1 Introduction

The damped, driven pendulum is an example of a chaotic physical system. The motion of a simple pendulum (mass m, length l, angle θ with the vertical) without driving or damping is governed by the differential equation

$$ml\ddot{\theta} + mg\sin\theta = 0$$

We add a velocity-depending damping term $\nu\dot{\theta}$ and a driving force of the form $A\cos\omega t$ to obtain

$$ml\ddot{\theta} + \nu\dot{\theta} + mg\sin\theta = A\cos\omega t$$

Which is the differential equation governing the damped, driven pendulum. Now, we introduce the following dimensionless variables:

$$\tau = \omega_0 t$$
$$Q = mg/(\nu\omega_0)$$
$$\hat{\omega} = \omega/\omega_0$$
$$\hat{A} = A/(mg)$$

The derivative becomes $d/dt = \omega_0 d/d\tau$, so that the differential equation can be written as

$$\omega_0^2\ddot{\theta} + \frac{\omega_0\nu}{ml}\dot{\theta} + \omega_0^2\sin\theta = \frac{A}{ml}\cos\hat{\omega}t$$

$$\implies \ddot{\theta} + \frac{\nu}{ml\omega_0}\dot{\theta} + \sin\theta = \frac{A}{ml\omega_0^2}\cos\hat{\omega}t$$

Where the derivatives in θ are now taken with respect to τ. Simplifying in terms of the new dimensionless variables,

$$\ddot{\theta} + \frac{1}{Q}\dot{\theta} + \sin\theta = \hat{A}\cos\hat{\omega}t$$

Which can be split into two couple first-order ODEs by defining $\phi = \dot{\theta}$:

$$\frac{d\theta}{d\tau} = \phi \tag{1}$$

$$\frac{d\phi}{d\tau} = \frac{1}{Q}\phi - \sin\theta + \hat{A}\cos\hat{\omega}t \tag{2}$$

Which will allow us to apply the Runge-Kutta (RK4) method.

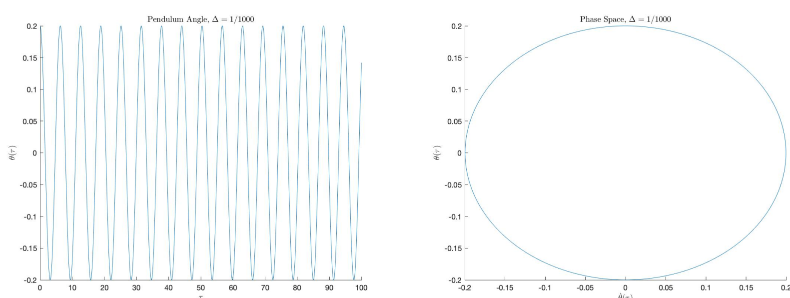
2 Numerical Simulations

2.1 Simple Pendulum

We will first analyse the simple pendulum by setting $A = \nu = 0$, as this will be a good test to see if our algorithm at least returns what we expect in this case where we have an (approximate) analytical solution for small angles. If $\theta << 1$ we may set $\sin\theta \approx \theta$ to obtain

$$ml\ddot{\theta} = -\theta$$

Which admits sinusoidal solutions. These solutions (using a small initial condition $\theta_0 = 0.2$) can be seen in figure 1.

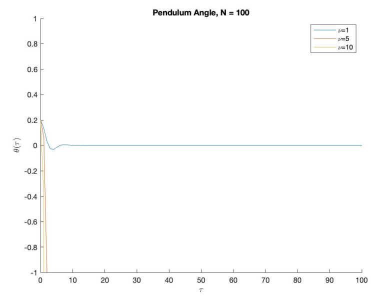


(a) The simple pendulum, which admits a sinusoidal solution using RK4 as expected. $\theta_0 = 0.2$ and $\dot{\theta}_0 = 0$.

