Chapter 4

Interpolation

It is often the case that an engineer needs to find the value, at some arbitrary point, of some unknown function which is defined by a set of measured data *e.g.,* steam tables in thermodynamics. This problem is defined as finding the value of some function y at some point $x \neq x_i$ given a set of data in the form $y_i = y(x_i)$, $i = 0, \ldots, N$ *i.e.,* we want to find the function $y(x)$ which best "fits" the data. *Interpolation* is the name give to the problem of finding $y(x)$ when x is *inside* the range of the given data *i.e.,* $x_0 \leq x \leq x_N$ and *extrapolation* is the name given to this problem when x is *outside* the range of the given data *i.e.,* $x < x_0$ or $x > x_N$. In this chapter we will only be concerned with interpolation.

4.1 Polynomial Fitting

One possible solution to an interpolation problem is to find the unique polynomial of order N which passes through each of the $N + 1$ data points. If the polynomial is of the form

$$y(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_N x^N \tag{4.1}$$

then the a_i coefficients of the polynomial are required to satisfy the linear system

$$\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^N \\ 1 & x_1 & x_1^2 & \cdots & x_1^N \\ 1 & x_2 & x_2^2 & \cdots & x_2^N \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & x_N^2 & \cdots & x_N^N \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \tag{4.2}$$

The matrix in Equation (4.2) is a *Vandermonde matrix* of order $N + 1$. It should be noted that

Vandermonde matrices, like Hilbert matrices, are notoriously ill-conditioned.

Once the linear system in Equation (4.2) has been solved the value of the unknown function at some point can be found from Equation (4.1). Unfortunately this polynomial interpolation scheme has a number of problems which means it is not used in practice. Some of these reasons are:

1. If the value of the interpolated function is only required at a small number of points the amount of work required to calculate these values (*i.e.*, the amount of work required to solve Equation (4.2)) is significant for problems with a large number of data points.

2. The Vandermonde matrix is ill-conditioned for a large number of data points and hence the polynomial coefficients are hard to calculate accurately.

3. For a large number of data points the resultant high-order polynomial can oscillate widely between the data points or outside the data range. This is shown in Figure 4.1. This is a particular problem if the data points contain some error (as most measured data does!).

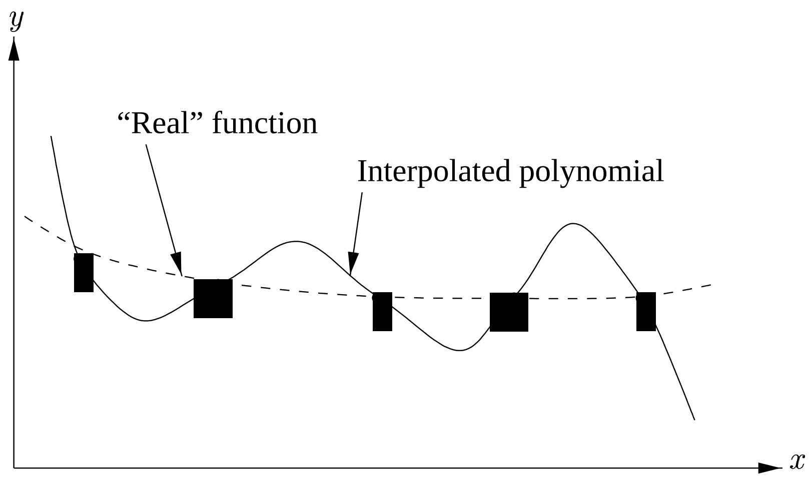

FIGURE 4.1: Oscillations between the data points in a high-order polynomial interpolation.

4.2 Lagrangian Interpolation

To avoid the problems of polynomial interpolation with a large number of data points Lagrangian interpolation is used. Lagrangian interpolation breaks the range of the interpolation interval into a number of sub-intervals. A (low-order) polynomial is then used to interpolate the data within this sub-interval.

