Case Study

The objective of this case study is to construct a state-feedback stabilizing controller of the nonlinear model of the inverted pendulum on a cart, also known as the stick balancer. The controller is constructed using fuzzy model of the nonlinear model. We also investigate the performance of two different LMI solvers for computing the stabilizing fuzzy controller's gains.

Recall the nonlinear model of the stick balancer,

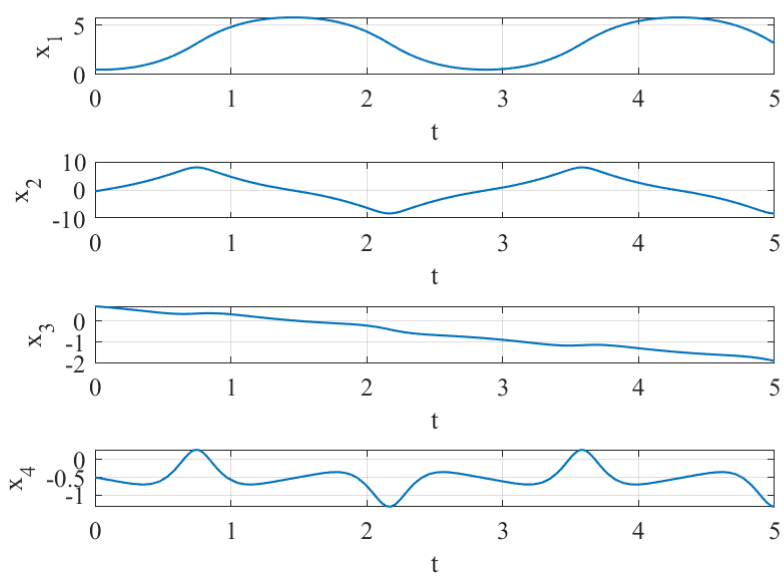
$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{g\sin(x_1) - mlax_2^2\sin(2x_1)/2}{4l/3 - mla\cos^2(x_1)} \\ x_4 \\ \frac{-mag\sin(2x_1)/2 + alx_2^2\sin(x_1)4m/3}{4/3 - ma\cos^2(x_1)}\end{bmatrix} + \begin{bmatrix} 0 \\ \frac{-a\cos(x_1)}{4l/3 - mla\cos^2(x_1)} \\ 0 \\ \frac{4a/3}{4/3 - ma\cos^2(x_1)}\end{bmatrix} u,$$

where

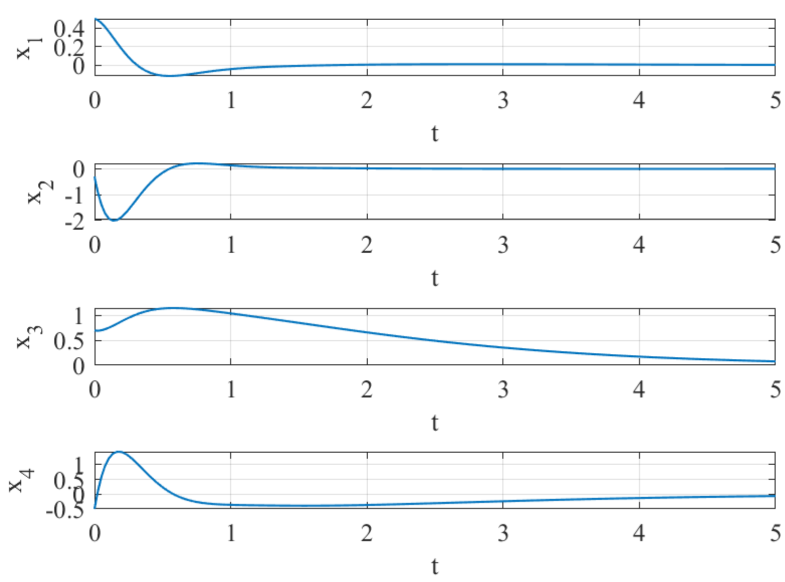
$$x_1 = \theta,\ x_2 = \dot{\theta},\ x_3 = x,\ \text{and}\ x_4 = \dot{x}.$$

The parameters are: $g = 9.8$ meter/sec^2, $m = 2$ kg, $M = 8$ kg, $a = 1/(m+M)$, and $l = 0.5$ meter.

We start with simulating the uncontrolled system behavior. In the figure below, we show plots of the state variables of the uncontrolled nonlinear model of the stick balancer.



We first design stabilizing controller using CVX. In the figure below, we show plots of the state variables of the controlled inverted pendulum on a cart with the fuzzy controller's gains obtained using CVX.



