Homework #10

Math 131-05D-spring 2017

5/5/17 at 12:00 PM

By Yeash Patel

1. For each of the following one-step time methods

   (a) Euler's method

   (b) Runge-Kutta method of order 2

   (c) Runge-Kutta method of order 4

   write a function that solves the IVP

$$y'(t) = f(t,y) \text{ for } a \leqslant t \leqslant b, \text{ with } y(a) = \alpha.$$

   In each case, the function header should look something like

   `function w = method_name(f,a,b,alpha,N)`

   where N is the number of intervals used, so that $h = \frac{b-a}{N}$

One can see the functions in appendix A. To show how to use the these functions we will be solving for y`=y which we know that y=exp(t) which shall solve from interval 0 to 2 for 100 iterations. The functions should return a numerical value close to 7.38905609893

MATLAB Input and Output

```
>> f= @(t,y)y

f =

  function_handle with value:

    @(t,y)y

>> Euler(f,0,2,1,100)

ans =

    7.2446

>> RK2(f,0,2,1,100)

ans =

    7.3881
```

```
>> RK4(f,0,2,1,100)

ans =

    7.3891
```

2. Write a function that solves the IVP using a 2-step Adams-Bashforth method that computes w_1 using a second order Runge–Kutta method. The header should have the same format as the headers in problem 1.

One can also see the functions in appendix A. To show how to use the functions we will be solving for y`=y which we know that y=exp(t) which shall solve from interval 0 to 2 for 100 iterations. The functions should return a numerical value close to 7.38905609893

MATLAB Input and Output

```
>> f=@(t,y) y

f =

  function_handle with value:

    @(t,y)y

>> AB2(f,0,2,1,100)

ans =

    7.3866
```

3. Consider the IVP

$$y'(t) = \frac{\sin(2t) - 2ty}{t^2} \text{ for } 1 \leqslant t \leqslant 2, \text{ with } y(1) = 2.$$

(a) Use all 4 methods that you have developed in problems 1 and 2 to solve the IVP given with $N = 10; 10^2; 10^3; 10^4$.

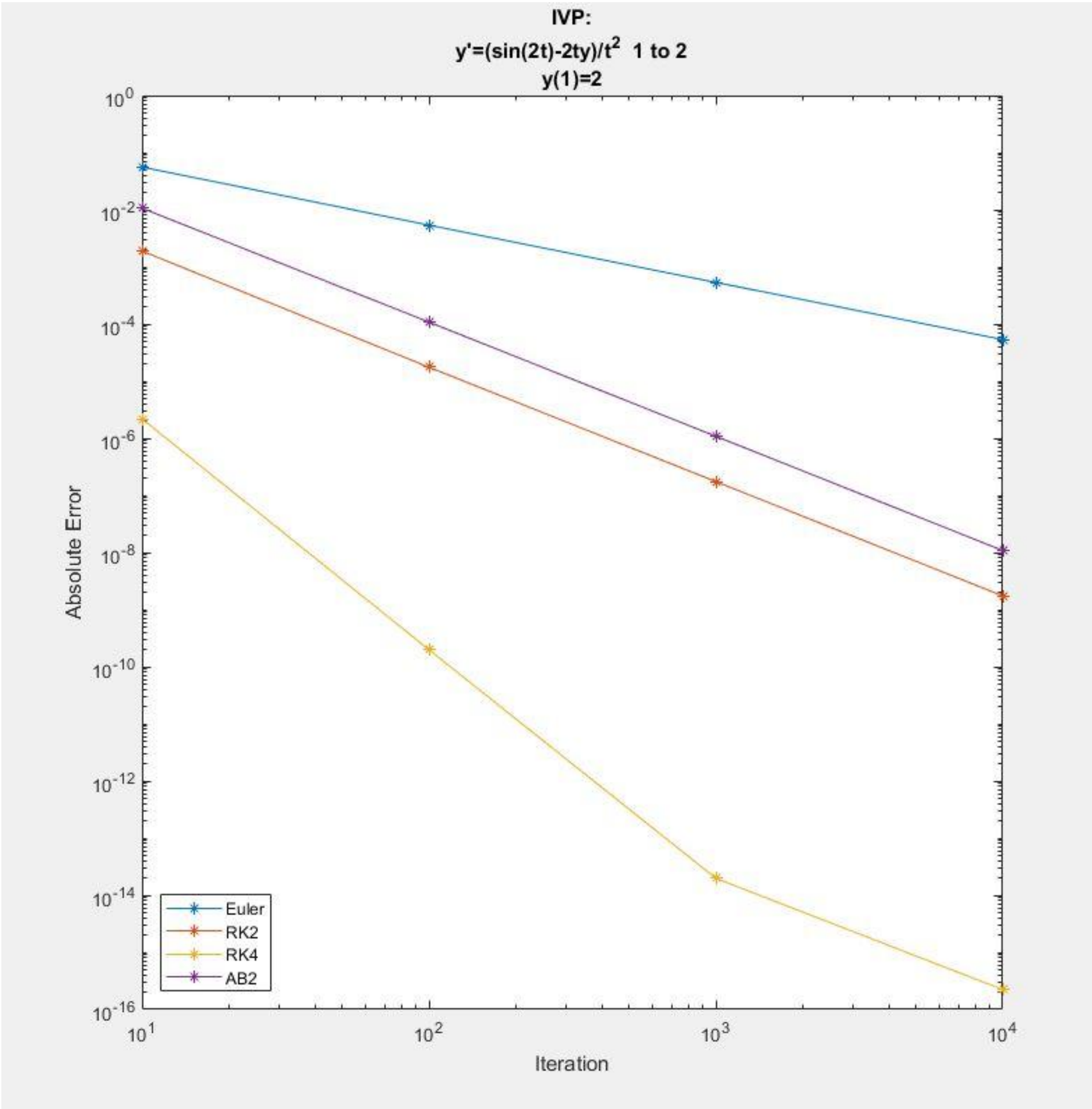
(b) Make a `loglog` plot of absolute error at $t = 2$ versus the number of intervals for all four methods on the same plot. Make sure to create a legend, showing which plot corresponds to which method. Comment on your results: which method works best, which method works the worst? Why?
Hint: To compute the absolute error, you need to find the exact/true/actual solution of this IVP. One can show that it is $y(t) = \frac{4+\cos(2)-\cos(2t)}{2t^2}$ (using the knowledge you learned from Math 24).

