CS450 Homework 6

Xiaowen Lin

8.4 Based on Richardson extrapolation method,

$$F(h) = a_0 + a_1 h + \mathcal{O}(h^2)$$

Given that $F(0.2) = -0.8333$ and $F(0.1) = -0.9091$.

$$F(0) = a_0 = F(h) + \frac{F(h) - F(h/2)}{(1/2) - 1} = 2F(h/2) - F(h) = -0.9849$$

9.4

(a) Solution to the equation is $y(t) = \mathrm{e}^{-5t}$. For real λ, if $\lambda < 0$, all nonzero solutions decay exponentially, so every solution is not only stable, but asymptotically stable.

(b) Applying Euler method to$y^{(1)} = \lambda y$ using fixed step size h,

$$y_{k+1} = (1 + h\lambda)y_k$$

which means $y_k = (1 + h\lambda)^h y_0$ We must have

$$h \leq -\frac{2}{\lambda} = \frac{2}{5} = 0.4$$

But $h = 0.5$, so it is unstable.

(c) $y(0.5) = y(0) + h \times y^{(1)}(0) = 1 + 0.5 \times (-5) = -1.5$

(d) $|\frac{1}{1-h*\lambda}| = 0.286 < 1$ so backward Euler is stable with $h = 0.5$

(e)

$$y(0.5) = y(0) + h \times y^{(1)}(0.5) = y(0) - 2.5 \times y(0.5)$$

$$y(0.5) = \frac{y(0)}{3.5} = 0.286$$

9.7

(a)

$$\mathbf{U} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$u_1 = y(t), u_2 = y^{(1)}(t), u_3 = u_2^{(1)} = y^{(2)}(t) = y$$

$$\mathbf{U}' = \begin{bmatrix} u_1^{(1)} \\ u_2^{(1)} \end{bmatrix} = \begin{bmatrix} y^{(1)}(t) \\ y(t) \end{bmatrix}$$

(b)

$$\mathbf{U_0} = \begin{bmatrix} y(0) \\ y^{(1)}(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

(c)

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \times \begin{bmatrix} u_1^{(1)} \\ u_2^{(1)} \end{bmatrix}$$

Eigenvalues of $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ is -1 and 1. Because there is one eigenvalue 1 that is larger than 0, so the solutions are no stable.

(d)

$$\mathbf{U_{t=0.5}} = \mathbf{U_0} + h \times \mathbf{U_0^{(1)}} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} + 0.5 \times \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 2.5 \end{bmatrix}$$

(e)

$$\boldsymbol{I} + h\boldsymbol{J} = \begin{bmatrix} 0 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

$\rho(\boldsymbol{I} + h_k \boldsymbol{J}_f)$ is no bigger than 1, so error doesn't grow. And Euler's method is stable.

(f) $(\boldsymbol{I}–h \times \boldsymbol{J}(u^{(1)})) - 1 = \begin{bmatrix} 1 & -0.5 \\ -0.5 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 4/3 & 2/3 \\ 2/3 & 4/3 \end{bmatrix}$ Spectral radius is bigger than one, so it is unstable.

