Math 128A: Programming Assignment 3

Michael Pham

Summer 2024

Problems

Problem 1.1 3
Problem 1.2 4
Problem 1.3 9

Problem 1.1. Assuming the lengths of the bars are 1, the masses at the end of the bars are 1, and that the constant of gravity is 1, the equations of motion for the double pendulum can be written:

$$\theta_1'' = \frac{-3\sin\theta_1 - \sin(\theta_1 - 2\theta_2) - 2\sin(\theta_1 - \theta_2)\cdot(\theta_2'^2 + \theta_1'^2\cos(\theta_1 - \theta_2))}{3 - \cos(2\theta_1 - 2\theta_2)}$$
$$\theta_2'' = \frac{2\sin(\theta_1 - \theta_2)(2\theta_1'^2 + 2\cos\theta_1 + \theta_2'^2\cos(\theta_1 - \theta_2))}{3 - \cos(2\theta_1 - 2\theta_2)}$$

Rewrite these two equations as a system of first-order equations by introducing $\omega_1 = \theta_1'$ and $\omega_2 = \theta_2'$. Then, write a MATLAB function `fpend.m` of the form `function ydot = fpend(y)`, which evaluates $f(y)$.

Solution. First, we rewrite the equation as follow:

$$\theta_1' - \omega_1$$
$$\theta_2' = \omega_2$$
$$\omega_1' = \frac{-3\sin\theta_1 - \sin(\theta_1 - 2\theta_2) - 2\sin(\theta_1 - \theta_2)\cdot(\omega_2^2 + \omega_1^2\cos(\theta_1 - \theta_2))}{3 - \cos(2\theta_1 - 2\theta_2)}$$
$$\omega_2' = \frac{2\sin(\theta_1 - \theta_2)(2\omega_1^2 + 2\cos\theta_1 + \omega_2^2\cos(\theta_1 - \theta_2))}{3 - \cos(2\theta_1 - 2\theta_2)}$$

And below is the code for `fpend.m`:

Code: `fpend.m` code.

```
function ydot = fpend(y)
% theta1
t1 = y(1);
% theta2
t2 = y(2);
% w1
w1 = y(3);
% w2
w2 = y(4);

% tdot1
tdot1 = w1;
% tdot2
tdot2 = w2;
% wdot1
wdot1 = (-3*sin(t1)-sin(t1 - 2*t2)- 2*sin(t1 - t2)*(w2^2 + w1^2 * cos(t1 - t2)))/(3-cos(2*t1- 2*t2));
% wdot2
wdot2 = (2*sin(t1-t2)*(2*w1^2 + 2*cos(t1) + w2^2 * cos(t1-t2)))/(3-cos(2*t1 - 2*t2));

ydot = [tdot1, tdot2, wdot1, wdot2];
end
```

■

Problem 1.2. Implement a fourth-order Runge-Kutta method that integrates the system $y' = f(y)$ from $t = 0$ to $t = 100$ with stepsize $h = 0.05$. Use the four initial conditions described in the table below.

Case	$\theta_1(0)$	$\theta_2(0)$	$\omega_1(0)$	$\omega_2(0)$
1	1	1	0	0
2	π	0	0	10^{-10}
3	2	2	0	0
4	2	$2 + 10^{-3}$	0	0

For each case, plot the function $\theta_2(t)$ versus time.

Solution. To begin with, we provide the code for the RK4 function below:

Code: `pa3_rk4.m` code.

```
function [t, w] = pa3_rk4(a, b, h, y)

% a, b = endpoints
% h = step-size
% y = [th1, th2, w1, w2]

% sets t to initial value
t = a;

% gets initial values
ydot = fpend(y);

% assigns initial values
w1 = y(1);
w2 = y(2);
w3 = y(3);
w4 = y(4);

% calculate N
N = (b-a)/h;

% array for theta_2
counter = 1;
th2 = zeros(1, N);
th2(counter) = w2;

fprintf(' t         theta_1       theta_2        w_1         w_2    \n');
fprintf('-----------------------------------------------------------\n');
fprintf('%.2f  %12.8f  %12.8f  %12.8f  %12.8f\n', t, w1, w2, w3, w4);

% calculate k1
for i = a:h:(b-h)
k1 = h * fpend([w1, w2, w3, w4]);
k2 = h * fpend([w1 + 0.5*k1(1), w2 + 0.5*k1(2), w3 + 0.5*k1(3), w4 + 0.5*k1(4)]);
k3 = h * fpend([w1 + 0.5*k2(1), w2 + 0.5*k2(2), w3 + 0.5*k2(3), w4 + 0.5*k2(4)]);
k4 = h * fpend([w1 + k3(1), w2 + k3(2), w3 + k3(3), w4 + k3(4)]);

w1 = w1 + (k1(1) + 2*k2(1) + 2*k3(1) + k4(1))/6;
w2 = w2 + (k1(2) + 2*k2(2) + 2*k3(2) + k4(2))/6;
w3 = w3 + (k1(3) + 2*k2(3) + 2*k3(3) + k4(3))/6;
w4 = w4 + (k1(4) + 2*k2(4) + 2*k3(4) + k4(4))/6;

t = t + h;
counter = counter + 1;

th2(counter) = w2;
```

