



SYSTEM	◀
MODELING	
ANALYSIS	
<hr/>	
CONTROL	◀
PID	
ROOT LOCUS	
FREQUENCY	
STATE-SPACE	
DIGITAL	
<hr/>	
SIMULINK	◀
MODELING	
CONTROL	
SIMSCAPE	

Inverted Pendulum: System Modeling

Key MATLAB commands used in this tutorial are: [tf](#) ,
[ss](#) , [set](#)

Related Tutorial Links

- [Intro](#) to [Modeling](#)
- [Pendulum](#) Activity

Related External Links

- [System](#) Rep in [MATLAB](#) Video
- [Modeling](#) Intro Video

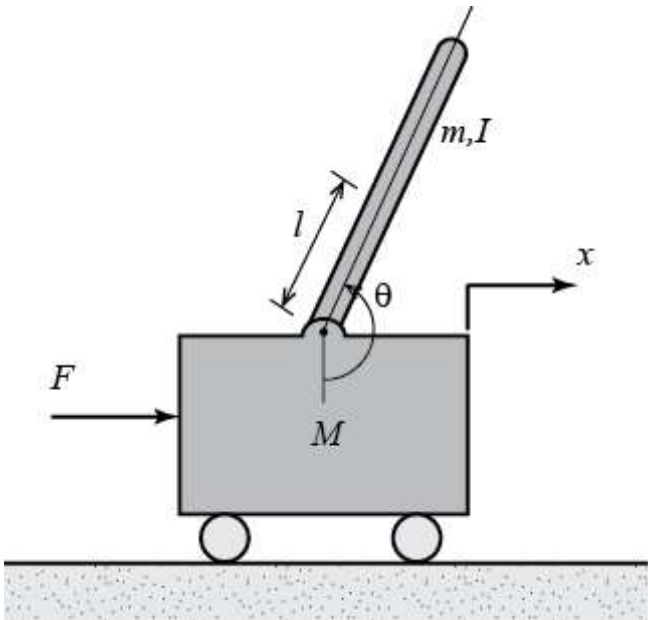
Contents

- [Problem setup and design requirements](#)
- [Force analysis and system equations](#)
- [MATLAB representation](#)

Problem setup and design requirements

The system in this example consists of an inverted pendulum mounted to a motorized cart. The inverted pendulum system is an example commonly found in control system textbooks and research literature. Its popularity derives in part from the fact that it is unstable without control, that is, the pendulum will simply fall over if the cart isn't moved to balance it. Additionally, the dynamics of the system are nonlinear. The objective of the control system is to balance the inverted pendulum by applying a force to the cart that the pendulum is attached to. A real-world example that relates directly to this inverted pendulum system is the attitude control of a booster rocket at takeoff.

In this case we will consider a two-dimensional problem where the pendulum is constrained to move in the vertical plane shown in the figure below. For this system, the control input is the force F that moves the cart horizontally and the outputs are the angular position of the pendulum θ and the horizontal position of the cart x .



For this example, let's assume the following quantities:

(M)	mass of the cart	0.5 kg
(m)	mass of the pendulum	0.2 kg
(b)	coefficient of friction for cart	0.1 N/m/sec
(l)	length to pendulum center of mass	0.3 m
(I)	mass moment of inertia of the pendulum	0.006 kg.m ²
(F)	force applied to the cart	
(x)	cart position coordinate	

(theta) pendulum angle from vertical (down)

For the PID, root locus, and frequency response sections of this problem, we will be interested only in the control of the pendulum's position. This is because the techniques used in these sections are best-suited for single-input, single-output (SISO) systems. Therefore, none of the design criteria deal with the cart's position. We will, however, investigate the controller's effect on the cart's position after the controller has been designed. For these sections, we will design a controller to restore the pendulum to a vertically upward position after it has experienced an impulsive "bump" to the cart. Specifically, the design criteria are that the pendulum return to its upright position within 5 seconds and that the pendulum never move more than 0.05 radians away from vertical after being disturbed by an impulse of magnitude 1 Nsec. The pendulum will initially begin in the vertically upward equilibrium, $\theta = \pi$.

