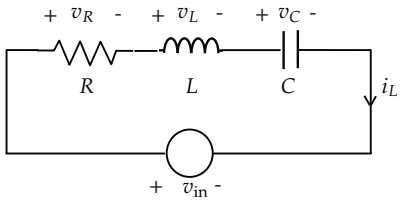
EE16B - Spring'20 - Lecture 6A Notes[1]

Murat Arcak

25 February 2020

State Space Representation of Dynamical Systems

State variables are internal variables that fully represent the state of a dynamical system at a given time. In previous lectures we used capacitor voltages and inductor currents as state variables of a circuit, and wrote differential equations that tell us how these variables evolve over time. The vector of such variables is called a *state vector* and the vector differential equation governing their evolution is called a *state model*.



Example 1: As a familiar example consider the RLC circuit depicted on the right where v_{in} denotes the input voltage.

Since the capacitor and inductor satisfy the relations

$$C\frac{dv_C(t)}{dt} = i_C(t) \tag{1}$$

$$L\frac{di_L(t)}{dt} = v_L(t), \tag{2}$$

we select v_C and i_L as the state variables. We then eliminate i_C from (1) by noting that $i_C = i_L$, and eliminate v_L from (2) using KVL ($v_L + v_C + v_R = v_{\text{in}}$), Ohm's Law ($v_R = Ri_R$), and $i_R = i_L$:

$$v_L = -v_C - v_R + v_{\text{in}} = -v_C - Ri_L + v_{\text{in}}. \tag{3}$$

Then the state model becomes

$$\begin{aligned} \frac{d}{dt}v_C(t) &= \frac{1}{C}i_L(t) \\ \frac{d}{dt}i_L(t) &= \frac{1}{L}\left(-v_C(t) - Ri_L(t) + v_{\text{in}}(t)\right). \end{aligned} \tag{4}$$

In a state model the left-hand side consists of derivatives of the state variables and the right-hand side depends only on the state variables and external inputs (v_{in} in this example). Other variables appearing in the equations, such as v_L in (2), must be eliminated by expressing them in terms of the state and input variables, as we did in (3).

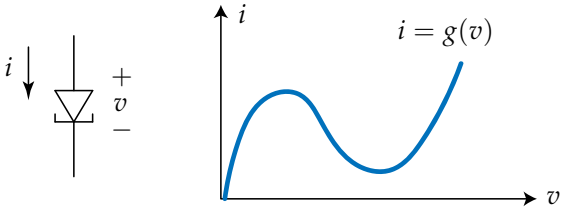
We say that a state model is *linear* if the right-hand side depends linearly on the state and input variables, as in (4) above. For a linear model the right-hand side can be written as a matrix multiplying the state vector, plus another matrix multiplying the input. Thus, for (4),

$$\frac{d}{dt}\begin{bmatrix} v_C(t) \\ i_L(t) \end{bmatrix} = \begin{bmatrix} 0 & 1/C \\ -1/L & -R/L \end{bmatrix}\begin{bmatrix} v_C(t) \\ i_L(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1/L \end{bmatrix} v_{\text{in}}(t). \tag{5}$$

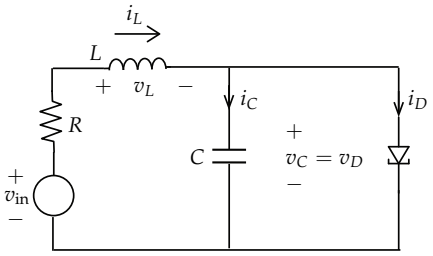
Most physical systems, however, are nonlinear. We have already seen nonlinear voltage-current curves for transistors[2]. The next example studies another nonlinear circuit element, the tunnel diode.

[2] However, since we focused on the low-voltage region where a linear approximation was adequate, we used linear differential equations.

Example 2: A tunnel diode is characterized by a voltage-current curve where, for a certain voltage range, the current decreases with increasing voltage. This is due to a quantum mechanical effect called tunneling.



Now consider the circuit on the right. We again use the state variables i_L and v_C, and start building a state model using the relations



$$C\frac{dv_C(t)}{dt} = i_C(t) \tag{6}$$

$$L\frac{di_L(t)}{dt} = v_L(t). \tag{7}$$

The next task is to rewrite the right-hand side in terms of state variables i_L and v_C, and input v_{in}. To do so note from KCL that $i_C = i_L - i_D$ and subtitute $i_D = g(v_D) = g(V_C)$, since $v_D = v_C$. Thus, (6) becomes

$$C\frac{dv_C(t)}{dt} = i_L(t) - g(v_C(t)), \tag{8}$$

where only the state variables i_L and v_C appear on the right-hand side. Likewise, using KVL, we substitute $v_L = -v_C - Ri_L + v_{in}$ in (7) and obtain

$$L\frac{di_L(t)}{dt} = -v_C(t) - Ri_L(t) + v_{in}(t). \tag{9}$$

Dividing both sides of (8) by C and both sides of (9) by L, we obtain the state model:

$$\begin{aligned}\frac{d}{dt}v_C(t) &= \frac{1}{C}i_L(t) - \frac{1}{C}g(v_C(t))\\ \frac{d}{dt}i_L(t) &= \frac{1}{L}\left(-v_C(t) - Ri_L(t) + v_{in}(t)\right).\end{aligned} \tag{10}$$

Since g is a nonlinear function, (10) is a nonlinear state model and can't be written in the matrix-vector form (5) we used in Example 1 to represent the linear model (4).

General form of State Equations

A general state model with n states and m inputs has the form

$$\begin{aligned} \frac{d}{dt}x_1(t) &= f_1(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)) \\ \frac{d}{dt}x_2(t) &= f_2(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)) \\ &\vdots \\ \frac{d}{dt}x_n(t) &= f_n(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)), \end{aligned} \tag{11}$$

where $f_1,\cdots,f_n$ are functions of the state and input variables.

In Examples 1 and 2 above we had $n=2$ states $x_1 = v_C$ and $x_2 = i_L$, and a single ($m=1$) input $u = v_{\text{in}}$. Thus (10) has the form above with

$$f_1(x_1,x_2) = \frac{1}{C}x_2 - \frac{1}{C}g(x_1), \quad f_2(x_1,x_2,u) = \frac{1}{L}\left(-x_1 - Rx_2 + u\right).$$

We will henceforth write (11) compactly as

$$\frac{d}{dt}\vec{x}(t) = f(\vec{x}(t),\vec{u}(t)) \tag{12}$$

where

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \vec{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}, \quad f(\vec{x},\vec{u}) = \begin{bmatrix} f_1(\vec{x},\vec{u}) \\ f_2(\vec{x},\vec{u}) \\ \vdots \\ f_n(\vec{x},\vec{u}) \end{bmatrix}.$$

The state model (11) is linear if for each $i = 1,\ldots,n$, the function f_i has the form

$$f_i(x_1,\cdots,x_n,u_1,\cdots,u_m) = a_{i1}x_1 + \cdots + a_{in}x_n + b_{i1}u_1 + \cdots + b_{im}u_m,$$

where $a_{i1},\cdots,a_{in},b_{i1},\cdots,b_{im}$ are coefficients. In this case we can write (12) in the matrix-vector form

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + B\vec{u}(t), \tag{13}$$

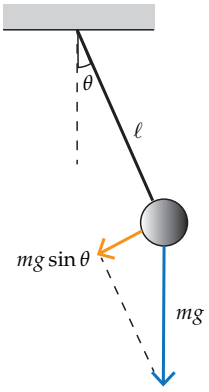
where A is a $n \times n$ matrix and B is a $n \times m$ matrix. The ith column of A consists of the coefficients $a_{i1},\cdots,a_{in}$ and ith column of B consists of $b_{i1},\cdots,b_{im}$. If there is only one input then B is $n \times 1$, that is a column vector, and we may write $\vec{b}$ instead of B:

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + \vec{b}u(t).$$

For example, (5) is of this form with

$$A = \begin{bmatrix} 0 & 1/C \\ -1/L & -R/L \end{bmatrix}, \quad \vec{b} = \begin{bmatrix} 0 \\ 1/L \end{bmatrix}.$$

In this module of the course we broaden our scope beyond circuits and analyze other dynamical systems, such as mechanical systems, again using state models. In circuit analysis we selected the state variables to be the inductor currents and capacitor voltages, as these variables are associated with the energy stored in these elements. Likewise, in modeling mechanical systems it is customary to select positions and velocities as the state variables, since the former is associated with potential energy and the latter with kinetic energy.



Example 3: The motion of the pendulum depicted on the right is governed by the differential equation

$$m\ell\frac{d^2\theta(t)}{dt^2} = -k\ell\frac{d\theta(t)}{dt} - mg\sin\theta(t) \tag{14}$$

where the left hand side is mass×accelaration in the tangential direction, and the right hand side is total force acting in that direction, including friction and the tangential component of the gravitational force.

To bring this second order differential equation to state space form we define the state variables to be the angle and angular velocity:

$$x_1(t) := \theta(t) \qquad x_2(t) := \frac{d\theta(t)}{dt},$$

and note that they satisfy

$$\begin{aligned}\frac{d}{dt}x_1(t) &= x_2(t)\\ \frac{d}{dt}x_2(t) &= -\frac{k}{m}x_2(t) - \frac{g}{\ell}\sin x_1(t).\end{aligned} \tag{15}$$

The first equation here follows from the definition of $x_2(t)$ as the angular velocity, and the second equation follows from (14).

Here we did not consider external forces that could act as inputs, so the equations (15) have the form (12) with the input omitted:

$$f(\vec{x}) = \begin{bmatrix} x_2 \\ -\frac{k}{m}x_2 - \frac{g}{\ell}\sin x_1 \end{bmatrix}. \tag{16}$$

Equilibrium States

For a system without inputs, $\frac{d}{dt}\vec{x}(t) = f(\vec{x}(t))$, the solutions of the static equation

$$f(\vec{x}) = 0$$

are called *equilibrium points*. If we pick an equilibrium point $\vec{x}^*$ as the initial state at t_0, then

$$\frac{d}{dt}\vec{x}(t) = f(\vec{x}^*) = 0$$

for all $t \geq t_0$, therefore the state remains at $\vec{x}^*$ in the future:

$$\vec{x}(t) = \vec{x}^* \quad t \geq t_0.$$

Example 3 revisited: We can find the equilibrium points for the pendulum example above by solving the equation

$$f(\vec{x}) = \begin{bmatrix} x_2 \\ -\frac{k}{m}x_2 - \frac{g}{\ell}\sin x_1 \end{bmatrix} = 0.$$

This consists of two equations,

$$x_2 = 0, \quad -\frac{k}{m}x_2 - \frac{g}{\ell}\sin x_1 = 0,$$

which have two distinct solutions:

$$x_1 = 0, \quad x_2 = 0,$$

that is the downward pointing position of the pendulum, and

$$x_1 = \pi, \quad x_2 = 0,$$

which is the upright position[3]. As this example illustrates, a system may have more than one equilibrium. We will see later that the upright position is *unstable*, meaning that the pendulum would diverge from this equilibrium when slightly perturbed. In contrast the downward position is *stable*, because the pendulum would return to this position after some oscillations with the help of the friction term.

[3] Other solutions, such as $(x_1, x_2) = (2\pi, 0)$, or $(x_1, x_2) = (3\pi, 0)$ are identical to one of the two equilibria already described.

EE16B - Spring'20 - Lecture 6B Notes[1]

Murat Arcak

27 February 2020

State Space Models Continued

In the last lecture we considered a general state model of the form

$$\begin{aligned} \frac{d}{dt}x_1(t) &= f_1(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)) \\ \frac{d}{dt}x_2(t) &= f_2(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)) \\ &\vdots \\ \frac{d}{dt}x_n(t) &= f_n(x_1(t),\cdots,x_n(t),u_1(t),\cdots,u_m(t)), \end{aligned} \tag{1}$$

and rewrote it compactly as

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

When the system (2) is linear we write it in the matrix-vector form

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + B\vec{u}(t), \tag{3}$$

where A is a $n \times n$ matrix and B is a $n \times m$ matrix.

Equilibrium States

Recall that, for a system without inputs, $\frac{d}{dt}\vec{x}(t) = f(\vec{x}(t))$, the solutions of the static equation

$$f(\vec{x}) = 0$$

are called *equilibrium points*. We can extend the definition of an equilibrium to systems with inputs, assuming that a constant input $\vec{u}$ is applied instead of a time-varying one. In this case $\vec{x}$ is an equilibrium point if it satisfies

$$f(\vec{x},\vec{u}) = 0, \tag{4}$$

where the solution depends on the constant input $\vec{u}$ applied.

For linear systems (3) we find equilibrium points by solving for $\vec{x}$ in

$$A\vec{x} + B\vec{u} = 0, \tag{5}$$

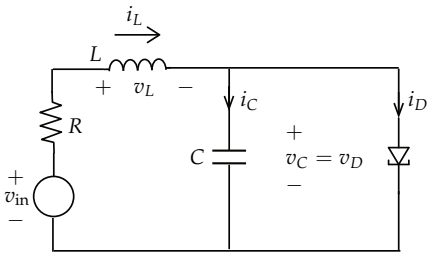
with $\vec{u}$ as a given constant. If A is invertible each constant input $\vec{u}$ produces a unique equilibrium.

When A is singular, there may be a continuum of infinitely many equilibrium points (*e.g.*, with $\vec{u} = 0$, every point in the null space of A is an equilibrium point) or there may be no equilibrium points, which happens when $B\vec{u}$ is not in the range space of A.

Note that multiple isolated equilibrium points – such as those in the pendulum example – can't occur in linear systems, since (5) has either a single solution, no solution, or a continuum of infinitely many solutions. Therefore, multiple isolated equilibria can arise only in nonlinear systems.

Example 1: In the last lecture we discussed the circuit on the right and obtained the state model:

$$\begin{aligned} \frac{d}{dt}v_C(t) &= \frac{1}{C}i_L(t) - \frac{1}{C}g(v_C(t)) \\ \frac{d}{dt}i_L(t) &= \frac{1}{L}\left(-v_C(t) - Ri_L(t) + v_{\text{in}}(t)\right), \end{aligned} \tag{6}$$



where g is a nonlinear function representing the tunnel diode's voltage-current characteristics (see figure below).

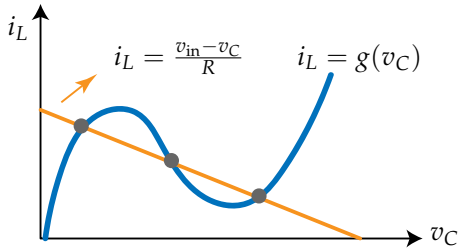
To find the equilibrium points we set the left-hand side of (6) to zero and solve for v_C and i_L:

$$\begin{aligned} \frac{1}{C}i_L - \frac{1}{C}g(v_C) &= 0 \\ \frac{1}{L}\left(-v_C - Ri_L + v_{\text{in}}\right) &= 0. \end{aligned}$$

It follows from these two equations that we can find the equilibrium points by superimposing the curves

$$i_L = g(v_C) \quad \text{and} \quad i_L = \frac{v_{\text{in}} - v_C}{R}, \tag{7}$$

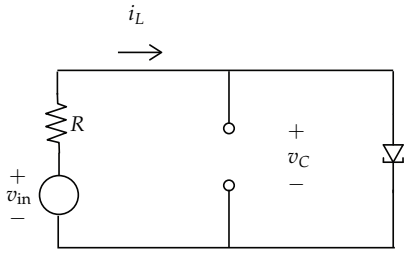
and finding their intersections. The figure below shows the case where there are three equilibrium points.



Depending on the values of the constants v_{in} and R, it is also possible to have only one or only two equilibrium points: imagine raising the orange line (*i.e.*, increasing v_{in}) until the two intersections on the left collapse into one, and then disappear.

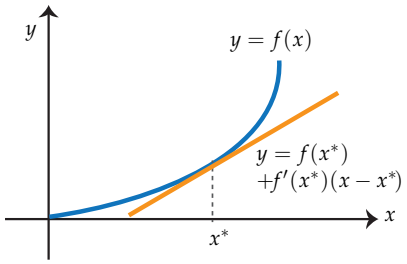
To gain further insight into equilibrium states of circuits, recall that we use capacitor voltages and inductor currents as state variables. Thus, to find equilibrium points, we must solve the circuit equations with the time derivatives of capacitor voltages and inductor currents set to zero. Since $C\frac{d}{dt}v_C(t) = i_C(t)$ and $L\frac{d}{dt}i_L(t) = v_L(t)$, this means setting the capacitor currents and inductor voltages to zero. Thus, at equilibrium the capacitor acts like an open circuit and the inductor like a short circuit.

The figure on the right shows the tunnel diode circuit, with the inductor treated as short circuit and capacitor as open circuit. As an exercise show that i_L and v_C in this circuit indeed satisfy the equilibrium equations (7).



Linearization

Linear models are advantageous because their solutions can be found analytically. The methods applicable to nonlinear models are limited; therefore it is common practice to approximate a nonlinear model with a linear one that is valid around an equilibrium state.



Recall that the Taylor approximation of a differentiable function f around a point x^* is:

$$f(x) \approx f(x^*) + \nabla f(x)|_{x=x^*}\,(x - x^*),$$

as illustrated on the right for a scalar-valued function of a single variable. When $f(\vec{x})$ is a vector of n functions $f_1, \cdots, f_n$ as in our state models, $\nabla f(\vec{x})$ is interpreted as the $n \times n$ matrix of partial derivatives:

$$\nabla f(x_1, \ldots, x_n) = \begin{bmatrix} \frac{\partial f_1(x_1,\ldots,x_n)}{\partial x_1} & \frac{\partial f_1(x_1,\ldots,x_n)}{\partial x_2} & \cdots & \frac{\partial f_1(x_1,\ldots,x_n)}{\partial x_n} \\ \frac{\partial f_2(x_1,\ldots,x_n)}{\partial x_1} & \frac{\partial f_2(x_1,\ldots,x_n)}{\partial x_2} & \cdots & \frac{\partial f_2(x_1,\ldots,x_n)}{\partial x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_n(x_1,\ldots,x_n)}{\partial x_1} & \frac{\partial f_n(x_1,\ldots,x_n)}{\partial x_2} & \cdots & \frac{\partial f_n(x_1,\ldots,x_n)}{\partial x_n} \end{bmatrix}.$$

We linearize nonlinear state models by applying this approximation around an equilibrium state. Let $\vec{x}^*$ be an equilbrium for the system

$$\frac{d}{dt}\vec{x}(t) = f(\vec{x}(t)), \tag{8}$$

that is $f(\vec{x}^*) = 0$, and define the deviation of $\vec{x}(t)$ from $\vec{x}^*$ as:

$$\boxed{\tilde{x}(t) := \vec{x}(t) - \vec{x}^*.} \tag{9}$$

Then we see that

$$\begin{aligned}\frac{d}{dt}\tilde{x}(t) &= \frac{d}{dt}\vec{x}(t) - \frac{d}{dt}\vec{x}^* \\ &= \frac{d}{dt}\vec{x}(t) = f(\vec{x}(t)) = f(\vec{x}^* + \tilde{x}(t)) \\ &\approx f(\vec{x}^*) + \nabla f(\vec{x})|_{\vec{x}=\vec{x}^*}\,\tilde{x}(t), \end{aligned} \tag{10}$$

where the second equality follows because $\vec{x}^*$ is constant and, thus, its derivative is zero. Substituting $f(\vec{x}^*) = 0$ in (10) and defining

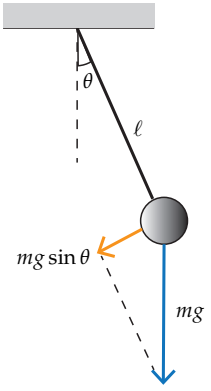
$$\boxed{A \triangleq \nabla f(\vec{x})|_{\vec{x}=\vec{x}^*}} \tag{11}$$

we obtain the linearization of (8) around the equilibrium $\vec{x}^*$ as:

$$\boxed{\frac{d}{dt}\tilde{x}(t) \approx A\tilde{x}(t).}$$

Example 2: Recall the pendulum model from the previous lecture:

$$\begin{aligned}\frac{dx_1(t)}{dt} &= x_2(t) \\ \frac{dx_2(t)}{dt} &= -\frac{k}{m}x_2(t) - \frac{g}{\ell}\sin x_1(t)\end{aligned} \tag{12}$$



where

$$x_1(t) := \theta(t) \quad \text{and} \quad x_2(t) := \frac{d\theta(t)}{dt}.$$

The two distinct equilibrium points are the downward position:

$$x_1 = 0, \quad x_2 = 0, \tag{13}$$

and the upright position:

$$x_1 = \pi, \quad x_2 = 0. \tag{14}$$

Since the entries of $f(\vec{x})$ are $f_1(\vec{x}) = x_2$ and $f_2(\vec{x}) = -\frac{k}{m}x_2 - \frac{g}{\ell}\sin x_1$, we have

$$\nabla f(\vec{x}) = \begin{bmatrix} \frac{\partial f_1(x_1,x_2)}{\partial x_1} & \frac{\partial f_1(x_1,x_2)}{\partial x_2} \\ \frac{\partial f_2(x_1,x_2)}{\partial x_1} & \frac{\partial f_2(x_1,x_2)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{\ell}\cos x_1 & \frac{-k}{m} \end{bmatrix}.$$

By evaluating this matrix at (13) and (14), we obtain the linearization around the respective equilibrium point:

$$A_{\text{down}} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{\ell} & \frac{-k}{m} \end{bmatrix} \qquad A_{\text{up}} = \begin{bmatrix} 0 & 1 \\ \frac{g}{\ell} & \frac{-k}{m} \end{bmatrix}. \tag{15}$$

As an exercise show that A_{up} has an eigenvalue with positive real part. We will see later that the presence of an eigenvalue with positive real part implies instability of the respective equilibrium state. In contrast A_{down} has eigenvalues with negative real parts, indicating stability of the downward position.

EE16B - Spring'20 - Lecture 7A Notes[1]

Murat Arcak

3 March 2020

Linearization and Discrete-Time Systems

Linearization with Inputs

In the last lecture we considered nonlinear systems with no inputs and linearized them by applying a Taylor approximation around an equilibrium. We can also apply linearization to systems with inputs,

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

around an equilibrium $\vec{x}^*$ maintained by a constant input $\vec{u}^*$ that satisfies $f(\vec{x}^*, \vec{u}^*) = 0$.

