EECS 16B Designing Information Devices and Systems II
Summer 2020 Note 12

1 Overview

In the previous note, we saw how to take a linear Continuous-Time system and discretize is by feeding in piecewise-constant inputs to get a Discrete-Time system that we could potentially control. However, we must now ask the following question: Can any system be controlled? In other words, is it possible after a finite number of time-steps for our system to reach any state in our state-space?

We will see that not every system can be controlled and that there is a quick test to see if a system is indeed **controllable**. To show this, we will take a step by step recursive approach similar to what we did for discretization.

2 Control Inputs

The first question that we should ask is whether we can control any discrete-time system. Let's start with the scalar case

$$x[n+1] = \lambda x[n] + bu[n] \tag{1}$$

Based on the observations from the previous note, it seems that this system can always be controlled, except for when $b = 0$. When $b = 0$, it is impossible for our input to even reach our system. Therefore, the system acts on its own and we will see its natural response.

Similarly for the vector case, if our system was represented as

$$\vec{x}[n+1] = A\vec{x}[n] + B\vec{u}[n] \tag{2}$$

Our B matrix may put some restrictions on how we can directly control our system. For example, take the case when $B = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$. Here, the input is only able to directly affect the second state.

Now that we have established that not every system can be controlled, let's take a look at a few examples.

2.1 One Shot Wonder

Consider the discrete-time system with initial state $\vec{x}[0]$.

$$\vec{x}[n+1] = A\vec{x}[n] + B\vec{u}[u] \tag{3}$$

Our goal is to reach the state $\vec{x}^*$ in one timestep. We can unroll the difference equation to see that

$$\vec{x}[1] = A\vec{x}[0] + B\vec{u}[0] \tag{4}$$

Since $\vec{u}[0]$ is an input we have control over, if $\vec{x}^* - A\vec{x}[0] \in \text{Range}(B)$, we can reach our target in one timestep.

However, note that the requirement of reaching a desired state in one-timestep is often a stringent one.

Therefore, in the next example, we observe what happens after two time steps.

2.2 Second Chance

Consider the discrete-time system with initial state $\vec{x}[0] = \vec{0}$.

$$\vec{x}[n+1] = A\vec{x}[n] + B\vec{u}[u] \tag{5}$$

Then after one time-step, we can unroll the difference equation and to see that

$$\vec{x}[1] = A\vec{x}[0] + B\vec{u}[0] = B\vec{u}[0] \tag{6}$$

Since $u[0]$ is an input we have control over, after one time-step, we can reach anywhere in the span of the columns of B. To see this, recall that[1]

$$B\vec{u}[0] = \begin{bmatrix} \vec{b}_1 & \vec{b}_2 \end{bmatrix} \begin{bmatrix} \vec{u}_1[0] \\ \vec{u}_2[0] \end{bmatrix} \tag{7}$$

After one more time-step, we again unroll the difference equation

$$\vec{x}[2] = A\vec{x}[1] + B\vec{u}[1] = AB\vec{u}[0] + B\vec{u}[1] \tag{8}$$

and it follows that after two time steps, we can reach anywhere in the span$\{B, AB\}$.

Since we have relaxed our constraints to two time steps, the range of points we can reach has also grown.

2.3 n^{th} Time's the Charm

Let's now see which states we can reach after n time-steps. Assuming a non-zero initial condition,

$$\begin{aligned} \vec{x}[n] &= A\vec{x}[n-1] + B\vec{u}[n-1] = A(A\vec{x}[n-2] + B\vec{u}[n-2]) + B\vec{u}[n-1] && (9) \\ &= A^2\vec{x}[n-2] + AB\vec{u}[n-2] + B\vec{u}[n-1] && (10) \\ &\vdots && (11) \\ &= A^n\vec{x}[0] + \sum_{k=0}^{n-1} A^{n-1-k}B\vec{u}[k] && (12) \end{aligned}$$

We could also write this out in the following matrix-vector form

$$\vec{x}[n] = A^n\,\vec{x}[0] + \begin{bmatrix} B & AB & \dots & A^{n-1}B \end{bmatrix} \begin{bmatrix} \vec{u}[n-1] \\ \vdots \\ \vec{u}[0] \end{bmatrix} \tag{13}$$

The inputs $\vec{u}[0], \vec{u}[1], \dots, \vec{u}[n-1]$ are all chosen by the person designing the controller, so after n time-steps, we can reach anywhere in the

$$\text{Span}\{B, AB, \dots, A^{n-1}B\}$$

[1]The notation $\vec{u}_i[0]$ refers to the i^{th} entry of the vector $\vec{u}[0]$.