In summary, the design requirements for this system are:

- Settling time for θ of less than 5 seconds
- Pendulum angle θ never more than 0.05 radians from the vertical

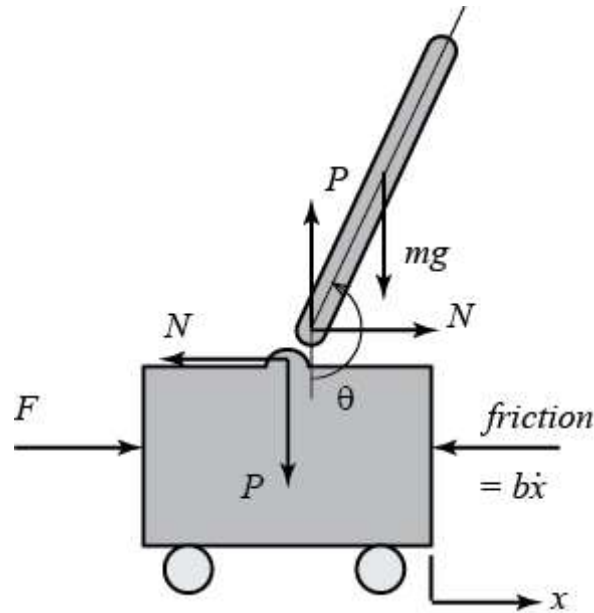
Employing state-space design techniques, we are more readily able to address a multi-output system. In our case, the inverted pendulum system is single-input, multi-output (SIMO). Therefore, for the state-space section of the Inverted Pendulum example, we will attempt to control both the pendulum's angle and the cart's position. To make the design more challenging in this section, we will command a 0.2-meter step in the cart's desired position. Under these conditions, it is desired that the cart achieve its commanded position within 5 seconds and have a rise time under 0.5 seconds. It is also desired that the pendulum settle to its vertical position in under 5 seconds, and further, that the pendulum angle not travel more than 20 degrees (0.35 radians) away from the vertically upward position.

In summary, the design requirements for the inverted pendulum state-space example are:

- Settling time for x and θ of less than 5 seconds
- Rise time for x of less than 0.5 seconds
- Pendulum angle θ never more than 20 degrees (0.35 radians) from the vertical
- Steady-state error of less than 2% for x and θ

Force analysis and system equations

Below are the free-body diagrams of the two elements of the inverted pendulum system.



Summing the forces in the free-body diagram of the cart in the horizontal direction, you get the following equation of motion.

$$M\ddot{x} + b\dot{x} + N = F \quad (1)$$

Note that you can also sum the forces in the vertical direction for the cart, but no useful information would be gained.

Summing the forces in the free-body diagram of the pendulum in the horizontal direction, you get the following expression for the reaction force N .

$$N = m\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta \quad (2)$$

If you substitute this equation into the first equation, you get one of the two governing equations for this system.

$$(M + m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F \quad (3)$$

To get the second equation of motion for this system, sum the forces perpendicular to the pendulum. Solving the system along this axis greatly simplifies the mathematics. You should get the following equation.

$$P\sin\theta + N\cos\theta - mg\sin\theta = ml\ddot{\theta} + m\ddot{x}\cos\theta \quad (4)$$

To get rid of the P and N terms in the equation above, sum the moments about the centroid of the pendulum to get the following equation.

$$-Pl\sin\theta - Nl\cos\theta = I\ddot{\theta} \quad (5)$$

Combining these last two expressions, you get the second governing equation.

