EECS 562 & AEROSP 551
Inverted Pendulum on a Cart Project

Due date: Tuesday, February 16, 2023 by 23:59 EST on Canvas
Firm Deadline: No extensions will be allowed.

Swayampakula, Rahul Kashyap
LAST NAME (PRINTED), *FIRST NAME*

HONOR PLEDGE: Copy and SIGN: I understand that this is an individual assignment and not a group exercise. I have neither given nor received aid on this project nor have I concealed any violation of the Honor Code. I have neither borrowed nor shared computer code for simulating my models and creating my plots.

s.rahul
SIGNATURE

This Should Be The Cover Page Of Your Project

General: The goal of this INDIVIDUAL project is to analyze a particular physical system, the so-called "broom balancing problem", from a linearization point of view. That is, you will design a controller for a nonlinear system using linear feedback design methods. This will give you a chance to use some of what you learned in EECS 560, plus maybe a few other techniques as well. More importantly, it will illustrate a very common design technique for nonlinear systems: linearize, then hope for the best! We will learn more analytical methods during the term and these will be illustrated in HW on this model. To complete the project, you may have to read a few things in the textbook before we cover them in class. Please be assured that students in previous terms have accomplished this successfully and you will too.

Honor Code: You are to do your own work. Discussing the project with a friend is fine. Sharing MATLAB code is not allowed.

Getting the model, Step 1: Consider the inverted pendulum positioning system described in the attached scanned-sheets. Assuming that friction is negligible, but not assuming that the mass of the rod is small in comparison to the mass of the cart, equations (1-15)-(1-18) of the attached sheets can be shown to yield:

$$\begin{aligned} \frac{d^2\phi}{dt^2} &= \frac{(M+m)mgL\sin(\phi) - (mL\dot{\phi})^2\sin(\phi)\cos(\phi) - (mL\cos(\phi))\mu}{(M+m)(J+mL^2) - (mL)^2\cos^2(\phi)} \\ \frac{d^2 s}{dt^2} &= \frac{(J+mL^2)mL\dot{\phi}^2\sin(\phi) - (mL)^2 g\sin(\phi)\cos(\phi) + (J+mL^2)\mu}{(M+m)(J+mL^2) - (mL)^2\cos^2(\phi)} \end{aligned} \tag{1}$$

where, here, the dots represent differentiation with respect to the physical time t. Indeed, you may wish to do the tedious derivation for your own pleasure. But this is NOT required. The method of Lagrange can be used to get the model in a much easier and quicker way. I have done this symbolically and verified the above equations. So, you can assume that they are correct and free of typos. [see the file: symb_model_inv_pend_cart.m in the Project folder.]

Getting the model, Step 2: The concept of dimensionless variables is very important in engineering practice because it often leads to models that are numerically better conditioned, and, in addition, it often leads to fewer unknown parameters (as is the case in our example below). It is becoming a lost art in engineering schools. I learned about the subject in Thermodynamics; how about you? Introduce the dimensionless variables:

$$\begin{array}{lll} \bar{s} = (\frac{M}{m}+1)(\frac{s}{L}) & \bar{\mu} = \frac{1}{(M+m)g}\mu & d = 1 + c \\ \bar{t} = \frac{t}{T}, & \text{where} & T^2 = (\frac{J}{mL^2} + \frac{M}{m+M})\frac{L}{g} \\ b = (\frac{M}{m}+1)(\frac{J}{mL^2}+1) & c = \frac{m^2L^2}{J(m+M)+mML^2} & \end{array} \tag{2}$$

Using the dimensionless variables defined above, the model dynamics from (1) become:

$$\frac{d^2\phi}{d\bar{t}^2} = \frac{-c\dot{\phi}^2 \sin(\phi)\cos(\phi)}{1 + c\sin^2(\phi)} + \frac{\sin(\phi)}{1 + c\sin^2(\phi)} - \frac{\cos(\phi)}{1 + c\sin^2(\phi)}\bar{\mu}$$
$$\frac{d^2\bar{s}}{d\bar{t}^2} = \frac{d \cdot \dot{\phi}^2 \sin(\phi)}{1 + c\sin^2(\phi)} - \frac{\cos(\phi)\sin(\phi)}{1 + c\sin^2(\phi)} + \frac{b}{1 + c\sin^2(\phi)}\bar{\mu} \tag{3}$$

where the dots now represent differentiation with respect to the normalized time $\bar{t}$.