3 Controllability

With all of our intuition from the previous section, let us now try to formalize an argument for when a system is **controllable.** A given system is controllable if we can reach any state in a finite number of time steps.

We saw that after k time-steps, we can reach anywhere in the span of the columns of $B, AB, \ldots, A^{k-1}B$. If the span after a finite number of time steps, were equal to $\mathbb{R}^n$, then we could say our system is controllable. Intuitively, the span should grow as k increases but at what point would the span stop increasing?

3.1 Scalar Inputs

We will formalize the argument for scalar inputs and will make a remark at the end on how we can generalize this to vector inpts. Suppose we had the discrete-time system

$$\vec{x}[n+1] = A\vec{x}[n] + \vec{b}u[n] \tag{14}$$

Now suppose after k time steps, $\text{Span}\{\vec{b}, A\vec{b}, \ldots, A^{k-1}\vec{b}\}$ has dimension k and adding $A^k\vec{b}$ does not change the span. This implies that $A^k\vec{b}$ is linearly dependent to the first k vectors meaning we can write it as a linear combination of the remaining vectors

$$A^k\vec{b} = \alpha_1\vec{b} + \ldots + \alpha_k A^{k-1}\vec{b} \tag{15}$$

Then we can also write A^{k+1} as a linear combination of the first k vectors

$$A^{k+1}\vec{b} = A(c_1\vec{b} + \ldots + c_k A^{k-1}\vec{b}) = c_1 A\vec{b} + \ldots + c_k A^k\vec{b} \tag{16}$$

$$= \alpha_1\vec{b} + \ldots \alpha_k A^{k-1}\vec{b} \in \text{Span}\{\vec{b}, A\vec{b}, \ldots, A^{k-1}\vec{b}\} \tag{17}$$

If $k < n$, the argument above shows that our system will never reach every vector in $\mathbb{R}^n$.

Therefore, to summarize our work, we define the controllability matrix

$$\mathscr{C} = \begin{bmatrix} \vec{b} & A\vec{b} & \ldots & A^{n-1}\vec{b} \end{bmatrix} \tag{18}$$

We can say that if our controllability matrix has rank $k < n$, then our system is **uncontrollable** while if our matrix is full rank, then our system is **controllable.**

3.2 Vector Inputs

The argument for vector inputs is slightly different, but the end result is the same. We can use the same controllability matrix to determine whether our system is controllable.

$$\mathscr{C} = \begin{bmatrix} B & AB & \ldots & A^{n-1}B \end{bmatrix} \tag{19}$$

However, to prove controllability for the multiple input case, we will have to invoke the **Cayley-Hamilton Theorem**[2] which states that a matrix satisfies its characteristic polynomial. What this means is if a matrix A has the following characteristic polynomial,

$$\lambda^n + \alpha_{n-1}\lambda^{n-1} + \ldots + \alpha_1\lambda + \alpha_0 = 0 \tag{20}$$

[2] The proof of this theorem is beyond the scope of this course. If you're interested in learning more about this theorem, take Math 110 or EE221A.

then the following matrix equation must also be true.

$$A^n + \alpha_{n-1}A^{n-1} + \ldots + \alpha_1 A + \alpha_0 I = 0 \tag{21}$$

In either case, this shows that we can write A^n as a linear combination of lower powers of A and we can make the same argument that adding A^nB to the controllability matrix will not change its rank.

