Multidimensional Interpolation Methods

1. Introduction

Data fitting methods such as spline fitting and interpolation are commonly used to model sparsely sampled data (e.g., for graphical surface rendering) or to speed up function calculation (e.g., for real-time measurement or control applications involving complex modeling simulations). This document describes function approximation algorithms for interpolating data that is sampled on a rectangular grid. The methods are focused on achieving high accuracy, minimal error bias, fast computational throughput, and minimal data storage requirements – especially for problems involving large data structures sampled on a multidimensional grid with many dimensions.

The simplest such method is linear interpolation in one dimension. For example, Figure 1 illustrates linear interpolation of the function $f[x] = \sin[\pi x]$ over the range $-1 \le x \le 1$, which covers six uniformly-sized data sampling intervals. Figure 2 illustrates the interpolation error, which is zero at the sampling points, but which exhibits a systematic bias within each sampling interval. The error is correlated with the function's second derivative, and can be reduced by applying a bias-compensating term to the sample data points. Figure 3 illustrates the interpolation using bias-compensated sample data, and Figure 4 illustrates the resulting interpolation error. The error is "unbiased" in the sense that the average error within each interval is approximately zero. (The bias-compensated interpolation algorithm differs from conventional interpolation because each data sample's ordinate does not represent the function $f[x]$ at a particular abscissa x – it is computed as a linear combination of $f[x]$ values evaluated at multiple x values.)

The "accuracy order" of an interpolation algorithm is defined as the highest integer n such that the interpolation error is zero for any polynomial of degree n or less.

The interpolation error typically scales in approximate proportion to δ^{n+1}, wherein δ is the sampling interval size. Linear interpolation has accuracy order 1, so the error is approximately proportional to δ^2. For example, Figure 5 illustrates the bias-compensated interpolation error for the above example using 24 sampling intervals. Compared to Figure 4, δ is reduced by a factor of 4, and the error is reduced by approximately a factor of $4^2 = 16$.

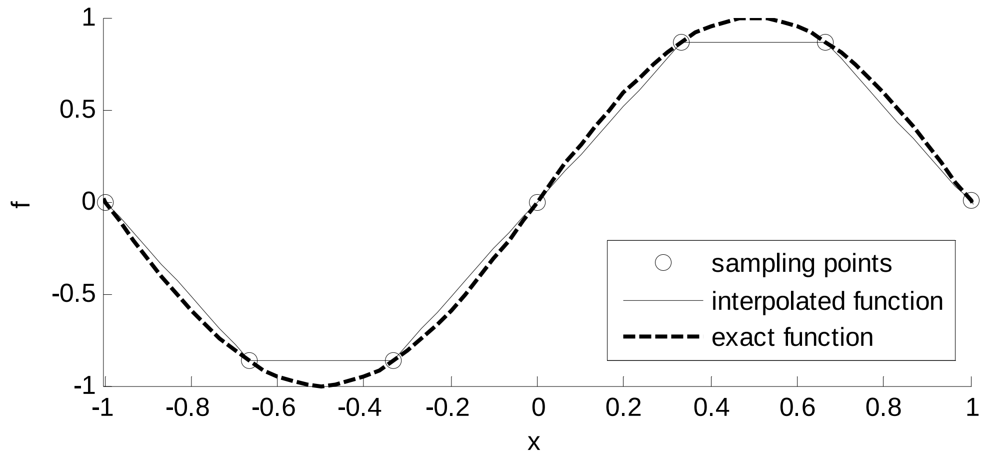

Figure 1. Linear interpolation.

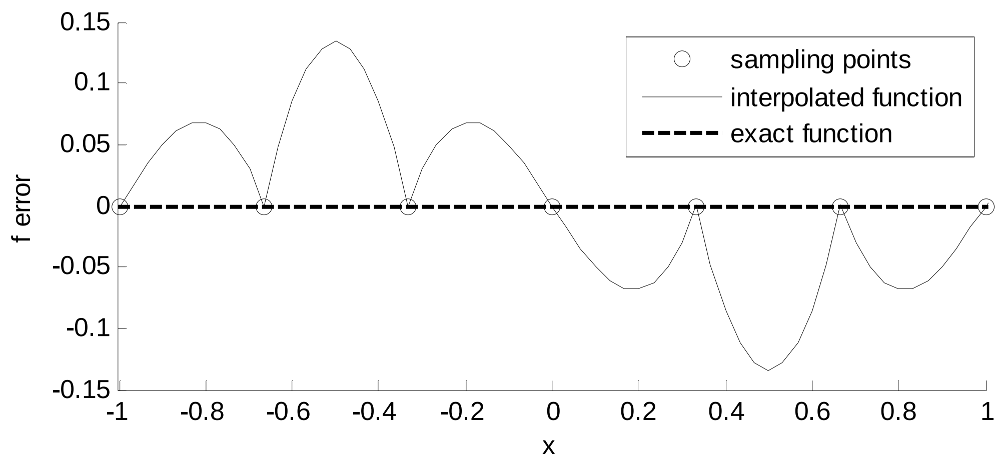

Figure 2. Linear interpolation error.

