College of Engineering and Computer Science
Mechanical Engineering Department
Numerical Analysis Notes
November 4, 2017 Instructor: Larry Caretto

Numerical Differentiation

Introduction

These notes provide a basic introduction to numerical differentiation using finite- difference grids. They consider the interplay between truncation error and roundoff error.

Finite-difference grids

In a finite-difference grid, a region is subdivided into a set of discrete points. The spacing between the points may be uniform or non-uniform. For example, a grid in the x direction, $x_{min} \leq x \leq x_{max}$ may be written as follows. First, we place a series of N+1 nodes numbered from zero to N in this region. The coordinate of the first node, x_0 equals x_{min}. The final grid node, $x_N = x_{max}$. The spacing between any two grid nodes, x_i and x_{i-1}, has the symbol Δx_i. These relations are summarized as equation [1].

$$x_0 = x_{min} \qquad x_N = x_{max} \qquad x_i - x_{i-1} = \Delta x_i \qquad [1]$$

A non-uniform grid, with different spacing between different nodes, is illustrated below.

●---------●------------●-----------------●---~   ~-------●----------●------●

x_0 x_1 x_2 x_3 x_{N-2} x_{N-1} x_N

For a uniform grid, all values of Δx_i are the same. In this case, the uniform grid spacing, in a one-dimensional problem is usually given the symbol h. I.e., $h = x_i - x_{i-1}$ for all values of i.

In these notes, we will limit our consideration to one-dimensional finite-difference problems. However, advanced courses consider multiple space dimensions discussed in equations [2] and [3] below.

In two space dimensions a grid is required for both the x and y, directions, which results in the following grid and geometry definitions, assuming that there are M+1 grid nodes in the y direction.

$$\begin{array}{lll} x_0 = x_{min} & x_N = x_{max} & x_i - x_{i-1} = \Delta x_i \\ y_0 = y_{mjn} & y_M = y_{may} & y_j - y_{j-1} = \Delta y_j \end{array} \qquad [2]$$

For a three-dimensional transient problem there would be four independent variables: the three space dimensions, x, y and z, and time. Each of these variables would be defined at discrete points, i.e.

$$\begin{array}{lll} x_0 = x_{min} & x_N = x_{max} & x_i - x_{i-1} = \Delta x_i \\ y_0 = y_{mjn} & y_M = y_{may} & y_j - y_{j-1} = \Delta y_j \\ z_0 = z_{mkn} & z_K = z_{maz} & z_k - z_{k-1} = \Delta z_k \\ t_0 = t_{min} & t_L = t_{may} & t_n - t_{n-1} = \Delta t_n \end{array} \qquad [3]$$

Jacaranda Hall 3314
Email: lcaretto@csun.edu
Mail Code 8348
Phone: N/A
Fax: 818.677.7062

Any dependent variable such as u(x,y,z,t) in a continuous representation would be defined only at discrete grid points in a finite-difference representation. The following notation is used for a one-dimensional problem.

$$f_k = f(x_k) \quad [4]$$

This notation can be extended to problems in more than one dimension including transient problems. In the most complex case the notation $u_{ijk}^n = u(x_i, y_j, z_k, t_n)$ is used to denote the value of the dependent at a particular point in the region, (x_i, y_j, z_k, t_n) where the variable is defined.

Finite-difference Expressions Derived from Taylor Series

The Taylor series provides a simple tool for deriving finite-difference approximations. It also gives an indication of the error caused by the finite difference expression. Recall that the Taylor series for a function of one variable, f(x), expanded about some point x = a, is given by the infinite series,

$$f(x) = f(a) \quad + \left.\frac{df}{dx}\right|_{x=a} (x-a) \quad + \frac{1}{2!}\left.\frac{d^2 f}{dx^2}\right|_{x=a} (x-a)^2 \quad + \frac{1}{3!}\left.\frac{d^3 f}{dx^3}\right|_{x=a} (x-a)^3 + ... \quad [5]$$

The "x = a" subscript on the derivatives reinforces the fact that these derivatives are evaluated at the expansion point, x = a. We can write the infinite series using a summation notation as follows:

$$f(x) = \sum_{n=0}^{\infty} \frac{1}{n!} \left.\frac{d^n f}{dx^n}\right|_{x=a} (x-a)^n \quad [6]$$

In the equation above, we use the definitions of 0! = 1! = 1 and the definition of the zeroth derivative as the function itself. I.e., $d^0f/dx^0|_{x=a} = f(a)$.

