Homework 1: Numerical Integration, differentiation, and bisection

A.L. Phillips II

Department of Physics,Rensselaer Polytechnic Institute

`philla3@rpi.edu`

12 February 2013

Chapter 1

Integration, differentiation, and bisection

1.1 Sources of Error

1. Roundoff vs Truncation

Two sources of error in numerical methods are Round-off and Truncation errors. Round-off errors are the result of systems having a finite quantity of significant figures to represent numbers. For example, if a computer is capable of can store three significant figures, then it could approximate $\frac{1}{3}$ as 0.333. Because the computer is not representing $\frac{1}{3}$ *exactly* as $0.\overline{3}$, a round-off error (ξ_{max_T}) occurs.

In this case, $\xi_{max_T} = \dfrac{1}{3} - 0.333 = 0.000\overline{3}$

Truncation error is the result of truncating (shortening) a mathematical procedure. The truncating of a mathematical procedure can occur whenever we are required to approach zero (integrate) or use an infinite number of terms (series expansions). For example, the Maclaurin series of $cos(x) = 1 - \frac{1}{2}x^2 + \frac{1}{24}x^4 - \frac{1}{720}x^6 + \dots$ .Since we must chose a finite number of terms, a truncation error will ultimately result. If we choose two terms, the truncation error would then equal the sum of all the excluded terms. Namely, $\frac{1}{24}x^4 - \frac{1}{720}x^6 + \dots$.

1.2 Numerical Integration Techniques

1. Trapezoidal Method:

The trapezoid rule calculates the area under a curve much like a Rie-

mann sum. Here, rather than using rectangles, the trapezoidal method employs trapezoids. The graphic below depicts the use of a single trapezoid to approximate the area under the blue curve. The more intervals that $(b - a)$ is divided into, the closer the "secant-shaped" tops of each trapezoid will become to the tangent on the curve above it.


.
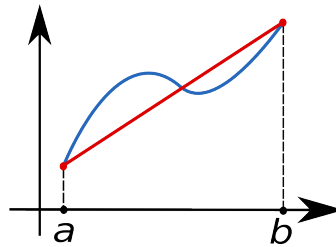
The Trapezoid Rule for approximating a definite integral is as follows:

$$\int_a^b f(x)\,dx \approx \frac{\Delta x}{2}\Big[f(x_0) + 2f(x_1) + \ldots + 2f(x_{n-1}) + f(x_n)\Big]$$

where $\qquad \Delta x = \dfrac{(b - a)}{n} \qquad$ and $\qquad x_i = a + i(\dfrac{b - a}{n})$

Trapezoidal Error:

When numerically computing $\displaystyle\int_1^5 x^2\,dx$, the number of intervals that are used will vary the accuracy of the calculated area. The below graphic shows the relative error ($\epsilon = \frac{|experimental - known|}{known}$) as a function of the number of intervals.

.
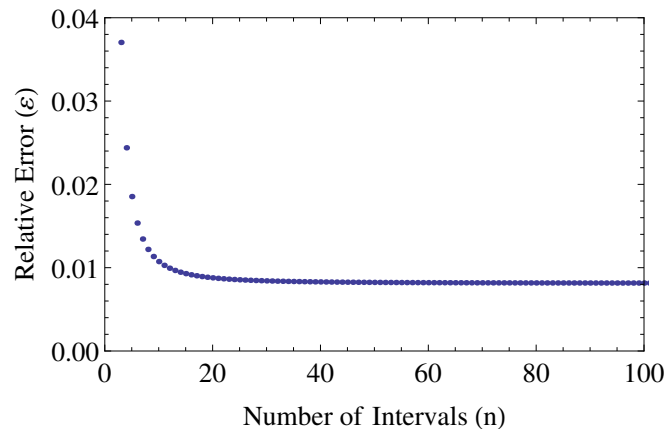
The maximum amount of truncation error (ξ_{max_T}) for the trapezoid method may be determined as follows:

$$\xi_{max_T} = \frac{(b-a)^3 \left| f^{(2)}_{(max)} \right|}{12n^2}$$

.

Here, b and a are the upper and lower limits of integration, n is the number of intervals that $(b - a)$ is divided into and $f^{(2)}_{(max)}$ is the second derivative of the function. The (max) subscript indicates that the second derivative should be evaluated at the point from $[a, b]$ which maximizes the value of $f^{(2)}$.

