algorithms.

4.5.1. Linearization of the inverted pendulum

We will see that, with the linearization, we can design controllers $u(t)$, i.e., a signal that our torque T must follow, to drive the state of the pendulum where we wish. Let us define this point of interest as $x^* = \begin{bmatrix} \theta^* \\ 0 \end{bmatrix}$, i.e., a fixed angle with (obviously) zero velocity. Indeed, this is an equilibrium point for the angle θ. In order to have an equilibrium, we need to find a $u(t)$ in (2.5) such that $\frac{\mathrm{d}}{\mathrm{dt}}\left(\begin{bmatrix} \theta \\ \dot\theta \end{bmatrix}\right) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$. A quick inspection to the dynamics (2.2) we have that

$$u^* = T^* = -\frac{g}{l}\sin\theta^*, \tag{4.17}$$

for example, for the vertical position of the pendulum corresponding to $\theta^* = 0$ we have that $T^* = 0$, i.e., $x^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and $u^* = 0$.

The calculation of the matrices A, B, C, and D are the corresponding Jacobians for (2,5) and (2,6), i.e.,

$$\begin{aligned}
\frac{\partial f_1}{\partial x_1} &= 0 \\
\frac{\partial f_1}{\partial x_2} &= 1 \\
\frac{\partial f_2}{\partial x_1} &= \frac{g}{l}\cos\theta \\
\frac{\partial f_2}{\partial x_2} &= -\frac{b}{ml^2} \\
\frac{\partial f_1}{\partial u_1} &= 0 \\
\frac{\partial f_2}{\partial u_1} &= 1 \\
\frac{\partial g_1}{\partial x_1} &= 1 \\
\frac{\partial g_1}{\partial x_2} &= 0 \\
\frac{\partial g_1}{\partial u_1} &= 0,
\end{aligned}$$

therefore we can arrive at

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{dt}}\left(\begin{bmatrix}\delta\theta \\ \dot{\delta\theta}\end{bmatrix}\right) &= \begin{bmatrix}0 & 1 \\ \frac{g}{l}\cos\theta & -\frac{b}{ml^2}\end{bmatrix}_{|\theta=\theta^*}\begin{bmatrix}\delta\theta \\ \dot{\delta\theta}\end{bmatrix} + \begin{bmatrix}0 \\ 1\end{bmatrix}\delta T \\
\delta y &= \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\delta\theta \\ \dot{\delta\theta}\end{bmatrix} + 0\,\delta T,
\end{aligned} \tag{4.18}$$

to model the dynamics of $x(t)$ and the output $y(t)$ around the points x^* and u^*.

Finally, we would like to hightlight that the Jacobians can have time-varying elements, and still have a linear system. For example, we can consider that the lenght l depends on the time explicitly, e.g., $l(t) = l + \sin(t)$. In such a case, we would have a $A(t)$.

4.6. Stability of nonlinear systems through linearization

Capítulo 5

Control por realimentación de estados

5.1. Guideline to design a linear controller for the inverted pendulum

1.

Given the model Σ of a dynamical system

$$\Sigma := \begin{cases} \dot{x}(t) = & f(x(t), u(t)) \\ y(t) = & g(x(t), u(t)) \end{cases},$$

choose an operational/equilibrium point x^* of interest.

For the pendulum, let us choose when it is in vertical position and at rest, i.e., $\theta^* = 0$, $\dot{\theta}^* = 0$. So you we have that $x^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$.

2.

Since x^* is an equilibrium, find out which u^* makes $\dot{x}(t) = 0$, i.e., $f(x^*, u^*) = 0$.
We have that $u = T$, and that $\ddot{\theta} = \frac{1}{ml^2}\left(mgl\sin\theta - b\dot{\theta} + T\right)$. To keep x^* fixed, we need to set $\ddot{\theta} = 0$, therefore $u^* = T^* = 0$. Note that for another x^*, we would have different T^*.

3.

Now we want the system around x^* and u^* to be an stable equilibrium, i.e., for a small deviation/disturbance δx, we need to calculate the necessary δu to keep the system at x^*.

We calculate the dynamics of δx and δu, i.e., we linearize Σ around x^* and u^*.

$$\Sigma := \begin{cases} \dot{x}(t) = & f(x(t), u(t)) \\ y(t) = & g(x(t), u(t)) \end{cases} \approx \begin{cases} \dot{\delta x}(t) & = A(t)\delta x(t) + B(t)\delta u(t) \\ \dot{\delta y}(t) & = C(t)\delta x(t) + D(t)\delta u(t) \end{cases} \quad \text{if } x \approx x^* + \delta x, u \approx u^* + \delta u,$$

where the matrices $A(t), B(t), C(t)$ and $D(t)$ are the Jacobians from Week 14. Note that it is usual to set the origin of the coordinates x the system at x^*, this is why you will find in many places

(including Week 14) $\delta x = x$ for the linearized version of Σ.
The Jacobians for the inverted pendulum are

$$
\begin{aligned}
A &= \begin{bmatrix} 0 & 1 \\ \frac{1}{ml^2}(mgl\cos\theta) & -\frac{b}{ml^2} \end{bmatrix} \\
B &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
C &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\
D &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}.
\end{aligned} \tag{5.1}
$$

Note that A has to be evaluated at x^* (check the notes from Week 14). Therefore, for $\theta = 0$, we have that $A = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} & -\frac{b}{ml^2} \end{bmatrix}$. We assume that $C = I$, i.e., we can measure all the elements from the state vector x.

4.

Let us calculate the linear controller

$$\delta u = K\delta y, \tag{5.2}$$

such that x^* is stable. We substitute (5.2) in the linearized Σ resulting in

$$
\begin{aligned}
\delta \dot{x}(t) &= A(t)\delta x(t) + B(t)K\delta y \\
&= A(t)\delta x(t) + B(t)KC(t)\delta x(t) + KD(t)\delta u(t) \\
&= (A(t) + B(t)KC(t))\, x(t) + KD(t)u(t)
\end{aligned} \tag{5.3}
$$

Consider that $D(t) = 0$, and $A(t)$ and $B(t)$ are constant matrices, i.e., their elements do not depend on time. Then, we can write (5.3) as

$$\delta \dot{x}(t) = (A + BKC)\, x(t) \tag{5.4}$$
$$= Mx(t). \tag{5.5}$$

Then, the linearized system Σ is stable around x^* under small disturbances if and only if M has all its eigenvalues with negative real part (Week 15). To have the addition $A + BKC$, we need K with the appropriate dimensions. For $C = I$ we have that $K = \begin{bmatrix} k_{11} & k_{12} \end{bmatrix}$, so we have that

$$M = A + KBC = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} + \frac{k_{11}}{ml^2} & -\frac{b}{ml^2} + \frac{k_{12}}{ml^2} \end{bmatrix} \tag{5.6}$$

5.

A matrix $M \in \mathbb{R}^{n\times n}$ has n eigenvalues. The eigenvalues of M can be calculated from the following determinant

$$\det\{M - \lambda_i I\} = 0, \quad i \in \{1, \dots, n\}. \tag{5.7}$$

For example, for a 2×2 matrix we have that $\det\{A\} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = a_{11}a_{22} - a_{12}a_{21}$. Therefore we have that (5.7) is

$$(m_{11} - \lambda_i)(m_{22} - \lambda_i) - m_{12}m_{21} = 0, \quad i \in \{1, 2\}. \tag{5.8}$$

The values for the elements of K are calculated by setting an arbitrary $\lambda_i < 0$. This solution is guaranteed for $C = I$ and $D = 0$.
We have that (5.8) from M in (5.6) is

$$\lambda^2 + \lambda\left(\frac{1}{ml^2}(b - k_{12})\right) - \frac{g}{l} - \frac{k_{11}}{ml^2},$$

whose solution is given by

$$\lambda_{1,2} = \frac{-\frac{1}{ml^2}(b - k_{12}) \pm \sqrt{\frac{(b-k_{12})^2}{m^2l^4} + 4(\frac{g}{l} + \frac{k_{11}}{ml^2})}}{2}. \tag{5.9}$$

Let us find some conditions for k_{11} and k_{12} such that we can guarantee that λ_1 and λ_2 are two real negative numbers. For example, if force

$$k_{12} < b, \tag{5.10}$$

then $-\frac{1}{ml^2}(b-k_{12})$ in (5.9) is a negative number. Note that b is a coefficient friction in the pendulum equation, therefore if $b = 1$, we can have $-\infty < k_{12} < 1$, i.e., the gain k_{12} can be even positive as long as it is smaller than b. Now we turn our attention at the square root in (5.9). Assume that we take the positive solution $r > 0$ of the square root in (5.9). Now we need to add or subtract r to $-\frac{1}{ml^2}(b - k_{12})$. We note that for λ_1 if $-\frac{1}{ml^2}(b - k_{12})$ is negative then $-\frac{1}{ml^2}(b - k_{12}) - r$ is still negative, so $\lambda_1 < 0$. For λ_2 we need to calculate k_{11} such that $-\frac{1}{ml^2}(b - k_{12}) + r < 0$. If we set $k_{11} < -gml$ then $\sqrt{\frac{(b-k_{12})^2}{m^2l^4} + 4(\frac{g}{l} + \frac{k_{11}}{ml^2})} < \sqrt{\frac{(b-k_{12})^2}{m^2l^4}} = \frac{1}{ml^2}(b - k_{12})$. Therefore we guarantee that $r < \frac{1}{ml^2}(b - k_{12})$, thus $\lambda_2 < 0$. Note that k_{11} not only needs to be negative but *negative enough*. Check in the Python script the consequences of playing around these limits for k_{11} and k_{12}.

5.2. Controller for the inverted pendulum

Design a controller for

$$x_1^* = \begin{cases} \theta^* & = 0 \\ \dot{\theta}^* & = 0 \end{cases}$$

$$x_2^* = \begin{cases} \theta^* & = \frac{\pi}{4} \\ \dot{\theta}^* & = 0 \end{cases}.$$

We will first assume that we can measure θ and $\dot{\theta}$, i.e., $C = I$. Note that for $M = (A - BKC)$ with $C = I$ the dimensions of K must be 1×2, i.e., $K = \begin{bmatrix} k_{11} & k_{12} \end{bmatrix}$. Note that in this case we have that $K\delta y = k_{11}\delta\theta + k_{12}\delta\dot{\theta}$. Remember that the input $u = u^* + \delta u$, and that for the pendulum $u = T$, i.e., the applied torque.

The exercise asks to find the values k_{11} and k_{12} for two arbitrary negative real eigenvalues λ_1 and λ_2 in (5.8).

Simulate your designed controller in the Python script. Check that your x^* are stable if you start close to them, and you can further check the robustness by applying small disturbances, e.g., add a small random number to x at every iteration.

Is it possible to design a stable controller with $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$? and for $C = \begin{bmatrix} 0 & 1 \end{bmatrix}$? Note that for this cases K will have different dimensions than for $C = I$ since C has different dimensions as well.