If the series is truncated after some finite number of terms, say m terms, the omitted terms are called the **truncation error**. These omitted terms are also an infinite series. This is illustrated below.

$$\underbrace{f(x) = \sum_{n=0}^{m} \frac{1}{n!} \left.\frac{d^n f}{dx^n}\right|_{x=a} (x-a)^n}_{Terms\ used} + \underbrace{\sum_{n=m+1}^{\infty} \frac{1}{n!} \left.\frac{d^n f}{dx^n}\right|_{x=a} (x-a)^n}_{Truncation\ error} \quad [7]$$

In this equation the second sum represents the truncation error, ε_m, from truncating the series after m terms.

$$\varepsilon_m = \sum_{n=m+1}^{\infty} \frac{1}{n!} \left.\frac{d^n f}{dx^n}\right|_{x=a} (x-a)^n \quad [8]$$

The theorem of the mean can be used to show that the infinite-series truncation error can be expressed in terms of the first term in the truncation error, that is

$$\varepsilon_m = \frac{1}{(m+1)!} \left. \frac{d^{m+1} f}{dx^{m+1}} \right|_{x=\xi} (x - a)^{m+1} \qquad [9]$$

Here the subscript, "x = ξ", on the derivative indicates that this derivative is no longer evaluated at the known point x = a, but is to be evaluated at x = ξ, an unknown point between x and a. Thus, the price we pay for reducing the infinite series for the truncation error to a single term is that we lose the certainty about the point where the derivative is evaluated. In principle, this would allow us to compute a bound on the error by finding the value of ξ, between x and a, that made the error computed by equation [9] a maximum. In practice, we do not usually know the exact functional form, f(x), let alone its $(m+1)^{th}$ derivative.

In using Taylor series to derive the basic finite-difference expressions, we start with uniform one-dimensional grid spacing. The difference, Δx_i, between any two grid points is the same and is given the symbol, h. This uniform grid can be expressed as follows.

$$\Delta x_i = x_i - x_{i-1} = h \quad \text{or} \quad x_i = x_0 + ih \qquad \text{for all } i = 0,\ldots,N \qquad [10]$$

Various increments in x at any point along the grid can be written as follows:

$$x_{i+1} - x_{i-1} = x_{i+2} - x_i = 2h \qquad x_{i-1} - x_i = x_i - x_{i+1} = -h \qquad x_{i-1} - x_{i+1} = x_i - x_{i+2} = -2h \qquad [11]$$

Using the increments in x defined above and the notation $f_i = f(x_i)$ the following Taylor series can be written using expansion about the point $x = x_i$ to express the values of f at some specific grid points, x_{i+1} , x_{i-1} , x_{i+2} and x_{i-2}. The conventional Taylor series expression for f(x) in equation [5] can be adapted for use in finite differences by writing an expansion equation about a particular grid point, $x = x_i$, to determine the value of f(x) at another grid point, x_{i+k}. From equation [10], we see that $x_{i+k} = x_i + kh$ so that $f(x_{i+k}) = f(x_i + kh)$. The difference, Δx, in the independent variable, x, between the evaluation point, $x_i + kh$, and the expansion point, x_i, is equal to kh. Using $x_i = a$ as the expansion point and kh as Δx allows us to rewrite equation [5] as shown below.

$$f(x_i + kh) = f(x_i) \quad + \left.\frac{df}{dx}\right|_{x=x_i} kh \quad + \frac{1}{2!}\left.\frac{d^2 f}{dx^2}\right|_{x=x_i} (kh)^2 \quad + \frac{1}{3!}\left.\frac{d^3 f}{dx^3}\right|_{x=x_i} (kh)^3 + \ldots.. \qquad [12]$$

The next step is to use the notation that $f(x_i + kh) = f_{i+k}$, and the following notation for the n^{th} derivative, evaluated at $x = x_i$.

