ME 526 - Homework 5

Use of Mathematica

Due Date:

26th April 2023

(Group № 2)

Member №1: Gabriel Goh

Member №2: Amirreza Rezaeepazhand

Member №3: Naiwen Hu

Member №4: Yury Luzhkov

Introduction:

This homework exercise is dedicated to the exploration of “Mathematica” software. Will will start by writing a short program that calculates digits of π. Then we will proceed with solving differential equation’s using numerical method and producing a plot for different values of damping coefficient. Finally will will work on understanding the non approximate solution for

Part 1: Calculation of π

We are going to calculate π using a method developed by Archimedes. The derivation is provided in the HW manual. Where S approaches π as n approaches infinity.

$$S_{n+1} = 2^{n+2}\left[\left(\frac{S_n}{2^{n+2}}\right)^2 + \left(1 - \left[1 - \left(\frac{S_n}{2^{n+2}}\right)^2\right]^{1/2}\right)^2\right]^{1/2}$$

$$\lim_{n\to\infty} S_n = \pi$$

Please see our code bellow:

```
In[15]:= Sn1[Sn_, n_] = 2^(n + 2) * ((Sn / 2^(n + 2))^(2) + (1 - (1 - (Sn / 2^(n + 2))^(2))^(1 / 2))^(2))^(1 / 2)
```

The line above defines function which calculates concurrent Sn

Out[15]= $2^{2+n}\sqrt{2^{-4-2n}\,\mathrm{Sn}^2 + \left(1 - \sqrt{1 - 2^{-4-2n}\,\mathrm{Sn}^2}\right)^2}$

```
In[16]:= For[i = 0; s1 = 2 * (2)^(1 / 2), i < 10, i++, s1 = Sn1[s1, i]; Print[N[s1, 20]]]
```

The line above is a for loop which uses the formula to calculate 10 values of Pi

```
3.0614674589207181738
3.1214451522580522856
3.1365484905459392638
3.1403311569547529123
3.1412772509327728681
3.1415138011443010763
3.1415729403670913841
3.1415877252771597006
3.1415914215111999740
3.1415923455701177423
```

```
In[17]:= N[Pi, 20]
```

The actual Pi value up to 19 Decimal places.

```
Out[17]= 3.1415926535897932385
```

Figure 1: Mathematica Pi code

```
In[74]:= x = Table[i, {i, 0, 9}]
Out[74]= {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

In[75]:= y = {N[2 * (2)^(1/2), 20]}
Out[75]= {2.8284271247461900976}

In[76]:= For[i = 0; s1 = 2 * (2)^(1/2), i < 9, i++, s1 = Sn1[s1, i]; y = Append[y, N[s1, 20]]]

In[77]:= y
Out[77]= {2.8284271247461900976, 3.0614674589207181738, 3.1214451522580522856, 3.1365484905459392638, 3.1403311569547529123,
         3.1412772509327728681, 3.1415138011443010763, 3.1415729403670913841, 3.1415877252771597006, 3.1415914215111999740}

In[80]:= ListPlot[Transpose@{x, y}, Joined → True, PlotStyle → Hue[.89], AxesLabel → {"n", "Sn"}, PlotRange → All, BaseStyle → {FontFamily → "Times", FontSize → 12},
          PlotLabel → "Sn approching  Pi"]
```