Define the perturbation variables $\tilde{x}(t)$ and $\tilde{u}(t)$ as:

$$\boxed{\tilde{x}(t) := \vec{x}(t) - \vec{x}^*, \quad \tilde{u}(t) := \vec{u}(t) - \vec{u}^*.} \tag{1}$$

Then,

$$\begin{aligned}\frac{d}{dt}\tilde{x}(t) &= \frac{d}{dt}\vec{x}(t) - \frac{d}{dt}\vec{x}^* \\ &= \frac{d}{dt}\vec{x}(t) = f(\vec{x}(t), \vec{u}(t)) = f(\vec{x}^* + \tilde{x}(t), \vec{u}^* + \tilde{u}(t)) \\ &\approx f(\vec{x}^*, \vec{u}^*) + \nabla_x f(\vec{x}, \vec{u})|_{\vec{x}^*, \vec{u}^*}\, \tilde{x}(t) + \nabla_u f(\vec{x}, \vec{u})|_{\vec{x}^*, \vec{u}^*}\, \tilde{u}(t) \end{aligned} \tag{2}$$

where

$$\nabla_x f(\vec{x}, \vec{u}) := \begin{bmatrix} \frac{\partial f_1(\vec{x},\vec{u})}{\partial x_1} & \frac{\partial f_1(\vec{x},\vec{u})}{\partial x_2} & \cdots & \frac{\partial f_1((\vec{x},\vec{u})}{\partial x_n} \\ \frac{\partial f_2(\vec{x},\vec{u})}{\partial x_1} & \frac{\partial f_2(\vec{x},\vec{u})}{\partial x_2} & \cdots & \frac{\partial f_2((\vec{x},\vec{u})}{\partial x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_n(\vec{x},\vec{u})}{\partial x_1} & \frac{\partial f_n(\vec{x},\vec{u})}{\partial x_2} & \cdots & \frac{\partial f_n(\vec{x},\vec{u})}{\partial x_n} \end{bmatrix}$$

$$\nabla_u f(\vec{x}, \vec{u}) := \begin{bmatrix} \frac{\partial f_1(\vec{x},\vec{u})}{\partial u_1} & \frac{\partial f_1(\vec{x},\vec{u})}{\partial u_2} & \cdots & \frac{\partial f_1((\vec{x},\vec{u})}{\partial u_m} \\ \frac{\partial f_2(\vec{x},\vec{u})}{\partial u_1} & \frac{\partial f_2(\vec{x},\vec{u})}{\partial u_2} & \cdots & \frac{\partial f_2((\vec{x},\vec{u})}{\partial u_m} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_n(\vec{x},\vec{u})}{\partial u_1} & \frac{\partial f_n(\vec{x},\vec{u})}{\partial u_2} & \cdots & \frac{\partial f_n(\vec{x},\vec{u})}{\partial u_m} \end{bmatrix}.$$

Substituting $f(\vec{x}^*, \vec{u}^*) = 0$ in (2) and defining

$$\boxed{A := \nabla_x f(\vec{x}, \vec{u})|_{\vec{x}^*, \vec{u}^*} \quad B := \nabla_u f(\vec{x}, \vec{u})|_{\vec{x}^*, \vec{u}^*}} \tag{3}$$

we obtain the linearization:

$$\boxed{\frac{d}{dt}\tilde{x}(t) \approx A\tilde{x}(t) + B\tilde{u}(t).}$$

Example 1: The velocity $v(t)$ of a vehicle is governed by

$$M\frac{d}{dt}v(t) = -\frac{1}{2}\rho ac\, v(t)^2 + \frac{1}{R}u(t) \tag{4}$$

where $u(t)$ is the wheel torque, M is vehicle mass, ρ is air density, a is vehicle area, c is drag coefficient, and R is wheel radius. Note that we can maintain the velocity at a desired value v^* if we apply the torque

$$u^* = \frac{R}{2}\rho ac\, v^{*2},$$

which counterbalances the drag force at that velocity. We rewrite the model (4) as $\frac{d}{dt}v(t) = f(v(t), u(t))$, where

$$f(v,u) = -\frac{1}{2M}\rho ac\, v^2 + \frac{1}{RM}u.$$

Then the linearized dynamics for the perturbation $\tilde{v}(t) = v(t) - v^*$ is

$$\frac{d}{dt}\tilde{v}(t) = \lambda\tilde{v}(t) + b\tilde{u}(t), \tag{5}$$

where $\tilde{u}(t) = u(t) - u^*$,

$$\lambda = \left.\frac{\partial f(v,u)}{\partial v}\right|_{v^*,u^*} = -\frac{1}{M}\rho acv^*, \quad b = \left.\frac{\partial f(v,u)}{\partial u}\right|_{v^*,u^*} = \frac{1}{RM}.$$

Here we used the letters λ and b instead of A and B to emphasize that they are scalars. Note that if we apply $u(t) = u^*$, that is $\tilde{u}(t) = 0$, then the solution of the scalar differential equation (5) is

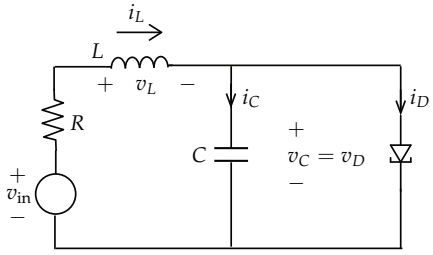
$$\tilde{v}(t) = \tilde{v}(0)e^{\lambda t},$$

which converges to 0 since $\lambda < 0$. This means that if $v(t)$ is perturbed from v^*, it will return[2] to v^*. Equilibrium points with this property are called *stable*, a concept we will study in detail later.

[2] The rate of convergence depends on λ. For a typical sedan at $v^* = 29$ m/s (≈ 65 mph) we would get $\lambda \approx -0.01$ sec^{-1} with parameters $M = 1700$ kg, $a = 2.6$ m^2, $\rho = 1.2$ kg/m^3, $c = 0.2$.

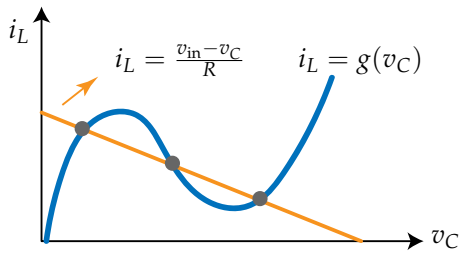
Example 2: In previous lectures we discussed the tunnel diode circuit on the right and obtained the state model:

$$\begin{aligned}\frac{d}{dt}v_C(t) &= \frac{1}{C}i_L(t) - \frac{1}{C}g(v_C(t))\\ \frac{d}{dt}i_L(t) &= \frac{1}{L}\left(-v_C(t) - Ri_L(t) + v_{\text{in}}(t)\right),\end{aligned} \tag{6}$$



where g is a nonlinear function representing the tunnel diode's voltage-current characteristics (see figure below). We also showed that the equilibrium points are the intersections of the curves

$$i_L = g(v_C) \quad \text{and} \quad i_L = \frac{v_{\text{in}} - v_C}{R}. \tag{7}$$



Let v_{in}^* be a constant input voltage and let (v_C^*, i_L^*) denote one of the resulting equilibrium states, that is one of the intersections of the two curves above. Since the right-hand side of (6) has the form

$$f(v_C, i_L, v_{in}) = \begin{bmatrix} f_1(v_C, i_L, v_{in}) \\ f_2(v_C, i_L, v_{in}) \end{bmatrix} = \begin{bmatrix} \frac{1}{C} i_L - \frac{1}{C} g(v_C) \\ \frac{1}{L}\left(-v_C - R i_L + v_{in}\right) \end{bmatrix},$$

the matrices A and B in (3) are:

$$A = \left.\begin{bmatrix} \frac{\partial f_1(v_C, i_L, v_{in})}{\partial v_C} & \frac{\partial f_1(v_C, i_L, v_{in})}{\partial i_L} \\ \frac{\partial f_2(v_C, i_L, v_{in})}{\partial v_C} & \frac{\partial f_2(v_C, i_L, v_{in})}{\partial i_L} \end{bmatrix}\right|_{(v_C^*, i_L^*)} = \begin{bmatrix} \frac{-1}{C} g'(v_C^*) & \frac{1}{C} \\ \frac{-1}{L} & \frac{-R}{L} \end{bmatrix}$$

$$B = \left.\begin{bmatrix} \frac{\partial f_1(v_C, i_L, v_{in})}{\partial v_{in}} \\ \frac{\partial f_2(v_C, i_L, v_{in})}{\partial v_{in}} \end{bmatrix}\right|_{(v_C^*, i_L^*)} = \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix}.$$

Discrete-Time Systems

In a *discrete*-time system, the state vector $\vec{x}(t)$ evolves according to a *difference* equation rather than a differential equation:

$$\vec{x}(t+1) = f(\vec{x}(t), \vec{u}(t)) \quad t = 0, 1, 2, \ldots \tag{8}$$

Here $f(\vec{x}, \vec{u})$ is a function that gives the state vector at the next time instant based on the present values of the states and inputs.

As in the continuous-time case, when $f(\vec{x}, \vec{u}) \in \mathbb{R}^n$ is linear in $\vec{x} \in \mathbb{R}^n$ and $\vec{u} \in \mathbb{R}^m$, we can rewrite it in the form

$$f(\vec{x}, \vec{u}) = A\vec{x} + B\vec{u}$$

where A is $n \times n$ and B is $n \times m$. The state model is then

$$\vec{x}(t+1) = A\vec{x}(t) + B\vec{u}(t). \tag{9}$$

Example 3: Let $s(t)$ denote the inventory of a manufacturer at the start of the t-th business day. The inventory at the start of the next day, $s(t+1)$, is the sum of $s(t)$ and the goods $g(t)$ manufactured, minus the goods $u_1(t)$ sold on day t. Assuming it takes a day to do the manufacturing, the amount of goods $g(t)$ manufactured is equal

to the raw material available the previous day, $r(t-1)$. The raw material $r(t)$ is equal to the order placed the previous day, $u_2(t-1)$, assuming it takes a day for the order to arrive.

The state variables $s(t)$, $g(t)$, $r(t)$, thus evolve according to the model

$$\begin{aligned} s(t+1) &= s(t) + g(t) - u_1(t) \\ g(t+1) &= r(t) \\ r(t+1) &= u_2(t), \end{aligned} \tag{10}$$

where u_1 and u_2 are two distinct inputs, one representing the customer demand and the other the manufacturer's raw material order.

Note that this system is linear, and we can write (10) as:

$$\underbrace{\begin{bmatrix} s(t+1) \\ g(t+1) \\ r(t+1) \end{bmatrix}}_{\vec{x}(t+1)} = \underbrace{\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} s(t) \\ g(t) \\ r(t) \end{bmatrix}}_{\vec{x}(t)} + \underbrace{\begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix}}_{\vec{u}(t)}.$$

Example 4: Let $p(t)$ be the number of EECS professors in a country in year t, and let $r(t)$ be the number of industry researchers with a PhD degree. A fraction, γ, of the PhDs become professors themselves and the rest become industry researchers. A fraction, δ, in each profession leaves the field every year due to retirement or other reasons.

Each professor graduates, on average, $u(t)$ PhD students per year. We treat this number as a control input because it can be manipulated by the government using research funding. This means there will be $p(t)u(t)$ new PhDs in year t, and $\gamma p(t)u(t)$ new professors. The state model is then

$$\begin{aligned} p(t+1) &= (1-\delta)p(t) + \gamma p(t)u(t) \\ r(t+1) &= (1-\delta)r(t) + (1-\gamma)p(t)u(t). \end{aligned} \tag{11}$$

Note that this system is nonlinear due to the product of the state variable p with the input u. □

When the input $\vec{u}(t)$ in (8) is a constant vector $\vec{u}^*$, the equilibrium points are obtained by solving for $\vec{x}$ in the equation[3]:

$$\vec{x} = f(\vec{x}, \vec{u}^*). \tag{12}$$

[3] Note that the equilibrium condition (12) in discrete time differs from the continuous time condition $0 = f(\vec{x}, \vec{u}^*)$.

If $\vec{x}^*$ satisfies this equation and we start with the initial condition $\vec{x}^*$, the next state is $f(\vec{x}^*, \vec{u}^*)$, which is again $\vec{x}^*$. The same argument applies to subsequent time instants, so $\vec{x}(t)$ remains at $\vec{x}^*$.

For the linear system (9) the equilibrium condition (12) becomes:

$$\vec{x} = A\vec{x} + B\vec{u}^*, \quad \text{or, equivalently} \quad (I - A)\vec{x} = B\vec{u}^*.$$

EE16B - Spring'20 - Lecture 7B Notes[1]

Murat Arcak

5 March 2020

Discrete-Time Systems and Discretization

Recall that in a *discrete*-time system, the state vector $\vec{x}(t)$ evolves according to a *difference* equation rather than a differential equation:

$$\vec{x}(t+1) = f(\vec{x}(t), \vec{u}(t)) \quad t = 0, 1, 2, \ldots \tag{1}$$

Here $f(\vec{x}, \vec{u})$ is a function that gives the state vector at the next time instant based on the present values of the states and inputs.

As in the continuous-time case, when $f(\vec{x}, \vec{u}) \in \mathbb{R}^n$ is linear in $\vec{x} \in \mathbb{R}^n$ and $\vec{u} \in \mathbb{R}^m$, we can rewrite it in the form

$$f(\vec{x}, \vec{u}) = A\vec{x} + B\vec{u}$$

where A is $n \times n$ and B is $n \times m$. The state model is then

$$\vec{x}(t+1) = A\vec{x}(t) + B\vec{u}(t). \tag{2}$$

When the input $\vec{u}(t)$ in (1) is a constant vector $\vec{u}^*$, the equilibrium points are obtained by solving for $\vec{x}$ in the equation[2]:

$$\vec{x} = f(\vec{x}, \vec{u}^*). \tag{3}$$

[2] Note that the equilibrium condition (3) in discrete time differs from the continuous time condition $0 = f(\vec{x}, \vec{u}^*)$.

If $\vec{x}^*$ satisfies this equation and we start with the initial condition $\vec{x}^*$, the next state is $f(\vec{x}^*, \vec{u}^*)$, which is again $\vec{x}^*$. The same argument applies to subsequent time instants, so $\vec{x}(t)$ remains at $\vec{x}^*$.

For the linear system (2) the equilibrium condition (3) becomes:

$$\vec{x} = A\vec{x} + B\vec{u}^*, \quad \text{or, equivalently} \quad (I - A)\vec{x} = B\vec{u}^*.$$

Linearization for nonlinear discrete-time systems is performed similarly to continuous-time. The perturbation variables $\tilde{x}(t) := \vec{x}(t) - \vec{x}^*$ and $\tilde{u}(t) := \vec{u}(t) - \vec{u}^*$ satisfy:

$$\begin{aligned} \tilde{x}(t+1) = &\ \vec{x}(t+1) - \vec{x}^* = f(\vec{x}(t), \vec{u}(t)) - \vec{x}^* \\ \approx &\ f(\vec{x}^*, \vec{u}^*) + \nabla_x f(\vec{x}, \vec{u})|_{\vec{x}^*, \vec{u}^*}\, \tilde{x}(t) + \nabla_u f(\vec{x}, \vec{u})|_{\vec{x}^*, \vec{u}^*}\, \tilde{u}(t) - \vec{x}^*. \end{aligned}$$

Substituting $f(\vec{x}^*, \vec{u}^*) - \vec{x}^* = 0$, which follows because $\vec{x}^*$ is an equilibrium, we get

$$\boxed{\tilde{x}(t+1) \approx A\tilde{x}(t) + B\tilde{u}(t)}$$

where $A = \nabla_x f(\vec{x}, \vec{u})|_{\vec{x}^*, \vec{u}^*}$ and $B = \nabla_u f(\vec{x}, \vec{u})|_{\vec{x}^*, \vec{u}^*}$.

Changing State Variables

Given the state vector $\vec{x} \in \mathbb{R}^n$ any transformation of the form

$$\vec{z} := T\vec{x}, \tag{4}$$

where T is a $n \times n$ invertible matrix, defines new variables z_i, $i = 1, \ldots, n$, as a linear combination of the original variables $x_1, \ldots, x_n$.

To see how this change of variables affects the state equation

$$\vec{x}(t+1) = A\vec{x}(t) + B\vec{u}(t),$$

note that

$$\vec{z}(t+1) = T\vec{x}(t+1) = TA\vec{x}(t) + TB\vec{u}(t)$$

and substitute $\vec{x} = T^{-1}\vec{z}$ in the right hand side to obtain:

$$\vec{z}(t+1) = TAT^{-1}\vec{z}(t) + TB\vec{u}(t).$$

Thus the original A and B matrices are replaced with:

$$\boxed{A_{\text{new}} = TAT^{-1}, \quad B_{\text{new}} = TB.} \tag{5}$$

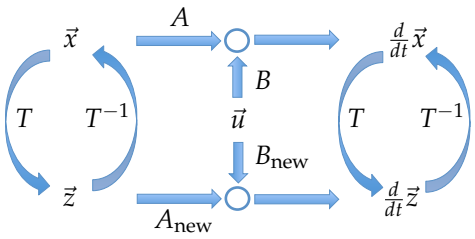
The same change of variables brings the *continuous-time* system

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + B\vec{u}(t)$$

to the form

$$\frac{d}{dt}\vec{z}(t) = A_{\text{new}}\vec{z}(t) + B_{\text{new}}\vec{u}(t)$$

as depicted below.



We use particular choices of T to obtain special forms of A_{new} and B_{new} that make the analysis easier. For example, we saw in Lecture 3A that we can make A_{new} diagonal if the $n \times n$ matrix A has n independent eigenvectors $\vec{v}_1, \cdots, \vec{v}_n$. This is because the matrix $V = [\vec{v}_1 \cdots \vec{v}_n]$ satisfies

$$AV = [A\vec{v}_1 \cdots A\vec{v}_n] = [\lambda_1\vec{v}_1 \cdots \lambda_n\vec{v}_n] = \underbrace{[\vec{v}_1 \cdots \vec{v}_n]}_{=V} \underbrace{\begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}}_{=:\Lambda},$$

therefore $V^{-1}AV = \Lambda$. This means that the choice

$$T = V^{-1}$$

gives $A_{\text{new}} = TAT^{-1} = \Lambda$, which is diagonal.

Digital Control

In upcoming lectures we will be designing the input signal $\vec{u}$ of a continuous-time system

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + B\vec{u}(t) \tag{6}$$

to ensure that the solution $\vec{x}(t)$ meets requirements, such as reaching a target state in a given amount of time.

The input signal is typically generated digitally in a computer, by using measurements of $\vec{x}(t)$ sampled every T units of time. Thus the computer receives a discrete sequence

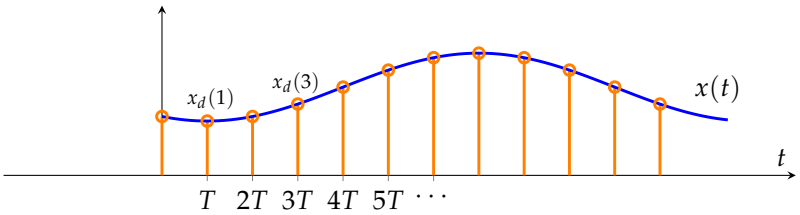
$$\vec{x}(0), \vec{x}(T), \vec{x}(2T), \cdots$$

as shown in the figure below. We use the notation

$$\vec{x}_d(k) := \vec{x}(kT) \tag{7}$$

where the subscript 'd' stands for 'discrete', so that we can represent the samples $\vec{x}(0), \vec{x}(T), \vec{x}(2T), \cdots$ as a discrete-time signal

$$\vec{x}_d(0), \vec{x}_d(1), \vec{x}_d(2), \cdots$$



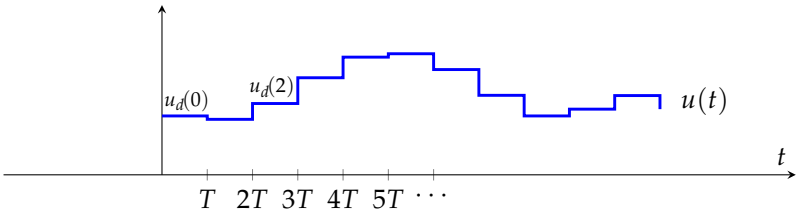
Using this sequence an appropriate control algorithm generates inputs to the system, again as a discrete sequence

$$\vec{u}_d(0), \vec{u}_d(1), \vec{u}_d(2), \ldots$$

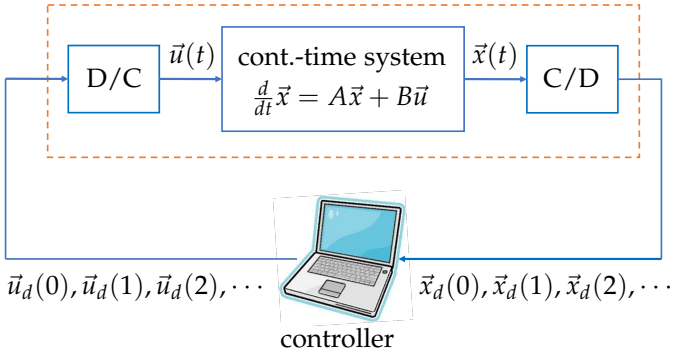
However, since the system (6) admits only continuous-time inputs, this sequence must be converted to continuous-time. This is typically done with a *zero-order hold* device that keeps $\vec{u}(t)$ constant at $\vec{u}_d(0)$ in the interval $t \in [0, T)$, at $\vec{u}_d(1)$ for $t \in [T, 2T)$, and so on. Therefore,

$$\vec{u}(t) = \vec{u}_d(k) \quad t \in [kT, (k+1)T), \tag{8}$$

which has a staircase shape as shown below.



The overall control scheme is illustrated below where the D/C (discrete-to-control) block represents zero-order hold and the C/D (continuous-to-discrete) block represents sampling.



Discretization

From the viewpoint of the controller, the system combined with D/C and C/D blocks (dashed box in the figure above) receives a discrete input sequence $\vec{u}_d(k)$ and generates a discrete state sequence $\vec{x}_d(k)$ that consists of snapshots of $\vec{x}(t)$.

We now wish to derive a discrete-time model

$$\vec{x}_d(k+1) = A_d\vec{x}_d(k) + B_d\vec{u}_d(k) \tag{9}$$

that describes how the state evolves from one snapshot to the next. That is, we want (9) to return the next sample of the continuous-time system (6) when the input $\vec{u}(t)$ is constant in between the samples.

To see how such a discrete-time model can be derived, first assume the continuous-time system has a single state and single input:

$$\frac{d}{dt}x(t) = \lambda x(t) + bu(t). \tag{10}$$

Since the value of $x(t)$ at $t = kT$ is $x_d(k)$, the solution of the scalar differential equation above with initial time kT is

$$x(t) = e^{\lambda(t-kT)}x_d(k) + \int_{kT}^{t} e^{\lambda(t-\tau)}bu(\tau)d\tau.$$

We also know that the input $u(t)$ from $t = kT$ to $t = kT + T$ is the constant $u_d(k)$. Thus, the solution at time $t = kT + T$ is

$$x(kT+T) = e^{\lambda T} x_d(k) + \int_{kT}^{kT+T} e^{\lambda(kT+T-\tau)} b u_d(k) d\tau.$$

Substituting $x(kT+T) = x_d(k+1)$ and factoring $bu_d(k)$ out of the integral (since it is constant) we get

$$x_d(k+1) = e^{\lambda T} x_d(k) + \left(\int_{kT}^{kT+T} e^{\lambda(kT+T-\tau)} d\tau \right) b u_d(k). \tag{11}$$

We next simplify the integral in brackets by defining the variable $s := kT + T - \tau$:

$$\int_{kT}^{kT+T} e^{\lambda(kT+T-\tau)} d\tau = \int_{T}^{0} e^{\lambda s}(-ds) = \int_{0}^{T} e^{\lambda s} ds.$$

Substituting in (11) we conclude

$$x_d(k+1) = \lambda_d x_d(k) + b_d u_d(k) \tag{12}$$

where

$$\lambda_d = e^{\lambda T}, \quad b_d = b \int_0^T e^{\lambda s} ds = \begin{cases} bT & \text{if } \lambda = 0 \\ b\frac{e^{\lambda T}-1}{\lambda} & \text{if } \lambda \neq 0. \end{cases}$$

Thus, (12) evaluates the state of the continuous-time model (10) at the next sample time. We refer to (12) as the 'discretization' of (10).

