EECS 16B Designing Information Systems and Devices II UC Berkeley Spring 2024

Homework 9

This homework is due on Saturday, March 23, 2024, at 11:59PM.

1. Stability Criterion

Consider the complex plane below, which is broken into non-overlapping regions A through H. The circle drawn on the figure is the unit circle $|\lambda| = 1$.

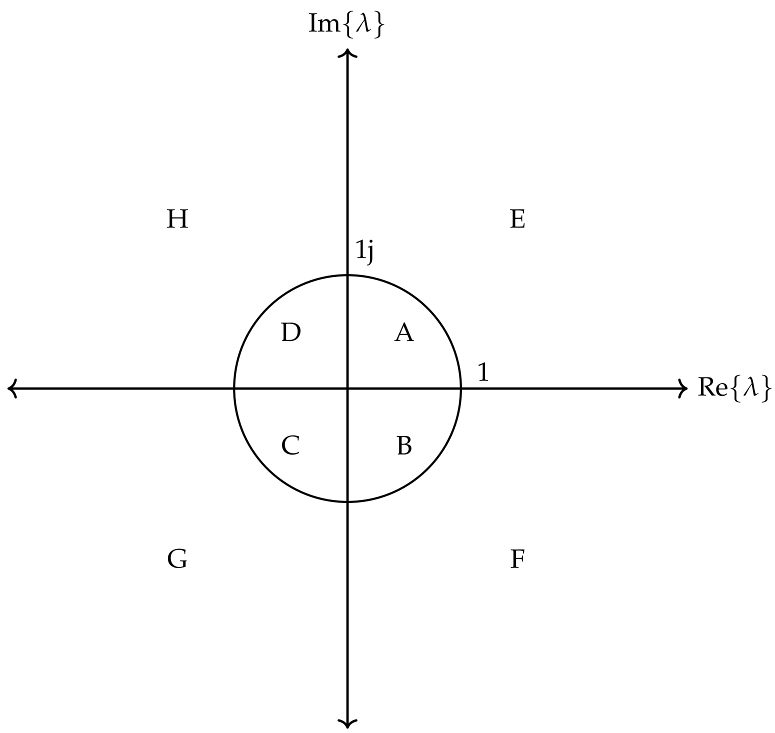


Figure 1: Complex plane divided into regions.

Consider the continuous-time system $\frac{d}{dt}x(t) = \lambda x(t) + v(t)$ and the discrete-time system $y[i+1] = \lambda y[i] + w[i]$. Here $v(t)$ and $w[i]$ are both disturbances to their respective systems.

For both the continuous-time system and discrete-time system, **list the regions (A-H) in which the eigenvalue λ can lie if the system is *stable*.** Assume that the eigenvalue λ does not fall directly on the boundary between two regions.

Solution: For the continuous time system to be stable, we need the real part of λ to be less than zero. Hence, C, D, G, H satisfy this condition.

On the other hand, for the discrete time system to be stable, we need the norm of λ to be less than one. Hence, A, B, C, D satisfy this condition.

2. Bounded-Input Bounded-Output (BIBO) Stability

BIBO stability is a system property where bounded inputs lead to bounded outputs. This is important because it allows us to certify that, provided our system inputs are bounded, the outputs will not "blow up". In this problem, we will gain a better understanding of BIBO stability by considering some simple continuous and discrete systems and showing whether they are BIBO stable or not.

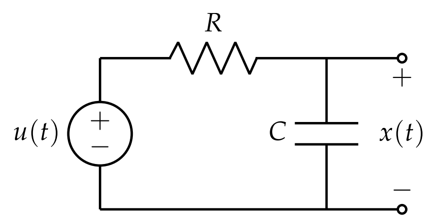
Recall that for the following simple scalar differential equation, we have the corresponding solution:

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t) = ax(t) + bu(t) \qquad x(t) = \mathrm{e}^{at}x(0) + \int_0^t \mathrm{e}^{a(t-\tau)}bu(\tau)\,\mathrm{d}\tau\,. \tag{1}$$

And for the following discrete system, we have the corresponding solution:

$$x[i+1] = ax[i] + bu[i] \qquad x[i] = a^i x[0] + \sum_{k=0}^{i-1} a^k bu[i-1-k] \tag{2}$$

(a) Consider the circuit below with $R = 1\Omega$, $C = 0.5\text{F}$. Let $x(t)$ be the voltage across the capacitor.