Figure 2: Mathematica Sn plot code

Out[80]=

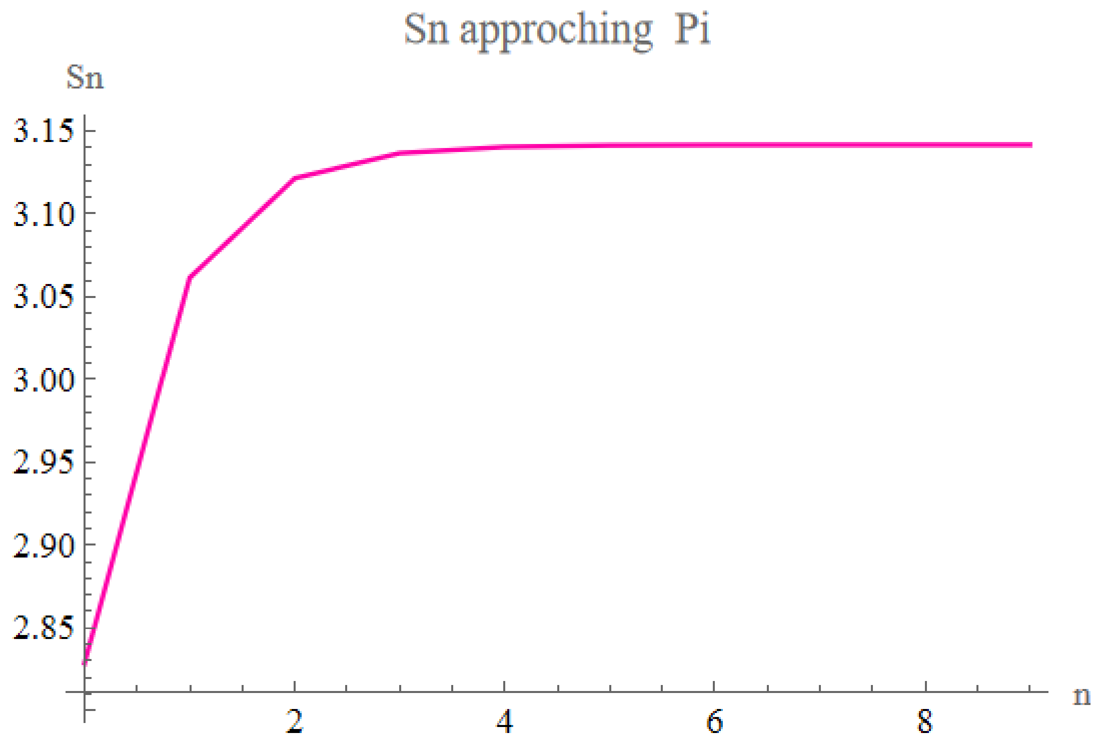

Figure 3: Mathematica Sn Plot

```
yerror = y - Pi
{-0.3131655288436031409, -0.0801251946690750646, -0.0201475013317409529, -0.0050441630438539746, -0.0012614966350403261,
 -0.0003154026570203704, -0.0000788524454921621, -0.0000197132227018543, -4.9283126335378×10^-6, -1.2320785932645×10^-6}

ListPlot[Transpose[{x, yerror}], Joined → True, PlotStyle → Hue[.89], AxesLabel → {"n", "Sn"}, PlotRange → All, BaseStyle → {FontFamily → "Times", FontSize → 12},
 PlotLabel → "Approximation Error"]
```

Figure 4: Mathematica Error Plot code

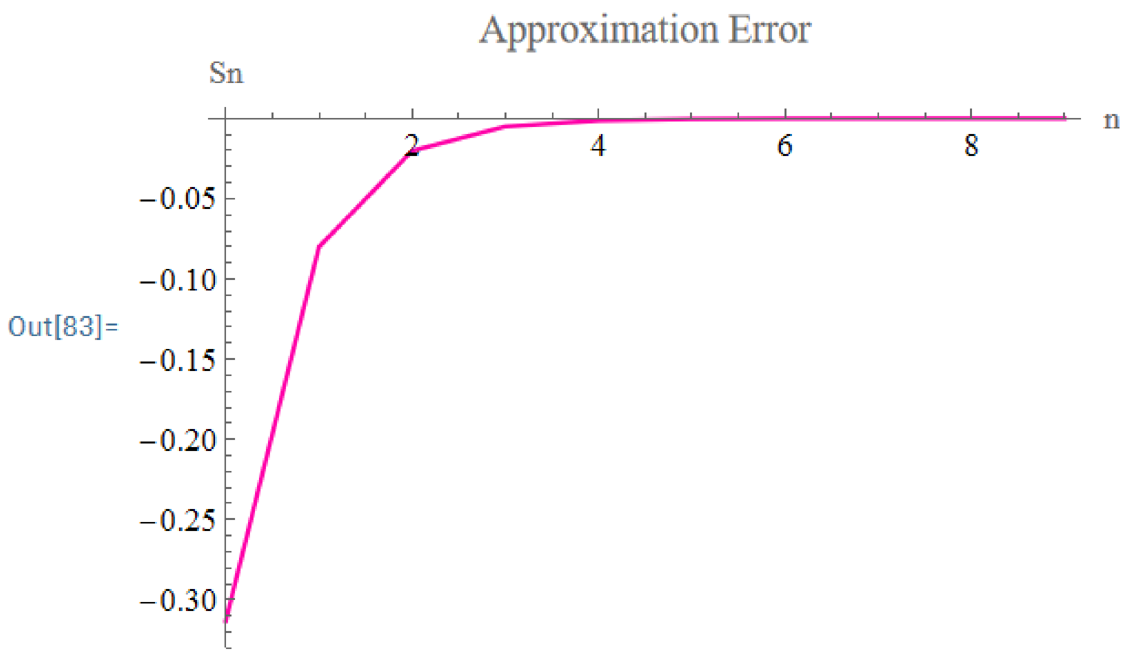

Figure 5: Mathematica Error Plot

Part 2: 1D Oscillator

We have differential equation:

$$m\frac{d^2x}{dt^2} = -m\omega_0^2 x - b\frac{dx}{dt} + F_0 cos(\omega t)$$

$$m = 1, \omega_0 = 3, F_0 = 1 \; for \; b = \{0,1,2,4,8\}$$

Equation becomes:

$$\frac{d^2x}{dt^2} = -9x - \{0,1,2,4,8\}\frac{dx}{dt} + cos(\omega t)$$

For the purpose of simplicity we will only run program for b = 1

Plot Single:

In[333]:=
```
Plot[Evaluate[x[t] /. solutions[[4]]], {t, 500, 550}, PlotPoints → 1000, PlotRange → All, Frame → True, FrameLabel → {"time", "displacement"},
 ImageSize → {600, 400}, PlotStyle → Hue[.9], FrameTicks → {{350, 500}, {-2, 0, 2}, {}, {}}, BaseStyle → {FontFamily → "Times", FontSize → 16}]
```

Out[333]=

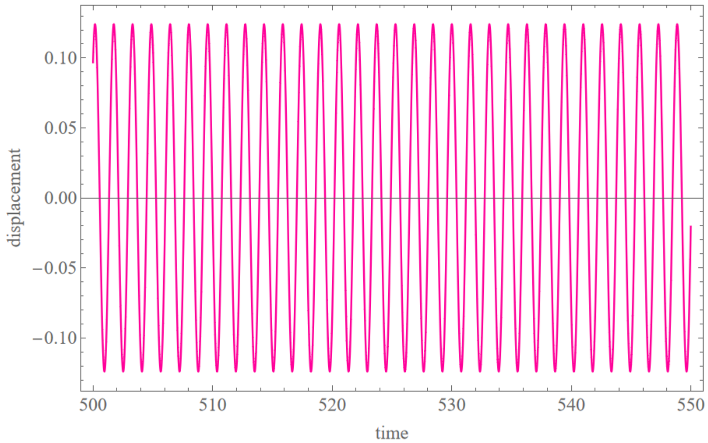

Figure 6: Single Plot

In[350]:=
```
Clear["Global`*"]
omegas = Table[omega, {omega, 0.1, 9, 0.1}];
```

In[352]:=
```
odes = Table[x''[t] == -9*x[t] - 1*x'[t] + Cos[omega*t], {omega, 0, 9, 0.1}];
Length[odes]
```

Out[353]= 91

In[354]:=
```
solutions = {};
For[i = 2, i ≤ Length[odes], i++, var = NDSolve[{odes[[i]], x[0] == 1, x'[0] == 0}, x, {t, 500, 550}, MaxSteps → 5000, Method → ExplicitRungeKutta];
  solutions = Append[solutions, var]];
solutions;
amplitudes = {};
For[i = 1, i ≤ Length[solutions], i++,
  answers = Table[N[Evaluate[x[t] /. solutions[[i]], 10]], {t, 500, 550, 0.001}];
  a = Max[answers];
  amplitudes = Append[amplitudes, a]];
Length[amplitudes]
```

Plot Magnitude:

In[360]:=
```
ListPlot[Transpose@{omegas, amplitudes}, Joined → True, PlotStyle → Hue[.89], PlotRange → All, BaseStyle → {FontFamily → "Times", FontSize → 12},
 PlotLabel → "b=1", AxesLabel → {"ω", "Amplitude"}]
```