Similarly, the minimum number of intervals needed to approximate an integral to within a given accuracy (ξ_{max}) may be determined by solving the above equation for n:

$$n \geq \sqrt{\frac{(b-a)^3 \left| f^{(2)}_{(max)} \right|}{12(\xi_{max})}}$$

.

Note: n must be a whole number; therefore if $n \geq 10.5$ then choose $n = 11$

2. Simpson's Rule:

Simpson's rule calculates the area under a curve much like a Riemann sum. Here, rather than using rectangles, Simpson's rule employs parabolas. The graphic below depicts the use of a red parabola between two intervals to approximate the area under the blue curve. The more intervals that are added between a and b, the closer the "parabolic-shaped" tops of each interval will become to the tangent on the curve above it.


.
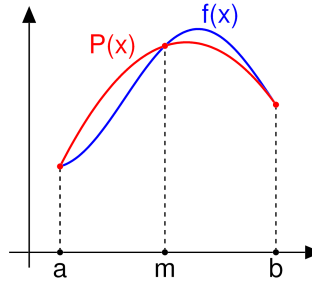
The formula for calculating Simpson's Rule is as follows:

$$\int_a^b f(x)\,dx \approx \frac{\Delta x}{3}\Big[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \ldots + 2f(x_{n-2}) +$$

$$4f(x_{n-1}) + f(x_n)\Big], \quad \text{where } \Delta x = \frac{(b-a)}{n} \quad \text{and} \quad x_i = a + i\left(\frac{b-a}{n}\right)$$

Note: Simpson's Rule requires an even number of intervals (n).

Simpson Rule Error:

This The accuracy of the numerical computation of $\int_1^5 x^2\,dx$ will vary with the number of intervals used. The below graphic shows the relative error (ϵ) of both Simpson's Rule (red) and the Trapezoid Method (blue). Notice how quickly Simpson's rule converges versus the Trapezoid Rule. This is largely due to Simpson's use of a parabolic curve to approximate the area under a parabola (x^2). Here Simpson's approximation may be exact. In the case of this quadratic function, Simpson bests Trapezoid, however, were the function linear, then the opposite would be true.

.
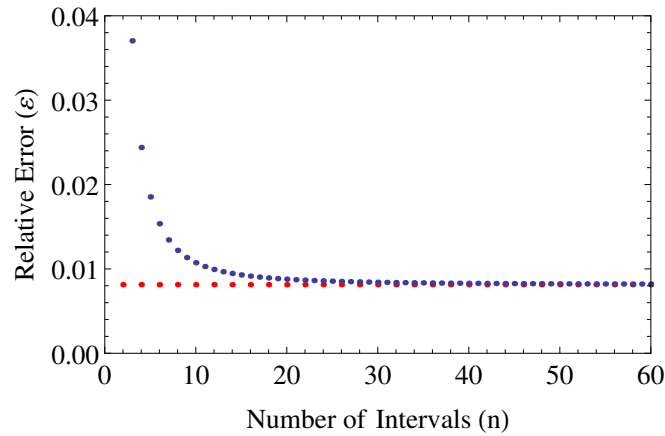
The maximum amount of truncation error (ξ_{max_T}) for Simpson's Rule may be determined as follows:

$$\xi_{max_T} = \frac{(b-a)^5 \left| f_{(max)}^{(4)} \right|}{180 n^4}$$

The minimum number of intervals needed to approximate an integral to within a given accuracy (ξ_{max}) may be determined by isolating n from the equation above:

$$n \geq \sqrt[4]{\frac{(b-a)^5 \left| f_{(max)}^{(4)} \right|}{180(\xi_{max})}}$$

Note: For Simpson's Rule, n must be an even and whole number

4

3. Three-Point Gaussian Quadrature (GQ):

Gaussian Quadrature is a numerical method of integration. Of the three methods discussed, GQ obtains the most accurate approximations or equivalent accuracy with less intervals. It does this by evaluating $f(x)$ at an abscissas which is found by determining the roots of an orthogonal polynomial of the same interval and weight. GQ is not limited to three-points, and is optimal as it can exactly fit and n-degree polynomial.

