

%%%

%%% Example of Balanced Realization Procedure of Section 7.4 of Chen

%%%

%%% First, create an A matrix with stable eigenvalues -1,-10, and -100

D = [-100 0 0;0 -10 0;0 0 -1];

P1 = [1 2 0;1 0 1;0 0 -1];

iP1 = eye(3)/P1;

%disp('Verify that P*P^{-1} = I');

iP1*P1;

%%% The system

A = P1*D*iP1

B = [0 0 1]'

C = [1 0 0;
0 0 1]

pause

%%% Verify that the system is controllable, observable, and minimal

disp('=====')

disp(' ')

disp('Verify that the system is controllable, observable, and minimal')

disp(' ')

disp('=====')

disp(' ')

disp('Verify that system is controllable')

disp('rank([B,A*B A^2*B])')

rank([B,A*B A^2*B])

disp('Verify that system is observable')

disp('rank([C;C*A;C*A^2])')

rank([C;C*A;C*A^2])

pause

disp('Verify that system is minimal')

disp('sys = ss(A,B,C,0); minreal(sys)')

sys = ss(A,B,C,0)

minreal(sys)

pause

%%% Find the controllability and observability Gramians

disp('=====')

disp(' ')

disp('Find the controllability and observability Gramians')

disp(' ')

```

disp('=====')
disp(' ')
disp(' ')
disp('Find the controllability Gramian Wc')
disp('Wc = lyap(A,B*B')')
Wc = lyap(A,B*B')
disp('Verify that Wc is positive definite via eig(Wc)')
eig(Wc)
disp('Find the observability Gramian Wo')
disp('Wo = lyap(A',C'*C)')
Wo = lyap(A',C'*C)
disp('Verify that Wo is positive definite via eig(Wo)')
eig(Wo)

pause

%%% Compute the similarity transformation P that will give us Wcbar = Wobar = Sigma

disp('=====')
disp(' ')
disp('Compute the similarity transformation P that will give us Wcbar = Wobar = Sigma')
disp(' ')
disp('=====')
disp(' ')

disp('Obtain the Cholesky factorization of Wc')
disp('R = chol(Wc)')
R = chol(Wc)
disp('Verify that R'*R-Wc = 0')
R'*R-Wc
iR = eye(3)/R
disp('Verify that R*iR = I')
R*iR

pause

disp('[U,S,V] = svd(R*Wo*R')')
[U,S,V] = svd(R*Wo*R')
disp('Verify singular value decomposition')
disp('U-V')
U-V
disp('U*S*V'-R*Wo*R')
U*S*V'-R*Wo*R'

pause

Sig = sqrt(S)
sqrtSig = sqrt(Sig)
disp('Verify that have correctly computed sqrt(S)')
disp('sqrtSig^4-S^2')
sqrtSig^4-S

P = sqrtSig*U'*iR'

```

```
iP = eye(3)/P
disp('Verify that P*iP = I')
P*iP
```

```
pause
```

```
%%% Find Wobar and Wcbar
```

```
disp('Wcbar = P*Wc*P''')
Wcbar = P*Wc*P'
disp('Verify that Wcbar = Sig')
disp('Wcbar-Sig')
Wcbar-Sig
```

```
pause
```

```
disp('Wobar = iP''*Wo*iP')
Wobar = (eye(3)/P)'*Wo/P %% Wobar = iP''*Wo*iP
disp('Verify that Wobar = Sig')
disp('Wobar-Sig')
Wobar-Sig
```

```
pause
```

```
%%% Model Reduction
```

```
disp('=====')
disp(' ')
disp('Create reduced order model')
disp(' ')
disp('=====')
disp(' ')

```

```
disp('Resulting balanced reduced order system')
```

```
%%% Find Abar, Bbar, Cbar
```

```
disp('Abar = P*A*iP; Bbar = P*B; Cbar = C*iP')
```

```
Abar = P*A*iP
Bbar = P*B
Cbar = C*iP
```

```
pause
```

```
disp('brsys = ss(Abar(1:2,1:2),Bbar(1:2,1),Cbar(:,1:2),zeros(2,1))')
brsys = ss(Abar(1:2,1:2),Bbar(1:2,1),Cbar(:,1:2),zeros(2,1))
disp('Eigenvalues of reduced order system')
eig(Abar(1:2,1:2))
```

```
pause
```

```
%%% Matlab's approaches
```

```

disp('=====')
disp(' ')
disp('Compare results using Matlab order reduction commands')
disp(' ')
disp('=====')
disp(' ')

```

```

disp('Method 1: balred command')
disp('[sysb,g] = balred(sys,2)')
[sysb] = balred(sys,2)
disp('Extract the A matrix')
disp('[Ar,Br,Cr,Dr] = ssdata(sysb)')
[Ar,Br,Cr,Dr] = ssdata(sysb)
disp('Eigenvalues of reduced order system')
eig(Ar)

```

pause

```

disp('Method 2a: modred command with matching of DC gain')

```

```

disp('elim = (g<1e-2)')
elim = (g<1e-2)
disp('rdcsys = modred(sys,elim,"MatchDC")')
rdcsys = modred(sys,elim,'MatchDC')
disp('Extract the A matrix')
disp('[Ardc,Brdc,Crdc,Drdc] = ssdata(sysb)')
[Ardc,Brdc,Crdc,Drdc] = ssdata(rdcsys)
disp('Eigenvalues of reduced order system with matched DC gain')
eig(Ardc)

```

pause

```

disp('Method 2b: modred command with truncation')

```

```

disp('rtrsys = modred(sys,elim,"truncate")')
rtrsys = modred(sys,elim,'truncate')
disp('Eigenvalues of reduced order system')
disp('Extract the A matrix')
disp('[Artr,Brtr,Crtr,Drtr] = ssdata(rtrsys)')
[Artr,Brtr,Crtr,Drtr] = ssdata(rtrsys)
disp('Eigenvalues of truncated reduced order system')
eig(Artr)

```

pause

```

disp('Method 4: balancmr command')
disp('[sysb2,redinfo] = balancmr(sys,2)')
[sysb2,redinfo] = balancmr(sys,2)
disp('Extract the A matrix')
disp('[Ar2,Br2,Cr2,Dr2] = ssdata(sysb2)')
[Ar2,Br2,Cr2,Dr2] = ssdata(sysb2)
disp('Eigenvalues of reduced order system')
eig(Ar2)

```