Figure 7: Code

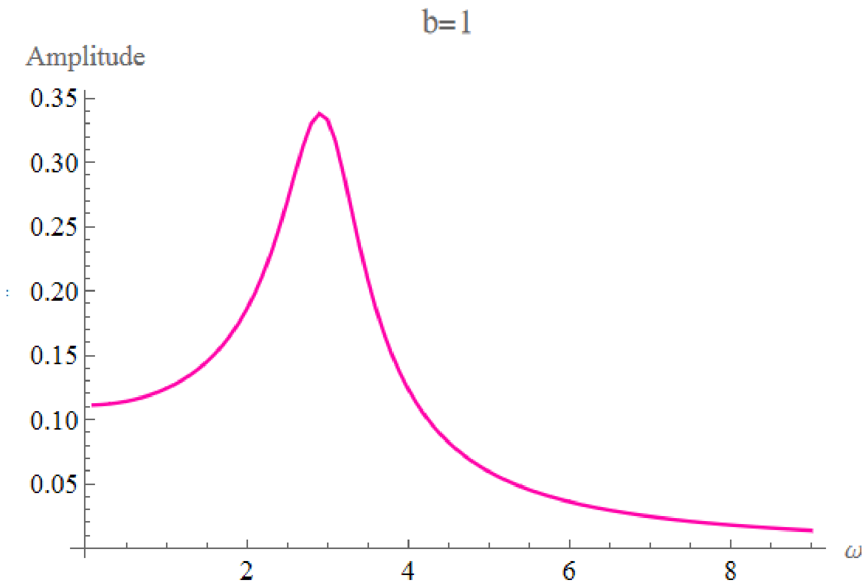

Figure 8: Plot

b is the damping coefficient in the system. If b is small, the system will show resonance at a specific frequency where the amplitude of the displacement reaches maximum. As b increases, the peak of the resonance curve will shift slightly, meanwhile the amplitude of the displacement at that frequency will decrease largely.

If b becomes very large, the system will no longer show resonance, the amplitude of the displacement will decrease gradually with increasing frequency. This is because the damping force will dominate over the restoring force.

When ω (driving frequency) approaches ω0 (natural frequency), the system will show resonance because the driving force is able to supply energy to the system at the same frequency, causing amplitude to increase to a large extent. As ω gets closer to ω0, the shape of the plot becomes narrower and amplitude gets higher. When ω is equal to ω0, the amplitude of displacement can be infinite.

```
In[395]:= ParametricPlot[Evaluate[{y[t], y'[t]} /. sol], {t, 10, 500}, PlotStyle → Hue[.9], ImageSize → {600, 400}, BaseStyle → {FontFamily → "Times", FontSize → 16}]
```

Out[395]=

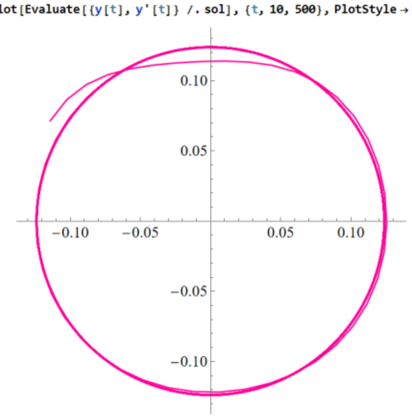

Figure 9: Code

```
In[396]:= Plot[Evaluate[y[t] /. sol], {t, 400, 500}, PlotPoints → 1000, PlotRange → All, Frame → True, FrameLabel → {"time", "displacement"}, ImageSize → {600, 400},
  PlotStyle → Hue[.9], FrameTicks → {{350, 500}, {-2, 0, 2}, {}, {}}, BaseStyle → {FontFamily → "Times", FontSize → 16}]
```

Out[396]=

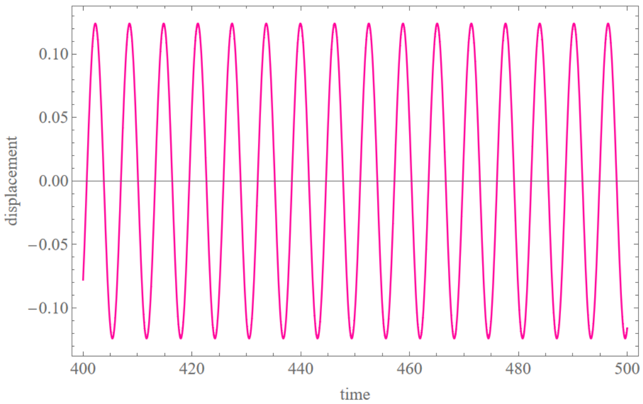

Figure 10: Plot

Chosen omega was 1. As it provided stable state. The plots above illustrate Parametric plot for this value and displacement vs time plots.

Part 3: Nonlinear pendulum (Amirreza)

As $g/L = 1$, we would have: $\theta'' = -\sin(\theta)$ => $\theta'' + \sin(\theta) = 0$.