The formula for three-point Gaussian Quadrature is as follows:

$$\int_a^b f(x)\,dx \approx c_1 f(x_1) + c_2 f(x_2) + c_3 f(x_3),\text{ where}$$
$x_1 = -0.7746, x_2 = 0, x_3 = 0.7746$ and weights $c_1 = 0.555\overline{6}, c_2 = 0.888\overline{9}$ and $c_3 = 0.555\overline{6}$

1.3   Numerical Differentiation Techniques

1. Forward Divided Difference (FDD)

Forward difference is a numerical method of differentiation. It employs the limit definition of a derivative but substitutes $(\Delta x \to 0)$ with a finite step size (h). The forward difference formula is as follows:

$f'(x) \approx \dfrac{f(x + \Delta x) - f(x)}{\Delta x}$ where Δx is a finite term (h).

FDD Error:

Because $(\Delta x \to 0)$ is approximated by a finite step size (h), a truncation error will result. To minimize the truncation error, h should be taken sufficient small. However, there is a computational limit to how small h may be. Machine Epsilon (ϵ) is the largest number that when added to another number, does not change it. This number is specific to both the systems architecture and the data structure storing the number. In my case, my machine epsilon (ϵ) for a double is $2.2204460492503131 \times 10^-16$. If h is chosen to be $\geq \epsilon$ then ξ_{max} will be ≥ 1. Therefore, to minimize error, and thereby maximize accuracy, an h should be chosen such that the truncation error and round-off error or approximately the same.

The following $LogLog$-plot depicts the relative error(ϵ) versus step size (h) of the FDD approximations for: $x^2 (blue)$, $x^3 (red)$, $e^{-x} (green)$ and $sin(x)ln(x) (brown)$

.
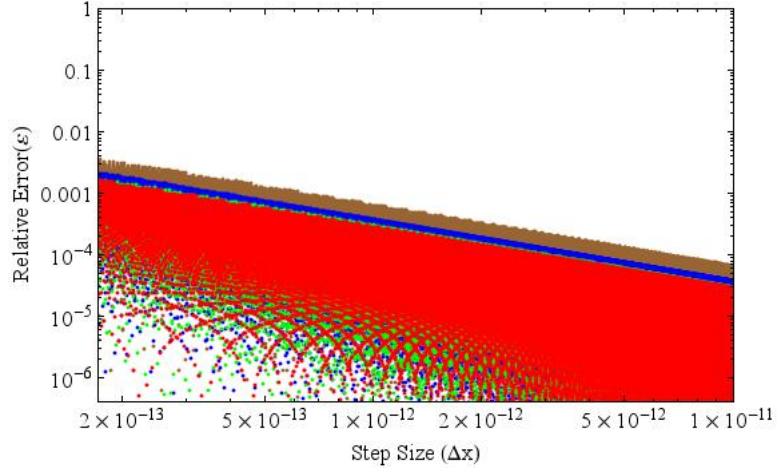
2. Central Divided Difference (CDD)

The Central Difference is similar to FDD except that CDD is evaluated at half of the step size. Due to this, CDD has an additional order of accuracy for smooth functions. The Central Fifference formula is as follows:

$$f'(x) \approx \frac{f(x + \Delta x) - f(x)}{2\Delta x} \text{ where } \Delta x \text{ is a finite term } (h).$$

The following *LogLog*-plot depicts the relative error(ϵ) versus step size (h) of the CDD approximations for: $x^2(blue)$, $x^3(red)$, $e^{-x}(green)$ and $sin(x)ln(x)(brown)$