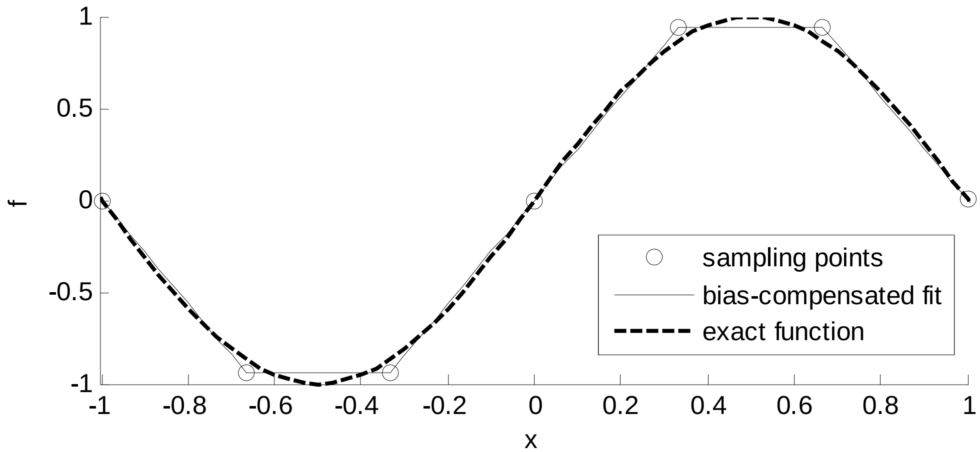

Figure 3. Interpolation with bias-compensated sample data.

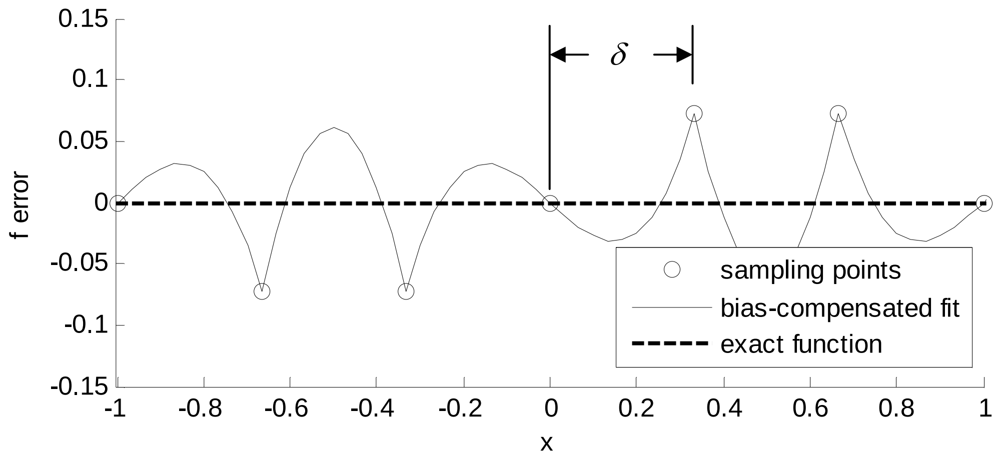

Figure 4. Interpolation error with bias-compensated sample data.

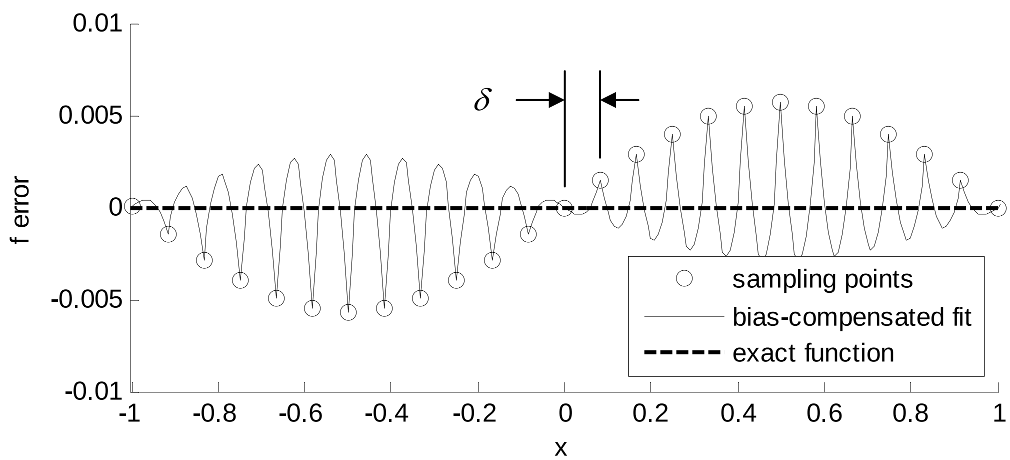

Figure 5. Bias-compensated interpolation error with 24 sampling intervals.

Cubic interpolation, which fits both a function and its derivative at the data sample points, has accuracy order 3. Figure 6 illustrates the cubic interpolation error for the above example, again using 24 sampling intervals. The error and its derivative are both zero at the sample points. In this example, the interpolation is computed using exact values for the derivative $f'[x]$ at the sample points. In practice, a finite-difference derivative estimator may be used, but if a simple, two-point centered-difference estimator is used the accuracy order is reduced to 2. Figure 7 illustrates the error for the preceding example using a two-point difference estimator. (Note the scale difference in Figures 6 and 7. The finite-difference approximation increases the interpolation error by an order of magnitude.) Order-3 interpolation accuracy can be restored by using a 4th-degree Savitzky-Golay derivative estimator, resulting in the interpolation error illustrated in Figure 8.

