Yingjie Lian
Assignment 05
CS-3200
04.10.2018

Assignment 05 Report

In my matlab files, those are called programs are my source code. And the readme notes are in the matlab file too. I wrote those readme files as comments.

Question 1:

Since it is hard to represent the equation in word, so I solved that on hand. Then I scanned that, the BLUE circle is the solution:

Solve $\frac{dT_c}{dt} = -r(T_c - T_s)$

First, let $s = (T_c - T_s)$

Then $\frac{ds}{dt} = \frac{d}{dt}(T_c - T_s)$

since T_s is a constant

So we have $\frac{ds}{dt} = \frac{d}{dt} T_c$

$= -r(T_c - T_s)$

$\frac{ds}{dt} = -rs$

$\int ds = \int dt \cdot (-rs)$

$\therefore s = const \cdot e^{-rt}$ $\quad \because const = T_c - T_s$

$T_c - T_s = const \cdot e^{-rt}$

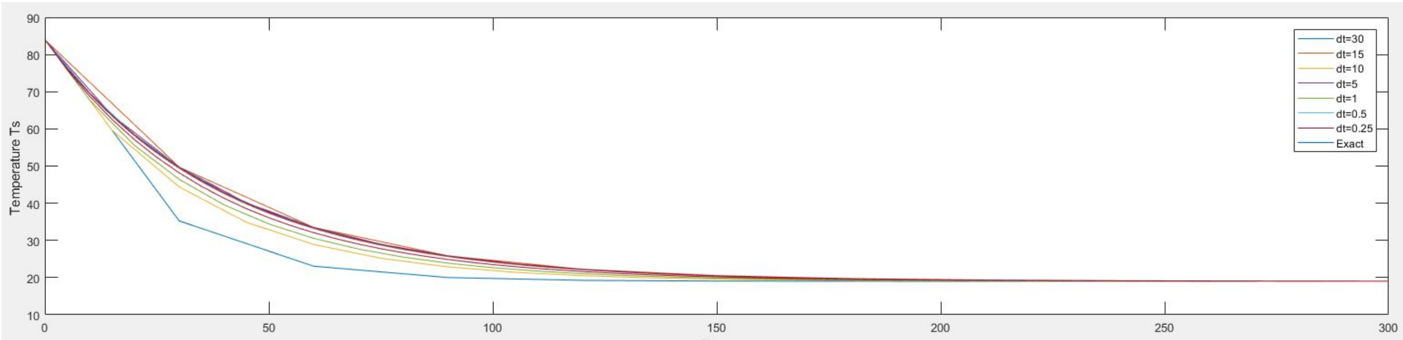
$T_{c\,exact} = (T_c - T_s) e^{-rt} + T_s$

$\because T_s = 19^{\circ}C,\ T_c = 84^{\circ}C,\ r = 0.025/second,\ t = 5min = 300s$

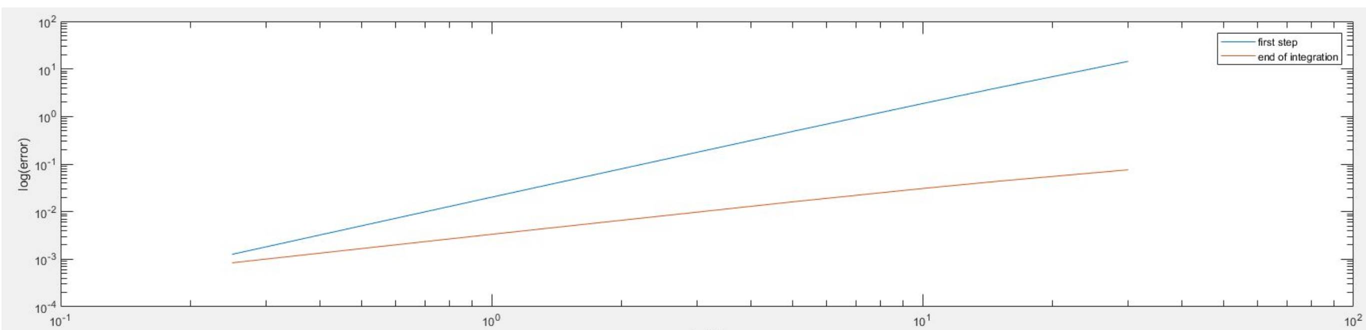
$T_{exact} = (84 - 19)e^{-0.025 \cdot 300} + 19$

≈ 19.0359505

Question 2:



The graph above shows the results for all algorithms using several different values for the step size h ($h = 30s$, $15s$, $10s$, $5s$, $1s$, $0.5s$, $0.25s$). Notice, h also means dt here.



