Int J Thermophys (2017) 38:39
DOI 10.1007/s10765-016-2174-6

TEMPMEKO 2016

Propagation of Uncertainty and Comparison of Interpolation Schemes

D. R. White[1]

Received: 24 July 2016 / Accepted: 16 December 2016 / Published online: 5 January 2017

Abstract The numerical information in the calibration reports for indicating instruments is typically sparse, often comprising a simple table of corrections or indicated values against a small number of reference values. Users are left to interpolate between the tabulated values using one of several well-known interpolation algorithms, including straight-line, spline, Lagrange, and least-squares interpolation. Although these algorithms are well known, there has apparently been no comparison of their performance in respect of uncertainty propagation. This paper provides an overview of the advantages and disadvantages of the most common interpolation algorithms with respect to uncertainty propagation, immunity to interpolation error, and sensitivity to data spacing. Secondly, the paper illustrates an unconventional method for the uncertainty analysis. The method exploits the linear dependence of the interpolations on measurements of the interpolated quantity, and is easily applied to any linear functional interpolation. In many respects, the best all-round interpolation scheme is a polynomial fitted by least-squares methods, which has a low propagated uncertainty, continuity to the chosen order of the fitted polynomial, and a good immunity to large gaps in the data.

Keywords Interpolation · Lagrange interpolation · Least-squares · Linear interpolation · Measurement uncertainty · Spline

Selected Papers of the 13th International Symposium on Temperature, Humidity, Moisture and Thermal Measurements in Industry and Science.

✉ D. R. White
rod.white@callaghaninnovation.govt.nz

[1] Measurement Standards Laboratory, Lower Hutt, New Zealand

1 Introduction

The calibration of indicating instruments typically involves repeated measurements of a quantity with a known reference value using the instrument under test. Typically, the high cost of calibrations dictates that the number and range of the measurements is little more than the minimum necessary to confidently characterise the major features of the instrument's performance. In the subsequent use of the calibrated instrument, it is usually necessary to interpolate between the calibration points. If the users of the instrument are fortunate, the calibration laboratory will recommend an interpolating equation, ideally based on expert understanding of the measurement. But, in any case, either the laboratory or the user must select an interpolating equation.

There are literally dozens of recognised interpolating equations that could be chosen. Apart from the well-known straight-line and polynomial interpolations, there are many different splines, trigonometric, and rational-function interpolating equations [1,2]. From a thermometry perspective, it is important that the interpolating equation minimises interpolation error by closely approximating the combined behaviour of the temperature sensor and the indicating instrument. For this reason, we routinely use interpolating equations based on the known quadratic behaviour of platinum resistance thermometers [3], the Arrhenius law behaviour of thermistors [4], and the Planckian behaviour of radiation thermometers [5,6].

For instruments that read the temperature directly, the certificate users' problems are simpler: the nonlinear response of most sensors is well compensated in the instrument software so that the most likely errors manifest in the instrument's indications are an offset, scale error, and perhaps some minor residual nonlinearity. The choice of interpolating equation then largely rests on properties such as continuity, ease of computation, and the ability to accommodate the residual nonlinearity. Although such properties are of keen interest to mathematicians [7–10], there remains the issue of propagation of uncertainty, which is of keen interest to metrologists. Remarkably, although the various interpolation techniques are well understood from a mathematical perspective, there has apparently been no comparison of the performance of interpolating equations in respect of uncertainty propagation.

The aim of this paper is twofold. Firstly, it illustrates a method for determining the propagation of uncertainty that is readily applied to most interpolating equations. Secondly, it presents a comparison of the performance of the most commonly used interpolating equations. The first of the following sections summarises the propagation of uncertainty for Lagrange (polynomial) interpolation, based on [11], emphasising the aspects applicable to the analysis of other interpolating equations. Section 3 then applies the technique to straight-line interpolation, the natural cubic spline, and least-squares fits. The final section summarises the results and draws conclusions.

2 Lagrange Interpolation

There is a unique first-order equation that passes through two points, a unique second-order equation that passes through three points, etc. Lagrange found a very simple representation of such equations by constructing them in terms of the polynomials that now bear his name. For example, the unique quadratic equation that passes through the three points (x_1, y_1), (x_2, y_2), and (x_3, y_3) is

$$\hat{y}(x) = y_1 L_1(x) + y_2 L_2(x) + y_3 L_3(x), \tag{1}$$

where the caret on $\hat{y}(x)$ indicates that the interpolated value is an estimate that may be different from the true value $y(x)$, and the three second-order Lagrange polynomials are

$$L_1(x) = \frac{(x - x_2)(x - x_3)}{(x_1 - x_2)(x_1 - x_3)}, \quad L_2(x) = \frac{(x - x_1)(x - x_3)}{(x_2 - x_1)(x_2 - x_3)},$$
$$L_3(x) = \frac{(x - x_1)(x - x_2)}{(x_3 - x_1)(x_3 - x_2)}.$$

