Math 2280 - Computer Assignment 2

due Wednesday, September 27

Start Early!!

The goal of this assignment is to investigate the different numerical methods for solving IVP which we talked about in class, namely Euler's method, Improved Euler's method, and Runge-Kutta. Use the MATLAB code provided on the course website to get started. You will need to modify the file **rhs_fun.m** and **driver.m** accordingly. In addition to the write-up answering the questions below, also turn in plots of your numerical simulation and the MATLAB code for problem 3.

1. Consider the following initial value problem

$$\frac{dx}{dt} = 2tx^2$$

with initial condition $x(0) = 1$.

   (a) First, find the exact solution by hand.What is the behavior as $t \to 1$?

   (b) Use a step size of $h = 0.1$ to find the approximate solutions with all three methods above. Compute only up to $t = 0.95$. Find the errors at the end.

   (c) Now use a step size of $h = 0.01$ and compute the error again. How much is the error reduced by decreasing the step size? Are these numbers consistent with the order of the method?

2. *Stability* The position of a mass attached to a spring can be described by the equation

$$\frac{dx}{dt} = -kx$$

x is the distance from a rest state, k is the spring constant and taken to be $k = 22$ here. Suppose the initial position is $x(0) = 0.9$.

   (a) What is the exact solution of the system?

   (b) Use a step size of $h = 0.1$ to approximate the solution up to $t = 1$. Do all three methods at least capture what the general behavior of the system (i.e. decaying exponential)? Which method is best in this particular case? Can you think about why Euler's method is behaving this way in terms of mass and spring?

   (c) Now reduce h to 0.05 then to 0.01. Describe what you see now.

3. *Implicit Euler scheme* The behavior of the Euler method in the previous problem has to do with the stability of the numerical method. When k in the equation above is large, the problem is referred to as a stiff one (large spring constant). For stiff problem h has to be taken to be small when using Euler's method to get the correct numerical approximation as you saw in the previous problem.

   Now consider a slight modification to Euler's method: instead of using left end point in the integral approximation, use the right end point so that

$$x_{n+1} = x_n + hf(t_{n+1}, x_{n+1})$$

Use this method to solve the same problem as before

$$\frac{dx}{dt} = -kx$$

with $k = 22$ and $x(0) = 0.9$ still. For this particular problem, we can then write

$$x_{n+1} = x_n - hkx_{n+1}$$

Solving for x_{n+1},

$$x_{n+1} = \frac{x_n}{1 + hk}$$

(a) Implement the Implicit Euler scheme for this particular problem in MATLAB.

(b) Use $h = 0.1$ to approximate the solution up to $t = 1$. Are you still having the same problem as when you were using the regular Euler's method?

This scheme is still first order just like Euler's method but it does not have stability issue. Implicit scheme is generally stable in the sense that there's no restricition on the step size to just capture the behavior of the system. However, it uses the value of x_{n+1} to approximate the integral. For a general problem, this can be a challenge as one would have to do a function solve (note how we solve for x_{n+1} above) at each time-step.

4. *Round-off Error Problem* A similar problem is discussed in Example 4 (section 2.6) in your text. Consider the following problem

$$\frac{dx}{dt} = 4x - 5e^{-t}$$

with $x(0) = 1$.

(a) Find the general solution to the system and use the initial condition to find the particular solution to this system. In this particular case, what is the limit of $x(t)$ as $t \to \infty$? What about for a different initial condition?

(b) Now, use Runge Kutta to approximate the solution to the system with $h = 0.1$ and $h = 0.05$ and compute up to $t = 3.5$. What is the error at the end? Try to explain why you are obtain such huge errors in terms of the general solution to the system.

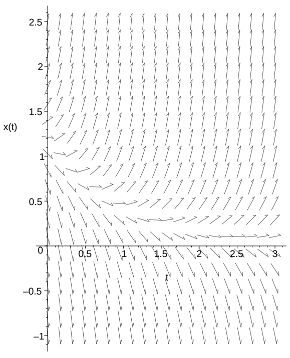

Figure 1: Slope fields for $\frac{dx}{dt} = 4x - 5e^{-t}$