$$f_i^{'} = \left.\frac{df}{dx}\right|_{x=x_i} \qquad f_i^{''} = \left.\frac{d^2 f}{dx^2}\right|_{x=x_i} \qquad \ldots \qquad f_i^{n} = \left.\frac{d^n f}{dx^n}\right|_{x=x_i} \qquad [13]$$

With these notational changes, the Taylor series in equation [12] can be written as follows.

$$f_{i+k} = f_i + f_i^{'} kh + f_i^{''}\frac{(kh)^2}{2!} + f_i^{'''}\frac{(kh)^3}{3!} + \ldots.. \qquad [14]$$

Finite-difference expressions for various derivatives can be obtained by writing the Taylor series shown above for different values of k, combining the results, and solving for the derivative. The simplest example of this is to use only the series for k = 1.

$$f_{i+1} = f_i + f_i^{'}h + f_i^{''}\frac{h^2}{2} + f_i^{'''}\frac{h^3}{6} + ... \quad [15]$$

We can rearrange this equation to solve for the first derivative, f'_i; recall that this is the first derivative at the point $x = x_i$.

$$f_i^{'} = \frac{f_{i+1} - f_i}{h} - f_i^{''}\frac{h}{2} - f_i^{'''}\frac{h^2}{6} + ... = \frac{f_{i+1} - f_i}{h} + O(h) \quad [16]$$

The first term to the right of the equal sign gives us a simple expression for the first derivative; it is simply the difference in the function at two points, $f(x_i+h) - f(x_i)$, divided by h, which is the difference in x between those two points. The remaining terms in the first form of the equation are an infinite series. That infinite series gives us an equation for the error that we would have if we used the simple finite difference expression to evaluate the first derivative.

Representing the truncation error as the order of the error

As noted above, we can replace the infinite series for the truncation error by the leading term in that series. Remember that we pay a price for this replacement; we no longer know the point at which the leading term is to be evaluated. Because of this we often write the truncation error as shown in the second equation. Here we use a capital oh followed by the grid size in parentheses. In general, the grid size is raised to some power. (Here we have the first power of the grid size, $h = h^1$.) For a truncation error proportional to the n^{th} power of the step size we would use the notation, $O(h^n)$. This notation tells us how the truncation error depends on the step size.

The order of the error dependence on the step size is an important concept. If the error is proportional to h, cutting h in half would cut the error in half. If the error is proportional to h^2, then cutting the step size in half would reduce the error by ¼. When the truncation error is written with this $O(h^n)$ notation, we call n the **order of the error**. In two calculations, with step sizes h_1 and h_2, we expect the following relation between the truncation errors, ε_1 and ε_2 for the calculations.

$$\varepsilon_2 \approx \varepsilon_1\left(\frac{h_2}{h_1}\right)^n \quad [17]$$

We use the approximation sign (≈) rather than the equality sign in this equation because the error term also includes an unknown factor of some higher order derivative, evaluated at some unknown point in the region. The approximation shown in equation [17] would be an equality if this other factor were the same for both step sizes.

Another important idea about the order of the error is that an n^{th} order finite-difference expression will give an exact value for the derivative of an n^{th} order polynomial. Because a Taylor series is a polynomial series, it can represent a polynomial exactly if a sufficient number of terms are used. This is illustrated further below.

The expression for the first derivative that we derived in equation [16] is said to have a first order error. We can obtain a similar finite difference approximation by writing the general series in equation [14] for k = -1. This gives the following result.

$$f_{i-1} = f_i - f_i^{'}h + f_i^{''}\frac{h^2}{2} - f_i^{'''}\frac{h^3}{6} + ... \quad [18]$$

We can rearrange this equation to solve for the first derivative, f'_i; recall that this is the first derivative at the point $x = x_i$.

$$f_i^{'} = \frac{f_i - f_{i-1}}{h} + f_i^{''}\frac{h}{2} - f_i^{'''}\frac{h^2}{6} + ... = \frac{f_i - f_{i-1}}{h} + O(h) \quad [19]$$

Here again, as in equation [16], we have a simple finite-difference expression for the first derivative that has a first-order error. The expression in equation [16] is called a forward difference. It gives an approximation to the derivative at point i in terms of values at that point and points forward (in the +x direction) of that point. The expression in equation [19] is called a backwards difference for similar reasons.

Derivative expressions with higher order errors

An expression for the first derivative that has a second-order error can be found by subtracting equation [18] from equation [15]. When this is done, terms with even powers of h cancel giving the following result.

$$f_{i+1} - f_{i-1} = 2f_i^{'}h + 2f_i^{'''}\frac{h^3}{6} + 2f_i^{'''''}\frac{h^5}{120} + ... \quad [20]$$

Solving this equation for the first derivative gives the following result.

$$f_i^{'} = \frac{f_{i+1} - f_{i-1}}{2h} - f_i^{'''}\frac{h^2}{6} - f_i^{'''''}\frac{h^4}{120} + ... = \frac{f_{i+1} - f_{i-1}}{2h} + O(h^2) \quad [21]$$

The finite-difference expression for the first derivative in equation [21] is called a central difference. The point at which the derivative is evaluated, x_i, is central to the two points (x_{i+1} and x_{i-1}) at which the function is evaluated. The central difference expression provides a higher order (more accurate) expression for the first derivative as compared to the forward or backward derivatives. There is only a small amount of extra work (a division by 2) in getting this more accurate result. Because of their higher accuracy, central differences are usually preferred in finite difference expressions.

Central difference expressions are not possible at the start of end of a boundary. It is possible to get higher order finite difference expressions for such points by using more complex expressions. For example, at the start of a region, $x = x_0$, we can write the Taylor series in equation [14] for the first two points in from the boundary, x_1 and x_2, expanding around the boundary point, x_0.

$$f_1 = f_0 + f_0^{'}h + f_0^{''}\frac{h^2}{2} + f_0^{'''}\frac{h^3}{6} + ... \quad [22]$$

$$f_2 = f_0 + f_0^{'}(2h) + f_0^{''}\frac{(2h)^2}{2} + f_0^{'''}\frac{(2h)^3}{6} + ... \quad [23]$$

These equations can be combined to eliminate the h^2 terms. To start, we multiply equation [22] by 4 and subtract it from equation [23].

$$f_2 - 4f_1 = \left[f_0 + f_0^{'}(2h) + f_0^{''}\frac{(2h)^2}{2} + f_0^{'''}\frac{(2h)^3}{6} + ... \right] - 4\left[f_0 + f_0^{'}(h) + f_0^{''}\frac{(h)^2}{2} + f_0^{'''}\frac{(h)^3}{6} + .. \right]$$

This equation can be simplified as follows

$$f_2 - 4f_1 + 3f_0 = -f_0^{'}(2h) + 4f_0^{'''}\frac{(h)^3}{6} + ... \quad [24]$$

When this equation is solved for the first derivative at the start of the region a second order accurate expression is obtained.

$$f_0^{'} = \frac{-f_2 + 4f_1 - 3f_0}{2h} - f_0^{'''}\frac{h^2}{3} + ... = \frac{-f_2 + 4f_1 - 3f_0}{2h} + O(h^2) \quad [25]$$

A similar equation can be found at the end of the region, $x = x_N$, by obtaining the Taylor series expansions about the point $x = x_N$, for the values of f(x) at $x = x_{N-1}$ and $x = x_{N-2}$. This derivation parallels the derivation used to obtain equation [25]. The result is shown below.

$$f_N^{'} = \frac{f_{N-2} - 4f_{N-1} + 3f_N}{2h} + f_N^{'''}\frac{h^2}{3} + ... = \frac{f_{N-2} - 4f_{N-1} + 3f_N}{2h} + O(h^2) \quad [26]$$

Equations [25] and [26] give second-order accurate expressions for the first derivative. The expression in equation [25] is a forward difference; the one in equation [26] is a backwards difference.

The evaluation of three expressions for the first derivative is shown in Table 1. These are (1) the second-order, central-difference expression from equation [21], (2) the first-order, forward-difference from equation [16], and (3) the second-order, forward-difference from equation [25]. The first derivative is evaluated for $f(x) = e^x$. For this function, the first derivative, $df/dx = e^x$. Since we know the exact value of the first derivative, we can calculate the error in the finite difference results.

In Table 1, the results are computed for three different step sizes: h = 0.4, h = 0.2 and h = 0.1. The table also shows the ratio of the error as the step size is changed. The next-to-last column shows the ratio of the error for h = 0.4 to the error for h = 0.2. The final column shows the ratio of the error for h = 0.2 to the error for h = 0.1.

