Joseph Ingenito

5/10/19

Project

Part I

1)

(a)

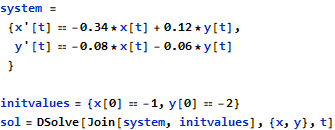






(b)

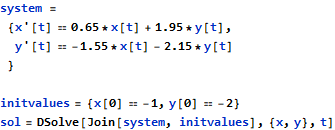






(c)





{{x -> Function[{t}, (-1. +

0. i) e^(-0.75 t) ((1. + 0. i) Cos[

1.03078 t] + (5.14176 + 3.1486\*10^-17 i) Sin[1.03078 t])],

y -> Function[{t}, -2. e^(-0.75 t) (1. Cos[

1.03078 t] - (2.11006 + 5.50981\*10^-17 i) Sin[1.03078 t])]}}

I couldn’t preserve Mathematica’s Formatting for this solution because the output is too long.

2)

For each system I made functions that also return the true solution found on Mathematica so making and editing the plots can be done easily.

(a)

function [t,x,y] = EulSystem\_1(h, t0, x0, y0, tf)

k = 1;

t(k) = t0;

x(k) = x0;

y(k) = y0;

while((t(k) + h <= tf + 10^-10))

t(k+1) = t(k) + h;

x(k+1) = x(k) + h\*(-0.5\*x(k) + 0.6\*y(k));

y(k+1) = y(k) + h\*(-0.4\*x(k) + 0.9\*y(k));

k = k + 1;

end

end

function [t,x,y] = TrueSolution\_1()

t = 0:0.1:4;

x = -exp(0.7.\*t);

y = -2.\*exp(0.7.\*t);

end

(b)

function [t,x,y] = EulSystem\_2(h, t0, x0, y0, tf)

k = 1;

t(k) = t0;

x(k) = x0;

y(k) = y0;

while((t(k) + h <= tf + 10^-10))

t(k+1) = t(k) + h;

x(k+1) = x(k) + h\*(-0.34\*x(k) + 0.12\*y(k));

y(k+1) = y(k) + h\*(-0.08\*x(k) - 0.06\*y(k));

k = k + 1;

end

end

function [t,x,y] = TrueSolution\_2()

t = 0:0.1:4;

x = -exp(-0.1.\*t);

y = -2.\*exp(-0.1.\*t);

end

(c)

function [t,x,y] = EulSystem\_3(h, t0, x0, y0, tf)

k = 1;

t(k) = t0;

x(k) = x0;

y(k) = y0;

while((t(k) + h <= tf + 10^-10))

t(k+1) = t(k) + h;

x(k+1) = x(k) + h\*(0.65\*x(k) + 1.95\*y(k));

y(k+1) = y(k) + h\*(-1.55\*x(k) - 2.15\*y(k));

k = k + 1;

end

end

function [t,x,y] = TrueSolution\_3()

t = 0:0.1:4;

x = -exp(-0.75.\*t).\*(cos(1.03078.\*t) + 5.14176.\*sin(1.03078.\*t));

y = -2.\*exp(-0.75.\*t).\*(cos(1.03078.\*t)-2.11006.\*sin(1.03078.\*t));

end

3)

function CreateSubplots(t1,x1,y1,t2,x2,y2,tvec,xvec,yvec)

ax1 = subplot(4,2,1);

ax2 = subplot(4,2,2);

ax34 = subplot(4,2,[3,4]);

ax5 = subplot(4,2,5);

ax6 = subplot(4,2,6);

ax78 = subplot(4,2,[7,8]);

plot(ax1,t1,x1,"blue",tvec,xvec,"black")

title(ax1,"x(t) vs t with h=0.25")

xlabel(ax1,"t Axis")

ylabel(ax1,"x(t) Axis")

plot(ax2,t1,y1,"blue",tvec,yvec,"black")

title(ax2,"y(t) vs t with h=0.25")

xlabel(ax2,"t Axis")

ylabel(ax2,"y(t) Axis")

