AE353: Additional Notes on Controllability

(to be treated as an appendix to the presentation)

A. Borum T. Bretl M. Ornik P. Thangeda

1 Controllability

Consider the state-space system

$$\dot{x} = Ax + Bu \tag{1}$$

$$y = Cx. \tag{2}$$

So far this semester, we've considered controllers that have the form

$$u = -Kx + k_{\text{ref}} r + d,$$

and we've focused on choosing K so that the matrix $A - BK$ has eigenvalues at desired locations. We've also focused on reference tracking when the output y is a scalar, i.e., making the output y go to the reference value r.

Another goal that we might want to achieve is to make the entire state x go to a desired value at some time t_f. Let's suppose that the initial value for the state at time t_0 is x_0, so that $x(t_0) = x_0$. Our goal is to choose an input $u(t)$, which is a function of time, so that $x(t_f) = x_f$, where x_f is some desired final value of the state. If it is possible to move the state x from any initial state x_0 to any final state x_f in a finite time, we say that the state-space system (1)-(2) is **controllable**. Is it always possible to choose an input $u(t)$ to make this happen?

The short answer is no, it is not always possible. However, we have a very simple test to tell us when it is possible. The controllability matrix of the state-space system (1)-(2), which we've already seen in the notes on Ackermann's method, is given by

$$W = \begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix},$$

where n is the dimension of the state x. The test for controllability can be stated as follows:

The system (1)-(2) is controllable if and only if W is **full rank**.

Let's break the above statement down. First, let's consider the case when the input u in (1) is a scalar. In that case, B is an $n \times 1$ matrix and, as always, A is an $n \times n$ matrix. We then see that the matrix W has size $n \times n$, and W being full rank simply means that it is invertible.

Now let's suppose that we have more than one input. Let's say that we have $m > 1$ inputs, so that B has size $n \times m$. The matrix W then has size $n \times nm$, and W is no longer square. Although we can't invert a non-square matrix, we can still check its rank. You can read here to learn more about the rank of non-square matrices. It is easy to check the rank of a non-square matrix in Matlab using the rank function.

2 An algorithm for steering a state-space system

We won't provide all of the details that prove the system (1)-(2) is controllable if and only if W is full rank. We will, however, describe an algorithm for finding an input u that drives the system to a desired final state when the system is controllable.

Let's consider a system with a scalar input. Let's also suppose that the initial time is $t_0 = 0$, the initial state is $x_0 = 0$, the final time is t_f, and the desired final state is x_f. Now let's divide the time interval $[t_0, t_f]$ into n intervals each having a length of h. (We could allow each of the intervals to have a different length, but we'll stick with one common length for simplicity.) Let's also assume that on each of the intervals, the input u is constant. We can write u as

$$u(t) = \begin{cases} u^1 & \text{for } t \in [0, h) \\ u^2 & \text{for } t \in [h, 2h) \\ \vdots & \\ u^n & \text{for } t \in [(n-1)h, nh) \end{cases}$$

where each of the u^i's is a constant. Note that the last time interval in the definition of $u(t)$ is equivalent to $[(n-1)h, nh) = [t_f - h, t_f)$.

The state at time $t = h$ is given by

$$x(h) = e^{Ah}x(0) + A^{-1}\left(e^{Ah} - I\right)Bu^1.$$

Since $x(0) = 0$, this simplifies to

$$x(h) = A^{-1}\left(e^{Ah} - I\right)Bu^1. \tag{3}$$

Next, the state at time $t = 2h$ is given by

$$x(2h) = e^{Ah}x(h) + A^{-1}\left(e^{Ah} - I\right)Bu^2,$$

which, from (3), is equivalent to

$$x(2h) = e^{Ah}A^{-1}\left(e^{Ah} - I\right)Bu^1 + A^{-1}\left(e^{Ah} - I\right)Bu^2.$$