The controller's gains are:

$$K_1 = [-483.0468 \quad -100.5525 \quad -22.2813 \quad -52.1780] \text{ and } K_2 = [-609.5297 \quad -127.5252 \quad -28.2537$$

The weights are:

$$w_1(x_1) = \frac{1 - 1/\left(1 + \exp(-14(x_1 - \pi/8))\right)}{1 + \exp(-14(x_1 + \pi/8))} \quad \text{and} \quad w_2(x_1) = 1 - w_1(x_1).$$

The eigenvalues of $(A_1 - B_1K_1)$ are located at

$\{-4.9356+4.7990i,\ -4.9356-4.7990i,\ -0.8674+0.2465i,\ -0.8674-0.2465i\}$

.

The eigenvalues of $(A_2 - B_2K_2)$ are located at
$\{-2.6499+5.7163i,\ -2.6499-5.7163i,\ -0.6414+0.0000i,\ -1.5877+0.0000i\}$

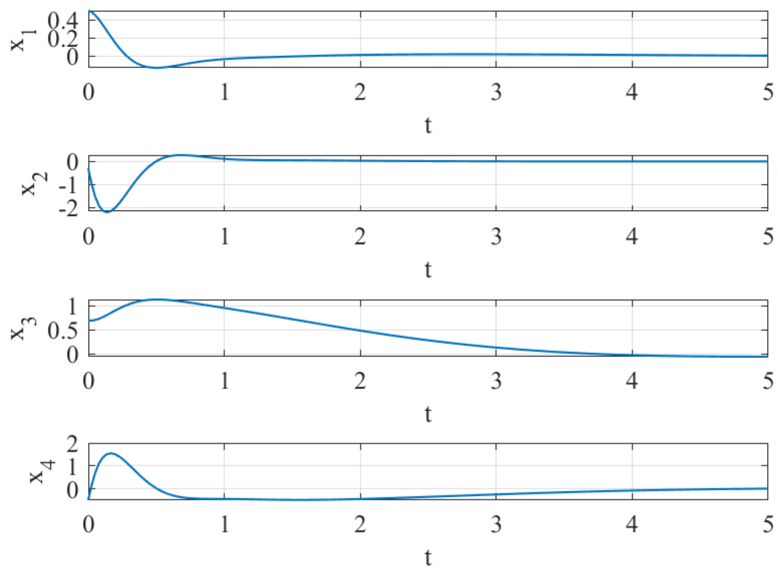
.

The common Lyapunov matrix P is

$$P = \begin{bmatrix} 1.0797 & 0.3043 & 0.0791 & 0.1747 \\ 0.3043 & 0.1208 & 0.0262 & 0.0720 \\ 0.0791 & 0.0262 & 0.0147 & 0.0169 \\ 0.1747 & 0.0720 & 0.0169 & 0.0512 \end{bmatrix}.$$

Its eigenvalues are located at $\{0.0051,\ 0.0084,\ 0.0479,\ 1.2050\}$.

We next design stabilizing controller using MATLAB's LMI toolbox. In the figure below, we show plots of the state variables of the controlled fuzzy system with controller gains obtained from CVX.



The controller's gains in this case are:

$$K_1 = \begin{bmatrix} -502.7021 & -92.4172 & -27.9490 & -45.8516 \end{bmatrix} \quad \text{and} \quad K_2 = \begin{bmatrix} -641.7226 & -119.1441 & -36.178 \end{bmatrix}$$

The eigenvalues of $(A_1 - B_1K_1)$ are located at
$\{-4.8114+5.6274i,\ -4.8114-5.6274i,\ -0.6459+0.6816i,\ -0.6459-0.6816i\}$

.

The eigenvalues of $(A_2 - B_2K_2)$ are located at
$\{-2.7817+6.0629i,\ -2.7817-6.0629i,\ -0.8722+0.6345i,\ -0.8722-0.6345i\}$

.

The common Lyapunov matrix P is

$$P = \begin{bmatrix} 0.0490 & 0.0118 & 0.0041 & 0.0064 \\ 0.0118 & 0.0042 & 0.0011 & 0.0024 \\ 0.0041 & 0.0011 & 0.0008 & 0.0007 \\ 0.0064 & 0.0024 & 0.0007 & 0.0017 \end{bmatrix}.$$

Its eigenvalues are located at $\{0.0002,\ 0.0004,\ 0.0019,\ 0.0532\}$.

