Report

of

Exercise sheet 1:

Pendulum with harmonically driven pivot

for

269002 VO Computational Concepts

in Physics I (2021W)

Student name:	Clemens Wager, BSc
Matriculation nr:	01635477
Degree program:	Master of Computational Science
Supervisor:	Univ.-Prof. Dr. techn. Cesare Franchini

Vienna, 12.01.2022

Introduction

The pendulum is the simplest example for oscillatory motion. It consists of a mass (called a “bob”) that is connected by a string to a supporting point. In this computer experiment this point is a harmonically driven pivot. Thus, both the bob and the pivot exhibit oscillatory motion. A simulation of their behavior was studied and visualized in a Python implementation.

Method

Theory

The main task was to write a simulation for the bob’s motion with respect to angle θ over time (see figure 1). The string that connects the bob to the supporting point is of constant length L so it is not stretchable and rigid. The pendulums initial parameters are angle θ_0 and angular velocity ω_0. The driven pivot has a period T_D and an amplitude A_0. Figure 1 also illustrates the gravitational force $F_g = mg$ which can be split into its tangent component F_t and radial component F_r, where m is the mass of the bob and g the gravitational acceleration on Earth. Due to the nature of the string F_r has no effect, so only F_g is significant. There was no damping factor involved.

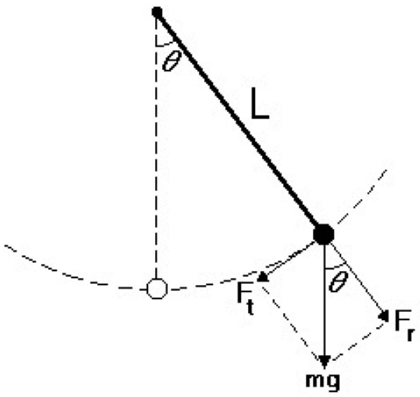

Figure 1 shows the components of a simple pendulum with fixed pivot and forces acting on the bob.

From Newton’s second law ($F = ma$) we get that

$$F_t = -mg\sin\theta = m\frac{d^2 s}{dt^2},$$

where $s = L\,\theta$. We can simplify this expression w.r.t. m.

With the simple approximation of $\sin(\theta) \approx \theta$ for small θ we obtain the central equation of simple harmonic motion as linear ordinary differential equation (ODE)

$$\frac{d^2\theta}{dt^2} = -\frac{g}{l}\theta,$$

which can be solved analytically.

We can split this second order differential equations (DE) into two first order DEs like so:

$$\frac{d\omega}{dt} = -\frac{g}{l}\theta$$

$$\frac{d\theta}{dt} = \omega$$
,

where ω is the angular velocity.

In order to simulate also for large input angle θ_0 a non-linear version of the ODEs was implemented with $sin(\theta)$.

$$\frac{d\omega}{dt} = -\frac{g}{L}sin(\theta)$$

$$\frac{d\theta}{dt} = \omega$$

ω and θ were propagated through the Euler-Cromer Method (see algorithm 1) and acceleration was updated in each iteration as well. Together with the acceleration $a_d(t)$ of the harmonically driven pivot the implemented equation for acceleration is

$$\frac{d^2\theta}{dt^2} = -\frac{g + a_d(t)}{L}\sin\theta$$
,

where

$$a_d(t) = A_0 \sin(2\pi t/T_d)$$
.

This equation was implemented in Python as

```
accel = -(g_over_L + A0 * numpy.sin(2*numpy.py*time/Td)) *
                    numpy.sin(theta)
```

where g_over_L is $\frac{g}{L} = 1$.

To propagate the pendulum's angle over time the Euler-Cromer Method was used. This algorithm is well-suited for systems where energy is conserved. To plot the data in the end θ was recorded in each time step. Below is a pseudocode of the algorithm:

Euler-Cromer Method for Pendulum

1: $\omega_{i+1} = \omega_i - \frac{g}{L}sin(\theta_i)\Delta t$

2: $\theta_{i+1} = \theta_i + \omega_{i+1}\Delta t$

3: $t_{i+1} = t_i + \Delta t$

Algorithm 1 is the Euler-Cromer Method written in pseudocode. ω is the angular velocity and θ is the angle. These steps are repeated for the total number of steps.

Implementation

The simulation was implemented in Python 3.7 and uses the libraries numpy and matplotlib.pyplot. The implementation incorporates some parts of the source code given in the lecture. When executing the program, one is asked whether to proceed with standard parameters or if one wants to type in different values into the CLI. After the computation is done, a plot is created to visualize the angle versus time. The user has the option to change the Boolean `plotOmega` to also plot the angular velocity over time. Extra lines for orientation will be added to the first plot when setting `plotExtraLines = True`.