Whether the interpolation uses exact derivatives (as in Figure 6) or a 4th-degree derivative estimator (Figure 8), the cubic interpolation error exhibits a bias that is approximately correlated to the function's fourth derivative. As in the case of linear interpolation, this bias can be substantially eliminated by using bias-compensated sample data. For example, the error function illustrated in Figure 8 would be modified as illustrated in Figure 9 with bias compensation.

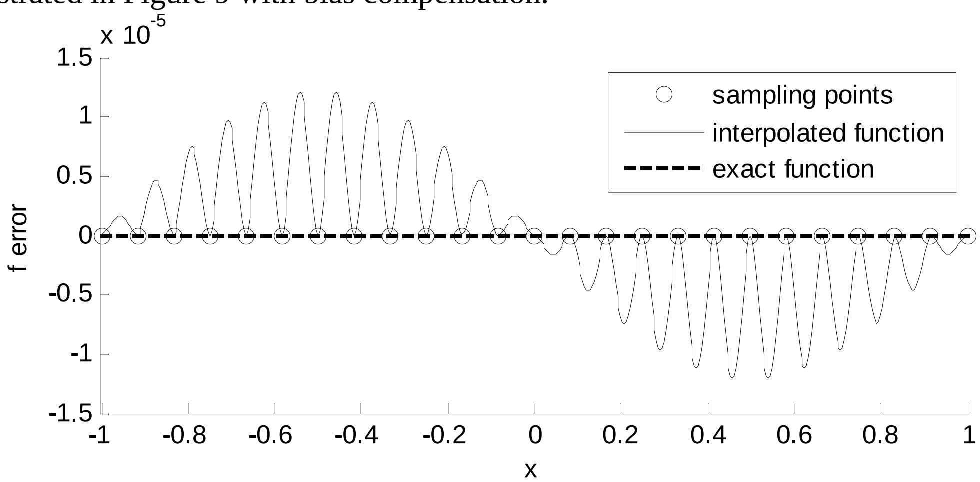

Figure 6. Cubic interpolation error with 24 sampling intervals.

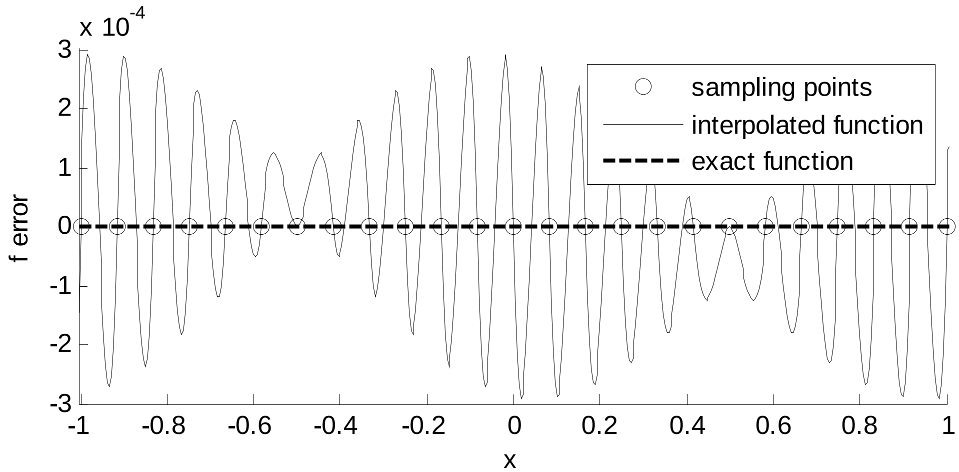

Figure 7. Cubic interpolation error using a two-point, centered-difference estimator for $f'[x]$.

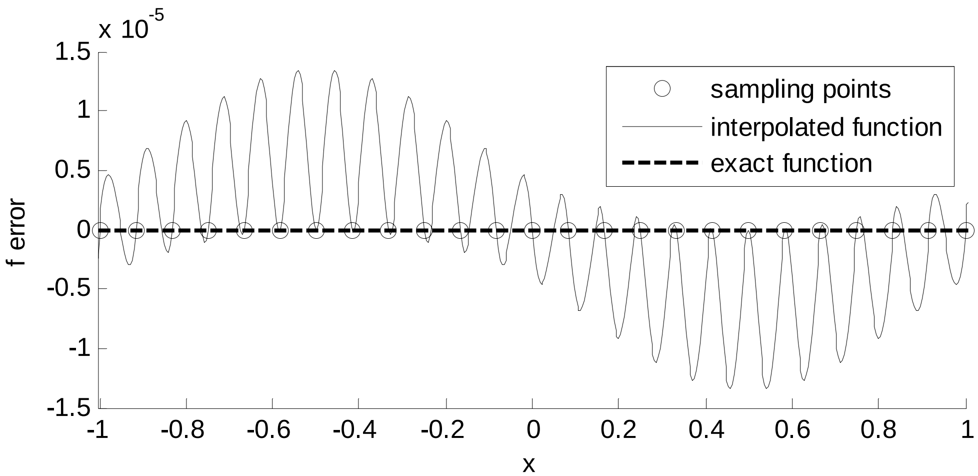

Figure 8. Cubic interpolation error using a 4^{th}-degree Savitzky-Golay derivative estimator.