In general, a Lagrange interpolation is expressed as

$$\hat{y}(x) = \sum_{i=1}^{N} y_i L_i(x), \tag{2}$$

where the Nth-order Lagrange polynomials are

$$L_i(x) = \prod_{j=1, j \neq i}^{N} \left(\frac{x - x_j}{x_i - x_j} \right). \tag{3}$$

The key to understanding Lagrange's formulation is to recognise: (i) the constants multiplying each of the Lagrange polynomials in Eq. 2 are the y_i values, (ii) the polynomials are functions of x and the x_i only, (iii) each Lagrange polynomial is constructed with zeros at $N - 1$ of the data points, and (iv) the denominator of each polynomial is constructed to ensure the polynomial takes the value 1.0 at the remaining data point. The last two features ensure the polynomials have a type of orthonormality property:

$$L_i(x_j) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} . \tag{4}$$

Once the pattern of construction of the Lagrange polynomials has been recognised, it is possible to write down (e.g., directly into a spreadsheet) any low-order Langrage interpolating equation. For higher-order equations (many data points), writing the equations in this form is clumsy and error prone, and it is better to use Neville's or Aitken's algorithm [1] to construct the interpolation.

The interpolation might also be implemented as a subroutine in a software application. For example, the subroutine call might be $Poly[x : (x_1, y_1), (x_2, y_2), (x_3, y_3)]$. In that case, the Lagrange form of the interpolation can be written

$$\begin{aligned}
&Poly\,[x : (x_1, y_1)\,, (x_2, y_2)\,, (x_3, y_3)] \\
&\quad = y_1 \times Poly\,[x : (x_1, 1)\,, (x_2, 0)\,, (x_3, 0)] \\
&\qquad + y_2 \times Poly\,[x : (x_1, 0)\,, (x_2, 1)\,, (x_3, 0)] \\
&\qquad + y_3 \times Poly\,[x : (x_1, 0)\,, (x_2, 0)\,, (x_3, 1)]\,.
\end{aligned} \tag{5}$$

The right-hand side of (5) identifies the Lagrange polynomials as specific quadratic interpolations with y_i values set according to Eq. 4. All interpolations that are linear functions of the y_i values can be rewritten in a form similar to Eq. 5.

With the interpolating equation written in the form of Eq. 1, the propagation of uncertainty is simple. In particular, if we want to propagate the uncertainty in the y_i values, then Eq. 1 shows that the Lagrange polynomials are the sensitivity coefficients: they tell us how error in the measured y_i values propagates and causes errors in interpolated values. Therefore, the uncertainty in the interpolated y value due to uncertainties in the y_i values is [12]

$$u^2(\hat{y}) = L_1^2(x)u^2(y_1) + L_2^2(x)u^2(y_2) + L_3^2(x)u^2(y_3). \tag{6}$$

For simplicity, uncertainties in the x_i values have been neglected in Eq. 6. If required, the uncertainties in the x_i values can be included by using the full propagation of uncertainty equation [11]:

$$u^2(\hat{y}) = \sum_{i=1}^{N} L_i^2(x)u^2(y_i) - \sum_{i=1}^{N} L_i^2(x)\left(\left.\frac{\mathrm{d}\hat{y}}{\mathrm{d}x}\right|_{x=x_i}\right)^2 u^2(x_i) + \left(\frac{\mathrm{d}\hat{y}}{\mathrm{d}x}\right)^2 u^2(x), \tag{7}$$

where N is the number of data points, $u(x_i)$ are the uncertainties in the x values at the calibration points, and $u(x)$ is the uncertainty in x determined during use of the calibrated instrument.

Figure 1 shows the Lagrange polynomials and the total uncertainty according to Eq. 7 with $u(y_i) = 1.0$, $u(x_i) = 0.0$, and $u(x) = 0$ for a polynomial interpolation through six equally spaced points. Note that some parts of the total uncertainty curve are below the $u(\hat{y}) = 1.0$ line, while other parts lie above the line. The values of uncertainty below 1.0 are indicative of an interpolated value that depends on an average of y_i values. One of the most interesting features of the plots is the amplification of uncertainty, which is evident when any of the Lagrange polynomials takes an absolute value greater than 1.0. The amplification occurs with all the polynomials when they are outside the interpolation range ($3 \leq x \leq 8$ in Fig. 1), i.e., when the equation is used for extrapolation. The amplification is also often noticeable within the interpolation range, as it is in Fig. 1 with the $L_2(x)$ and $L_5(x)$ functions. Amplification effects become increasingly likely with high-order interpolations. For functions well represented by polynomials, the amplification problem can be solved by choosing the x-spacing of the data points according to the Chebyshev criterion [7,8]:

$$x_i = x_\mathrm{L} + (x_\mathrm{H} - x_\mathrm{L})\cos\left(\frac{2i-1}{2N}\right), \quad i = 1..N, \tag{8}$$