$$(I + ml^2)\ddot{\theta} + mgl\sin\theta = -ml\ddot{x}\cos\theta \quad (6)$$

Since the analysis and control design techniques we will be employing in this example apply only to linear systems, this set of equations needs to be linearized. Specifically, we will linearize the equations about the vertically upward equilibrium position, $\theta = \pi$, and will assume that the system stays within a small neighborhood of this equilibrium. This assumption should be reasonably valid since under control we desire that the pendulum not deviate more than 20 degrees from the vertically upward position. Let ϕ represent the deviation of the pendulum's position from equilibrium, that is, $\theta = \pi + \phi$. Again presuming a small deviation (ϕ) from equilibrium, we can use the following small angle approximations of the nonlinear functions in our system equations:

$$\cos \theta = \cos(\pi + \phi) \approx -1 \quad (7)$$

$$\sin \theta = \sin(\pi + \phi) \approx -\phi \quad (8)$$

$$\dot{\theta}^2 = \dot{\phi}^2 \approx 0 \quad (9)$$

After substituting the above approximations into our nonlinear governing equations, we arrive at the two linearized equations of motion. Note u has been substituted for the input F .

$$(I + ml^2)\ddot{\phi} - mgl\phi = ml\ddot{x} \quad (10)$$

$$(M + m)\ddot{x} + b\dot{x} - ml\ddot{\phi} = u \quad (11)$$

1. Transfer Function

To obtain the transfer functions of the linearized system equations, we must first take the Laplace transform of the system equations assuming zero initial conditions. The resulting Laplace transforms are shown below.

$$(I + ml^2)\Phi(s)s^2 - mgl\Phi(s) = mlX(s)s^2 \quad (12)$$

$$(M + m)X(s)s^2 + bX(s)s - ml\Phi(s)s^2 = U(s) \quad (13)$$

Recall that a transfer function represents the relationship between a single input and a single output at a time. To find our first transfer function for the output $\Phi(s)$ and an input of $U(s)$ we need to eliminate $X(s)$ from the above equations. Solve the first equation for $X(s)$.

$$X(s) = \left[\frac{I + ml^2}{ml} - \frac{g}{s^2} \right] \Phi(s) \quad (14)$$

Then substitute the above into the second equation.

$$(M + m) \left[\frac{I + ml^2}{ml} - \frac{g}{s^2} \right] \Phi(s)s^2 + b \left[\frac{I + ml^2}{ml} - \frac{g}{s^2} \right] \Phi(s)s - ml\Phi(s)s^2 = U(s) \quad (15)$$

Rearranging, the transfer function is then the following

$$\frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s^2}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mgl}{q}s^2 - \frac{bmgl}{q}s} \quad (16)$$

where,

$$q = [(M+m)(I+ml^2) - (ml)^2] \quad (17)$$

From the transfer function above it can be seen that there is both a pole and a zero at the origin. These can be canceled and the transfer function becomes the following.

$$P_{pend}(s) = \frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b(I+ml^2)}{q}s^2 - \frac{(M+m)mgl}{q}s - \frac{bmgl}{q}} \quad \left[\frac{rad}{N}\right] \quad (18)$$

Second, the transfer function with the cart position $X(s)$ as the output can be derived in a similar manner to arrive at the following.

$$P_{cart}(s) = \frac{X(s)}{U(s)} = \frac{\frac{(I+ml^2)s^2 - gml}{q}}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mgl}{q}s^2 - \frac{bmgl}{q}s} \quad \left[\frac{m}{N}\right] \quad (19)$$

2. State-Space

The linearized equations of motion from above can also be represented in state-space form if they are rearranged into a series of first order differential equations. Since the equations are linear, they can then be put into the standard matrix form shown below.

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-(I+ml^2)b}{I(M+m)+Mml^2} & \frac{m^2gl^2}{I(M+m)+Mml^2} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{-mlb}{I(M+m)+Mml^2} & \frac{mgl(M+m)}{I(M+m)+Mml^2} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{I+ml^2}{I(M+m)+Mml^2} \\ 0 \\ \frac{ml}{I(M+m)+Mml^2} \end{bmatrix} u \quad (20)$$

$$\mathbf{y} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \quad (21)$$

The \mathbf{C} matrix has 2 rows because both the cart's position and the pendulum's position are part of the output. Specifically, the cart's position is the first element of the output \mathbf{Y} and the pendulum's deviation from its equilibrium position is the second element of \mathbf{Y} .