If we define $F = e^{Ah}$ and $G = A^{-1}\left(e^{Ah} - I\right)B$, we can rewrite $x(2h)$ as

$$x(2h) = FGu^1 + Gu^2.$$

Continuing this process, we can see that

$$x(3h) = F^2Gu^1 + FGu^2 + Gu^3$$

and

$$x(nh) = F^{n-1}Gu^1 + \cdots + FGu^{n-1} + Gu^n.$$

Since $t_f = nh$, we can rewrite this expression as

$$x(t_f) = \begin{bmatrix} G & FG & \cdots & F^{n-1}G \end{bmatrix} \begin{bmatrix} u^n \\ u^{n-1} \\ \vdots \\ u^1 \end{bmatrix}$$

Since we've assumed that the input u is a scalar, the matrix $\begin{bmatrix} G & FG & \cdots & F^{n-1}G \end{bmatrix}$ is square. If this matrix is invertible, we can find the values of u^i. Since we want $x(t_f) = x_f$, the values of u_i are given by

$$\begin{bmatrix} u^n \\ u^{n-1} \\ \vdots \\ u^1 \end{bmatrix} = \begin{bmatrix} G & FG & \cdots & F^{n-1}G \end{bmatrix}^{-1} x_f.$$

Notice that the matrix $\begin{bmatrix} G & FG & \cdots & F^{n-1}G \end{bmatrix}$ in the above expression has a similar structure to that of the controllability matrix W for the system (1)-(2).

3 Immovable eigenvalues

What happens when a state-space system is not controllable? To understand this case, let's reconsider Ackermann's method. In the notes on Ackermann's method, we saw that if the controllability matrix W is invertible, we can place eigenvalues at desired locations. It turns out that if the controllability matrix W is not invertible, then only some of the system's eigenvalues can be placed at desired locations, and the remaining eigenvalues cannot be moved. In other words, some of the eigenvalues of the matrix $A - BK$ remain the same for every choice of K.

Is it possible to determine how many eigenvalues we can place (these are called controllable eigenvalues or controllable modes) and how many cannot be moved (these are called uncontrollable eigenvalues or uncontrollable modes)? Yes, we can, by looking at the rank of the controllability matrix. If the rank of the controllability matrix W is n, we can place all of the eigenvalues at desired locations, and the system is controllable. If the rank of W is $n-1$, we can place $n-1$ eigenvalues, and 1 eigenvalue is uncontrollable. If the rank of W is $n-2$, we can place $n-2$ eigenvalues, and 2 eigenvalues are uncontrollable. Continuing this pattern, if the rank of W is $n-k$, we can place $n-k$ eigenvalues, and k eigenvalues are uncontrollable.

4 Example and MATLAB Snippets

This sections presents an example that demonstrates the algorithm for finding an input u that drives the system to a desired final state when the system is controllable. Let us start with checking if the system is controllable:

```matlab
% define system state-space model
A = [-0.30 -0.70 -0.30 -0.60 0.00; 1.00 -0.90 -0.20 -0.70 0.70; 0.30 0.20 0.00 -0.30 0.20; 1.00 0.10 0.00 0.30 -0.40; 0.20 0.90 0.50 0.00 -0.50];
B = [-0.20; 0.40; -1.00; 0.10; -0.50];

% check controllability
unco = length(A) - rank(ctrb(A,B));
if unco == 0
    disp('system is controllable')
else
    disp('system has uncontrollable states')
end
```

The given system is controllable. Also, we know that $n = 5$ for this system. Let us assume that $h = 0.7$. Unlike the algorithm described above which assumes $x(0) = 0$, let us consider a non-zero $x(0)$ (can you derive the expression for obtaining u when $x(0) \neq 0$ following the steps in the algorithm described above?). We then proceed find the sequence of inputs u that drives the system to a desired x_f:

```matlab
% define required data
n = length(A);
h = 0.70;
x0 = [0.10; -0.20; -0.70; -0.70; -0.80];ssssssss
xn = [-0.30; -0.30; -0.10; 1.00; -0.20];

% finding u by inversion
F = expm(A*h);
G = inv(A)*(expm(A*h)-eye(n))*B;
E = ctrb(F,G);
u = inv(E)*(xn - F^(n)*x0);
```