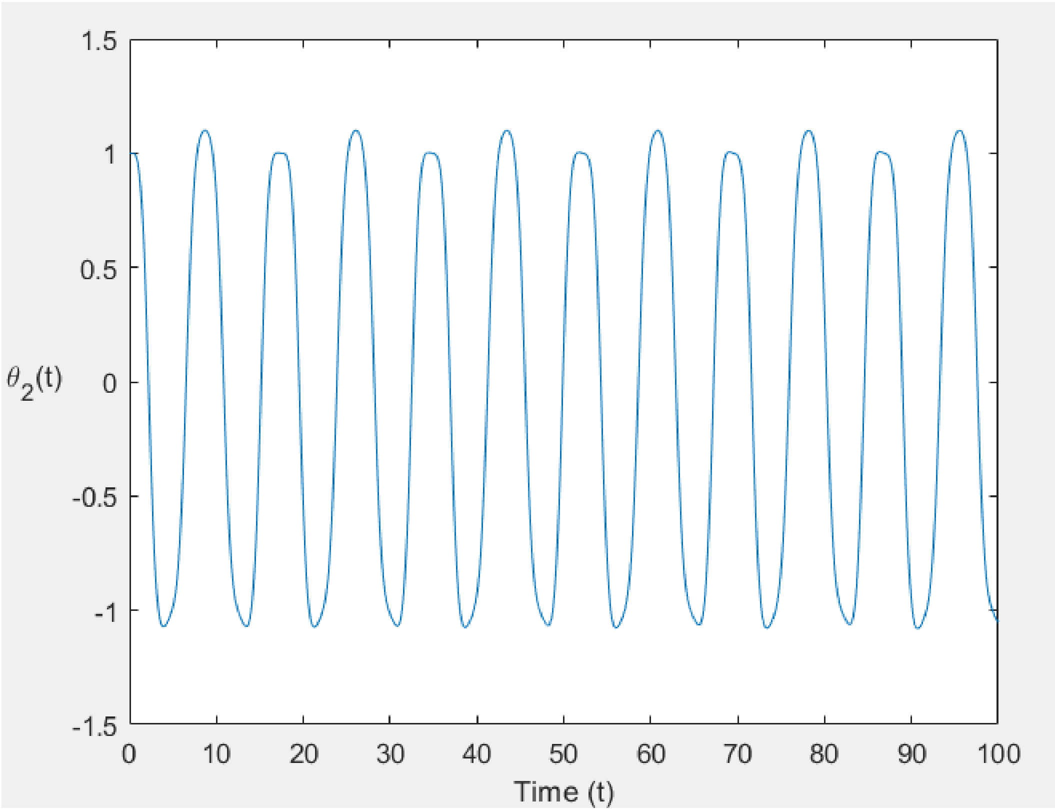
```
if (mod(counter, 200) == 1)
fprintf('%.2f  %12.8f  %12.8f  %12.8f  %12.8f\n', t, w1, w2, w3, w4);
end
end

plot(0:0.05:100, th2);
w = [w1, w2, w3, w4];
end
```

Next, we look at the four different cases outlined:

Code: Case 1 table.

```
>> y = [1, 1, 0, 0];
a = 0;
b = 100;
h = 0.05;
[t, w] = pa3_rk4(a, b, h, y)
t         theta_1       theta_2        w_1          w_2
------------------------------------------------------------
0.00    1.00000000    1.00000000    0.00000000    0.00000000
10.00    0.43847219    0.80860398   -0.58460006   -0.62730809
20.00   -0.20450368   -0.54965041   -0.47238912   -1.01545324
30.00   -0.93623699   -1.01717090   -0.24623498   -0.10097749
40.00   -0.68085588   -0.92668690    0.58387836    0.27402725
50.00    0.01805359    0.15168304    0.35809114    1.25881255
60.00    0.76501476    1.03045986    0.46879170    0.20916509
70.00    0.86649467    0.98829902   -0.44900116   -0.04812281
80.00    0.13463285    0.29047570   -0.43195714   -1.13760524
90.00   -0.53464669   -0.95936675   -0.59853000   -0.41848814
100.00   -0.95284989   -1.05206811    0.20644310   -0.05002658

t =

100.0000


w =

-0.9528   -1.0521    0.2064   -0.0500
```

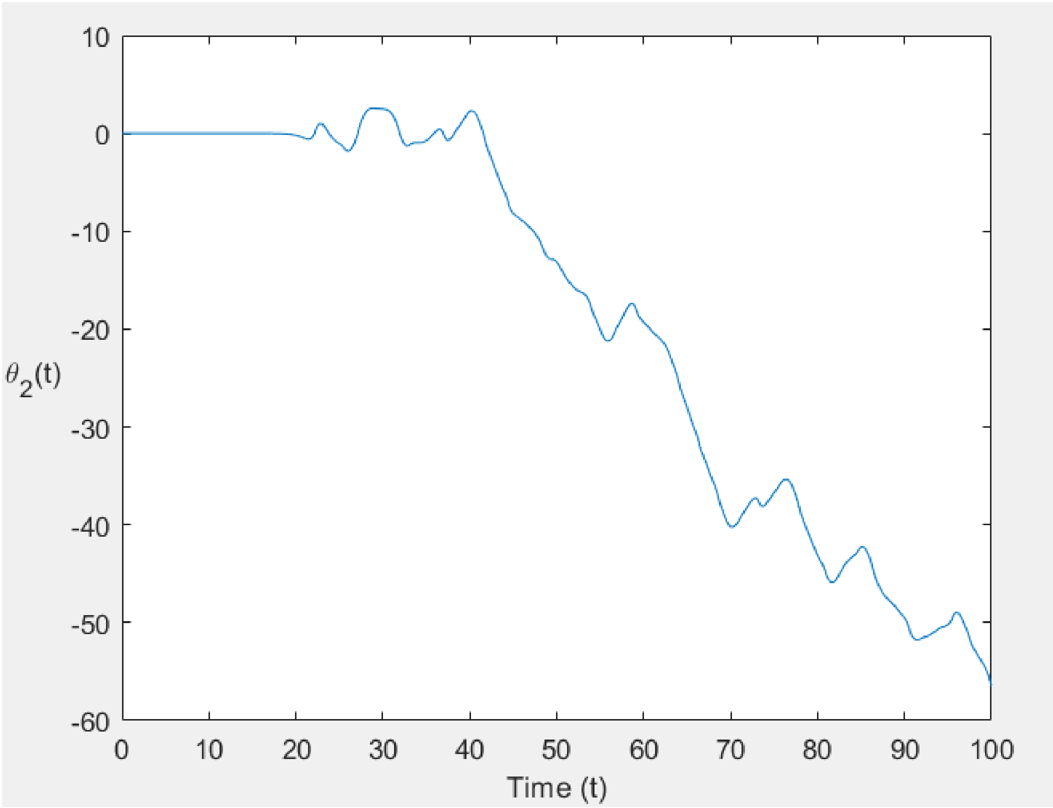
And the graph constructed is seen below:



Next, for the second case:

Code: Case 2 table.