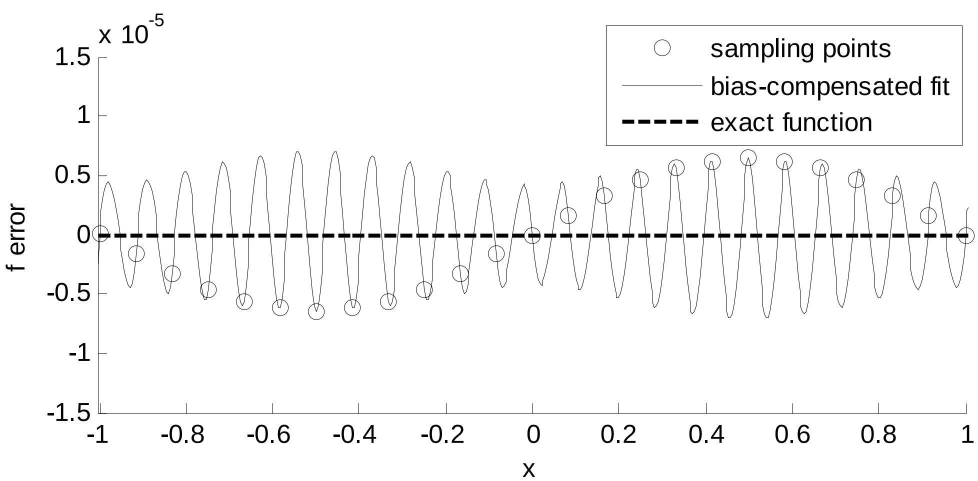

Figure 9. Bias-compensated cubic interpolation error using a 4^{th}-degree Savitzky-Golay derivative estimator.

The interpolation methods illustrated above can be generalized to apply to a multidimensional function $f[\boldsymbol{x}]$, wherein $\boldsymbol{x}$ is an N-dimensional vector, $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$, and f is sampled on a uniform rectangular grid. For example, linear interpolation generalizes to multilinear interpolation by applying linear interpolation to each coordinate x_j ($j = 1 \ldots N$). Similarly, applying cubic interpolation to each separate coordinate results in a "multicubic" interpolation that not only matches f and its first derivatives $\partial f / \partial x_j$ at the data sampling points, but also matches all mixed first-order derivatives (e.g., $\partial^2 f / \partial x_1 \, \partial x_2$, $\partial^3 f / \partial x_1 \, \partial x_2 \, \partial x_3$, etc.). This requires that 2^N quantities (the f value and derivatives) be specified at each sampling point, but for large N many of these quantities can be omitted without reducing the accuracy order. The number of quantities required per sample point to achieve accuracy order 3 is $1 + N(5 + N^2)/6$. Furthermore, accuracy order 2 can be achieved with only $1 + N(1 + N)/2$ quantities per sample point.

Multilinear interpolation yields an interpolation function that is globally continuous. A multicubic interpolation function is globally continuous and smooth (i.e., continuously differentiable), but if smoothness is not strictly required an alternative "reduced-cubic" interpolation method may be used.[1] This method also has accuracy order 3, but it only requires that $1 + N$ quantities (the function f and its derivatives $\partial f / \partial x_j$) be specified at each sample point. Moreover, the reduced-cubic interpolation function's gradient discontinuity is comparatively mild. (The gradient is continuous at the sampling grid points, and the discontinuity between grid cells is only of order δ^3. By contrast, the gradient discontinuity for multilinear interpolation is of order δ.)

The bias compensation methods illustrated above for one dimension can be generalized to multidimensional interpolation. Also, the multidimensional derivatives may be approximated by finite differences or Savitzky-Golay estimates.

All of the interpolation methods discussed above require data sampled on a uniform, rectangular grid; however the methods can also be adapted to accommodate nonuniform data sampling. For example, Figures 10-12 illustrate several types of two-dimensional sampling grids. In each of these illustrations the grid points are denoted as $\boldsymbol{x}^{[m_1, m_2]} = (x_1^{[m_1, m_2]}, x_2^{[m_1, m_2]})$, wherein m_1 and m_2 are integer indices labeling the points. Figure 10 illustrates a uniform rectangular grid (which has grid lines parallel to the coordinate axes); Figure 11 illustrates a nonuniform rectangular grid (also with axes-aligned grid lines), and Figure 12 illustrates a more general nonuniform, non-rectangular grid.

[1] The reduced-cubic interpolation method has its genesis in optical metrology; see U. S. Patent 6,947,135, "Reduced multicubic database interpolation method for optical measurement of diffractive microstructures," filed on July 1, 2003, issued on Sept. 20, 2005. Also see ip.com, Document ID IPCOM000022148D, "Accuracy-Optimized Function Approximation Method for Optical Metrology," Feb. 27, 2004.