4.2.1 Linear Interpolation

Consider allowing a linear variation of y with x within an interpolation interval $I_j = [x_j, x_{j+1}]$. This is shown in Figure 4.2.

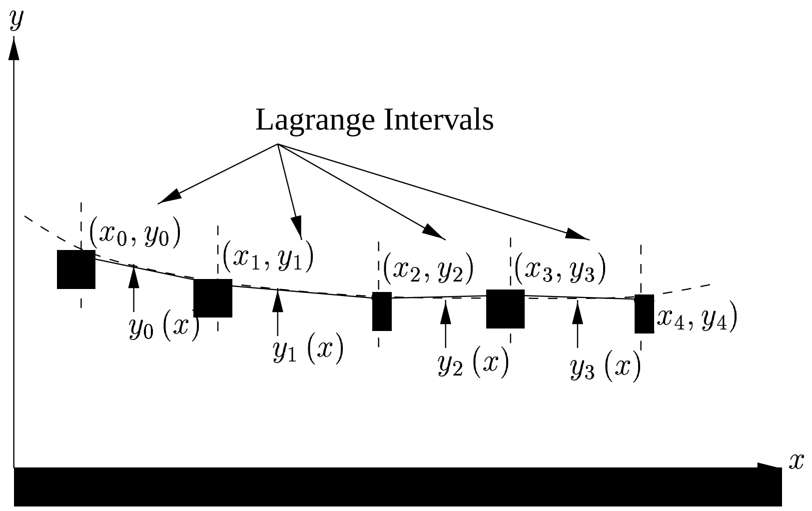

FIGURE 4.2: Linear Lagrange interpolation.

From the geometry of Figure 4.2 the interpolation formula within this sub-interval is

$$\begin{aligned} y_j(x) &= y_j + \frac{\Delta y}{\Delta x}(x - x_j) \\ &= y_j + \frac{y_{j+1} - y_j}{x_{j+1} - x_j}(x - x_j) \end{aligned} \tag{4.3}$$

Note that this linear interpolation formula can be rearranged to give

$$y_j(x) = \frac{x - x_{j+1}}{x_j - x_{j+1}} y_j + \frac{x - x_j}{x_{j+1} - x_j} y_{j+1} \tag{4.4}$$

that is the interpolation formula consists of a sum of one polynomial per data point within the sub-interval. Each data point polynomial hence serves to *weight* the overall interpolating polynomial in terms of the data function values at the individual data points.

4.2.2 Quadratic Interpolation

Consider now allowing a quadratic variation inside a sub-interval containing three data points. To do this we extend the idea of having a weighted sum of polynomials. In this case we have three polynomials (one for each of the data points) *i.e.*,

$$\begin{aligned} y_j(x) &= L_1(x)\,y_1 + L_2(x)\,y_2 + L_3(x)\,y_3 \\ &= \sum_{i=1,3} L_i(x)\,y_i \end{aligned} \tag{4.5}$$

Note that with three (or more) data points making up a sub-interval we are at some liberty as to which three data points are used to interpolate a particular value of x.

To ensure that the interpolation function passes through the data points the $L_i(x)$ polynomial needs to be equal to 1 at $x = x_i$ and be equal to 0 at all the other data points *i.e.*, $x = x_j$, $j \neq i$. This is shown graphically in Figure 4.3.

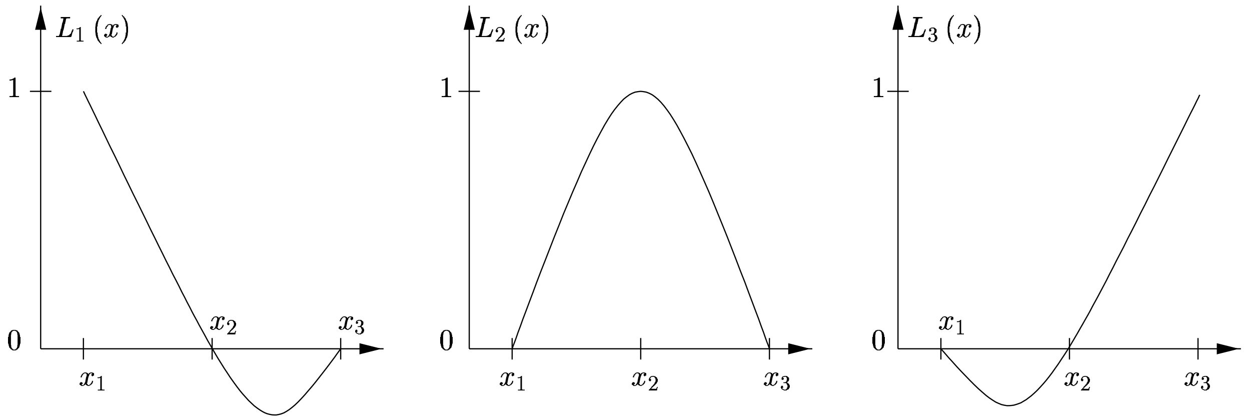

FIGURE 4.3: The three quadratic Lagrange polynomials. Note that $L_i(x)$ is 1 at x_i and 0 at the other two points.

From Figure 4.3 these quadratic polynomials are hence

$$\begin{aligned} L_1(x) &= \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} \\ L_2(x) &= \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} \\ L_3(x) &= \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} \end{aligned} \tag{4.6}$$

As an example consider a quadratic Lagrange interpolation between the points: $(1, 8)$, $(2, 1)$ and $(4, 5)$.

The interpolation formula for this example is hence

$$\begin{aligned}
y(x) &= \frac{(x-2)(x-4)}{(1-2)(1-4)}.8 + \frac{(x-1)(x-4)}{(2-1)(2-4)}.1 + \frac{(x-1)(x-2)}{(4-1)(4-2)}.5 \\
&= \frac{8}{3}(x-2)(x-4) - \frac{1}{2}(x-1)(x-4) + \frac{5}{6}(x-1)(x-2) \\
\{ &= 3x^2 - 16x + 21\}
\end{aligned} \tag{4.7}$$

To check this formula we can evaluate it at one of the data points *i.e.,*

$$\begin{aligned}
y(1) &= \frac{8}{3}(1-2)(1-4) - \frac{1}{2}(1-1)(1-4) + \frac{5}{6}(1-1)(1-2) \\
&= 8 + 0 + 0 = 8
\end{aligned} \tag{4.8}$$

To use this formula consider finding the value of $y(3)$ *i.e.,*

$$\begin{aligned}
y(3) &= \frac{8}{3}(3-2)(3-4) - \frac{1}{2}(3-1)(3-4) + \frac{5}{6}(3-1)(3-2) \\
&= \frac{8}{3}. - 1 - \frac{1}{2}.2 + \frac{5}{6}.2 \\
&= 0
\end{aligned} \tag{4.9}$$

4.2.3 General Form

In general a m^{th} order Lagrangian interpolation involves $m+1$ data points and hence the interpolation formula

$$y(x) = \sum_{i=1}^{m+1} L_i(x)\, y_i \tag{4.10}$$

where the Lagrange polynomials, $L_i(x)$, are given by

$$L_i(x) = \prod_{j=1,m+1;j\neq i} \frac{x-x_j}{x_i-x_j} = \begin{cases} 1 & \text{if } x = x_i \\ 0 & \text{if } x = x_j;\, j \neq i \end{cases} \tag{4.11}$$

For example consider $m = 3$ (cubic interpolation). In this case the interpolation formula is

$$y(x) = L_1(x)\, y_1 + L_2(x)\, y_2 + L_3(x)\, y_3 + L_4(x)\, y_4 \tag{4.12}$$

and the Lagrangian polynomials are

$$\begin{aligned} L_1(x) &= \frac{x-x_2}{x_1-x_2}\frac{x-x_3}{x_1-x_3}\frac{x-x_4}{x_1-x_4} \\ L_2(x) &= \frac{x-x_1}{x_2-x_1}\frac{x-x_3}{x_2-x_3}\frac{x-x_4}{x_2-x_4} \\ L_3(x) &= \frac{x-x_1}{x_3-x_1}\frac{x-x_2}{x_3-x_1}\frac{x-x_4}{x_3-x_4} \\ L_4(x) &= \frac{x-x_1}{x_4-x_1}\frac{x-x_2}{x_2-x_4}\frac{x-x_3}{x_4-x_3} \end{aligned} \tag{4.13}$$

There is one problem with Lagrangian interpolation in that the derivative of the interpolated function is not continuous across Lagrange sub-interval boundaries. This is shown in Figure 4.4.

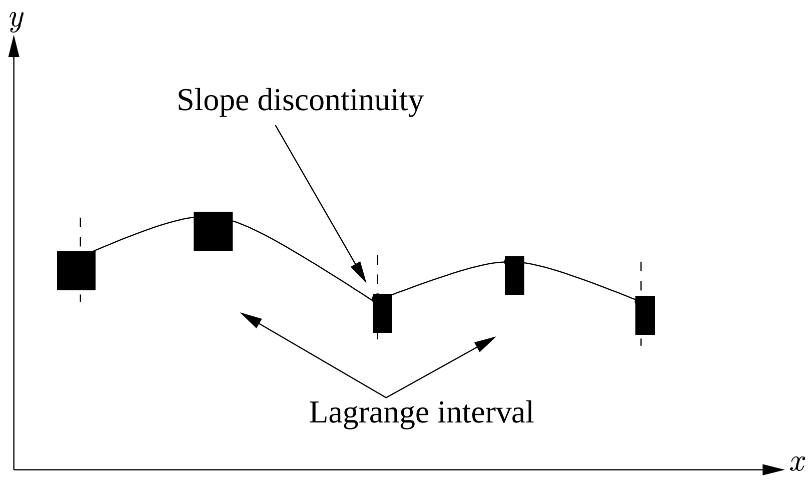

FIGURE 4.4: Discontinuity in the slope of the interpolating function across a Lagrange sub-interval boundary.

4.3 Cubic Splines

In order to overcome the discontinuity in the slope of the interpolating function with Lagrangian interpolation *cubic splines* are used. Spline based interpolation breaks the range of the interpolation into N sub-intervals of the form $I_j = [x_j, x_{j+1}]$. Cubic polynomials are then used to interpolate the unknown function within each sub-interval. These polynomials are chosen so that the overall interpolating function is continuous *and* that the overall interpolating function has continuous first and second derivatives.

To calculate the cubic spline interpolation polynomial consider a cubic polynomial $p_j(x)$ on the sub-interval I_j of the form

$$p_j(x) = a_0 + a_1(x - x_j) + a_2(x - x_j)^2 + a_3(x - x_j)^3 \tag{4.14}$$

Now the cubic spline polynomial must agree with the data values at the end points of this interval *i.e.*,

$$p_j(x_j) = y_j \quad \text{and} \quad p_j(x_{j+1}) = y_{j+1} \tag{4.15}$$

and, as the spline has a defined first derivative, we can also write

$$p_j'(x_j) = k_j \quad \text{and} \quad p_j'(x_{j+1}) = k_{j+1} \tag{4.16}$$

where k_j is the value of the first derivative of the unknown function at the point x_j. There are two cases to consider. The first case is when k_0 and k_N are given values. The second case is where k_0 and k_N are unknown. For the first case the values of the derivatives of the unknown function at the $N-1$ internal points ($k_1, \ldots, k_{N-1}$) hence need to be determined.