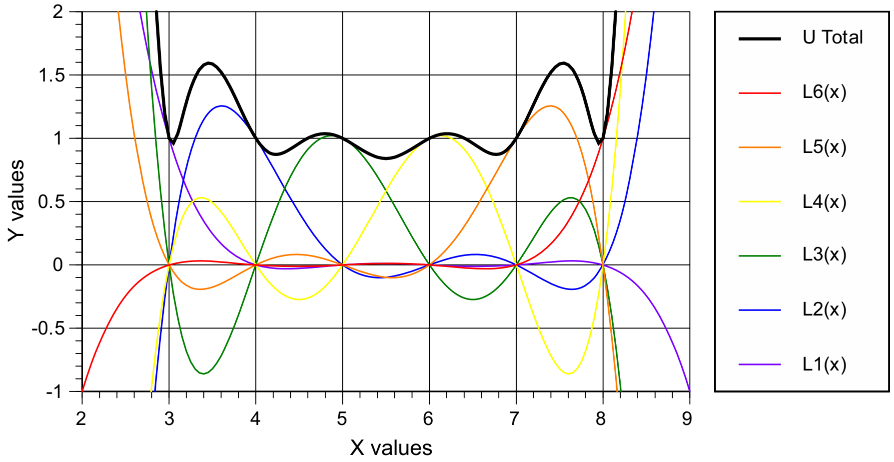

Fig. 1 Propagation of uncertainty with Lagrange interpolation through six equally spaced points (Color figure online)

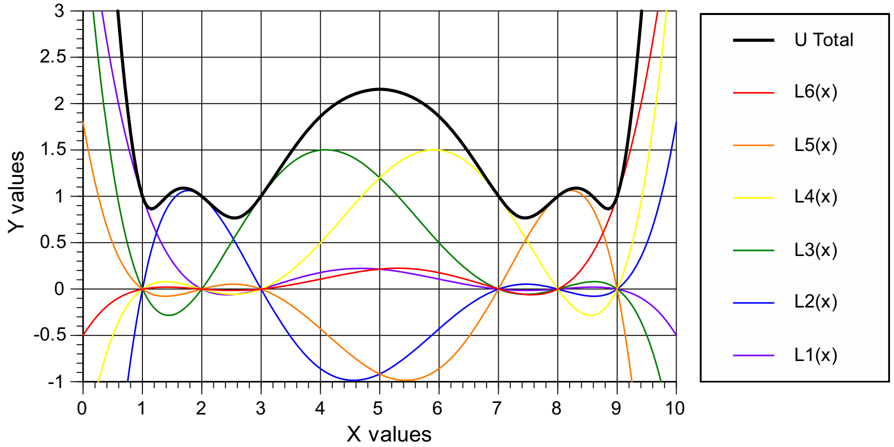

Fig. 2 Amplification of uncertainty that occurs with Lagrange interpolation with unequally spaced data points (Color figure online)

where x_L and x_H are the minimum and maximum values of x defining the interpolation range. In practice, other considerations usually dictate the selection and placement of the calibration points.

If there are large gaps in the spacing of the data points, the amplification of uncertainty is typically much worse. Figure 2 shows the Lagrange polynomials and total uncertainty for a six-point interpolation with a large gap between the third and fourth data points. In the range $4 \leq x \leq 6$, the total uncertainty has approximately doubled.

Interpolations that do not properly model the true behaviour of the measuring instrument exhibit interpolation error. For an N-point Lagrange interpolation, there is a formula for the error [7,8]

$$\hat{y}(x) - y(x) = e(x) = (x_1 - x)(x_2 - x)(...)(x_N - x)y^{(N)}(\xi)/(N)!, \tag{9}$$

where ξ is some unknown value of x lying within the interpolating range, and the superscript (N) indicates the Nth derivative with respect to x. Equation 9 shows the interpolation error is zero (i) at each of the calibration points and (ii) if the order of the interpolation equals or exceeds the order of the function modelled by the interpolation. For some functions, the amplitudes of the derivatives increase faster than $N!$ so the interpolation error unexpectedly increases as the number of the data points increases. This effect is known as Runge's phenomenon [7,9], and is one reason why high-order Lagrange interpolation is usually strongly discouraged [9]. For functions well approximated by simple polynomials, Runge's phenomenon should not be a problem. For data equally spaced at intervals of Δx, the interpolation error is less than [10]

$$|e(x)| < \frac{\Delta x^N y_{\max}^{(N)}}{4N}, \tag{10}$$

where $y_{\max}^{(N)}$ is the maximum absolute value of the Nth derivative of the interpolated function $y(x)$. If interpolation error is significant, then some estimate of it should be made and included in the uncertainty assessment. There are only a few examples of such analysis [4,11,13,14], but an expression based on Eqs. (9) or (10) with an estimate of $y_{\max}^{(N)}$ is a good starting point.

A simple test can be applied to any interpolating equation to find out if it interpolates a function exactly. For example, we expect all Lagrange interpolations to interpolate simple polynomials exactly, including the function $y = 1$. Therefore, from Eq. 1, we should find

$$L_1(x) + L_2(x) + L_3(x) = 1.0. \tag{11}$$

The second-order interpolation should also interpolate $y = x$ and $y = x^2$ exactly, and in general

$$\sum_{i=1}^{N} x_i^n L_i(x) = x^n, \quad n = 0..N \tag{12}$$

for all Lagrange interpolations. These identities suggest that to test whether any function $g(x)$ will be interpolated exactly, the following identity should be tested:

$$\sum_{i=1}^{N} g(x_i)L_i(x) = g(x). \tag{13}$$

If the identity holds, there will be no interpolation error. This test will be generalised below to other interpolating equations.

3 Other Interpolating Equations

3.1 Straight-Line Interpolation

Lagrange interpolation can be used to interpolate using the whole of a set of calibration data or over a small local section of data within a larger set. The most common example of local interpolation is straight-line interpolation between two neighbouring data points, but cubic interpolation between pairs of nearest neighbours may also be used in this way. These interpolations have the advantage of simplifying computations and minimising uncertainty amplification at the expense of a loss of continuity at each data point.

Straight-line interpolation has the form of the first-order Lagrange interpolation

$$\hat{y}(x) = y_1 \frac{(x - x_2)}{(x_1 - x_2)} + y_2 \frac{(x - x_1)}{(x_2 - x_1)}, \tag{14}$$

where x_1 and x_2 are the two points neighbouring x, i.e., $x_1 \leq x \leq x_2$.The corresponding propagation of uncertainty equation (considering the uncertainty in the y_i values only) is

$$u^2(\hat{y}) = \left(\frac{(x - x_2)}{(x_1 - x_2)} \right)^2 u^2(y_1) + \left(\frac{(x - x_1)}{(x_2 - x_1)} \right)^2 u^2(y_2). \tag{15}$$

Figure 3 shows the first-order Lagrange polynomials and the total uncertainty for the same situation as Fig. 1. The 'washing line' shape of the total uncertainty curve is characteristic of straight-line interpolation (see, for example, [15]). Straight-line interpolation demonstrates well the effects of averaging. For example, when interpolating midway between a pair of data points, $x = (x_1 + x_2)/2$, then $\hat{y}(x) = (y_1 + y_2)/2$, the average of the two values, and the total uncertainty is reduced by $\sqrt{2}$. Another

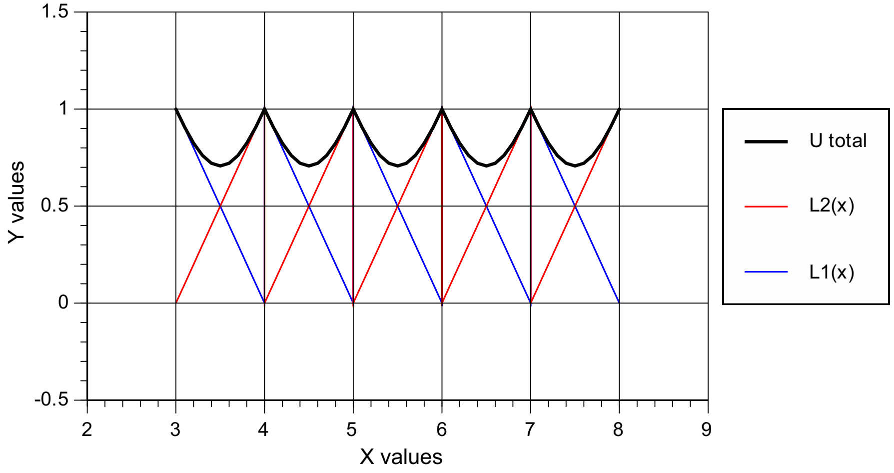

Fig. 3 First-order Lagrange polynomials (*red* and *blue lines*) and total uncertainty (*heavy black line*) for straight-line interpolation (Color figure online)

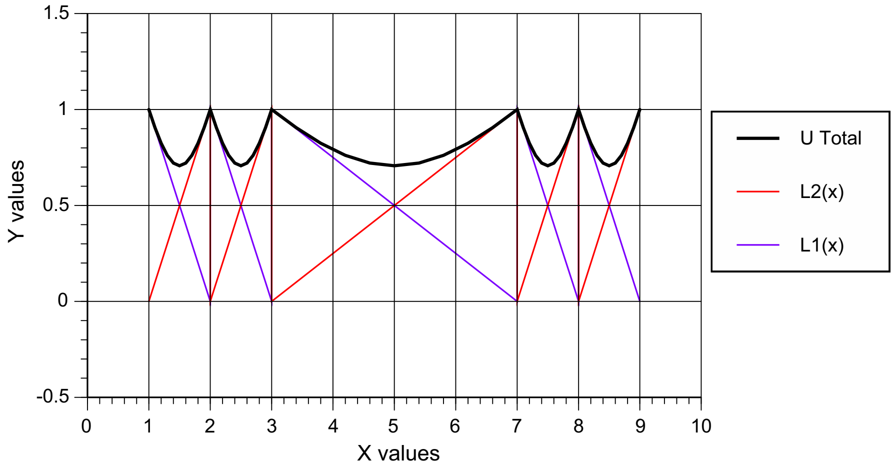

Fig. 4 First-order Lagrange polynomials (*red* and *blue lines*) and total uncertainty (*heavy black line*) for straight-line interpolation with unequally-spaced data points (Color figure online)

notable feature of straight-line interpolation is the complete absence of any uncertainty amplification, one of its major advantages. This point is emphasised by Fig. 4, which shows the uncertainty propagation for the unequally spaced data of Fig. 2.

3.2 The Natural Cubic Spline

Mathematical splines are the equivalent of the thin flexible wooden slat (the spline) used historically by draftsmen to draw a curved line passing through several points. The approach is similar to the idea of using a polynomial to interpolate between two points, except that constraints are placed on the derivatives of the interpolated values where they pass through the data points (the knots). The most popular spline is the natural cubic spline, which forces a smooth curve through four data points. The spline is continuous in its first and second derivatives, and the second derivative is identically zero at the two end points. The details of the calculation of the spline will not be given here; they are covered well by [1] and implemented in many mathematical packages such as Maple®, Mathematica®, and Matlab®, and in many graphing packages. One of the advantages of splines is that, although they require an Nth-order matrix inversion to compute the various coefficients of the equations, the matrix is sparse and enables the use of fast optimised algorithms with large data sets.

As with Lagrange interpolation, the spline equations are all linear functions of the y_i values, which means the equations can in principle (discouraged in practice) be expressed in the form

$$\hat{y}(x) = \sum_{i=1}^{N} y_i F_i(x : x_1 \ldots x_N), \tag{16}$$

that is, by using a series of functions of the x and x_i values only, with the y_i values as coefficients. This means, once again, that the $F_i(x)$ functions are the sensitivity

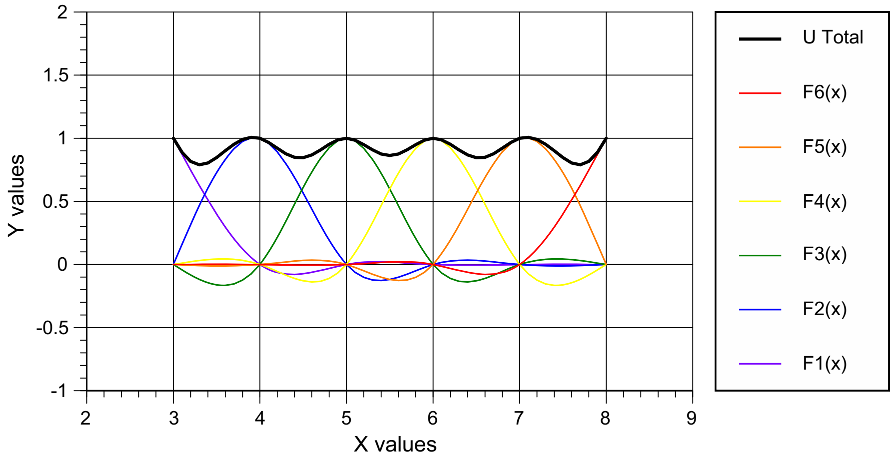

Fig. 5 Uncertainty propagation for the natural cubic spline applied to the equally spaced data of Fig. 1 showing total uncertainty (*black line*), and the sensitivity coefficients $F_i(x)$ (Color figure online)

coefficients and the propagation of uncertainty equation takes the same form as for Lagrange interpolation (Eq. 7):

$$u^2(\hat{y}) = \sum_{i=1}^{N} F_i^2(x)u^2(y_i) - \sum_{i=1}^{N} F_i^2(x)\left(\left.\frac{\mathrm{d}\hat{y}}{\mathrm{d}x}\right|_{x=x_i}\right)^2 u^2(x_i) + \left(\frac{\mathrm{d}\hat{y}}{\mathrm{d}x}\right)^2 u^2(x). \quad (17)$$

