

ECE 717

Homework 3: Realization theory

due: Wednesday October 23, 2019

- 1. Jordan normal form.** Any matrix $A \in \mathbb{R}^{n \times n}$ can be transformed (using a similarity transform) into *Jordan normal form*, which is a block-diagonal matrix that looks like:

$$T^{-1}AT = \begin{bmatrix} J_1 & & 0 \\ & \ddots & \\ 0 & & J_r \end{bmatrix} \quad \text{with} \quad J_i = \begin{bmatrix} \lambda_i & 1 & \cdots & 0 \\ \vdots & \lambda_i & \ddots & \vdots \\ 0 & \vdots & \ddots & 1 \\ 0 & 0 & \cdots & \lambda_i \end{bmatrix}.$$

Each ‘‘Jordan block’’ J_i is constant along the diagonal and this diagonal entry is an eigenvalue of A . The diagonal immediately above the main diagonal is all 1’s. Jordan blocks can only exist when eigenvalues are repeated, and the size of each Jordan block depends on the geometric multiplicity of the corresponding eigenvalue. In the special case where A happens to be diagonalizable, each Jordan block is 1×1 and $T^{-1}AT$ is a diagonal matrix. So we recover the standard eigenvalue decomposition in this case.

Suppose that a SISO LTI system (A, B, C, D) has an A -matrix that is a single Jordan block:

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right] = \left[\begin{array}{ccccc|c} \lambda & 1 & 0 & \cdots & 0 & b_1 \\ 0 & \lambda & 1 & \cdots & 0 & b_2 \\ \vdots & \vdots & \lambda & \ddots & \vdots & \vdots \\ 0 & 0 & \vdots & \ddots & 1 & b_{n-1} \\ 0 & 0 & 0 & \cdots & \lambda & b_n \\ \hline c_1 & c_2 & c_3 & \cdots & c_n & d \end{array} \right]$$

- a) Show that (A, B) is controllable if and only if $b_n \neq 0$.
- b) Show that (C, A) is observable if and only if $c_1 \neq 0$.

SOLUTION:

- a) We’ll use the PBH test for controllability, which states that the following are equivalent:

(a) (A, B) is controllable

(b) If $x \in \mathbb{C}^n$ (and $x \neq 0$) and $\mu \in \mathbb{C}$ satisfy $A^T x = \mu x$, then $B^T x \neq 0$.

Suppose $A^T x = \mu x$. Letting $x = [x_1^* \ \cdots \ x_n^*]^*$, we have:

$$\begin{array}{lll} \lambda x_1 = \mu x_1 & (\lambda - \mu)x_1 = 0 & (\lambda - \mu)x_1 = 0 \\ \lambda x_2 + x_1 = \mu x_2 & (\lambda - \mu)x_2 + x_1 = 0 & (\lambda - \mu)^2 x_2 = 0 \\ \lambda x_3 + x_2 = \mu x_3 & \iff (\lambda - \mu)x_3 + x_2 = 0 & \implies (\lambda - \mu)^3 x_3 = 0 \\ \vdots & \vdots & \vdots \\ \lambda x_n + x_{n-1} = \mu x_n & (\lambda - \mu)x_n + x_{n-1} = 0 & (\lambda - \mu)^n x_n = 0 \end{array}$$

Since $x \neq 0$, the only way to satisfy these equations is if $\mu = \lambda$. Making this substitution back into the original equations, we conclude that $x_1 = x_2 = \cdots = x_{n-1} = 0$ and $x_n \neq 0$ is arbitrary. But now $B^T x = b_n x_n$. So $B^T x \neq 0$ if and only if $b_n \neq 0$, as required.

- b) We can prove this part in a very similar fashion to how we proved part a. The PBH test says that (C, A) is observable if and only if every eigenpair (x, μ) of A also satisfies $Cx \neq 0$. This is essentially the same as part a except we are working with A rather than A^T . Once again, we conclude that $\mu = \lambda$ except this time, $x_1 \neq 0$ is arbitrary and $x_2 = x_3 = \dots = x_n = 0$. $Cx = c_1x_1$, so (C, A) is observable if and only if $c_1 \neq 0$, as required.

2. Realizing a transfer function. For the transfer function $H(s) = \frac{s+1}{s^2+2}$, find:

- a) an uncontrollable and observable realization,
- b) a controllable and unobservable realization,
- c) an uncontrollable and unobservable realization,
- d) a controllable and observable (minimal) realization.