Results and Discussion

In order to run a standardized simulation, the program was executed with the given initial parameters: $A_0 = 100$g, $T_D = 0.2$, g/L $= 1$, tau $= 0.004$ (time step). They remained unchanged in the following simulations discussed unless specified explicitly. A pendulum was simulated with 3000 iterations, initial $\theta_0 = 20°$ and initial velocity $\omega_0 = 0.0\ \frac{degree}{s}$ (see figure 1). The maximum displacement reached just above 25.5° due to the driving force pushing the pendulum higher than its initial position. The bob swings back to about − 25.5° and thus a relatively stable motion pattern sets in, where the extreme angles vary between ± 25.4° and ±25.7° (see figure 2). The average period length was 4.276 seconds.

(see table 1 in appendix for more information)

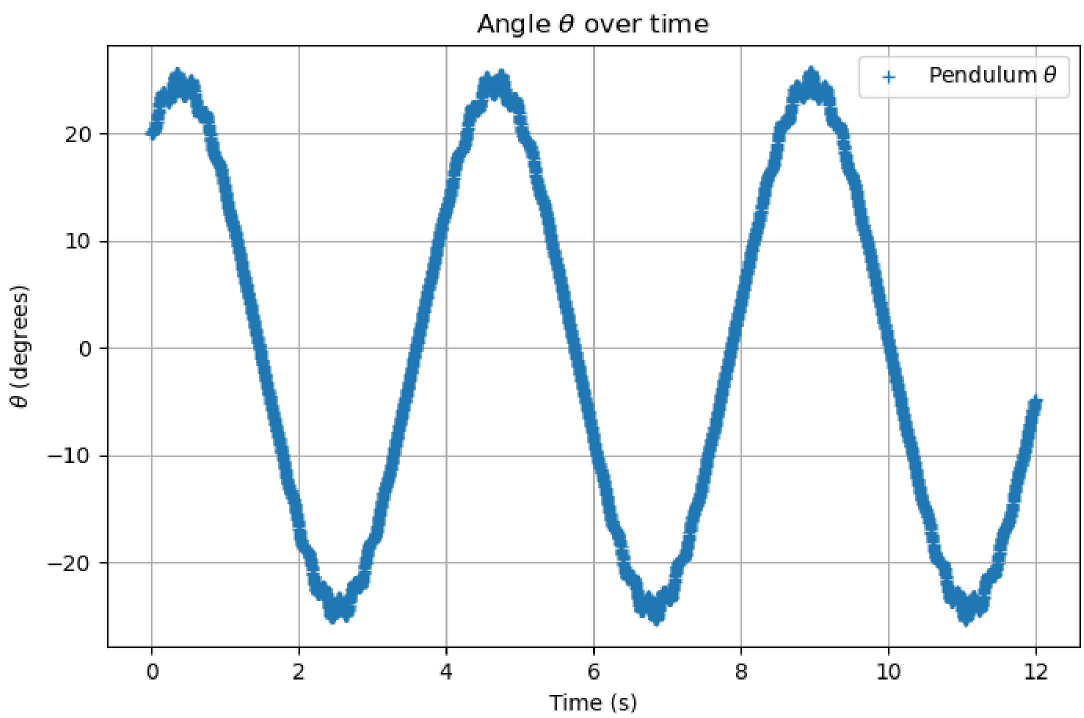

Figure 1 is a visualization of the standardized simulation. It shows the angular displacement in degrees over time in seconds. The y-axis represents the angle θ in degrees. Average period length is 4.276 seconds in this simulation with standard parameters.

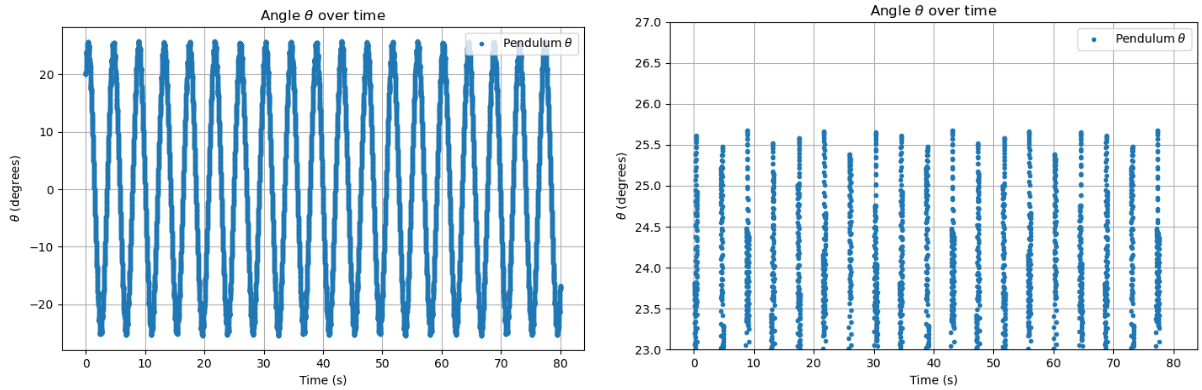

Figure 2 shows the standardized simulation over 80 seconds. The left and right side show the same plot but the y-axis on the right one was limited to 23°-27° to visualize the small variations in the extreme angles.