3.3 Continuous-Time Controllability

So far, all of our analysis has been done for discrete-time systems. However, can we show whether a Continuous-Time system is controllable? It turns out that as long as our state-vector is differentiable n times, we can show that

$$\begin{aligned}
\frac{d}{dt}\vec{x}(t) &= A\vec{x}(t) + B\vec{u}(t) && (22)\\
\frac{d^2}{dt^2}\vec{x}(t) &= A\frac{d}{dt}\vec{x}(t) + B\frac{d\vec{u}(t)}{dt} = A\left(A\vec{x}(t) + B\vec{u}(t)\right) + B\frac{d\vec{u}(t)}{dt} && (23)\\
&= A^2\vec{x}(t) + AB\vec{u}(t) + B\frac{d\vec{u}(t)}{dt} && (24)\\
&\vdots && (25)\\
&= A^n\vec{x}(t) + \sum_{k=0}^{n-1} A^{n-1-k}B\frac{d^k\vec{u}(t)}{dt^k} && (26)\\
&= A^n\vec{x}(t) + \begin{bmatrix} B & AB & \ldots & A^{n-1}B \end{bmatrix}\begin{bmatrix} \vec{u}(t) \\ \vdots \\ \frac{d^{n-1}\vec{u}(t)}{dt^{n-1}} \end{bmatrix} && (27)
\end{aligned}$$

Therefore, we can show that the controllability matrix is in fact the same for Continuous-Time systems.

$$\mathscr{C} = \begin{bmatrix} B & AB & \ldots & A^{n-1}B \end{bmatrix} \tag{28}$$

4 Controllability Examples

4.1 Finding our Inputs

Consider the following discrete-time system with initial state $\vec{x}[0] = \begin{bmatrix} 1 & -2 \end{bmatrix}^T$.

$$\vec{x}[n+1] = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}\vec{x}[n] + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u[n] \tag{29}$$

Then the controllability matrix can be computed as

$$\mathscr{C} = \begin{bmatrix} \vec{b} & A\vec{b} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 2 \end{bmatrix} \implies \text{Rank}\,\mathscr{C} = 2 \tag{30}$$

Therefore our system is controllable and we can reach any state $\vec{x} \in \mathbb{R}^2$ in two time steps. To figure out what inputs $u(0)$ and $u(1)$ we must give to reach $\vec{x}^*$, we can solve the following system of equations

$$\vec{x}[2] = A^2\vec{x}[0] + \begin{bmatrix} \vec{b} & A\vec{b} \end{bmatrix} \begin{bmatrix} u(0) \\ u(1) \end{bmatrix} \implies \begin{bmatrix} u(0) \\ u(1) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 2 \end{bmatrix}^{-1} \left(\vec{x}^* - A^2 \begin{bmatrix} 1 \\ -2 \end{bmatrix} \right) \tag{31}$$

4.2 Diagonal System

Suppose we have the following diagonal system

$$\vec{x}[n+1] = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \vec{x}[n] + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u[n] \tag{32}$$

The controllability matrix can be computed as

$$\mathscr{C} = \begin{bmatrix} \vec{b} & A\vec{b} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & \lambda_2 \end{bmatrix} \implies \text{Rank}\,\mathscr{C} = 1 < 2 \tag{33}$$

However, we could have also broken up the system into the two individual scalar equations

$$x_1[n+1] = \lambda_1 x_1[n] + 0 \cdot u[n] \tag{34}$$
$$x_2[n+1] = \lambda_2 x_2[n] + u[n] \tag{35}$$

Note how $u[n]$ cannot reach our first state x_1. Therefore, the system must be uncontrollable.

4.3 Another Diagonal System

Now consider following diagonal system with $\lambda \neq 0$

$$\vec{x}[n+1] = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \vec{x}[n] + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u[n] \tag{36}$$

The controllability matrix can be computed as

$$\mathscr{C} = \begin{bmatrix} \vec{b} & A\vec{b} \end{bmatrix} = \begin{bmatrix} 1 & \lambda \\ 1 & \lambda \end{bmatrix} \implies \text{Rank}\,\mathscr{C} = 1 < 2 \tag{37}$$

Another way to see that this system is uncontrollable is to again break down the system into individual scalar equations

$$x_1[n+1] = \lambda x_1[n] + u[n] \tag{38}$$
$$x_2[n+1] = \lambda x_2[n] + u[n] \tag{39}$$

Since the same input is influencing both systems simulatenously, the trajectory of the state $\vec{x}[n]$ will move continue to move in a line.