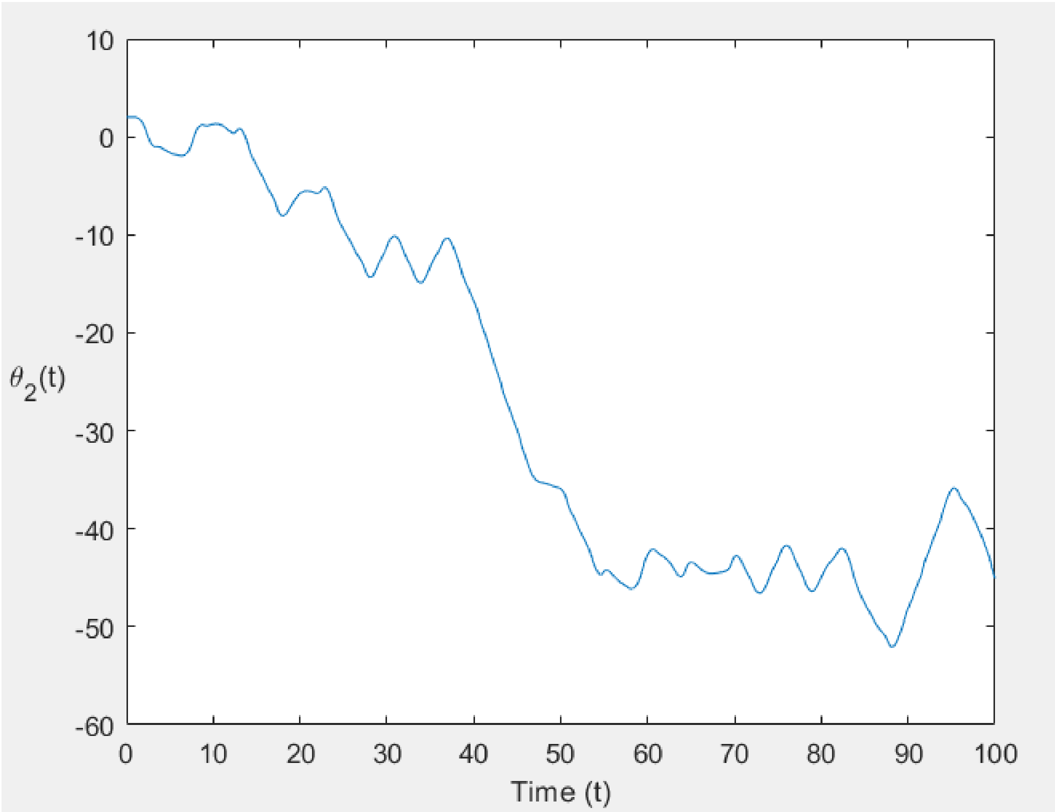
```
>> y = [pi, 0, 0, 10^(-10)];
a = 0;
b = 100;
h = 0.05;
[t, w] = pa3_rk4(a, b, h, y)
t          theta_1        theta_2         w_1          w_2
-----------------------------------------------------------
0.00     3.14159265     0.00000000     0.00000000     0.00000000
10.00     3.14159048    -0.00000127    -0.00000258    -0.00000151
20.00     2.82652174    -0.18225021    -0.36938068    -0.20819032
30.00     1.56468852     2.50502124    -0.42746975    -0.08491447
40.00     0.93837915     2.25829202    -1.25216627     0.54513375
50.00    -1.20785253   -13.15226318    -1.01760665    -1.09580021
60.00     1.84424925   -19.27685961     0.76525912    -0.85169218
70.00    -0.34868718   -40.16116165    -1.54235743    -0.52281970
80.00    -1.70455472   -43.11211278    -0.31937476    -1.88765946
90.00    -1.23855070   -49.51645070     1.05321419    -1.22478220
100.00     1.12966733   -56.43191547     1.09775705    -2.48213182

t =

100.0000


w =

1.1297   -56.4319    1.0978   -2.4821
```



For the third case, we have:

Code: Case 3 table.