The second aim of this computer experiment was to observe what would happen if the pendulum was simulated with initial $\theta_0 \approx 180°$ meaning upside-down. The hypothesis was that the pendulum would be stable in the inverted position when the driving acceleration is sufficiently high ($A_0 \gg g$) such that it would oscillate about the point $\theta = 180°$.

When $\theta_0 = 178°$ and $A_0 = 1000g$ the pendulum oscillated around 180° swinging about 3° to each side (see figure 3). This means that the harmonically driven pivot stabilized the pendulum in this motion pattern. Thus, the hypothesis could be verified.

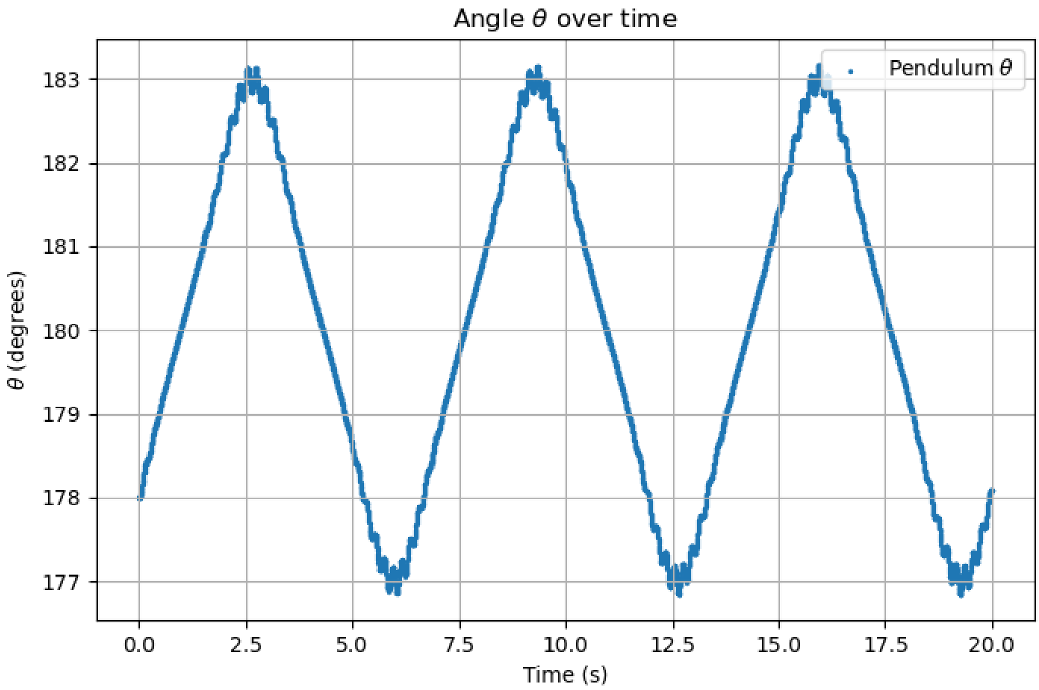

Figure 3 shows the plot of a simulation with $\theta_0 = 178°, A_0 = 1000g$ over 20 seconds. The pendulum oscillates around 180° and remains stable in this motion pattern.

On the other hand, when started at $\theta_0 = 180.1°$ the pendulum only oscillated above 180.1°, but never below that (figure 4). When A_0 was increased from the 100g (left) to 10 000g (right) the amplitude and period length decreased. While the maximum amplitude was about 208.5° on the left plot, it was only at about 180.250° on the right one. The pendulum's inverted position was more stable with higher driving acceleration. However, the pendulum only leaned to the side it started at and did not oscillate around 180° for small θ_0. In general, the pendulum was stable in this position verifying the hypothesis.