Remark: You may wish to verify that the denominator in the model never vanishes, so that the equations are well defined. Do not include this in your solutions.

Problem 1: Verify the Dimensionless Model (10 points): Verify that the equation for ϕ in (3) indeed follows from (1) after substitution of the given dimensionless quantities and proper application of the chain rule. (NOTE: Doing the same for s is similar and is not required here.)

Hint: First note that $\bar{t} = \frac{t}{T}$ implies that $t = T\bar{t}$. Hence, by the chain rule

ok

$$\frac{d\phi(\bar{t})}{d\bar{t}} = \left.\frac{d\phi(t)}{dt}\right|_{t=T\bar{t}} \frac{dt}{d\bar{t}},$$

where of course, $\frac{dt}{d\bar{t}} = T$. You have to figure out the second derivative on your own. No help on this part in office hours!

Problem 2: Linear State-Variable Model (10 points): Construct a state-variable representation of (3) and linearize the equations about the upright equilibrium point. Define the state vector $x \triangleq [x_1 \;\; x_2 \;\; x_3 \;\; x_4]^T$ in the following manner: $x_1 = \phi$, $x_2 = \dot{\phi}$, $x_3 = \bar{s}$, $x_4 = \dot{\bar{s}}$. For the remainder of this project, the following data is assumed:

$$M = 25 \; (kg) \qquad m = 20 \; (kg) \quad L = 9.81 \; (m)$$
$$g = 9.81 \; (m/s^2) \quad J = \tfrac{1}{3}mL^2 \tag{4}$$

doable

1. Write down the dynamics for state-vector x in the form $\dot{x} = f(x, \bar{\mu})$.
2. Linearizing the system in part 1 about the origin should yield the lineraized system $\dot{x} = Ax + B\bar{\mu}$. Find matrices A and B.

Problem 3: Stabilizing Feedback (10 points): Verify that the linearized system is completely controllable and then determine a state variable feedback control law $\bar{\mu} = Kx$

ok

that will place the closed-loop eigenvalues at $(-3, -2, -0.7 \pm j0.2)$. You may use the `place` command in MATLAB.

Problem 4: Stability with Full State Feedback (10 points): Apply the control law $\bar{\mu} = Kx$ to both the linearized system and nonlinear system. This gives you two closed-loop systems:

$$\dot{x} = (A + BK)x \tag{5}$$

ok

and

$$\dot{x} = f(x, Kx) \tag{6}$$

Verify that $A + BK = \frac{\partial}{\partial x} f(x, Kx)|_{x=0}$, that is (5) is the linearization of (6) about the origin. What can you conclude about the stability properties of the origin for the nonlinear closed-loop system? See Theorem 4.7 on page 139 of Khalil. See also Definition 4.1 on page 112 of our textbook.

Problem 5: Simulation with Full State Feedback (10 points): Simulate both closed-loop systems for several initial conditions, comparing the results of the linearized model to the nonlinear model. With $x_2(0) = 0$, $x_3(0) = 0$, and $x_4(0) = 0$, explore how large you can make $x_1(0)$ before the nonlinear model goes unstable.

Problem 6: Luenberger Observer Design (15 points): Suppose now that you can only measure the outputs $y_1 = \phi$ and $y_2 = \bar{s}$. Write the output equation $y = Cx$. Verify that the pair (A, C) is completely observable and then design an observer to implement the control law synthesized in Problem 3. This means that you have to design an observer-based design control law using the following steps:

1. Design an asymptotically stabvle observer. You may use pole-placement to design the observer. Your linear observer should have the form:

   needs check...mostly place cmd works

   $$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \\ u &= K\hat{x} \end{aligned} \tag{7}$$

   You recognize the observer as being a copy of the linearized model plus an "output injection" term $L(y = \hat{y})$, which adjusts the state estimates as a function of errors in the estimate of the output.