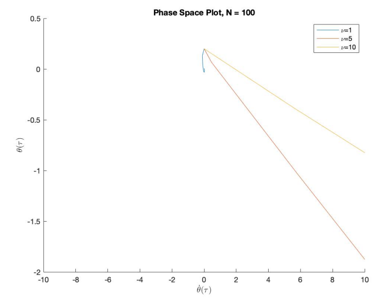
(b) The simple pendulum, which admits a sinusoidal solution using RK4 as expected. $\theta_0 = 0.2$ and $\dot{\theta}_0 = 0$.

2.2 Damped Pendulum

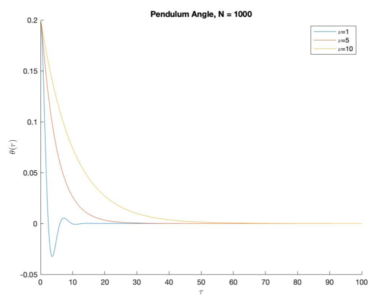
Now we will investigate the damped, but undriven pendulum by setting $A = 0$. Plots using $\nu = 1$, 5, and 10 and are given in figure 2 at various values of N and Δ. All physical parameters are set to 1.



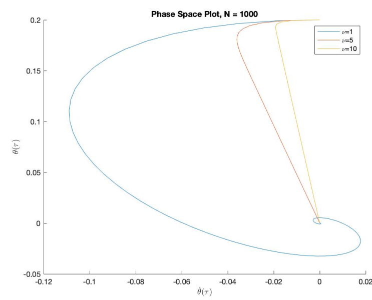
(a) Various values of ν, with $N = 100$ and $\Delta = 1$.



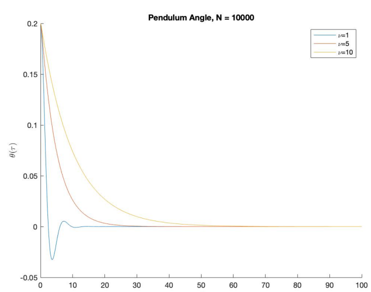
(b) Various values of ν, with $N = 100$ and $\Delta = 1$.



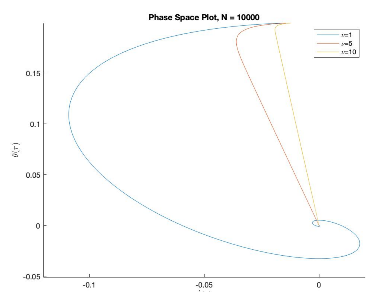
(c) Various values of ν, with $N = 1000$ and $\Delta = 1/10$.



(d) Various values of ν, with $N = 1000$ and $\Delta = 1/10$.



(e) Various values of ν, with $N = 10000$ and $\Delta = 1/100$.



(f) Various values of ν, with $N = 10000$ and $\Delta = 1/100$.

Figure 2: Comparing the accuracy of the damped oscillator with different values of N.

Clearly $N = 100$, $\Delta = 1$ is not sufficient for calculating accurately. However, there seems to be very little difference between using $\Delta = 1/10$ and $\Delta = 1/100$.

Now, I'll classify each case. Recall that, in the differential equation $m\ddot{\theta} + b\dot{\theta} + k\theta = 0$, there are three cases:

- $b^2 < 4mk$ is underdamping
- $b^2 > 4mk$ is overdamping
- $b^2 = 4mk$ is critical damping

In our case, assuming small angles, we have $\ddot{\theta} + 1/Q\dot{\theta} + \sin\theta \approx \ddot{\theta} + 1/Q\dot{\theta} + \theta = 0$, so we have $m = 1$, $b = 1/Q$, and $k = 1$. Now, recall that the physical parameters were set to $g = m = \omega_0 = 1$. Thus $Q = mg/(\omega_0 \nu) = 1/\nu$, hence $b = \nu$. So we have the following cases:

- $\nu = 1$ gives underdamping, since $b^2 = \nu^2 = 1$ and $4mk = 4$.
- $\nu = 5$ gives overdamping since $b^2 = \nu^2 = 25$ and $4mk = 4$.
- $\nu = 10$ gives overdamping, since $b^2 = \nu^2 = 100$ and $4mk = 4$

This is displayed in figure 2. $\nu = 1$ oscillated before dying off, whereas $\nu = 5$ and $\nu = 10$ both decay very quickly and do not oscillate.