Equation 16 also indicates that the $F_i(x)$ functions satisfy the same orthonormality property given by Eq. 4. This observation and Eq. 5 suggest a simple way of computing the $F_i(x : x_1 \ldots x_N)$ functions: simply substitute

$$y_j = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases} \quad (18)$$

into the spline algorithm (using the same x_i values as the data), and the resulting spline will be the required $F_i(x)$. Although this process is a little clumsy, it is easily programmed into the mathematics packages that have the spline algorithm available. For example, the subroutine call for calculating $F_2(x)$ might be written as

$$F_2(x) = \mathit{CubicSpline}\left[x : (x_1, 0), (\boldsymbol{x_2}, \mathbf{1}), (x_3, 0), (x_4, 0), (x_5, 0), (x_6, 0)\right], \quad (19)$$

where the second point (highlighted in bold) has the only nonzero y_i value. Figure 5 shows all six $F_i(x)$ functions and total uncertainty when the natural cubic spline is applied to the data of Fig. 1. As with Lagrange polynomials, the $F_i(x)$ are highly oscillatory, but the minor effects of uncertainty amplification observed in Fig. 1 are now absent. Cubic splines can be applied to data sets of any number of equally spaced points without the amplification occurring.

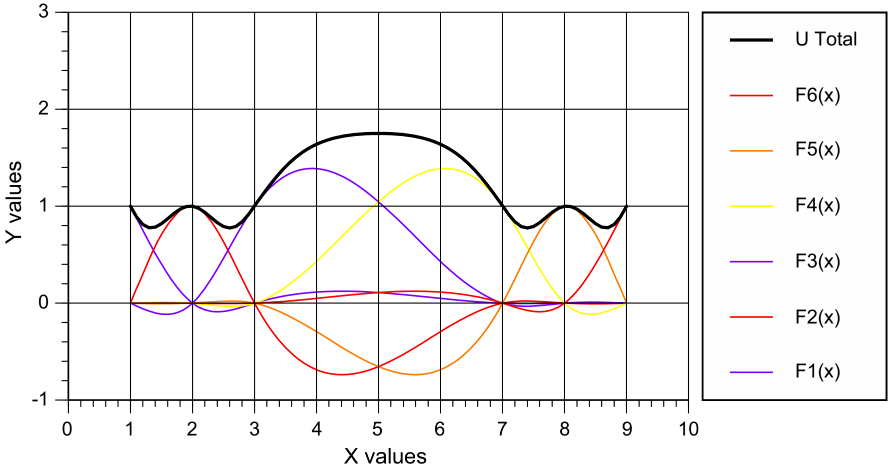

Fig. 6 Uncertainty propagation for the natural cubic spline applied to the unequally spaced data of Fig. 2 showing the total uncertainty (*black line*), and the sensitivity coefficients (Color figure online)

Figure 6 shows the uncertainty propagation for the cubic spline applied to the non-equally-spaced data of Fig. 2. Comparison with Fig. 2 shows that the uncertainty amplification in the large interval between $x = 3$ and $x = 7$ has been reduced, compared to Fig. 2, but not eliminated.

Algebraic experiments with the relations (based on Eq. 12)

$$\sum_{i=1}^{N} x_i^n F_i(x) = x^n, \quad n = 0 \ldots N, \tag{20}$$

show that Eq. 20 is true only for $N = 0, 1, 2$; and therefore, the natural cubic spline will interpolate only combinations of zeroth, first, and second-order polynomial functions without interpolation error. In contrast, a local cubic polynomial interpolation using the nearest two pairs of points will interpolate cubic functions exactly, but will, in general, have a discontinuous first derivative. With the natural cubic spline, the ability to interpolate cubic functions has been traded for a guarantee of second-order continuity.

3.3 Least-Squares Interpolations

All the interpolating equations described above are constructed to pass through all data points. This means that, near a data point, the interpolation is dominated by that point, and there can be little averaging and reduction of uncertainty, as occurs midway between data points. Because an interpolating equation fitted using the least-squares method is not constrained to pass through any data points, the benefits of averaging occur throughout the interpolation range, and this is a major advantage of least-squares fits. Least-squares fitting, as a means for interpolation, has several advantages in addition to the lower propagated uncertainty. The order of the equation can be chosen independently of the number of data points, and unlike the cubic spline, it uses a single formula for the entire data range, so is continuous to all orders. Most spreadsheet and graphing applications include least-squares fits as a feature, so the technique is also readily accessible. Since the least-squares method is very well known and well described in many texts, e.g., [1,2], the details of the equations will not be given here. However, one simple explanatory example is described, as follows.