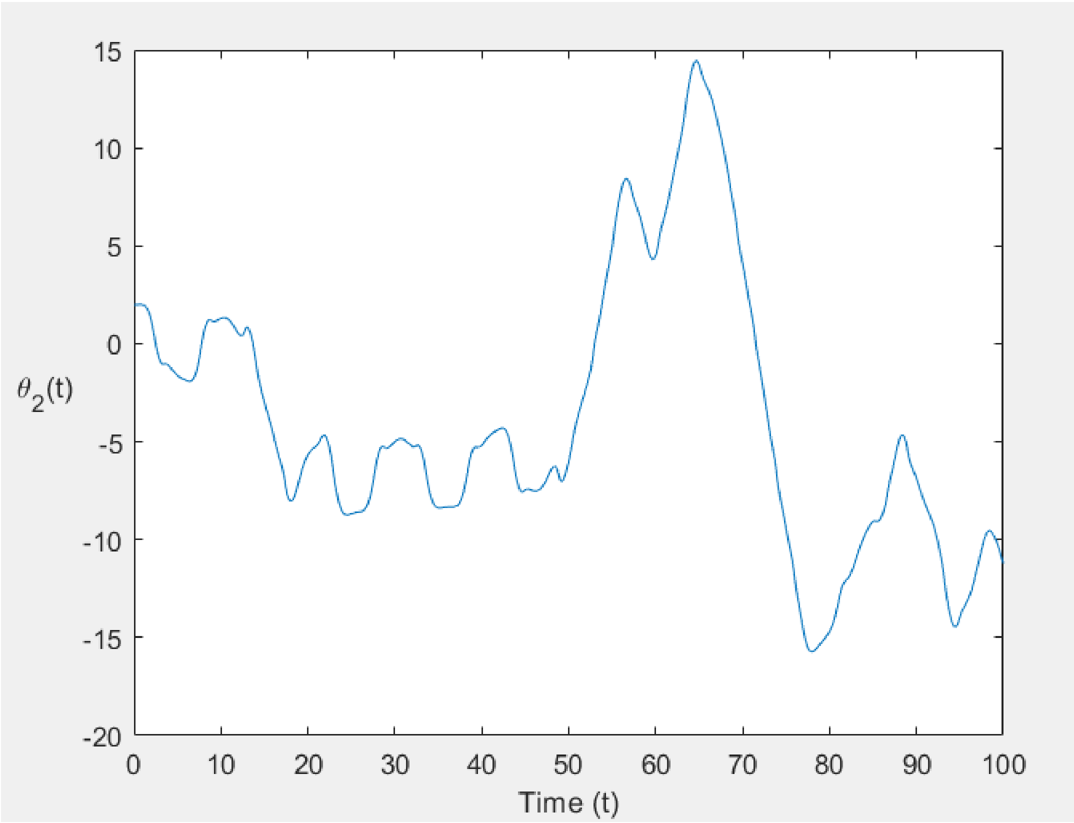
```
>> y = [2, 2, 0, 0];
a = 0;
b = 100;
h = 0.05;
[t, w] = pa3_rk4(a, b, h, y)
t          theta_1        theta_2        w_1          w_2
------------------------------------------------------------
0.00     2.00000000     2.00000000     0.00000000     0.00000000
10.00     2.25131254     1.28850976     0.44721104     0.19449473
20.00     2.83699361    -5.80010237     0.12368645     0.74182466
30.00     1.48292733   -11.21830948    -1.64487000     2.36069826
40.00    -0.15830099   -16.76625594     0.48085015    -1.95747798
50.00    -0.66335483   -35.90073363     1.30474317    -0.66390570
60.00     0.09875555   -42.64116679     1.12134061     1.76973230
70.00    -6.41464953   -42.86202600    -1.94151870     1.00677028
80.00    -4.42224962   -44.92668658     1.38230620     1.93656068
90.00    -5.81308231   -48.10676267    -0.30227007     2.14721428
100.00    -5.75650085   -45.07378010    -0.53230203    -2.53533024

t =

100.0000


w =

-5.7565  -45.0738   -0.5323   -2.5353
```



And for Case 4, we see that:

Code: Case 4 table.

```
>> y = [2, 2 + 10^(-3), 0, 0];
a = 0;
b = 100;
h = 0.05;
[t, w] = pa3_rk4(a, b, h, y)
t          theta_1        theta_2         w_1          w_2
-------------------------------------------------------------
0.00     2.00000000     2.00100000     0.00000000     0.00000000
10.00     2.24818826     1.29353869     0.44470389     0.20082079
20.00     2.59312438    -5.68694672    -0.13769572     0.78270744
30.00     2.22173496    -4.99803075     0.38039021     0.40640172
40.00     1.74532911    -5.17985229     0.93799551     0.47728451
50.00     1.53402464    -6.01539157     0.40288835     1.94399061
60.00    -0.28164431     4.51137918    -1.56834926     1.14400155
70.00     0.10989773     4.13338767    -0.02210913    -2.33250426
80.00    -1.34712724   -14.70195209     0.67511959     0.79813498
90.00     2.07735342    -6.86590344     0.25699328    -1.23515442
100.00     1.88478012   -11.21456672     0.17457708    -1.43550388

t =

100.0000


w =

1.8848   -11.2146     0.1746    -1.4355
```



■

Problem 1.3. Run Case 1 in Problem 1.2 with the five stepsizes $h = 0.05/2^{k-1}$, $k = 1, 2, 3, 4$, and $h = 0.001$. Compute the value of $\theta_2(t = 100)$ for each stepsize. Consider the last result the exact solution, and plot the four errors as a function of h in a loglog-plot. Estimate the order of convergence from the slope.