Notice that as N gets larger (and therefore Δ gets smaller), the solutions clearly become more stable. Plotted in the following figure is the energy (which works out to be $E = 1 + 1/2\nu^2 - \cos\theta$) of the system for various values of ν.

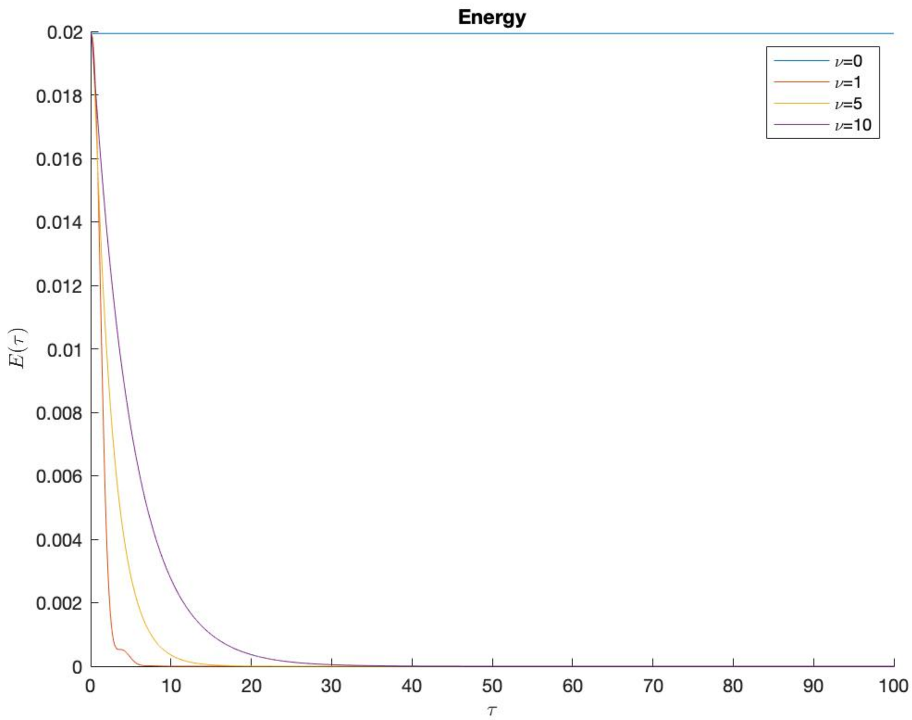
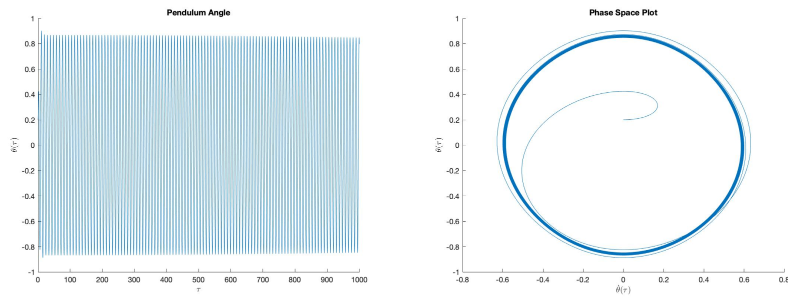


Figure 3: The energy of the pendulum. The $\nu = 0$ case is perfectly constant, as expected. For the damped systems, we see the energy die off to zero as the oscillations die off.