This circuit can be modeled by the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t) = -2x(t) + 2u(t) \tag{3}$$

Intuitively, we know that the voltage on the capacitor can never exceed the (bounded) voltage from the voltage source, so this system is BIBO stable. **Demonstrate mathematically that this system is indeed BIBO stable, meaning that $x(t)$ remains bounded for all time if the input $u(t)$ is bounded. Equivalently, show that if we assume $|u(t)| < \epsilon$, $\forall t \geq 0$ and $|x(0)| < \epsilon$, then $|x(t)| < M$, $\forall t \geq 0$ for some positive constant M.** This should help us interpret what bounded-input-bounded-output stability means for a physical circuit.

(HINT: You may want to use eq. (1) to express an upper bound on $|x(t)|$ in terms of $u(t)$ and $x(0)$. Remember that the norm in 1D is an absolute value.

(HINT: Some formulas you may find useful are $|ab| = |a||b|$, the triangle inequality $|a+b| \leq |a| + |b|$, and the integral version of the triangle inequality $\left|\int_a^b f(\tau)\,\mathrm{d}\tau\right| \leq \int_a^b |f(\tau)|\,\mathrm{d}\tau$.))

Solution:

Using eq. (1), we get the solution to the scalar differential equation as

$$x(t) = \mathrm{e}^{-2t}x(0) + \int_0^t \mathrm{e}^{-2(t-\tau)}2u(\tau)\,\mathrm{d}\tau\,. \tag{4}$$

Then we can try to bound $x(t)$ for $t \geq 0$. We first use the triangle inequality ($|a+b| \leq |a| + |b|$) to get

$$|x(t)| = \left|\mathrm{e}^{-2t}x(0) + \int_0^t \mathrm{e}^{-2(t-\tau)}2u(\tau)\,\mathrm{d}\tau\right| \tag{5}$$

$$|x(t)| \leq \left|e^{-2t}x(0)\right| + \left|\int_0^t e^{-2(t-\tau)}2u(\tau)\,d\tau\right| \tag{6}$$

We then use the property that the integral of absolute value will always be greater than the absolute value of the integral (equation (6) to (7)), and that an exponential is always positive (equation (7) to (8)):

$$|x(t)| \leq \left|e^{-2t}x(0)\right| + \int_0^t \left|e^{-2(t-\tau)}2u(\tau)\right|d\tau \tag{7}$$

$$= e^{-2t}|x(0)| + \int_0^t e^{-2(t-\tau)}2|u(\tau)|\,d\tau \tag{8}$$

Finally, plugging in our bounds for $|u(\tau)|$ and $|x(0)|$ and doing the integral:

$$|x(t)| \leq e^{-2t}\epsilon + \int_0^t e^{-2(t-\tau)}2\epsilon\,d\tau \tag{9}$$

$$= e^{-2t}\epsilon + 2\epsilon e^{-2t}\int_0^t e^{2\tau}\,d\tau \tag{10}$$

$$= e^{-2t}\epsilon + 2\epsilon e^{-2t}\frac{1}{2}\left(e^{2t}-1\right) \tag{11}$$

$$= e^{-2t}\epsilon + \epsilon\left(1-e^{-2t}\right) \tag{12}$$

$$= \epsilon,\ \forall t \geq 0 \tag{13}$$

So we see that our state's magnitude is bounded for all time. Note that the negative exponent of the exponential is what makes this system stay bounded.

(b) Assume that $x(0) = 0$. **Show that the system in eq. (1) is BIBO unstable when $a = j2\pi$ by constructing a bounded input that leads to an unbounded $x(t)$.**

It can be shown that the system in eq. (1) is unstable for any purely imaginary a by a similar construction of a bounded input.

Solution: Recall the solution of $x(t)$ with the initial condition at zero

$$x(t) = \int_0^t e^{a(t-\tau)}bu(\tau)\,d\tau\,. \tag{14}$$

Remember, the style of argumentation here is the "counterexample" style. The question asks you to show that *some* bounded input exists that will make the state grow without bound.

Because we know we can get an integral to diverge if we are just integrating a nonzero constant, we decide to try the bounded input $u(t) = \epsilon e^{j2\pi t}$, whose magnitude is equal to ϵ for all t.