Solution. We do the following to construct our loglog plot:

Code: Constructing the loglog plot.

```
>> ths = zeros(1, 5);
ks = [1, 2, 3, 4];
hs = [0.05./2.^(ks - 1)];
y = [1, 1, 0, 0];
[ys, ws] = pa3_rk4(a, b, hs(1), y);
ths(1) = ws(2);
[ys, ws] = pa3_rk4(a, b, hs(2), y);
ths(2) = ws(2);
[ys, ws] = pa3_rk4(a, b, hs(3), y);
ths(3) = ws(2);
[ys, ws] = pa3_rk4(a, b, hs(4), y);
ths(4) = ws(2);
[ys, ws] = pa3_rk4(a, b, 0.001, y);
ths(5) = ws(2);
loglog([hs(1), hs(2), hs(3), hs(4)], [abs(ths(1) - ths(5)), abs(ths(2) - ths(5)), abs(ths(3) - ths(5)),
    abs(ths(4) - ths(5))], 'o-')
xlabel("Stepsize h")
xticks([hs(4), hs(3), hs(2), hs(1)])
```

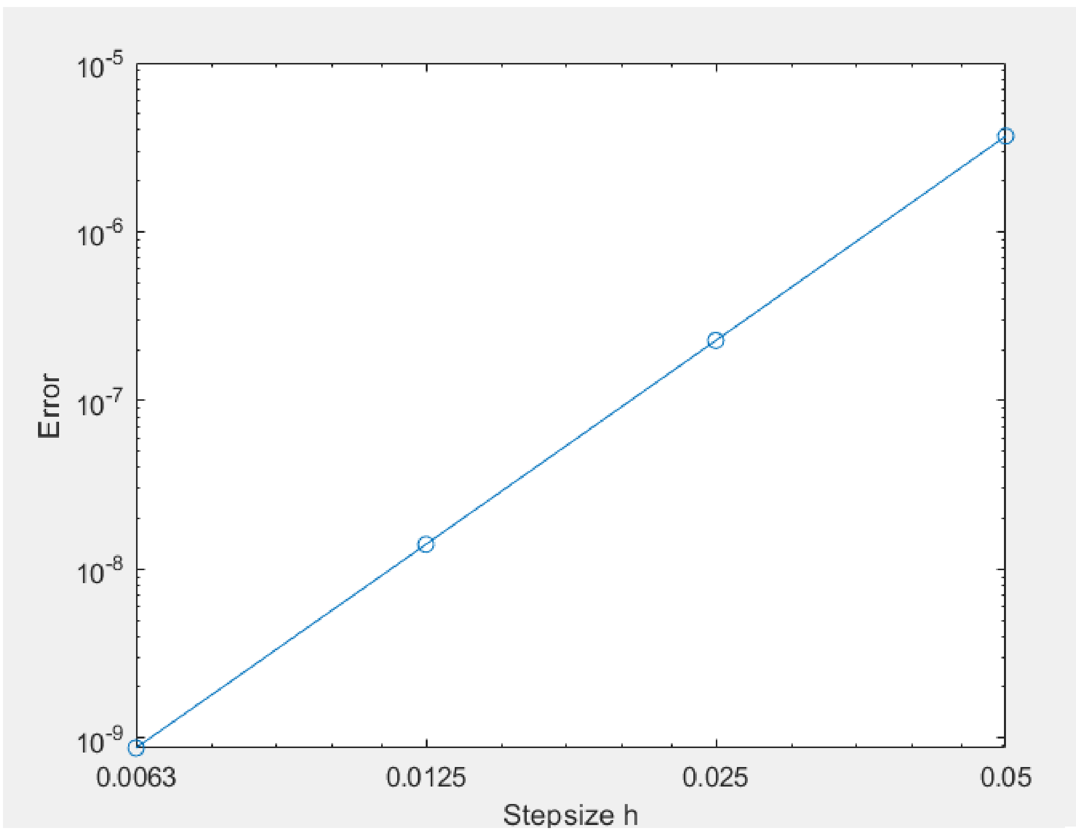
```
ylabel("Error")
```

Furthermore, we have:

Code: Values for thetas

```
>> ths

ths =

-1.052068113668714   -1.052071573182922   -1.052071785738407   -1.052071798892810   -1.052071799764450
```

Then, this yields us the following graph:



Looking at this, we see the most accurate points are at $h = 0.00625$ and $h = 0.0125$. Then, with this in mind, we see that we have a slope of `(log(abs(ths(3) - ths(5))) - log(abs(ths(4) - ths(5))))/( log(hs(3)) - log(hs(4)) )`, which equals to approximately 4. ■