2.3 Damped Driven Pendulum

Now we can analyse the forced system, with $\nu \neq 0$ and $A \neq 0$. Throughout this section, I'll set $l = g = m = 1$, $\nu = 1/2$, and $\omega = 2/3$. The initial conditions will again be $\theta_0 = 0.2$, and $\dot{\theta}_0 = 0$. I'll choose $\Delta = 1/10000$.

In figure 4, with $A = 0.5$, we can see that the pendulum quickly starts to follow a stable, periodic motion.



(a) A result which tends toward periodic motion.

(b) The phase space diagram tends toward a circular pattern, indicating periodicity.

Figure 4: The driven, damped pendulum with $A = 0.5$.

We can see this approaches the analytical solution derived in class for small angles in figure 5.

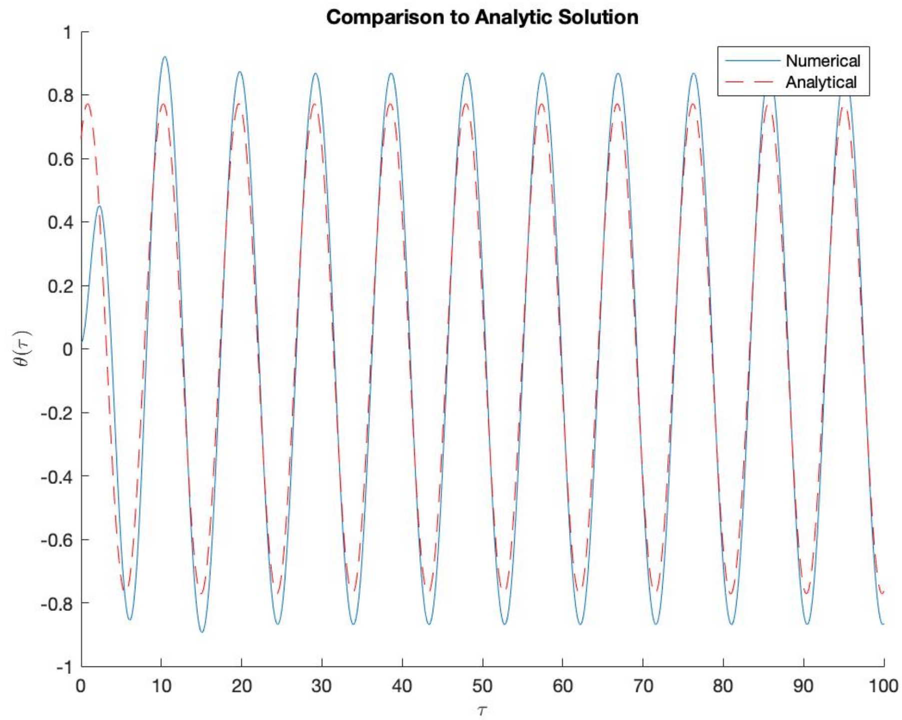
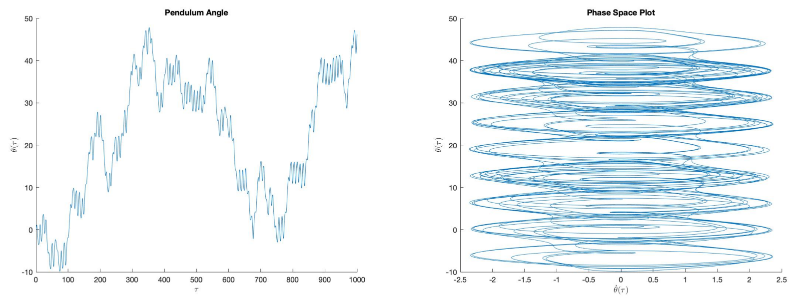


Figure 5: The numerical solution starts approach the (approximate) analytic solution in long-time behaviour.

However, in figure 6, we see a more chaotic, unpredictable behaviour when $A = 1.2$. Unlike in the $A = 0.5$ case, there is no periodic motion at all; this is especially clear in the phase space plot, where there is certainly no circular pattern.