Plugging this input and a value in, we see

$$x(t) = \int_0^t e^{j2\pi(t-\tau)}b\epsilon e^{j2\pi\tau}\,d\tau = \int_0^t e^{j2\pi t}b\epsilon\,d\tau\,. \tag{15}$$

Factoring out the terms that do not depend on τ, we are left with

$$x(t) = b\epsilon e^{j2\pi t}\int_0^t d\tau\,. \tag{16}$$

Solving this integral, we get

$$x(t) = b\epsilon t e^{j2\pi t}. \tag{17}$$

Now taking the magnitude of $x(t)$ using the fact that $|e^{j\omega t}| = 1$ for all ω, we get $|x(t)| = \epsilon|b|t$ which clearly diverges as $t \to \infty$.

(c) Consider the discrete-time system and its solution in eq. (2). **Show that if $|a| > 1$, then even if $x[0] = 0$, a bounded input can result in an unbounded output, i.e. the system is BIBO unstable.**

(HINT: The formula for the sum of a geometric sequence may be helpful.)

Solution: Consider when $x[0] = 0$ and $u[i] = 1\ \forall i$. This gives

$$x[i] = a^i x[0] + \sum_{k=0}^{i-1} a^k b u[i-1-k] \tag{18}$$

$$= \sum_{k=0}^{i-1} a^k b \tag{19}$$

$$= b\frac{a^i - 1}{a-1} \quad \text{(as this is the sum of a geometric series)} \tag{20}$$

When $|a| > 1$, then a^i has magnitude that grows without bound, and thus $|x[i]|$ does as well. We also know this from the convergence criteria for geometric series; when the common ratio $a > 1$, the series does not converge to a finite number as $i \to \infty$.

(d) Consider the discrete-time system

$$x[i+1] = -3x[i] + u[i]. \tag{21}$$

Is this system stable or unstable? Give an initial condition $x(0)$ and a sequence of non-zero inputs for which the state $x[i]$ will always stay bounded.

(HINT: See if you can find any input pattern that results in an oscillatory behavior.)

Solution:

The system is unstable since the eigenvalue -3 has magnitude ≥ 1. To see this more explicitly, any non-zero $x[0]$ and (bounded) $u[i] = 0\ \forall i \in \mathbb{N}$ will lead to unbounded x.

Consider $x[0] = 0$ and the input $u[i] = 1, 3, 1, 3, 1, 3, \ldots$.

t	0	1	2	3	$\cdots$
$x[i]$	0	1	0	1	$\cdots$
$u[i]$	1	3	1	3	$\cdots$
$-3x[i] + u[i]$	1	0	1	0	$\cdots$

In this case, we get $x[i] = 0$ when t is even, and $x[i] = 1$ when i is odd. In fact, there are an infinite number of input sequences that would result in bounded outputs.

3. Eigenvalue Placement through State Feedback

Consider the following discrete-time linear system:

$$\vec{x}[i+1] = \begin{bmatrix} -2 & 2 \\ -2 & 3 \end{bmatrix} \vec{x}[i] + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u[i]. \tag{22}$$

In standard language, we have $A = \begin{bmatrix} -2 & 2 \\ -2 & 3 \end{bmatrix}, \vec{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ in the form: $\vec{x}[i+1] = A\vec{x}[i] + \vec{b}u[i]$.

(a) **Is this discrete-time linear system stable in open loop (without feedback control)?**

Solution: We have to calculate the eigenvalues of matrix A. Thus,

$$\begin{align} 0 &= \det(\lambda I - A) \tag{23} \\ &= \det \begin{bmatrix} \lambda+2 & -2 \\ 2 & \lambda-3 \end{bmatrix} \tag{24} \\ &= \lambda^2 - \lambda - 2 \tag{25} \\ \implies \lambda_1 &= 2, \qquad \lambda_2 = -1 \tag{26} \end{align}$$

Since at least one eigenvalue has a magnitude that is greater than or equal to 1, the discrete-time system is unstable. In this case, both of the eigenvalues are unstable.

(b) Suppose we use state feedback of the form $u[i] = \begin{bmatrix} f_1 & f_2 \end{bmatrix} \vec{x}[i] = F\vec{x}[i]$.

Find the appropriate state feedback constants, f_1, f_2 so that the state space representation of the resulting closed-loop system has eigenvalues $\lambda_1 = -\frac{1}{2}, \lambda_2 = \frac{1}{2}$.