Table 1
Tests of Finite-Difference Formulae to Compute the First Derivative – f(x) = exp(x)

x	f(x)	Exact f'(x)	h = .4		h = .2		h = .1		Error Ratios	
			f'(x)	Error	f'(x)	Error	f'(x)	Error	(h=.4)/(h=.2)	(h=.2)/(h=.1)
Results using second-order central differences										
0.6	1.8221	1.8221								
0.7	2.0138	2.0138					2.0171	0.0034		
0.8	2.2255	2.2255			2.2404	0.0149	2.2293	0.0037		4.01
0.9	2.4596	2.4596			2.4760	0.0164	2.4637	0.0041		4.01
1.0	2.7183	2.7183	2.7914	0.0731	2.7364	0.0182	2.7228	0.0045	4.02	4.01
1.1	3.0042	3.0042			3.0242	0.0201	3.0092	0.0050		4.01
1.2	3.3201	3.3201			3.3423	0.0222	3.3257	0.0055		4.01

1.3	3.6693	3.6693					3.6754	0.0061		
1.4	4.0552	4.0552								
Results using first-order forward differences										
0.6	1.8221	1.8221	2.2404	0.4183	2.0171	0.1950	1.9163	0.0942	2.15	2.07
0.7	2.0138	2.0138	2.4760	0.4623	2.2293	0.2155	2.1179	0.1041	2.15	2.07
0.8	2.2255	2.2255	2.7364	0.5109	2.4637	0.2382	2.3406	0.1151	2.15	2.07
0.9	2.4596	2.4596	3.0242	0.5646	2.7228	0.2632	2.5868	0.1272	2.15	2.07
1.0	2.7183	2.7183	3.3423	0.6240	3.0092	0.2909	2.8588	0.1406	2.15	2.07
1.1	3.0042	3.0042			3.3257	0.3215	3.1595	0.1553		2.07
1.2	3.3201	3.3201			3.6754	0.3553	3.4918	0.1717		2.07
1.3	3.6693	3.6693					3.8590	0.1897		
1.4	4.0552	4.0552								
Results using second-order forward differences										
0.6	1.8221	1.8221	1.6895	0.1327	1.7938	0.0283	1.8156	0.0066	4.69	4.32
0.7	2.0138	2.0138			1.9825	0.0313	2.0065	0.0072		4.32
0.8	2.2255	2.2255			2.1910	0.0346	2.2175	0.0080		4.32
0.9	2.4596	2.4596			2.4214	0.0382	2.4508	0.0088		4.32
1.0	2.7183	2.7183			2.6761	0.0422	2.7085	0.0098		4.32
1.1	3.0042	3.0042					2.9934	0.0108		
1.2	3.3201	3.3201					3.3082	0.0119		
1.3	3.6693	3.6693								
1.4	4.0552	4.0552								

For the second-order formulae, the error ratios in the last two columns of Table 2-1 are about 4, showing that the second-order error increases by a factor of 4 as the step size is doubled. For the first order expression, these ratios are about 2. This shows that the error increases by the same factor as the step size for the first order expressions. The expected values of the error ratios are only obtained in the limit of very small step sizes. We see that the values in the last column of this table (where the actual values of h are smaller than they are in the next-to-last column) are closer to the ideal error ratio.

The techniques that have been used here to derive forward, backward, and central derivative expressions with first- and second-order error can be expanded to consider higher order derivatives and higher order errors. Lists of various finite-difference formulas can be found in numerical analysis texts.

Roundoff error

Truncation errors are not the only kind of error that we encounter in finite difference expressions. As the step sizes get very small the terms in the numerator of the finite difference expressions become very close to each other. We lose significant figure when we do the subtraction. For example, consider the previous problem of finding the numerical derivative of $f(x) = e^x$. Pick $x = 1$ as the point where we want to evaluate the derivative. With $h = 0.1$ we have the following data for calculating the derivative by the central-difference formula in equation [21].

$$f_i^{'} = f'(x) = \frac{f_{i+1} - f_{i-1}}{2h} = \frac{f(x+h) - f(x-h)}{2h} = \frac{3.004166 - 2.722815}{2(0.1)} = 2.722815$$

Since the first derivative of e^x is e^x, the correct value of the derivative at x = 1 is e^1 = 2.718282; so the error in this value of the first derivative is 4.5x10^{-3}. For h = 0.0001, the numerical value of the first derivative is found as follows.

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} = \frac{2.7185536702 - 2.7180100139}{2(0.0001)} = 2.718281832990$$

Here, the error is 4.5x10^{-9}. This looks like our second-order error. We cut the step size by a factor of 1,000 and our error decreased by a factor of 1,000,000, as we would expect for a second order error. We are starting to see potential problems in the subtraction of the two numbers in the numerator. Because the first four digits are the same, we have lost four significant figures in doing this subtraction. What happens if we decrease h by a factor of 1,000 again? Here is the result for h = 10^{-7}.

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} = \frac{2.71828210028724 - 2.71828155660388}{2(0.0000001)} = 2.7182851763$$

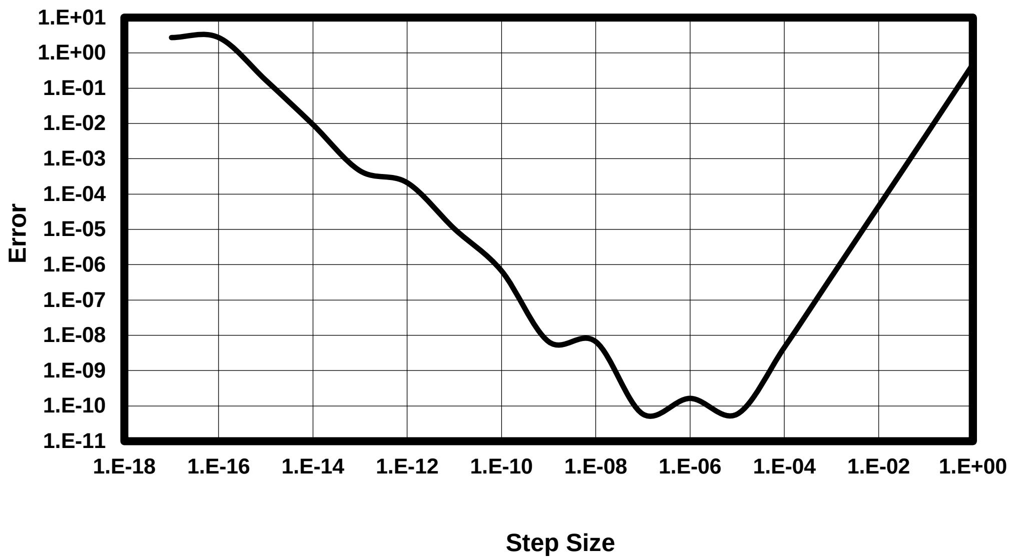
Our truncation analysis leads us to expect another factor of one million in the error reduction as we decrease the step size by 1,000. This should give us an error of 4.5x10^{-15}. However, we find that the actual error is 5.9x10^{-9}. We see the reason for this in the numerator of the finite difference expression. As the difference between f(x+h) and f(x-h) shrinks, we are taking the difference of nearly equal numbers. This kind of error is called **roundoff error** because it results from the necessity of a computer to round off real numbers to some finite size. (These calculations were done with an excel spreadsheet which has about 15 significant figures. Figure 2-1 shows the effect of step size on error for a large range of step sizes.

For the large step sizes to the right of Figure 2-1, the plot of error versus step size appears to be a straight line on this log-log plot. This is consistent with equation [17]. If we take logs of both sides of that equation and solve for n, we get the following result.

$$n \approx \frac{\log\left(\frac{\varepsilon_2}{\varepsilon_1}\right)}{\log\left(\frac{h_2}{h_1}\right)} = \frac{\log(\varepsilon_2) - \log(\varepsilon_1)}{\log(h_2) - \log(h_1)} \qquad [27]$$

Equation [27] shows that the order of the error is just the slope of a log(error) versus log(h) plot. If we take the slope of the straight-line region on the right of Figure 2-1, we get a value of approximately two for the slope, confirming the second order error for the central difference expression that we are using here. However, we also see that as the step size reaches about 10^{-5}, the error starts to level off and then increase. At very small step sizes the numerator of the finite-difference expression becomes zero on a computer and the error is just the exact value of the derivative.