EE16B - Spring'20 - Lecture 8A Notes[1]

Murat Arcak

10 March 2020

Discretization and Controllability

Discretization for Vector State Models

In the last lecture we considered the linear continuous-time system

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + B\vec{u}(t), \tag{1}$$

where $\vec{x}(t)$ is sampled every T units of time, leading to the sequence

$$\vec{x}_d(k) := \vec{x}(kT), \quad k = 0, 1, 2, \ldots \tag{2}$$

If $\vec{u}(t)$ is constant between the samples:

$$\vec{u}(t) = \vec{u}_d(k) \quad t \in [kT, (k+1)T), \tag{3}$$

then we can derive a discrete-time model

$$\vec{x}_d(k+1) = A_d\vec{x}_d(k) + B_d\vec{u}_d(k) \tag{4}$$

that describes how the state of the continuous-time system evolves from one sample to the next.

Last time we did this derivation for the scalar system

$$\frac{d}{dt}x(t) = \lambda x(t) + bu(t), \tag{5}$$

and obtained

$$x_d(k+1) = \lambda_d x_d(k) + b_d u_d(k) \tag{6}$$

where

$$\boxed{\lambda_d = e^{\lambda T}, \quad b_d = b\int_0^T e^{\lambda s}ds = \begin{cases} bT & \text{if } \lambda = 0 \\ b\frac{e^{\lambda T}-1}{\lambda} & \text{if } \lambda \neq 0. \end{cases}} \tag{7}$$

To generalize this result to the vector state model (1) let's first assume A is diagonal and B is a column vector:

$$A = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}, \quad B = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}.$$

Then (1) consists of decoupled scalar equations

$$\frac{d}{dt}x_i(t) = \lambda_i x_i(t) + b_i u(t)$$

and we can discretize each as in (6)-(7). We then assemble the discretized scalar equations into the vector form (4) with

$$A_d = \begin{bmatrix} e^{\lambda_1 T} & & \\ & \ddots & \\ & & e^{\lambda_n T} \end{bmatrix}, \quad B_d = \begin{bmatrix} \int_0^T e^{\lambda_1 s} ds & & \\ & \ddots & \\ & & \int_0^T e^{\lambda_n s} ds \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}.$$

Next suppose A is not diagonal, but *diagonalizable*[2]; that is, it has linearly independent eigenvectors $\vec{v}_1, \cdots, \vec{v}_n$. Then $V = \begin{bmatrix} \vec{v}_1 & \cdots & \vec{v}_n \end{bmatrix}$ is invertible and, as we saw last time, the change of variables

$$\vec{z} = V^{-1}\vec{x}$$

results in the new state equations

$$\frac{d}{dt}\vec{z}(t) = \underbrace{\begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}}_{A_{\text{new}}} \vec{z}(t) + \underbrace{V^{-1}B}_{B_{\text{new}}} u(t).$$

Since A_{new} is diagonal we apply the result above for the diagonal case and obtain

$$\vec{z}_d(k+1) = \begin{bmatrix} e^{\lambda_1 T} & & \\ & \ddots & \\ & & e^{\lambda_n T} \end{bmatrix} \vec{z}_d(k) + \begin{bmatrix} \int_0^T e^{\lambda_1 s} ds & & \\ & \ddots & \\ & & \int_0^T e^{\lambda_n s} ds \end{bmatrix} V^{-1} B u_d(k).$$

To return to the original state variables, note that

$$\vec{x}_d(k) = V\vec{z}_d(k), \quad \vec{z}_d(k) = V^{-1}\vec{x}_d(k),$$

and, therefore,

$$\begin{aligned} \vec{x}_d(k+1) &= V\vec{z}_d(k+1) \\ &= V\left(\begin{bmatrix} e^{\lambda_1 T} & & \\ & \ddots & \\ & & e^{\lambda_n T} \end{bmatrix} \vec{z}_d(k) + \begin{bmatrix} \int_0^T e^{\lambda_1 s} ds & & \\ & \ddots & \\ & & \int_0^T e^{\lambda_n s} ds \end{bmatrix} V^{-1} B u_d(k) \right) \\ &= \underbrace{V \begin{bmatrix} e^{\lambda_1 T} & & \\ & \ddots & \\ & & e^{\lambda_n T} \end{bmatrix} V^{-1}}_{= A_d} \vec{x}_d(k) + \underbrace{V \begin{bmatrix} \int_0^T e^{\lambda_1 s} ds & & \\ & \ddots & \\ & & \int_0^T e^{\lambda_n s} ds \end{bmatrix} V^{-1} B}_{= B_d} u_d(k). \end{aligned} \tag{8}$$

Summary: If A in (1) has linearly independent eigenvectors $\vec{v}_1, \cdots, \vec{v}_n$ with corresponding eigenvalues $\lambda_1, \cdots, \lambda_n$, then we form the invertible matrix $V = \begin{bmatrix} \vec{v}_1 & \cdots & \vec{v}_n \end{bmatrix}$ and obtain the discretized model (4) where A_d and B_d are as in (8).

[2] Recall that A is diagonalizable if it has distinct eigenvalues. If there are repeated eigenvalues A may or may not be diagonalizable: we need to check whether it has n linearly independent eigenvectors or not.

Example 1: Consider the system (1) with

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

and no input. The LC circuit model studied in Lecture 4A with $L = 1$, $C = 1$ had this form. As shown then, the eigenvalues/vectors are

$$\lambda_1 = j, \quad \lambda_2 = -j, \quad \vec{v}_1 = \begin{bmatrix} 1 \\ -j \end{bmatrix}, \quad \vec{v}_2 = \begin{bmatrix} 1 \\ j \end{bmatrix}.$$

Thus,

$$V = \begin{bmatrix} \vec{v}_1 & \cdots & \vec{v}_n \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -j & j \end{bmatrix} \quad \text{and} \quad V^{-1} = \frac{1}{2j} \begin{bmatrix} j & -1 \\ j & 1 \end{bmatrix}.$$

Then, from (8),

$$\begin{aligned} A_d &= V \begin{bmatrix} e^{\lambda_1 T} & \\ & e^{\lambda_2 T} \end{bmatrix} V^{-1} = V \begin{bmatrix} e^{jT} & \\ & e^{-jT} \end{bmatrix} V^{-1} \\ &= \frac{1}{2j} \begin{bmatrix} 1 & 1 \\ -j & j \end{bmatrix} \begin{bmatrix} e^{jT} & \\ & e^{-jT} \end{bmatrix} \begin{bmatrix} j & -1 \\ j & 1 \end{bmatrix} \\ &= \frac{1}{2j} \begin{bmatrix} 1 & 1 \\ -j & j \end{bmatrix} \begin{bmatrix} je^{jT} & -e^{jT} \\ je^{-jT} & e^{-jT} \end{bmatrix} \\ &= \frac{1}{2j} \begin{bmatrix} j(e^{jT} + e^{-jT}) & -(e^{jT} - e^{-jT}) \\ -j^2(e^{jT} - e^{-jT}) & j(e^{jT} + e^{-jT}) \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{2}(e^{jT} + e^{-jT}) & -\frac{1}{2j}(e^{jT} - e^{-jT}) \\ \frac{1}{2j}(e^{jT} - e^{-jT}) & \frac{1}{2}(e^{jT} + e^{-jT}) \end{bmatrix} \\ &= \begin{bmatrix} \cos T & -\sin T \\ \sin T & \cos T \end{bmatrix}. \end{aligned}$$

Controllability

The solution of the discrete-time state model

$$\vec{x}(t+1) = A\vec{x}(t) + B\vec{u}(t), \tag{9}$$

where $\vec{x}(t)$ is an n-dimensional vector, can be obtained recursively as:

$$\begin{aligned} \vec{x}(1) &= A\vec{x}(0) + B\vec{u}(0) \\ \vec{x}(2) &= A\vec{x}(1) + B\vec{u}(1) = A(A\vec{x}(0) + B\vec{u}(0)) + B\vec{u}(1) \\ &= A^2\vec{x}(0) + AB\vec{u}(0) + B\vec{u}(1) \\ \vec{x}(3) &= A\vec{x}(2) + B\vec{u}(2) = A(A^2\vec{x}(0) + AB\vec{u}(0) + B\vec{u}(1)) + B\vec{u}(2) \\ &= A^3\vec{x}(0) + A^2B\vec{u}(0) + AB\vec{u}(1) + B\vec{u}(2) \\ &\vdots \\ \vec{x}(t) &= A^t\vec{x}(0) + A^{t-1}B\vec{u}(0) + A^{t-2}B\vec{u}(1) + \cdots + AB\vec{u}(t-2) + B\vec{u}(t-1) \end{aligned}$$

or, equivalently,

$$\vec{x}(t) = A^t\vec{x}(0) + \begin{bmatrix} B & AB & \cdots & A^{t-2}B & A^{t-1}B \end{bmatrix} \begin{bmatrix} \vec{u}(t-1) \\ \vec{u}(t-2) \\ \vdots \\ \vec{u}(1) \\ \vec{u}(0) \end{bmatrix}. \quad (10)$$

Can we find an input sequence $\vec{u}(0), \vec{u}(1), \ldots, \vec{u}(t-1)$ that brings the state from $\vec{x}(0)$ to any desired value $\vec{x}(t) = \vec{x}_{\text{target}}$ at some time t? If the answer is yes for any $\vec{x}_{\text{target}} \in \mathbb{R}^n$, the system is called *controllable*. Otherwise, the system is called *uncontrollable*. More precisely:

Definition. If, for every $\vec{x}_{\text{target}} \in \mathbb{R}^n$, there exist a t and an input sequence $\vec{u}(0), \vec{u}(1), \ldots, \vec{u}(t-1)$ such that $x(t) = \vec{x}_{\text{target}}$, then the system is *controllable*. If, for some $\vec{x}_{\text{target}} \in \mathbb{R}^n$, there exist no t and no input sequence $\vec{u}(0), \vec{u}(1), \ldots, \vec{u}(t-1)$ such that $x(t) = \vec{x}_{\text{target}}$, then the system is *uncontrollable*.

To investigate controllability further we assume the system has a single input, that is B is a column vector $\vec{b} \in \mathbb{R}^n$, and rewrite (10) as

$$\vec{x}(t) - A^t\vec{x}(0) = \begin{bmatrix} \vec{b} & A\vec{b} & \cdots & A^{t-2}\vec{b} & A^{t-1}\vec{b} \end{bmatrix} \begin{bmatrix} u(t-1) \\ u(t-2) \\ \vdots \\ u(1) \\ u(0) \end{bmatrix}. \quad (11)$$

Achieving $x(t) = \vec{x}_{\text{target}}$ means making the left hand side equal to $\vec{x}_{\text{target}} - A^t\vec{x}(0)$. Thus, the system is controllable if we can arbitrarily assign the the left hand side to any desired vector in $\mathbb{R}^n$ with an appropriate choice of t and input sequence $u(0), u(1), \ldots, u(t-1)$.

This means that the system is controllable if the column space of

$$\begin{bmatrix} \vec{b} & A\vec{b} & \cdots & A^{t-2}\vec{b} & A^{t-1}\vec{b} \end{bmatrix}, \quad (12)$$

that is $\text{span}\{\vec{b}, A\vec{b}, \cdots, A^{t-2}\vec{b}, A^{t-1}\vec{b}\}$, is $\mathbb{R}^n$ for some t.

Note that (12) has t columns. Since we can't span $\mathbb{R}^n$ with fewer than n columns, we must try $t = n$ or higher to check whether the span is $\mathbb{R}^n$. However, as we will prove later, if the n columns

$$\vec{b}, A\vec{b}, \cdots, A^{n-2}\vec{b}, A^{n-1}\vec{b}$$

do not already span $\mathbb{R}^n$, adding more columns $A^n\vec{b}, A^{n+1}\vec{b}, \cdots$ will not enlarge the span to $\mathbb{R}^n$. This leads to the following conclusion:

$$\text{Controllability} \quad \Leftrightarrow \quad \text{span}\{\vec{b}, A\vec{b}, \cdots, A^{n-2}\vec{b}, A^{n-1}\vec{b}\} = \mathbb{R}^n.$$

We will further discuss this condition and its proof in the next lecture; for now we illustrate it with two examples.

Example 2: The system

$$\vec{x}(t+1) = \underbrace{\begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}}_{A} \vec{x}(t) + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{\vec{b}} u(t),$$

where $n = 2$, is controllable because

$$\vec{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{and} \quad A\vec{b} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

are linearly independent and together span $\mathbb{R}^2$. If we wish to reach $\vec{x}_{\text{target}}$ from $\vec{x}(0)$ we can do so in $t = 2$ steps by solving

$$\vec{x}_{\text{target}} - A^2\vec{x}(0) = \begin{bmatrix} \vec{b} & A\vec{b} \end{bmatrix} \begin{bmatrix} u(1) \\ u(0) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} u(1) \\ u(0) \end{bmatrix}$$

for $u(0)$ and $u(1)$:

$$\begin{bmatrix} u(1) \\ u(0) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 2 \end{bmatrix}^{-1} (\vec{x}_{\text{target}} - A^2\vec{x}(0))$$

Example 3: The system

$$\vec{x}(t+1) = \underbrace{\begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}}_{A} \vec{x}(t) + \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{\vec{b}} u(t),$$

where only $\vec{b}$ is different from Example 2, is *uncontrollable* because

$$A\vec{b} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

which is the same as $\vec{b}$, therefore $\text{span}\{\vec{b}, A\vec{b}\} \neq \mathbb{R}^2$. You can see that adding $A^2\vec{b}, A^3\vec{b}, \cdots$ does not enlarge the span, because all of these vectors are the same as $\vec{b}$.

The reason for uncontrollability becomes clear if we write the equation for the second state variable $x_2(t)$ explicitly:

$$x_2(t+1) = 2x_2(t).$$

The right hand side doesn't depend on $u(t)$ or $x_1(t)$, which means that $x_2(t)$ evolves independently and can be influenced neither directly by input $u(t)$, nor indirectly through the other state $x_1(t)$.

EE16B - Spring'20 - Lecture 8B Notes[1]

Murat Arcak

12 March 2020

Controllability and System Identification

Controllability Continued

Recall from the last lecture that the discrete-time system

$$\vec{x}(t+1) = A\vec{x}(t) + B\vec{u}(t), \quad \vec{x}(t) \in \mathbb{R}^n \tag{1}$$

is called controllable if, for every $\vec{x}_{\text{target}} \in \mathbb{R}^n$, there exist a t and an input sequence $\vec{u}(0), \vec{u}(1), \ldots, \vec{u}(t-1)$ such that $x(t) = \vec{x}_{\text{target}}$.

To investigate controllability we assumed B is a column vector $\vec{b} \in \mathbb{R}^n$, and wrote the solution of (1) as

$$\vec{x}(t) - A^t\vec{x}(0) = \begin{bmatrix} \vec{b} & A\vec{b} & \cdots & A^{t-1}\vec{b} \end{bmatrix} \begin{bmatrix} u(t-1) \\ u(t-2) \\ \vdots \\ u(0) \end{bmatrix}. \tag{2}$$

Next we observed that the system is controllable if

$$\text{span}\{\vec{b}, A\vec{b}, \cdots, A^{t-1}\vec{b}\} = \mathbb{R}^n \text{ for some } t \tag{3}$$

because, then, we can choose $u(0), \cdots, u(t-1)$ to match the right-hand side of (2) to $\vec{x}_{\text{target}} - A^t\vec{x}(0)$ for any $\vec{x}_{\text{target}} \in \mathbb{R}^n$.

Henceforth we assume $\vec{b} \neq 0$, since it is trivial to conclude uncontrollability otherwise. We also assume $n \geq 2$, since $b \neq 0$ already guarantees controllability when $n = 1$.

Now imagine an algorithm that starts with $t = 1$, checks if (3) holds; if not, increments t by one and checks (3) again, and so on. Two scenarios are possible:

1. The span grows with every increase in t up to and including $t = n$, at which point we have n linearly independent columns:

$$\text{span}\{\vec{b}, A\vec{b}, \cdots, A^{n-2}\vec{b}, A^{n-1}\vec{b}\} = \mathbb{R}^n.$$

Therefore, the system is controllable.

2. The span grows with every increase in t up to and including $t = m$, where $m < n$, and incrementing t to $t = m + 1$ does not grow the span further – the dimension is still m.

In the second scenario we may be tempted to increase t further and expect that the span may eventually start growing again. This, however, is futile and the dimension of the span will be stuck at $m < n$ no matter how much we increase t.

Here is why: since the span stopped growing when t was raised from $t = m$ to $t = m + 1$, this means the new column $A^m\vec{b}$ was a linear combination of the previous columns $\vec{b}, A\vec{b}, \cdots, A^{m-1}\vec{b}$, that is

$$A^m\vec{b} = \alpha_0\vec{b} + \alpha_1 A\vec{b} + \cdots + \alpha_{m-1}A^{m-1}\vec{b} \tag{4}$$

for some coefficients $\alpha_0, \alpha_1, \ldots, \alpha_{m-1}$. Raising t further to $m+2$ means adding the new column $A^{m+1}\vec{b}$, but

$$A^{m+1}\vec{b} = A(A^m\vec{b}) = \alpha_0 A\vec{b} + \alpha_1 A^2\vec{b} + \cdots + \alpha_{m-1}A^m\vec{b}$$

and substituting (4) for the last term in this sum, we see that $A^{m+1}\vec{b}$ is still a linear combination of $\vec{b}, A\vec{b}, \cdots, A^{m-1}\vec{b}$. The same argument applies to subsequent columns $A^{m+2}\vec{b}$, $A^{m+3}\vec{b}, \ldots$, which means that the dimension of the span remains stuck at m.

Therefore, in scenario 2, the span in (3) will not reach $\mathbb{R}^n$ no matter how much we increase t, and the system is uncontrollable.

It follows from the discussion above that, instead of checking (3) for varying values of t, we need to only check it for $t = n$. If it holds for $t = n$, then scenario 1 applies and the systems is controllable. If not, scenario 2 applies and the system is uncontrollable. This leads to the following simplified controllability test:

$$\text{Controllability} \quad \Leftrightarrow \quad \text{span}\{\vec{b}, A\vec{b}, \cdots, A^{n-2}\vec{b}, A^{n-1}\vec{b}\} = \mathbb{R}^n.$$

Extensions to Multi-Input and Continuous-Time Systems

The controllability test above was derived for the single-input case where B is a single column $\vec{b}$. The same test is also applicable to a multi-input system[2] where B is $n \times m$. In this case we form the *controllability matrix*

$$C = \begin{bmatrix} B & AB & \cdots A^{n-1}B \end{bmatrix}$$

which now has nm columns, and check whether its column space is $\mathbb{R}^n$. The system is controllable if so, and uncontrollable otherwise.

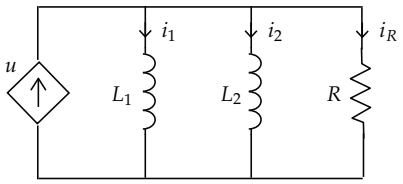
The controllability condition for the continuous-time system

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + B\vec{u}(t)$$

[2] The derivation for the multi-input case uses the Cayley-Hamilton Theorem that was alluded to in Discussion 8B. This theorem is beyond the scope of this course, but you can consult the Wikipedia article if you are interested.

is exactly the same: form the controllability matrix C above and check whether its column space is $\mathbb{R}^n$. We omit the derivation for this case but illustrate the result with a circuit example.

Example 1: For the circuit depicted on the right we treat the current source as the control $u(t)$, and the inductor currents $i_1(t)$ and $i_2(t)$ as the state variables.



Since the voltage across each capacitor is the same as the voltage across the resistor, we have

$$\begin{aligned} L_1 \frac{di_1(t)}{dt} &= Ri_R(t) \\ L_2 \frac{di_2(t)}{dt} &= Ri_R(t). \end{aligned} \tag{5}$$

Substituting $i_R = u - i_1 - i_2$ from KCL and dividing the equations by L_1 and L_2 respectively, we get

$$\begin{bmatrix} \frac{di_1(t)}{dt} \\ \frac{di_2(t)}{dt} \end{bmatrix} = \underbrace{\begin{bmatrix} -\frac{R}{L_1} & -\frac{R}{L_1} \\ -\frac{R}{L_2} & -\frac{R}{L_2} \end{bmatrix}}_{A} \begin{bmatrix} i_1(t) \\ i_2(t) \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{R}{L_1} \\ \frac{R}{L_2} \end{bmatrix}}_{\vec{b}} u(t).$$

Note that

$$A\vec{b} = \begin{bmatrix} -\frac{R}{L_1}\left(\frac{R}{L_1} + \frac{R}{L_2}\right) \\ -\frac{R}{L_2}\left(\frac{R}{L_1} + \frac{R}{L_2}\right) \end{bmatrix} = -\left(\frac{R}{L_1} + \frac{R}{L_2}\right)\vec{b}$$

which means that $A\vec{b}$ and $\vec{b}$ are linearly dependent. Thus the system is *not* controllable.

To see the physical obstacle to controllability note that the two inductors in parallel share the same voltage:

$$L_1 \frac{di_1(t)}{dt} = L_2 \frac{di_2(t)}{dt}.$$

Thus,

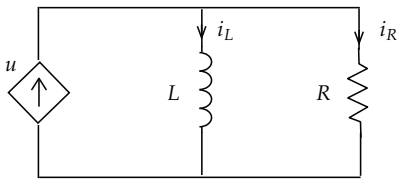
$$\frac{d}{dt}\left(L_1 i_1(t) - L_2 i_2(t)\right) = 0$$

which means that $L_1 i_1(t) - L_2 i_2(t)$ remains constant no matter what u we apply: $L_1 i_1(t) - L_2 i_2(t) = L_1 i_1(0) - L_2 i_2(0)$. Because of this constraint we can't control i_1 and i_2 independently. For example, if $i_1(0) = i_2(0) = 0$, then $L_1 i_1(t) - L_2 i_2(t) = 0$ for all t, and we can't move i_1 and i_2 to target values that don't meet this constraint.

We can, however, control the total current $i_L = i_1 + i_2$ which obeys, from (5),

$$\frac{di_L(t)}{dt} = \left(\frac{1}{L_1} + \frac{1}{L_2}\right) Ri_R(t) = \frac{R}{L}(-i_L(t) + u(t))$$

where $L \triangleq \left(\frac{1}{L_1} + \frac{1}{L_2}\right)^{-1}$. Note that this is the governing equation for the circuit on the right where the two inductors are lumped into one.