SOLUTION: Recall the Kalman Canonical form:

$$H(s) = \left[\begin{array}{cccc|c} A_{c\bar{o}} & A_{12} & A_{13} & A_{14} & B_{c\bar{o}} \\ 0 & A_{co} & 0 & A_{24} & B_{co} \\ 0 & 0 & A_{\bar{c}\bar{o}} & A_{34} & 0 \\ 0 & 0 & 0 & A_{\bar{c}o} & 0 \\ \hline 0 & C_{co} & 0 & C_{\bar{c}o} & 0 \end{array} \right]$$

The controllable canonical form for $H(s)$ is: $H(s) = \left[\begin{array}{cc|c} 0 & 1 & 0 \\ -2 & 0 & 1 \\ \hline 1 & 1 & 0 \end{array} \right]$ and it has two states, which is

the same as the degree of $H(s)$ (and there are no pole-zero cancellations). Therefore this realization is minimal, and we can use it for (A_{co}, B_{co}, C_{co}) . We can get solutions to the other parts by adding a component to the appropriate part of the Kalman canonical form. I give possible solutions below, and color in red the states that are removable.

a) Uncontrollable and observable: $H(s) = \left[\begin{array}{ccc|c} 0 & 1 & 0 & 0 \\ -2 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{array} \right]$

b) Controllable and unobservable: $H(s) = \left[\begin{array}{ccc|c} 0 & 1 & 0 & 0 \\ -2 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{array} \right]$

c) Uncontrollable and unobservable: $H(s) = \left[\begin{array}{ccc|c} 0 & 1 & 0 & 0 \\ -2 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{array} \right]$

d) Controllable and Observable: $H(s) = \left[\begin{array}{cc|c} 0 & 1 & 0 \\ -2 & 0 & 1 \\ \hline 1 & 1 & 0 \end{array} \right]$

3. **Kalman canonical form.** In this problem, you will write a MATLAB function that transforms a system into Kalman canonical form. We will do it in three parts. The following MATLAB commands might come in handy as you write your code:

- `P = ctrb(A,B)` returns the controllability matrix.
- `Q = obsv(A,C)` returns the observability matrix.
- `T = orth(M)` returns a matrix whose columns are a basis for $\text{range}(M)$.
- `T = null(M)` returns a matrix whose columns are a basis for $\text{null}(M)$.

a) Given two subspaces of \mathbb{R}^n with bases given by the columns of T_1 and T_2 respectively, the MATLAB function below returns a matrix with columns that form a basis for the intersection of the two subspaces. Explain how it works.

```
function T = intersect_subspaces(T1,T2)
    [n,r] = size(T1);
    S = null( [T1 T2] );
    T = T1 * S(1:r,:);
end
```

SOLUTION: If x belongs to the intersection of the subspaces with bases T_1 and T_2 , then we must be able to write $x = T_1 z_1$ and $x = -T_2 z_2$ for some choice of z_1 and z_2 . In other words, $T_1 z_1 + T_2 z_2 = 0$. This means that $[T_1 \ T_2] \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = 0$. Put another way, $z \in \text{null}([T_1 \ T_2])$.

The set of all z that make this possible is therefore $\text{null}([T_1 \ T_2])$. Given such a z , we can recover x by computing $T_1 z_1$. i.e. we should multiply T_1 by the first r rows of z , where r is the size of z_1 (number of columns of T_1).

In the code, we start by computing r , the number of columns of T_1 (which is also the size of z_1). Then, we compute the nullspace of $[T_1 \ T_2]$, which gives us a basis for the set of possible z 's (which we call S). Then, the required basis of the intersection of both subspaces is the product of T_1 and the first r rows of S , as explained above.

b) Given two subspaces of \mathbb{R}^n with bases given by the columns of T_1 and T_2 respectively, with $\text{range}(T_1) \subseteq \text{range}(T_2)$, the MATLAB function below returns a matrix T with columns that complete the basis. That is, $\text{range}([T_1 \ T]) = \text{range}(T_2)$. Explain how it works.

```
function T = complete_basis(T1,T2)
    T1bar = null(T1');
    T = intersect_subspaces(T2,T1bar);
end
```

SOLUTION: The first line computes $\text{null}(T_1^T)$. Let's dissect this. If $z \in \text{null}(T_1^T)$, then $T_1^T z = 0$. This is the set of vectors orthogonal to the range of T_1 . To see why, note that every vector $x \in \text{range}(T_1)$ can be written as $T_1 w$ for some w . Now if $T_1^T z = 0$, then $x^T z = w^T T_1^T z = w^T 0 = 0$. In the code, the set of z 's is suggestively named \bar{T}_1 . Now we know why: $[T_1 \ \bar{T}_1]$ is square and invertible (a basis for \mathbb{R}^n). Since \bar{T}_1 completes T_1 in \mathbb{R}^n , then we should take the intersection $\text{range}(T_2) \cap \text{range}(\bar{T}_1)$ to find the vectors that are orthogonal to T_1 but also belong to T_2 (i.e. the completion of T_1 in T_2). This is precisely what the second line of code does.

c) Write a MATLAB function `[T1,T2,T3,T4] = kcf(A,B,C)` that takes as input the matrices for a state-space realization and returns the blocks of the matrix T that transforms the system into to Kalman canonical form. Use the convention from class that $T = [T_{c\bar{o}} \ T_{co} \ T_{\bar{c}\bar{o}} \ T_{\bar{c}o}]$.