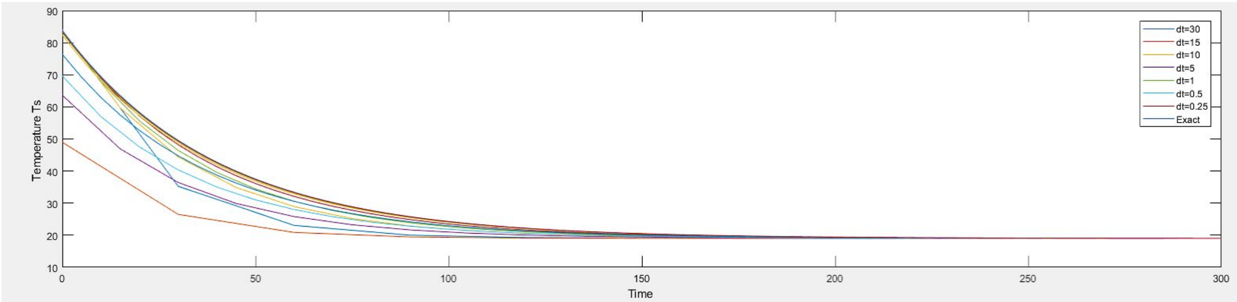
The graph above shows the error results for the first step and at the end of the integration. As we can see, when h increases, the errors increase too.

Since error = c * h^p, and this graph has used log to solve the problem. So in this graph, p is the slope of each line.
As a result, first step error = O(h^1.943640682288348), p = Ratefirst in matlab = 1.943640682288348
end of the integration error = O(h^0.930869622879538), p = Ratelast in matlab = 0.930869622879538

Question 3:

Please run Question 3 file in my matlab.

Question 4:



The graph above shows the results for all algorithms using several different values for the step size $\hbar$ ($\hbar = 30s, 15s, 10s, 5s, 1s, 0.5s, 0.25s$). Notice, h also means dt here.

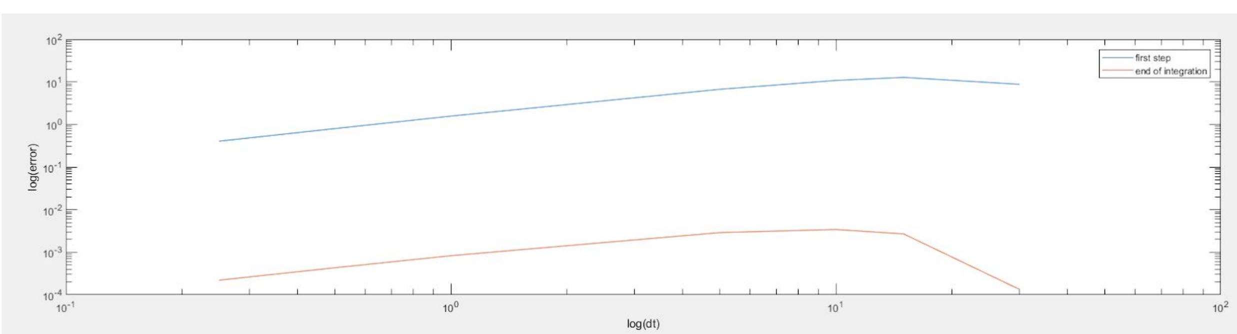
Question 5:

By using the ODE 23 method we got the error after the first step is firsterr =

8.7598
12.7350
10.7910
6.6836
1.5647
0.7973
0.4025

and the error at the end of the integration is lasterr =

0.0001
0.0027
0.0034
0.0029
0.0008
0.0004
0.0002



The graph above shows the error results for the first step and at the end of the integration. It becomes much more smaller.

Then if we compare to Newton's law method results: the error after the first step is firsterr =

14.4538
4.0488
1.8721
0.4873
0.0201
0.0051
0.0013

and the error at the end of the integration is lasterr =

0.0759
0.0437
0.0307
0.0161
0.0033
0.0017
0.0008

We will see that the ODE 23 method is very accurate

Question 6:

Yes. After changing r = 0.6. The first rate is 1.4732136.3 and last rate is 15.3734926.3 now. Which means the error estimator did blow up.