System Identification

In many applications the matrices A and B in the state model

$$\vec{x}(t+1) = A\vec{x}(t) + B\vec{u}(t)$$

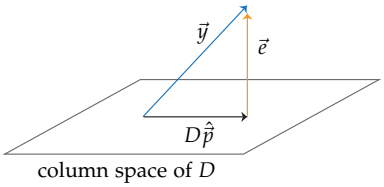
are not known exactly and change with operating conditions. The goal in system identification is to learn these matrices by observing the input sequence and the resulting state sequence.

Before we explain how this is done, let's recall Least Squares estimation from 16A. Suppose we have the relation

$$\vec{y} = D\vec{p} + \vec{e} \tag{6}$$

where $\vec{y} \in \mathbb{R}^\ell$ is a vector of measurements, $\vec{p} \in \mathbb{R}^k$ is a vector of unknown parameters, D is a known $\ell \times k$ matrix, and $\vec{e}$ represents uncertainty, *e.g.*, due to measurement error. We assume $k < \ell$, which means we have fewer unknowns than measurements.

Least Squares gives an estimate $\hat{\vec{p}}$ such that $D\hat{\vec{p}}$ is as close to $\vec{y}$ as possible, *i.e.*, $\hat{\vec{p}}$ fits the measurements with the least magnitude of error, $\|\vec{e}\|$. As you saw in 16A, this is achieved when $D\hat{\vec{p}}$ matches the projection of $\vec{y}$ onto the column space of D, as depicted on the right.



In this case $\vec{e}$ is orthogonal to the column space of D, which means it is orthogonal to each column of $D = [\vec{d}_1 \cdots \vec{d}_k]$:

$$\vec{d}_i^T \vec{e} = 0, \quad i = 1, 2, \ldots, k, \quad \text{or equivalently} \quad D^T\vec{e} = 0.$$

Now, since $\vec{e} = \vec{y} - D\vec{p}$ from (6), $D^T\vec{e} = 0$ means

$$D^T(\vec{y} - D\vec{p}) = 0 \quad \Rightarrow \quad D^TD\vec{p} = D^T\vec{p}.$$

In particular, when D^TD is invertible, the least squares estimate is:

$$\hat{\vec{p}} = (D^TD)^{-1}D^T\vec{y}.$$

Returning to the problem of system identification, let's first consider the scalar system:

$$x(t+1) = \lambda x(t) + bu(t) + e(t)$$

where $e(t)$ is a disturbance term. It follows that

$$\begin{aligned} x(1) &= \lambda x(0) + bu(0) + e(0) \\ x(2) &= \lambda x(1) + bu(1) + e(1) \\ &\vdots \\ x(\ell) &= \lambda x(\ell-1) + bu(\ell-1) + e(\ell-1) \end{aligned}$$

where ℓ is the number of measurements. Rewriting the above as

$$\underbrace{\begin{bmatrix} x(0) & u(0) \\ x(1) & u(1) \\ \vdots & \cdots \\ x(\ell-1) & u(\ell-1) \end{bmatrix}}_{=: D} \underbrace{\begin{bmatrix} \lambda \\ b \end{bmatrix}}_{=: \vec{p}} + \underbrace{\begin{bmatrix} e(0) \\ e(1) \\ \vdots \\ e(\ell-1) \end{bmatrix}}_{=: \vec{e}} = \underbrace{\begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(\ell) \end{bmatrix}}_{=: \vec{y}}$$

we obtain a standard Least Squares problem. Thus, when the 2×2 matrix $D^T D$ is invertible, we obtain the estimates $\hat{\lambda}$, $\hat{b}$ from

$$\hat{\vec{p}} = \begin{bmatrix} \hat{\lambda} \\ \hat{b} \end{bmatrix} = (D^T D)^{-1} D^T \vec{y}.$$

In practice $D^T D$ is invertible when the measurements contain enough information about the unknown parameters. A trivial scenario where $D^T D$ is not invertible is when we apply zero input $u(t) = 0$ for all t. In this case the measurements contain no information about the parameter b, and naturally the estimation problem is ill-posed.

EE16B - Spring'20 - Lecture 9A Notes[1]

Murat Arcak

17 March 2020

Learning

System Identification Continued

In many applications the matrices A and B in the state model

$$\vec{x}(t+1) = A\vec{x}(t) + B\vec{u}(t) \tag{1}$$

are not known exactly and change with operating conditions. The goal in system identification is to learn these matrices by observing the input sequence and the resulting state sequence.

Last time we considered the scalar system:

$$x(t+1) = \lambda x(t) + bu(t) + e(t)$$

where e is a disturbance term. From $t = 1$ to $t = \ell$ we have

$$\begin{aligned} x(1) &= \lambda x(0) + bu(0) + e(0) \\ x(2) &= \lambda x(1) + bu(1) + e(2) \\ &\vdots \\ x(\ell) &= \lambda x(\ell-1) + bu(\ell-1) + e(\ell) \end{aligned} \tag{2}$$

which we rewrite in the following standard form for Least Squares:

$$\underbrace{\begin{bmatrix} x(0) & u(0) \\ x(1) & u(1) \\ \vdots & \cdots \\ x(\ell-1) & u(\ell-1) \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} \lambda \\ b \end{bmatrix}}_{\vec{p}} + \underbrace{\begin{bmatrix} e(0) \\ e(1) \\ \vdots \\ e(\ell-1) \end{bmatrix}}_{\vec{e}} = \underbrace{\begin{bmatrix} x(1) \\ x(2) \\ \vdots \\ x(\ell) \end{bmatrix}}_{\vec{y}}. \tag{3}$$

Thus, when D^TD is invertible, we obtain the estimates $\hat{\lambda}$, $\hat{b}$ from

$$\hat{\vec{p}} = \begin{bmatrix} \hat{\lambda} \\ \hat{b} \end{bmatrix} = (D^TD)^{-1}D^T\vec{y}.$$

Now let's return to the vector case (1), with disturbance $\vec{e}(t)$ added to the right-hand side. The equations below are analogous to (2):

$$\begin{aligned} \vec{x}(1) &= A\vec{x}(0) + B\vec{u}(0) + \vec{e}(0) \\ \vec{x}(2) &= A\vec{x}(1) + B\vec{u}(1) + \vec{e}(1) \\ &\vdots \\ \vec{x}(\ell) &= A\vec{x}(\ell-1) + B\vec{u}(\ell-1) + \vec{e}(\ell-1). \end{aligned} \tag{4}$$

If we transpose these equations, we get:

$$\begin{aligned}\vec{x}(1)^T &= \vec{x}(0)^T A^T + \vec{u}(0)^T B^T + \vec{e}(0)^T \\ \vec{x}(2)^T &= \vec{x}(1)^T A^T + \vec{u}(1)^T B^T + \vec{e}(1)^T \\ &\vdots \\ \vec{x}(\ell)^T &= \vec{x}(\ell-1)^T A^T + \vec{u}(\ell-1)^T B^T + \vec{e}(\ell-1)^T,\end{aligned} \tag{5}$$

which we can rewrite in a form similar to (3):

$$\underbrace{\begin{bmatrix} \vec{x}(0)^T & \vec{u}(0)^T \\ \vec{x}(1)^T & \vec{u}(1)^T \\ \vdots & \cdots \\ \vec{x}(\ell-1)^T & \vec{u}(\ell-1)^T \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} A^T \\ B^T \end{bmatrix}}_{[\vec{p}_1 \cdots \vec{p}_n]} + \underbrace{\begin{bmatrix} \vec{e}(0)^T \\ \vec{e}(1)^T \\ \vdots \\ \vec{e}(\ell-1)^T \end{bmatrix}}_{[\vec{e}_1 \cdots \vec{e}_n]} = \underbrace{\begin{bmatrix} \vec{x}(1)^T \\ \vec{x}(2)^T \\ \vdots \\ \vec{x}(\ell)^T \end{bmatrix}}_{[\vec{y}_1 \cdots \vec{y}_n]}. \tag{6}$$

Note that the unknowns and measurements are now contained in matrices with n columns:

$$\begin{bmatrix} A^T \\ B^T \end{bmatrix} =: \begin{bmatrix} \vec{p}_1 & \cdots & \vec{p}_n \end{bmatrix} \quad \begin{bmatrix} \vec{x}(1)^T \\ \vec{y}(x)^T \\ \vdots \\ \vec{x}(\ell)^T \end{bmatrix} = \begin{bmatrix} x_1(1) & \cdots & x_n(1) \\ x_1(2) & \cdots & x_n(2) \\ \vdots & & \vdots \\ x_1(\ell) & \cdots & x_n(\ell) \end{bmatrix} =: \begin{bmatrix} \vec{y}_1 & \cdots & \vec{y}_n \end{bmatrix}.$$

Thus, we can separate (6) into n separate equations

$$D\vec{p}_i + \vec{e}_i = \vec{y}_i, \quad i = 1, 2, \ldots, n$$

and apply Least Squares to each one independently:

$$\hat{\vec{p}}_i = (D^T D)^{-1} D^T \vec{y}_i.$$

Note that $\vec{y}_i$ here is a column consisting of measurements of the ith state variable collected from $t = 1$ to $t = \ell$, and $\hat{\vec{p}}_i$ is our estimate for the ith column of A^T concatenated with the ith column of B^T.

Singular Value Decomposition (SVD)

SVD separates a rank-r matrix $A \in \mathbb{R}^{m \times n}$ into a sum of r rank-1 matrices, each written as a column times row. Specifically, we can find:

1) orthonormal vectors $\vec{u}_1, \ldots, \vec{u}_r \in \mathbb{R}^m$,

2) orthonormal vectors $\vec{v}_1, \ldots, \vec{v}_r \in \mathbb{R}^n$,

3) real, positive numbers $\sigma_1, \ldots, \sigma_r$ such that

$$\boxed{A = \sigma_1 \vec{u}_1 \vec{v}_1^T + \sigma_2 \vec{u}_2 \vec{v}_2^T + \cdots + \sigma_r \vec{u}_r \vec{v}_r^T.} \tag{7}$$

The numbers $\sigma_1, \ldots, \sigma_r$ are called *singular values* and, by convention, we order them from the largest to smallest:

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0.$$

In its original form A has mn entries to be stored. In the SVD form each of the r terms is the product of a column of m entries with a row of n entries; therefore we need $r(m+n)$ numbers to store. This is an advantage when r is small relative to m and n, that is $r(m+n) \ll mn$.

In a typical application the exact rank r may not be particularly small, but we may find that the first few singular values, say $\sigma_1, \ldots, \sigma_{\hat{r}}$, are much bigger than the rest, $\sigma_{\hat{r}+1}, \ldots, \sigma_r$. Then it is reasonable to discard the small singular values and approximate A as

$$A \approx \sigma_1 \vec{u}_1 \vec{v}_1^T + \sigma_2 \vec{u}_2 \vec{v}_2^T + \cdots + \sigma_{\hat{r}} \vec{u}_{\hat{r}} \vec{v}_{\hat{r}}^T \tag{8}$$

which has rank $= \hat{r}$, thus $\hat{r}(m+n) \ll mn$ numbers to store.

Besides enabling data compression, SVD allows us to extract important features of a data set as illustrated in the next example.

Example (Netflix): Suppose we have a $m \times n$ matrix that contains the ratings of m viewers for n movies. A truncated SVD as suggested above not only saves memory; it also gives insight into the preferences of each viewer. For example we can interpret each rank-1 matrix $\sigma_i \vec{u}_i \vec{v}_i^T$ to be due to a particular attribute, *e.g.*, comedy, action, sci-fi, or romance content. Then σ_i determines how strongly the ratings depend on the ith attribute, the entries of $\vec{v}_i^T$ score each movie with respect to this attribute, and the entries of $\vec{u}_i$ evaluate how much each viewer cares about this particular attribute. Then truncating the SVD as in (8) amounts to identifying a few key attributes that underlie the ratings. This is useful, for example, in making movie recommendations as you will see in a homework problem.

Finding a SVD

To find a SVD for A we use either the $n \times n$ matrix $A^T A$ or the $m \times m$ matrix AA^T. We will see later that these matrices have only *real eigenvalues*, r of which are positive and the remaining zero, and a complete set of *orthonormal eigenvectors*. For now we take this as a fact and outline the following procedure to find a SVD using $A^T A$:

1. Find the eigenvalues λ_i of $A^T A$ and order them from the largest to smallest, so that $\lambda_1 \geq \cdots \geq \lambda_r > 0$ and $\lambda_{r+1} = \cdots = \lambda_n = 0$.

2. Find orthonormal eigenvectors $\vec{v}_i$, so that

$$A^T A \, \vec{v}_i = \lambda_i \vec{v}_i \quad i = 1, \ldots, r. \tag{9}$$

3. Let $\sigma_i = \sqrt{\lambda_i}$ and obtain $\vec{u}_i$ from

$$A\vec{v}_i = \sigma_i \vec{u}_i \quad i = 1, \ldots, r. \tag{10}$$

We will provide a justification for this procedure in the next lecture. For now we provide an example:

Example: Let

$$A = \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix}.$$

Since this matrix is rank-1 it is not difficult to write it as a column times row, but we will instead practice the general procedure above. Note that

$$A^T A = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 3 & 6 \\ 6 & 12 \end{bmatrix}$$

and the eigenvalues of $A^T A$ are obtained from:

$$\det(\lambda I - A) = \det \begin{bmatrix} \lambda - 3 & -6 \\ -6 & \lambda - 12 \end{bmatrix} = \lambda^2 - 15\lambda = \lambda(\lambda - 15) = 0.$$

Therefore, $\lambda_1 = 15$ and $\lambda_2 = 0$. Next we find an eigenvector $\vec{v}_1$ from

$$\begin{bmatrix} \lambda_1 - 3 & -6 \\ -6 & \lambda_1 - 12 \end{bmatrix} \vec{v}_1 = \begin{bmatrix} 12 & -6 \\ -6 & 3 \end{bmatrix} \vec{v}_1 = 0,$$

with length normalized to one:

$$\vec{v}_1 = \frac{1}{\sqrt{5}} \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

We compute the singular value from $\sigma_1 = \sqrt{\lambda_1} = \sqrt{15}$, and $\vec{u}_1$ from (10):

$$\vec{u}_1 = \frac{1}{\sigma_1} A\vec{v}_1 = \frac{1}{\sqrt{15}} \frac{1}{\sqrt{5}} \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \frac{1}{\sqrt{15}} \frac{1}{\sqrt{5}} \begin{bmatrix} 5 \\ 5 \\ 5 \end{bmatrix} = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

Thus we have obtained the SVD:

$$A = \sigma_1 \vec{u}_1 \vec{v}_1^T = \sqrt{15} \begin{bmatrix} \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix}.$$

EE16B - Spring'20 - Lecture 9B Notes[1]

Murat Arcak

19 March 2020

Singular Value Decomposition (SVD)

Recall that SVD separates a rank-r matrix $A \in \mathbb{R}^{m\times n}$ into a sum of r rank-1 matrices:

$$\boxed{A = \sigma_1 \vec{u}_1 \vec{v}_1^T + \sigma_2 \vec{u}_2 \vec{v}_2^T + \cdots + \sigma_r \vec{u}_r \vec{v}_r^T} \tag{1}$$

where $\vec{u}_1, \ldots, \vec{u}_r \in \mathbb{R}^m$ are orthonormal, $\vec{v}_1, \ldots, \vec{v}_r \in \mathbb{R}^n$ are orthonormal, and $\sigma_1, \ldots, \sigma_r$ are real, positive numbers called *singular values*. By convention, we order them from the largest to smallest:

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0.$$

Finding a SVD

To find a SVD of the form (1) we use either the $n \times n$ matrix A^TA or the $m \times m$ matrix AA^T. We will see later that these matrices have only *real eigenvalues*, r of which are positive and the remaining zero, and a complete set of *orthonormal eigenvectors*. For now we take this as a fact and propose the following procedures to find a SVD for A:

SVD procedure using A^TA:

1. Find the eigenvalues λ_i of A^TA and order them from the largest to smallest, so that $\lambda_1 \geq \cdots \geq \lambda_r > 0$ and $\lambda_{r+1} = \cdots = \lambda_n = 0$.

2. Find orthonormal eigenvectors $\vec{v}_i$, so that

$$A^TA\,\vec{v}_i = \lambda_i \vec{v}_i \quad i = 1, \ldots, r. \tag{2}$$

3. Let $\sigma_i = \sqrt{\lambda_i}$ and obtain $\vec{u}_i$ from

$$\vec{u}_i = \frac{1}{\sigma_i} A\vec{v}_i \quad i = 1, \ldots, r. \tag{3}$$

Justification of the procedure: As stated above we will see later that, for any rank-r matrix $A \in \mathbb{R}^{m\times n}$, $A^TA \in \mathbb{R}^{n\times n}$ has r positive eigenvalues and $n - r$ eigenvalues at zero, along with orthonormal eigenvectors $\vec{v}_1, \cdots, \vec{v}_n$. Taking these eigenvectors as given, we will show:

1) $\vec{u}_1, \cdots, \vec{u}_r$ computed as in (3) are themselves orthonormal;

2) the right-hand side of (1), with $\vec{v}_i$ and $\vec{u}_i$ generated according to the procedure, indeed matches A.

Now the details for each:

1) To see that $\vec{u}_i$, $i = 1, \ldots, r$, given by (3) are orthonormal, note that:

$$\vec{u}_j^T \vec{u}_i = \frac{1}{\sigma_j \sigma_i} (A\vec{v}_j)^T A\vec{v}_i = \frac{1}{\sigma_j \sigma_i} \vec{v}_j^T A^T A \vec{v}_i = \frac{\lambda_i}{\sigma_j \sigma_i} \vec{v}_j^T \vec{v}_i \tag{4}$$

where, in the last step, we substituted (2). The vectors $\vec{v}_i$, $i = 1, \ldots, r$, are orthonormal by construction, which means $\vec{v}_j^T \vec{v}_i = 1$ if $i = j$, and 0 if $i \neq j$. Thus, (4) becomes

$$\vec{u}_j^T \vec{u}_i = \begin{cases} \frac{\lambda_i}{\sigma_j \sigma_i} = \frac{\lambda_i}{\sigma_i^2} = 1 & \text{if } i = j, \\ 0 & \text{if } \ i \neq j \end{cases} \tag{5}$$

proving orthonormality of $\vec{u}_i$, $i = 1, \ldots, r$.

2) To see why σ_i, $\vec{u}_i$, $\vec{v}_i$ resulting from the procedure above satisfy (1), note that (3) implies $A\vec{v}_i = \sigma_i \vec{u}_i$, which we write in matrix form as:

$$A \underbrace{\begin{bmatrix} \vec{v}_1 \cdots \vec{v}_r \end{bmatrix}}_{=: V_1} = \begin{bmatrix} \vec{u}_1 \cdots \vec{u}_r \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix}.$$

Next, multiply both sides from the right by V_1^T:

$$\begin{aligned} AV_1V_1^T &= \begin{bmatrix} \vec{u}_1 \cdots \vec{u}_r \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix} \underbrace{\begin{bmatrix} \vec{v}_1^T \\ \vdots \\ \vec{v}_r^T \end{bmatrix}}_{V_1^T} \qquad (6) \\ &= \sigma_1 \vec{u}_1 \vec{v}_1^T + \sigma_2 \vec{u}_2 \vec{v}_2^T + \cdots + \sigma_r \vec{u}_r \vec{v}_r^T. \qquad (7) \end{aligned}$$

Since the right hand side is indeed the decomposition in (1), we need to show that the left hand side is equal to A, that is $AV_1V_1^T = A$. To this end define $V_2 = \begin{bmatrix} \vec{v}_{r+1} & \cdots & \vec{v}_n \end{bmatrix}$ whose columns are the remaining orthonormal eigenvectors for $\lambda_{r+1} = \cdots = \lambda_n = 0$. Then $V = \begin{bmatrix} V_1 & V_2 \end{bmatrix}$ is an orthonormal matrix and, thus,

$$VV^T = \begin{bmatrix} V_1 & V_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} = V_1V_1^T + V_2V_2^T = I.$$

Multiplying both sides from the left by A, we get

$$AV_1V_1^T + AV_2V_2^T = A. \tag{8}$$

Since the columns of V_2 are eigenvectors of A^TA for zero eigenvalues we have $A^TAV_2 = 0$, and multiplying this from the left by V_2^T we get $V_2^TA^TAV_2 = (AV_2)^T(AV_2) = 0$. This implies $AV_2 = 0$ and it follows

from (8) that $AV_1V_1^T = A$. Thus, the left hand side of (6) is A, which proves that σ_i, $\vec{u}_i$, $\vec{v}_i$ proposed by the procedure above satisfy (1). □

An alternative approach is to use the $m \times m$ matrix AA^T which is preferable to using the $n \times n$ matrix A^TA when $m < n$. Below we summarize the procedure and leave its justification as an exercise.

SVD procedure using AA^T:

1. Find the eigenvalues λ_i of AA^T and order them from the largest to smallest, so that $\lambda_1 \geq \cdots \geq \lambda_r > 0$ and $\lambda_{r+1} = \cdots = \lambda_m = 0$.

2. Find orthonormal eigenvectors $\vec{u}_i$, so that

$$AA^T\vec{u}_i = \lambda_i\vec{u}_i \quad i = 1, \ldots, r. \tag{9}$$

3. Let $\sigma_i = \sqrt{\lambda_i}$ and obtain $\vec{v}_i$ from

$$\vec{v}_i = \frac{1}{\sigma_i}A^T\vec{u}_i \quad i = 1, \ldots, r. \tag{10}$$

Example: Let's follow this procedure to find a SVD for

$$A = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}.$$

We calculate

$$AA^T = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}\begin{bmatrix} 4 & -3 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 32 & 0 \\ 0 & 18 \end{bmatrix}$$

which happens to be diagonal, so the eigenvalues are $\lambda_1 = 32$, $\lambda_2 = 18$, and we can select the orthonormal eigenvectors:

$$\vec{u}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \vec{u}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{11}$$

The singular values are $\sigma_1 = \sqrt{\lambda_1} = 4\sqrt{2}$, $\sigma_2 = \sqrt{\lambda_2} = 3\sqrt{2}$ and, from (10),

$$\vec{v}_1 = \frac{1}{\sigma_1}A^T\vec{u}_1 = \frac{1}{4\sqrt{2}}\begin{bmatrix} 4 \\ 4 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix},$$
$$\vec{v}_2 = \frac{1}{\sigma_2}A^T\vec{u}_2 = \frac{1}{3\sqrt{2}}\begin{bmatrix} -3 \\ 3 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

which are indeed orthonormal. We leave it as an exercise to derive a SVD using, instead, A^TA.

Note that we can change the signs of $\vec{u}_1$ and $\vec{u}_2$ in (11), and they still serve as orthonormal eigenvectors. This implies that SVD is not

unique. However, changing the sign of $\vec{u}_i$ changes the sign of $\vec{v}_i$ in (10) accordingly, therefore the product $\vec{u}_i\vec{v}_i^T$ remains unchanged.