To determine the unknown k_j values we substitute Equations (4.15) and (4.16) into Equation (4.14) to give the system of equations

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{c_j^2} & \frac{1}{c_j^3} \\ 0 & 0 & \frac{2}{c_j} & \frac{3}{c_j^2} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} y_j \\ k_j \\ y_{j+1} - y_j - \frac{k_j}{c_j} \\ k_{j+1} - k_j \end{bmatrix} \tag{4.17}$$

where $c_j = \dfrac{1}{x_{j+1} - x_j}$.

The solution to Equation (4.17) is

$$\begin{aligned} a_0 &= y_j \\ a_1 &= k_j \\ a_2 &= 3c_j^2(y_{j+1} - y_j) - c_j(k_{j+1} + 2k_j) \\ a_3 &= 2c_j^3(y_j - y_{j+1}) + c_j^2(k_{j+1} - k_j) \end{aligned} \tag{4.18}$$

Thus from Equation (4.18) the cubic spline is determined once the k_j's have been determined. To find these values consider differentiating Equation (4.14) twice and evaluating it at $x = x_j$ to give

$$p_j''(x_j) = -6c_j^2 y_j + 6c_j^2 y_{j+1} - 4c_j k_j - 2c_j k_{j+1} \tag{4.19}$$

and evaluating it again at $x = x_{j+1}$ to give

$$p_j''(x_{j+1}) = 6c_j^2 y_j - 6c_j^2 y_{j+1} + 2c_j k_j + 4c_j k_{j+1} \tag{4.20}$$

Now, by definition, the cubic spline has a continuous second derivative at each of the $j = 1, \ldots, N-1$ internal points. This gives the conditions

$$p_{j-1}''(x_j) = p_j''(x_j) \qquad j = 1, \ldots, N-1 \tag{4.21}$$

Hence equating Equation (4.19) and Equation (4.20) with j replaced by $j-1$ yields $N-1$ equations of the form

$$c_{j-1}k_{j-1} + 2(c_{j-1} - c_j) k_j + c_j k_{j+1} = 3\left[c_{j-1}^2 (y_j - y_{j-1}) + c_j^2 (y_{j+1} - y_j)\right] \tag{4.22}$$

for $j = 1, \ldots, N-1$. Solution of Equation (4.22) yields the unknown values of $k_1, \ldots, k_{N-1}$.

For the special case of equally spaced data points *i.e.*, $x_0, x_1 = x_0 + h, \ldots, x_n = x_0 + nh$ we have $c_j = \dfrac{1}{x_{j+1} - x_j} = \dfrac{1}{h}$ and so Equation (4.22) becomes

$$k_{j-1} + 4k_j + k_{j+1} = \frac{3}{h}(y_{j+1} - y_{j-1}) \tag{4.23}$$

for $j = 1, \ldots, N-1$.

Hence once the unknown values of the k_j's are known Equation (4.18) can be used to defined the spline coefficients which can then be used in Equation (4.14) to define the interpolating cubic spline in the sub-interval I_j.

For the second case of a cubic spline in which k_0 and k_N are unknown we can still define a cubic spline by fixing the value of the second derivative of the interpolating polynomial to zero at the points x_0 and x_N *i.e.*, $p_0''(x_0) = 0$ and $p_{N-1}''(x_N) = 0$. Such splines are known as *natural cubic splines*. For natural cubic splines the unknown k_0 and k_N values can be found by adding two extra equations to Equation (4.22). These equations are found by setting Equation (4.19) to zero for $j = 0$ and Equation (4.20) to zero for $j = N-1$ to give

$$4c_0 k_0 + 2c_0 k_1 = -6c_0^2 y_0 + 6c_0^2 y_1 \tag{4.24}$$

and

$$2c_{N-1}k_{N-1} + 4c_{N-1}k_N = 6c_{N-1}^2 y_{N-1} + 6c_0^2 y_1 \tag{4.25}$$