Part-1

ZSO zero state observable

IF THEN ZeroStateObservable

Part-2

ZSD zero state detectable

IF THEN ZeroStateDetectable

Part-3

DISSIPATIVITY

IF THEN dissipative-wrt-supply-rate-

Part-4

PASSIVITY

IF THEN passive

Part-5

Passive Memoryless system

IF THEN passive-memoryless-system

Part-6

Corollary 10.8.4 Alberto Isidori, NL-CTRL-SYSs II (1999)

IF

is passive and ZSD

is passive-memoryless-system

THEN

Diagram, schematic

Description automatically generated

This “Memoryless negative feedback” is GAS.

EXAMPLE-1 SISO-nonlinear-passive-dynamical-system

This example is taken from the following paper

Text

Description automatically generated

Text

Description automatically generated

Question-1: is this system passive?

Question-2: is this system ZSD?

|  |
| --- |
| Matlab code |
| %% add MOSEK  %% add YALMIP  %% passivity: "V(x)>0" "[u'\*y]-Vdot>=0"  x1 = sdpvar(1,1);x2 = sdpvar(1,1);x3 = sdpvar(1,1);u = sdpvar(1,1);  f1=-(x1^3+x1\*x3^2)\*(1+x3^2)  f2=-(x1^2\*x2+x2)\*(1+x3^2)  f3=-(x3+x1^2\*x3)\*(1+x3^2)-3\*x3+u  eps1=1e-4; eps1\_poly=eps1\*(x1^2+x2^2+x3^3);  y=x3  P=sdpvar(3,3);  a1=sdpvar(1,1);a2=sdpvar(1,1);a3=sdpvar(1,1);  V=[x1;x2;x3]'\*P\*[x1;x2;x3]+a1\*x1^4+a2\*x2^4+a3\*x3^4;  options = sdpsettings('solver','mosek');  Vdot=jacobian(V,[x1;x2;x3])\*[f1;f2;f3];  % [sol,m,Q,residuals,everything] = solvesos(F,obj,options,params,candidateMonomials)  % [c,v] = COEFFICIENTS(p,x) p(x) = c'\*v(x)  [coeff\_V,mono\_V] = coefficients(V,[x1,x2,x3])  [sol,m,Q,residuals,everything] = solvesos([sos(V-eps1\_poly),sos(u\*x3-Vdot)],[],options,[coeff\_V],[])  value(P)  value(a1)  value(a2)  value(a3) |

To simulate the system with ode45

|  |
| --- |
| Matlab code |
| clear all,close all,clc;  fig1=figure(1);fig1.Color=[1,1,1];  ax1=axes('Parent',fig1);  set(0,'CurrentFigure',fig1);  set(fig1,'currentaxes',ax1);  for ii=1:1:10  tspan=[0:0.01:5]; x0=randi([-10,10],3,1);  wt=tspan;  % f=randi([1,20],1,1); w=sin(2\*pi\*f\*tspan);  w=square(tspan);  [t,x]=ode45(@(t,x) odefcn(t,x,wt,w),tspan,x0);  x1\_vec=x(:,1);x2\_vec=x(:,2);x3\_vec=x(:,3);  plot(t,x1\_vec,'r-','LineWidth',[2],"Parent",ax1);hold on;  plot(t,x2\_vec,'g-','LineWidth',[2],"Parent",ax1);hold on;  plot(t,x3\_vec,'b-','LineWidth',[2],"Parent",ax1);hold on;  hold on;yline(1);yline(-1);  end  function xdot=odefcn(t,x,wt,w)  w=interp1(wt,w,t);  xdot=zeros(3,1);  x1=x(1);x2=x(2);x3=x(3);  f1=-(x1^3+x1\*x3^2)\*(1+x3^2)  f2=-(x1^2\*x2+x2)\*(1+x3^2)  f3=-(x3+x1^2\*x3)\*(1+x3^2)-3\*x3+[-atan(x3)];  xdot(1)=f1;  xdot(2)=f2;  xdot(3)=f3;  end |

Example-2

Text, letter

Description automatically generated

Question-1: is this system passive?

Question-2: is this system ZSD?

|  |
| --- |
| Matlab code |
| %% add MOSEK  %% add YALMIP  %% passivity: "V(x)>0" "[u'\*y]-Vdot>=0"  x1 = sdpvar(1,1); x2 = sdpvar(1,1); x3 = sdpvar(1,1);  u1 = sdpvar(1,1); u2 = sdpvar(1,1);  f1=-(x1^3+x1\*x3^2)\*(1+x3^2)  f2=-(x1^2\*x2+x2)\*(1+x3^2)+u2  f3=-(x3+x1^2\*x3)\*(1+x3^2)-3\*x3+u1  eps1=1e-4; eps1\_poly=eps1\*(x1^2+x2^2+x3^3);  y=[x3;x2]  P=sdpvar(3,3);  a1=sdpvar(1,1);a2=sdpvar(1,1);a3=sdpvar(1,1);  V=[x1;x2;x3]'\*P\*[x1;x2;x3]+a1\*x1^4+a2\*x2^4+a3\*x3^4;  options = sdpsettings('solver','mosek');  Vdot=jacobian(V,[x1;x2;x3])\*[f1;f2;f3];  % [sol,m,Q,residuals,everything] = solvesos(F,obj,options,params,candidateMonomials)  % [c,v] = COEFFICIENTS(p,x) p(x) = c'\*v(x)  [coeff\_V,mono\_V] = coefficients(V,[x1,x2,x3])  [sol,m,Q,residuals,everything] = solvesos([sos(V-eps1\_poly),sos([u1\*x3+u2\*x2]-Vdot)],[],options,[coeff\_V],[])  value(P)  value(a1)  value(a2)  value(a3) |

To simulate the system with ode45

|  |
| --- |
| Matlab code |
| clear all,close all,clc;  fig1=figure(1);fig1.Color=[1,1,1];  ax1=axes('Parent',fig1);  set(0,'CurrentFigure',fig1);  set(fig1,'currentaxes',ax1);  for ii=1:1:20  tspan=[0:0.01:10]; x0=randi([-10,10],3,1);  wt=tspan;  % f=randi([1,20],1,1); w=sin(2\*pi\*f\*tspan);  w=square(tspan);  [t,x]=ode45(@(t,x) odefcn(t,x,wt,w),tspan,x0);  x1\_vec=x(:,1);x2\_vec=x(:,2);x3\_vec=x(:,3);  plot(t,x1\_vec,'r-','LineWidth',[1],"Parent",ax1);hold on;  plot(t,x2\_vec,'g-','LineWidth',[1],"Parent",ax1);hold on;  plot(t,x3\_vec,'b-','LineWidth',[1],"Parent",ax1);hold on;  end  hold on;yline(1);yline(-1);  axis square;  function xdot=odefcn(t,x,wt,w)  w=interp1(wt,w,t);  xdot=zeros(3,1);  x1=x(1);x2=x(2);x3=x(3);  f1=-(x1^3+x1\*x3^2)\*(1+x3^2);  f2=-(x1^2\*x2+x2)\*(1+x3^2)+[-atan(x2)];  f3=-(x3+x1^2\*x3)\*(1+x3^2)-3\*x3+[-atan(x3)];  xdot(1)=f1;  xdot(2)=f2;  xdot(3)=f3;  end |