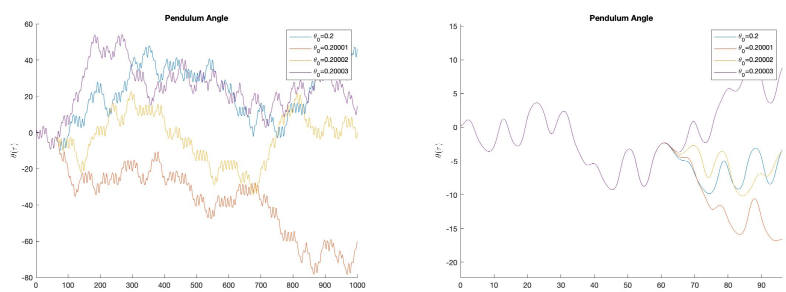


(a) Chaotic result with $A = 1.2$.

(b) The phase space plot clearly indicates there is no periodic motion.

Figure 6: Chaos.

Finally, just for interest's sake, I've plotted the chaotic $A = 1.2$ case for various initial conditions which vary by mere fractions of a percent in figure 7 (I used $\theta_0 = 0.2$, 0.20001, 0.20002, and 0.20003, to be exact).



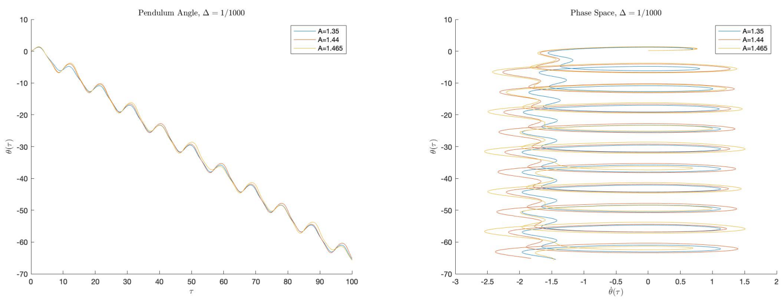
(a) The strong dependence on initial conditions indicates that this solution is indeed chaotic.

(b) The same plot as the above figure, but zoomed in. Interestingly, the trajectories are all almost exactly the same until they suddenly start to differ significantly after about 60 seconds.

Figure 7: The same chaotic result using $A = 1.2$, but with very slightly different initial conditions.

Now I'll check the results using the same values for the variables but with

$A = 1.35$, 1.44, and 1.465. This is shown in figure 8.



(a) The angle plots indicates that the solutions are all quite similar. The persistently decreasing angle indicates that the pendulum is looping around in a circle.

(b) The phase space plots show that the larger A values tend to alter the motion of the pendulum

Figure 8: Plots with $A = 1.35$, 1.44, and 1.465.

The value $A = 1.35$ actually settles on a phase space plot which always has precisely the same peaks and troughs. Larger values of A tend to increase the peaks on one loop, and then decrease it on the next. This will become more clear when we analyze them with the Poincaré maps. The Poincaré maps are generated by plotting only those points which satisfy $\omega t = 2\pi n$, with $n \in \mathbb{Z}$. They are plotted in figure 9.