2. Use the control law $\bar{\mu} = K\hat{x}$, where K is the control gain designed in Problem 3.

Note that this is a more difficult pole placement problem than the one you did in Problem 3 because there are two outputs instead of just one; hence there are many ways of achieving the same pole positions. If you have had EECS 565, you may use methods from that course to design the observer gain; otherwise, use once again the `place` command.

Problem 7: Simulation Results for Observer Based Compensator (15 Points): Simulate the observer-based control law (also known as observer-based compensator or dynamic controller) as obtained in Problem 6 on both the linearized and nonlinear system models for a variety of initial conditions. Plot the outputs $y_1(t)$ and $y_2(t)$. How do the responses between the static and observer-based compensator(also known as dynamic controller) compare? Always take the initial conditions of the observer to be $[x_1(0), 0, 0, 0]$. Explore how large you can make $x_1(0)$ (the initial angle of the pendulum) before the nonlinear model goes unstable. Generally speaking, it should be smaller than in Problem 5, though in some cases, one may find the opposite. Do not worry about this.

Problem 8: Observer Based Compensator using Nonlinear Model (20 points) For the nonlinear plant, consider implementing the observer as

$$\begin{aligned} \dot{\hat{x}} &= f(\hat{x}, u) + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \\ u &= K\hat{x} \end{aligned} \tag{8}$$

which you recognize as being a copy of the nonlinear model plus linear output injection. See if this observer provides better estimates than (**??**) and hence better performance, for the nonlinear system. Why should the origin of the closed-loop nonlinear system still be asymptotically stable? **Hint:** linearize the closed-loop system (8) about the origin and compare to (7).

Write-up for Inverted Pedulum Project
(Required Report Format)

You will have many pages of simulations, computations, etc. While I want most of this included in your report, I wish to receive an organized package that is easy to read and follow because I have to go through all of them. I will deduct points for poorly presented work.

Page 1: Use the required title page.

Page 2: : Contents page. List Problems one through eight and include page numbers.

Page 3: : Provide answers to the following questions:

A Using linear state variable feedback and with the nonlinear model initialized at $x_2(0) = 0$, $x_3(0) = 0$, $x_4(0) = 0$, the largest I could make $x_1(0)$ before the closed-loop system went unstable was _______(radians). The linear state variable feedback that I used was _________________ (give the feedback gains, upto 2 digits after the decimal point is sufficient for all answers.).

B Using linear state variable feedback plus an observer, with the observer initialized at the origin and the nonlinear model initialized at $x_2(0) = 0$, $x_3(0) = 0$, $x_4(0) = 0$, the largest I could make $x_1(0)$ before the closed-loop system went unstable was _______(radians). The observer that I used to obtain the best response was linear/nonlinear (specify which one) and the observer gain was _________________ (give the gain matrix). Put all documentation in Problem 7 or 8, as appropriate.

Page 4: Begin answers to problems one through eight.

- Start each problem on a separate page, starting the page with a statement of the problem that is being worked. You can use both sides of a page if you wish.
- DO NOT group all simulations, MATLAB printouts or hand computations at the end as one giant appendix. Instead, group them with the problem that is being worked. That is, all simulations for problem five will follow problem five, those for problem seven will be with problem seven, etc.
- <u>Only include the most important of your printouts.</u>
- Include your MATLAB code, even the code used to generate your plots.
- Put labels or numbers on your plots. You can number thwm by hand or using MATLAB; either is fine for me.
- You do NOT have to typeset your solutions. Handwriting is perfectly fine.
- The submission is via Canvas. You only have to upload ONE .zip file, including all the codes and the report. Label the zip file as "uniquename_project_W23.zip".