Here's the mathematica code used for solving the equation and finding the period of oscillation; $\theta'(t = 0) = 0$. In this code we started with $\theta(t = 0) = 0.05$.

```
eqn = {y''[t] + Sin[y[t]] == 0, y[0] == 0.05, y'[0] == 0};
sol = NDSolve[eqn, y, {t, 0, 50}];
y[t_] := Evaluate[y[t] /. sol[[1]]];
periods = Table[t /. FindRoot[y[t] == y[0] && y'[t] > 0, {t, i}], {i, 0, 40, 5}];
period = Mean[Differences[periods]];
Plot[y[t], {t, 0, 50}, AxesLabel -> {"t", "y(t)"}]
Print[ period]
```

Here are the results of the period duration with different start points:

Tetha0	T		Tetha0	T
0.05	6.2843		0.7	6.4813
0.1	6.2873		0.8	6.5443
0.15	6.2922		0.9	6.617
0.2	6.2991		1	6.7001
0.25	6.308		1.1	6.7943
0.3	6.3189		1.2	6.9004
0.35	6.3318		1.3	7.0194
0.4	6.3467		1.4	7.1525
0.45	6.3638		1.5	7.301
0.5	6.3829		$\pi/2$	7.4164
0.6	6.4277			

And in a plot:

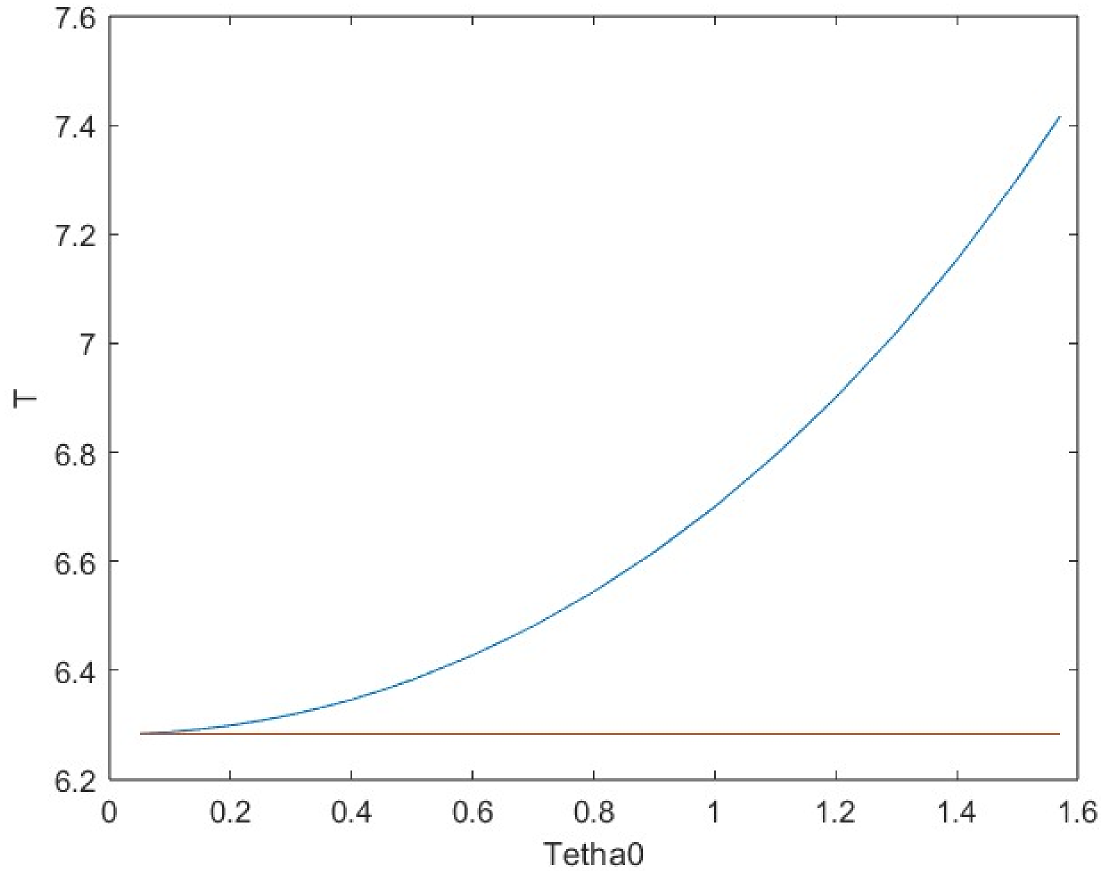

Figure 11. Period(T) with respect to starting point($\theta(t = 0)$)

The blue curve in the plot shows the period duration with different starting points and the red line shows the value of 2π. As we can see, when we start the pendulum with a small angle, the value of period duration is approximately equal to 2π. However, as we increase the angle, the value of period duration increases dramatically.

Conclusion:

We can conclude that mathematica is a very powerful tool and that unsolvable analytically equation can be solved here using just a few lines of codes to what ever precision needed. The precision was investigated in Part 1. The-non linear applications were investigated in Part 2&3.