Vanderpol Oscillator

Shubham Chauhan

22/10/2016

1 Introduction

Vandepol oscillator is defined as following differential equation.

$$x'' + \mu(1 - x^2)x' + x = 0 \tag{1}$$

Above equation represents family of solutions. For different initial values we get a new kind of oscillator. Oscillation are dependent on the value μ which is the only parameter for this kind of system.

For the purpose of this poject, behaviour of ode with respect to time is noted. Generated output for default values are saved in "output" directory.

2 Usage

2.1 Input Format

This system needs to be initialized for x, x", μ. Default values are set if no input is provided. To alter initial values change "input.txt" file in the root folder. To genrate animation of the differential equation, it needs 2 more input time and frames. Sample input is shown below.

```
mu;x;x_dot;time;frames
0.3;0.0;1.0;20;500
```

Second line in input file contains all values. Corresponding labels are written in first line of the text file. Values are expected to be numbers. Delete the output folder and make again with different input values.

2.2 Compile

The simplest way to compile this package is:

1. 'cd' to the directory containing the package's source code
2. Change input values to get an oscillator of you choice
3. Type 'make' to compile the package
4. Check for results in 'output' folder

2.3 Working

Source folder contains all the codes.

1. "Vanderpol.py" has a solver class for differential equation solving, written in a separate to make use of the code in better way.

2. ”graphics.py” generates all required plots and video for a given case.
3. ”130010021.tex” compiles to create pdf report of this project.

3 Conclusions

In this report the behaviour of Vanderpol Oscillatoris explored with respect to time. The theory involved is verified with the observation. Curve moves outwards with respect to time and remains there. The plot shown in Figure 1 depicts the phase difference bettween x’ and x”.

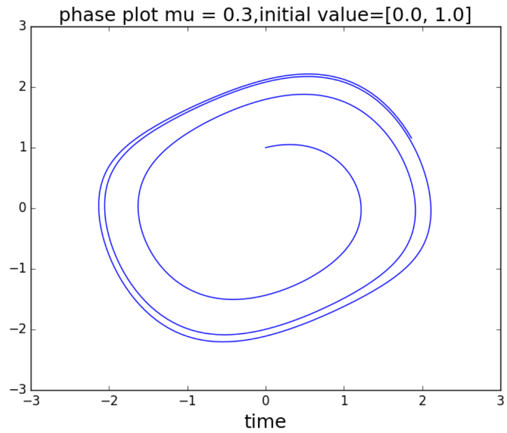

Figure 1: Phase plot

$$\frac{dx^n}{dx} = (n+1)x^n \text{ if } x \neq -1$$

A plot of position and velocity Vs time is shown in Figure 2.

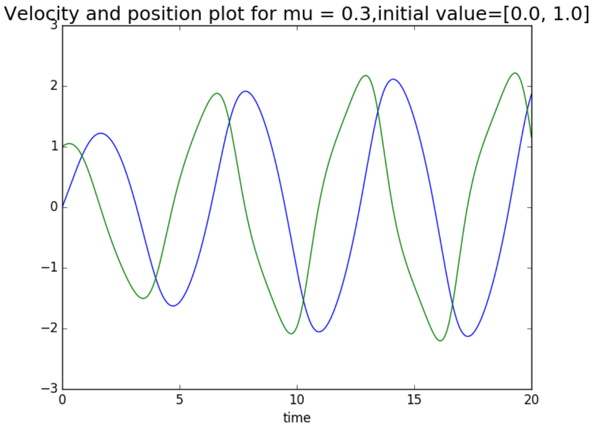

Figure 2: Position and Velocity Variations

4 Libraries

Pulblic git hub repository is maintained here `https://github.com/shubhamchauhan/vanderpol-oscillator`
Softwares used:

1. Python version 3.5
2. Scipy 0.17.1
3. matplotlib 1.5.1