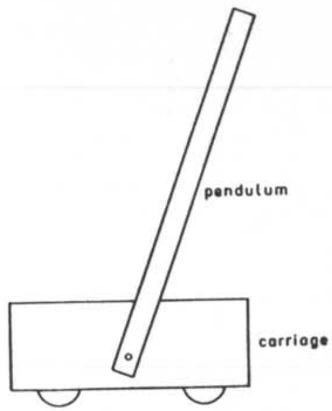


Fig. 1.1. An inverted pendulum positioning system.

Example 1.1. *Inverted pendulum positioning system.*

Consider the inverted pendulum of Figure 1.1 (see also, for this example, Cannon, 1967; Elgerd, 1967). The pivot of the pendulum is mounted on a carriage which can move in a horizontal direction. The carriage is driven by a small motor that at time t exerts a force $\mu(t)$ on the carriage. This force is the input variable to the system.

Figure 1.2 indicates the forces and the displacements. The displacement of the pivot at time t is $s(t)$, while the angular rotation at time t of the pendulum is $\phi(t)$. The mass of the pendulum is m, the distance from the pivot to the center of gravity L, and the moment of inertia with respect to the center of gravity J. The carriage has mass M. The forces exerted on the pendulum are

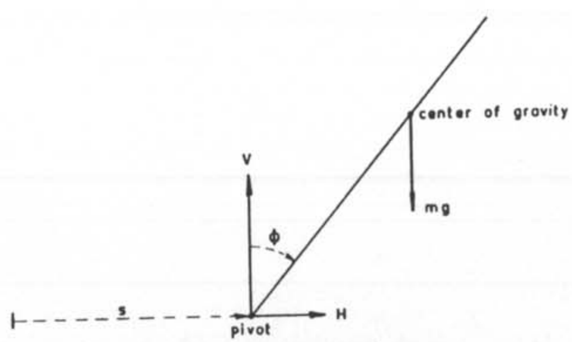


Fig. 1.2. Inverted pendulum: forces and displacements.

the force mg in the center of gravity, a horizontal reaction force $H(t)$, and a vertical reaction force $V(t)$ in the pivot. Here g is the gravitational acceleration. The following equations hold for the system:

$$m \frac{d^2}{dt^2}[s(t) + L \sin \phi(t)] = H(t), \qquad \textbf{1-11}$$

$$m \frac{d^2}{dt^2}[L \cos \phi(t)] = V(t) - mg, \qquad \textbf{1-12}$$

$$J \frac{d^2\phi(t)}{dt^2} = LV(t) \sin \phi(t) - LH(t) \cos \phi(t), \qquad \textbf{1-13}$$

$$M \frac{d^2 s(t)}{dt^2} = \mu(t) - H(t) - F \frac{ds(t)}{dt}. \qquad \textbf{1-14}$$

Friction is accounted for only in the motion of the carriage and not at the pivot; in **1-14**, F represents the friction coefficient. Performing the differentiations indicated in **1-11** and **1-12**, we obtain

$$m\ddot{s}(t) + mL\ddot{\phi}(t) \cos \phi(t) - mL\dot{\phi}^2(t) \sin \phi(t) = H(t), \qquad \textbf{1-15}$$

$$-mL\ddot{\phi}(t) \sin \phi(t) - mL\dot{\phi}^2(t) \cos \phi(t) = V(t) - mg, \qquad \textbf{1-16}$$

$$J\ddot{\phi}(t) = LV(t) \sin \phi(t) - LH(t) \cos \phi(t), \qquad \textbf{1-17}$$

$$M\ddot{s}(t) = \mu(t) - H(t) - F\dot{s}(t). \qquad \textbf{1-18}$$

To simplify the equations we assume that m is small with respect to M and therefore neglect the horizontal reaction force $H(t)$ on the motion of the carriage. This allows us to replace **1-18** with

$$M\ddot{s}(t) = \mu(t) - F\dot{s}(t). \qquad \textbf{1-19}$$

Elimination of $H(t)$ and $V(t)$ from **1-15**, **1-16**, and **1-17** yields

$$(J + mL^2)\ddot{\phi}(t) - mgL \sin \phi(t) + mL\ddot{s}(t) \cos \phi(t) = 0. \qquad \textbf{1-20}$$