I wrote script to simplify the commands needed to evaluate this problem. As seen in the graph the Runge-Kutta 4 method preformed the best out the four however it should be noted that the method requires a lot of function evaluations. The worst performer according the graph was by the Euler method. The Euler method did not perform as well, but has a lower function evaluation requirement and therefore requires less resources.

MALAB Input

>> Pr3

MATLAB Output



IVP:
y'=(sin(2t)-2ty)/t^2 1 to 2
y(1)=2
Absolute Error
Iteration
Euler
RK2
RK4
AB2

APENDIX A

Euler.m

```
function [w] = Euler( f,a,b,alpha,N )
h = (b-a)/N;
t=a;
w=alpha;
for i=(1:N)
    w=w+h*f(t,w);
    t=a+i*h;

end

end
```

RK2.m

```
function [ w ] = RK2(f,a,b,alpha,N)
h = (b-a)/N;
t=a;
w=alpha;
for i=(1:N)
    k1=f(t,w);
    t=a+i*h;
    k2=f(t,w+h*k1);
    w=w+h/2*(k1+k2);
end

end
```

RK4.m

```
function [ w ] = RK4(f,a,b,alpha,N)
h = (b-a)/N;
t=a;
w=alpha;
for i=(1:N);
    k1=h*f(t,w);
    k2=h*f(t+h/2,w+k1/2);
    k3=h*f(t+h/2,w+k2/2);
    k4=h*f(t+h,w+k3);

    w= w+1/6*(k1+2*k2+2*k3+k4);
    t=a+i*h;

end

end
```

AB2.m

```
function [ Z ] = AB2( f,a,b,alpha,N )
h = (b-a)/N;
t(1)=a;
w(1)=alpha;
k1=f(t(1),w(1));
t(2)=a+h;
k2=f(t(2),w(1)+h*k1);
w(2)=w(1)+h./2*(k1+k2);
for i=(2:N)
    w(i+1)=w(i)+h/2*(3*f(t(i),w(i))-f(t(i-1),(w(i-1))));
    t(i+1)=a+i*h;
    Z=w(i+1);
end

end
```

Pr3.m

```
clc
clear all
close all

f = @(t,y) (sin(2*t)-2*t*y)./t.^2;
a=1;
b=2;
alpha=2;
t=2;
y=(4+cos(2)-cos(2*t))./(2*t.^2);
for i=(1:4)
    N=10^i;
    xx(i)=N;

    Euz(i)= Euler(f,a,b,alpha,N);
    errEuz(i)=abs(Euz(i)-y);
    RK2z(i)= RK2(f,a,b,alpha,N);
    errRK2z(i)=abs(RK2z(i)-y);
    RK4z(i) = RK4(f,a,b,alpha,N);
    errRK4z(i)=abs(RK4z(i)-y);
    AB2z(i) = AB2(f,a,b,alpha,N);
    errAB2z(i)=abs(AB2z(i)-y);
end

loglog(xx,errEuz,'-*',xx,errRK2z,'-*',xx,errRK4z,'-*',xx,errAB2z,'-*')
legend('Euler','RK2','RK4','AB2','Location', 'Southwest')
title({'IVP:','y''=(sin(2t)-2ty)/t^2  1 to 2','y(1)=2'})
xlabel('Iteration')
ylabel('Absolute Error')
```