Nicholas Crane

PHY 307

Professor Freeman

01 October 2019

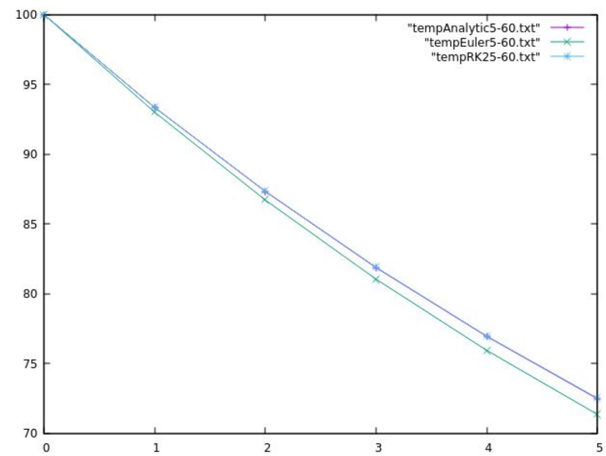
Project 3: Numerically Solving ODEs

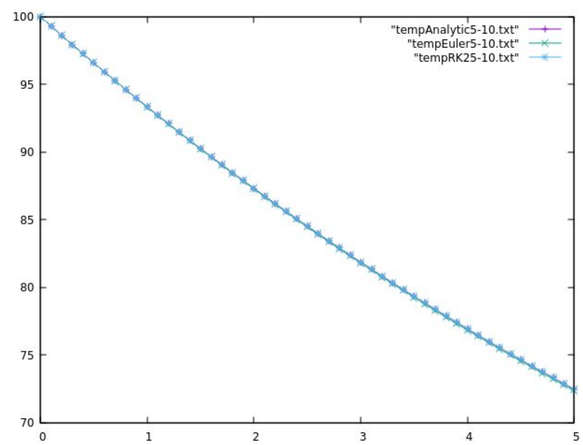
In this project, we utilized and analyzed two different methods for solving ordinary differential equations and analyzed their error with the analytic result. We are investigating these approximations because many ordinary differential equations (ODEs) do not have analytic solutions, and so must be approximated by other means. The two methods we investigated were Euler's Method and the Runge-Kutta Method.

Before analyzing the error and behavior of the two methods it is important to understand how they work. The ODE we are analyzing gives us the slope of a line at a given point, say (x,y). Using Euler's Method with a given initial starting point (x_0,y_0), we can calculate the slope at that point and move a small distance along that line to a new point (x_1,y_1), where f'(y) gives the slope of the line and dx is a small change in distance along the x axis, $y_1 = y_0 + f'(y_0)*dx$.We then do this many times to approximate the solution to the ODE. The Runge-Kutta (RK2, the 2 comes from the fact that this is the second-order Runge-Kutta Method) Method works in a similar fashion, but instead of using the slope of the 'left' point to move to the right point, we use the slope of the point in between the left and right points. But how do we know the slope at the middle point? We again return to our ODE to solve this problem. Using the slope at the left point, we can approximate the middle point by using Euler's Method while moving dx/2 along the x-axis. So, $y_1 = y_0 + f'(y_0 + f'(y_0)*dx/2)*dx$. This is again repeated many times to approximate a solution.