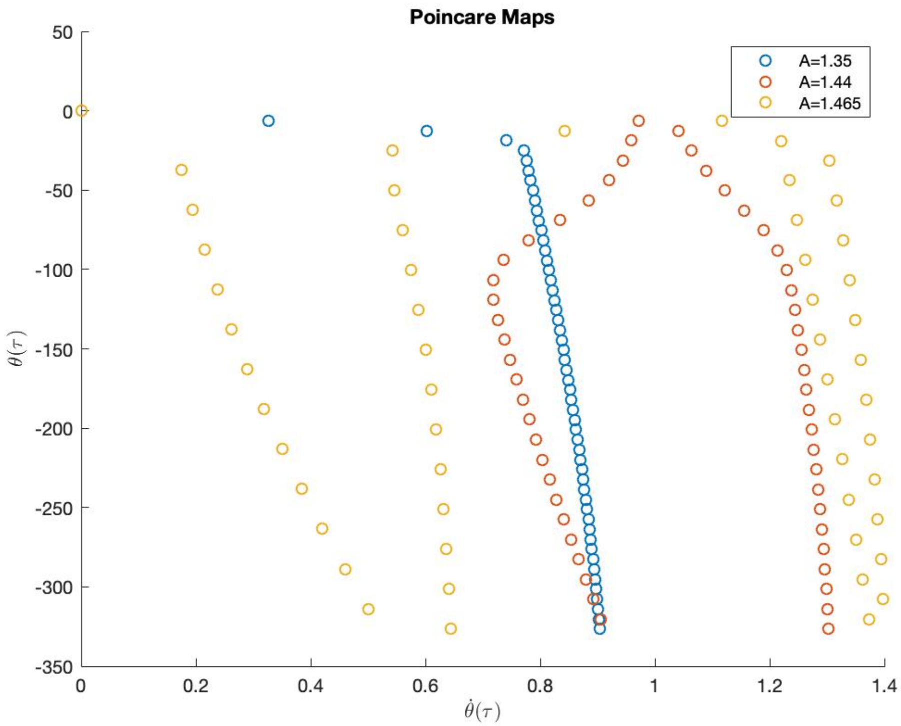


Figure 9: The Poincaré Maps for various values of A, corresponding to the graphs provided in figure 8.

We see that $A = 1.35$ gives a period of 1, while $A = 1.44$ gives a period of 2, and $A = 1.465$ gives a period of 4.

For the sake of completeness, the Poincaré maps for $A = 0.5$ and $A = 1.2$ (the angle and phase space plots of which were plotted earlier) are plotted in figures 10 and 11, respectively.

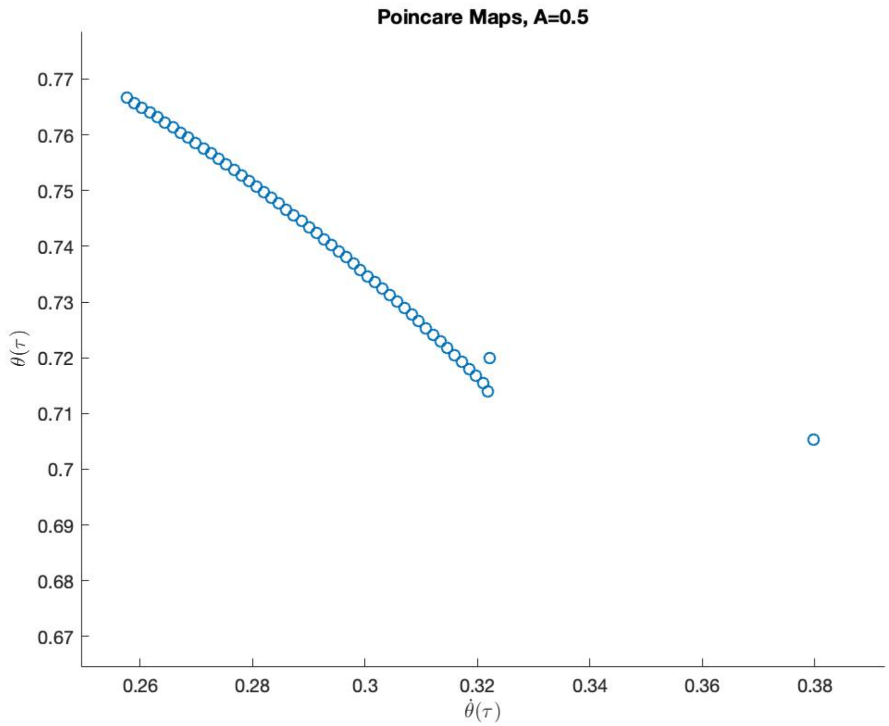


Figure 10: The Poincaré Map for $A = 0.5$, which indicates that after a short period of time, the motion becomes periodic.