Solution: The closed loop system using state feedback has the form

$$\begin{align} \vec{x}[i+1] &= \begin{bmatrix} -2 & 2 \\ -2 & 3 \end{bmatrix} \vec{x}[i] + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u[i] \tag{27} \\ &= \begin{bmatrix} -2 & 2 \\ -2 & 3 \end{bmatrix} \vec{x}[i] + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} f_1 & f_2 \end{bmatrix} \vec{x}[i] \tag{28} \\ &= \left(\begin{bmatrix} -2 & 2 \\ -2 & 3 \end{bmatrix} + \begin{bmatrix} f_1 & f_2 \\ f_1 & f_2 \end{bmatrix} \right) \vec{x}[i] \tag{29} \end{align}$$

Thus, the closed loop system has the form

$$\vec{x}[i+1] = \underbrace{\begin{bmatrix} -2+f_1 & 2+f_2 \\ -2+f_1 & 3+f_2 \end{bmatrix}}_{A_{\text{cl}}} \vec{x}[i] \tag{30}$$

Finding the characteristic polynomial of the above system, we have

$$\begin{align} \det\left(\lambda I - \begin{bmatrix} -2+f_1 & 2+f_2 \\ -2+f_1 & 3+f_2 \end{bmatrix}\right) &= (\lambda+2-f_1)(\lambda-3-f_2) - (-2-f_2)(2-f_1) \tag{31} \\ &= \lambda^2 - f_1\lambda - f_2\lambda - \lambda + f_1 f_2 - 6 - 2f_2 + 3f_1 \tag{32} \\ &\quad - (-4 + f_1 f_2 + 2f_1 - 2f_2) \tag{33} \end{align}$$

$$= \lambda^2 - (1 + f_1 + f_2)\lambda + f_1 - 2 \tag{34}$$

However, we want to place the eigenvalues at $\lambda_1 = -\frac{1}{2}, \lambda_2 = \frac{1}{2}$. That means we want

$$\lambda^2 - (1 + f_1 + f_2)\lambda + f_1 - 2 = \left(\lambda + \frac{1}{2}\right)\left(\lambda - \frac{1}{2}\right) \tag{35}$$

or equivalently:

$$\lambda^2 - (1 + f_1 + f_2)\lambda + f_1 - 2 = \lambda^2 - \frac{1}{4} \tag{36}$$

Equating the coefficients of the different powers of λ on both sides of the equation, we get,

$$1 + f_1 + f_2 = 0 \tag{37}$$

$$f_1 - 2 = -\frac{1}{4} \tag{38}$$

Solving the above system of equations gives us $f_1 = \frac{7}{4}, f_2 = -\frac{11}{4}$.

4. (Lab) Open-Loop and Closed-Loop Control

In last week's lab-related System ID problem, we developed a linear model for the velocity of each wheel. We are one step away from our goal: to have SIXT33N drive in a straight line! We will see how to use the model we developed in the System ID problem to control SIXT33N's trajectory to be a straight line.

Part 1: Open-Loop Control

An open-loop controller is one in which the input is predetermined using your system model and the goal, and not adjusted at all during operation. To design an open-loop controller for your car, you would set the PWM duty-cycle values of the left and right wheels (inputs $u_L[i]$ and $u_R[i]$) such that the predicted velocity of both wheels is your target wheel velocity (v_t). You can calculate these inputs from the target velocity v_t and the $\theta_L, \theta_R, \beta_L, \beta_R$ values you learned from data. In the System ID problem and lab, we have modeled the velocity of the left and right wheels as

$$v_L[i] = d_L[i+1] - d_L[i] = \theta_L u_L[i] - \beta_L; \tag{39}$$
$$v_R[i] = d_R[i+1] - d_R[i] = \theta_R u_R[i] - \beta_R \tag{40}$$

where $d_{L,R}[i]$ represent the distance traveled by each wheel.

(a) Find the open-loop control that would give us $v_L[i] = v_R[i] = v_t$. That is, **solve the model (Equations (39) and (40)) for the inputs $u_L[i]$ and $u_R[i]$ that make the velocities $v_L[i] = v_R[i] = v_t$.**

Solution: Starting from Equations (39) and (40) and substituting in the target velocity v_t, we get the following equations.

$$v_t = \theta_L u_L[i] - \beta_L \tag{41}$$
$$v_t = \theta_R u_R[i] - \beta_R \tag{42}$$
$$v_t + \beta_L = \theta_L u_L[i] \tag{43}$$
$$v_t + \beta_R = \theta_R u_R[i] \tag{44}$$
$$\frac{v_t + \beta_L}{\theta_L} = u_L[i] \tag{45}$$
$$\frac{v_t + \beta_R}{\theta_R} = u_R[i] \tag{46}$$

In practice, the parameters $\theta_L, \theta_R, \beta_L, \beta_R$ are learned from noisy data, so they can be wrong and lead us to calculate the velocities for the two wheels incorrectly. When the velocities of the two wheels disagree, the car will go in a circle instead of a straight line. Thus, to make the car go in a straight line, we need the distances traveled by both wheels to be the same at each timestep.