.
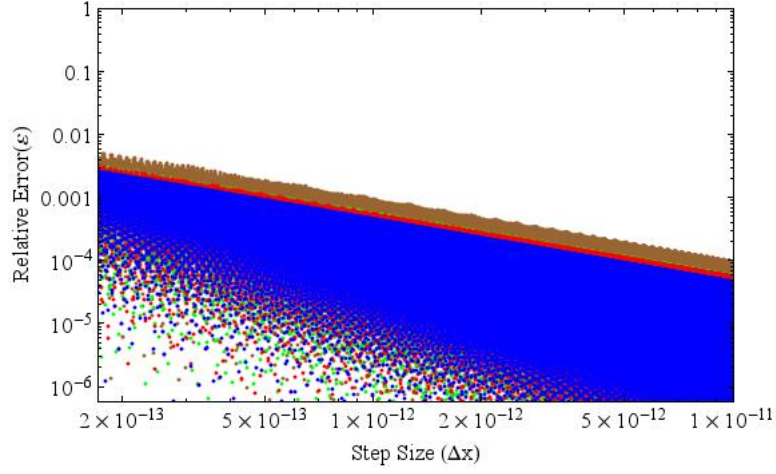
3. Five Point Stencil (FPS)y
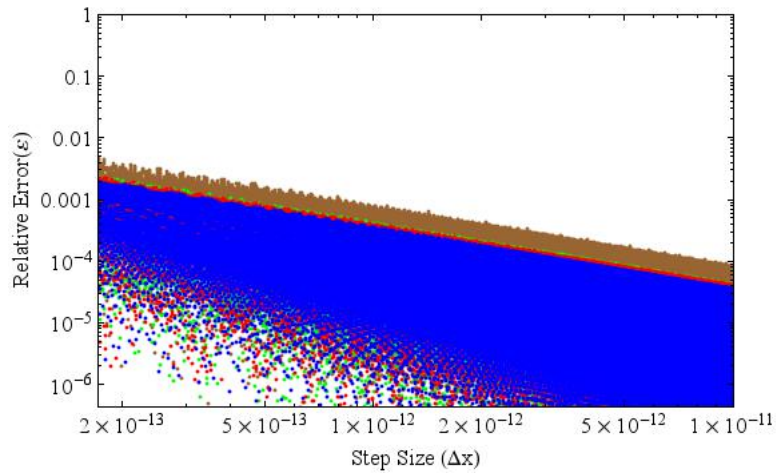   The formula for the FPS method is as follows:

   $$f'(x) \approx \frac{-f(x + \Delta x) + 8f(x + \Delta x) - 8f(x - \Delta x) + f(x - 2\Delta x)}{12\Delta x}$$

   where Δx is a finite term (h).

   Truncation error due to the FPS may be estimated as follows:

   $$\xi_{max} = f'(x) - \frac{1}{30}f^5(x)h^4 + O(h^5)$$

   The following *LogLog*-plot depicts the relative error(ϵ) versus step size
   (h) of the FPS approximations for: $x^2 (blue)$, $x^3 (red)$, $e^{-x} (green)$ and
   $sin(x)ln(x)(brown)$

.

4. Bisection Method

The bisection method is an order n ??? algorithm which can pinpoint the root of most functions. Several conditions must be satisfied in order for the method to prove successful. First, the user must have determined an approximate location for the root. This is necessary because the algorithm requires an interval whose upper and lower bound must contain the root. The principle function must also be real, continuous (within the interval) and have opposite signs at the initial upper and lower bounds as depicted below. If these preconditions are met, the bisection is guaranteed to converge on a root.
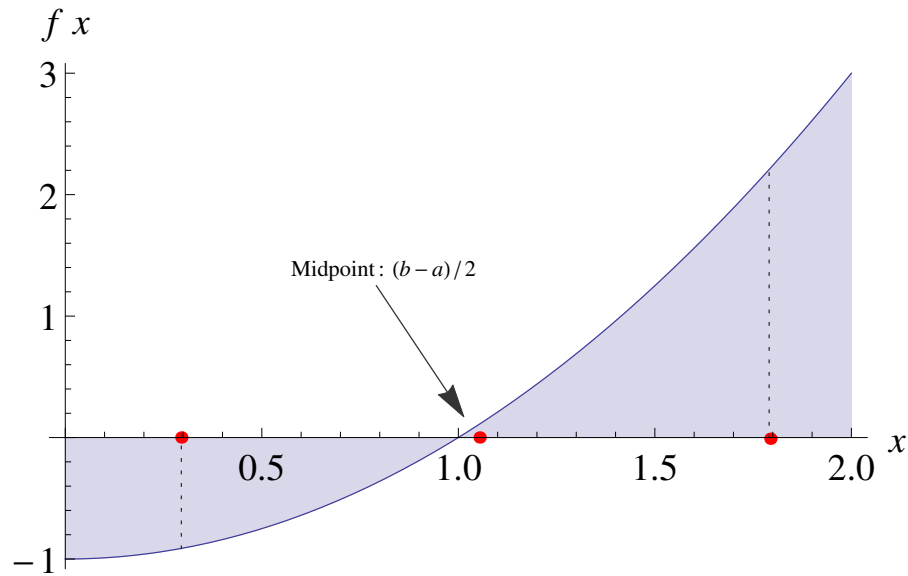
Algorithm:

The method is implemented by recursive comparison of the sign of the function at a specified lower (or upper) boundary with the sign of the function at a calculated midpoint between the upper and lower boundaries. If the product of the signs are negative, then the midpoints value is assigned to the the upper boundary and a new midpoint is calculated for the next iteration. If the product of lower bound and the newly calculated midpoint is positive, then the value of midpoint is assigned to the lower boundary and another midpoint is calculated for the next iteration. This process is repeated until the product of the lower bound and the midpoint is 0. The zero product indicates discovery of the root: $f(midpoint) = 0$.