Division of this equation by $J + mL^2$ yields

$$\ddot{\phi}(t) - \frac{g}{L'} \sin \phi(t) + \frac{1}{L'} \ddot{s}(t) \cos \phi(t) = 0, \qquad \textbf{1-21}$$

where

$$L' = \frac{J + mL^2}{mL}. \qquad \textbf{1-22}$$

7

This quantity has the significance of "effective pendulum length" since a mathematical pendulum of length L' would also yield **1-21**.

Let us choose as the nominal solution $\mu(t) \equiv 0$, $s(t) \equiv 0$, $\phi(t) \equiv 0$. Linearization can easily be performed by using Taylor series expansions for $\sin \phi(t)$ and $\cos \phi(t)$ in **1-21** and retaining only the first term of the series. This yields the linearized version of **1-21**:

$$\ddot{\phi}(t) - \frac{g}{L'}\phi(t) + \frac{1}{L'}\ddot{s}(t) = 0. \qquad \textbf{1-23}$$

We choose the components of the state $x(t)$ as

$$\begin{aligned} \xi_1(t) &= s(t), \\ \xi_2(t) &= \dot{s}(t), \\ \xi_3(t) &= s(t) + L'\phi(t), \\ \xi_4(t) &= \dot{s}(t) + L'\dot{\phi}(t). \end{aligned} \qquad \textbf{1-24}$$

The third component of the state represents a linearized approximation to the displacement of a point of the pendulum at a distance L' from the pivot. We refer to $\xi_3(t)$ as the displacement of the pendulum. With these definitions we find from **1-19** and **1-23** the linearized state differential equation

$$\begin{aligned} \dot{\xi}_1(t) &= \xi_2(t), \\ \dot{\xi}_2(t) &= \frac{1}{M}\mu(t) - \frac{F}{M}\xi_2(t), \\ \dot{\xi}_3(t) &= \xi_4(t), \\ \dot{\xi}_4(t) &= g\phi(t) = \frac{g}{L'}[\xi_3(t) - \xi_1(t)]. \end{aligned} \qquad \textbf{1-25}$$

In vector notation we write

$$\dot{x}(t) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{F}{M} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{g}{L'} & 0 & \frac{g}{L'} & 0 \end{pmatrix} x(t) + \begin{pmatrix} 0 \\ \frac{1}{M} \\ 0 \\ 0 \end{pmatrix} \mu(t), \qquad \textbf{1-26}$$

where $x(t) = \text{col}\,[\xi_1(t), \xi_2(t), \xi_3(t), \xi_4(t)]$.

Later the following numerical values are used:

$$\begin{aligned} \frac{F}{M} &= 1\ \text{s}^{-1}, \\ \frac{1}{M} &= 1\ \text{kg}^{-1}, \\ \frac{g}{L'} &= 11.65\ \text{s}^{-2}, \\ L' &= 0.842\ \text{m}. \end{aligned} \qquad \textbf{1-27}$$

Example 1.2. *A stirred tank.*

As a further example we treat a system that is to some extent typical of process control systems. Consider the stirred tank of Fig. 1.3. The tank is fed