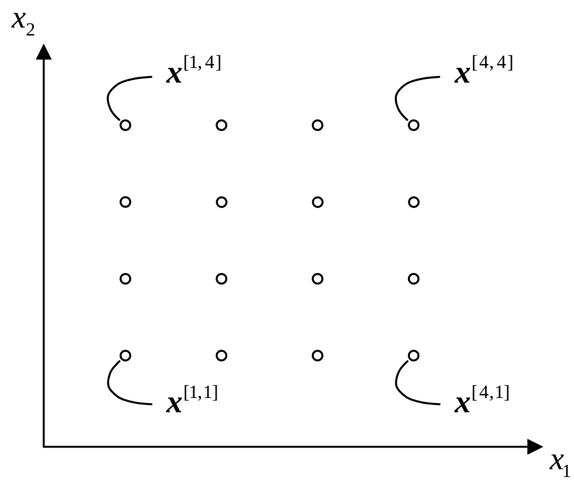

Figure 10. Uniform, rectangular sampling grid.

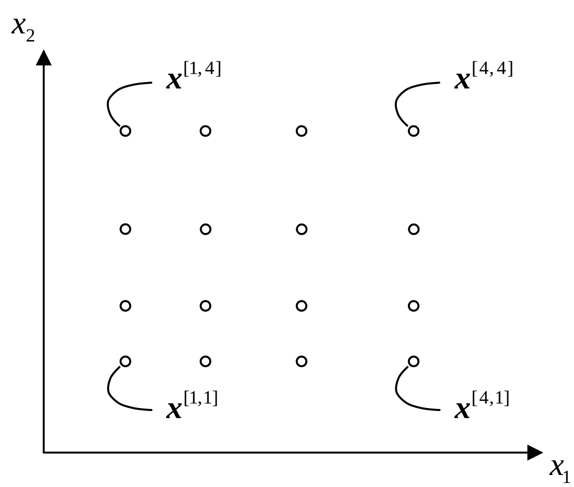

Figure 11. Nonuniform, rectangular sampling grid.

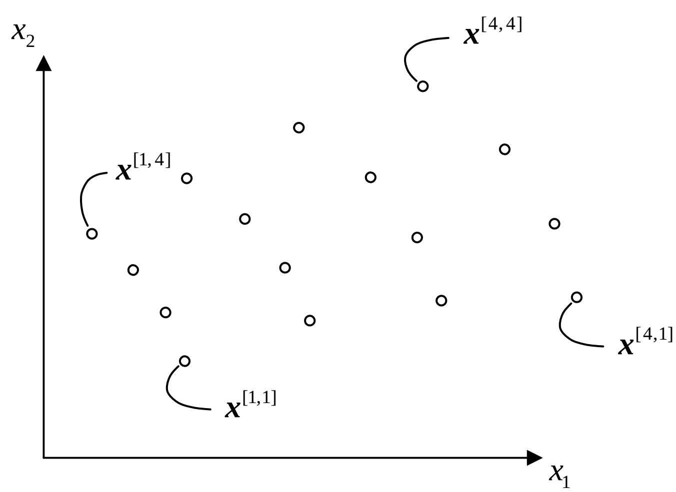

Figure 12. Nonuniform, non-rectangular sampling grid.

The function $\boldsymbol{x}^{[m_1, m_2]}$ can be defined for real values of m_1 and m_2 by interpolating between grid points. A function $f[\boldsymbol{x}]$ is interpolated by first performing a numerical inversion of $\boldsymbol{x}^{[m_1, m_2]}$ to determine the m_1 and m_2 indices corresponding to $\boldsymbol{x}$, and then interpolating the function $f^{[m_1, m_2]} = f[\boldsymbol{x}^{[m_1, m_2]}]$ at the determined m_1 and m_2 values.

2. Notational conventions and definitions

The following notational conventions are used herein: Parentheses – "(…)" – are used for grouping and for delimiting lists and matrices. Square braces – "[…]" – delimit function arguments and superscripts (e.g., $y[\boldsymbol{x}]$, $\boldsymbol{x}^{[m_1, m_2]}$ – this avoids ambiguity with products and exponents). Curly braces – "{…}" – denote sets. Vectors are indicated in bold type, as in $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$. The parentheses may be omitted if there is no ambiguity; for example a multivariate function $y[\boldsymbol{x}] = y[(x_1, x_2, \ldots, x_N)]$ may alternatively be written as $y[x_1, x_2, \ldots, x_N]$. (A similar convention applies to superscript and subscript arguments, e.g., $\boldsymbol{x}^{[m_1, m_2, \ldots, m_N]} = \boldsymbol{x}^{[\boldsymbol{m}]}$ with $\boldsymbol{m} = (m_1, m_2, \ldots, m_N)$.) $\boldsymbol{e}_j$ represents the unit vector $(e_{j,1}, e_{j,2}, \ldots, e_{j,N})$ with

$$e_{j,k} = \begin{cases} 1, & j = k \\ 0, & j \neq k \end{cases} \tag{2.1}$$

The following derivative notation will be used,

$$\partial_j{}^k y[x_1, x_2, \ldots, x_N] = \frac{\partial^k}{\partial x_j{}^k} y[x_1, x_2, \ldots, x_N] \tag{2.2}$$