Considerations:

While the bisection method is guaranteed to work, its preconditions are stringent and its application may be limited for certain functions. For example, $f(x) = x^2$. Here, since the function is $\geq 0 \; \forall x$, the bisection methods preconditions cannot be satisfied. The precondition are unmet because the value of the function at the initial boundaries have identical signs. By definition, the bisection method requires opposite signs and so the routine goes uninitialized.

Depending on the choice of boundaries, bisection can require excessive iterations. If one of the boundaries is chosen sufficiently close to the root while the other is not, it will take many iterations before of $\frac{|upper - lower|}{2}$ before the algorithm can converge. The number of iterations increase as $(upper)$ is chosen further from the root and $(lower)$ is chosen closer.

The maximum error of the bisection method may be controlled by the initial selection of the boundaries. That is, the true error can be no larger then $(upper - lower)$. Due to the implementation of the midpoint, the error is halved for each iteration.

5. The Newton Raphson Method (NRM)
   The Newton Raphson method is an order n ??? algorithm which can locate the root of most functions. Unlike the bisection algorithm with its many preconditions, the NRM requires only a single guess (seed) from the user from the user from which it begins its search. While NRM does not require previous knowledge of the location of the root, it is also not guaranteed to converge on a root.

Algorithm:

NRM is implemented by recursively comparing positions on the x-axis until the relative error between them meets some user specified tolerance or maximum iteration. The method is depicted as follows:
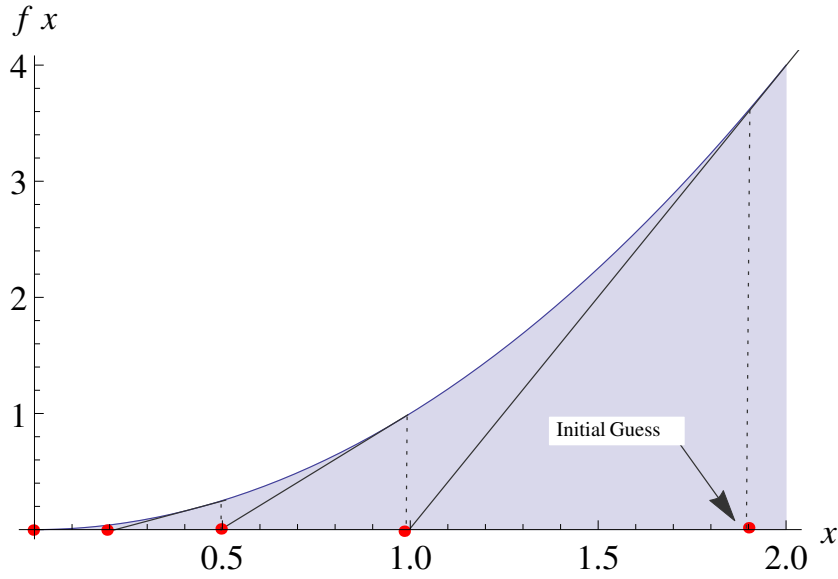
The initial guess is the first point (x_0). It is provided as a seed by the user from which the method begins its search. To determine the second point, the tangent of the function at $f(x_0)$ is extended out to x_1 on the x-axis.

. $$\text{Here, } x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

The above process is repeated until a specified number of iterations are reached or the relative error between two points, $\xi = \dfrac{|x_1 - x_0|}{x_1}$, is less then a specified accuracy. Once the maximum iteration or accuracy is met, x_i is determined to be the root.

Considerations:

While the NRM requires little input from the user, it is not guaranteed that the method will converge on any or all roots of a function. For example, note how in the depictions below, $x^2 + 2$ may diverge and $sin(x)$ may fail to find all roots: