Section 3 Newton Divided-Difference Interpolating Polynomials

The standard form for representing an nth order interpolating polynomial is straightforward. There are however, other representations of nth order polynomials which on the surface may seem a bit more unwieldy, but require less manipulation to arrive at. One such form is the subject of this section.

Given a set of $n+1$ data points (x_i, y_i), i = 0, 1, 2, …., n where the x_i are all different and the y_i are sampled from an underlying function $y = f(x)$, the nth order interpolating polynomial can be expressed as

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + ... + b_n(x - x_0)(x - x_1)\cdots\cdots(x - x_{n\text{-}1}) \qquad (3.1)$$

Before we consider how to determine the coefficients, observe that Equation (3.1) is in fact an nth order polynomial as evidenced by the last term which includes the highest power of x, namely x^n. All powers of x are present in Equation (3.1) despite the fact that the coefficients of say x^0, x^1, x^2,…etc. are not as obvious as when the polynomial is expressed in standard form (see Equation 2.1). Since there are $n+1$ independent coefficients b_i available, $i = 0, 1, 2,\ldots, n$ we can be assured there is at most an nth order polynomial that passes through the given data points.

The rationale for selecting the analytical form in Equation (3.1) will be apparent after we look at a few simple examples.

A) Given (x_0, y_0), (x_1, y_1) where $y_0 = f(x_0)$ and $y_1 = f(x_1)$

The first order polynomial for the case when $n = 1$ through the two data points is

$$f_1(x) = b_0 + b_1(x - x_0) \qquad (3.2)$$

Substitution of the two data points (x_0, y_0), (x_1, y_1) into Equation (3.2) gives

$$f_1(x_0) = b_0 + b_1(x_0 - x_0) \qquad (3.3)$$

$$f_1(x_1) = b_0 + b_1(x_1 - x_0) \qquad (3.3a)$$

Solving for b_0 and b_1 yields,

$$b_0 = f_1(x_0) \qquad (3.4)$$

$$b_1 = \frac{f_1(x_1) - f_1(x_0)}{x_1 - x_0} \tag{3.4a}$$

By design, the interpolating function $f_1(x)$ and the actual function $f(x)$ from which the data points were obtained are equal at $x = x_0$ and $x = x_1$ (see Figure 2.2). As a result, b_0 and b_1 can be expressed in terms of the given data,

$$b_0 = f(x_0) \tag{3.5}$$

$$b_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \tag{3.5a}$$

Example 3.1

The monthly payment on a 30-yr mortgage of $100,000. for two different annual interest rates is given in Table 3.1 below. Use an interpolation formula in the form of Equation (3.2) to estimate the monthly payment corresponding to an interest rate of 8.25 % per year.

Data Point Number k	Annual Interest Rate i_k	Monthly Payment $A_k = f(i_k)$
0	7 %	$ 665.30
1	10 %	$ 877.57

Table 3.1 Monthly Payments for $100,000 30-yr Mortgage with Different Interest Rates: Two Data Points

The first order interpolating polynomial is written

$$f_1(i) = b_0 + b_1(i - i_0) \tag{3.6}$$

where

$$b_0 = f(i_0) = f(7) = 665.30$$

$$b_1 = \frac{f(i_1) - f(i_0)}{i_1 - i_0} = \frac{f(10) - f(7)}{10 - 7} = \frac{877.57 - 665.3}{10 - 7} = 70.76$$

The estimated monthly payment is therefore

$$f_1(8.25) = 665.3 + 70.76(8.25 - 7) = 753.68$$

B) Given (x_0, y_0), (x_1, y_1), (x_2, y_2) where $y_0 = f(x_0)$, $y_1 = f(x_1)$, and $y_2 = f(x_2)$

The second order polynomial is written

$$f_2(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) \tag{3.7}$$

where b_0, b_1, and b_2 are determined using the same procedure employed in the previous example.

$$f_2(x_0) = b_0 + b_1(x_0 - x_0) + b_2(x_0 - x_0)(x_0 - x_1) \tag{3.8}$$

$$f_2(x_1) = b_0 + b_1(x_1 - x_0) + b_2(x_1 - x_0)(x_1 - x_1) \tag{3.8a}$$

$$f_2(x_2) = b_0 + b_1(x_2 - x_0) + b_2(x_2 - x_0)(x_2 - x_1) \tag{3.8b}$$

Solving for b_0, b_1, and b_2 gives

$$b_0 = f(x_0) \tag{3.9}$$

$$b_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0} \tag{3.9a}$$

$$b_2 = \frac{\left[\dfrac{f(x_2) - f(x_1)}{x_2 - x_1}\right] - \left[\dfrac{f(x_1) - f(x_0)}{x_1 - x_0}\right]}{x_2 - x_0} \tag{3.9b}$$

Once again, each $f_2(x_i)$ was replaced by $f(x_i)$ so the coefficients could be expressed in terms of known quantities.

There are definite advantages to representing interpolating polynomials in the nonstandard form. In the previous example, there is a sharp contrast in the method of solving for the three coefficients between the standard form representation of a second order polynomial in Equation 2.7 and the nonstandard form of Equation (3.7). In standard form, a_0, a_1, and a_2 are obtained by solving the system of simultaneous equations given in Equation 2.10. On the other hand, b_0, b_1, and b_2 are obtained sequentially as indicated in Equations (3.9), (3.9a) and (3.9b).

Given the choice, a sequential solution is usually preferable. The savings in computation is real. It is faster to evaluate explicit formulas for a set of coefficients one at a time than it is to implement a matrix-based solution where the coefficients are obtained in a parallel fashion, i.e. solution to a system of simultaneous equations. This argument may be less convincing when using calculators or computers with programs designed to solve simultaneous equations. Nonetheless, the solutions are obtained in fundamentally different ways.

The second advantage of the nonstandard representation is more compelling. The expressions for b_0 and b_1 in Equations (3.9) and (3.9a) are the same as in Equation (3.5). Why is this important? Suppose you just completed the process of finding an interpolating polynomial for a given set of data points. There may be some doubt in your mind concerning the accuracy of results based on the use of this polynomial. Later on there will be a discussion of quantitative methods for approximating the errors inherent in interpolation. More data points may be required to reduce the estimated errors. Incorporating additional data points is easier with the nonstandard representation for the interpolating polynomials. This is because each additional data point requires the computation of a single coefficient for the new term in the polynomial. This is illustrated in the following example which extends the results obtained in Example 3.1.

Example 3.2

Suppose we obtain two additional data points in the previous example dealing with the estimation of mortgage payments. The new data points correspond to 8 % and 9 % loans. Use one of the two additional points to obtain a second order interpolating polynomial. Estimate the monthly payment for an 8.25 % loan.

We must choose one of the two new data points to find the second order interpolating polynomial $f_2(i)$. Our intuition suggests the data point for an 8 % loan is the wiser choice because its closer to the point where the estimate is required, i.e. 8.25 %. The second order polynomial is obtained from the data points corresponding to k = 0, 1, and 2 in the following table. It is given in Equation (3.10).

Data point Number k	Annual Interest Rate i_k	Monthly Payment $A_k = f(i_k)$
0	7 %	\$ 665.30
1	10 %	\$ 877.57
2	8 %	\$ 733.76
3	9 %	\$ 804.62

Table 3.2 Monthly Payments for \$100,000 30-yr Mortgage with Different Interest Rates: 4 Data Points

$$f_2(i) = b_0 + b_1(i - i_0) + b_2(i - i_0)(i - i_1) \tag{3.10}$$

Coefficients b_0 and b_1 were determined in Example 3.1 using the first two data points. The remaining coefficient b_2 is obtained from Equation (3.9b) as

$$b_2 = \frac{\left[\dfrac{f(i_2)-f(i_1)}{i_2-i_1}\right]-\left[\dfrac{f(i_1)-f(i_0)}{i_1-i_0}\right]}{i_2-i_0}$$

$$= \frac{\left[\dfrac{733.76-877.57}{8-10}\right]-\left[\dfrac{877.57-665.30}{10-7}\right]}{8-7}$$

$$= 1.148$$

The polynomial $f_2(i)$ is therefore,

$$f_2(i) = 665.3 + 70.76(i - i_0) + 1.148(i - i_0)(i - i_1) \tag{3.11}$$

and the estimate of payments for an 8.25 % mortgage is now

$$f_2(8.25) = 665.3 + 70.76(8.25-7) + 1.148(8.25-7)(8.25-10)$$

$$= 751.24$$

The previous two examples are somewhat academic in nature. Quite obviously, the loan officer at the bank will not be interested in your calculations to estimate the monthly payments. He or she has access to the function $A = f(i)$ from which you obtained several data points in the Sunday paper. Let's see how close your two estimates were to the correct answer. Figure 3.1 includes graphs of the true function $f(i)$ as well as the interpolating polynomials $f_1(i)$ and $f_2(i)$. The upper plot includes the data points as well.

It is clear from looking at the upper plot that either interpolating function $f_1(i)$ or $f_2(i)$ will provide accurate estimates of the true monthly payment function $f(i)$ over the entire range from i = 7 % to i = 10 % and then some. The lower plot is an enlargement of each graph in the region about i = 8.25 %, the interest rate under consideration. Here we see that the quadratic interpolating polynomial $f_2(i)$ and the real function $f(i)$ are virtually indistinguishable. You may be interested in knowing that the true function A = $f(i)$ is given by

$$A = f(i) = P\frac{(i/1200)(1 + i/1200)^n}{\left[(1 + i/1200)^n - 1\right]} \tag{3.12}$$

where P is the mortgage amount, n is the loan period in months and i is the annual percent interest rate. Perhaps you are surprised at how close the graphs of the low order interpolating polynomials $f_1(i)$ and $f_2(i)$ are to the true function $f(i)$ which, based on its analytical form in Equation (3.12), does not appear to resemble a polynomial function.

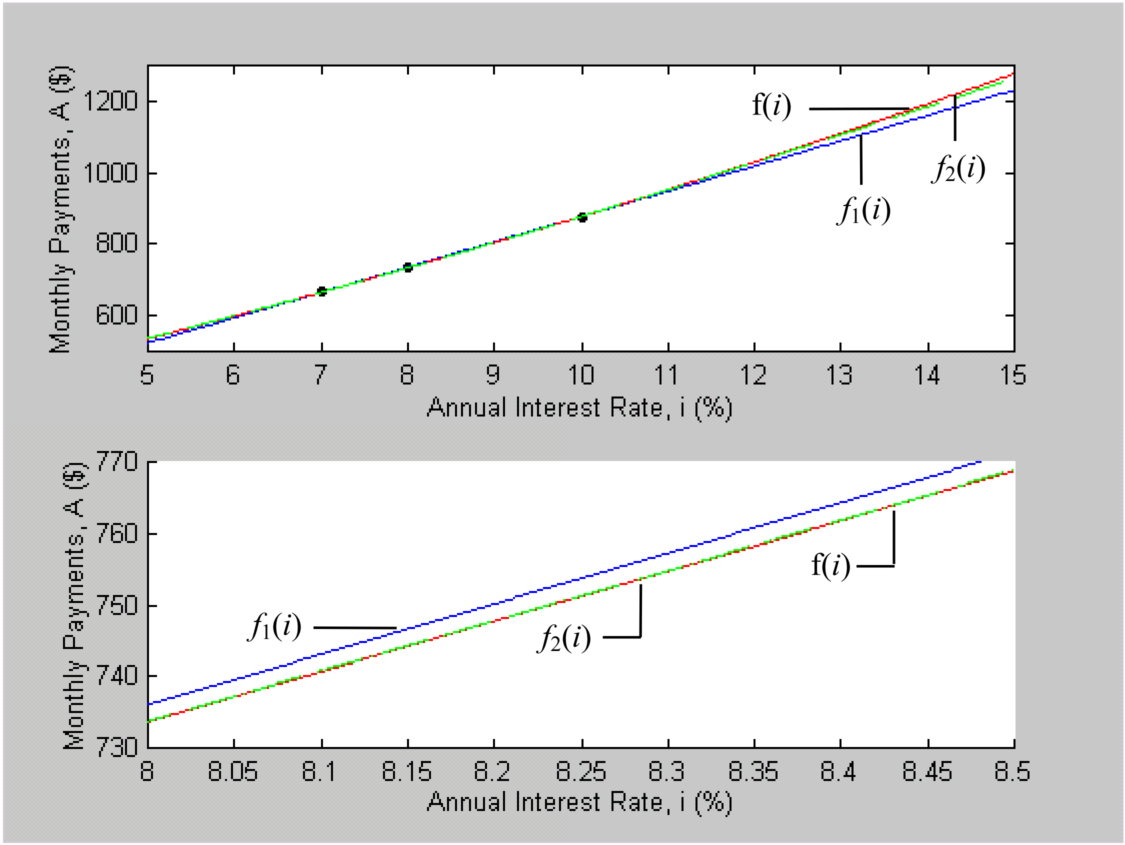

Figure 3.1 First and Second Order Interpolating Polynomials $f_1(i)$, $f_2(i)$ and the True Function $f(i)$

Table 3.3 illustrates the accuracy one can expect when using low order polynomial interpolation over the range of interest rates under consideration.

i (%)	$f(i)$	$f_1(i)$	E_T	$f_2(i)$	E_T
7.5	699.21	700.68	-1.47	699.25	-0.04
8.5	768.91	771.43	-2.52	768.86	0.05
9.5	840.85	842.19	-1.34	840.76	0.09

Table 3.3 Errors in First and Second Order Interpolation for Examples 3.1 and 3.2

We still have an additional data point that could be used to increase the order of the interpolating polynomial. The formulas for higher order coefficients become unwieldy. Fortunately, there is a framework for determining the coefficients b_i, $i = 0, 1, 2, \ldots, n$ of a general nth order interpolating polynomial as given in Equation (3.1).

The general approach is based on the evaluation of finite divided differences. Given a set of n+1 data points $[x_i, f(x_i)]$, $i = 0, 1, 2, \ldots, n$ the finite divided differences of various orders are defined in Equations (3.13) - (3.13e). Note how the divided differences are obtained recursively from two divided differences of order one less.

$$\text{0 th order} \qquad f[x_i] = f(x_i) \tag{3.13}$$

$$\text{1 st order} \qquad f[x_i, x_j] = \frac{f[x_i] - f[x_j]}{x_i - x_j}, \qquad i \neq j \tag{3.13a}$$

$$= \frac{f(x_i) - f(x_j)}{x_i - x_j} \tag{3.13b}$$

$$\text{2 nd order} \qquad f[x_i, x_j, x_k] = \frac{f[x_i, x_j] - f[x_j, x_k]}{x_i - x_k}, \qquad i \neq j \neq k \tag{3.13c}$$

$$= \frac{\dfrac{f(x_i) - f(x_j)}{x_i - x_j} - \dfrac{f(x_j) - f(x_k)}{x_j - x_k}}{x_i - x_k} \tag{3.13d}$$

$$n\text{th order} \qquad f[x_n, x_{n\text{-}1}, \ldots, x_1, x_0] = \frac{f[x_n, x_{n\text{-}1}, \ldots, x_2, x_1] - f[x_{n-1}, x_{n\text{-}2}, \ldots, x_1, x_0]}{x_n - x_0} \tag{3.13e}$$

When there are 3 data points (n = 2), the divided differences $f[x_0]$, $f[x_1, x_0]$, and $f[x_2, x_1, x_0]$ are identical to the coefficients b_0, b_1, and b_2. [see Equations (3.14) - (3.14h)]

The coefficient b_0 of the interpolating polynomial $f_2(x)$ is numerically equal to the first of the three zero order finite divided differences, i.e. the one that depends on the data point $[x_0, f(x_0)]$. The coefficient b_1 is equal to the first of the two first order finite divided differences, i.e. the one that depends on the data points $[x_0, f(x_0)]$ and $[x_1, f(x_1)]$. Finally, the coefficient b_2 is equal to the first and only second order finite divided difference, i.e. the one that requires all three data points, $[x_0, f(x_0)]$, $[x_1, f(x_1)]$ and $[x_2, f(x_2)]$.

In general, with $n + 1$ data points there are n +1 zero order divided differences, n first order divided differences, n-1 second order divided differences, etc. up to one nth order divided difference. The first computed divided difference of order "i" is equal to the coefficient b_i, $i = 0, 1, 2, \ldots, n$.

$$f[x_0] = f(x_0) \tag{3.14}$$

$$= b_0 \tag{3.14a}$$

$$f[x_1, x_0] = \frac{f[x_1] - f[x_0]}{x_1 - x_0} \tag{3.14c}$$

$$= \frac{f(x_1) - f(x_0)}{x_1 - x_0} \tag{3.14d}$$

$$= b_1 \tag{3.14e}$$

$$f[x_2, x_1, x_0] = \frac{f[x_2, x_1] - f[x_1, x_0]}{x_2 - x_0} \tag{3.14f}$$

$$= \frac{\left[\dfrac{f(x_2) - f(x_1)}{x_2 - x_1}\right] - \left[\dfrac{f(x_1) - f(x_0)}{x_1 - x_0}\right]}{x_2 - x_0} \tag{3.14g}$$

$$= b_2 \tag{3.14h}$$

Table 3.4 is a helpful aid in remembering how to compute the finite divided differences. The coefficients b_i, $i = 0, 1, 2, \ldots, n$ of the interpolating polynomial in Equation (3.1) are the first finite divided-difference entries in their respective columns. The polynomial is referred to as the Newton divided-difference interpolating polynomial. The arrows indicate which finite divided differences are used to calculate the higher order ones. Always remember, the denominator of any divided difference is the difference between its first and last argument.

		$f_1[\ ]$	$f_2[\ ]$	$f_{n-1}[\ ]$	$f_n[\ ]$
x_0	$f[x_0]$				
		$f[x_1, x_0]$			
x_1	$f[x_1]$		$f[x_2, x_1, x_0]$		
		$f[x_2, x_1]$			
x_2	$f[x_2]$		$f[x_3, x_2, x_1]$		
		$f[x_3, x_2]$			
x_3	$f[x_3]$		$f[x_4, x_3, x_2]$		
.	.	$f[x_4, x_3]$	.		
.	.	.	.	. $f[x_{n-1}, x_{n-2}, \ldots, x_2, x_0]$	
.	.	.	.	.	$f[x_n, x_{n-1}, x_{n-2} \ldots, x_1, x_0]$
.	.	.	.	. $f[x_n, x_{n-1}, \ldots, x_2, x_1]$	
x_{n-2}	$f[x_{n-2}]$	.	.		
		$f[x_{n-1}, x_{n-2}]$	.		
x_{n-1}	$f[x_{n-1}]$		$f[x_n, x_{n-1}, x_{n-2}]$		
		$f[x_n, x_{n-1}]$			
x_n	$f[x_n]$				

Table 3.4 Table of Finite Divided Differences for a Function $f(x)$ Known at Discrete Data Points

Example 3.3

Torque-speed data for an electric motor is given in the first two columns of the table below. Find the equation of the Newton divided-difference interpolating polynomial that passes through each data point and use it to estimate the torque at 1800 rpm.

Speed, ω (rpm x 1000)	Torque, T_i (ft-lb)	$f_1[\ ]$	$f_2[\ ]$	$f_3[\ ]$	$f_4[\ ]$
0.5	31 (b_0)				
		-6 (b_1)			
1.0	28		-2 (b_2)		
		-8		-6.667 (b_3)	
1.5	24		-12		6 (b_4)
		-20		5.333	
2.0	14		-4		
		-24			
2.5	2				

Table 3.5 Data Points and Finite Divided Differences for Example 3.6

The notation $f_1[\ ]$, $f_2[\ ]$, $f_3[\ ]$, and $f_4[\ ]$ represent finite divided differences of order 1 through 4, respectively. The finite divided differences and the coefficients b_0, b_1, b_2, b_3, and b_4 of the interpolating polynomial are shown in the table. The result is,

$$T(\omega) = b_0 + b_1(\omega-\omega_0) + b_2(\omega-\omega_0)(\omega-\omega_1) + b_3(\omega-\omega_0)(\omega-\omega_1)(\omega-\omega_2)$$

$$+ \ b_4(\omega-\omega_0)(\omega-\omega_1)(\omega-\omega_2)(\omega-\omega_3) \tag{3.15}$$

$$T(\omega) = 31 - 6\,(\omega-0.5) - 2\,(\omega-0.5)(\omega-1) - 6.667\,(\omega-0.5)(\omega-1)(\omega-1.5)$$

$$+ \ 6\,(\omega-0.5)(\omega-1)(\omega-1.5)(\omega-2) \tag{3.16}$$

$$T(1.8) = 31 - 6\,(1.8-0.5) - 2\,(1.8-0.5)(1.8-1) - 6.667\,(1.8-0.5)(1.8-1)(1.8-1.5)$$

$$+ \ 6\,(1.8-0.5)(1.8-1)(1.8-1.5)(1.8-2)$$

$$T(1.8) = 18.7$$

The true function $T = f(\omega)$ is not available to assess the accuracy of this estimate. Conceivably, one could derive a structure or analytical form of the function based on established engineering principles and scientific laws. A number of physical parameters, i.e. constants would likewise have to be known before the function could be used for evaluation of motor performance. Rarely is this ever attempted in situations where all that is required is a reliable estimate of how a particular device, component or system will perform. Conversely, a model of the torque motor based on scientific principles is far more useful to an engineer designing a motor to satisfy specific design criteria.

There are situations where a combination of empirical and scientific modeling are used in conjunction with each other. Imagine a situation where a theoretical model is known, however it may be of such complexity that data points are expensive to obtain. In this situation, a minimum number of data points can be obtained from solution of equations (differential and algebraic) comprising the scientific model. The data points are then used as a basis to generate an empirical model for interpolation.

Little has been said up to this point about the ordering of the data points, primarily because an nth order polynomial passing through $n+1$ data points is unique and the ordering is irrelevant. Changing the order of the data points will affect the representation of the polynomial; however the polynomial itself has not changed. The selection of which $n+1$ data points to draw from a larger sample to obtain an nth order polynomial is a different matter. In this case, some thought should be given as to which $n+1$ points produce the best interpolating polynomial. Example 3.4 illustrates these points.

Example 3.4

Consider the data in Table 3.2. Find the third order Newton divided-difference interpolating polynomial $f_3(i)$ containing the 4 data points using different orderings of the data points. Estimate the function value $f(8.25)$.

The data from Table 3.2 and the divided differences are given in Table 3.6.

k	i_k	$A_k = f(i_k)$	$f_1[\ ]$	$f_2[\ ]$	$f_3[\ ]$
0	7	665.30 (b_0)			
			68.46 (b_1)		
1	8	733.76		1.200 (b_2)	
			70.86		-0.05167 (b_3)
2	9	804.62		1.045	
			72.95		
3	10	877.57			

Table 3.6 Finite Divided Differences and Newton Interpolating Polynomial Coefficients for Data in Example 3.2

The coefficients b_0, b_1, b_2 and b_3 are read directly from the table. The resulting third order Newton divided difference interpolating polynomial is given in Equation (3.18)and the polynomial is then used for the interpolation of $f(8.25)$.

$$f_3(i) = b_0 + b_1(i - i_0) + b_2(i - i_0)(i - i_1) + b_3(i - i_0)(i - i_1)(i - i_2) \tag{3.17}$$

$$f_3(i) = 65.3 + 68.46(i - 7) + 1.2(i - 7)(i - 8) - 0.05167(i - 7)(i - 8)(i - 9) \tag{3.18}$$

$$f_3(8.25) = 65.3 + 68.46(8.25 - 7) + 1.2(8.25 - 7)(8.25 - 8) - 0.05167(8.25 - 7)(8.25 - 8)(8.25 - 9)$$

$$= 751.26$$

In Table 3.7 the data points are reordered and the process of finding the interpolating polynomial is repeated.

k	i_k	$A_k = f(i_k)$	$f_1[\ ]$	$f_2[\ ]$	$f_3[\ ]$
0	7	665.30 (b_0)			
			70.757 (b_1)		
1	10	877.57		1.148 (b_2)	
			71.905		-0.0515 (b_3)
2	8	733.76		1.045	
			70.860		
3	9	804.62			

Table 3.7 Finite Divided Differences and Newton Interpolating Polynomial Coefficients for Data in Example 3.2 with Reordering of Data Points

The new polynomial based on reordered data points and the interpolated value are

$$f_3(i) = 65.3 + 70.757(i - 7) + 1.148(i - 7)(i - 10) - 0.0515(i - 7)(i - 10)(i - 8) \tag{3.19}$$

$$f_3(8.25) = 65.3 + 70.757(8.25 - 7) + 1.148(8.25 - 7)(8.25 - 10)$$

$$-0.0515(8.25 - 7)(8.25 - 10)(8.25 - 8)$$

$$= 751.26$$

As expected, the estimated values are the same in both cases. It's a simple matter to show that the polynomial expressions in Equations (3.18) and (3.19) are in fact the same third order polynomial.

Restricting the interpolating polynomial to be 2nd order will produce as many different 2nd order polynomials as there are ways to select groups of three points from a total of four data points. In general, for a specific value of *i,* referred to as the interpolant, each interpolating polynomial $f_2(i)$ will yield a different estimate of $f(i)$. One such estimate of $f(8.25)$ has already been computed using the first three points in Table 3.2. The optimal choice of data points for determining the second order interpolating polynomial should be the set of points closest to the interpolant. This will be proven later.

Generally speaking, when Newton divided-difference polynomials $f_n(x)$ are used for interpolation to estimate values of a function $f(x)$, we should expect a difference between $f_n(x)$ and the true function value $f(x)$. In the case of linear interpolation, the error incurred, $R_1(x)$ is the difference between $f(x)$ and $f_1(x)$ and it varies over the interval $x_0 \le x \le x_1$ as shown in Figure 3.2. By definition, $R_1(x)$ satisfies

$$f(x) = f_1(x) + R_1(x) \tag{3.20}$$

Solving for $R_1(x)$,

$$R_1(x) = f(x) - f_1(x) \tag{3.21}$$

We should not expect to implement Equation (3.21) when $f(x)$ is unknown. If there was some way to estimate $f(x)$, we could obtain an approximation for the error term or remainder $R_1(x)$ as its sometimes referred to. From Figure 3.2, observe that $f_2(x)$ could be used for that purpose. That is,

$$R_1(x) \approx f_2(x) - f_1(x) \tag{3.22}$$

The right hand side of Equation (3.22) is nothing more than the second order term of the polynomial $f_2(x)$. This is easily verified by looking at Equations (3.2) and (3.7). Thus,

$$R_1(x) \approx b_2(x - x_0)(x - x_1) \tag{3.23}$$

However, in order to use Equation (3.23), an additional data point $[x_2, f(x_2)]$ is required to obtain b_2. How good is the approximation of $R_1(x)$ from Equation (3.22) or equivalently Equation (3.23)? Comparison of Equations (3.21) and (3.22) reveals that the approximation of the error term $R_1(x)$ depends on how close the function $f(x)$ and the interpolating polynomial $f_2(x)$ are. Figure 3.2 illustrates this point graphically. If the function $f(x)$ happens to be a quadratic polynomial, then $f_2(x)$ and $f(x)$ are identical and the estimate of $R_1(x)$ would be exact for any value of x.

A numerical example will help clarify the process of approximating the errors occurring from the use of Newton divided-difference interpolating polynomials.

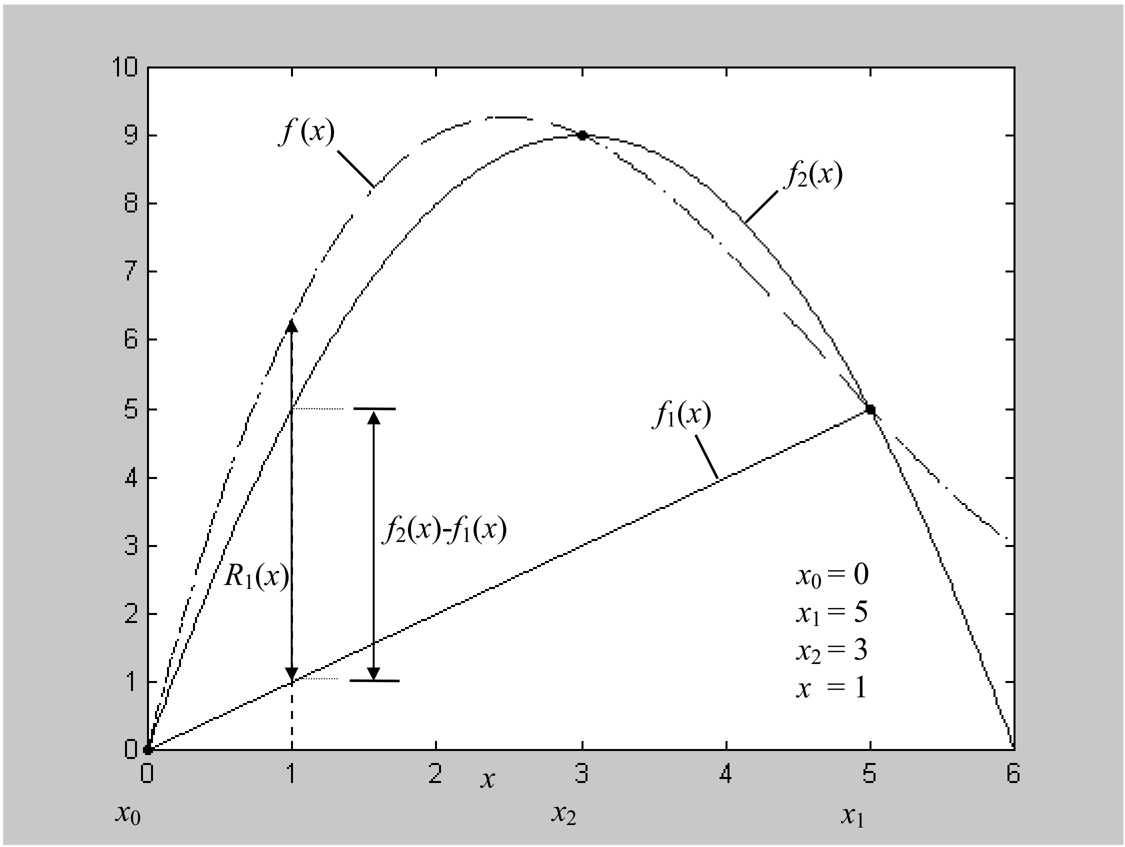

Figure 3.2 Estimating the error $R_1(x)$ in Linear Interpolation

Example 3.5

The solution to the differential equation $\frac{d}{dt}y(t) + y(t) = 1$ when $y(0) = 0$ is $y(t) = 1 - e^{-t}$, $t \geq 0$. Let T be the time it takes for the solution $y(t)$ to reach the value A, where $0 \leq A \leq 1$, i.e. $y(T) = A$. Several points (T, A) on the solution $y(t)$ are tabulated below. A low order polynomial is needed for interpolation of T for a given value of A.

T	A	T	A
0.0000	0.0	0.6931	0.5
0.1054	0.1	0.9163	0.6
0.2231	0.2	1.2040	0.7

0.3567	0.3	1.6094	0.8
0.5108	0.4	2.3026	0.9

Table 3.8 Data Points for Finding Interpolating Polynomial to Estimate *T* Given *A*

The underlined data points were selected to determine a third order Newton divided-difference polynomial which can be used for interpolation over the interval. The divided difference table is shown below.

i	A_i	$T_i = f(A_i)$	$f_1[\ ]$	$f_2[\ ]$	$f_3[\ ]$	$f_4[\ ]$
0	0.0	0.0000				
			1.1890			
1	0.3	0.3567		1.3275		
			2.1183		4.7745	
2	0.7	1.2040		5.6246		7.6676
			5.4930		8.6083	
3	0.9	2.3026		7.3462		
			4.0238			
4	0.5	0.6931				

Table 3.9 Divided Difference Table for Finding $f_3(A)$ and $f_4(A)$

From the table, the third order interpolating polynomial $f_3(A)$ is

$$f_3(A) = 1.189A + 1.3275A(A - 0.3) + 4.7745A(A - 0.3)(A - 0.7) \qquad (3.24)$$

Suppose we wish to estimate $f(0.45)$, the time required for the solution to reach 0.45, using $f_3(0.45)$. The result is

$$\begin{aligned} f_3(0.45) &= 1.189(0.45) + 1.3275(0.45)(0.45 - 0.3) + 4.7745(0.45)(0.45 - 0.3)(0.45 - 0.7) \\ &= 0.5441 \end{aligned}$$

Choosing the additional data point (0.5, 0.6931) because of its proximity to the interpolant value 0.45 allows us to find the 4th order interpolating polynomial $f_4(A)$. Table 3.9 includes the additional finite divided differences calculated from the new data point. The result is

$$\begin{aligned} f_4(A) = &\ 1.189A + 1.3275A(A - 0.3) + 4.7745A(A - 0.3)(A - 0.7) \\ &+ 7.6675A(A - 0.3)(A - 0.7)(A - 0.9) \qquad (3.25) \end{aligned}$$

and the improved estimate of $f(0.45)$ is therefore

$$
\begin{aligned}
f_4(0.45) &= f_3(0.45) + 7.6675A(A-0.3)(A-0.7)(A-0.9) \\
&= 0.5441 + 7.6675(0.45)(0.45-0.3)(0.45-0.7)(0.45-0.9) \\
&= 0.5441 + 0.0582 \\
&= 0.6023
\end{aligned}
$$

Based on reasoning analogous to that used to obtain $R_1(x)$ in Equation (3.22), the error term $R_3(0.45)$ is approximated as

$$
\begin{aligned}
R_3(0.45) &\approx f_4(0.45) - f_3(0.45) \\
&\approx 0.0582
\end{aligned}
$$

It is easily shown that the true function relating T and A is given by

$$T = f(A) = -\ln(1-A) \tag{3.26}$$

Figure 3.3 shows two graphs. Each one contains the complete set of data points from Table 3.8 and the true function $f(A)$. The top graph contains the third order polynomial $f_3(A)$ and the four data points used to find it are shown with an asterisk. The lower graph contains the fourth order polynomial $f_4(A)$. The same four data points used to find $f_3(A)$ and the one additional point used to find $f_4(A)$ are shown as asterisks.

The true error $R_3(0.45)$ is also shown in Figure 3.3. As expected $R_3(A)$ is zero when A is any of the four data point values. The exact value for $R_3(0.45)$ is obtained as the difference between $f(0.45)$ and $f_3(0.45)$. The result is

$$
\begin{aligned}
R_3(0.45) &= f(0.45) - f_3(0.45) \\
&= -\ln(1-0.45) - 0.5441 \\
&= 0.0537
\end{aligned}
$$

which should be compared to the estimated value of 0.0582 previously computed.

Keep in mind that $f_3(A)$, $f_4(A)$, and $R_3(A)$ are all sensitive to the choice of data points selected from Table 3.8. Knowing a priori that $f(0.45)$ is to be estimated would dictate a different choice of data points as the basis for determining an interpolating function to approximate $f(A)$. The 4 closest points to the interpolant value 0.45 are (0.3,0.3567), (0.4,0.5108), (0.5,0.6931) and (0.6,0.9163,).

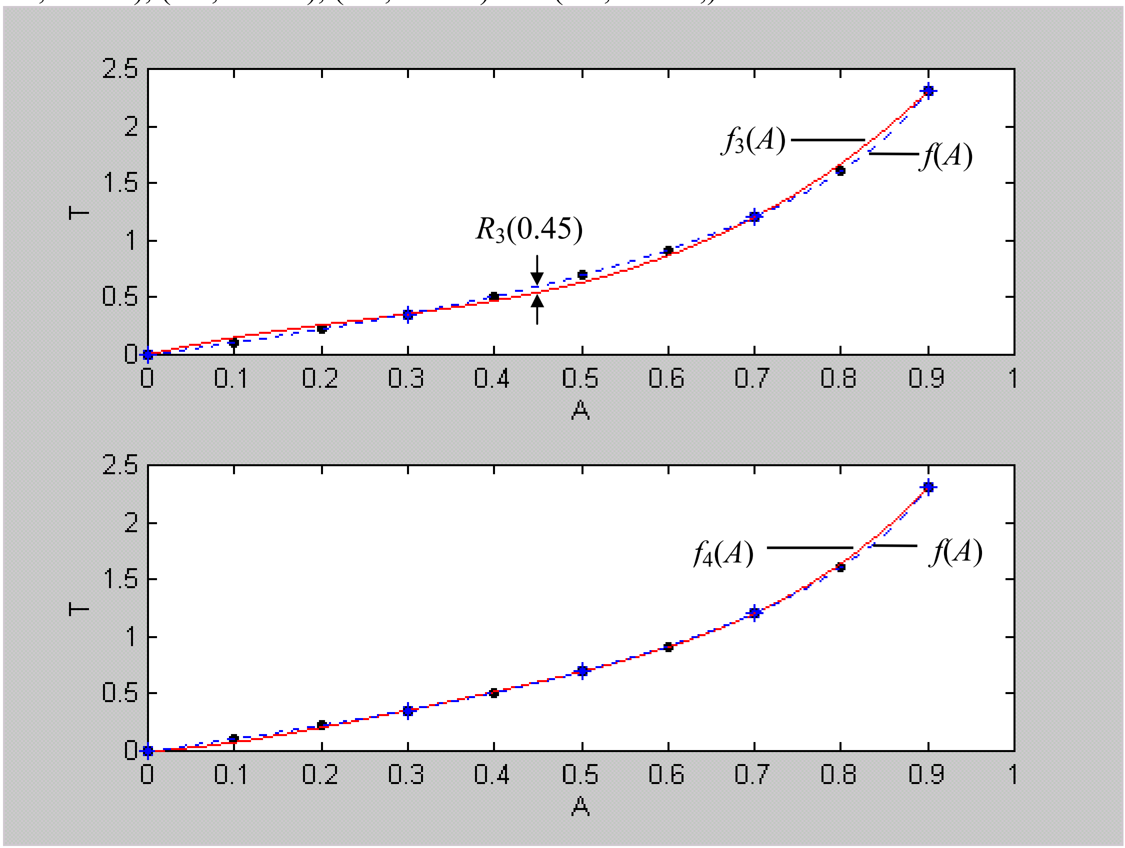

Figure 3.3 Third and Fourth Order Interpolating Polynomials and the True Function for Data in Table 3.8

At this point it is necessary to introduce a new function $f[x, x_1, x_0]$, similar to a second order divided difference, defined as

$$f[x,x_1,x_0] = \frac{f[x,x_1]-f[x_1,x_0]}{x-x_0} \tag{3.27}$$

$$= \frac{\dfrac{f(x)-f(x_1)}{x-x_1} - \dfrac{f(x_1)-f(x_0)}{x_1-x_0}}{x-x_0} \tag{3.27a}$$

However, unlike the second order finite divided differences previously encountered, its first argument x is treated as a variable. Of course, once x assumes a numerical value, $f[x, x_1, x_0]$ becomes an ordinary second order divided difference. The importance of $f[x, x_1, x_0]$ is its relationship to $R_1(x)$ which is presented here and left as an exercise for later.

$$R_1(x) = (x - x_0)(x - x_1) f[x, x_1, x_0] \qquad (3.28)$$

If $f(x)$ is a slow varying function over the interval (x_0, x_1), then $f[x, x_1, x_0]$ will likewise vary slowly over the same interval. Under these conditions, for any value of x_2 in the interval (x_0, x_1),

$$f[x, x_1, x_0] \approx f[x_2, x_1, x_0] \qquad (3.29)$$

Replacing $f[x, x_1, x_0]$ with its approximation $f[x_2, x_1, x_0]$ in Equation (3.28) gives

$$R_1(x) \approx (x - x_0)(x - x_1) f[x_2, x_1, x_0] \qquad (3.30)$$

which is identical to Equation (3.23). It was previously stated that if $f(x)$ happens to be a quadratic polynomial, then the estimate of $R_1(x)$ in Equation (3.22) would be exact. What does this imply about the function $f[x, x_1, x_0]$ under the same conditions?

We can now generalize the preceding analysis about the error in Newton divided-difference interpolating polynomials. With n+1 data points [x_i, $f(x_i)$], i = 0, 1,2, …., n from a function $f(x)$, the Newton divided-difference interpolating polynomial $f_n(x)$ that passes through each point is related to the function $f(x)$ in the following way.

$$f(x) = f_n(x) + R_n(x) \qquad (3.31)$$

The error term $R_n(x)$ accounts for the difference between the true function $f(x)$ and the interpolated value $f_n(x)$. See Equation (3.20) for the case when n=1. Equation (3.28) is a special case (n=1) of a general expression for $R_n(x)$ which follows.

$$R_n(x) = (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})(x - x_n) f[x, x_n, x_{n-1}, \ldots., x_1, x_0] \qquad (3.32)$$

The (n+1)st order finite divided-difference term $f[x, x_n, x_{n-1}, \ldots., x_1, x_0]$ in Equation (3.32) cannot be evaluated at the interpolant value of x unless the function $f(x)$ is known. Instead, an estimate of $f[x, x_n, x_{n-1}, \ldots., x_1, x_0]$ is possible if an additional data point [x_{n+1}, $f(x_{n+1})$] is available. The estimate is

$$f[x, x_n, x_{n-1}, \ldots., x_1, x_0] \approx f[x_{n+1}, x_n, x_{n-1}, \ldots., x_1, x_0] \qquad (3.33)$$

The right hand side of Equation (3.33) is of course b_{n+1}, the coefficient of the (n+1)st order term of $f_{n+1}(x)$. Using this approximation in Equation (3.32) provides a means of estimating the error term $R_n(x)$. The result is

$$R_n(x) \approx (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})(x - x_n) f[x_{n+1}, x_n, x_{n-1}, \ldots., x_1, x_0] \qquad (3.34)$$

$$R_n(x) \approx b_{n+1}(x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})(x - x_n) \qquad (3.35)$$

The estimate of $R_n(x)$ is therefore the highest order term of the (n+1)st degree divided-difference polynomial $f_{n+1}(x)$, i.e. the new term added to $f_n(x)$ made possible by the acquisition of an additional data point. In the case of linear interpolation (n=1), the estimate of the error term $R_1(x)$ from Equation (3.35) reduces to the expression given in Equation (3.23).

Equation (3.32) for $R_n(x)$ is important for several reasons. Imagine a situation where exhibits little or no variation over the interval corresponding to the n+1 data points $[x_i, f(x_i)]$, $i = 0, 1, 2, \ldots, n$. In the extreme case where $f[x, x_n, x_{n-1}, \ldots, x_1, x_0]$ is constant, Equations (3.33) and (3.34) are no longer approximations and the true value of $f(x)$ is computed from Equation (3.31). In this case, the interpolating polynomial $f_{n+1}(x)$ and $f(x)$ are identical. The following example illustrates this situation.

Example 3.6

Three data points $[x_i, f(x_i)]$, $i = 0, 1, 2$ from the function $f(x) = x^3 - 1$ are given in the table below. A second order divided-difference interpolating polynomial $f_2(x)$ is required. The finite divided differences are also given in Table 3.10.

i	x_i	$f(x_i)$	$f_1[\,]$	$f_2[\,]$
0	1	0		
			13	
1	3	26		10
			63	
2	6	215		

Table 3.10 Divided-difference table for Data in Example 3.6

The second order divided-difference interpolating polynomial is

$$f_2(x) = 13(x-1) + 10(x-1)(x-3) \tag{3.36}$$

If we use Equation (3.36) for interpolation, the error term $R_2(x)$ can be approximated after we obtain an additional data point from $f(x)$. Instead, three additional data points $[x_i, f(x_i)]$, $i = 3, 4, 5$ are calculated from the cubic function $f(x)$ and the complete listing of finite divided differences is given in Table 3.8.

We will show that the estimate of the error term $R_2(x)$ is exact regardless of which of the three additional points is used to estimate $R_2(x)$. Moreover, we will see that the third order divided difference $f[x, x_2, x_1, x_0]$ is constant.

Choosing $[x_3, f(x_3)] = (5,124)$ as the new data point, the third order divided difference $f[x_3, x_2, x_1, x_0] = 1$. From Equation (3.34), $R_2(x)$ is approximated by

$$R_2(x) \approx (x - x_0)(x - x_1)(x - x_2)\, f[x_3, x_2, x_1, x_0] \quad (3.37)$$

$$R_2(x) \approx (x - 1)(x - 3)(x - 6) \cdot 1 \quad (3.38)$$

i	x_i	$f(x_i)$	$f_1[\ ]$	$f_2[\ ]$	$f_3[\ ]$	$f_4[\ ]$	$f_5[\ ]$
0	1	0					
			13				
1	3	26		10			
			63		1		
2	6	215		14		0	
			91		1		0
3	5	124		13		0	
			39		1		
4	2	7		11			
			28				
5	4	63					

Table 3.11 Divided Difference Table - 6 Data Points From A Cubic Function $f(x) = x^3$-1

Using $f_2(x)$ for interpolation at $x = 3.5$,

$$f_2(3.5) = 13(3.5-1) + 10(3.5-1)(3.5-3)$$

$$= 45$$

and estimating the error $R_2(3.5)$ from Equation (3.38),

$$R_2(3.5) \approx (3.5-1)(3.5-3)(3.5-6) \cdot 1$$

$$\approx -3.125$$

How good is this estimate of $R_2(3.5)$? We can answer this question because the function $f(x)$ is known. The true error is

$$R_2(3.5) = f(3.5) - f_2(3.5)$$

$$= [(3.5)^3 - 1] - 45$$

$$= -3.125$$

The estimate of $R_2(3.5)$ is exact and $f(3.5)$ will be computed exactly from $f_3(3.5)$. An expected result, since $f_3(x)$ and $f(x)$ are different representations of the same third order polynomial function.

In this example, the estimate of $R_2(x)$ is exact, regardless of where the interpolation occurs. The reason is simple. We learned that $R_n(x)$ in Equation (3.34) is exact whenever the $(n+1)$st divided difference $f[x, x_n, x_{n\text{-}1}, ..., x_1, x_0]$ is constant, i.e. independent of x. This is precisely what happened in this example, namely $f[x, x_2, x_1, x_0]$ is constant See if you can show why this is true whenever $f(x)$ is a cubic polynomial.

Table 3.11 includes three additional data points despite the fact that only one extra point is needed to approximate $R_2(x)$. Observe that all three third order divided differences $f[x_3, x_2, x_1, x_0]$, $f[x_4, x_3, x_2, x_1]$, and $f[x_5\, x_4, x_3, x_2]$ are equal.

Let's reconsider this example from a different perspective.. Suppose the function $f(x)$ is in unknown and we have obtained the six data points $[x_i, f(x_i)]$, $i = 0, 1, 2, 3, 4, 5$ from it given in Table 3.11. Unaware of the true function, we set out to compute all the finite divided differences, intending to employ a Newton divided-difference interpolating polynomial of order as high as 5, if necessary. In the process of completing the table, however, we notice that the column of third order divided differences has the same numerical value. The unknown function $f(x)$ is exhibiting a property characteristic of third order polynomials, namely $f[x, x_2, x_1, x_0]$ is constant.

However, we need to be careful before concluding that the function $f(x)$ is a cubic polynomial. It is correct to say there is a third order polynomial that contains our six data points. It is given by

$$f_3(x) = 13(x-1) + 10(x-1)(x-3) + 1(x-1)(x-3)(x-6) \qquad (3.39)$$

Assuming that $f_3(x)$ and the true function $f(x)$ are one in the same is not justified simply because $f_3(x)$ passes through the six data points sampled from $f(x)$. The higher order interpolating polynomials $f_4(x)$ and $f_5(x)$ reduce to $f_3(x)$ because b_4 and b_5 are both zero.

If $f_3(x)$ and $f(x)$ are the same function then subsequent points obtained from $f(x)$ would produce the same result as we saw in Table 3.11, namely all third order finite divided differences would equal one. If there exists a function $f(x)$, not a cubic polynomial, that happens to pass through four, five or even all six data points of the function x^3 - 1, then an additional data point from $f(x)$ would result in a different value of the third order divided difference. Can you think of such a function?

Figure 3.4 is a graph of two possible functions $f(x)$ along with $f_3(x)$. Note that one of the possible $f(x)$ functions passes through four points of $f_3(x)$ and the other passes through five points of $f_3(x)$. The first function was generated in MATLAB by specifying

4 points from $f_3(x)$ and one additional point not on $f_3(x)$. The 'polyfit' function of order 4 was then used to find $f(x)$. A similar procedure using 'polyfit' of order 5 was used to find the second function $f(x)$ which passes through 5 points of $f_3(x)$ and one additional point not on $f_3(x)$.

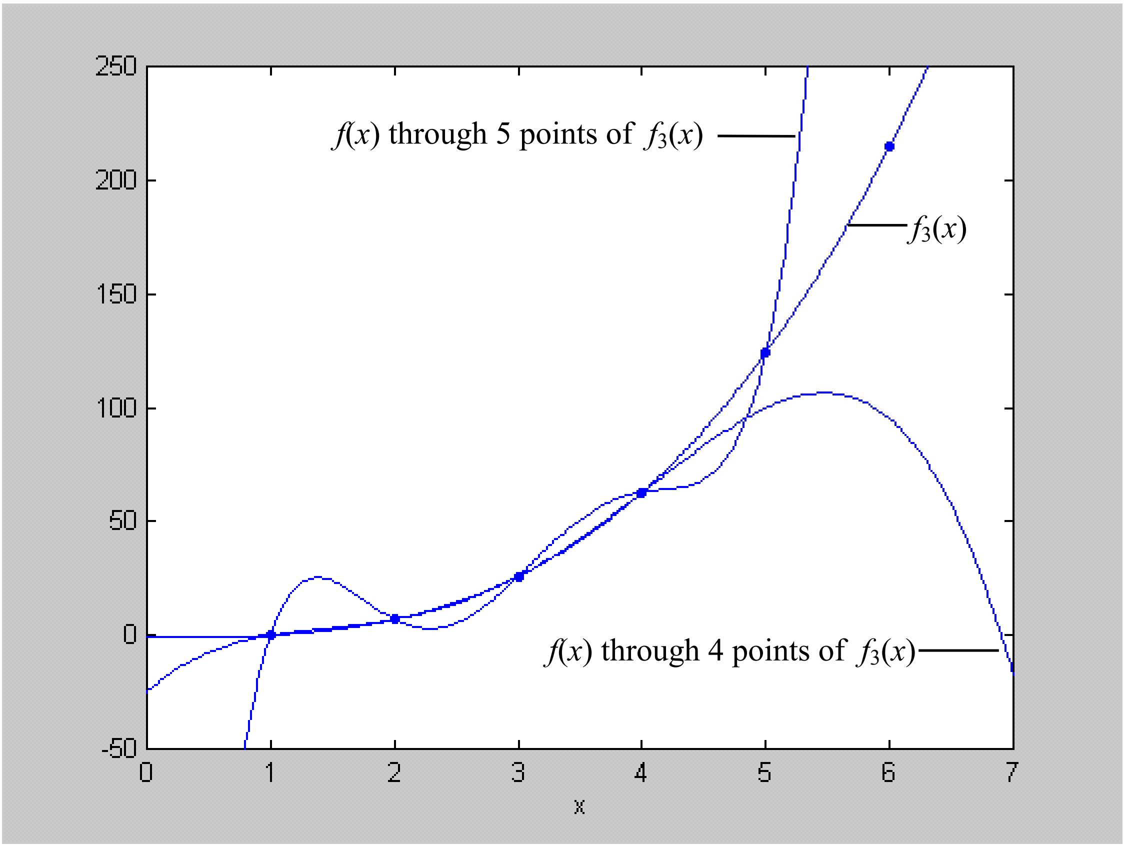

Figure 3.4 Two Non-Cubic Functions $f(x)$ with Points Common to $f_3(x)$

In summary, inferences concerning the structure of $f(x)$ cannot be made solely on the basis of the divided-difference table computed from the sampled data points.

The importance of the previous discussion is not whether a function $f(x)$ is actually a polynomial. Rarely is the underlying function a true polynomial. What's important is that the table of divided differences reveals in a quantitative way if a polynomial will be suitable for interpolation. If a particular column of divided differences is more or less constant, then an interpolating polynomial exists that accurately matches the given data points. Furthermore, this polynomial may be lower than the n^{th} order polynomial which is guaranteed to pass through the $n+1$ data points as the following example illustrates.

Example 3.7

Measurements of coffee temperature taken at 2 minute intervals are given in the table below. Find the lowest order polynomial suitable for interpolation of the function relating temperature and elapsed time.

i	Time, t_i (min)	Temp, T_i (°F)	$f_1[\ ]$	$f_2[\ ]$	$f_3[\ ]$	$f_4[\ ]$	$f_5[\ ]$	$f_6[\ ]$
0	0	212						
			-22.5					
1	2	167		1.875				
			-15.0		-0.1250			
2	4	137		1.125		0.0130		
			-10.5		-0.0208		-0.0021	
3	6	116		1.000		-0.0078		0.00028
			-6.5		-0.0833		0.0013	
4	8	103		0.500		0.0052		
			-4.5		-0.0417			
5	10	94		0.250				
			-3.5					
6	12	87						

Table 3.12 Divided-Difference Table For Data Points in Example 3.7

Results of the divided differences in Table 3.12 suggest that we should consider a third order polynomial for interpolation. To be sure, lets graph the third order polynomial based on the data points (t_i, T_i), $i = 0, 1, 2, 3$. This polynomial is

$$f_3(t) = 212 - 22.5t + 1.875t(t-2) - 0.125t(t-2)(t-4) \qquad (3.40)$$

and is shown in Figure 3.5 as $f_3^A(t)$.

Not exactly what we expected. Using $f_3(t)$ in Equation (3.40) for interpolation would result in significant errors for t in the interval from 8 to 12 minutes. Of course, if you like your coffee at 120 °F, the interpolating polynomial will do just fine.

There are several options we can explore at this point. A higher order interpolating polynomial is certainly one of them. The 4^{th}, 5^{th}, and 6^{th} order polynomials can be obtained and then graphed to assure that no anomalies or pronounced fluctuations are present as a result of using higher order polynomials.

If we are intent on using a third polynomial for interpolation, we can combine the four third order finite divided differences $f[t_3, t_2, t_1, t_0]$, $f[t_4, t_3, t_2\ t_1]$, $f[t_5, t_4, t_3, t_2]$, and $f[t_6, t_5, t_4, t_3]$ and use the average value in place of $f[t_3, t_2, t_1, t_0]$ in forming $f_3(t)$. The resulting third order interpolating polynomial, shown in Figure 3.5 as $f_3^{\,B}(t)$ is an improvement over the one given in Equation (3.40).

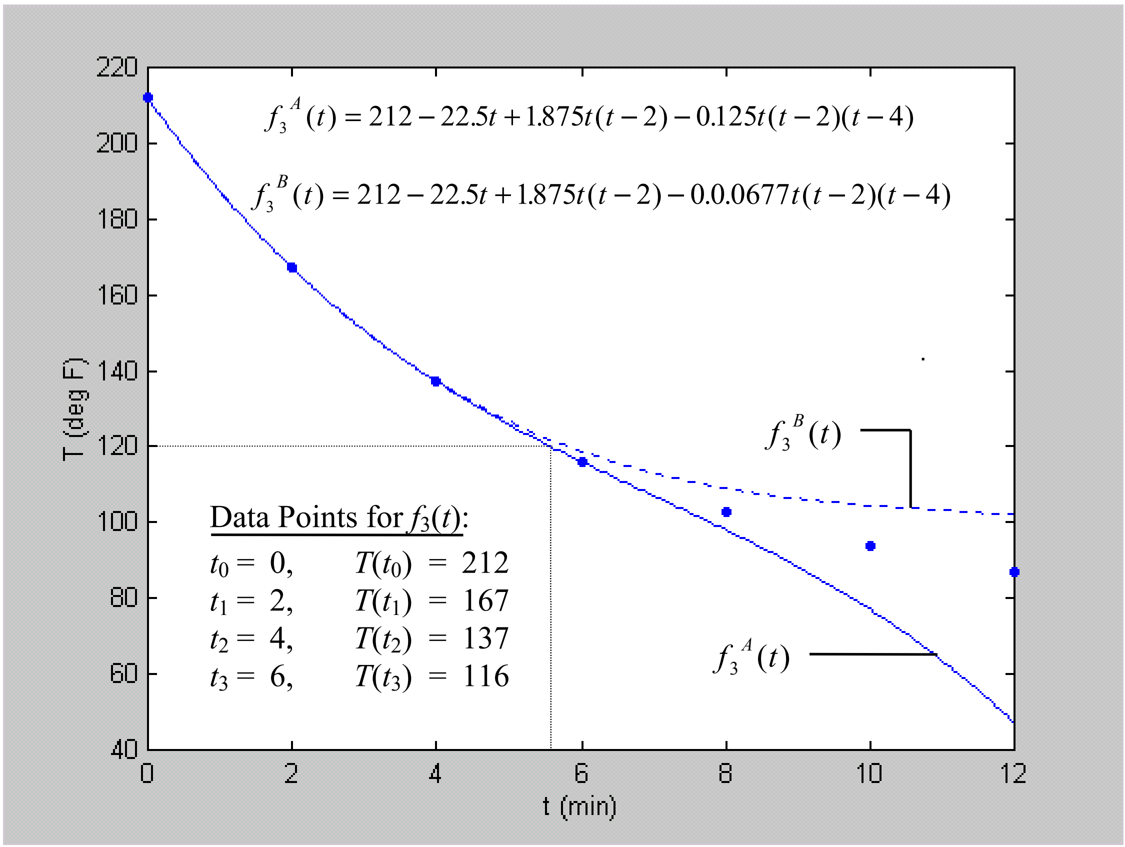

Figure 3.5 Third Order Interpolating Polynomial From First 4 Data Points in Table 3.12

There is yet another alternative which merits consideration. The decision to consider a third order polynomial for interpolation was based on our observation of the computed divided differences in Table 3.12. The third order divided differences varied by a factor of approximately 6 (-0.0208 to -0.1250) implying the third order divided-difference function $f[t, t_2, t_1, t_0]$ may not be relatively constant as initially assumed. This means the polynomial $f_3(t)$ used for interpolation will be sensitive to the choice of base points used to define it.

Perhaps we should explore the possibility of reordering the data points to see if a better fit for $f_3(t)$ can be obtained. Ordering of the data points is irrelevant only when the entire set is used to determine the unique polynomial which passes through each point. If we intend to use $f_3(t)$ for interpolation anywhere between 0 and 12 minutes, then it makes sense to include the first and last measured data point in the determination of $f_3(t)$. Table 3.13 below contains the same data points as Table 3.12, however the ordering is different.

Rational approximations were used in the MATLAB execution to minimize round-off errors in the calculations.

The new interpolating polynomial is

$$f_3(t) = 212 - 16t + \tfrac{67}{72}t(t-6) - \tfrac{1}{18}t(t-6)(t-12) \tag{3.41}$$

i	Time, t_i (min)	Temp, T_i (°F)	$f_1[\,]$	$f_2[\,]$	$f_3[\,]$	$f_4[\,]$	$f_5[\,]$	$f_6[\,]$
0	0	212						
			-16					
1	6	116		67/72				
			129/6		-1/18			
2	12	87		17/24		-5/2304		
			-25/4		-7/96		-7/4608	
3	4	137		9/16		-1/92		13/46080
			-17/2		-5/96		1/768	
4	8	103		13/12		0		
			-32/3		-5/96			
5	2	167		37/48				
			-73/8					
6	10	94						

Table 3.13 Divided-Difference Table for Example 3.7 with Reordered Data Points

and the graph in Figure 3.6 is more to our liking. Notice the smaller variation in the column of third order divided-differences in Table 3.13 compared to Table 3.12. There are other choices for the 4 base points that determine $f_3(t)$ but we shouldn't expect much if any improvement over the interpolating polynomial in Equation (3.41). After all, the true function $T(t)$ for the cooling coffee is certainly not a cubic polynomial.

We now explore the relationship between the finite divided differences and the derivatives of the interpolating polynomials. The order of a polynomial function is implied by the highest order derivative which is a constant. For example, the first derivative of a first order polynomial is constant, the second derivative of a quadratic is constant, etc. In other words, the nth derivative of an nth order polynomial is constant and the $(n+1)$ st derivative is zero. In Equations (3.42) and (3.43) below, $f(x)$ is an nth order polynomial.

$$f^{(n)}(x) = \frac{d^n}{dx^n} f(x) = \text{constant} \tag{3.42}$$

and

$$f^{(n+1)}(x) = \frac{d^{n+1}}{dx^{n+1}} f(x) = 0 \tag{3.43}$$

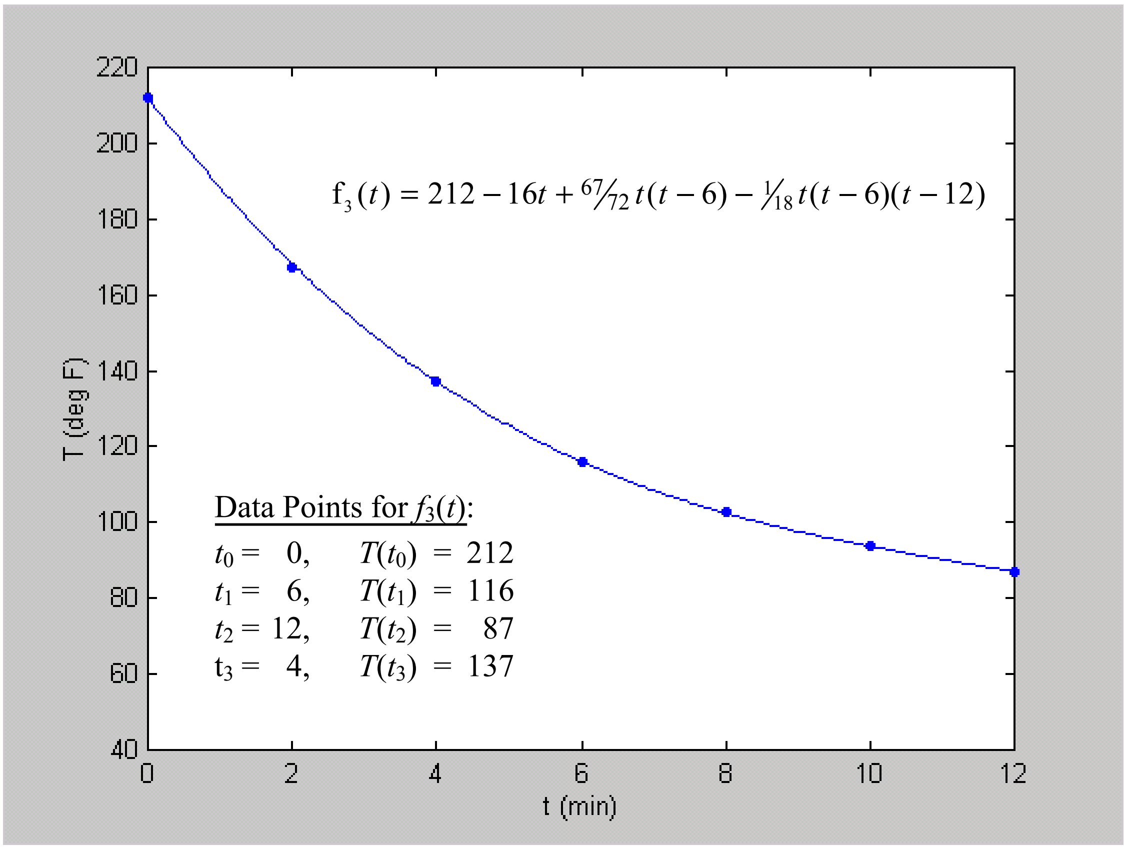

Figure 3.6 Third Order Interpolating Polynomial After Reordering of Data Points

Suppose the *n*th order polynomial $f(x)$ passes through the n+1 points $[x_i, f(x_i)]$, $i = 0,1,2,...,n$. We have already seen that the *n*th order divided-difference function is constant (See Example 3.6 and Table 3.11). That is,

$$f[x, x_{n-1}, ..., x_1, x_0] = \text{constant} \tag{3.44}$$

Substituting $R_n(x)$ from Equation (3.32) into Equation (3.31) yields,

$$f(x) = f_n(x) + (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})(x - x_n)\, f[x, x_n, x_{n-1}, ..., x_1, x_0] \tag{3.45}$$

The interpolating polynomial $f_n(x)$ will be identical to the *n*th order polynomial $f(x)$ and therefore the error term in Equation (3.45) must be zero. That is,

$$(x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})(x - x_n)\, f[x, x_n, x_{n-1}, \ldots, x_1, x_0] = 0 \qquad (3.46)$$

Since Equation (3.46) must hold for all x, we conclude that the (n+1)st divided difference function satisfies

$$f[x, x_n, x_{n-1}, \ldots, x_1, x_0] = 0 \qquad (3.47)$$

In summary, for an nth order polynomial function $f(x)$,

$$f^{(n)}(x) = \text{constant} \qquad (3.42)$$

$$f[x, x_{n-1}, \ldots, x_1, x_0] = \text{constant} \qquad (3.44)$$

$$f^{(n+1)}(x) = 0 \qquad (3.43)$$

$$f[x, x_n, x_{n-1} \ldots, x_1, x_0] = 0 \qquad (3.47)$$

The constants in Equations (3.42) and (3.44) are not equal; however the fact that both $f^{(n)}(x)$ and $f[x, x_{n-1}, \ldots, x_1, x_0]$ are constant is the important point. Equations (3.42) through (3.44) along with Equation (3.47) suggest the existence of a relationship between finite divided differences and derivatives when $f(x)$ is an nth order polynomial.

Exploring this further, consider the case where $n = 2$ and the polynomial function $f(x)$ is given by

$$f(x) = a_0 + a_1 x + a_2 x^2 \qquad (3.48)$$

Its left as an exercise to show that the second divided-difference function $f[x, x_1, x_0]$ is constant, as is $f^{(2)}(x)$. What about the first divided-difference function $f[x, x_0]$ and the first derivative $f^{(1)}(x)$? Figure 3.7 shows a quadratic function $f(x)$ and a graphical interpretation of the first divided difference function $f[x, x_0]$, keeping in mind that x is variable. For a given x, $f[x, x_0]$ will be the slope of the line segment connecting x_0 and x. There is a point ζ located between x_0 and x where the first derivative $f^{(1)}(\zeta)$ and $f[x, x_0]$ are equal. Its fairly straightforward to show that ζ is given by

$$\zeta = \frac{x_0 + x}{2} \qquad (3.49)$$

What happens if $f(x)$ in Figure 3.7 is no longer a quadratic function? As long as $f(x)$ is continuous and differentiable over (x_0, x), the Mean Value Theorem from Differential Calculus guarantees there will be at least one ζ where the first derivative $f^{(1)}(\zeta)$ and $f[x, x_0]$ are equal.

When $f(x)$ is no longer an nth order polynomial, none of the derivative functions are identically zero for all x. However, the nth order divided-difference function $f[x, x_{n\text{-}1}, \ldots, x_1, x_0]$ and the nth derivative $f^{(n)}(x)$ are still related in a similar way. The relationship is presented here without proof. It applies for any value of "n".

$$f[x, x_{n\text{-}1}, \ldots, x_1, x_0] = \frac{f^{(n)}(\zeta)}{n!} \quad \text{where } \zeta \in (x, x_{n\text{-}1}, \ldots, x_0) \tag{3.50}$$

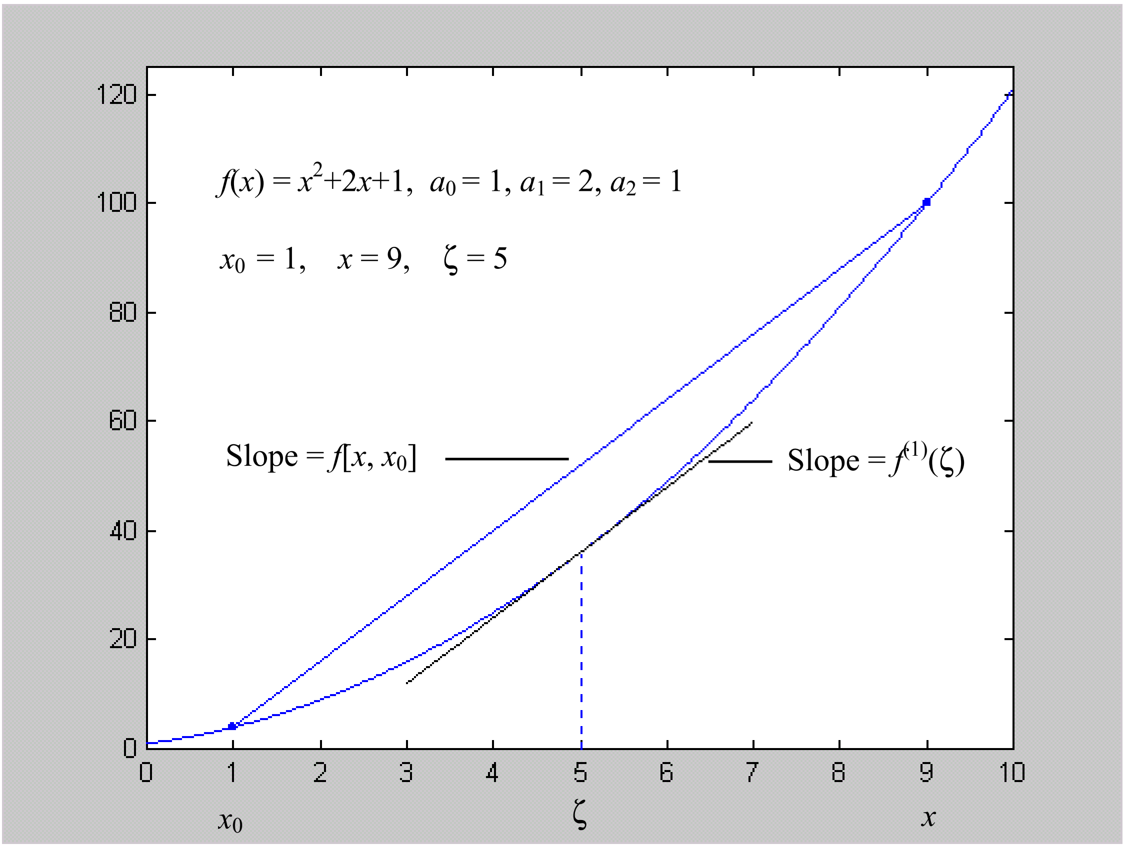

Figure 3.7 First Divided-Difference Function $f[x, x_0]$ and the First Derivative $f^{(1)}(x)$

Why is this useful? Recall that when a column of divided differences in a divided-difference table is relatively constant, we expect a polynomial of the corresponding order to be well suited for interpolation. Refer to Table 3.13 where we concluded that a third order interpolating polynomial should work well because all four 3rd order divided differences were nearly equal. From Equation (3.50) we realize that the third derivative of the underlying function $T(t)$ in Example 3.7 must also be relatively constant. In fact, we can approximate it using the average 3rd order divided difference on the left side of Equation (3.50). Doing this,

$$f^{(3)}(\zeta) \approx 3!\left[\frac{f[t_2,t_1,t_0]+f[t_3,t_2,t_1]+f[t_4,t_3,t_2]+f[t_5,t_4,t_3]}{4}\right] \tag{3.51}$$

$$\approx \quad 6\left[\frac{-\frac{1}{18}-\frac{7}{96}-\frac{5}{96}-\frac{5}{96}}{4}\right]$$

$$\approx \quad -\tfrac{67}{192}$$

Incidentally, the third derivative of the cubic interpolating polynomial in Equation (3.41), the one based on reordering of the data points to include the entire cooling period of 12 minutes, is easily computed to be -1/3.

Equation (3.50) guarantees that when the computed nth order finite divided differences are small, the nth derivative of the underlying function is likewise small over the entire interval associated with the data points. Hence, the order of the interpolating polynomial is established with confidence.

When more data points are available than needed for a given polynomial, our intuition suggested use of the data points closest to the interpolant. (See Examples 3.2 and 3.5). Our intuition has a solid analytic foundation. The error term $R_n(x)$ resulting from the use of an nth order interpolating polynomial was given in Equation (3.32) and repeated below.

$$R_n(x) = (x - x_0)(x - x_1)(x - x_2)\cdots(x - x_{n-1})(x - x_n)\, f[x, x_n, x_{n-1}, \ldots, x_1, x_0] \qquad (3.32)$$

The $(n+1)$ st divided difference $f[x, x_n, x_{n-1}, \ldots, x_1, x_0]$ is unobtainable when the function $f(x)$ is unknown. Our only alternative is to choose the $n+1$ base points $[x_i, f(x_i)]$, $i = 0,1,2,\ldots,n$ from the larger set of data points to minimize the product of the linear factors in the expression above.

In our cooling coffee problem, Example 3.7, if we require an estimate of $T(3)$, the cooled temperature after 3 minutes, the interpolating polynomial in Equation (3.40) based on the initial four data points in Table 3.11 is the logical one to use. In this case, the product $(3 - t_0)(3 - t_1)(3 - t_2)(3 - t_3)$ is minimized and hopefully the term $R_3(3)$, given below, as well.

$$R_3(3) = (3 - t_0)(3 - t_1)(3 - t_2)(3 - t_3)\ \mathrm{f}[3, t_3, t_2, t_1, t_0] \qquad (3.52)$$

One look at Figure 3.5 reminds us that the same polynomial is not the one to use for estimating $T(10)$.

We have seen how its possible to estimate the error $R_n(x)$ incurred when estimating $f(x)$ with an nth order interpolating polynomial $f_n(x)$. All that was needed was an additional point and the use of Equation (3.34). Unfortunately, the application of Equation (3.34) can underestimate $R_n(x)$ by a significant amount.

What would be nice to have is an upper bound on $R_n(x)$. Let's see if this is possible. From Equation (3.32) and Equation (3.50) with n replaced by n+1,

$$R_n(x) = (x - x_0)(x - x_1)\cdots(x - x_n)\frac{f^{(n+1)}(\zeta)}{(n+1)!} \quad \text{where } \zeta \in (x, x_n, x_{n-1}, \ldots, x_0) \tag{3.53}$$

Suppose M is an upper bound (not necessarily the maximum) on the absolute value of the (n+1)st derivative $f^{(n+1)}(x)$ over the interval containing the data points. It follows from Equation (3.53) that an upper bound on the absolute value of the error is

$$|R_n(x)| \le |(x - x_0)(x - x_1)\cdots(x - x_n)|\frac{M}{(n+1)!} \tag{3.54}$$

Equation (3.54) is of little use when $f(x)$ is available only in tabular form as sampled data points. Nonetheless, there may be times when an upper bound for the (n+1)st derivative $f^{(n+1)}(x)$ is obtainable. Of course, if the upper bound M happens to be the maximum absolute value of the (n+1)st derivative $f^{(n+1)}(x)$ over the interval, the upper bound on $|R_n(x)|$ in Equation (3.54) is even better.

Example 3.8

A uniform cable hanging under its own weight is suspended between two poles 500 ft apart as shown in Figure 3.8. The cable sags 100 ft at its lowest point. Measurements of the cable height were taken at several points and included in Table 3.14.

x (ft)	y (ft)	x (ft)	y (ft)
-250	428.0	50	331.8
-200	390.9	100	343.4
-100	343.4	200	390.9
-50	331.8	250	428.0

Table 3.14 Measured Points Along Suspended Cable

Due to symmetry, only the point x = 0, y = 328 and the last four points were considered. These five points were reordered, the divided differences computed and entered in Table 3.15.

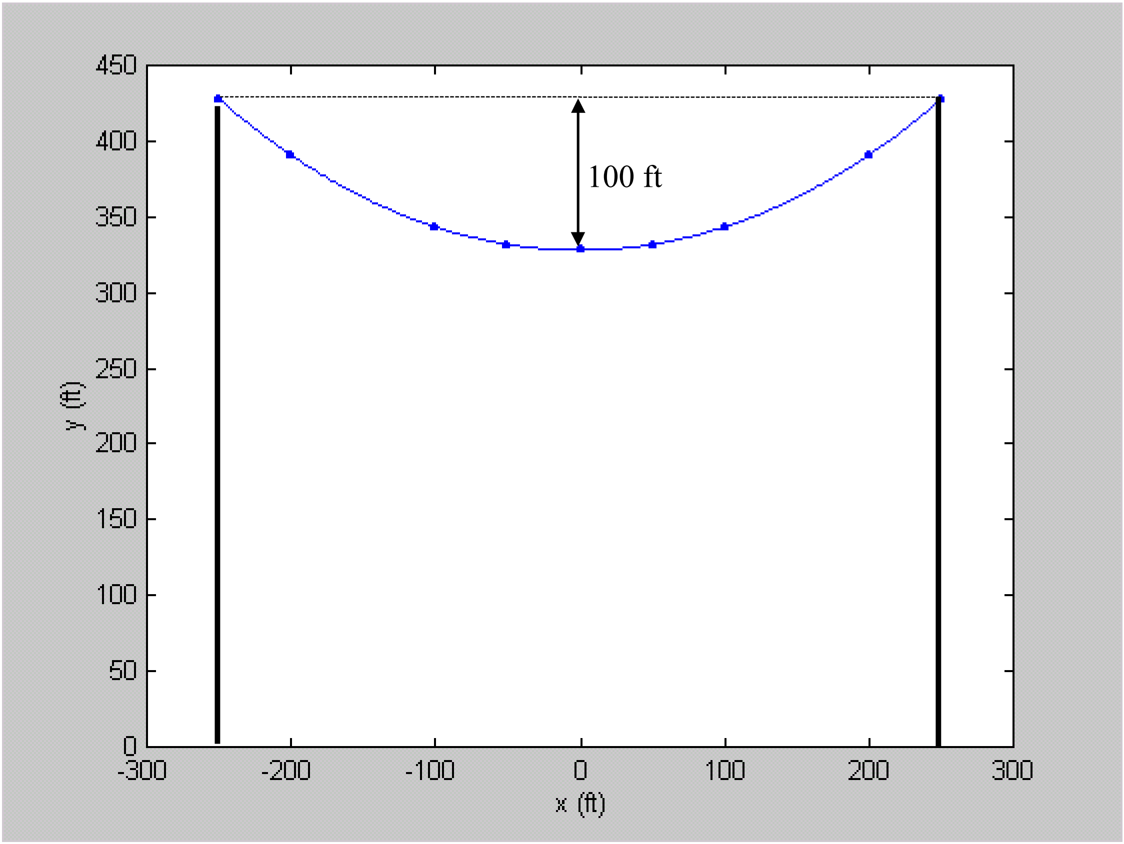

Figure 3.8 Suspended Cable Profile with Data Points Shown

i	x_i	y_i	$f_1[\,]$	$f_2[\,]$	$f_3[\,]$	$f_4[\,]$
0	0	328.0				
			0.4000			
1	250	428.0		0.00162		
			0.4810		6.0000 e-007	
2	50	331.8		0.00174		2.0000 e-009
			0.3940		8.0000 e-007	
3	200	390.9		0.00162		
			0.4750			
4	100	343.4				

Table 3.15 Divided-Difference Table For Example 3.8

Looking at the magnitudes of the higher order divided differences, it appears that the true function $y = f(x)$ can be approximated very well by a second order polynomial. From Table 3.15, $f_2(x)$ is

$$f_2(x) = 328 + 0.4x + 0.00162x(x - 250) \tag{3.55}$$

A graph of the quadratic interpolating polynomial $f_2(x)$, based on the first three points in Table 3.15, and the right half of the cable profile $f(x)$ are shown in Figure 3.9. The polynomial $f_2(x)$ passes very close to the two additional points (x_3, y_3) and (x_4, y_4) as expected.

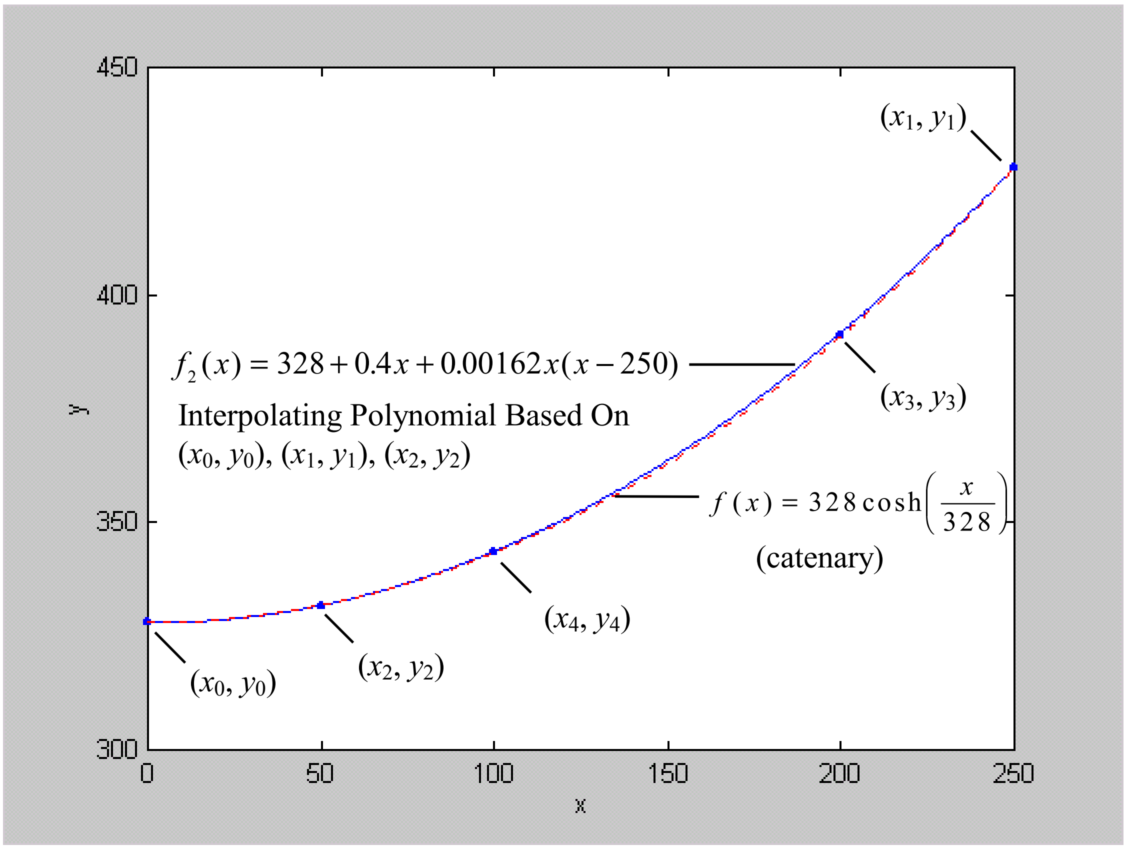

Figure 3.9 Cable Profile $f(x)$ and Quadratic Interpolating Polynomial $f_2(x)$

An upper bound on the error $R_2(x)$ when using quadratic interpolation requires some knowledge of the behavior of the third derivative, $f^{(3)}(x)$. We can approximate the third derivative from Equation (3.50) using $f[x_3, x_2, x_1, x_0]$ in Table 3.15 on the left hand side of the equation to give

$$f^{(3)}(x) \approx (3!)\, f[x_3, x_2, x_1, x_0] \quad \text{for} \quad 0 \le x \le 250 \qquad (3.56)$$

$$\approx (3!)\,(6\times10^{-7})$$

$$\approx 3.6\times10^{-6}$$

Suppose we wish to estimate the cable height 125 ft from its lowest point i.e. $f(125)$. Using Equation (3.55) gives $f_2(125) = 352.69$ ft. From Equation (3.53) with $n = 2$,

$$R_2(125) \quad = \quad (125 - x_0)(125 - x_1)(125 - x_2)\ f^{(3)}(\zeta) \tag{3.57}$$

Approximating $f^{(3)}(\varsigma)$ by the value obtained in Equation (3.56) gives

$$R_2(125) \quad \approx \quad (125 - 0)(125 - 250)(125 - 50)(3.6 \times 10^{-6})$$

$$\approx \quad 4.22$$

Ordinarily there is no way of computing an upper bound on the error $R_n(x)$ since we have no way of determining M, an upper bound on $f^{(n+1)}(x)$ in Equation (3.54). However, in this example the cable profile function $f(x)$ shown in Figure 3.9 assumes a form known as a catenary given by

$$f(x) = 328 \cosh\left(\frac{x}{328}\right) \tag{3.58}$$

Knowing the true function, we can compute M, the maximum value of the third derivative over the interval corresponding to the first three data points. The work is shown below.

$$f^{(3)}(x) \quad = \quad 328 \frac{d}{dx^3} \cosh\left(\frac{x}{328}\right) \tag{3.59}$$

$$= \quad \frac{1}{328^2} \sinh\left(\frac{x}{328}\right) \tag{3.60}$$

$$\text{M} \quad = \quad \left|\text{Max}\ f^{(3)}(x)\right| \quad \text{for} \quad 0 \le x \le 250 \tag{3.61}$$

$$= \quad \frac{1}{328^2} \sinh\left(\frac{250}{328}\right)$$

$$= \quad 7.7908 \times 10^{-6}$$

An upper bound on the interpolation error $R_2(125)$ using $M = 7.7908 \times 10^{-6}$ in Equation (3.54) is

$$\left| R_2(125) \right| \quad \le \quad \left| (125 - x_0)(125 - x_1)(125 - x_2) \right| M \tag{3.62}$$

$$\left| R_2(0.3491) \right| \quad \le \quad \left| (125 - 0)(125 - 250)(125 - 50) \right| 7.7908 \times 10^{-6}$$

$$\le \quad 9.13$$

We can compare this upper bound on the interpolation error $R_2(125)$ with the true error because the function $f(x)$ is known. Doing so,

$$E_T = f(x) - f_2(x) \tag{3.63}$$

$$= f(125) - f_2(125)$$

$$= \frac{1}{328}\cosh\left(\frac{125}{328}\right) - 352.69$$

$$= -0.58$$

The magnitude of the true error, $|E_T| = 0.58$, is of course less then the estimate of 4.22 obtained by approximating the 3rd derivative $f^{(3)}(x)$ using the third order finite divided difference $f[x_3, x_2, x_1, x_0]$ in Equation (3.56). It is also less than the upper bound 9.13, computed in Equation (3.62), which was based on the maximum 3rd derivative of the catenary function over the entire interval $0 \le x \le 250$.

Exercises

1. A tutoring service has kept records of performance on a standardized test and the number of days students attend their review classes. The performance rating Y represents the per cent improvement in the test score students attain after taking the exam a second time. X is the number of attendance days in the review class.

X, Attendance Days	x_0 1	x_1 2.5	x_2 5	x_3 6.5	x_4 9
Y, % Improvement	y_0 2	y_1 5	y_2 11	y_3 14	y_4 17

Data Points for Problem 1

a) Plot the data points.

b) Prepare a complete table of finite divided differences.

c) Estimate $f(4)$, the % improvement in one's score after 4 days of attending review classes, assuming $y = f(x)$ is the true function relating X and Y . Base your result on a third order Newton divided-difference interpolating polynomial $f_3(x)$.

d) Plot $f_3(x)$ on the same graph with the data points.

e) Estimate the error in the interpolated value $f(4)$. Use the extra data point. i.e. the one not used in Part c) to obtain your answer.

f) Repeat Part e) with $x = 0$, $y = 0$ as the additional data point. Compare results of Parts e) and f).

g) Suppose the last point in the table was $x_4 = 0$, $y_4 = 0$. Repeat Parts a) though e) with $x = 3$ instead of 4.

2. The following table contains the (x, y) coordinates of several points along the catenary shown in Figure 3.8. The variable s in the last column is the length of cable from the lowest point $x = 0$, $y = 328$ to the point (x, y).

i	x_i	y_i	s_i
0	0	328.0	0.
1	50	331.8	50.2
2	100	343.4	101.6
3	150	362.9	155.3
4	200	390.9	212.6
5	250	428.0	274.9

Data Points for Problems 2 and 3

a) Plot the points (x_i, s_i), $i = 0, 1, 2, 3, 4, 5$.

b) Find the Newton first order interpolating polynomial $f_1(x)$ through the end points (x_0, s_0) and (x_5, s_5). Plot the linear function on the same graph with the data points.

c) Use the interpolating function $f_1(x)$ to estimate the length of cable required to span a horizontal distance of 150 ft starting from the lowest point on the cable (0, 328).

d) Suppose the true function relating s and x is $s = f(x)$. Estimate the true error $R_1(150) = f(150) - f_1(150)$ by using the additional data point (200, 212.6).

e) Find the second order Newton interpolating polynomial $f_2(x)$ based on the two end points (x_0, s_0) and (x_5, s_5) and the additional data point (x_4, s_4). Plot it on the same graph with $f_1(x)$ and the data points.

f) The true function $f(x)$ is $s = c\sinh\left(\frac{x}{c}\right)$. Plot it on the same graph with $f_1(x)$, $f_2(x)$ and the data points.

g) Calculate the true error $R_1(150\ )$.

h) Find an upper bound on the true error $R_1(150\ )$.

3. Refer to the same table of data points used in Problem 2.

a) Plot the points (y_i, s_i), $i = 0, 1, 2, 3, 4, 5$.

b) Find $f_1(y)$, the Newton first order interpolating polynomial through the end points (y_0, s_0) and (y_5, s_5). Plot the linear function on the same graph with the data points.

c) Use the interpolating function $f_1(y)$ to estimate the length of cable required to span a vertical distance of 34.9 ft starting at the lowest point on the cable (0, 328).

d) Suppose the true function relating s and y is $s = f(y)$. Estimate the true error $R_1(362.9\ ) = f(362.9) - f_1(362.9)$ by using the additional data point (390.9, 212.6).

e) Find the second order Newton interpolating polynomial $f_2(x)$ based on the two end points (y_0, s_0) and (y_5, s_5) and the additional data point (y_4, s_4). Plot it on the same graph with $f_1(y)$ and the data points.

f) The true function $f(y)$ is $s = \sqrt{y^2 - 328^2}$. Plot it on the same graph with $f_1(y)$, $f_2(y)$ and the data points.

g) Calculate the true error $R_1(362.9)$.

h) Find an upper bound on the true error $R_1(362.9)$.

4. The region from sea level to approximately 36,000 ft is called the troposphere where the temperature of the standard atmosphere varies linearly with altitude.

a) Describe the true variation $T = T(h)$ if the temperature at sea level (h = 0 ft) is 518.69 °R and 447.43 °R at an altitude of 20,000 ft.

b) Air pressure and the speed of sound vary with altitude in the troposphere. The following table contains several data points reflecting the variation of these quantities with altitude.

Altitude, h (ft)	Pressure, p (psi)	Speed of Sound, v (ft/sec)
0	14.7	1116.4
5000	12.2	1097.1
10000	10.1	1077.4
20000	6.76	1036.9
30000	4.37	994.85
36000	3.31	968.75

Atmospheric Properties for Problems 4 and 5
(Source: Fundamentals of Flight, R.Shevell, Prentice-Hall, 1989)

An interpolating polynomial is required for estimating p over the entire range of altitudes. Reorder the data points if necessary, calculate the complete set of finite divided differences, and enter the values in a finite divided difference table. Determine the lowest order Newton divided-difference polynomial $f_n(h)$ suitable for interpolation of p. Do not choose the full fifth order polynomial, i.e. $n < 5$.

c) Plot the data points (h_i, p_i), i = 0,1,2,3,4,5 along with the interpolating polynomial $p = f_n(h)$ where $1 \le n \le 4$ on the same graph.

d) Use $f_n(25{,}000)$ to estimate the atmospheric pressure in the troposphere at an altitude of 25,000 ft., i.e. $f(25{,}000)$ where $p = f(h)$ is the true relationship.

e) Select one of the data point(s) from the table that was not used to determine $f_n(h)$ and use it to estimate the error in $f_n(25{,}000)$.

f) The true function relating p (in psi) and h (in ft) is

$$p = f(h) = p(0)\left[\frac{T(h)}{T(0)}\right]^{-\frac{g}{aR}}$$

where g = gravitational constant at sea level (32.17 ft/sec^2)
R= gas constant for air (1718 ft-lb/slug °R)
a = atmospheric temperature gradient in the troposphere, i.e. slope of linear function $T = T(h)$
$p(0)$ = atmospheric pressure at sea level, psi

$T(0)$ = atmospheric temperature at sea level, °R

Plot the function $p = f(h)$ for $0 \le h \le 36{,}000$ on the same graph with the data points (h_i, p_i), $i = 0,1,2,3,4,5$ and the interpolating polynomial $p = f_n(h)$, $1 \le n \le 4$.

g) Calculate the true error $f(25{,}000) - f_n(25{,}000)$.

5. Refer to the same table of data points used in Problem 4.

a) An interpolating polynomials is required for estimating v over the entire range of altitudes. Reorder the data points if necessary, calculate the complete set of finite divided differences, and enter the values in a finite divided difference table. Determine the lowest order Newton divided-difference polynomial $f_n(h)$ suitable for interpolation of v. Do not choose the full fifth order polynomial, i.e. $n < 5$.

b) Plot the data points (h_i, v_i), $i = 0,1,2,3,4,5$ along with the interpolating polynomial $v = f_n(h)$ where $1 \le n \le 4$ on the same graph.

c) Estimate the speed of sound in the troposphere at an altitude of 15,000 ft., i.e. $f(15{,}000)$ where $v = f(h)$ is the true relationship.

d) Select one of the data point(s) from the table that was not used to determine $f_n(h)$ and use it to estimate the error in $f_n(15{,}000)$.

e) The true speed of sound at 15,000 ft altitude is 1057.4 ft/sec. Calculate the true error $f(15000) - f_n 15000)$.

6. A tank is filled with water to a certain level H_0. A valve is opened and the time T required for the tank to empty completely is recorded. The following data was obtained.

Initial Height H_0 (ft)	Emptying Time T (min)
50	58.9
37	50.7
28	44.1
10	26.4
5	18.7

Data for Problem 6

a) Plot the data points with T as the dependent variable.

b) Find the third order Newton divided-difference interpolating polynomial $f_3(H_0)$ suitable for estimating the emptying time when the tank is initially filled with somewhere between 20 and 30 ft of water. Plot $f_3(H_0)$ on the same graph.

c) Estimate the time it takes for the tank to empty when there is initially 25 ft of water in it.

d) Estimate the error in your answer to Part c) by using the remaining data point.

e) Estimate the error in your answer to Part c) by using an additional data point of (0 ft, 0 min). Compare the answers to Parts d) and e) and comment on the results.

f) The actual function for calculating the height of fluid in the tank at any time is

$$H(t) = \left[H_0^{1/2} - \frac{ct}{2A} \right]^2 \quad \text{for} \quad 0 \le t \le \frac{2A}{c} H_0^{1/2}$$

where H_0 is the initial height of water in the tank, ft

A is the cross-sectional area of the tank, ft^2

c is a constant specific to the tank, $\text{ft}^3/\text{min}/\text{ft}^{1/2}$

Plot the graph of $T = f(H_0)$ for a tank with $A = 100$ ft^2 and $c = 12$ $\text{ft}^3/\text{min}/\text{ft}^{1/2}$.

g) Calculate an upper bound on the absolute value of $R_3(25) = f(25) - f_3(25)$, the true error in your answer to Part c).

h) Calculate the true error $R_3(25)$.

7. The dynamics of physical systems are often modeled by second order differential equations of the form

$$\frac{d}{dt^2} y(t) + 2\zeta\omega_n \frac{d}{dt} y(t) + \omega_n^{\,2} y(t) = K\omega_n^{\,2} u(t)$$

where $u(t)$ is the input and $y(t)$ is the response. When the input $u(t) = 1$ for $t \ge 0$, the output $y(t)$ is called the unit step response and is given by

$$y(t) = K\left[1 - \frac{1}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t} \sin(\sqrt{1-\zeta^2}\,\omega_n t + \theta) \right]$$

where $\theta = \cos^{-1}\zeta$, and $0 < \zeta < 1$

The parameters of the system ω_n, ζ and K are called the natural frequency, damping ratio and steady-state gain respectively. A typical response is shown when $\omega_n = 1$ rad/sec, $K = 1$, and $0 < \zeta < 1$. The percent overshoot P.O. refers to the amount by which the peak value M_P of the time response overshoots the final value of K. The P.O. for various values of ζ are tabulated below.

a) Graph the response $y(t)$ with ζ one of the values in the table and verify the percent overshoot of the response is in agreement with the tabulated P.O. value.
($\omega_n = 1$ rad/sec and $K = 1$)

b) Plot the data points (ζ_i, P.O.$_i$), i =0,1,…,9

c) Find the second order Newton divided-difference interpolating polynomial $f_2(\zeta)$ through the data points corresponding to ζ = 0, 0.3 and 0.9. Plot the polynomial on the same graph as the data points and comment on how well you believe it approximates the true function P.O. = $f(\zeta)$.

d) Use $f_2(\zeta)$ to estimate $f(0.2)$ and $f(0.8)$.

e) Estimate the true errors $R_2(0.2) = f(0.2) - f_2(0.2)$ and $R_2(0.8) = f(0.8) - f_2(0.8)$ by using the additional data point (ζ = 0.6, P.O. = 9.48).

f) Calculate the true errors $R_2(0.2)$ and $R_2(0.8)$.

g) Use the additional data point in Part e) to find a third order interpolating polynomial $f_3(\zeta)$. Plot it on the same graph with the data points and $f_2(\zeta)$.

h) The true function P.O. = $f(\zeta)$ is given by P.O.= $f(\zeta) = 100e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}}$. Plot $f(\zeta)$ on the same graph with the data points and the two interpolating polynomials.

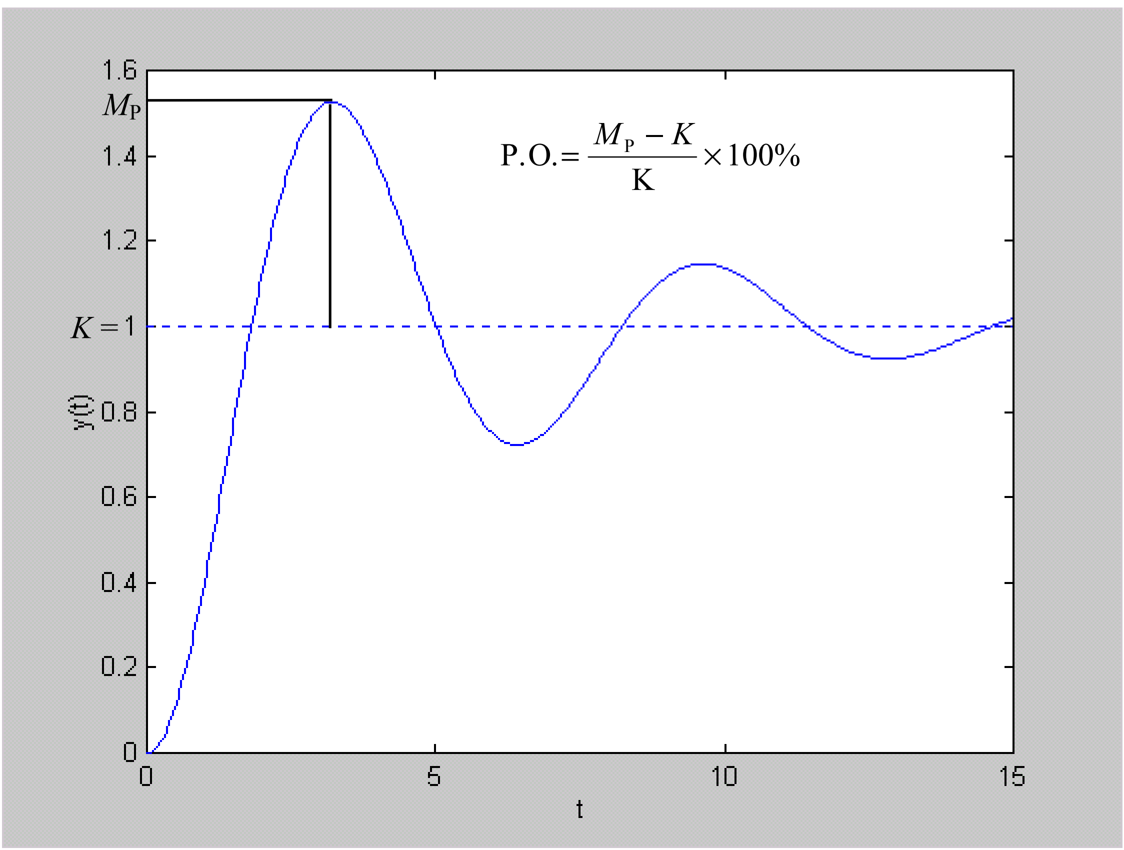

Figure for Problem 3.7

ζ	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
P.O.	100	73.0	52.7	37.2	25.4	16.3	9.48	4.60	1.52	0.15

Table for Problem 3.7

8. Two points on a circle of 1 mile radius are shown in the diagram below. The initial point O is fixed and the second point P is located at coordinates (x,y). The distance from O to P along the circle is D_1 and the straight line distance is D_2. The difference $\Delta = D_1 - D_2$ is of interest. Values of D_1 and D_2 were measured for various points along the circle. The results are summarized in tabular form.

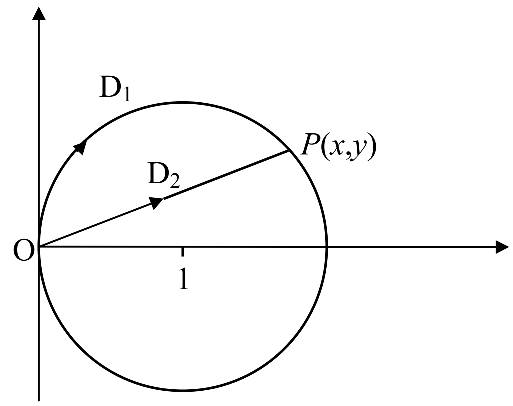

Figure for Problem 3.8

i	x_i, miles	y_i, miles	D_{1i}, miles	D_{2i}, miles	Δ_i, miles
0	0.00	0.0000	0.0000	0.0000	0.0000
1	0.25	0.6614	0.7227	0.7071	0.0156
2	0.50	0.8660	1.0472	1.0000	0.0472
3	0.75	0.9682	1.3181	1.2247	0.0934
4	1.00	1.0000	1.5708	1.4142	0.1566
5	1.25	0.9682	1.8235	1.5811	0.2423
6	1.50	0.8660	2.0944	1.7321	0.3623
7	1.75	0.6614	2.4189	1.8708	0.5480
8	2.00	0.0000	3.1416	2.0000	1.1416

Data Points for Problem 3.8

a) Plot the data points (x_i, Δ_i), $i = 0, 1, \ldots, 8$.

b) Find the lowest order Newton interpolating polynomial that fits the data points reasonably well. Plot it on the same graph.

c) Plot the full 8^{th} order Newton interpolating polynomial $f_8(x)$ on the same graph.

d) Plot the data points (y_i, Δ_i), $i = 0, 1, \ldots, 4$ on a new graph.

e) Is it possible to find a Newton interpolating polynomial from the data points (y_i, Δ_i), $i = 0, 1, \ldots, 4$ to adequately approximate the true function $\Delta = f_Y(y)$? Explain why the data points used to find the interpolating polynomial are restricted to a subset of those in the table, i.e. only y_i corresponding to $0 \le x_i \le 1$? If you can find an acceptable interpolating polynomial, plot it on the same graph. If not, explain why.

f) Show that the true function $\Delta = f_X(x)$ is given by $\Delta = \cos^{-1}(1-x) - (2x)^{1/2}$.

g) Plot the true function $\Delta = f_X(x)$ on the graph used for Parts a), b) and c). Do either of the interpolating polynomials in Part b) or c) approximate the true function reasonably well?

h) Find the equation of the true function $\Delta = f_Y(y)$. Use it to generate several more points (y_i, Δ_i).

i) Repeat Parts d) and e) using the expanded data set.

j) Discuss what steps are necessary to find an interpolating polynomial to approximate $\Delta = f_Y(y)$ for $0 \le y \le 1$ and $1 \le x \le 2$.

9. A digital filter is used to extract information from an analog signal corrupted with noise. The analog signal is sampled to generate discrete data which is then numerically processed by a conventional microprocessor or a specialized digital signal processor (DSP) optimized to perform the required calculations. The sampling rate is determined by the frequencies of the noise and signal components of the unfiltered wave form. A common type of digital filter is called an infinite impulse response (IIR) filter. The order of an IIR filter is related to its noise rejection capabilities. The amount of numerical computations increases with the order of the filter. A DSP chip with I/O, A/D conversion and additional memory was tested to determine the maximum sampling rate achievable to implement IIR filters of various orders. The results are summarized in a table.

i	IIR Filter Order n_i	Maximum Sampling Rate $(f_{max})_i$ (kHz)
0	1	1700
1	12	550
2	24	270
3	36	200
4	48	150
5	60	130

Table for Problem 3.9

a) Plot the data points $[n_i, (f_{max})_i]$, $i = 0,1,\ldots 5$.

b) Find the 5th order Newton interpolating polynomial $f_5(n)$ through the data points.

c) In a particular application where the signal and noise frequency characteristics require a sampling rate of 400 kHz, is it possible to use a 15th order IIR filter using the tested DSP hardware? Base your answer on the interpolating polynomial found in Part b).

d) Use $f_5(n)$ to estimate the maximum sampling rate for a 30th order IIR digital filter.

e) Find the 2nd order Newton interpolating polynomial $f_2(n)$ through the data points corresponding to filter orders of 12, 24 and 36. Repeat Part d) using $f_2(n)$. Compare $f_2(30)$ and $f_5(30)$ and discuss which one is likely to be closer to the true value.

f) Two additional data points become available, i.e. (20, 320) and (50,140). Which one would you choose to estimate the true error $R_5(30) = f(30) - f_5(30)$, where $f_{\max} = f(n)$ is the true function relating the filter order n and the maximum sampling rate $f_{\max}$. Explain.

g) Choose the appropriate new data point and calculate the estimate of $R_5(30)$.

10. A signalized intersection is the subject of traffic delay study. Records taken over the same time period for a duration of several weeks indicated the major flow averages 1200 vehicles per hour from both directions and the minor flow averages 600 vehicles per hour entering the intersection from both directions. The stopped time delay D refers to the time a vehicle is stopped on its way through the intersection. Measurements were taken of the stopped time delay of every car through the intersection during the time period of interest. Vehicles entering the intersection when the signal is green are assigned a zero delay since they do not stop. The average delay per vehicle D_{ave} is then calculated.

The cycle time C of the signal was fixed at 90 seconds. The portion of cycle time when the major flow receives a green light is designated G/C. The traffic engineer is trying to determine the optimum G/C split during the time period of interest. The following data resulted from the study.

i	$(G/C)_i$	$(D_{\text{ave}})_i$ sec/vehicle
0	0.55	32.6
1	0.60	32.2
2	0.50	34.6
3	0.65	33.0
4	0.45	40.8
5	0.70	35.5

Table of Data Points for Problem 10

a) Plot the data points $[(G/C)_i, (D_{ave})_i]$, $i = 0,1,\ldots,5$.

b) Calculate the complete set of finite divided differences and enter them in a table. Comment on the suitability of using Newton interpolating polynomials $f_n(G/C)$, $n = 2, 3, 4, 5$ to approximate the underlying function $(D_{ave}) = f(G/C)$.

c) Plot the Newton interpolating polynomials $f_n(G/C)$ for $n = 2, 3, 4$ and 5.

d) Estimate the optimum cycle G/C ratio and corresponding minimum average delay for each polynomial in Part c).

11. Given two data points $[x_0, f(x_0)]$ and $[x_1, f(x_1)]$ from a function $f(x)$, demonstrate the validity of Equation (3.28) which states

$$R_1(x) = (x - x_0)(x - x_1) f[x, x_1, x_0]$$