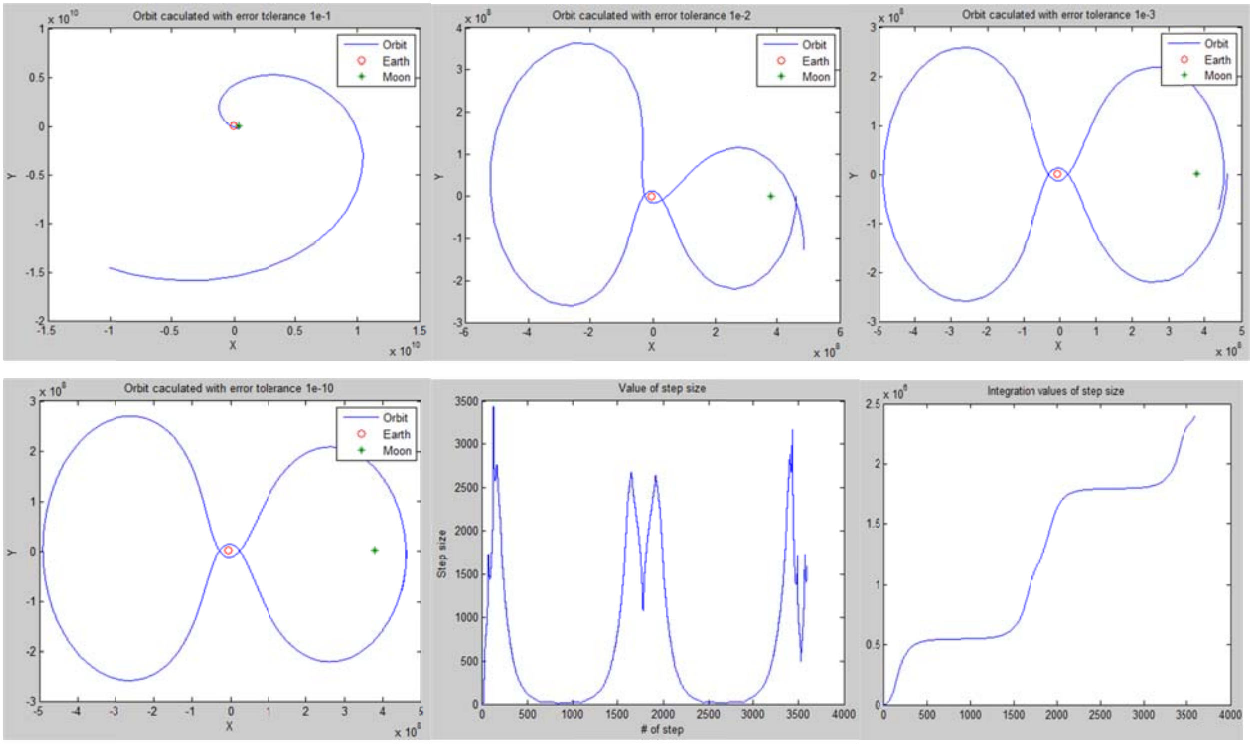
```
function cp08_04
syms x;
%Numerically evaluate integral, adaptive Simpson quadrature with "quad"
%funciton provided by matlab

disp('(a)');
f1=quad('x.^(3/2)',0,1);
disp(f1);

disp('(b)');
f2=quad('1./(1+10*(x.^2))',0,1);
disp(f2);

disp('(c)');
f3=quad('(exp(-9*x.^2)+exp(-1024*(x-1./4).^2))./pi.^(1./2)',0,1);
disp(f3);

disp('(d)');
f4=quad('(50./(pi*(2500*x.^2+1)))',0,10);
disp(f4);

disp('(e)');
f5=quad('1./abs(x).^0.5',-9,100);
disp(f5);

disp('(f)');
f6=quad('25*exp(-25*x)',0,10);
disp(f6);

disp('(g)');
f6=quad('log(x)',0,1);
disp(f6);

disp('So (a), (b), (c), (d), (e), (f) and (g) are all right solutions.');
```

```
>> cp08_04
(a)
    0.4000

(b)
    0.3999

(c)
    0.1979

(d)
    0.4994

(e)
   25.9955

(f)
    1.0000

(g)
   -1.0000

So (a), (b), (c), (d), (e), (f) and (g) are all right solutions.
>>
```

```
function cp09_08

G=6.67259e-11;
M=5.974e+24;
m=7.348e+22;
u_s=M/(m+M);
u=m/(m+M);
D=3.844e+8;
d=4.669e+6;
omiga=2.661e-6;

x0=4.613e+8;
y0=0;
dx0=0;
dy0=-1074;

tspan=[0,2400000];

figure(1);

options = odeset('RelTol',1e-1);
[t,x]=ode45(@odes,tspan,[x0 y0 dx0 dy0], options);
plot(x(:,1),x(:,3),(-d),0,'ro',(D-d),0,'*');
legend('Orbit','Earth','Moon');
title('Orbit caculated with error tolerance 1e-1');
xlabel('X');ylabel('Y');

figure(2);
options = odeset('RelTol',1e-2);
[t,x]=ode45(@odes,tspan,[x0 y0 dx0 dy0], options);
plot(x(:,1),x(:,3),(-d),0,'ro',(D-d),0,'*');
legend('Orbit','Earth','Moon');
title('Orbit caculated with error tolerance 1e-2');
xlabel('X');ylabel('Y');

figure(3);
options = odeset('RelTol',1e-3);
[t,x]=ode45(@odes,tspan,[x0 y0 dx0 dy0], options);
plot(x(:,1),x(:,3),(-d),0,'ro',(D-d),0,'*');
legend('Orbit','Earth','Moon');
title('Orbit caculated with error tolerance 1e-3');
xlabel('X');ylabel('Y');

figure(4);
options = odeset('RelTol',1e-10);
[t,x]=ode45(@odes,tspan,[x0 y0 dx0 dy0], options);
plot(x(:,1),x(:,3),(-d),0,'ro',(D-d),0,'*');
legend('Orbit','Earth','Moon');
xlabel('X');ylabel('Y');
title('Orbit caculated with error tolerance 1e-3');
```

```
disp('When error tolerance goes from 1e-1 to 1e-10, the orbit becomes more and more
actually closed.');

figure(5);
plot(t);
dt=t(2:end)-t(1:(end-1));
title('Integration values of step size');

figure(6);
plot(dt);
title('Value of step size');
xlabel('# of step');ylabel('Step size');

disp('Based on these figures titled "Integration values of step size" and "Value of step
size", we know that step size changes during the process. ');


[mx,nx]=size(x);
V=zeros(mx,1);
for i=1:mx
    V(i,1)=norm([x(i,1)+d, x(i,3)],2);
end

min(V);

disp('The minimum distance from spacecraft to earth is');
disp(min(V)-6.378e6);

function fcode=odes(t,x)
    u1=x(2);
    u2=-G*(M*(x(1)+u*D)/(((x(1)+d)^2+x(3)^2)^(1/2))^3+m*(x(1)-u_s*D)/(((D-d-x(1))^2+x(3)
^2)^(1/2))^3)+omiga^2*x(1)+2*omiga*x(4);
    u3=x(4);
    u4=-G*(M*x(3)/(((x(1)+d)^2+x(3)^2)^(1/2))^3+m*x(3)/(((D-d-x(1))^2+x(3)^2)^(1/2))^3)
+omiga^2*x(3)-2*omiga*x(2);
    fcode=[u1;u2;u3;u4];
end


end
```

```
>> cp09_08
When error tolerance goes from 1e-1 to 1e-10, the orbit becomes more and more actually↙
closed.
Based on these figures titled "Integration values of step size" and "Value of step size",↙
we know that step size changes during the process.
The minimum distance from spacecraft to earth is
   6.8558e+06

>>
```


Orbit caculated with error tolerance 1e-1
Orbit
Earth
Moon
X
Y
Orbit caculated with error tolerance 1e-2
Orbit
Earth
Moon
X
Y
Orbit caculated with error tolerance 1e-3
Orbit
Earth
Moon
X
Y
Orbit caculated with error tolerance 1e-10
Orbit
Earth
Moon
X
Y
Value of step size
Step size
# of step
Integration values of step size