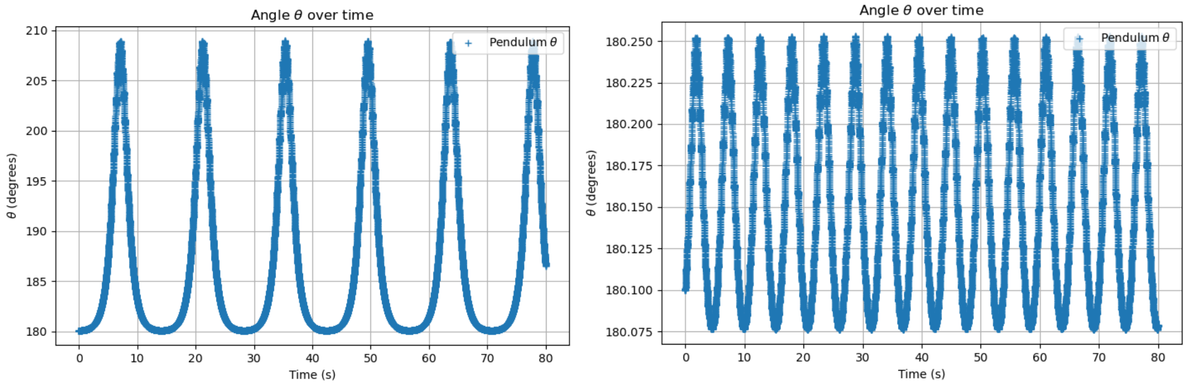

Figure 4 shows the pendulum's motion over 80 seconds when released at $\theta_0 = 180.1°$ in both plots. The only difference in parameters was that in the left simulation $A_0 = 100g$ and in the right one $A_0 = 10\,000g$. The amplitude and period length decreased as A_0 was increased.

Conclusion

In this computer experiment the motion of a harmonically driven pendulum was studied. The oscillating behavior was observed under various parameters and visualized. The simulation incorporated the Euler-Cromer method to calculate the discretization steps.

Further, the hypothesis that the pendulum would be stable in the inverted position when driving acceleration is sufficiently high turned out to be true, because it oscillated around 180° at certain initial angles. Interestingly, when θ_0 was too close to 180° the oscillation turned into a one-sided "seesaw-motion".

Appendix

Size	Value
period length	4.276 s
Turning point #1	1.468 s
Turning point #2	3.596 s
Turning point #3	5.744 s
Turning point #4	7.868 s
Turning point #5	10.02 s
Turning point #6	12.144 s
Turning point #7	14.292 s
Turning point #8	16.42 s
Turning point #9	18.568 s

Table 1: Additional information of the standardized simulation.

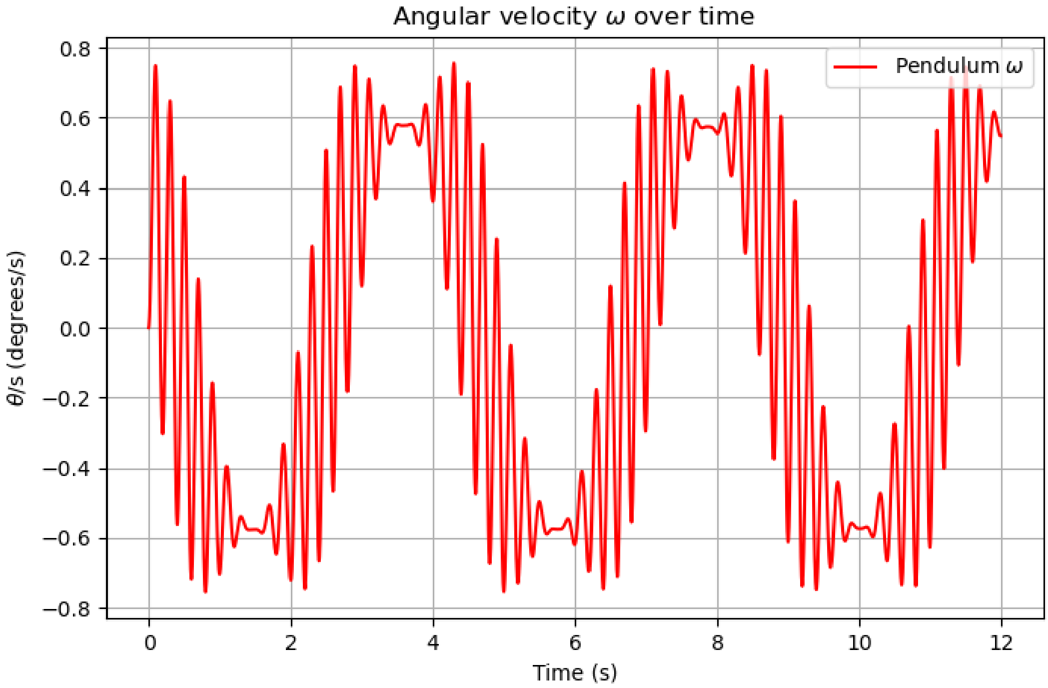

Figure 5 shows the angular velocity over 12 seconds of a standardized simulation.