SOLUTION: We can now put everything together and follow the steps laid out in class for constructing the T_i matrices for the Kalman canonical decomposition. Code is below.

```
function [T1,T2,T3,T4] = kcf( A,B,C )
% KCF: takes as input the state space matrices (A,B,C) and returns the
%      transformation that converts a system to Kalman canonical form.

n = size(A,1);

P = ctrb(A,B); % controllability matrix
Q = obsv(A,C); % observability matrix

Ctr = orth(P); % basis for controllable subspace
Obs = null(Q); % basis for unobservable subspace

T1 = intersect_subspaces(Ctr,Obs); % basis for (Ctr n Obs)
T2 = complete_basis(T1,Ctr); % complete basis in Ctr
T3 = complete_basis(T1,Obs); % complete basis in Obs
T4 = complete_basis([T1 T2 T3],eye(n)); % complete the rest

end

% given two subspaces with bases given by the columns of T1 and T2
% respectively, this returns a matrix whose columns form a basis for the
% intersection of the two subspaces.
function T = intersect_subspaces(T1, T2)
    [n,r] = size(T1);
    S = null( [T1 T2] );
    T = T1*S(1:r,:);
end

% given a subspace T1 that is a subset of another subspace T2, this
% returns a basis that completes the first subspace into the second one.
% That is, we have [T1 T] is a basis for T2.
function T = complete_basis(T1, T2)
    T1bar = null(T1');
    T = intersect_subspaces(T2,T1bar);
end
```

- d) Write a MATLAB function `[Am,Bm,Cm] = reduce(A,B,C)` that returns a minimal realization via the Kalman canonical form. To test your function, find a minimal realization for

$$\left[\begin{array}{cccc|c} 5 & 9 & 2 & 1 & 2 \\ -5 & -6 & -1 & 0 & -1 \\ -7 & -13 & -5 & -2 & -3 \\ 14 & 5 & 2 & -4 & 4 \\ \hline 7 & 16 & 3 & 3 & 0 \end{array} \right]$$

Compare your result to the MATLAB command `minreal(ss(A,B,C,0))`. You can see if the realizations are the same by comparing their transfer function computed using `tf(...)`.

SOLUTION: To find a minimal realization, we begin by converting to Kalman canonical form, and then extracting the controllable and observable parts of the realization. Here is the code.

```

function [Am,Bm,Cm] = reduce( A,B,C )
% REDUCE: returns a minimal realization of the LTI system with state-space
% matrices (A,B,C). We assume D=0 since it doesn't matter.

% compute Kalman canonical form
[T1, T2, T3, T4] = kcf(A,B,C);
T = [T1 T2 T3 T4];

% transform to a KCF realization
At = T\A*T;
Bt = T\B;
Ct = C*T;

% isolate controllable + observable coordinates (the T2 block).
[~,n1] = size(T1);
[~,n2] = size(T2);
ix = n1+1:n1+n2 ;

% extract corresponding blocks of the realization.
Am = At(ix,ix);
Bm = Bt(ix,:);
Cm = Ct(:,ix);

end

```

We can run this on the example given in the problem as follows:

```

A = [5 9 2 1; -5 -6 -1 0; -7 -13 -5 -2; 14 5 2 -4];
B = [2; -1; -3; 4];
C = [7 16 3 3];
D = 0;

[Am,Bm,Cm] = reduce(A,B,C)

Hmin = ss(Am,Bm,Cm,0)           % minimal realization
Hmin2 = minreal(ss(A,B,C,0))   % using matlab commands

% compare transfer functions
tf(Hmin)
tf(Hmin2)

% show Kalman Canonical Form
% compute Kalman canonical form (round to 4 decimal digits)
[T1, T2, T3, T4] = kcf(A,B,C);
T = [T1 T2 T3 T4];
At=round(T\A*T,4);
Bt=round(T\B,4);
Ct=round(C*T,4);
ss(At,Bt,Ct,0) % print out the state-space realization

```

The Kalman canonical form approach yields $(A_m, B_m, C_m) = (-2, -4.369, -0.2289)$ while the matlab `minreal` command yields $(A_m, B_m, C_m) = (-2, -3.09, -0.3237)$. However, these are actually equivalent! Both have transfer function $\frac{1}{s+2}$. The Kalman Canonical Form returned by the code is

(with the controllable+observable part shown in red):

$$\left(\begin{array}{cccc|c} -1 & 1.757 & -4.715 & 13.65 & -6.396 \\ 0 & -2 & 0 & 24.97 & -4.369 \\ 0 & 0 & -3 & -10.35 & 0 \\ 0 & 0 & 0 & -4 & 0 \\ \hline 0 & -0.2289 & 0 & -12.7 & 0 \end{array} \right)$$

and we can see that this is indeed in the KCF form we expect to see.