Tutorial

Pablo Rodríguez-Sánchez

August 16, 2019

Installation

Using git

```
git clone https://github.com/PabRod/pendulum
cd pendulum
python setup.py install --user
```

Python modules

The following modules will be needed:

```
## Import the required modules
from pendulum.models import *
import matplotlib.pyplot as plt
import numpy as np
```

The following modules are optional, but recommended:

```
# If using animations
import matplotlib.animation as animation

# If using data as input
from scipy.interpolate import interp1d
import pandas as pd
```

Examples

Simple pendulum

Problem set-up

The dynamics of a simple pendulum are characterized by its length l, the damping constant d and the acceleration of gravity g.

```
## Set-up your problem
l = 1.5 # Length
g = 9.8 # Gravity
d = 0.5 # Damping
```

For non-inertial pendula, we also have to input the pivot's movement. The natural way of coding movement is a time-dependent function for each coordinate. In the present case we want to use a sudden movement to the right happening at $t = 0$. These equations will do:

$$\begin{cases} x(t) &= \arctan(5t) \\ y(t) &= 0 \end{cases}$$

```
# Pivot's position
## The pivot is moving, so its position is a function of time
pos_x = lambda t : np.arctan(5*t)
pos_y = lambda t : 0*t
```

As in any other mechanical problem, we need to introduce the initial conditions (θ_0, ω_0) and the simulation times.

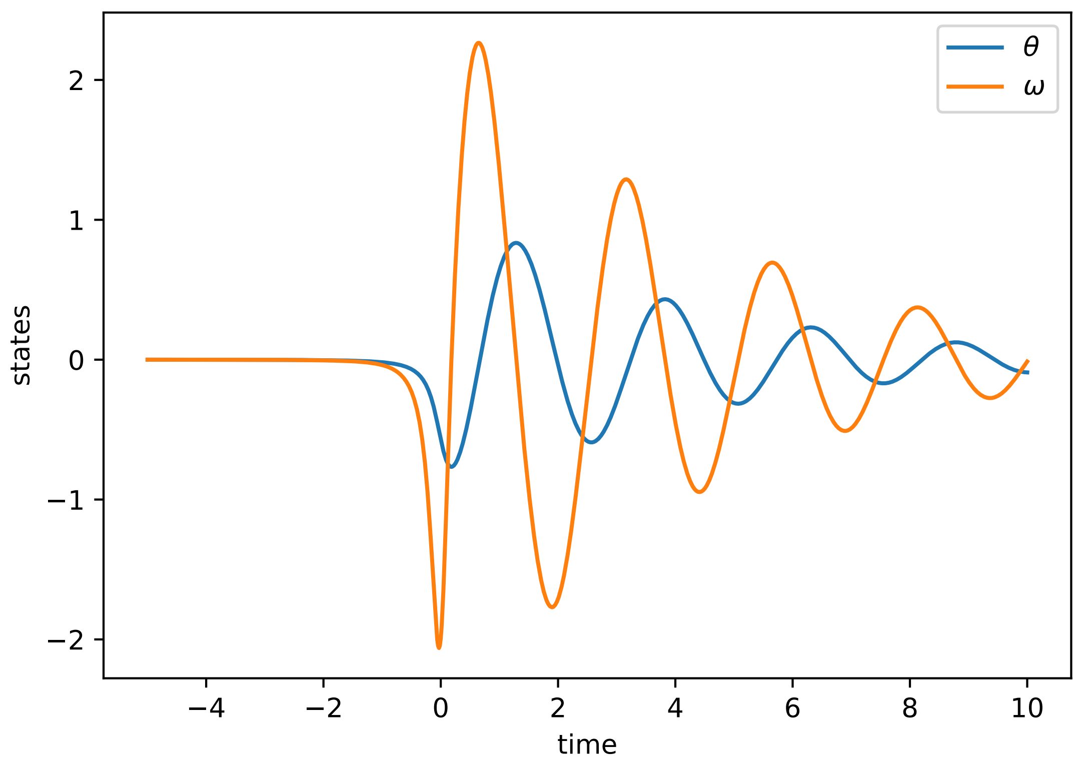
```
ts = np.linspace(-5, 10, 1000) # Simulation time
yinit = (0, 0) # Initial condition (th_0, w_0)
```

Problem solving

Now we are ready to solve our problem:

```
## Solve it
sol = pendulum(yinit, ts, pos_x, pos_y, l = l, g = g, d = d)
```