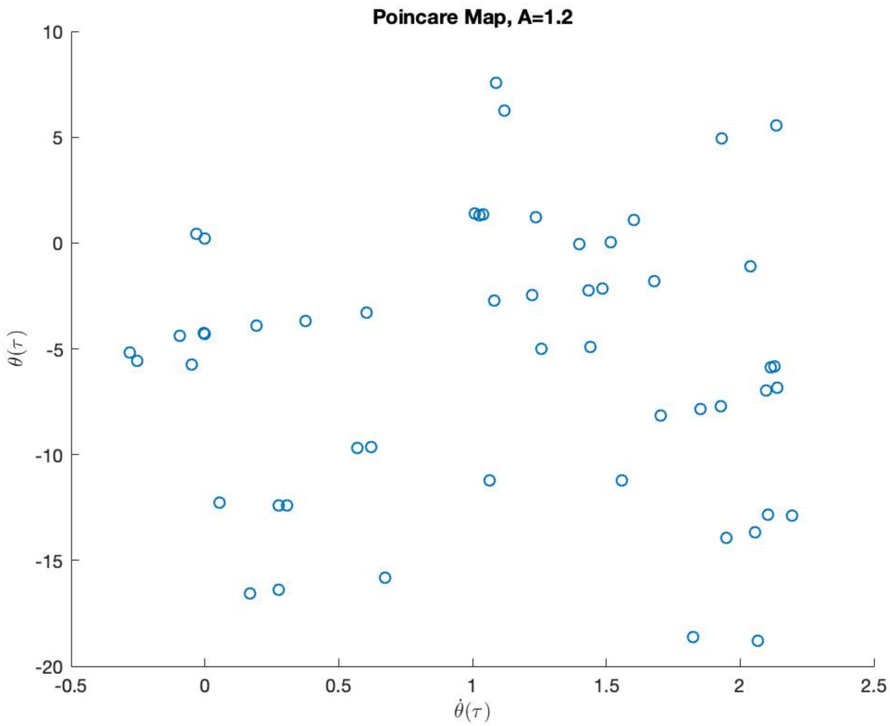


Figure 11: The Poincaré Map for $A = 1.2$ (the chaotic result), which gives an unpredictable result.

3 MATLAB Code

```
clear;clc

%Simulation Parameters
N = 1000000; Delta = 1/1000;
tau_min = 0; tau_max = Delta*N;

%Variables
nuArray = [0.5];
l = 1;
w_true = 2/3;
A_trueArray = [1.2];
m = 1;
g_Earth = 1;
```

```
w_0 = g_Earth/l;

%Dimensionless variables
w = w_true/w_0;

%Couple differential equations in phi and theta.
f = @(phi) phi;
g = @(phi, theta, tau, Q, A) -1/Q*phi - sin(theta) + A*cos(w*
    tau);

%Analytic solutions for small angles
theta = @(tau, A, Q, w) A*((1-w^2)*cos(w*tau) + w/Q*sin(w*tau
    )) / ((1-w^2)^2 + w^2/Q^2);

%Arrays for the equations of motion
tau = linspace(tau_min, tau_max, N);
phiArray = zeros(1,N); thetaArray = zeros(1,N);

nPoincarePoints = floor(tau_max*w / (2*pi));
PoincareTimes = zeros(length(nPoincarePoints), 1);
PoincareStep = floor((2*pi/w)/Delta);
index = 1;
for i = 1:nPoincarePoints
    PoincareTimes(i) = index;
    index = index + PoincareStep;
end

%Initial Conditions for theta (If there's more than one it
    will plot them
%all)
theta_0 = [0.2, 0.2000001];

%Plots various values of A
for h = 1:length(A_trueArray)

    A = A_trueArray(h)/(m*g_Earth);

    %Plots various values of nu
    for r = 1:length(nuArray)

        Q = m*g_Earth/(nuArray(r)*w_0);

        %Inital conditions
        for j = 1:length(theta_0)