This prompts us to simplify our model. Instead of having two state variables $\vec{v}_L$ and $\vec{v}_R$, we can just have a state variable determining how far we are from the desired behavior of going in a line – a state which we will want to drive to 0.

This prompts us to define our state variable δ to be the *difference* in the distance traveled by the left wheel and the right wheel at a given timestep:

$$\delta[i] := d_L[i] - d_R[i] \tag{47}$$

We want to find a scalar discrete-time model for $\delta[i]$ of the form

$$\delta[i+1] = \lambda_{\text{OL}}\delta[i] + f(u_L[i], u_R[i]). \tag{48}$$

Here λ_{OL} is a scalar and $f(u_L[i], u_R[i])$ is the control input to the system (as a function of $u_L[i]$ and $u_R[i]$).

(b) Suppose we apply the open-loop control inputs $u_L[i], u_R[i]$ to the original system. **Using Equations (39) and (40), write $\delta[i+1]$ in terms of $\delta[i]$, in the form of Equation (48). What is the eigenvalue λ_{OL} of the model in Equation (48)? Would the model in Equation (48) be stable with open-loop control if it also had a disturbance term?**

(HINT: For open-loop control, we set the velocities to $v_L[i] = v_R[i] = v_t$. What happens when we substitute that into Equations (39) and (40) and then apply the definition of $\delta[i]$ and $\delta[i+1]$?)

Solution: Proceeding by the hint,

$$\begin{aligned}
\delta[i+1] &= d_L[i+1] - d_R[i+1] && (49)\\
&= (v_L[i] + d_L[i]) - (v_R[i] + d_R[i]) && (50)\\
&= v_t + d_L[i] - (v_t + d_R[i]) && (51)\\
&= d_L[i] - d_R[i] && (52)\\
&= \delta[i] && (53)
\end{aligned}$$

From the derivation above, $\lambda_{\text{OL}} = 1$ and $f(u_L[i], u_R[i]) = 0$. To check stability, we already know our eigenvalue does not meet the stability criteria: $|\lambda_{\text{OL}}| = 1$, so we have an unstable system if we add disturbances (whereas if we don't then the system is *marginally stable*).

Part 2: Closed-Loop Control

Now, in order to make the car drive straight, we must implement closed-loop control – that is, control inputs that depend on the current state and are calculated dynamically – using feedback in real time.

(c) **If we want the car to drive straight starting from some timestep $i_{\text{start}} > 0$, i.e. $v_L[i] = v_R[i]$ for $i \geq i_{\text{start}}$, what condition does this impose on $\delta[i]$ for $i \geq i_{\text{start}}$?**

(HINT: Part (d) gives a clue as to one solution that is isn't correct.)

Solution: Let $i \geq i_{\text{start}}$. Then

$$\begin{aligned}
0 &= v_L[i] - v_R[i] && (54)\\
&= d_L[i+1] - d_L[i] - (d_R[i+1] - d_R[i]) && (55)
\end{aligned}$$

$$= (d_L[i+1] - d_R[i+1]) - (d_L[i] - d_R[i]) \tag{56}$$
$$= \delta[i+1] - \delta[i]. \tag{57}$$

Thus

$$\delta[i+1] = \delta[i], \qquad i \geq i_{\text{start}} \tag{58}$$

which implies that

$$\delta[i] = \delta[i_{\text{start}}], \qquad i \geq i_{\text{start}}. \tag{59}$$

In other words, we have that for every timestep beyond i_{start}, the difference in distances the wheels have traveled does not change.

(d) **How is the condition you found in the previous part different from the condition:**

$$\delta[i] = 0, \qquad i \geq i_{\text{start}}? \tag{60}$$

Assume that $i_{\text{start}} > 0$, and that $d_L[0] = 0, d_R[0] = 0$.

This is a subtlety that is worth noting and often requires one to adjust things in real systems.