plot(ax34,x1,y1,"blue",xvec,yvec,"black")

title(ax34,"y(t) vs x(t) with h=0.25")

xlabel(ax34,"x(t) Axis")

ylabel(ax34,"y(t) Axis")

plot(ax5,t2,x2,"blue",tvec,xvec,"black")

title(ax5,"x(t) vs t with h=0.1")

xlabel(ax5,"t Axis")

ylabel(ax5,"x(t) Axis")

plot(ax6,t2,y2,"blue",tvec,yvec,"black")

title(ax6,"y(t) vs t with h=0.1")

xlabel(ax6,"t Axis")

ylabel(ax6,"y(t) Axis")

plot(ax78,x2,y2,"blue",xvec,yvec,"black")

title(ax78,"y(t) vs x(t) with h=0.1")

xlabel(ax78,"x(t) Axis")

ylabel(ax78,"y(t) Axis")

end

(a)

>> [t1,x1,y1] = EulSystem\_1(0.25,0,-1,-2,4);

>> [t2,x2,y2] = EulSystem\_1(0.1,0,-1,-2,4);

>> [tvec,xvec,yvec] = TrueSolution\_1();

>> CreateSubplots(t1,x1,y1,t2,x2,y2,tvec,xvec,yvec);



Figure 1: Plots of the solutions to the first system with step sizes 0.25, and 0.1

(b)

>> [t1,x1,y1] = EulSystem\_2(0.25,0,-1,-2,4);

>> [t2,x2,y2] = EulSystem\_2(0.1,0,-1,-2,4);

>> [tvec,xvec,yvec] = TrueSolution\_2();

>> CreateSubplots(t1,x1,y1,t2,x2,y2,tvec,xvec,yvec);



Figure 2: Plots of the solutions for the second system with step sizes 0.25, and 0.1

(c)

>> [t1,x1,y1] = EulSystem\_3(0.25,0,-1,-2,4);

>> [t2,x2,y2] = EulSystem\_3(0.1,0,-1,-2,4);

>> [tvec,xvec,yvec] = TrueSolution\_3();

>> CreateSubplots(t1,x1,y1,t2,x2,y2,tvec,xvec,yvec);



Figure 3: Plots of the solutions for the third system with step sizes 0.25, and 0.1

4)

(a)

Using a step size of 0.25 deviated further from the true solution as x approaches 4 as expected, but as x gets arbitrarily close to 4 the estimated solution deviates too far from the true solution to give a meaningful result. With a step size of 0.1 the estimated solution is close enough to the true solution to give a meaningful approximation. Also the solution for y(t) is exactly 2 \* x(t) so plotting y(t) vs x(t) actually produces a linear graph.

(b)

The approximated solutions with a step size of 0.25 and 0.1 were both almost identical to the true solution. For this system even a step size of only 0.25 is close enough to the true solution to give a meaningful approximation. Similar to the first system the solution for y(t) is exactly 2 \* x(t) so plotting y(t) vs x(t) also produces a linear graph. This result can most likely be attributed to the initial conditions for x(t) and y(t) since x(0) = -1 and y(0) = -2, and since the solutions for these two systems don’t involve cosines or sines it can almost be expected for y(t) to be exactly double x(t).

(c)

The solutions for the third system involved cosines and sines, and it seems that for both a step size of 0.25 and 0.1 the inflection points seem to be the same for the estimated solution and the true solution. The end points of the interval also are both approximated close to the true solution for both step sizes. What differs between the two step sizes are the approximated portions of the curve in between the two inflection points, in which a step size of 0.1 is close enough to the true solution for a meaningful approximation.

5)

>> a1 = [-0.5 0.6; -0.4 0.9];

>> a2 = [-0.34 0.12; -0.08 -0.06];

>> a3 = [0.65 1.95; -1.55 -2.15];

>> eig(a1)

ans =

-0.3000

0.7000

>> eig(a2)

ans =

-0.3000

-0.1000

>> eig(a3)

ans =

-0.7500 + 1.0308i

-0.7500 - 1.0308i