```

```
            phiArray(1) = 0; thetaArray = theta_0(j);

            %Implement RK4
            for i = 1:N-1

                k1 = Delta * f(phiArray(i));
                l1 = Delta * g(phiArray(i), thetaArray(i), tau(i), Q, A);

                k2 = Delta * f(phiArray(i) + l1/2);
                l2 = Delta * g(phiArray(i) + l1/2, thetaArray(i) + k1/2, tau(i) + Delta*tau(i)/2, Q, A);

                k3 = Delta * f(phiArray(i) + l2/2);
                l3 = Delta * g(phiArray(i) + l2/2, thetaArray(i) + k2/2, tau(i) + Delta*tau(i)/2, Q, A);

                k4 = Delta * f(phiArray(i) + l3);
                l4 = Delta * g(phiArray(i) + l3, thetaArray(i) + k3, tau(i) + Delta*tau(i), Q, A);

                phiArray(i+1) = 1/6 * (l1+2*l2+2*l3+l4) + phiArray(i);
                thetaArray(i+1) = 1/6 * (k1+2*k2+2*k3+k4) + thetaArray(i);

            end

            figure(1)
            hold on
            plot(tau, thetaArray)
            xlabel('$\tau$', 'Interpreter', 'LaTeX')
            ylabel('$\theta(\tau)$', 'Interpreter', 'LaTeX')
            title('Pendulum Angle, $\Delta=1/1000$', 'Interpreter', 'LaTeX')
            hold off

            figure(2)
            hold on
            plot(phiArray, thetaArray)
            xlabel('$\dot{\theta}(\tau)$', 'Interpreter', 'LaTeX')
            ylabel('$\theta(\tau)$', 'Interpreter', 'LaTeX')
            title('Phase Space, $\Delta=1/1000$', 'Interpreter', 'LaTeX')
```

```
        hold off


        %Poincare Maps

        PoincareTheta = zeros(nPoincarePoints, 1);
        PoincarePhi = zeros(nPoincarePoints, 1);
        for i = 1:nPoincarePoints
            PoincareTheta(i) = thetaArray(PoincareTimes(i));
            PoincarePhi(i) = phiArray(PoincareTimes(i));
        end

        figure(3)
        hold on
        plot(PoincarePhi, PoincareTheta, 'o')
        xlabel('$\dot{\theta}(\tau)$', 'Interpreter', 'LaTeX')
        ylabel('$\theta(\tau)$', 'Interpreter', 'LaTeX')
        title('Poincare Map, A=1.2')
        hold off

        %Energy (Unforced)
        E = zeros(length(tau),1);
        for i = 1:length(tau)
            E(i) = 1 + 0.5*phiArray(i)^2 - cos(thetaArray(i));
        end
        figure(4)
        hold on
        plot(tau, E)
        xlabel('$\tau$', 'Interpreter', 'LaTeX')
        ylabel('$E(\tau)$', 'Interpreter', 'LaTeX')
        title('Energy')
        hold off

        Analytic = zeros(length(tau),1);
        for i = 1:length(tau)
            Analytic(i) = theta(tau(i), A, Q, w);
        end

        %Comparison to analytic solution. Only holds for certain
        %parameters.
        figure(5)
```

```
            hold on
            plot(tau, thetaArray)
            plot(tau, Analytic, 'r--')
            title('Comparison to Analytic Solution')
            xlabel('$\tau$', 'Interpreter', 'LaTeX')
            ylabel('$\theta(\tau)$', 'Interpreter', 'LaTeX')
            hold off
        end
    end
end
```