The above results were obtained using the following script:

```
function[]=Module49_CaseStudy()
clear all
clc
x0=[0 0 0 0]';
syms x1 x2 x3 x4 u m a g l
f=[x2;
  (g*sin(x1)-m*l*a*x2^2*sin(2*x1)/2)/(4*l/3-m*l*a*cos(x1)^2);
   x4;
  (-m*a*g*sin(2*x1)/2 + a*l*x2^2*sin(x1)*(4*m/3))/(4/3-m*a*cos(x1)^2)];
G=[0;-a*cos(x1)/(4*l/3-m*l*a*cos(x1)^2);0;(4*a/3)/(4/3-m*a*cos(x1)^2)];
F=f+G*u;
DF=jacobian(F,[x1 x2 x3 x4]);
B1=subs(G,[x1 x2 x3 x4],[x0(1) x0(2) x0(3) x0(4)])
B1=eval(subs(B1,[m a g l],[2 0.1 9.8 0.5]))
A1=subs(DF,[x1 x2 x3 x4 u],[x0(1) x0(2) x0(3) x0(4) 0])
A1=eval(subs(A1,[m a g l],[2 0.1 9.8 0.5]))
%-------------------
x0=[pi/4 0 0 0]';
Df=jacobian(f,[x1 x2 x3 x4]);
Dfx0=subs(Df,[x1 x2 x3 x4],[x0(1) x0(2) x0(3) x0(4)]);
fx0=subs(f,[x1 x2 x3 x4],[x0(1) x0(2) x0(3) x0(4)]);
A2=simplify(collect(Dfx0+((fx0-Dfx0*x0)*x0')/norm(x0)^2))
A2=eval(subs(A2,[m a g l],[2 0.1 9.8 0.5]))
B2=[0;-a*cos(x1)/(4*l/3-m*l*a*cos(x1)^2);0;(4*a/3)/(4/3-m*a*cos(x1)^2)];
B2=eval(subs(B2,[m a g l],[2 0.1 9.8 0.5]))
B2=eval(subs(B2,[x1 x2 x3 x4],[x0(1) x0(2) x0(3) x0(4)]))

[n,~]=size(A1);
[~,m]=size(B1);

% Uncomment from cvx_begin through cvx_end and comment from setlmis through Z2 to
% compute the controller"s gains using cvx

% cvx_begin sdp
% % Variable definition
% variable S(n, n) symmetric
% variable Z1(m, n)
% variable Z2(m, n)
% % LMIs
% A1*S + S*A1' - B1*Z1 - Z1'*B1'  <= 0
% A2*S + S*A2' - B2*Z2 - Z2'*B2'  <= 0
% (A1+A2)*S+S*(A1+A2)'-Z2'*B1'-Z1'*B2'-B1*Z2-B2*Z1 <=0
% S >= eps*eye(n)
% cvx_end
setlmis([]);
S=lmivar(1,[n,1]);
Z1=lmivar(2,[m,n]);
Z2=lmivar(2,[m,n]);
lmiterm([1 1 1 S],A1,1,'s');
lmiterm([1 1 1 Z1],-B1,1,'s');
lmiterm([2 1 1 S],A2,1,'s');
lmiterm([2 1 1 Z2],-B2,1,'s');
lmiterm([3 1 1 S],A1+A2,1,'s');
lmiterm([3 1 1 Z1],-B2,1,'s');
lmiterm([3 1 1 Z2],-B1,1,'s');
lmiterm([-4 1 1 S],1,1);
lmiterm([4 1 1 0],0.01);
lmis=getlmis;
[tmin,xfeas]=feasp(lmis);
S=dec2mat(lmis,xfeas,S);
Z1=dec2mat(lmis,xfeas,Z1);
Z2=dec2mat(lmis,xfeas,Z2);
disp('K1 and K2')
global K1 K2
 K1 = Z1/S % compute K1 matrix
 K2 = Z2/S % compute K2 matrix
 disp('P')
P=inv(S)
disp('eig(P)')
eig(inv(S))
disp('eig(A1-B1*K1)')
eig(A1-B1*K1)
disp('eig(A2-B2*K2)')
eig(A2-B2*K2)

x_0=[.5 -.3 .7 -.5]';
tspan = [0 5];
[t, x] = ode23(@model, tspan, x_0);

figure;
for i=1:n
    subplot(4,1,i);
    plot(t, x(:,i), 'linewidth', 1);
    xlabel('t', 'fontsize', 12);
    ylabel(['x_', num2str(i)],'fontsize',12);
    set(gca, 'fontsize', 12,'FontName','Times');
    grid
end

function xdot = model(t, x)
global K1 K2
% Control input
```

```
w1=(1-1/(1+exp(-14*(x(1)-pi/8))))/(1+exp(-14*(x(1)+pi/8)));
w2=1-w1;
u = -(w1*K1+w2*K2)*x;

% u=0; % Uncomment u=0 to simulate the uncontrolled system behavior

% System
f=[x(2);
  (9.8*sin(x(1))-0.1*x(2)^2*sin(2*x(1))/2)/(2/3-0.1*cos(x(1))^2);
   x(4);
  (-0.2*9.8*sin(2*x(1))/2 + 0.05*x(2)^2*sin(x(1))*(8/3))/(4/3-0.2*cos(x(1))^2)];
g=[0;-0.1*cos(x(1))/(2/3-0.1*cos(x(1))^2);0;(2/15)/(4/3-0.2*cos(x(1))^2)];
xdot = f + g*u;
```

PURDUE UNIVERSITY.