For an unweighted (ordinary) least-squares fit to the quadratic equation

$$\hat{y} = a + bx + cx^2, \tag{21}$$

the parameter values are

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} \sum x_i^0 & \sum x_i^1 & \sum x_i^2 \\ \sum x_i^1 & \sum x_i^2 & \sum x_i^3 \\ \sum x_i^2 & \sum x_i^3 & \sum x_i^4 \end{pmatrix}^{-1} \begin{pmatrix} \sum y_i \\ \sum x_i y_i \\ \sum x_i^2 y_i \end{pmatrix}. \tag{22}$$

The key feature of Eq. 22 is that each of the parameters, a, b, and c, is a linear function of the y_i; and therefore, the fitted quadratic equation is also a linear function of the y_i. This is true for all linear least-squares fits (i.e., where the parameters $a, b, c\ldots$ are linear in the interpolating equation), whether for ordinary, weighted, or generalised least-squares methods [16]. Therefore, as with the cubic spline, the least-squares equations can be expressed in the form of Eq. 16, and the propagation of uncertainty equation in the form of Eq. 17. Also, as with the cubic spline, the sensitivity coefficients, $F_i(x)$, can be evaluated by making the substitution

$$y_j = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases} \tag{23}$$

into the least-squares algorithm (using the same x_i values as the data), and the resulting equation will be the required $F_i(x)$. The statement in software for determining $F_2(x)$ might read

$$F_2(x) = LeastSquares\left[x : (x_1, 0), (\boldsymbol{x_2}, \mathbf{1}), (x_3, 0), (x_4, 0), (x_5, 0), (x_6, 0)\right]. \tag{24}$$

Figure 7 shows the propagation of uncertainty for a quadratic least-squares fit for the same data as for Fig. 1. The resulting curve is identical to that obtained by conventional methods for calculating uncertainty based on propagation of uncertainty through the covariance matrix and then to the interpolated value. The most significant feature is the much lower total uncertainty achieved with the least-squares fit compared to the other interpolating equations. Indeed, even with a single surplus point (number of data points exceed number of fitted parameters by 1), there is a reduction in propagated uncertainty compared to polynomial interpolation. (If the number of fitted parameters equals the number of data points, then the fitted equation is identical to the equivalent Lagrange interpolation.)

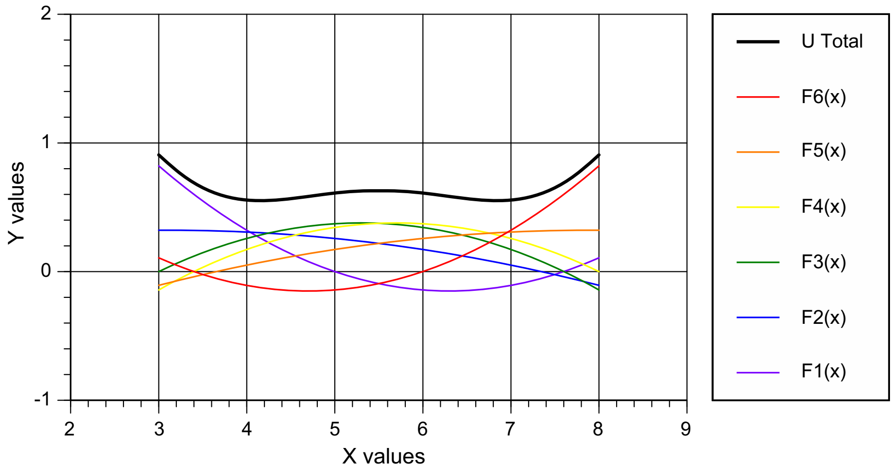

Fig. 7 Total uncertainty and sensitivity coefficients for a quadratic least-squares fit for six equally spaced data points, as in Fig. 1 (Color figure online)

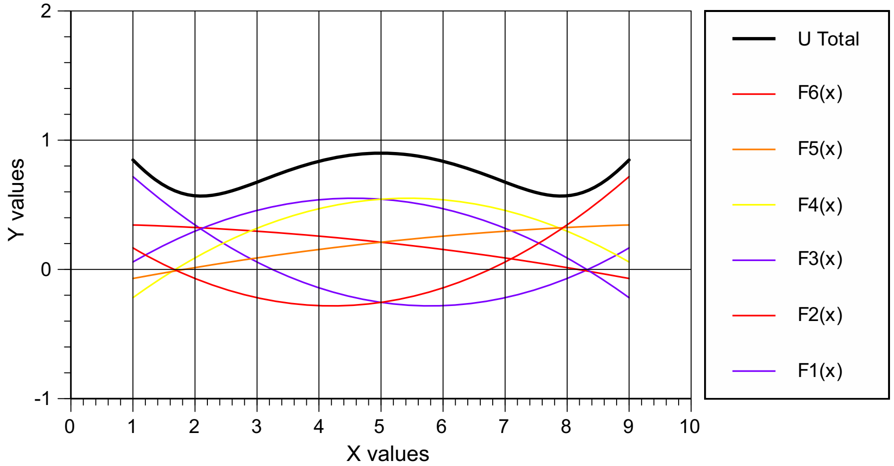

Fig. 8 Total uncertainty and sensitivity coefficients for a quadratic least-squares fit for six unequally spaced data points, as in Fig. 2 (Color figure online)

Another interesting feature of Fig. 7 is the rapid rise in uncertainty towards the ends of the interpolation range. This trend continues when the equation is used to extrapolate, and the uncertainty increases quadratically (for a quadratic fit).

