Project 1

Ove Haugvaldstad, Eirik Gallefoss

September 4, 2019

Abstract

Introduction

Computing has had and still have an undeniable influence on science. has allowed scientist to explore everything from the tiniest scale of an atom,to tropical cyclones and galaxies. Therefore understanding the inner workings behind a computer program is critical in order to avoid unwanted errors. Errors which in the worst case can have catastrophic consequences [1].

Our aim is to investigate some of the common errors one might face if one doesn't think when developing code. To begin with we will look at a how to solve a second order differential equation, specifically the general one dimmensional Poisson's equation ().

$$f(x) = -\frac{\partial^2 u}{\partial x^2} \tag{1}$$

Method

Numerical differentiation

Computers operate in discrete steps, which means that variables are stored as discrete variables. For a discrete variable over a particular range there is a minimum step length between each value that variable can take in that range. Compared to a continuous variable which can take on any value in that particular range. The step length h can either be set manually or it can be determined based on the start and end point our particular range, $h = \frac{x_n - x_0}{n}$. Where n is the number of points we choose to have in our particular range. The smaller the step length in of the discrete variable the better it will approximate the continuous variable.

Then if we look at the definition of the derivative it is clear to see how this would work for the discrete case.

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h} \tag{2}$$

Since we obviously can't on a computer let $h = 0$ the derivate will always be an approximation. A natural question to ask when making approximations is what is our error. The error can be derived if we do a taylor series expansion of the $f(x+h)$ term in around x.

$$f(x+h) = f(x) + h'f(x) + \frac{h^2 f''(x)}{2} + \frac{h^3 f'''(x)}{6} + \dots \tag{3}$$

If we next insert this taylor expansion into eq.(3) we get:

$$f'(x) = f'(x) + \frac{hf''(x)}{2} + \frac{h^2 f'''(x)}{6} + \dots \tag{4}$$

Then we see that our derivate includes $f'(x)$ and some terms proportional to $h, h^2, h^3 \dots$ and since h is small the h terms would dominate.

Inorder to solve eq(1) numerically, we need to discretize our problem. We also assume Dirichlet boundary conditions $u(0) = u(1) = 0$, that $x = \in (0, 1)$ and that our equation is scaled to avoid dealing with physical units.

The first step in discretizing any problem is let our input variable x be a discrete variable $x_i \in [x_0, x_1, x_2, ..., x_n]$. The distance between each x_i variable is controlled by the step size, $h = \frac{x_n - x_0}{n}$, this gives an expression for $x_i = x_0 + i \cdot h$ where $i = 1, 2, 3, ..., n$.

A widely used method to calculate the derivate numerically is what is called the 3 point method, (equation (5)), where $f_{i\pm1}$ is introduced as a shorthand for $f(x_i \pm h)$.

$$f'_i = \frac{f_{i+1} - f_{i-1}}{2h} \tag{5}$$

The three point formula is derived by evaluating the derivate at both sides of a chosen point x_i and then taking the average afterwards. The three point formula has an error of order of magnitude $O(h^2)$ compared to a two point method which has $O(h)$, while requiring the same number of floating point operations (flops). The reasoning behind the three point method is that you evaluate $f(x_i)$ at x_{i-1} and x_{i+1}

n	run time (s)
10	1.21e-04
100	3.63e-04
1000	4.33e-03
10000	3.76e-02
100000	4.06e-01
1000000	3.13e+00
10000000	3.09e+01

n	run time (s)
10	1.03e-04
100	2.67e-04
1000	2.48e-03
10000	1.93e-02
100000	2.22e-01
1000000	2.15e+00
10000000	1.94e+01

log_{10}(h)	max(relative error)
-1.04	0.03
-2.00	-0.15
-3.00	-0.02
-4.00	-0.00
-5.00	-0.00
-6.00	-0.00
-7.00	-0.00

References

1. Arnold, D. N. *The sinking of the Sleipner A offshore platform* http://www-users.math.umn.edu/~arnold//disasters/sleipner.html. (accessed: 01.09.2019).