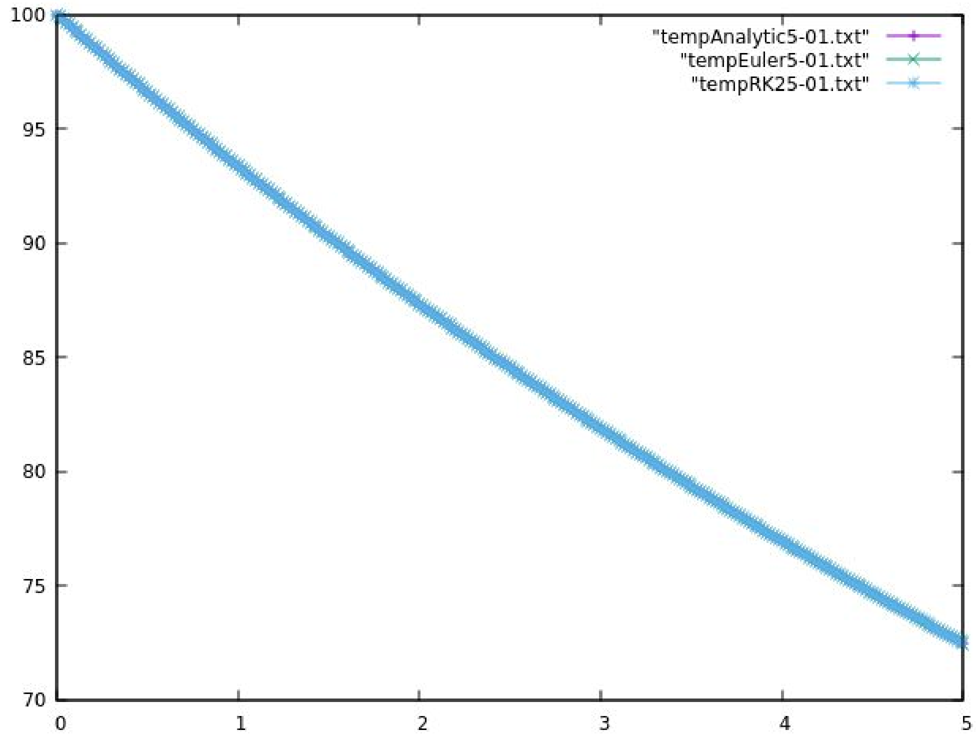
With that understanding of the two methods, we can now analyze how they perform. To begin with, I plotted the analytic solution to the ODE we are analyzing, along with solutions from the two methods we are using with step sizes of dt = 1 min, 10s, and 1s:

Temp vs Time (Analytic, RK2, Euler, dt = 60s) Temp vs Time (Analytic, RK2, Euler, dt = 10s)