The object `sol` contains the simulated time series of θ (row 0) and ω (row 1). Plotting them we obtain:



Double pendulum

Problem set-up

The dynamics of a double pendulum are characterized by both lengths (l_0, l_1), masses (m_0, m_1) and the acceleration of gravity g.

```
## Set-up your problem
m = (2, 1) # Masses
l = (1, 1) # Lengths
g = 9.8 # Gravity
```

For the pivot's movement we want to use a sudden movement to the right. These equations will do:

$$\begin{cases} x(t) & = \arctan(3t) \\ y(t) & = 0 \end{cases}$$

```
# Pivot's position
## The pivot is moving, so its position is a function of time
pos_x = lambda t : np.arctan(3*t)
pos_y = lambda t : 0.0*t
```

As in any other mechanical problem, we need to introduce the initial conditions $(\theta_0, \omega_0, \theta_1, \omega_1)$ and the simulation times.

```
ts = np.linspace(-3, 8, 1000) # Simulation time
yinit = (0, 0, 0, 0) # Initial condition (th_1, w_1, th_2, w_2)
```

Problem solving

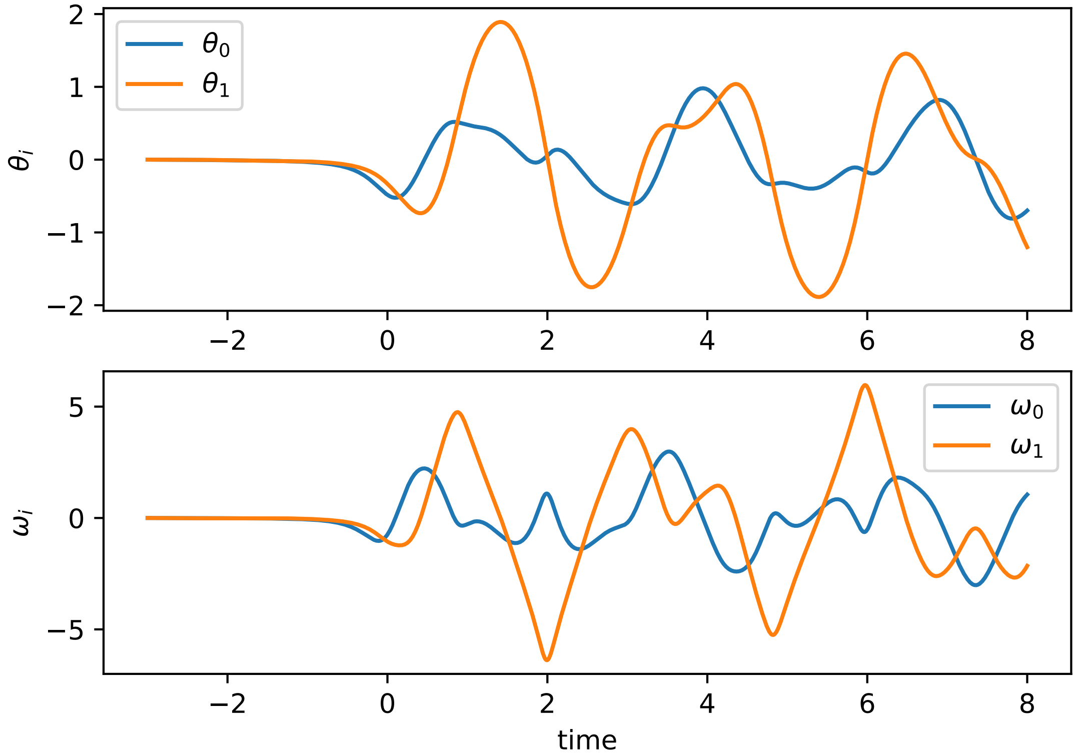
Now we are ready to solve our problem:

```
## Solve it
sol = double_pendulum(yinit, ts, pos_x, pos_y, m = m, l = l, g = g)
```

The object `sol` contains the simulated time series. Each row represents a state, and each column corresponds to a simulated time.

State	Row
θ_0	0
ω_0	1
θ_1	2
ω_1	3

Plotting them:



θ_0
θ_1
θ_i
2
1
0
−1
−2
−2
0
2
4
6
8
ω_0
ω_1
ω_i
5
0
−5
−2
0
2
4
6
8
time