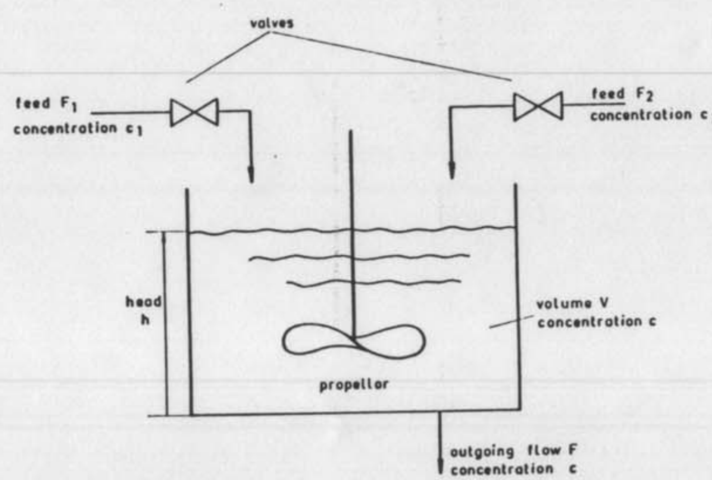


Fig. 1.3. A stirred tank.

with two incoming flows with time-varying flow rates $F_1(t)$ and $F_2(t)$. Both feeds contain dissolved material with constant concentrations c_1 and c_2, respectively. The outgoing flow has a flow rate $F(t)$. It is assumed that the tank is stirred well so that the concentration of the outgoing flow equals the concentration $c(t)$ in the tank.

```
% symb_model_inv_pend_cart.m
%
clear *
%
% This is for the EECS 562 project of an inverted pendulum on a cart
%
syms s phi ds dphi real
syms g L m M J real
%
% reference = absolute angle for the pendulum, measured clockwise from the vertical
% referebce = position of cart measured from left to right
%
q=[phi s].';
dq=[dphi ds].';
%
%
pcart=[s;0];
ppend_centermass=pcart+L*[sin(phi);cos(phi)];
%
%
vcart =  jacobian(pcart,q)*dq;
vpend_centermass = jacobian(ppend_centermass,q)*dq;
%
%
```

One way to get the model using Lagrange's method.
You can ignore this ☺

```
KEcart = simplify(M*(1/2)*vcart.'*vcart)
KEpend_centermass = simplify(m*(1/2)*vpend_centermass.'*vpend_centermass)
KErotation=simplify(J*(1/2)*(dphi)^2)
%
%
KE = (KEcart+KEpend_centermass+KErotation);
KE = simple(KE)
%
%
%
PE = g*m*ppend_centermass(2);
PE = simple(PE);
%
%
% Model NOTATION: Spong and Vidyasagar, page 142, Eq. (6.3.12)
%                 D(q)ddq + C(q,dq)*dq + G(q) = B*tau + F_external
%
```

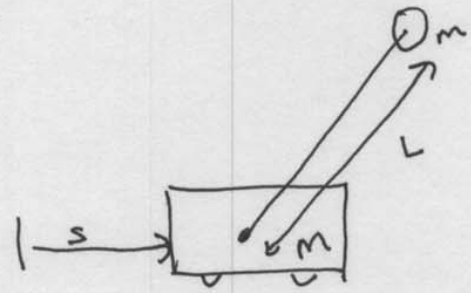
Roboticists like this form

```
L=KE-PE;

G=jacobian(PE,q).';
G=simple(G);
D=simple(jacobian(KE,dq).');
D=simple(jacobian(D,dq));

syms C real
n=max(size(q));
for k=1:n
   for j=1:n
      C(k,j)=0*g;
      for i=1:n
         C(k,j)=C(k,j)+(1/2)*(diff(D(k,j),q(i))+diff(D(k,i),q(j))-
diff(D(i,j),q(k)))*dq(i);
      end
   end
end
C=simple(C);
%
% Compute the matrix for the input force
%
B=[0;1];
%
%
% Compute D^{-1}
%
```

```
d=simple(det(D));
DI=inv(D);
DI=simple(DI);
%
%
% compute RHS of model in the form ddq = f + g mu
f=DI*(-C*dq-G); f=simple(f)
g=DI*B; g=simple(g)
pretty(f(1))
pretty(f(2))
pretty(g(1))
pretty(g(2))
save work_symb_model_inv_pend_cart
return
```

More useful for us

Note: If you set $J=0$, then you have



the classic 'bob' on a rod.