Another source of non-uniqueness arises when we have repeated singular values, as illustrated in the next example.

Example: To find a SVD for

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

note that AA^T is the identity matrix, which has repeated eigenvalues at $\lambda_1 = \lambda_2 = 1$ and admits any pair of orthonormal vectors as eigenvectors. We parameterize all such pairs as

$$\vec{u}_1 = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \quad \vec{u}_2 = \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix} \tag{12}$$

where θ is a free parameter. Since $\sigma_1 = \sigma_2 = 1$, we obtain from (10):

$$\vec{v}_1 = \frac{1}{\sigma_1}A^T\vec{u}_1 = \begin{bmatrix} \cos\theta \\ -\sin\theta \end{bmatrix} \qquad \vec{v}_2 = \frac{1}{\sigma_2}A^T\vec{u}_2 = \begin{bmatrix} -\sin\theta \\ -\cos\theta \end{bmatrix}. \tag{13}$$

Thus, (12)-(13) with $\sigma_1 = \sigma_2 = 1$ constitute a valid SVD for any choice of θ. You can indeed verify that

$$\begin{aligned} \vec{u}_1\vec{v}_1^T + \vec{u}_2\vec{v}_2^T = \begin{bmatrix} \vec{u}_1 & \vec{u}_2 \end{bmatrix} \begin{bmatrix} \vec{v}_1^T \\ \vec{v}_2^T \end{bmatrix} &= \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta \\ -\sin\theta & -\cos\theta \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \end{aligned} \tag{14}$$

EE16B - Spring'20 - Lecture 11A Notes[1]

Murat Arcak

31 March 2020

Singular Value Decomposition (SVD) Continued

Recall that SVD separates a rank-r matrix $A \in \mathbb{R}^{m\times n}$ into a sum of r rank-1 matrices:

$$\boxed{A = \sigma_1 \vec{u}_1 \vec{v}_1^T + \sigma_2 \vec{u}_2 \vec{v}_2^T + \cdots + \sigma_r \vec{u}_r \vec{v}_r^T} \tag{1}$$

where $\vec{u}_1, \ldots, \vec{u}_r \in \mathbb{R}^m$ are orthonormal, $\vec{v}_1, \ldots, \vec{v}_r \in \mathbb{R}^n$ are orthonormal, and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$.

In most textbooks the SVD (1) is written as a product of three matrices. To derive this alternative form we first rewrite (1) as

$$A = U_1 S V_1^T \tag{2}$$

where $U_1 = \begin{bmatrix} \vec{u}_1 \cdots \vec{u}_r \end{bmatrix}$ is $m \times r$, $V_1 = \begin{bmatrix} \vec{v}_1 \cdots \vec{v}_r \end{bmatrix}$ is $n \times r$, and S is the $r \times r$ diagonal matrix with entries $\sigma_1, \ldots, \sigma_r$:

$$S = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix}.$$

Recall that $\vec{u}_1, \cdots, \vec{u}_r$ correspond to eigenvectors of AA^T for non-zero eigenvalues, and similarly $\vec{v}_1, \cdots, \vec{v}_r$ are eigenvectors of $A^T A$ for non-zero eigenvalues.

Next we form the $m \times m$ orthonormal matrix

$$U = [U_1 \; U_2]$$

where the columns of $U_2 = [\vec{u}_{r+1} \; \cdots \; \vec{u}_m]$ are eigenvectors of AA^T corresponding to *zero* eigenvalues. Likewise we define $V_2 = [\vec{v}_{r+1} \; \cdots \; \vec{v}_n]$ whose columns are orthonormal eigenvectors of $A^T A$ for zero eigenvalues, and obtain the $n \times n$ orthogonal matrix

$$V = [V_1 \; V_2].$$

Then we write

$$\boxed{A = U \underbrace{\begin{bmatrix} S & 0_{r\times(n-r)} \\ 0_{(m-r)\times r} & 0_{(m-r)\times(n-r)} \end{bmatrix}}_{=:\,\Sigma} V^T} \tag{3}$$

which is identical to (2) but exhibits square and orthonormal matrices U and V^T, that is $U^TU = I$ and $V^TV = I$. Having square and orthonormal matrices U and V will allow us to give a geometric interpretation of SVD in the next section.

It is important to understand the dimensions of the matrices in (3). Σ is $m \times n$, same as A. U and V, however, are square: U is $m \times m$ and V is $n \times n$. If A is square ($m = n$), then all three are square. If A is a wide matrix with full row rank ($r = m < n$), then

$$A = U \underbrace{\begin{bmatrix} S & 0_{m\times(n-m)} \end{bmatrix}}_{=\Sigma} V^T$$

If A is a tall matrix with full column rank ($m > n = r$), then

$$A = U \underbrace{\begin{bmatrix} S \\ 0_{(m-n)\times n} \end{bmatrix}}_{=\Sigma} V^T$$

Geometric Interpretation of SVD

Note that multiplying a vector by an orthonormal matrix does not change its length. This follows because $U^TU = I$, which implies

$$\|U\vec{x}\|^2 = (U\vec{x})^T(U\vec{x}) = \vec{x}^TU^TU\vec{x} = \vec{x}^T\vec{x} = \|\vec{x}\|^2.$$

Thus we can interpret multiplication by an orthonormal matrix as a combination of operations that don't change length, such as rotations, and reflections.

Since S is diagonal with entries $\sigma_1, \ldots, \sigma_r$, multiplying a vector by Σ defined in (3) stretches the first entry of the vector by σ_1, the second entry by σ_2, and so on.

Combining these observations we interpret $A\vec{x}$ as the composition of three operations:

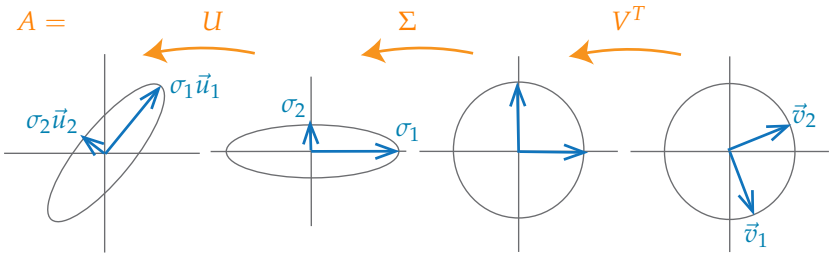
1) $V^T\vec{x}$ which reorients $\vec{x}$ without changing its length,

2) $\Sigma V^T\vec{x}$ which stretches the resulting vector along each axis with the corresponding singular value,

3) $U\Sigma V^T\vec{x}$ which again reorients the resulting vector without changing its length.

The figure below illustrates these three operations moving from the right to the left:

The geometric interpretation above reveals that σ_1 is the largest amplification factor a vector can experience upon multiplication by A:



if the length of $\vec{x}$ is $\|\vec{x}\| = 1$ then $\|A\vec{x}\| \leq \sigma_1$.

For $\vec{x} = \vec{v}_1$ we get $\|A\vec{x}\| = \sigma_1$ with *equality* because $V^T\vec{v}_1$ is the first unit vector which, when multiplied by Σ, gets stretched by σ_1.

Symmetric Matrices

We say that a square matrix Q is *symmetric* if

$$Q = Q^T.$$

Note that the matrices A^TA and AA^T we used to compute a SVD for A are automatically symmetric: using the identities $(AB)^T = B^TA^T$ and $(A^T)^T = A$ you can verify $(A^TA)^T = A^TA$ and $(AA^T)^T = AA^T$.

Below we derive important properties of symmetric matrices that we used without proof in our SVD procedures.

A symmetric matrix has real eigenvalues and eigenvectors.

Let Q be symmetric and let

$$Qx = \lambda x, \tag{4}$$

that is λ is an eigenvalue and x is an eigenvector. Let $\lambda = a + jb$ and define the conjugate $\bar{\lambda} = a - jb$. To show that $b = 0$, that is λ is real, we take conjugates of both sides of $Qx = \lambda x$ to obtain

$$Q\bar{x} = \bar{\lambda}\bar{x} \tag{5}$$

where we used the fact that Q is real. The transpose of (5) is

$$\bar{x}^TQ^T = \bar{\lambda}\bar{x}^T \tag{6}$$

and, since $Q = Q^T$, we write

$$\bar{x}^TQ = \bar{\lambda}\bar{x}^T. \tag{7}$$

Now multiply (4) from the left by $\bar{x}^T$ and (7) from the right by x:

$$\begin{aligned} \bar{x}^TQx &= \lambda\bar{x}^Tx \\ \bar{x}^TQx &= \bar{\lambda}\bar{x}^Tx \end{aligned} \tag{8}$$

Since the left hand sides are the same we have $\lambda \bar{x}^T x = \bar{\lambda} \bar{x}^T x$, and since $\bar{x}^T x \neq 0$, we conclude $\lambda = \bar{\lambda}$. This means $a + jb = a - jb$ which proves that $b = 0$.

Now that we know the eigenvalues are real we can conclude the eigenvectors are also real, because they are obtained from the equation $(Q - \lambda I)x = 0$ where $Q - \lambda I$ is real. □

The eigenvectors can be chosen to be orthonormal.

We will prove this for the case where the eigenvalues are distinct although the statement is true also without this restriction[2]. Orthonormality of the eigenvectors means they are orthogonal and each has unit length. Since we can easily normalize the length to one, we need only to show that the eigenvectors are orthogonal.

[2] A further fact is that a symmetric matrix admits a complete set of eigenvectors even in the case of repeated eigenvalues and is thus diagonalizable.

Pick two eigenvalue-eigenvector pairs: $Qx_1 = \lambda_1 x_1$, $Qx_2 = \lambda_2 x_2$, $\lambda_1 \neq \lambda_2$. Multiply $Qx_1 = \lambda_1 x_1$ from the left by x_2^T, and $Qx_2 = \lambda_2 x_2$ by x_1^T:

$$\begin{aligned} x_2^T Q x_1 &= \lambda_1 x_2^T x_1 \\ x_1^T Q x_2 &= \lambda_2 x_1^T x_2. \end{aligned} \tag{9}$$

Note that $x_2^T Q x_1$ is a scalar, therefore its transpose is equal to itself: $x_1^T Q x_2 = (x_1^T Q x_2)^T = x_2^T Q^T x_1 = x_2^T Q x_1$. This means that the left hand sides of the two equations above are identical, hence

$$\lambda_1 x_2^T x_1 = \lambda_2 x_1^T x_2.$$

Note that $x_1^T x_2 = x_2^T x_1$ is the inner product of x_1 and x_2. Since $\lambda_1 \neq \lambda_2$, the equality above implies that this inner product is zero, that is x_1 and x_2 are orthogonal. □

The final property below proves our earlier assertion that AA^T and $A^T A$ have nonnegative eigenvalues. (Substitute $R = A^T$ below for the former, and $R = A$ for the latter.)

If Q can be written as $Q = R^T R$ for some matrix R, then the eigenvalues of Q are nonnegative.

To show this let x_i be an eigenvector of Q corresponding λ_i, so that

$$R^T R x_i = \lambda_i x_i.$$

Next multiply both sides from the left by x_i^T:

$$x_i^T R^T R x_i = \lambda_i x_i^T x_i = \lambda_i \|x_i\|^2.$$

If we define $y = Rx_i$ we see that the left hand side is $y^T y = \|y\|^2$, which is nonnegative. Thus, $\lambda_i \|x_i\|^2 \geq 0$. Since the eigenvector is nonzero, we have $\|x_i\| \neq 0$ which implies $\lambda_i \geq 0$. □

EE16B - Spring'20 - Lecture 11B Notes[1]

Murat Arcak

2 April 2020

Applications of SVD

Last time we wrote the SVD for the $m \times n$ matrix A as

$$A = U_1 S V_1^T \tag{1}$$

where $U_1 = \begin{bmatrix} \vec{u}_1 \cdots \vec{u}_r \end{bmatrix}$ is $m \times r$, $V_1 = \begin{bmatrix} \vec{v}_1 \cdots \vec{v}_r \end{bmatrix}$ is $n \times r$, and S is the $r \times r$ diagonal matrix with entries $\sigma_1, \ldots, \sigma_r$:

$$S = \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix}.$$

We also formed the square and orthonormal matrices $U = [U_1 \; U_2]$, $V = [V_1 \; V_2]$, and rewrote (1) as

$$A = U \Sigma V^T \tag{2}$$

where Σ is $m \times n$ and subsumes S in its $r \times r$ upper left block:

$$\Sigma = \begin{bmatrix} S & 0_{r \times (n-r)} \\ 0_{(m-r) \times r} & 0_{(m-r) \times (n-r)} \end{bmatrix}.$$

Before discussing some applications of SVD we note that the columns of U_1 in (1) form an orthonormal basis for the column space of A. Similarly the columns of V_2 span the null space of A. This is because they are orthogonal to the columns of V_1, so $V_1^T \vec{x} = 0$ for any $\vec{x}$ that is in the column space of V_2 and, thus, (1) implies $A\vec{x} = 0$.

Least Squares with SVD

Recall that in Least Squares we consider the equation

$$\vec{y} = A\vec{x} + \vec{e}$$

where $\vec{y} \in \mathbb{R}^m$ represents measurements, $\vec{x} \in \mathbb{R}^n$ unknowns, $\vec{e}$ errors.

Typically there are more measurements than unknowns ($m > n$) so $A \in \mathbb{R}^{m \times n}$ is a tall matrix. We also assume A has linearly independent columns, so the rank is $r = n$. Then the SVD of A has the form

$$A = U \underbrace{\begin{bmatrix} S \\ 0_{(m-n) \times n} \end{bmatrix}}_{= \Sigma} V^T. \tag{3}$$

The goal in Least Squares is to find $\vec{x}$ such that $\vec{e} = \vec{y} - A\vec{x}$ has the least possible length. Substituting the SVD (3) for A, note

$$\|\vec{e}\| = \|\vec{y} - A\vec{x}\| = \left\|\vec{y} - U\begin{bmatrix} S \\ 0 \end{bmatrix} V^T\vec{x}\right\|.$$

Since $UU^T = I$ we can replace $\vec{y}$ in this expression with $UU^T\vec{y}$, so that we can factor out U:

$$\|\vec{e}\| = \left\|UU^T\vec{y} - U\begin{bmatrix} S \\ 0 \end{bmatrix} V^T\vec{x}\right\| = \left\|U\left(U^T\vec{y} - \begin{bmatrix} S \\ 0 \end{bmatrix} V^T\vec{x}\right)\right\|. \tag{4}$$

Remembering that multiplication with the orthonormal matrix U does not change the length of a vector, we conclude

$$\|\vec{e}\| = \left\|U^T\vec{y} - \begin{bmatrix} S \\ 0 \end{bmatrix} V^T\vec{x}\right\|.$$

Next note

$$U^T\vec{y} - \begin{bmatrix} S \\ 0 \end{bmatrix} V^T\vec{x} = \begin{bmatrix} U_1^T\vec{y} \\ U_2^T\vec{y} \end{bmatrix} - \begin{bmatrix} SV^T\vec{x} \\ 0 \end{bmatrix} = \begin{bmatrix} U_1^T\vec{y} - SV^T\vec{x} \\ U_2^T\vec{y} \end{bmatrix} \tag{5}$$

and our goal is to minimize the length of this vector by choosing $\vec{x}$. The solution is apparent: since S and V^T have inverses S^{-1} and V, we can zero out the top component by choosing:

$$\boxed{\vec{x} = VS^{-1}U_1^T\vec{y}.} \tag{6}$$

There is nothing we can do about the bottom component $U_2^T\vec{y}$, since $\vec{x}$ does not appear there. Therefore, (6) will minimize the norm of (5).

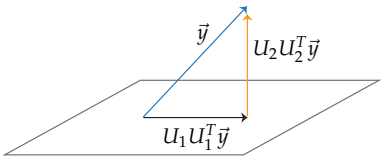
The solution (6) is identical to the familiar Least Squares formula

$$\vec{x} = (A^TA)^{-1}A^T\vec{y}. \tag{7}$$

You can verify this equivalence by substituting (3) in (7), which should give (6) after a little algebra.

The advantage of (6) is the transparency with which we obtained it and the geometric insight it gives. When we substitute $\vec{y} = UU^T\vec{y}$ in (4) we implicitly split $\vec{y}$ into two components:

$$\vec{y} = UU^T\vec{y} = \begin{bmatrix} U_1 & U_2 \end{bmatrix}\begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix}\vec{y} = U_1U_1^T\vec{y} + U_2U_2^T\vec{y}.$$



The first component, $U_1U_1^T\vec{y}$, is the projection of $\vec{y}$ onto the column space of A. This is because the columns of $U_1 = [\vec{u}_1 \ \cdots \ \vec{u}_r]$ form an orthonormal basis for the column space of A. The second component, $U_2U_2^T\vec{y}$, is the remaining part of $\vec{y}$ that is orthogonal to the column space. The Least Squares solution (6) simply matches $A\vec{x}$ to the first component, $U_1U_1^T\vec{y}$, which lies within the column space of A.

Minimum Norm Solution

Above we studied an "overdetermined" problem with more equations than unknowns. We now consider the "underdetermined" equation

$$\vec{y} = A\vec{x} \tag{8}$$

where $\vec{y} \in \mathbb{R}^m$ has smaller dimension than $\vec{x} \in \mathbb{R}^n$; that is $A \in \mathbb{R}^{m \times n}$ is a wide matrix. We assume it has linearly independent rows ($r = m < n$), which means there are infinitely many solutions for $\vec{x}$.

With so many choices for $\vec{x}$ we may want to pick the one with the smallest length. To do so we substitute the SVD

$$A = U \underbrace{\begin{bmatrix} S & 0_{m \times (n-m)} \end{bmatrix}}_{= \Sigma} V^T \tag{9}$$

and write

$$\vec{y} = A\vec{x} = U \begin{bmatrix} S & 0 \end{bmatrix} V^T \vec{x} = U \begin{bmatrix} S & 0 \end{bmatrix} \begin{bmatrix} V_1^T \vec{x} \\ V_2^T \vec{x} \end{bmatrix} = U S V_1^T \vec{x}.$$

Since S and U have inverses S^{-1} and U^T, it follows that

$$V_1^T \vec{x} = S^{-1} U^T \vec{y}. \tag{10}$$

Any $\vec{x}$ satisfying (10) solves (8), but which solution has the least norm? Recall that multiplication with V^T does not change the norm, so

$$\|\vec{x}\| = \|V^T \vec{x}\| = \left\| \begin{bmatrix} V_1^T \vec{x} \\ V_2^T \vec{x} \end{bmatrix} \right\|$$

The first component, $V_1^T \vec{x}$, is fixed by (10). The second component, $V_2^T \vec{x}$, is free and we set it to zero so the norm above is minimized. Thus, the minimum norm solution for $\vec{x}$ is given by

$$V^T \vec{x} = \begin{bmatrix} V_1^T \vec{x} \\ V_2^T \vec{x} \end{bmatrix} = \begin{bmatrix} S^{-1} U^T \vec{y} \\ 0 \end{bmatrix}. \tag{11}$$

Since the inverse of V^T is $V = [V_1 \; V_2]$, (11) implies

$$\vec{x} = [V_1 \; V_2] \begin{bmatrix} S^{-1} U^T \vec{y} \\ 0 \end{bmatrix} \tag{12}$$

or, equivalently,

$$\boxed{\vec{x} = V_1 S^{-1} U^T \vec{y}.} \tag{13}$$

Note from the zero entry in (12) that the minimum-norm solution leaves no component in the column space of V_2, which is the null

space of A as discussed on page 1. Indeed a nonzero component in the null space would not change $A\vec{x}$ but increase the norm of $\vec{x}$.

As an exercise you can show[2] that (13) is equivalent to the formula:

$$\vec{x} = A^T(AA^T)^{-1}\vec{y}. \tag{14}$$

[2] Substitute (9) in (14) and simplify to get (13).

Principal Component Analysis (PCA)

PCA is an application of SVD in statistics that aims to find the most informative directions in a data set.

Suppose the $m \times n$ matrix A contains n measurements from m samples, for example n test scores for m students. If we subtract from each measurement the average over all samples, then each column of A is an m-vector with zero mean, and the $n \times n$ matrix

$$\frac{1}{m-1}A^TA$$

constitutes what is called the "covariance matrix" in statistics. Recall that the eigenvalues of this matrix are the singular values of A except for the scaling factor $m-1$, and its orthonormal eigenvectors correspond to $\vec{v}_1, \ldots, \vec{v}_n$ in the SVD of A.

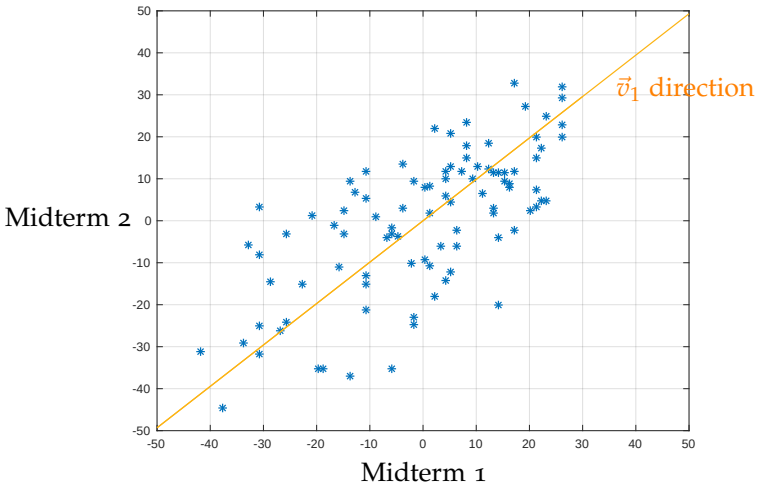
The vectors $\vec{v}_1, \vec{v}_2, \ldots$ corresponding to large singular values are called principal components and identify dominant directions in the data set along which the samples are clustered. The most significant direction is $\vec{v}_1$ corresponding to σ_1.

As an illustration, the scatter plot below shows $n = 2$ midterm scores in a class of $m = 94$ students that I taught in the past. The data points are centered around zero because the class average is subtracted from the test scores. Each data point corresponds to a student and those in the first quadrant (both midterms ≥ 0) are those students who scored above average in each midterm. You can see that there were students who scored below average in the first and above average in the second, and vice versa.

For this data set the covariance matrix is:

$$\frac{1}{93}A^TA = \begin{bmatrix} 297.69 & 202.53 \\ 202.53 & 292.07 \end{bmatrix}$$

where the diagonal entries correspond to the squares of the standard deviations 17.25 and 17.09 for Midterms 1 and 2, respectively. The positive sign of the $(1,2)$ entry implies a positive correlation between the two midterm scores as one would expect.