MATLAB representation

1. Transfer Function

We can represent the transfer functions derived above for the inverted pendulum system within MATLAB employing the following commands. Note

that you can give names to the outputs (and inputs) to differentiate between the cart's position and the pendulum's position. Running this code in the command window produces the output shown below.

```
M = 0.5;
m = 0.2;
b = 0.1;
I = 0.006;
g = 9.8;
l = 0.3;
q = (M+m)*(I+m*l^2)-(m*l)^2;
s = tf('s');

P_cart = (((I+m*l^2)/q)*s^2 - (m*g*l/q))/(s^4 + (b*(I + m*l^2))*s^3/q - (M+m)*m*g*l/q);

P_pend = (m*l*s/q)/(s^3 + (b*(I + m*l^2))*s^2/q - ((M + m)*m*g*l)*s/q - l);

sys_tf = [P_cart ; P_pend];

inputs = {'u'};
outputs = {'x'; 'phi'};

set(sys_tf, 'InputName', inputs)
set(sys_tf, 'OutputName', outputs)

sys_tf
```

```
sys_tf =
```

```
From input "u" to output...
```

$$4.182e-06 \, s^2 - 0.0001025$$

```
x: -----
```

$$2.3e-06 \, s^4 + 4.182e-07 \, s^3 - 7.172e-05 \, s^2 - 1.025e-05 \, s$$

$$1.045e-05 \, s$$

```
phi: -----
```

$$2.3e-06 \, s^3 + 4.182e-07 \, s^2 - 7.172e-05 \, s - 1.025e-05$$

```
Continuous-time transfer function.
```

2. State-Space

We can also represent the system using the state-space equations. The following additional MATLAB commands create a state-space model of the inverted pendulum and produce the output shown below when run in the MATLAB command window. Again note that the names of the inputs, outputs, and states can be specified to make the model easier to understand.

```
M = .5;
m = 0.2;
b = 0.1;
I = 0.006;
g = 9.8;
l = 0.3;

p = I*(M+m)+M*m*l^2; %denominator for the A and B matrices

A = [0      1      0      0;
      0 -(I+m*l^2)*b/p (m^2*g*l^2)/p 0;
      0      0      0      1;
      0 -(m*l*b)/p      m*g*l*(M+m)/p 0];
B = [      0;
      (I+m*l^2)/p;
      0;
      m*l/p];
C = [1 0 0 0;
      0 0 1 0];
D = [0;
      0];

states = {'x' 'x_dot' 'phi' 'phi_dot'};
inputs = {'u'};
outputs = {'x' 'phi'};

sys_ss = ss(A,B,C,D,'statename',states,'inputname',inputs,'outputname',ou
```

```
sys_ss =
```

```
A =
```

	x	x_dot	phi	phi_dot
x	0	1	0	0
x_dot	0	-0.1818	2.673	0
phi	0	0	0	1
phi_dot	0	-0.4545	31.18	0


```

B =

           u
x         0
x_dot    1.818
phi       0
phi_dot  4.545

C =

           x    x_dot    phi    phi_dot
x         1         0         0         0
phi        0         0         1         0

D =

           u
x         0
phi      0

```

Continuous-time state-space model.

The above state-space model can also be converted into transfer function form employing the `tf` command as shown below. Conversely, the transfer function model can be converted into state-space form using the `ss` command.

```
sys_tf = tf(sys_ss)
```

```
sys_tf =
```

From input "u" to output...

```

           1.818 s^2 + 1.615e-15 s - 44.55
x:  -----
      s^4 + 0.1818 s^3 - 31.18 s^2 - 4.455 s

           4.545 s - 1.277e-16
phi: -----
      s^3 + 0.1818 s^2 - 31.18 s - 4.455

```

Continuous-time transfer function.

Examining the above, note the existence of some terms with very small coefficients. These terms should actually be zero and show up due to numerical round-off errors that accumulate in the conversion algorithms that MATLAB employs. If you set these coefficients to zero, then the above transfer function models will match those generated earlier in the **Transfer Function** section of the example.

Published with MATLAB® 9.2

All contents licensed under a Creative Commons Attribution-ShareAlike 4.0
International License.