(For $k = 0$, $\partial_j{}^k y = y$.) If y is a function of just one scalar argument, a similar notation is applied,

$$d^k y[x] = \frac{d^k}{dx^k} y[x] \tag{2.3}$$

Product notation is used to denote mixed derivatives,

$$\left(\prod_{j \in \{1, 2, \ldots\}} \partial_j{}^{k_j} \right) y = \partial_1{}^{k_1} \partial_2{}^{k_2} \ldots y \tag{2.4}$$

We consider the problem of interpolating a function $y[\boldsymbol{x}]$, based on a sampling of y at points $\boldsymbol{x} = \boldsymbol{x}^{[\boldsymbol{m}]}$, wherein $\boldsymbol{x}$ is a real-valued, N-dimensional vector argument ($\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$, $\boldsymbol{x}^{[\boldsymbol{m}]} = (x_1^{[\boldsymbol{m}]}, x_2^{[\boldsymbol{m}]}, \ldots, x_N^{[\boldsymbol{m}]})$), and $\boldsymbol{m}$ is an integer-valued, N-dimensional index list ($\boldsymbol{m} = (m_1, m_2, \ldots, m_N)$). The data sampling limits are defined as

$$m_j \in \{0, 1, \ldots, M_j\} \tag{2.5}$$

The interpolation algorithms require a uniform, rectangular sampling grid, which has the property that $x_j^{[\boldsymbol{m}]}$ depends only on m_j and is a linear function of m_j. Without loss of generality, the coordinate origin and dimensional units are chosen so that $x_j^{[\boldsymbol{m}]} = m_j$,

$$\boldsymbol{x}^{[\boldsymbol{m}]} = \boldsymbol{m} \tag{2.6}$$

The interpolation function is denoted as $yFit$,

$$yFit[\boldsymbol{x}] \cong y[\boldsymbol{x}] \tag{2.7}$$

$yFit$ is a piecewise-polynomial function. The function has a polynomial form $yFit^{[\boldsymbol{m}]}$ within grid cell $\boldsymbol{m}$, which is defined as the set of $\boldsymbol{x}$ points such that $m_j \le x_j \le m_j + 1$,

$$yFit[\boldsymbol{x}] = yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] \quad \text{for} \quad m_j \le x_j \le m_j + 1 \quad (j \in \{1, \ldots, N\}) \tag{2.8}$$

The accuracy order of the interpolation is defined as highest integer n such that the interpolation error ($yFit[\boldsymbol{x}] - y[\boldsymbol{x}]$) is zero for any polynomial $y[\boldsymbol{x}]$ of degree n or less. (The "degree" of a polynomial in $\boldsymbol{x}$ is defined as the maximum degree of its monomial terms, wherein a monomial of the form $\prod_j (x_j)^{k_j}$ has a degree of $\sum_j k_j$.) For a general function $y[\boldsymbol{x}]$, the interpolation error typically scales in approximate proportion to δ^{n+1}, wherein δ is a factor that represents the dimensional scale of the coordinate sampling intervals. To state this condition more formally, we suppose that y is defined in terms of a function f such that the f argument is sampled on a rectangular grid with a sampling interval of δ in all dimensions, i.e.,

$$y[\boldsymbol{x}] = f[\delta\, \boldsymbol{x}] \tag{2.9}$$

(The y argument, $\boldsymbol{x}$, is sampled at unit intervals, so this relation implies that the f argument is sampled at size-δ intervals.) $y[\boldsymbol{x}]$ can typically be expanded in a Taylor series about the grid point $\boldsymbol{x} = \boldsymbol{m}$,

$$y[\boldsymbol{x}] = \sum_{k_1,\ldots,k_N \in \{0,1,2,\ldots\}} \left(\left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\,k_j} \right) y[\boldsymbol{m}] \right) \prod_{j\in\{1,\ldots,N\}} \frac{(x_j - m_j)^{k_j}}{k_j!} \qquad (2.10)$$

(Note: In equation 2.10 and throughout this document the expression 0^0 is defined as 1.) The derivatives of y implicitly include factors of δ,

$$\left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\,k_j} \right) y[\boldsymbol{m}] = \left(\prod_{j\in\{1,\ldots,N\}} \delta^{k_j} \partial_j^{\,k_j} \right) f[\delta\,\boldsymbol{m}] = \delta^{\sum \boldsymbol{k}} \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\,k_j} \right) f[\delta\,\boldsymbol{m}] \qquad (2.11)$$

wherein $\sum \boldsymbol{k}$ is shorthand for

$$\sum \boldsymbol{k} = \sum_{j\in\{1,\ldots,N\}} k_j \qquad (2.12)$$

($\sum \boldsymbol{k}$ is termed the “order” of the differential operator $\prod_j \partial_j^{\,k_j}$.)

Using the above representation and expanding the interpolation error for grid cell $\boldsymbol{m}$ in powers of δ, the interpolation error is determined to be of order δ^{n+1}, wherein n is the interpolation accuracy order,

$$yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = O\,\delta^{n+1} \qquad (2.13)$$

In this type of expression the residual error $O\,\delta^{n+1}$ includes y derivative factors of order $(n+1)$. The error will converge to zero as δ becomes small, provided that the derivatives up to order $(n+1)$ all exist.