Solution: At time $i = 0$, the car has not moved yet, so $\delta[0] = d_L[0] - d_R[0] = 0$. If at some later time i_{start} we have $\delta[i_{\text{start}}] = 0$ and $\delta[i] = 0$ for later times as well, we remain moving in the same direction we started with. When $\delta[i] \neq 0$, this means the wheels have moved different distances, and therefore has moved along a curved path and changed the direction the car is pointing.

While not required, Fig. 2 illustrates the two different cases where $\delta[i] = 0$ for all times $i \geq 0$ (left) and when $\delta \neq 0$ initially but we have $\delta[i_{\text{start}}] = 0$ for some $i = i_{\text{start}}$ and $\delta[i] = 0$ for $i \geq i_{\text{start}}$ (right).

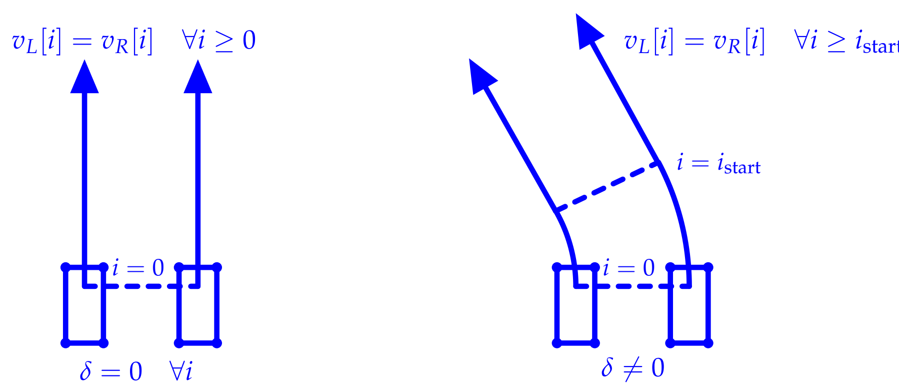


Figure 2

(e) From here, assume that we have reset the distance travelled counters at the beginning of this maneuver so that $\delta[0] = 0$. We will now implement a feedback controller by selecting two dimensionless positive coefficients, f_L and f_R, such that the closed loop system is stable with eigenvalue λ_{CL}. To implement closed-loop feedback control, we want to adjust $v_L[i]$ and $v_R[i]$ at each timestep by an amount proportional to $\delta[i]$. Not only do we want our wheel velocities to be some target velocity v_t, we also wish to drive $\delta[i]$ towards zero. This is in order to have the car drive straight along the initial direction it was pointed in when it started moving. If $\delta[i]$ is positive, the left wheel has traveled more distance than the right wheel, so relatively speaking, we can slow down the left wheel and speed up the right wheel to cancel this difference (i.e., drive it to zero) in the next few timesteps. The action of such a control is captured by the following velocities.

$$v_L[i] = v_t - f_L\delta[i]; \tag{61}$$
$$v_R[i] = v_t + f_R\delta[i]. \tag{62}$$

Give expressions for $u_L[i]$ and $u_R[i]$ as a function of v_t, $\delta[i]$, f_L,f_R, and our system parameters $\theta_L.\theta_R, \beta_L, \beta_R$, to achieve the velocities above.
(HINT: Remember our System ID equations. How can we relate them to Equations (61) and (62)? This should bear some similarity to our open-loop solution from part (a))

Solution: As in the open loop case, we substitute the velocity expressions above into the equations that relate $v[i]$ and $u[i]$.

For the left wheel we have:

$$v_t - f_L\delta[i] = \theta_L u_L[i] - \beta_L \tag{63}$$
$$v_t - f_L\delta[i] + \beta_L = \theta_L u_L[i] \tag{64}$$
$$\frac{v_t - f_L\delta[i] + \beta_L}{\theta_L} = u_L[i] \tag{65}$$

For the right wheel we have:

$$v_t + f_R\delta[i] = \theta_R u_R[i] - \beta_R \tag{66}$$
$$v_t + f_R\delta[i] + \beta_R = \theta_R u_R[i] \tag{67}$$
$$\frac{v_t + f_R\delta[i] + \beta_R}{\theta_R} = u_R[i] \tag{68}$$

(f) Using the control inputs $u_L[i]$ and $u_R[i]$ found in part (e), **write the closed-loop system equation for $\delta[i+1]$ as a function of $\delta[i]$. What is the closed-loop eigenvalue λ_{CL} for this system in terms of λ_{OL}, f_L, and f_R?**