The eigenvalues of A^TA, that is the singular values of A are $\sigma_1 = 215.08$, $\sigma_2 = 92.66$, and the corresponding eigenvectors of A^TA are:

$$\vec{v}_1 = \begin{bmatrix} 0.7120 \\ 0.7022 \end{bmatrix} \quad \vec{v}_2 = \begin{bmatrix} -0.7022 \\ 0.7120 \end{bmatrix}.$$

The principal component $\vec{v}_1$ is superimposed on the scatter plot and we see that the data is indeed clustered around this line. Note that it makes an angle of $\tan^{-1}(0.7022/0.7120) \approx 44.6^\circ$ which is skewed slightly towards the Midterm 1 axis because the standard deviation in Midterm 1 was slightly higher than in Midterm 2. We may interpret the points above this line as students who performed better in Midterm 2 than in Midterm 1, as measured by their scores relative to the class average that are then compared against the factor $\tan(44.6^\circ)$ to account for the difference in standard deviations.

The $\vec{v}_2$ direction, which is perpendicular to $\vec{v}_1$, exhibits less variation than the $\vec{v}_1$ direction ($\sigma_2 = 92.66$ vs. $\sigma_1 = 215.08$).

EE16B - Spring'20 - Lecture 12A Notes[1]

Murat Arcak

7 April 2020

Case Study: Minimum Energy Control

In this example we review discretization, controllability, and minimum norm solutions. Consider the model of a car moving in a lane

$$\begin{aligned}\frac{dp(t)}{dt} &= v(t)\\ \frac{dv(t)}{dt} &= \frac{1}{RM}u(t)\end{aligned}$$

where $p(t)$ is position, $v(t)$ is velocity, $u(t)$ is wheel torque, R is wheel radius, and M is mass. This model is similar to an example discussed in Lecture 7A, but here we ignore friction for simplicity.

First we discretize this continuous-time model. If we apply the constant input $u(t) = u_d(k)$ from $t = kT$ to $(k+1)T$, then by integration

$$\begin{aligned}v(t) &= v(kT) + (t-kT)\frac{1}{RM}u_d(k)\\ p(t) &= p(kT) + (t-kT)v(kT) + \frac{1}{2}(t-kT)^2\frac{1}{RM}u_d(k)\end{aligned}$$

for $t \in [kT, (k+1)T)$. In particular, at $t = (k+1)T$:

$$\begin{aligned}p((k+1)T) &= p(kT) + Tv(kT) + \frac{T^2}{2RM}u_d(k)\\ v((k+1)T) &= v(kT) + \frac{T}{RM}u_d(k).\end{aligned}$$

Putting these equations in matrix/vector form and substituting $p_d(k) = p(kT)$, $v_d(k) = v(kT)$, we get

$$\begin{bmatrix}p_d(k+1)\\ v_d(k+1)\end{bmatrix} = \underbrace{\begin{bmatrix}1 & T\\ 0 & 1\end{bmatrix}}_{A}\begin{bmatrix}p_d(k)\\ v_d(k)\end{bmatrix} + \underbrace{\frac{1}{RM}\begin{bmatrix}\frac{1}{2}T^2\\ T\end{bmatrix}}_{\vec{b}}u_d(k). \tag{1}$$

Now suppose the vehicle is at rest with $p(0) = v(0) = 0$ at $t = 0$ and the goal is to reach a target position p_{target} and stop there ($v_{\text{target}} = 0$). Recall from the lectures on controllability that if we can find a sequence $u_d(0), u_d(1), \cdots, u_d(\ell-1)$ such that

$$\begin{bmatrix}p_{\text{target}}\\ 0\end{bmatrix} = \underbrace{\begin{bmatrix}\vec{b} & A\vec{b} & \cdots A^{\ell-1}\vec{b}\end{bmatrix}}_{C_\ell}\begin{bmatrix}u_d(\ell-1)\\ u_d(\ell-2)\\ \vdots\\ u_d(0)\end{bmatrix} \tag{2}$$

then we reach the desired state in ℓ time steps, that is at time $t = \ell T$.

Since we have $n = 2$ state variables the controllability test we learned checks whether C_ℓ with $\ell = 2$ spans $\mathbb{R}^2$. This is indeed the case, since

$$C_2 = \begin{bmatrix} \vec{b} & A\vec{b} \end{bmatrix} = \frac{1}{RM} \begin{bmatrix} \frac{1}{2}T^2 & \frac{3}{2}T^2 \\ T & T \end{bmatrix}$$

has linearly independent columns.

Although this test also suggests we can reach the target state in two steps, the resulting values of $u_d(0)$ and $u_d(1)$ will likely exceed physical limits. For example, if we take the values[2] $RM = 5000$ kg m, $T = 0.1$ s, $p_{\text{target}} = 1000$ m, then

$$\begin{bmatrix} u_d(1) \\ u_d(0) \end{bmatrix} = C_2^{-1} \begin{bmatrix} p_{\text{target}} \\ 0 \end{bmatrix} = \begin{bmatrix} -5 \cdot 10^8 \\ 5 \cdot 10^8 \end{bmatrix} \text{kg m}^2/\text{s}^2,$$

which exceeds the torque and braking limits of a typical car by 5 orders of magnitude.[3]

[2] say, for a sedan with mass $M \approx 1700$ kg and wheel radius $R \approx 0.3m$

[3] If our car could deliver the torque $u_d(0) = 5 \cdot 10^8$ kg m^2/s^2, then from (1) we would reach $v_d(1) = v(T) = 10^4$ m/s (22,369 mph) in $T = 0.1$ seconds!

Therefore, in practice we need to select a sufficiently large number of time steps ℓ. This leads to a wide controllability matrix C_ℓ and allows for infinitely many input sequences that satisfy (2). Among them we can select the minimum norm solution so we spend the least control energy. Using the minimum-norm solution formula

$$\begin{bmatrix} u_d(\ell-1) \\ u_d(\ell-2) \\ \vdots \\ u_d(0) \end{bmatrix} = C_\ell^T (C_\ell C_\ell^T)^{-1} \begin{bmatrix} p_{\text{target}} \\ 0 \end{bmatrix}$$

and quite a bit of algebra, one will obtain the input sequence

$$u_d(k) = \frac{6RM(\ell - 1 - 2k)}{T^2 \ell(\ell^2 - 1)} p_{\text{target}}, \quad k = 0, \cdots, \ell - 1.$$

In the plot below we show this input sequence, as well as the resulting velocity and position profiles for $RM = 5000$ kg m, $p_{\text{target}} = 1000$ m, $T = 0.1$ s, and $\ell = 1200$. With these parameters we allow $\ell T = 120$ s (2 minutes) to travel 1 km. Note that the vehicle accelerates in the first half of this period and decelerates in the second half, reaching the maximum velocity 12.5 m/s ($\approx$ 28 mph) in the middle. The acceleration and deceleration are hardest at the very beginning and at the very end, respectively. The corresponding torque is within a physically reasonable range, $[-2000, 2000]$ Nm.

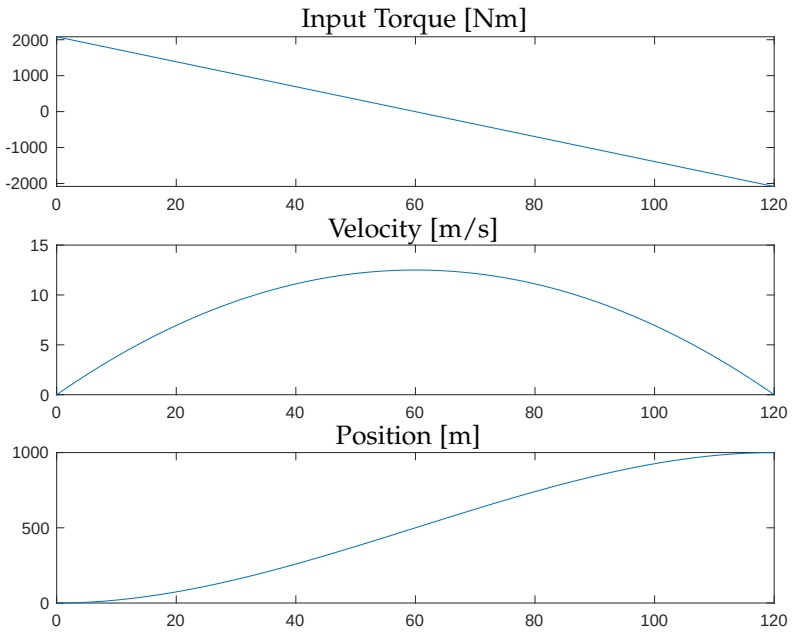


Figure 1: The minimum norm input torque sequence, and the resulting velocity and position profiles for $RM = 5000$ kg m, $p_{\text{target}} = 1000$ m, $T = 0.1$ s, and $\ell = 1200$. The horizontal axis is time, which ranges from 0 to $\ell T = 120$ s (2 minutes). The vehicle accelerates in the first half of this period and decelerates in the second half, reaching the maximum velocity 12.5 m/s (≈ 28 mph) in the middle.

Stability of Linear State Models

The Scalar Case

We first study a system with a single state variable $x(t)$ that obeys

$$x(t+1) = \lambda x(t) + bu(t) \tag{3}$$

where λ and b are constants. If we start with the initial condition $x(0)$, then we get by recursion

$$\begin{aligned}
x(1) &= \lambda x(0) + bu(0) \\
x(2) &= \lambda x(1) + bu(1) = \lambda^2 x(0) + \lambda bu(0) + bu(1) \\
x(3) &= \lambda x(2) + bu(2) = \lambda^3 x(0) + \lambda^2 bu(0) + \lambda bu(1) + bu(2) \\
&\vdots \\
x(t) &= \lambda^t x(0) + \lambda^{t-1} bu(0) + \lambda^{t-2} bu(1) + \cdots + \lambda bu(t-2) + bu(t-1),
\end{aligned}$$

rewritten compactly as:

$$\boxed{x(t) = \lambda^t x(0) + \sum_{k=0}^{t-1} \lambda^{t-1-k} bu(k) \quad t = 1, 2, 3, \ldots} \tag{4}$$

The first term $\lambda^t x(0)$ represents the effect of the initial condition and the second term $\sum_{k=0}^{t-1} \lambda^{t-1-k} bu(k)$ represents the effect of the input sequence $u(0), u(1), \ldots, u(t-1)$.

Definition. We say that a system is *stable* if its state $x(t)$ remains bounded for any initial condition and any bounded input sequence. Conversely, we say it is *unstable* if we can find an initial condition and a bounded input sequence such that $|x(t)| \to \infty$ as $t \to \infty$.

It follows from (4) that, if $|\lambda| > 1$, then a nonzero initial condition $x(0) \neq 0$ is enough to drive $|x(t)|$ unbounded. This is because $|\lambda|^t$ grows unbounded and, with $u(t) = 0$ for all t, we get $|x(t)| = |\lambda^t x(0)| = |\lambda|^t |x(0)| \to \infty$. Thus, (3) is unstable for $|\lambda| > 1$.

Next, we show that $|\lambda| < 1$ guarantees stability. In this case $\lambda^t x(0)$ decays to zero, so we need only to show that the second term in (4) remains bounded for any bounded input sequence. A bounded input means we can find a constant M such that $|u(t)| \leq M$ for all t. Thus,

$$\left| \sum_{k=0}^{t-1} \lambda^{t-1-k} b u(k) \right| \leq \sum_{k=0}^{t-1} |\lambda|^{t-1-k} |b| |u(k)| \leq |b| M \sum_{k=0}^{t-1} |\lambda|^{t-1-k}.$$

Defining the new index $s = t - 1 - k$ we rewrite the last expression as

$$|b| M \sum_{s=0}^{t-1} |\lambda|^s,$$

and note that $\sum_{s=0}^{t-1} |\lambda|^s$ is a geometric series that converges to $\frac{1}{1-|\lambda|}$ since $|\lambda| < 1$. Therefore, each term in (4) is bounded and we conclude stability for $|\lambda| < 1$.

Summary: The scalar system (3) is stable when $|\lambda| < 1$, and unstable when $|\lambda| > 1$.

When λ is a complex number, a perusal of the stability and instability arguments above show that the same conclusions hold if we interpret $|a|$ as the modulus of a, that is:

$$|\lambda| = \sqrt{\text{Re}\{\lambda\}^2 + \text{Im}\{\lambda\}^2}.$$

What happens when $|\lambda| = 1$? If we disallow inputs ($b = 0$), this case is referred to as "marginal stability" because $|\lambda^t x(0)| = |x(0)|$, which neither grows nor decays. If we allow inputs ($b \neq 0$), however, we can find a bounded input to drive the second term in (4) unbounded. For example, when $\lambda = 1$, the constant input $u(t) = 1$ yields:

$$\sum_{k=0}^{t-1} \lambda^{t-1-k} b u(k) = \sum_{k=0}^{t-1} b = bt$$

which grows unbounded as $t \to \infty$. Therefore, $|\lambda| = 1$ is a precarious case that must be avoided in designing systems.

The Vector Case

When $\vec{x}(t)$ is an n-dimensional vector governed by

$$\vec{x}(t+1) = A\vec{x}(t) + Bu(t), \tag{5}$$

recursive calculations lead to the solution

$$\boxed{\vec{x}(t) = A^t\vec{x}(0) + \sum_{k=0}^{t-1} A^{t-1-k}Bu(k) \quad t = 1,2,3,\ldots} \tag{6}$$

where the matrix power is defined as $A^t = \underbrace{A \cdots A}_{t \text{ times}}$.

Since A is no longer a scalar, stability properties are not apparent from (6). However, when A is diagonalizable we can employ the change of variables $\vec{z} := T\vec{x}$ and select the matrix T such that

$$A_{\text{new}} = TAT^{-1}$$

is diagonal. A and A_{new} have the same eigenvalues and, since A_{new} is diagonal, the eigenvalues appear as its diagonal entries:

$$A_{\text{new}} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}.$$

The state model for the new variables is

$$\vec{z}(t+1) = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \vec{z}(t) + B_{\text{new}}u(t), \quad B_{\text{new}} = TB, \tag{7}$$

which nicely decouples into scalar equations:

$$z_i(t+1) = \lambda_i z_i(t) + b_i u(t), \quad i = 1,\ldots,n \tag{8}$$

where we denote by b_i the i-th entry of B_{new}. Then, the results for the scalar case above imply stability when $|\lambda_i| < 1$ and instability when $|\lambda_i| > 1$.

For the whole system to be stable each subsystem must be stable, therefore we need $|\lambda_i| < 1$ for each $i = 1,\ldots,n$. If there exists at least one eigenvalue λ_i with $|\lambda_i| > 1$ then we conclude instability because we can drive the corresponding state $z_i(t)$ unbounded.

Summary: The discrete-time system (5) is stable if $|\lambda_i| < 1$ for each eigenvalue $\lambda_1,\ldots,\lambda_n$ of A, and unstable if $|\lambda_i| > 1$ for some eigenvalue λ_i.

EE16B - Spring'20 - Lecture 12B Notes[1]

Murat Arcak

9 April 2020

Stability of Linear State Models

Recall that a system with a single state variable $x(t)$ satisfying

$$x(t+1) = \lambda x(t) + bu(t) \tag{1}$$

is stable when $|\lambda| < 1$, and unstable when $|\lambda| > 1$. The system

$$\vec{x}(t+1) = A\vec{x}(t) + Bu(t), \tag{2}$$

$\vec{x}(t)$ is an n-dimensional vector, is stable if $|\lambda_i| < 1$ for each eigenvalue $\lambda_1, \ldots, \lambda_n$ of A, and unstable if $|\lambda_i| > 1$ for some eigenvalue.

Last time we justified this eigenvalue test for the case when A is diagonalizable. In particular we used the change of variables $\vec{z} := T\vec{x}$, where T is such that

$$A_{\text{new}} = TAT^{-1}$$

is diagonal[2]. A and A_{new} have the same eigenvalues and, since A_{new} is diagonal, the eigenvalues appear as its diagonal entries:

$$A_{\text{new}} = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}.$$

[2] Choose $T = V^{-1}$, where the columns of V are the linearly independent eigenvectors of A. Then $TAT^{-1} = V^{-1}AV$, which is diagonal as we saw early in the semester.

The state model for the new variables is

$$\vec{z}(t+1) = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} \vec{z}(t) + B_{\text{new}}u(t), \quad B_{\text{new}} = TB, \tag{3}$$

which nicely decouples into scalar equations:

$$z_i(t+1) = \lambda_i z_i(t) + b_i u(t), \quad i = 1, \ldots, n \tag{4}$$

where b_i is the i-th entry of B_{new}. Then, from the scalar result, we conclude stability when $|\lambda_i| < 1$ and instability when $|\lambda_i| > 1$.

For the whole system to be stable each subsystem must be stable, therefore we need $|\lambda_i| < 1$ for each $i = 1, \ldots, n$. If there exists at least one eigenvalue λ_i with $|\lambda_i| > 1$ then we conclude instability because we can drive the corresponding state $z_i(t)$ unbounded.

The same stability condition (all eigenvalues must satisfy $|\lambda_i| < 1$) holds when A is not diagonalizable. In that case a transformation

we will discuss later brings A_{new} to an upper-triangular form with eigenvalues on the diagonal. Thus, instead of (3) we have

$$\vec{z}(t+1) = \begin{bmatrix} \lambda_1 & \star & \cdots & \star \\ & \ddots & \ddots & \vdots \\ & & \ddots & \star \\ & & & \lambda_n \end{bmatrix} \vec{z}(t) + B_{\text{new}} u(t) \tag{5}$$

where the entries marked with '$\star$' may be nonzero, but we don't need their explicit values for the argument that follows. Then it is not difficult to see that z_n obeys

$$z_n(t+1) = \lambda_n z_n(t) + b_n u(t) \tag{6}$$

which does not depend on other states, so we conclude $z_n(t)$ remains bounded for bounded inputs when $|\lambda_n| < 1$. The equation for z_{n-1} has the form

$$z_{n-1}(t+1) = \lambda_{n-1} z_{n-1}(t) + [\star\, z_n(t) + b_{n-1} u(t)] \tag{7}$$

where we can treat the last two terms in brackets as a bounded input since we have already shown that $z_n(t)$ is bounded. If $|\lambda_{n-1}| < 1$ we conclude $z_{n-1}(t)$ is itself bounded and proceed to the equation:

$$z_{n-2}(t+1) = \lambda_{n-2} z_{n-2}(t) + [\star\, z_{n-1}(t) + \star\, z_n(t) + b_{n-2} u(t)]. \tag{8}$$

Continuing this argument recursively we conclude stability when $|\lambda_i| < 1$ for each eigenvalue λ_i.

To conclude instability when $|\lambda_i| > 1$ for some eigenvalue, note that the ordering of the eigenvalues in (5) is arbitrary: we can put them in any order we want by properly selecting T. Therefore, we can assume without loss of generality that an eigenvalue with $|\lambda_i| > 1$ appears in the nth diagonal entry, that is $|\lambda_n| > 1$. Then, instability follows from the scalar equation (6).

Stability of Continuous-Time Linear Systems

The solution of the scalar continuous-time system

$$\frac{d}{dt} x(t) = \lambda x(t) + b u(t) \tag{9}$$

is given by

$$x(t) = e^{\lambda t} x(0) + b \int_0^t e^{\lambda (t-s)} u(s) ds. \tag{10}$$

It follows that this system is stable when $\lambda < 0$ (in which case $e^{\lambda t} \to 0$ as $t \to \infty$) and unstable when $\lambda > 0$ (in which case $e^{\lambda t} \to \infty$).

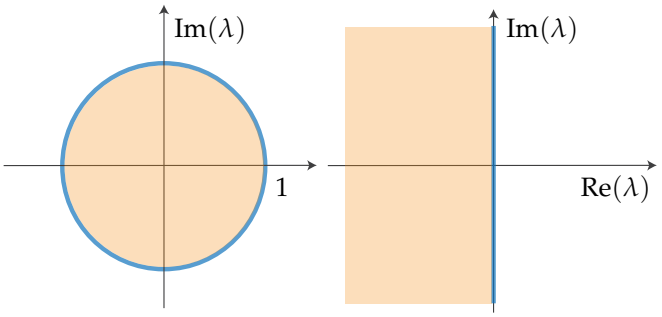
If λ is complex then we check $\text{Re}(\lambda) < 0$ for the real part. This is because, if we decompose λ into its real and imaginary parts, $\lambda = \alpha + j\omega$, then $e^{\lambda t} = e^{\alpha t}e^{j\omega t}$ where $|e^{j\omega t}| = 1$ and, thus, $e^{\lambda t} \to 0$ if $\alpha = \text{Re}(\lambda) < 0$.

Using reasoning similar to the discrete-time case, we conclude that the vector continuous-time system

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + B\vec{u}(t) \tag{11}$$

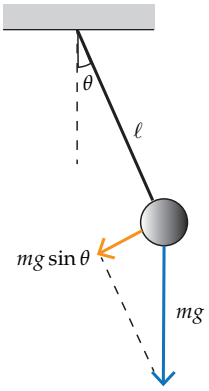
is stable if $\text{Re}(\lambda_i) < 0$ for each eigenvalue $\lambda_1, \ldots, \lambda_n$ of A, and unstable if $\text{Re}(\lambda_i) > 0$ for some eigenvalue λ_i.

The figures below highlight the regions of the complex plane where the eigenvalues must lie for stability of a discrete-time (left) and continuous-time (right) system.



Example 1: In Lecture 6A we modeled the motion of the pendulum depicted on the right as

$$\begin{aligned} \frac{d}{dt}x_1(t) &= x_2(t) \\ \frac{d}{dt}x_2(t) &= -\frac{k}{m}x_2(t) - \frac{g}{\ell}\sin x_1(t), \end{aligned} \tag{12}$$



where the state variables are the angle and angular velocity:

$$x_1(t) := \theta(t) \qquad x_2(t) := \frac{d\theta(t)}{dt}.$$

The two distinct equilibrium points are the downward position:

$$x_1 = 0, \quad x_2 = 0, \tag{13}$$

and the upright position:

$$x_1 = \pi, \quad x_2 = 0. \tag{14}$$

Since the entries of $f(\vec{x})$ are $f_1(\vec{x}) = x_2$ and $f_2(\vec{x}) = -\frac{k}{m}x_2 - \frac{g}{\ell}\sin x_1$, we have

$$\nabla f(\vec{x}) = \begin{bmatrix} \frac{\partial f_1(x_1,x_2)}{\partial x_1} & \frac{\partial f_1(x_1,x_2)}{\partial x_2} \\ \frac{\partial f_2(x_1,x_2)}{\partial x_1} & \frac{\partial f_2(x_1,x_2)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{\ell}\cos x_1 & \frac{-k}{m} \end{bmatrix}.$$

By evaluating this matrix at (13) and (14), we obtain the linearization around the respective equilibrium point:

$$A_{\text{down}} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{\ell} & \frac{-k}{m} \end{bmatrix} \qquad A_{\text{up}} = \begin{bmatrix} 0 & 1 \\ \frac{g}{\ell} & \frac{-k}{m} \end{bmatrix}. \tag{15}$$

The eigenvalues of A_{down} are the roots of $\lambda^2 + \frac{k}{m}\lambda + \frac{g}{\ell}$, which can be shown to have strictly negative real parts when $k > 0$. Thus the downward position is stable.