3. One-dimensional methods

3.1 Linear interpolation

Linear interpolation in one dimension is defined as follows,

$$yFit^{[m]}[x] = u_0^{[m]}[x]\, y[m] + u_1^{[m]}[x]\, y[m+1] \qquad (3.1.1)$$

wherein $u_0^{[m]}[x]$ and $u_1^{[m]}[x]$ are “cardinal functions”, which are defined so that

$$yFit^{[m]}[m] = y[m], \quad yFit^{[m]}[m+1] = y[m+1] \qquad (3.1.2)$$

for any function y. This implies that the cardinal functions satisfy the following defining conditions,

$$u_s^{[m]}[m] = 1 - s, \quad u_s^{[m]}[m+1] = s \quad (s \in \{0,1\}) \tag{3.1.3}$$

Furthermore, $u_0^{[m]}[x]$ and $u_1^{[m]}[x]$ are linear in x. These conditions imply that

$$u_s^{[m]}[x] = (1-s) + (2s-1)(x-m) \quad (s \in \{0,1\}) \tag{3.1.4}$$ [2]

The interpolation error, $yFit^{[m]}[x] - y[x]$, can be approximated by making the substitution

$$y[x] = f[\delta x] \tag{3.1.5}$$

(cf. equation 2.9) and expanding $yFit^{[m]}[x] - y[x]$ to 2nd order in δ. This yields

$$yFit^{[m]}[x] - y[x] = \tfrac{1}{2}(x-m)(m+1-x)\, d^2 y[m] + O\,\delta^3 \tag{3.1.6}$$

The factor $d^2 y[m]$ is equal to $\delta^2 d^2 f[\delta m]$, so the error is of order δ^2. The factor $(x-m)(m+1-x)$ in equation 3.1.6 is non-negative across the interpolation interval ($m \le x \le m+1$); thus the error is systematic and is proportional to $d^2 y$. The average error across the interval is

$$\int_m^{m+1} (yFit^{[m]}[x] - y[x])\, dx = \tfrac{1}{12} d^2 y[m] + O\,\delta^3 \tag{3.1.7}$$

The error bias can be substantially eliminated by including a bias-compensating term of $-\tfrac{1}{12} d^2 y[m]$ on the right side of equation 3.1.1. Taking advantage of the following two relationships,

$$u_0^{[m]}[x] + u_1^{[m]}[x] = 1 \tag{3.1.8}$$

$$d^2 y[m] = d^2 y[m+1] + O\,\delta^3 \tag{3.1.9}$$

(from equations 3.1.4 and 3.1.5), the bias-compensated interpolation function can be represented as

---

[2] An equivalent form of equation 3.1.4 (with $m \le x \le m+1$) is

$$u_s^{[m]}[x] = 1 - |x - m - s| \quad (s \in \{0,1\})$$

$$
\begin{aligned}
&u_0^{[m]}[x]\,y[m] + u_1^{[m]}[x]\,y[m+1] - \tfrac{1}{12}\,d^2 y[m] \\
&= u_0^{[m]}[x]\,(y[m] - \tfrac{1}{12}\,d^2 y[m]) + u_1^{[m]}[x]\,(y[m+1] - \tfrac{1}{12}\,d^2 y[m]) \\
&= u_0^{[m]}[x]\,(y[m] - \tfrac{1}{12}\,d^2 y[m]) + u_1^{[m]}[x]\,(y[m+1] - \tfrac{1}{12}\,d^2 y[m+1]) + O\,\delta^3
\end{aligned}
\tag{3.1.10}
$$

The $O\,\delta^3$ term on the right of equation 3.1.10 is negligible – it will be absorbed by the $O\,\delta^3$ residual in equation 3.1.6. Thus, the bias-corrected interpolation function $yFit'$ (primed, to distinguish it from the uncorrected function $yFit$) can be defined as

$$
yFit'^{[m]}[x] = u_0^{[m]}[x]\;yComp[m] + u_1^{[m]}[x]\;yComp[m+1] \tag{3.1.11}
$$

wherein $yComp$ is a bias-compensated function defined as

$$
yComp[x] = y[x] - \tfrac{1}{12}\,d^2 y[x] \tag{3.1.12}
$$

(Assuming that y and $d^2 y$ are continuous, bias compensation preserves continuity of $yComp$ and $yFit'$.)

With bias compensation, the interpolation error is

$$
yFit'^{[m]}[x] - y[x] = \left(\tfrac{1}{2}\,(x-m)\,(m+1-x) - \tfrac{1}{12}\right) d^2 y[m] + O\,\delta^3 \tag{3.1.13}
$$

and the interval-averaged error is

$$
\int_m^{m+1} (yFit'^{[m]}[x] - y[x])\;dx = O\,\delta^4 \tag{3.1.14}
$$

Note that both the δ^2 and the δ^3 terms average out to zero in equation 3.1.14, leaving a residual bias of order δ^4.