Solution: We can take the system equation explicitly in terms of $u_L[i]$ and $u_R[i]$ from the solution of part (c) in eq. (48), and substitute into this equation our control expressions from the previous part.

$$\begin{aligned}
\delta[i+1] &= \delta[i] + \theta_L u_L[i] - \theta_R u_R[i] - \beta_L + \beta_R && (69)\\
&= \delta[i] + \theta_L\left(\frac{v_t - f_L\delta[i] + \beta_L}{\theta_L}\right) - \theta_R\left(\frac{v_t + f_R\delta[i] + \beta_R}{\theta_R}\right) - \beta_L + \beta_R && (70)\\
&= \delta[i] + v_t - f_L\delta[i] - (v_t + f_R\delta[i]) && (71)\\
&= \delta[i] - f_L\delta[i] - f_R\delta[i] && (72)\\
&= (1 - f_L - f_R)\delta[i] && (73)
\end{aligned}$$

We see that our λ_{CL} will end up being $1 - f_L - f_R$, which is equal to $\lambda_{\mathrm{OL}} - f_L - f_R$.

(g) **Under what condition on f_L and f_R is the closed-loop system from the previous part stable in the presence of disturbance?**

Solution:

$$
\begin{aligned}
& |\lambda_{\mathrm{CL}}| < 1 && (74)\\
\implies & |1 - f_L - f_R| < 1 && (75)\\
\implies & -1 < 1 - f_L - f_R < 1 && (76)\\
\implies & 0 < f_L + f_R < 2 && (77)
\end{aligned}
$$

Stability in this case means that δ is bounded and will not go arbitrarily high. In fact, if our calculated β and θ are perfectly accurate, then $\delta[i] \to 0$, so the car will (eventually) drive straight!

5. Designing a Controllable Aircraft

Suppose we model an aircraft as the following discrete system, where $s[i]$ is speed and $p[i]$ is pitch:

$$\begin{bmatrix} s[i+1] \\ p[i+1] \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \begin{bmatrix} s[i] \\ p[i] \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} u[i]$$

You are working with aeronautical engineers to come up with a design for an aircraft that can always recover from a stall.

(a) Suppose that these are the designed system parameters.

$$\begin{bmatrix} s[i+1] \\ p[i+1] \end{bmatrix} = \begin{bmatrix} 4 & 7 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} s[i] \\ p[i] \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \end{bmatrix} u[i]$$

Is the system controllable?

Solution: The system is controllable because

$$\begin{aligned} C_2 &= \begin{bmatrix} A\vec{b} & \vec{b} \end{bmatrix} \\ &= \begin{bmatrix} 18 & 1 \\ -4 & 2 \end{bmatrix} \end{aligned}$$

is full rank.

(b) Based on your answer for the previous part, **is it possible for the aircraft to recover from a stall (i.e. an underisable state)? Why or why not?** Note: Recovering means that the aircraft leaves an undesirable state and gets steered to (or eventually reaches) a desirable state via control.

Solution: Yes. Since the system that describes the aircraft's state is controllable, the system can reach any final state from any initial state. Therefore, if we started from a state that is a stall, which is an undesirable state, the aircraft will be able to recover from the stall by reaching a desirable state.

(c) Regardless of the previous parts, assume you found that the system was actually not controllable. The engineers re-design the parts and you find the following system parameters:

$$\begin{bmatrix} s[i+1] \\ p[i+1] \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -2 & 3 \end{bmatrix} \begin{bmatrix} s[i] \\ p[i] \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u[i]$$

Given an initial state where the aircraft has stalled (an undesirable state), does there exist a sequence of inputs that can recover the aircraft? Why or why not? Note: Recall that recovering means that the aircraft leaves an undesirable state and gets steered to (or eventually reaches) a desirable state.

Solution: The system is controllable; we can compute the controllability matrix once again and show that it is full rank. Moreover, controllability implies reachability since the column space of A is all of $\mathbb{R}^2$. Therefore, there does exist a sequence of inputs that can recover the aircraft.

Contributors:
- Sidney Buchbinder.
- Tanmay Gautam.
- Ashwin Vangipuram.

- Nathan Lambert.
- Anant Sahai.
- Sally Hui.
- Druv Pai.
- Varun Mishra.
- Bozhi Yin.
- Kaitlyn Chan.
- Yi-Hsuan Shih.
- Vladimir Stojanovic.
- Moses Won.
- Anish Muthali.