Figure 1. Effect of Step Size on Error



Final Observations on Finite-Difference Expressions from Taylor Series

The notes above have focused on the general approach to the derivation of finite-difference expressions using Taylor series. Such derivations lead to an expression for the truncation error. That error is due to omitting the higher order terms in the Taylor series. We have characterized that truncation error by the power or order of the step size in the first term that is truncated. The truncation error is an important factor in the accuracy of the results. However, we also saw that very small step sizes lead to roundoff errors that can be even larger than truncation errors.

The use of Taylor series to derive finite difference expressions can be extended to higher order derivatives and expressions that are more complex, but have a higher order truncation error. One expression that will be important for subsequent course work is the central-difference expression for the second derivative. This can be found by adding equations [15] and [18].

$$f_{i+1} + f_{i-1} = 2f_i + 2f_i'' \frac{h^2}{2} + 2f_i'''' \frac{h^4}{24} + ... \qquad [28]$$

We can solve this equation to obtain a finite-difference expression for the second derivative.

$$f_i'' = \frac{f_{i+1} + f_{i-1} - 2f_i}{h^2} - f_i'''' \frac{h^2}{12} + ... = \frac{f_{i+1} + f_{i-1} - 2f_i}{h^2} + O(h^2) \qquad [29]$$

Although we have been deriving expressions here for ordinary derivatives, we will apply the same expressions to partial derivatives. For example, the expression in equation [29] for the second derivative could represent d^2f/dx^2 or $\partial^2 f/\partial x^2$.

The Taylor series we have been using here have considered x as the independent variable. However, these expressions can be applied to any coordinate direction or time.

Although we have used Taylor series to derive the finite-difference expressions, they could also be derived from interpolating polynomials. In this approach, one uses numerical methods for developing polynomial approximations to functions, then takes the derivatives of the approximating polynomials to approximate the derivatives of the functions. A finite-difference expression with an n^{th} order error that gives the value of any quantity should be able to represent the given quantity exactly for an n^{th} order polynomial.*

The expressions that we have considered are for constant step size. It is also possible to write the Taylor series for variable step size and derive finite difference expressions with variable step sizes. Such expressions have lower-order truncation error terms for the same amount of work in computing the finite difference expression.

In solving differential equations by finite-difference methods, the differential equation is replaced by its finite difference equivalent at each node. This gives a set of simultaneous algebraic equations that are solved for the values of the dependent variable at each grid point.

Finite difference expressions can be derived from Taylor series. This approach leads to an expression for the truncation error that provides us with knowledge of how this error depends on the step size. This is called the **order of the error**.

In finite-difference approaches, we need to be concerned about both truncation errors and roundoff errors. Roundoff errors were more of a concern in earlier computer applications where limitations on available computer time and memory restricted the size of real words, for many practical applications, to 32 bits. This corresponds to the single precision type in Fortran or the Single type in VBA. With modern computers, it is possible to do routine calculations using 64-bit (or higher precision) real words. This corresponds to the double precision type in Fortran* or the double type in VBA. The 32-bit real word allows about 7 significant figures; the 64-bit real word allows almost 16 significant figures.

---

* If a second order polynomial is written as $y = a + bx + cx^2$; its first derivative at a point $x = x_0$ is given by the following equation: $[dy/dx]_{x=x0} = b + 2cx_0$. If we use the second-order central-difference expression in equation [21] to evaluate the first derivative, we get the same result as shown below:

$$\left.\frac{dy}{dx}\right|_{x=x_0} = \frac{y(x_0+h) - y(x_0-h)}{2h} = \frac{a + b(x_0+h) + c(x_0+h)^2 - [a + b(x_0-h) + c(x_0-h)^2]}{2h}$$

$$= \frac{2bh + c(x_0^2 + 2x_0h + h^2) - c(x_0^2 - 2x_0h + h^2)}{2h} = \frac{2bh + 4cx_0h}{2h} = b + 2cx_0$$

* Also known as real(8) or real(KIND=8) in Fortran 90 and later versions; single precision is typed as real, real(4) or real(KIND=4) in these versions of Fortran.