ART MACHINE

QIAO ZHANG

January 26, 2010

1 Introduction

This visualization project is inspired by the Double Pendulum device constructed by Professor R. H. Romer from Amherst College. The Double Pendulum, dubbed "the Art Machine", produces intriguing Lissajous figures on paper. We will first replicate the curious patterns produced by Prof. Romer's "the Art Machine", and then extend to cases that are constrained by limitations of the experimental apparatus.

In this case, the period of the pendulum is directly proportional to $\sqrt{\text{Length of pendulum}}$. As a result, cases in which the relative frequency $\frac{\omega_y}{\omega_x} > 2.0$ are difficult to realize in experiments. However, such cases can be easily investigated with the help of computational and visualization tools. In addition, we will explore Triple Pendulum and produce some 3-dimensional Lissajous figures.

2 Theory & Model

The theory is surprisingly simple. Despite the deceptive appearance of the two pendulums being connected through the pen, the two pendulums are experimentally found to be, in fact, independent of each other. Hence, the motion of the pen can be decomposed to motions in two perpendicular directions (denoted by x and y directions from henceforth).

We approximate displacements in the x and y directions with the angular displacements of the two pendulums respectively. We model the damping effect with the exponential decay terms. The coordinates of the pen are thus given by

$$x = A_x \cos(\omega_x t) \exp(-\lambda_x t) \tag{1}$$

$$y = A_y \cos(\omega_y t + \delta) \exp(-\lambda_y t) \tag{2}$$

3 Methodology

3.1 Significance of parameters

There are four groups of parameters in the given model: the amplitudes, the angular frequencies, the phase difference and the damping constants.The siginificance of the different groups of parameters will be explored.

3.2 Replication of special curves

The special curves for $\frac{\omega_y}{\omega_x} = 1.0, 1.5, 2.0$ mentioned in Prof. Romer's paper will be replicated.

3.3 Beyond the experimental

Curves for $\frac{\omega_y}{\omega_x} > 2$ and 3D Lissajous figures will be presented.

4 Visualization in Mathematica

4.1 Significance of parameters

1. **Amplitude** When $A_x = A_y$, the Lissajous figure is just a straight line 45° to the horizontal, as seen in Fig. 1. However, when the relative amplitude $\frac{A_y}{A_x}$ is increased, the slope of the straight line rises, as seen in Fig. 2.

2. **Damping Constant** The damping constant determines how long the pendulums will oscillate and hence how long the pencil draws on the paper. A lower damping constant produces a more sharply defined curve, as seen in Fig. 3.

3. **Angular Frequency** The relative angular frequency $\frac{\omega_y}{\omega_x}$ apparently determines the number of loops present in the curve, as seen in Fig. 4. Moreover, only rational relative frequency will produce closed figures.

4. **Phase Difference** The phase difference adds a mystical level of complexity to the curve, as seen in Fig. 5.

4.2 Replication of special curves

The special curves for $\frac{\omega_y}{\omega_x} = 1.0, 1.5, 2.0$ mentioned in Prof. Romer's paper are replicated here.

1. $\frac{\omega_x}{\omega_y} = 1.0$ as in Fig. 6

2. $\frac{\omega_x}{\omega_y} = 1.5$ as in Fig. 7

3. $\frac{\omega_x}{\omega_y} = 2.0$ as in Fig. 8

4.3 Beyond the experimental

1. Curves for $\frac{\omega_x}{\omega_y} = 2.5$, as seen in Fig. 9.
2. Curves for $\frac{\omega_x}{\omega_y} = 3.0$, as seen in Fig. 10.
3. 3D Lissajous figures will be presented, as seen in Fig. 11(a),11(b),11(c).

4.4 Dynamic Visualization

The Mathematica package provided allows dynamic visualization of the 2D and 3D Lissajous Figures.

```
Manipulate[ParametricPlot[
   {a1 Cos[w1 t + \[Delta]] Exp[-(10^-5) t],
   a2 Cos[t] Exp[-(10^-5) t]}, {t, 0, 1000}, Axes -> None],
 {{a1, 1, "x-amplitude"}, 0, 2,
  Appearance -> "Labeled"}, {{a2, 1, "y-amplitude"}, 0, 2,
  Appearance -> "Labeled"},
 {{w1, 0, "relative frequency"}, 0, 5,
  Appearance -> "Labeled"}, {{\[Delta], 0, "phase difference"}, 0, 2,
  Appearance -> "Labeled"}
 ]

 Manipulate[ParametricPlot3D[
   {a1 Cos[w1 t + \[Delta]] Exp[-(10^-5) t],
   a2 Cos[w2 t + \[Gamma]] Exp[-(10^-5) t],
   a3 Cos[w3 t] Exp[-(10^-5) t]}, {t, 0, 1000}, Axes -> None,
  Boxed -> False],
 {{a1, 1, "x-amplitude"}, 0, 2,
  Appearance -> "Labeled"}, {{a2, 1, "y-amplitude"}, 0, 2,
  Appearance -> "Labeled"}, {{a3, 1, "z-amplitude"}, 0, 2,
  Appearance -> "Labeled"},
 {{w1, 0, "x-frequency"}, 0, 2,
  Appearance -> "Labeled"}, {{w2, 0, "y-frequency"}, 0, 2,
  Appearance -> "Labeled"}, {{w3, 0, "z-frequency"}, 0, 2,
  Appearance -> "Labeled"}, {{\[Delta], 0, "phase difference"}, 0, 2,
  Appearance -> "Labeled"}, {{\[Gamma], 0, "phase difference"}, 0, 2,
  Appearance -> "Labeled"}]
```

5 Visualization in Matlibplot

A visualization program in Python is also provided. The code in Python is provided below, and the figure produced is shown in Fig. 11.

```
import matplotlib as mpl
from mpl_toolkits.mplot3d import Axes3D
import numpy as np
import matplotlib.pyplot as plt

mpl.rcParams['legend.fontsize'] = 10

fig = plt.figure()
ax = Axes3D(fig)
t = np.linspace(0,1000,1000)
x = np.sin(t)
y = np.sin(2*t+5)
z = np.sin(5*t)
ax.plot(x, y, z, label='parametric curve')
ax.legend()

plt.show()
```

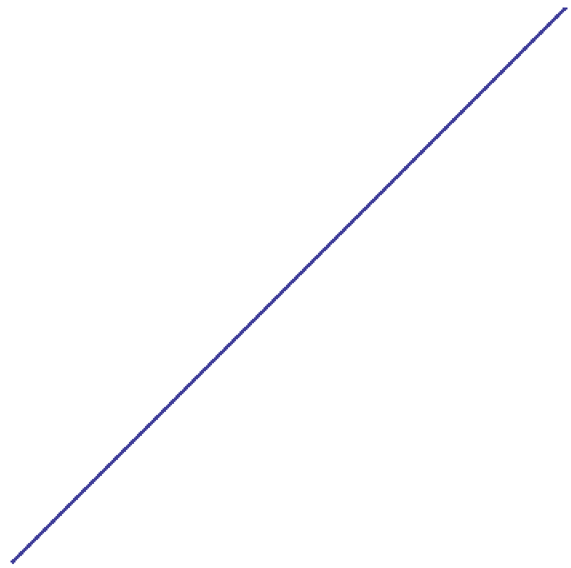


Figure 1: $\frac{A_y}{A_x} = 1$

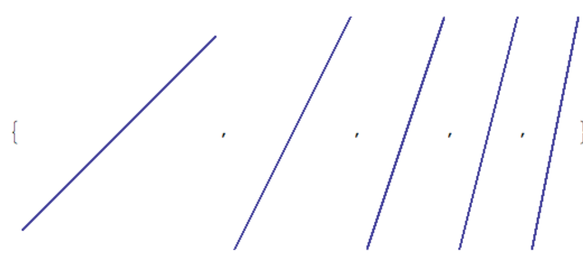


Figure 2: $\frac{A_y}{A_x} = 1, 2, 3, 4, 5$

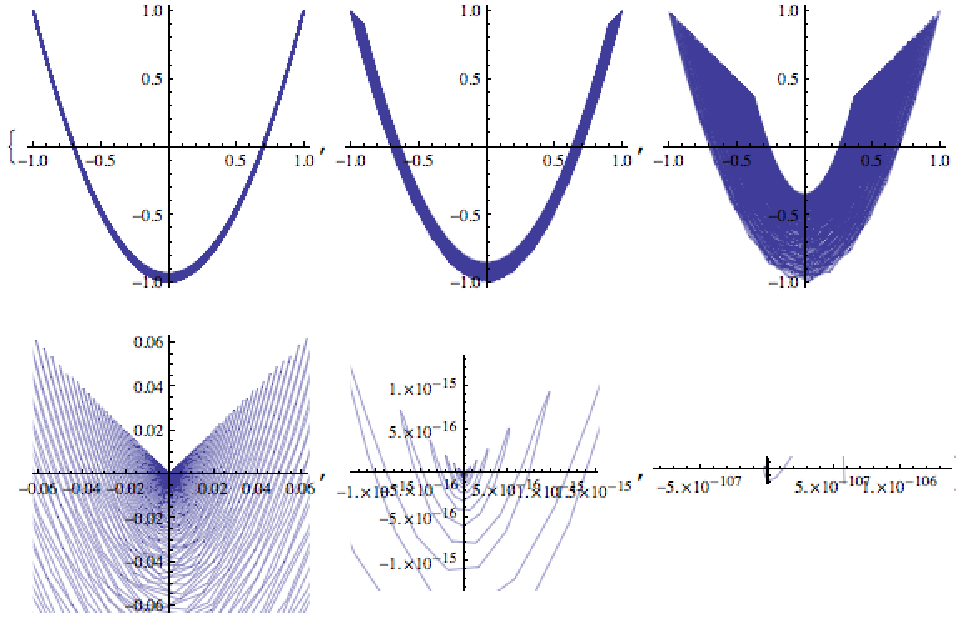


Figure 3: $\lambda_x = \lambda_y = 10^k, k = -5, -4, -3, -2, -1$

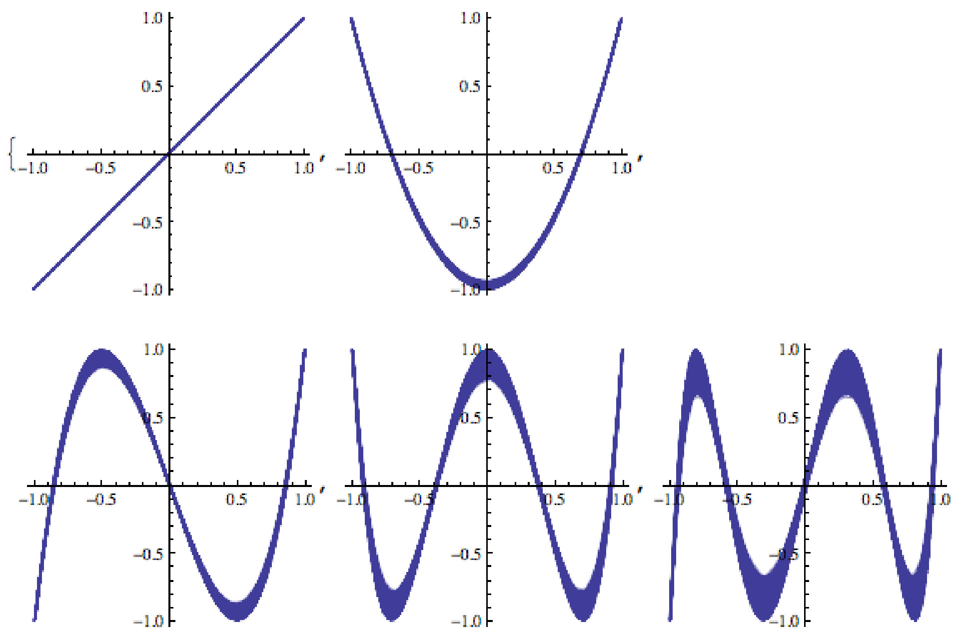


Figure 4: $\frac{\omega_y}{\omega_x} = 1, 2, 3, 4, 5$

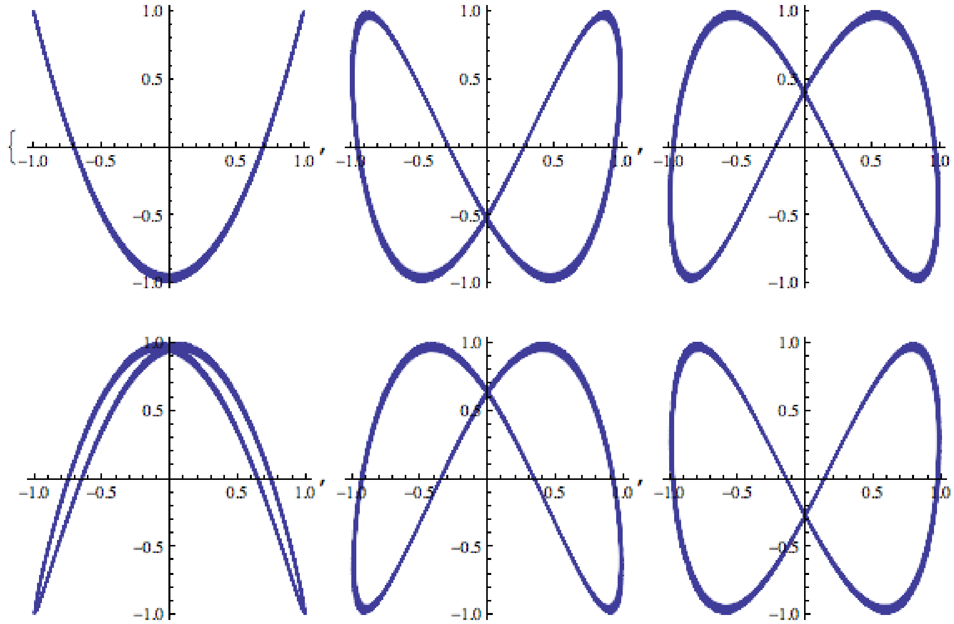


Figure 5: $\delta = 1, 2, 3, 4, 5$

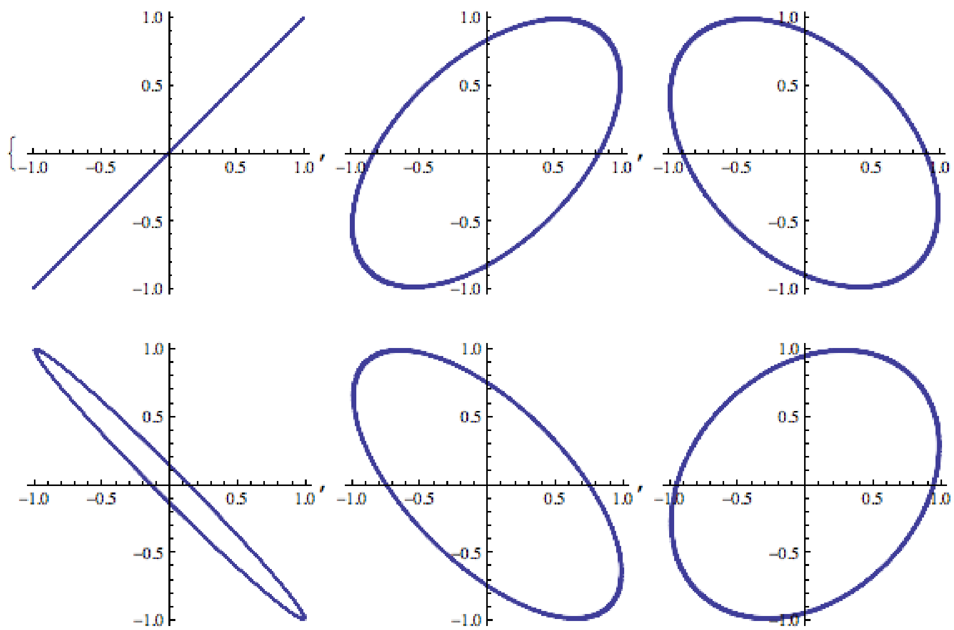


Figure 6: $\frac{\omega_x}{\omega_y} = 1.0$ with $\delta = 0, 1, 2, 3, 4, 5$

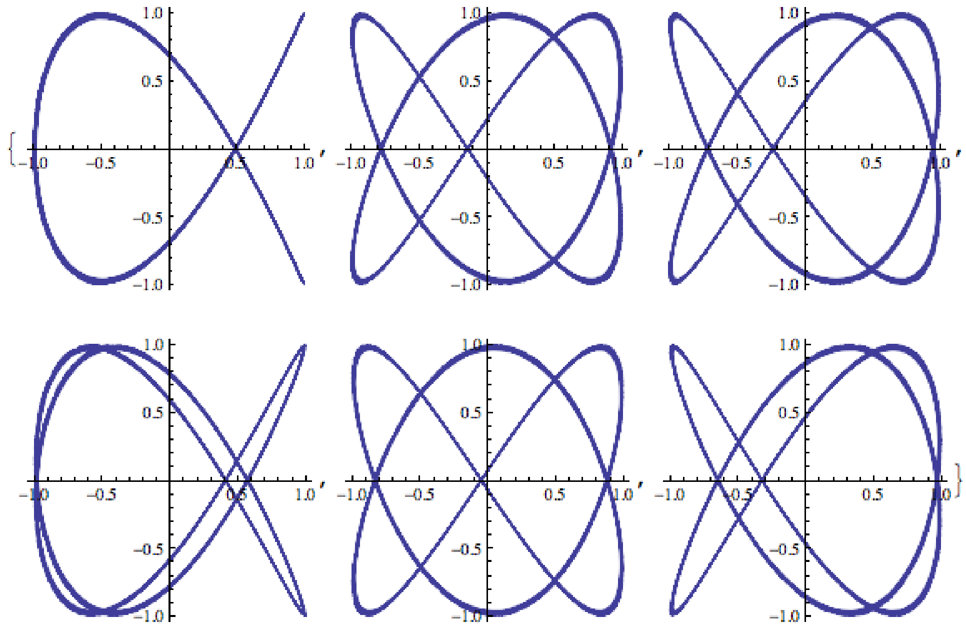


Figure 7: $\frac{\omega_x}{\omega_y} = 1.5$ with $\delta = 0, 1, 2, 3, 4, 5$

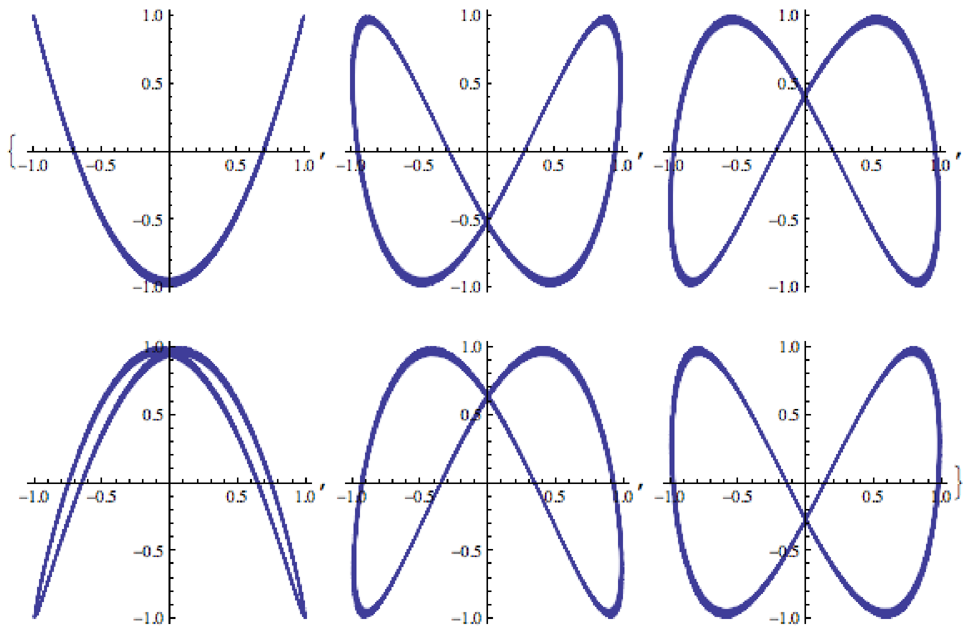


Figure 8: $\frac{\omega_x}{\omega_y} = 2.0$ with $\delta = 0, 1, 2, 3, 4, 5$

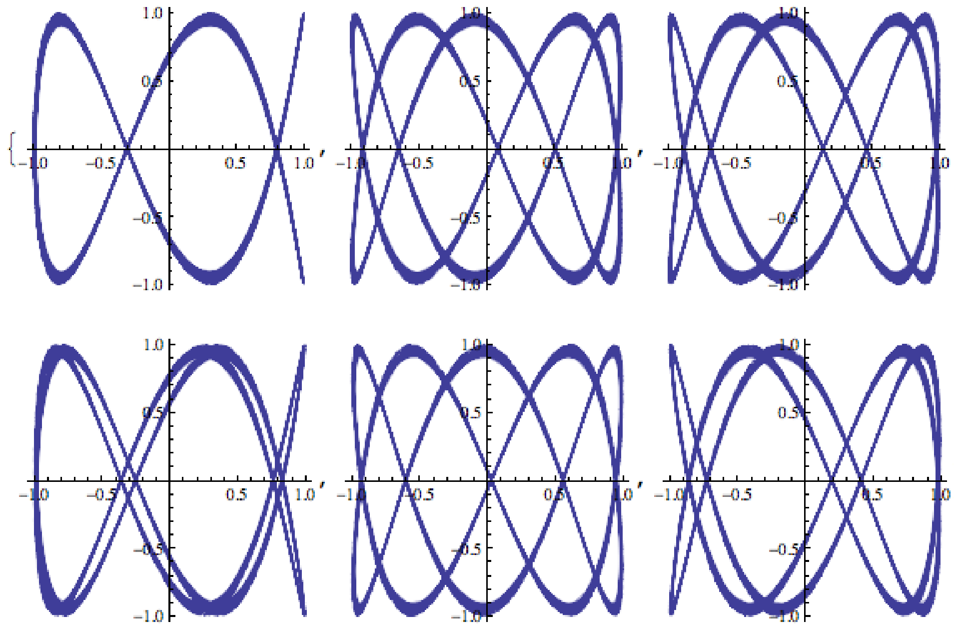


Figure 9: $\frac{\omega_x}{\omega_y} = 2.5$ with $\delta = 0, 1, 2, 3, 4, 5$

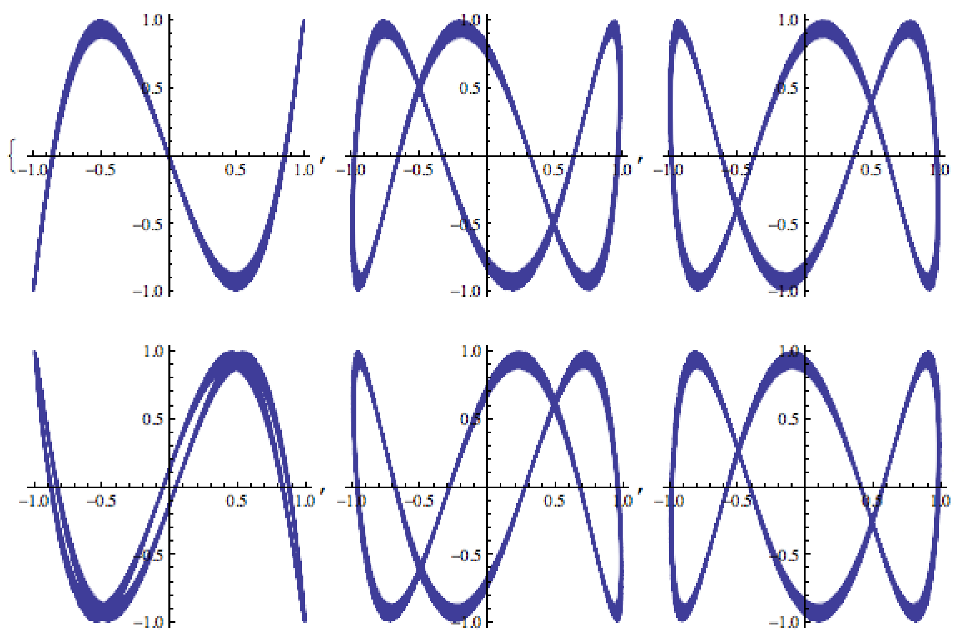
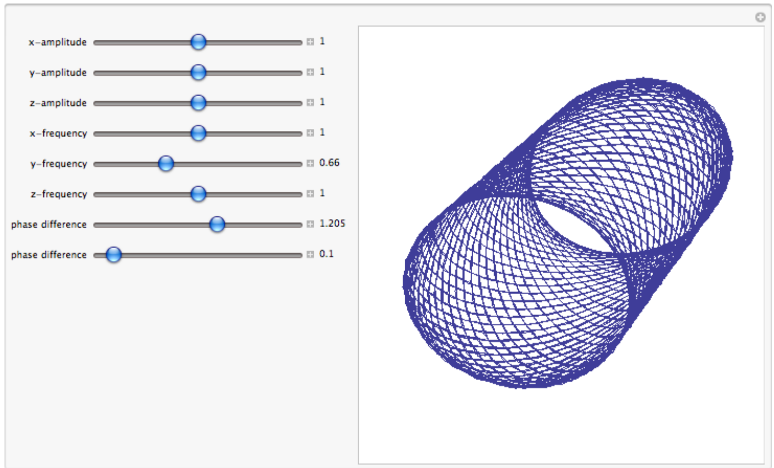
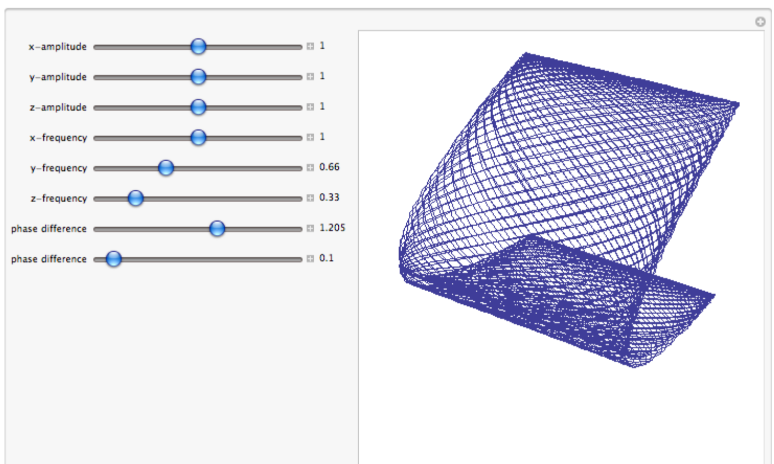


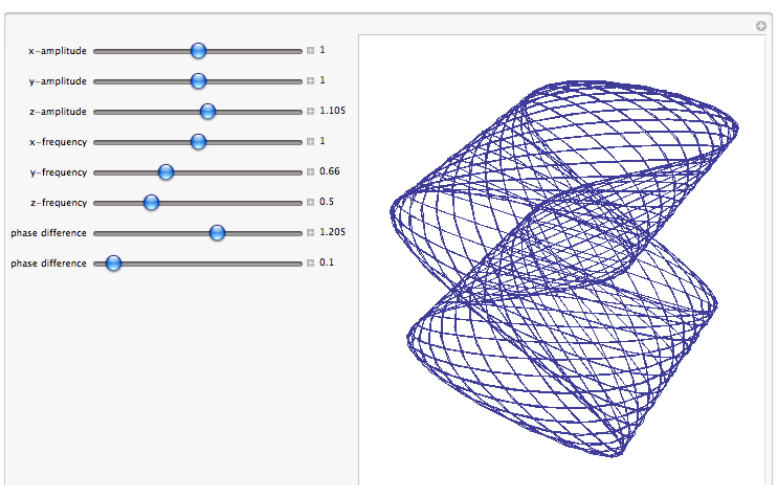
Figure 10: $\frac{\omega_x}{\omega_y} = 3.0$ with $\delta = 0, 1, 2, 3, 4, 5$



(a) 3D Lissajous figures 1



(b) 3D Lissajous figures 2



(c) 3D Lissajous figures 3

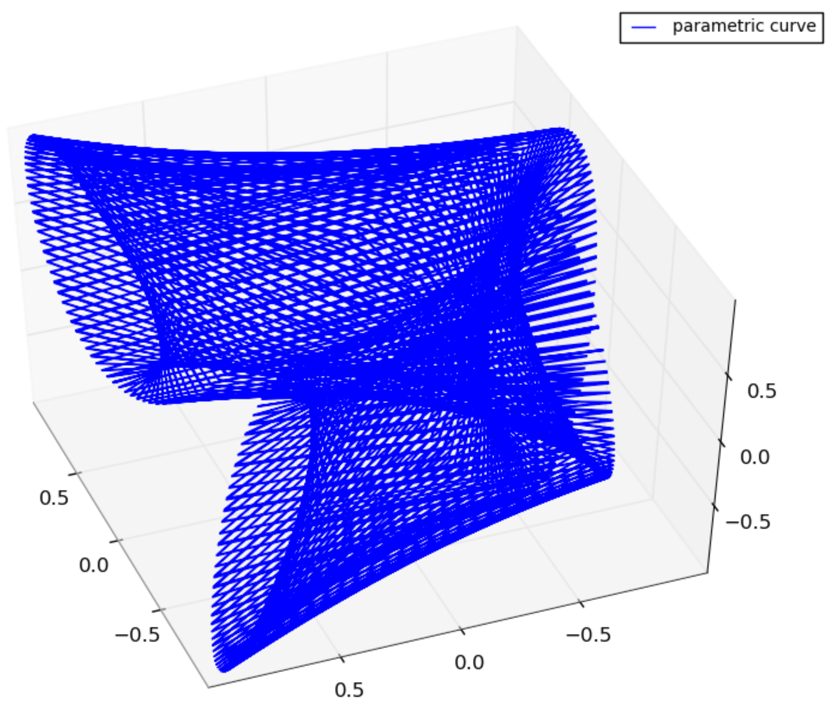


Figure 11: Figure produced in Python