Figure 8 shows the total uncertainty and sensitivity coefficients for the unequally spaced data and uncertainties as for Fig. 2. While there is a significant increase in the values of the sensitivity coefficient in the interval between $x = 3$ and $x = 7$, the total uncertainty is low and well behaved. In general, least-squares interpolation is more robust against data spacing than any of the other interpolations except straight-line interpolation.

4 Conclusions

This paper demonstrates a method for analysing the uncertainty propagation of interpolating equations that exploits the linear dependence of the interpolations on the y_i samples of the interpolated quantity. The technique is based on the linear expansion of the interpolating equation as a sum of interpolating functions each multiplied by a corresponding y_i value. The interpolating functions are therefore the sensitivity coefficients for the uncertainties in the y_i values. The interpolating functions can be determined by applying the interpolation to data points with the same x_i values, but with y_i values either unity or zero according to Eq. 4. The technique is applicable to all the most commonly used interpolating equations, including polynomial (Lagrange) interpolation, straight-line interpolation, splines, and least-squares fits.

There is no interpolating equation that is best for all applications. What is best depends on many factors: the complexity (order) of the interpolated function, the uncertainties in the measurements, whether continuity is important, the required ease of computation, the spacing of the data points, the relative magnitudes of measurement uncertainties, and the interpolation errors. However, interpolating equations determined by least-squares methods have some clear advantages: lowest propagated uncertainty of the equations considered here, good immunity to amplification effects due to the unequal spacing of the data points, continuity for all orders, and the order of the fitted equation can be selected independently from the number of data points.

Acknowledgements The author gratefully acknowledges members of the Metrology Group on LinkedIn whose questions prompted this study.

References

1. W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, *Numerical Recipes* (Cambridge University Press, Cambridge, 1986)
2. P.R. Bevington, *Data Reduction and Error Analysis for the Physical Sciences* (McGraw-Hill, New York, 1969)
3. P. Marcarino, P.P.M. Steur, A. Merlone, *Interpolating equations for Industrial Platinum Resistance Thermometers, in Advanced Mathematical and Computational Tools in Metrology VI* (World Scientific, Singapore, 2004), pp. 318–322
4. D.R. White, K. Hill, D. del Campo, C. Garcia Izquierdo, Guide on Secondary Thermometry: Thermistor Thermometry. (BIPM, Paris, 2014). http://www.bipm.org/utils/common/pdf/ITS-90/Guide-SecTh-Thermistor-thermometry.pdf
5. F. Sakuma, S. Hattori, Establishing a practical temperature scale standard by using a narrow-band radiation thermometer with a silicon detector, in *Temperature: Its Measurement and Control in Science and Industry*, vol. 5, ed. by J.F. Schooley (AIP, New York, 1982), pp. 421–427
6. P. Saunders, D.R. White, Physical basis of interpolation equations for radiation thermometry. Metrologia **40**, 195–204 (2003)
7. E.W. Cheney, *Introduction to Approximation Theory*, 2nd edn. (Chelsea Publishing, Providence, 1982)
8. T.J. Rivlin, *An Introduction to the Approximation of Functions* (Waltham, Blaisdell, 1969)
9. R.W. Hamming, *Numerical Methods for Scientists and Engineers*, 2nd edn. (Dover, New York, 1973)
10. M. Abramowitz, I.E. Stegun, *Handbook of Mathematical Functions* (US Gov. Printing Office, Washington DC, 1972)
11. D.R. White, P. Saunders, The propagation of uncertainty with calibration equations. Meas. Sci. Technol. **18**, 2157–2169 (2007)

12. BIPM, 2008, Evaluation of measurement data—guide to the expression of uncertainty in measurement JCGM 100:2008 (BIPM, Paris). http://www.bipm.org/utils/common/documents/jcgm/JCGM_100_2008_E.pdf
13. K.D. Hill, A.G. Steele, The non-uniqueness of the ITS-90: 13.8033 K to 273.16 K, in *Temperature: Its Measurement and Control in Science and Industry*, vol. 7, ed. by D.C. Ripple (American Institute of Physics, New York, 2002), pp. 53–58
14. P. Saunders, D.R. White, Interpolation errors for radiation thermometry. Metrologia **41**, 41–46 (2004)
15. J.L. Gardner, Correlations in primary spectral standards. Metrologia **40**, S167–S171 (2003)
16. J.V. Beck, K.J. Arnold, *Parameter Estimation in Science and Engineering* (Wiley, New York, 1977)