The eigenvalues of A_{up} are the roots of $\lambda^2 + \frac{k}{m}\lambda - \frac{g}{\ell}$, which are given by:

$$\lambda_1 = -\frac{k}{2m} - \frac{1}{2}\sqrt{\left(\frac{k}{m}\right)^2 + 4\frac{g}{\ell}} \quad \lambda_2 = -\frac{k}{2m} + \frac{1}{2}\sqrt{\left(\frac{k}{m}\right)^2 + 4\frac{g}{\ell}}.$$

Since $\lambda_2 > 0$, the upright position in unstable. Note that making the length ℓ smaller increases the value of λ_2. This suggests that a smaller length aggravates the instability of the upright position and makes the stabilization task more difficult, as you would experience when you try to balance a stick in your hand.

Predicting Transient Behavior from Eigenvalue Locations

We have seen that the solutions of a discrete-time system are composed of λ_i^t terms where λ_i's are the eigenvalues of A. Thus, to predict the nature of the solutions (damped, underdamped, unbounded, etc.), it is important to visualize the sequence λ^t, $t = 1, 2, \ldots$ for a given λ. If we rewrite λ as $\lambda = |\lambda| e^{j\omega}$ where $|\lambda|$ is the distance to the origin in the complex plane, then we get

$$\lambda^t = |\lambda|^t e^{j\omega t} = |\lambda|^t \cos(\omega t) + j\,|\lambda|^t \sin(\omega t),$$

the real part of which is depicted in Figure 1 for various values of λ. Note that the envelope $|\lambda|^t$ decays to zero when λ is inside the unit disk ($|\lambda| < 1$) and grows unbounded when it is outside ($|\lambda| > 1$), which is consistent with our stability criterion.

Likewise, for a continuous-time system each eigenvalue λ_i contributes a function of the form $e^{\lambda_i t}$ to the solution. Decomposing λ into its real and imaginary parts, $\lambda = \alpha + j\omega$, we get

$$e^{\lambda t} = e^{\alpha t} e^{j\omega t} = e^{\alpha t} \cos(\omega t) + j\, e^{\alpha t} \sin(\omega t).$$

Figure 2 depicts the real part of $e^{\lambda t}$ for various values of λ. Note that the envelope $e^{\alpha t}$ decays when $\alpha = \text{Re}(\lambda) < 0$ as in our stability condition.

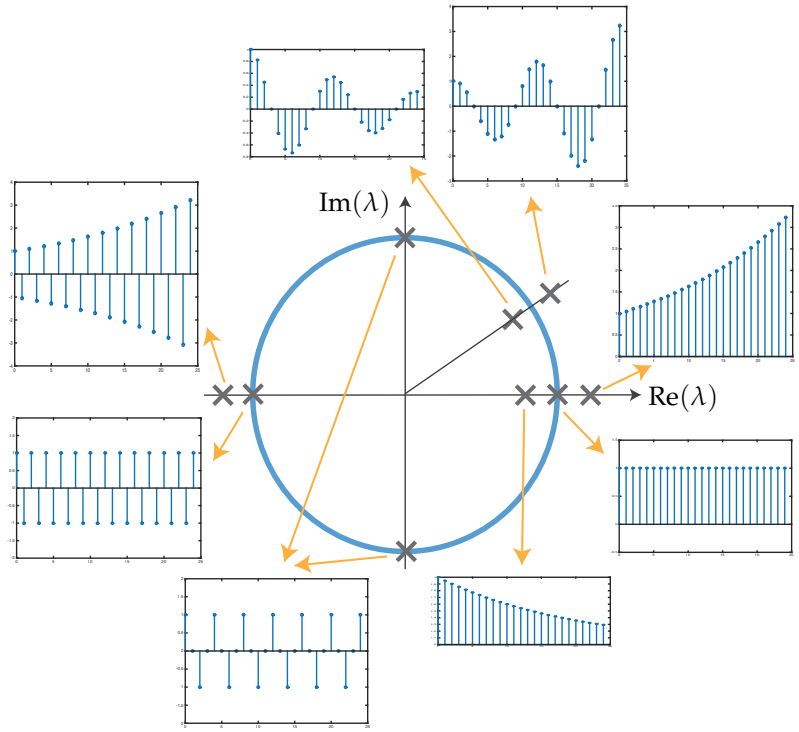


Figure 1: The real part of λ^t for various values of λ in the complex plane. It grows unbounded when $|\lambda| > 1$, decays to zero when $|\lambda| < 1$, and has constant amplitude when λ is on the unit circle ($|\lambda| = 1$).

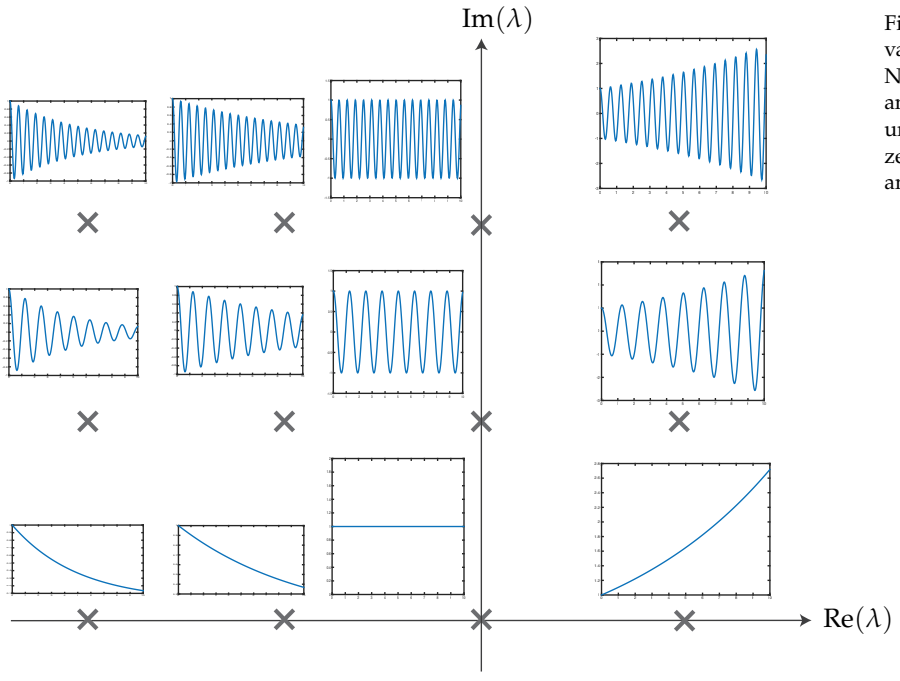


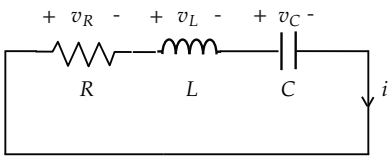
Figure 2: The real part of $e^{\lambda t}$ for various values of λ in the complex plane. Note that $e^{\lambda t}$ is oscillatory when λ has an imaginary component. It grows unbounded when $\text{Re}\{\lambda\} > 0$, decays to zero when $\text{Re}\{\lambda\} < 0$, and has constant amplitude when $\text{Re}\{\lambda\} = 0$.

Example 2: Recall that the RLC circuit depicted on the right can be modeled as

$$\frac{dx_1(t)}{dt} = \frac{1}{C}x_2(t)$$
$$\frac{dx_2(t)}{dt} = \frac{1}{L}\left(-x_1(t) - Rx_2(t)\right)$$

where $x_1 = v_C$ and $x_2 = i$. Since this model is linear we can rewrite it in the form

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) \quad \text{where} \quad A = \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L} & -\frac{R}{L} \end{bmatrix}.$$



Then the roots of

$$\det(\lambda I - A) = \lambda^2 + \frac{R}{L}\lambda + \frac{1}{LC}$$

give the eigenvalues:

$$\lambda_{1,2} = -\alpha \mp \sqrt{\alpha^2 - \omega_0^2} \quad \text{where} \quad \alpha := \frac{R}{2L},\ \omega_0 := \frac{1}{\sqrt{LC}}.$$

For $\alpha > \omega_0$ we have two real, negative eigenvalues which indicate a damped response. For $\alpha < \omega_0$, we get the complex eigenvalues

$$\lambda_{1,2} = -\alpha \mp j\omega \quad \text{where} \quad \omega = \sqrt{\omega_0^2 - \alpha^2},$$

indicating oscillations with frequency ω and decaying envelope $e^{-\alpha t}$.

EE16B - Spring'20 - Lecture 13A Notes[1]

Murat Arcak

14 April 2020

State Feedback Control

Suppose we are given a single-input control system

$$\vec{x}(t+1) = A\vec{x}(t) + Bu(t), \quad \vec{x}(t) \in \mathbb{R}^n,\ u(t) \in \mathbb{R}, \tag{1}$$

and we wish to bring $\vec{x}(t)$ to the equilibrium $\vec{x} = 0$ from any initial condition $\vec{x}(0)$. To do this we will use the "control policy"

$$u(t) = k_1 x_1(t) + k_2 x_2(t) + \cdots + k_n x_n(t) \tag{2}$$

where $k_1, k_2, \ldots, k_n$ are to be determined. Rewriting (2) as

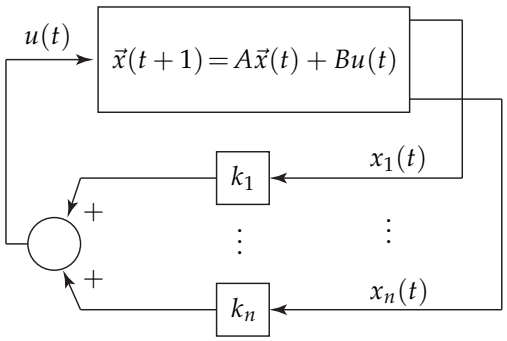
$$\boxed{u(t) = K\vec{x}(t)} \tag{3}$$

with row vector $K = [k_1\, k_2\, \cdots\, k_n]$, and substituting in (1), we get

$$\boxed{\vec{x}(t+1) = (A + BK)\vec{x}(t).} \tag{4}$$

Thus, if we can choose K such that all eigenvalues of $A + BK$ are inside the unit circle, $|\lambda_i(A + BK)| < 1$, $i = 1, \ldots, n$, then $\vec{x}(t) \to 0$ for any $\vec{x}(0)$ from our stability discussions in the previous lectures.

We will see that if the system (1) is controllable, then we can arbitrarily assign the eigenvalues of $A + BK$ by appropriately chosing K. Thus, in addition to bringing the eigenvalues inside the unit circle for stability, we can place them in favorable locations to shape the transients, *e.g.*, to achieve a well damped convergence.



We refer to (4) as the "closed-loop" system since the control policy (2) generates a feedback loop as depicted in the block diagram. The state variables are measured at every time step t and the input $u(t)$ is synthesized as a linear combination of these measurements.

Comparison to Open Loop Control

Recall that controllability allowed us to calculate an input sequence $u(0), u(1), u(2), \ldots$ that drives the state from $\vec{x}(0)$ to any $\vec{x}_{\text{target}}$. Thus, an alternative to the feedback control (2) is to select $\vec{x}_{\text{target}} = 0$, calculate an input sequence based on $\vec{x}(0)$, and to apply this sequence in an "open-loop" fashion without using further state measurements as depicted below.

$$u(0), u(1), u(2), \ldots \longrightarrow \boxed{\vec{x}(t+1) = A\vec{x}(t) + Bu(t)}$$

The trouble with this open-loop approach is that it is sensitive to uncertainties in A and B, and does not make provisions against disturbances that may act on the system.

By contrast, feedback offers a degree of robustness: if our design of K brings the eigenvalues of $A + BK$ to well within the unit circle, then small perturbations in A and B would not move these eigenvalues outside the circle. Thus, despite the uncertainty, solutions converge to $\vec{x} = 0$ in the absence of disturbances and remain bounded in the presence of bounded disturbances.

Eigenvalue Assignment by State Feedback: Examples

Example 1: Consider the second order system

$$\vec{x}(t+1) = \underbrace{\begin{bmatrix} 0 & 1 \\ a_1 & a_2 \end{bmatrix}}_{A} \vec{x}(t) + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{B} u(t)$$

and note that the eigenvalues of A are the roots of the polynomial

$$\det(\lambda I - A) = \lambda^2 - a_2\lambda - a_1.$$

If we substitute the control

$$u(t) = K\vec{x}(t) = k_1 x_1(t) + k_2 x_2(t)$$

the closed-loop system becomes

$$\vec{x}(t+1) = \underbrace{\begin{bmatrix} 0 & 1 \\ a_1 + k_1 & a_2 + k_2 \end{bmatrix}}_{A + BK} \vec{x}(t)$$

and, since $A + BK$ has the same structure as A with a_1, a_2 replaced by $a_1 + k_1$, $a_2 + k_2$, the eigenvalues of $A + BK$ are the roots of

$$\lambda^2 - (a_2 + k_2)\lambda - (a_1 + k_1).$$

Now if we want to assign the eigenvalues of $A + BK$ to desired values λ_1 and λ_2, we must match the polynomial above to

$$(\lambda - \lambda_1)(\lambda - \lambda_2) = \lambda^2 - (\lambda_1 + \lambda_2)\lambda + \lambda_1\lambda_2,$$

that is,

$$a_2 + k_2 = \lambda_1 + \lambda_2 \quad \text{and} \quad a_1 + k_1 = -\lambda_1\lambda_2.$$

This is indeed accomplished with the choice $k_1 = -a_1 - \lambda_1\lambda_2$ and $k_2 = -a_2 + \lambda_1 + \lambda_2$, which means that we can assign the closed-loop eigenvalues as we wish.

Example 2: Let's apply the eigenvalue assignment procedure above to

$$\vec{x}(t+1) = \underbrace{\begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}}_{A} \vec{x}(t) + \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{B} u(t).$$

Now we have

$$A + BK = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} k_1 & k_2 \end{bmatrix} = \begin{bmatrix} 1 + k_1 & 1 + k_2 \\ 0 & 2 \end{bmatrix}$$

and, because this matrix is upper triangular, its eigenvalues are the diagonal entries:

$$\lambda_1 = 1 + k_1 \quad \text{and} \quad \lambda_2 = 2.$$

Note that we can move λ_1 with the choice of k_1, but we have no control over λ_2. In fact, since $|\lambda_2| > 1$, the closed-loop system remains unstable no matter what control we apply.

This is a consequence of the uncontrollability[2] of this system: the second state equation

$$x_2(t+1) = 2x_2(t)$$

can't be influenced by $u(t)$, and $x_2(t) = 2^t x_2(0)$ grows exponentially.

[2] Note that B and AB are *not* linearly independent; therefore, the system is uncontrollable.

Continuous-Time State Feedback

The idea of state feedback is identical for a continuous-time system,

$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) + Bu(t), \quad u(t) \in \mathbb{R}.$$

To bring $\vec{x}(t)$ to the equilibrium $\vec{x} = 0$ we apply

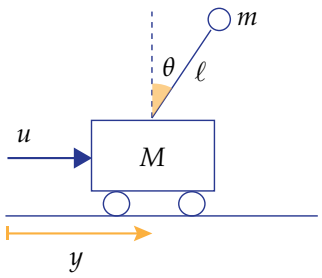
$$u(t) = K\vec{x}(t)$$

and obtain the closed-loop system

$$\frac{d}{dt}\vec{x}(t) = (A + BK)\vec{x}(t).$$

The only difference from discrete-time is the stability criterion: we must choose K such that $\text{Re}(\lambda_i(A + BK)) < 0$ for each eigenvalue λ_i.

Example 3: Consider the inverted pendulum depicted below



and let the state variables be θ: the angle, $\dot{\theta}$: angular velocity, $\dot{y}$: the velocity of the cart. Then the equations of motion are

$$\begin{aligned}
\frac{d\theta}{dt} &= \dot{\theta} \\
\frac{d\dot{\theta}}{dt} &= \frac{1}{\ell(\frac{M}{m} + \sin^2\theta)}\left(-\frac{u}{m}\cos\theta - \dot{\theta}^2\ell\cos\theta\sin\theta + \frac{M+m}{m}g\sin\theta\right) \\
\frac{d\dot{y}}{dt} &= \frac{1}{\frac{M}{m} + \sin^2\theta}\left(\frac{u}{m} + \dot{\theta}^2\ell\sin\theta - g\sin\theta\cos\theta\right)
\end{aligned}$$

and linearization about the upright position $\theta = 0$, $\dot{\theta} = 0$, $\dot{y} = 0$ gives

$$\frac{d}{dt}\begin{bmatrix}\theta(t)\\ \dot{\theta}(t)\\ \dot{y}(t)\end{bmatrix} = \underbrace{\begin{bmatrix}0 & 1 & 0\\ \frac{M+m}{M\ell}g & 0 & 0\\ -\frac{m}{M}g & 0 & 0\end{bmatrix}}_{A}\begin{bmatrix}\theta(t)\\ \dot{\theta}(t)\\ \dot{y}(t)\end{bmatrix} + \underbrace{\begin{bmatrix}0\\ -\frac{1}{M\ell}\\ \frac{1}{M}\end{bmatrix}}_{B}u(t).$$

We have omitted the cart position y from the state variables because we are interested in stabilizing the point $\theta = 0$, $\dot{\theta} = 0$, $\dot{y} = 0$, and we are not concerned about the final value of the position $y(t)$.

We now design a state feedback controller,

$$u(t) = k_1\theta(t) + k_2\dot{\theta}(t) + k_3\dot{y}(t).$$

Substituting the values $M = 1$, $m = 0.1$, $l = 1$, and $g = 10$, we get

$$\underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 11 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}}_{A} + \underbrace{\begin{bmatrix} 0 \\ -1 \\ 1 \end{bmatrix}}_{B} \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 11 - k_1 & -k_2 & -k_3 \\ -1 + k_1 & k_2 & k_3 \end{bmatrix}.$$

The characteristic polynomial of this matrix is

$$\lambda^3 + (k_2 - k_3)\lambda^2 + (k_1 - 11)\lambda + 10k_3 = 0$$

and, as in previous examples, we can choose k_1, k_2, k_3, to match the coefficients of this polynomial to those of

$$(\lambda - \lambda_1)(\lambda - \lambda_2)(\lambda - \lambda_3)$$

where λ_1, λ_2, λ_3 are desired closed-loop eigenvalues.

EE16B - Spring'20 - Lecture 13B Notes[1]

Murat Arcak

16 April 2020

Eigenvalue Assignment with State Feedback

In the previous lecture we studied the system

$$\vec{x}(t+1) = A\vec{x}(t) + Bu(t), \quad \vec{x}(t) \in \mathbb{R}^n,\ u(t) \in \mathbb{R}, \tag{1}$$

with the feedback control policy

$$u(t) = k_1 x_1(t) + k_2 x_2(t) + \cdots + k_n x_n(t), \tag{2}$$

which we rewrote as

$$u(t) = K\vec{x}(t) \tag{3}$$

with $K = [k_1\, k_2\, \cdots\, k_n]$. When we substitute (3) in (1), we get

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

and the task is to choose K such that all eigenvalues of $A + BK$ are inside the unit circle[2] for stability.

Ideally we would like to be able to assign the eigenvalues as we wish, so we can influence the transients, *e.g.*, for faster convergence[3]. We claimed last time that the controllability of the system (1) gives us this ability: given a set of desired eigenvalues $\lambda_1, \cdots, \lambda_n$ we can find a corresponding K such that $A + BK$ has those eigenvalues.

In this lecture we review some examples of designing K. We then outline a proof of the claim that controllability gives us the ability to assign the eigenvalues arbitrarily.

[2] For continuous-time systems the eigenvalues of $A + BK$ must have negative real parts for stability.

[3] See Figures 1-2 in Lecture 12B to see how eigenvalue locations affect the transients.

Example 1 (Cruise Control): In Lecture 7A we studied the nonlinear model of a vehicle moving in a lane

$$M\frac{d}{dt}v(t) = -\frac{1}{2}\rho a c\, v(t)^2 + \frac{1}{R}u(t) \tag{5}$$

where $v(t)$ is velocity, $u(t)$ is the wheel torque, M is vehicle mass, ρ is air density, a is vehicle area, c is drag coefficient, and R is wheel radius. To maintain $v(t)$ at a desired value v^* we apply the torque

$$u^* = \frac{R}{2}\rho a c\, v^{*2},$$

which counterbalances the drag force at that velocity. We rewrite the model (5) as $\frac{d}{dt}v(t) = f(v(t), u(t))$, where

$$f(v, u) = -\frac{1}{2M}\rho a c\, v^2 + \frac{1}{RM}u.$$

Then the linearized dynamics for the perturbation $\tilde{v}(t) = v(t) - v^*$ is

$$\frac{d}{dt}\tilde{v}(t) = \lambda\tilde{v}(t) + b\tilde{u}(t), \tag{6}$$

where $\tilde{u}(t) = u(t) - u^*$,

$$\lambda = \left.\frac{\partial f(v,u)}{\partial v}\right|_{v^*,u^*} = -\frac{1}{M}\rho acv^*, \quad b = \left.\frac{\partial f(v,u)}{\partial u}\right|_{v^*,u^*} = \frac{1}{RM}.$$

If we apply $u(t) = u^*$, that is $\tilde{u}(t) = 0$, then the solution of (6) is

$$\tilde{v}(t) = \tilde{v}(0)e^{\lambda t},$$

which converges to 0 since $\lambda < 0$. This means that if $v(t)$ is perturbed from v^*, it will converge back to v^*. However, the rate of convergence can be very slow. Taking $M = 1700$ kg, $a = 2.6$ m^2, $\rho = 1.2$ kg/m^3, $c = 0.2$, which are reasonable for a sedan, and assuming $v^* = 29$ m/s (≈ 65 mph) we get $\lambda \approx -0.01$ s^{-1}, *i.e.* a time constant of 100 seconds.

For faster convergence we can apply the feedback

$$\tilde{u}(t) = k\tilde{v}(t) \tag{7}$$

which leads to

$$\frac{d}{dt}\tilde{v}(t) = (\lambda + bk)\tilde{v}(t). \tag{8}$$

Then the convergence rate is determined by $\lambda + bk$, which we can assign arbitrarily by selecting k. Since $\tilde{u}(t) = u(t) - u^*$ and $\tilde{v}(t) = v(t) - v^*$, the actual torque applied to the vehicle is

$$u(t) = u^* + k(v(t) - v^*).$$

Example 2 (Robot Car): The robot car used in the lab has two wheels, each driven with a separate electric motor. Let $d_l(t)$ and $d_r(t)$ be the distance traveled by the left and right wheels, and let $u_l(t)$ and $u_r(t)$ denote the respective control inputs (duty cycle of pulse width modulated current). An appropriate model relating these variables is

$$\begin{aligned} d_l(t+1) - d_l(t) &= \theta_l u_l(t) - \beta_l \\ d_r(t+1) - d_r(t) &= \theta_r u_r(t) - \beta_r \end{aligned} \tag{9}$$

where the right hand sides approximate the speed for each wheel.

The parameters for the two wheels may be significantly different. Thus, applying an identical input to both wheels would lead to non-identical speeds, and the car would go in circles. To straighten the trajectory of the car we apply the control inputs

$$\begin{aligned} u_l(t) &= \frac{v^* + \beta_l}{\theta_l} + \frac{k_l}{\theta_l}(d_l(t) - d_r(t)) \\ u_r(t) &= \frac{v^* + \beta_r}{\theta_r} + \frac{k_r}{\theta_r}(d_l(t) - d_r(t)) \end{aligned} \tag{10}$$

where v^* is the desired velocity, and k_l and k_r are constants to be designed. Substitute (10) in (9) to get

$$\begin{aligned} d_l(t+1) - d_l(t) &= v^* + k_l(d_l(t) - d_r(t)) \\ d_r(t+1) - d_r(t) &= v^* + k_r(d_l(t) - d_r(t)). \end{aligned} \tag{11}$$

Next, define $\delta(t) := d_l(t) - d_r(t)$ and note from (11) that it satisfies

$$\delta(t+1) = (1 + k_l - k_r)\delta(t).$$

Thus, to ensure $\delta(t) \to 0$, we need to select k_l and k_r such that

$$|1 + k_l - k_r| < 1.$$

Without the feedback terms in (10), that is $k_l = k_r = 0$, we get

$$\delta(t+1) = \delta(t)$$

which means that the error accumulated in $\delta(t)$ persists and is in fact likely to grow if we incorporate a disturbance term. The feedback in (10) is thus essential to dissipate the error $\delta(t)$ and to keep it bounded in the presence of disturbances.

Example 3: Recall this example from the last lecture:

$$\vec{x}(t+1) = \underbrace{\begin{bmatrix} 0 & 1 \\ a_1 & a_2 \end{bmatrix}}_{A} \vec{x}(t) + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{B} u(t),$$

where the characteristic polynomial of A is

$$\det(\lambda I - A) = \lambda^2 - a_2\lambda - a_1.$$

If we substitute the control

$$u(t) = K\vec{x}(t) = k_1 x_1(t) + k_2 x_2(t)$$

the closed-loop system becomes

$$\vec{x}(t+1) = \underbrace{\begin{bmatrix} 0 & 1 \\ a_1 + k_1 & a_2 + k_2 \end{bmatrix}}_{A + BK} \vec{x}(t)$$

and, since $A + BK$ has the same structure as A with a_1, a_2 replaced by $a_1 + k_1$, $a_2 + k_2$, the eigenvalues of $A + BK$ are the roots of

$$\lambda^2 - (a_2 + k_2)\lambda - (a_1 + k_1).$$

If we want to assign the eigenvalues of $A + BK$ to desired values λ_1 and λ_2, we must match the polynomial above to

$$(\lambda - \lambda_1)(\lambda - \lambda_2) = \lambda^2 - (\lambda_1 + \lambda_2)\lambda + \lambda_1\lambda_2.$$

This is accomplished with the choice

$$k_1 = -a_1 - \lambda_1\lambda_2, \quad k_2 = -a_2 + \lambda_1 + \lambda_2.$$

For example, if we want $\lambda_1 = \lambda_2 = 0$, then $k_1 = -a_1$ and $k_2 = -a_2$.

Example 4: Here is a three-state example where A and B have a structure similar to Example 2:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a_1 & a_2 & a_3 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

and the characteristic polynomial is now

$$\det(\lambda I - A) = \lambda^3 - a_3\lambda^2 - a_2\lambda - a_1.$$

The closed-loop system is

$$\vec{x}(t+1) = \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ a_1 + k_1 & a_2 + k_2 & a_3 + k_3 \end{bmatrix}}_{A + BK} \vec{x}(t)$$

which has characteristic polynomial

$$\det(\lambda I - (A + BK)) = \lambda^3 - (a_3 + k_3)\lambda^2 - (a_2 + k_2)\lambda - (a_1 + k_1). \quad (12)$$

Note that each one of k_1, k_2 and k_3 appears in precisely one coefficient and can change it to any desired value. If we want eigenvalues at λ_1, λ_2, λ_3, we simply match the coefficients of (12) to those of

$$\begin{aligned}(\lambda - \lambda_1)(\lambda - \lambda_2)(\lambda - \lambda_3) \quad &= \quad \lambda^3 - (\lambda_1 + \lambda_2 + \lambda_2)\lambda^2 \\ &\quad -(\lambda_1\lambda_2 + \lambda_1\lambda_3 + \lambda_2\lambda_3)\lambda - \lambda_1\lambda_2\lambda_3\end{aligned}$$

by choosing $k_1 = \lambda_1\lambda_2\lambda_3 - a_1$, $k_2 = \lambda_1\lambda_2 + \lambda_1\lambda_3 + \lambda_2\lambda_3 - a_2$, and $k_3 = \lambda_1 + \lambda_2 + \lambda_3 - a_3$.

Why does controllability enable us to assign the eigenvalues?

We will now show that controllability allows us to arbitrarily assign the eigenvalues of $A + BK$ with the choice of K. The key to our argument is the special form of A and B in Examples 3 and 4, which we generalize to an arbitrary dimension n as:

$$A_c = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ a_1 & a_2 & \cdots & a_{n-1} & a_n \end{bmatrix} \qquad B_c = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix}. \quad (13)$$

This structure is called the "controller canonical form," hence the subscript "c." When A_c has this form, the entries of the last row $a_1, \ldots, a_n$ appear as the coefficients of the characteristic polynomial:

$$\det(\lambda I - A_c) = \lambda^n - a_n\lambda^{n-1} - a_{n-1}\lambda^{n-2} - \cdots - a_2\lambda - a_1.$$

In addition $A_c + B_cK$ preserves the structure of A_c, except that the entry a_i is replaced by $a_i + k_i$, $i = 1, \ldots, n$. Therefore,

$$\det(\lambda I - (A_c + B_cK)) = \lambda^n - (a_n + k_n)\lambda^{n-1} \cdots - (a_2 + k_2)\lambda - (a_1 + k_1)$$

where each one of $k_1, \cdots, k_n$ appears in precisely one coefficient and can change it to any desired value. Thus we can arbitrarily assign the eigenvalues of $A_c + B_cK$ as we did in Examples 3 and 4.

So how do we prove that for any controllable system

$$\vec{x}(t+1) = A\vec{x}(t) + Bu(t) \tag{14}$$

we can assign the eigenvalues of $A + BK$ arbitrarily? We simply show that an appropriate change of variables $\vec{z} = T\vec{x}$ brings A and B to the form (13); that is, there exists T such that

$$TAT^{-1} = A_c \quad \text{and} \quad TB = B_c. \tag{15}$$

This means that we can design a state feedback $u = K_c\vec{z}$ to assign the eigenvalues of $A_c + B_cK_c$ as discussed above for the controller canonical form. Since $\vec{z} = T\vec{x}$, $u = K_c\vec{z}$ is identical to $u = K\vec{x}$ where

$$K = K_cT. \tag{16}$$

Note that $T(A + BK)T^{-1} = A_c + B_cK_c$ and, thus, the eigenvalues of $A + BK$ are identical[4] to those of $A_c + B_cK_c$, which have been assigned to desired values.

Conclusion: If the system (14) is controllable, then we can arbitrarily assign the eigenvalues of $A + BK$ with an appropriate choice of K.

How do we know a matrix T satisfying (15) exists? Since we assumed (14) is controllable, the matrix

$$C = \begin{bmatrix} A^{n-1}B & \cdots & AB & B \end{bmatrix} \tag{17}$$

is full rank and, thus, has inverse C^{-1}. Denoting the top row of C^{-1} by $\vec{q}^T$, we note from the identity $C^{-1}C = I$ that

$$\vec{q}^TC = \begin{bmatrix} \vec{q}^TA^{n-1}B & \cdots & \vec{q}^TAB & \vec{q}^TB \end{bmatrix} = [1\ 0\ \cdots\ 0]. \tag{18}$$

[4] If λ, $\vec{v}$ is an an eigenvalue/eigenvector pair for $A + BK$, that is

$$(A + BK)\vec{v} = \lambda\vec{v},$$

then λ is also an eigenvalue for $A_c + B_cK_c$, with eigenvector $T\vec{v}$. This is because

$$(A_c + B_cK_c)T\vec{v} = (T(A + BK)T^{-1})T\vec{v}$$
$$= T(A + BK)\vec{v} = T\lambda\vec{v} = \lambda T\vec{v}.$$

We will use this equation to show that the choice

$$T = \begin{bmatrix} \vec{q}^T \\ \vec{q}^T A \\ \vdots \\ \vec{q}^T A^{n-1} \end{bmatrix}$$

indeed satisfies (15) where A_c and B_c are as in (13). The second equality in (15) follows because

$$TB = \begin{bmatrix} \vec{q}^T \\ \vec{q}^T A \\ \vdots \\ \vec{q}^T A^{n-1} \end{bmatrix} B = \begin{bmatrix} \vec{q}^T B \\ \vec{q}^T AB \\ \vdots \\ \vec{q}^T A^{n-1} B \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} = B_c$$

by (18). To verify the first equality in (15) note that

$$TA = \begin{bmatrix} \vec{q}^T A \\ \vec{q}^T A^2 \\ \vdots \\ \vec{q}^T A^n \end{bmatrix} \tag{19}$$

and compare this to

$$\begin{aligned} A_c T &= \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & 0 & 1 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 1 \\ a_1 & a_2 & \cdots & a_{n-1} & a_n \end{bmatrix} \begin{bmatrix} \vec{q}^T \\ \vec{q}^T A \\ \vdots \\ \vec{q}^T A^{n-1} \end{bmatrix} \\ &= \begin{bmatrix} \vec{q}^T A \\ \vdots \\ \vec{q}^T A^{n-1} \\ \vec{q}^T (a_1 I + a_2 A + \cdots + a_n A^{n-1}) \end{bmatrix}. \end{aligned} \tag{20}$$

Indeed the rows of (20) and (19) match[5] and thus $TAT^{-1} = A_c$, which is the first equality in (15).

[5] The bottom rows match as a consequence of the Cayley-Hamilton Theorem that you saw in Discussion 8B. It says that a matrix satisfies its own characteristic polynomial:

$$A^n - a_n A^{n-1} - \cdots - a_2 A - a_1 I = 0.$$

EE16B - Spring'20 - Lecture 14A Notes[1]

Murat Arcak

21 April 2020

Upper Triangularization

When a square matrix is not diagonalizable[2] the next best thing we can do is to bring it to an upper triangular form:

$$\begin{bmatrix} \lambda_1 & \star & \cdots & \star \\ & \ddots & \ddots & \vdots \\ & & \ddots & \star \\ & & & \lambda_n \end{bmatrix}. \tag{1}$$

[2] Remember an $n \times n$ matrix is diagonalizable if it has n linearly independent eigenvectors. This is the case when the eigenvalues are distinct. For matrices with repeated eigenvalues diagonalizability depends on the structure of the matrix.

In Lecture 12B we used this form to prove that a discrete-time system is stable when the eigenenvalues of A are inside the unit circle[3] without requiring diagonalizability.

[3] A similar proof establishes stability of a continuous-time system when the eigenenvalues have negative real parts.

In this lecture we will show that any square matrix can be brought to an upper triangular form. The proof is by induction – it is a good exercise in linear algebra as well as in making proofs by induction.

Let's first recall what proof by induction means. Suppose we want to prove that a statement $\mathbf{S}_n$ that depends on an integer $n = 1, 2, 3, \ldots$ is true regardless of n. To do so by induction, we show:

- $\mathbf{S}_1$ is true
- for any $k \geq 1$, if we assume $\mathbf{S}_k$ is true then $\mathbf{S}_{k+1}$ is also true.

In the case of upper triangularization, n is the dimension of the matrix and the statement $\mathbf{S}_n$ is:

For any $n \times n$ matrix A we can find an invertible matrix T such that TAT^{-1} has the upper triangular form (1).

Now $\mathbf{S}_1$ is true because any scalar A has the form (1) with $\lambda_1 = A$. Moving on to the second bullet above, we need to show that:

We can upper triangularize $(k+1) \times (k+1)$ matrices if we assume that $k \times k$ matrices can be upper triangularized.

To show this, let A be an arbitrary $(k+1) \times (k+1)$ matrix and let $\lambda, \vec{v}$ be an eigenvalue/vector pair[4]: $A\vec{v} = \lambda\vec{v}$. Normalize $\vec{v}$ so that $\|\vec{v}\| = 1$ and choose k other vectors $\vec{v}_1, \ldots, \vec{v}_k \in \mathbb{R}^{k+1}$ such that

$$\{\vec{v}, \vec{v}_1, \ldots, \vec{v}_k\} \tag{2}$$

is an orthonormal basis[5] for $\mathbb{R}^{k+1}$. Then, the $(k+1) \times (k+1)$ matrix

[4] We will assume these are real valued. If not, the arguments that follow can be modified by using the definition of inner product for complex vectors and defining orthonormality accordingly.

[5] The Gram-Schmidt procedure from Discussion 12A can be used to construct vectors $\vec{v}_1, \ldots, \vec{v}_k$.

$$V := \begin{bmatrix} \vec{v} & \vec{v}_1 & \cdots & \vec{v}_k \end{bmatrix} \quad \text{is orthogonal:} \quad V^{-1} = V^T = \begin{bmatrix} \vec{v}^T \\ \vec{v}_1^T \\ \vdots \\ \vec{v}_k^T \end{bmatrix}. \tag{3}$$

It follows that

$$\begin{aligned} AV &= A\begin{bmatrix} \vec{v} & \vec{v}_1 & \cdots & \vec{v}_k \end{bmatrix} = \begin{bmatrix} \lambda\vec{v} & A\vec{v}_1 & \cdots & A\vec{v}_k \end{bmatrix} \\ V^{-1}AV &= \begin{bmatrix} \vec{v}^T \\ \vec{v}_1^T \\ \vdots \\ \vec{v}_k^T \end{bmatrix} \begin{bmatrix} \lambda\vec{v} & A\vec{v}_1 & \cdots & A\vec{v}_k \end{bmatrix} \\ &= \begin{bmatrix} \lambda\vec{v}^T\vec{v} & \vec{v}^TA\vec{v}_1 & \cdots & \vec{v}^TA\vec{v}_k \\ \lambda\vec{v}_1^T\vec{v} & \vec{v}_1^TA\vec{v}_1 & \cdots & \vec{v}_1^TA\vec{v}_k \\ \vdots & \vdots & \ddots & \vdots \\ \lambda\vec{v}_k^T\vec{v} & \vec{v}_k^TA\vec{v}_1 & \cdots & \vec{v}_k^TA\vec{v}_k \end{bmatrix} \\ &= \begin{bmatrix} \lambda & \vec{v}^TA\vec{v}_1 & \cdots & \vec{v}^TA\vec{v}_k \\ 0 & \vec{v}_1^TA\vec{v}_1 & \cdots & \vec{v}_1^TA\vec{v}_k \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \vec{v}_k^TA\vec{v}_1 & \cdots & \vec{v}_k^TA\vec{v}_k \end{bmatrix} \end{aligned} \tag{4}$$

where, in the last step, we used the orthonormality of the basis (2). Thus, we can write

$$V^{-1}AV = \begin{bmatrix} \lambda & \vec{q}^T \\ 0 & A_0 \end{bmatrix}$$

where A_0 is the $k \times k$ lower right submatrix in (4) and $\vec{q}^T$ is the row above this submatrix. Since we assumed $k \times k$ matrices can be upper triangularized, there exists a matrix T_0 such that $T_0A_0T_0^{-1}$ is upper triangular. Now define the $(k+1) \times (k+1)$ matrix

$$T := \begin{bmatrix} 1 & 0 \\ 0 & T_0 \end{bmatrix} V^{-1}$$

and note

$$\begin{aligned} TAT^{-1} &= \begin{bmatrix} 1 & 0 \\ 0 & T_0 \end{bmatrix} V^{-1}AV \begin{bmatrix} 1 & 0 \\ 0 & T_0^{-1} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & T_0 \end{bmatrix} \begin{bmatrix} \lambda & \vec{q}^T \\ 0 & A_0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & T_0^{-1} \end{bmatrix} \\ &= \begin{bmatrix} \lambda & \vec{q}^TT_0^{-1} \\ 0 & T_0A_0T_0^{-1} \end{bmatrix} \end{aligned}$$

where $T_0A_0T_0^{-1}$ is upper triangular by assumption. Thus, TAT^{-1} is upper triangular and we conclude that if $k \times k$ matrices can be upper triangularized, then the same is true for $(k+1) \times (k+1)$ matrices. By induction the same conclusion extends to an arbitrary dimension n.

Recall that the eigenvalues of an upper triangular matrix are its diagonal entries[6]. In addition, A and TAT^{-1} have identical eigenvalues[7]. Thus, when we choose T such that TAT^{-1} is upper triangular, the eigenvalues of A appear in the diagonal entries of TAT^{-1}.

[6] See Homework 11.

[7] If λ, $\vec{v}$ is an an eigenvalue/eigenvector pair for A, that is

$$A\vec{v} = \lambda\vec{v},$$

then λ is also an eigenvalue for TAT^{-1}, with eigenvector $T\vec{v}$. This is because

$$(TAT^{-1})T\vec{v} = TA\vec{v} = T\lambda\vec{v} = \lambda T\vec{v}.$$

Example (Critically Damped RLC Circuit): Recall that the RLC circuit depicted on the right can be modeled as

$$\begin{aligned}\frac{dx_1(t)}{dt} &= \frac{1}{C}x_2(t)\\ \frac{dx_2(t)}{dt} &= \frac{1}{L}\left(-x_1(t) - Rx_2(t)\right)\end{aligned}$$

where $x_1 = v_C$ and $x_2 = i$. Rewrite this model in matrix/vector form

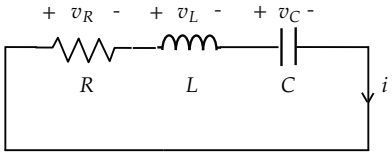
$$\frac{d}{dt}\vec{x}(t) = A\vec{x}(t) \quad \text{where} \quad A = \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L} & -\frac{R}{L} \end{bmatrix}$$

and note that the roots of

$$\det(\lambda I - A) = \lambda^2 + \frac{R}{L}\lambda + \frac{1}{LC}$$

give the eigenvalues:

$$\lambda_{1,2} = -\frac{R}{2L} \mp \sqrt{\left(\frac{R}{2L}\right)^2 - \frac{1}{LC}}. \tag{5}$$



We will analyze the critically damped case, where $C = \frac{4L}{R^2}$ and, thus, the square root term in (5) is zero and we have repeated eigenvalues

$$\lambda_{1,2} = \lambda_c := -\frac{R}{2L}.$$

When $C = \frac{4L}{R^2}$ the matrix A is

$$A = \begin{bmatrix} 0 & \frac{R^2}{4L} \\ -\frac{1}{L} & -\frac{R}{L} \end{bmatrix}$$

and you can verify that the null space of $A - \lambda_c I$ is one-dimensional. Thus, we can't find two linearly independent eigenvectors and A is not diagonalizable. To upper triangularize A we use the eigenvector

$$\vec{v} = \frac{2}{\sqrt{R^2+4}}\begin{bmatrix} \frac{R}{2} \\ -1 \end{bmatrix}$$

which is normalized so that $\|\vec{v}\| = 1$. Then we introduce

$$\vec{v}_1 = \frac{2}{\sqrt{R^2+4}}\begin{bmatrix} 1 \\ \frac{R}{2} \end{bmatrix}$$

so that $\{\vec{v}, \vec{v}_1\}$ is an orthonormal basis for $\mathbb{R}^2$.

Then the matrix V constructed as in (3) is

$$V = \frac{2}{\sqrt{R^2+4}} \begin{bmatrix} \frac{R}{2} & 1 \\ -1 & \frac{R}{2} \end{bmatrix}.$$

Since A is 2×2, it follows from (4) that $V^{-1}AV$ is upper triangular:

$$V^{-1}AV = \begin{bmatrix} -\frac{R}{2L} & \frac{1}{L}\frac{R^2+4}{4} \\ 0 & -\frac{R}{2L} \end{bmatrix}.$$

Thus, the change of variables $\vec{z} = V^{-1}\vec{x}$ leads to the model

$$\begin{aligned} \frac{d}{dt}z_1(t) &= -\frac{R}{2L}z_1(t) + \frac{1}{L}\frac{R^2+4}{4}z_2(t) \\ \frac{d}{dt}z_2(t) &= -\frac{R}{2L}z_2(t). \end{aligned}$$

The second equation for $z_2(t)$ has the simple solution:

$$z_2(t) = e^{-\frac{R}{2L}t}z_2(0).$$

If we substitute this solution into the first equation for $z_1(t)$ we get:

$$\frac{d}{dt}z_1(t) = -\frac{R}{2L}z_1(t) + \underbrace{\frac{1}{L}\frac{R^2+4}{4}e^{-\frac{R}{2L}t}z_2(0)}_{=:\,u(t)}$$

where we treat the second term as an input so that the solution is:

$$\begin{aligned} z_1(t) &= e^{-\frac{R}{2L}t}z_1(0) + \int_0^t e^{-\frac{R}{2L}(t-s)}u(s)ds \\ &= e^{-\frac{R}{2L}t}z_1(0) + \frac{1}{L}\frac{R^2+4}{4}\int_0^t e^{-\frac{R}{2L}(t-s)}e^{-\frac{R}{2L}s}z_2(0)ds \\ &= e^{-\frac{R}{2L}t}z_1(0) + \frac{1}{L}\frac{R^2+4}{4}e^{-\frac{R}{2L}t}\int_0^t z_2(0)ds \\ &= e^{-\frac{R}{2L}t}z_1(0) + \frac{1}{L}\frac{R^2+4}{4}e^{-\frac{R}{2L}t}tz_2(0). \end{aligned}$$

We can put the solution in matrix/vector form:

$$\begin{bmatrix} z_1(t) \\ z_2(t) \end{bmatrix} = \begin{bmatrix} e^{-\frac{R}{2L}t} & \frac{1}{L}\frac{R^2+4}{4}e^{-\frac{R}{2L}t}t \\ 0 & e^{-\frac{R}{2L}t} \end{bmatrix} \begin{bmatrix} z_1(0) \\ z_2(0) \end{bmatrix}$$

and return to the original state variables by substituting $\vec{z} = V^{-1}\vec{x}$:

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = V \begin{bmatrix} e^{-\frac{R}{2L}t} & \frac{1}{L}\frac{R^2+4}{4}e^{-\frac{R}{2L}t}t \\ 0 & e^{-\frac{R}{2L}t} \end{bmatrix} V^{-1} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix}.$$

Substituting V and V^{-1}, and simplifying, we get

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} e^{-\frac{R}{2L}t} + \frac{R}{2L}e^{-\frac{R}{2L}t}t & \frac{R^2}{4L}e^{-\frac{R}{2L}t}t \\ -\frac{1}{L}e^{-\frac{R}{2L}t}t & e^{-\frac{R}{2L}t} - \frac{R}{2L}e^{-\frac{R}{2L}t}t \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix}.$$

The main thing to note here is the presence of the terms $e^{-\frac{R}{2L}t}t$, where the exponential function is multiplied by t. Such terms are characteristic of systems where the matrix A cannot be diagonalized.