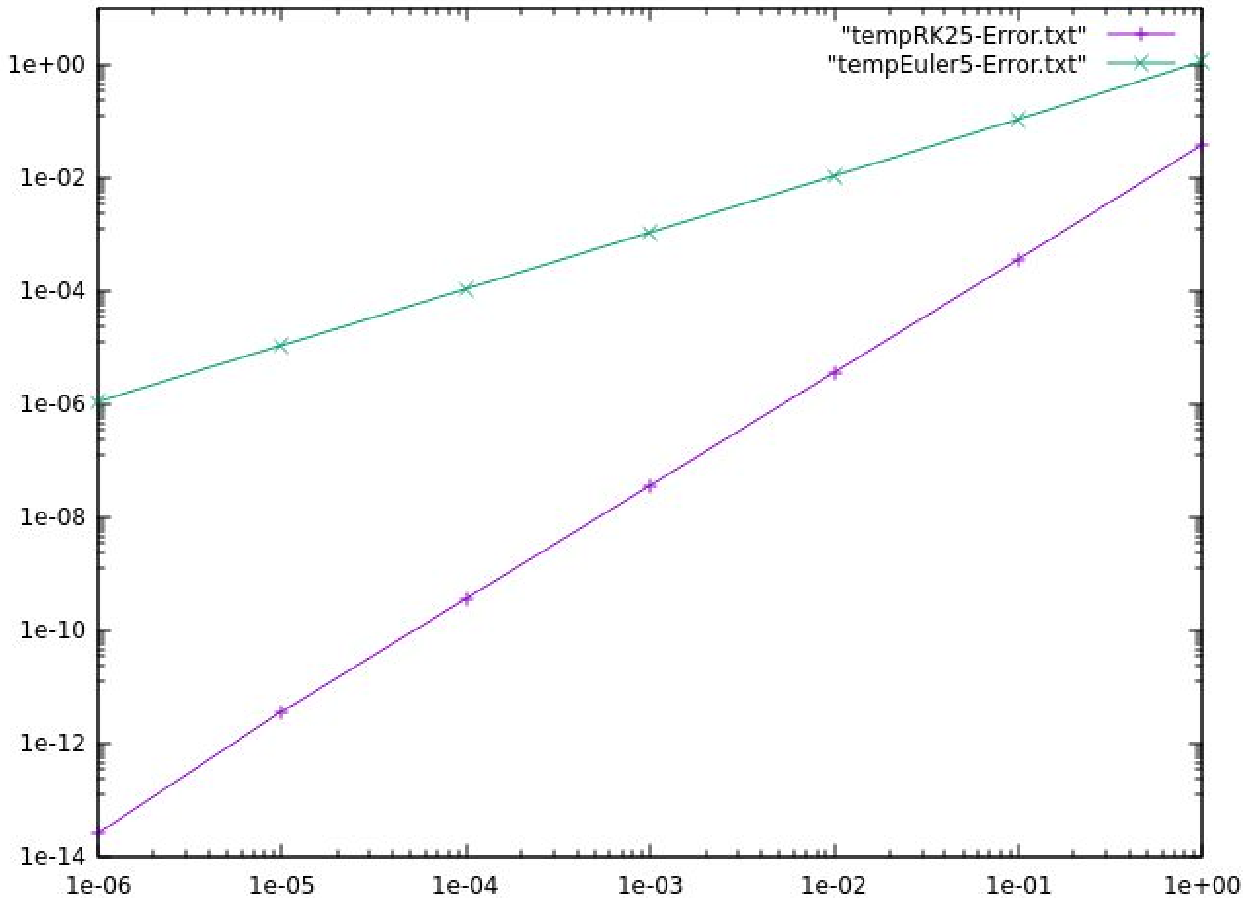


Temp vs Time (Analytic, RK2, Euler, dt = 1s)



While methods demonstrate the same behavior as the analytic solution, from the first graph we can see that Euler's Method is far less accurate than RK2. Just how much more accurate RK2 is over Euler's Method is what I analyzed next. Calculating the error between the final temperature after 5 minutes for each method with the analytic solution creates the following graph of ln(error) vs ln (stepSize), where dt ranges from 1e0 to 1e-6:

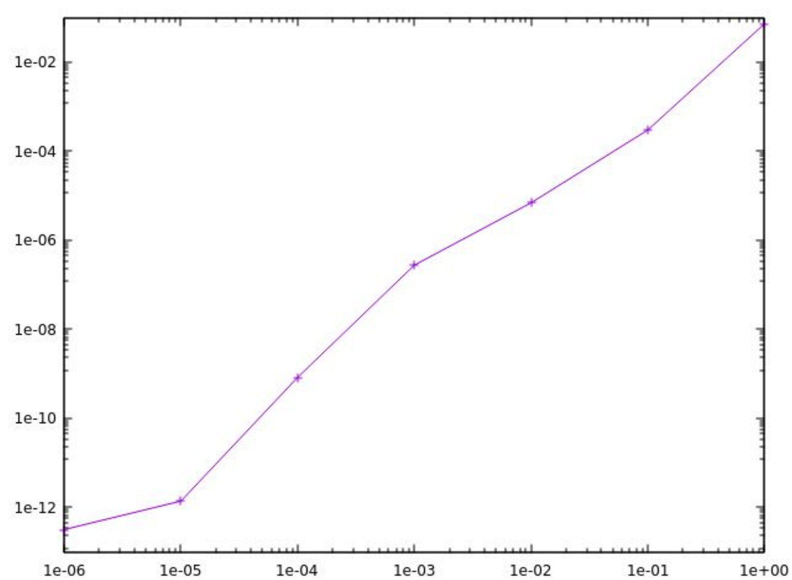
ln(error) vs ln(stepSize) for t = 5 min



Where the purple line is the error in RK2 and the green line is the error in Euler's Method. Euler's Method has a slope of 1, indicating the error is proportional to the step size, while RK2 has a slope of two indicating the error is proportional to the step size squared. This means RK2 offers far more accuracy over Euler's method for a given step size. This does come at a cost however; RK2 took about twice as long to complete, taking 0.153s over Euler's 0.088s. This is because of the extra computation required to find the middle point in RK2. The gain in accuracy is considerable enough to negate this point depending on the purpose of the program, but if only rough approximations are required however, Euler's Method is a good and fast option.

The final case I investigated was trying to approximate an exact solution to the ODE; in this case, for what value of t does T(t) = 70 degrees. Using RK2 and halting the program as soon as the temperature went below 70 degrees, we have a good approximation for tDone, the time when the temperature is equal to 70 degrees. This approximation can be improved however. This tDone is always after tTrue, the actual time when the temperature is 70 degrees. Using the slope at the time we have found, we can draw a line through the point we have found that intersects the line T = 70. Finding the time when this line intersects T = 70 gives a far better approximation for tDone, and a graph of ln(error) vs ln(stepSize) shows that:

ln(error) vs ln(stepSize) for when T = 70



Similarly to the RK2 method this line has a slope of about two so the error is proportional to the square of step size. This graph is not perfectly linear however. With different step sizes, the time when we halt RK2 finishes either closer to or further away from T = 70, so the method we used to calculate tDone has more or less error depending on that distance. It is like approximating further back if tDone is further away from T = 70.

In summation, Euler’s Method and the RK2 Method are both useful tools for approximating solutions to ODEs. While RK2 may be slower than Euler’s Method, it is also far more accurate. Depending on the goals of a program, the choice between these methods is a case of speed vs accuracy.