The $d^2 y$ term in equation 3.1.12 can be approximated using a finite-difference derivative estimator,

$$
d^2 y[x] = y[x-1] - 2\,y[x] + y[x+1] + O\,\delta^4 \tag{3.1.15}
$$

This substitution in equation 3.1.12 changes $yFit'^{[m]}[x]$ (equation 3.1.11) by an increment of order δ^4, so it has no effect on equations 3.1.13 and 3.1.14. If the interpolation interval is at one of the limits of the sampling range, constant extrapolation of $d^2 y$ may be used to avoid out-of range indexing in equation 3.1.15,

$$d^2 y[x \pm 1] = d^2 y[x] + O\,\delta^3 \tag{3.1.16}$$

In this case, equation 3.1.13 is unaffected, but the residual bias (equation 3.1.14) is of order δ^3,

$$\int_m^{m+1} (yFit'^{[m]}[x] - y[x])\, dx = O\,\delta^3 \tag{3.1.17}$$

3.2 Cubic interpolation

Cubic interpolation in one dimension is defined as follows,

$$\begin{aligned} yFit^{[m]}[x] = u_{0,0}^{[m]}[x]\, y[m] + u_{0,1}^{[m]}[x]\, y[m+1] + \\ u_{1,0}^{[m]}[x]\, dy[m] + u_{1,1}^{[m]}[x]\, dy[m+1] \end{aligned} \tag{3.2.1}$$

wherein the cardinal functions $u_{k,s}^{[m]}[x]$ are defined so that

$$\left.\begin{aligned} yFit^{[m]}[m+s] = y[m+s] \\ dyFit^{[m]}[m+s] = dy[m+s] \end{aligned}\right\} \quad (s \in \{0,1\}) \tag{3.2.2}$$

for any function y. This implies that the cardinal functions satisfy the following defining conditions,

$$\left.\begin{array}{ll} u_{0,s}^{[m]}[m] = 1 - s, & u_{0,s}^{[m]}[m+1] = s \\ du_{0,s}^{[m]}[m] = 0, & du_{0,s}^{[m]}[m+1] = 0 \\ u_{1,s}^{[m]}[m] = 0, & u_{1,s}^{[m]}[m+1] = 0 \\ du_{1,s}^{[m]}[m] = 1 - s, & du_{1,s}^{[m]}[m+1] = s \end{array}\right\} \quad (s \in \{0,1\}) \tag{3.2.3}$$

Furthermore, the cardinal functions $u_{k,s}^{[m]}[x]$ are cubic polynomials in x. These conditions imply that

$$\left.\begin{aligned} u_{0,s}^{[m]}[x] = s + (1 - 2\,s)\,(1 + 2\,(x - m))\,(1 + m - x)^2 \\ u_{1,s}^{[m]}[x] = (x - m)\,(1 + m - s - x)\,(1 + m - x) \end{aligned}\right\} \quad (s \in \{0,1\}) \qquad (3.2.4)^3$$

[3] An equivalent form of equation 3.2.4 (with $m \le x \le m+1$) is

$$\left.\begin{aligned} u_{0,s}^{[m]}[x] = \left(1 + 2\left|x - m - s\right|\right)\left(1 - \left|x - m - s\right|\right)^2 \\ u_{1,s}^{[m]}[x] = \left(x - m - s\right)\left(1 - \left|x - m - s\right|\right)^2 \end{aligned}\right\} \quad (s \in \{0,1\})$$

Making substitution 3.1.5 ($y[x] = f[\delta\, x]$) and expanding the interpolation error to 4th order in δ, the error has the form,

$$yFit^{[m]}[x] - y[x] = -\tfrac{1}{24}(x-m)^2\,(m+1-x)^2\; d^4 y[m] + O\,\delta^5 \tag{3.2.5}$$

The average error across the interpolation interval is

$$\int_m^{m+1} (yFit^{[m]}[x] - y[x])\; dx = -\tfrac{1}{720}\, d^4 y[m] + O\,\delta^5 \tag{3.2.6}$$

The error bias can be substantially eliminated by including a bias-compensating term of $\tfrac{1}{720}\, d^4 y[m]$ on the right side of equation 3.2.1. Taking advantage of the following relationships,

$$u_{0,0}^{[m]}[x] + u_{0,1}^{[m]}[x] = 1 \tag{3.2.7}$$

$$d^4 y[m] = d^4 y[m+1] + O\,\delta^5 \tag{3.2.8}$$

$$d^5 y[m] = O\,\delta^5, \quad d^5 y[m+1] = O\,\delta^5 \tag{3.2.9}$$

(from equations 3.1.5 and 3.2.4), the bias-compensated interpolation function can be represented as

$$\begin{aligned}
&u_{0,0}^{[m]}[x]\, y[m] + u_{0,1}^{[m]}[x]\, y[m+1] + \\
&\quad u_{1,0}^{[m]}[x]\, dy[m] + u_{1,1}^{[m]}[x]\, dy[m+1] + \tfrac{1}{720}\, d^4 y[m] \\
&= u_{0,0}^{[m]}[x](y[m] + \tfrac{1}{720}\, d^4 y[m]) + u_{0,1}^{[m]}[x](y[m+1] + \tfrac{1}{720}\, d^4 y[m]) + \\
&\quad u_{1,0}^{[m]}[x]\, dy[m] + u_{1,1}^{[m]}[x]\, dy[m+1] \\
&= u_{0,0}^{[m]}[x](y[m] + \tfrac{1}{720}\, d^4 y[m]) + u_{0,1}^{[m]}[x](y[m+1] + \tfrac{1}{720}\, d^4 y[m+1]) + \\
&\quad u_{1,0}^{[m]}[x](dy[m] + \tfrac{1}{720}\, d^5 y[m]) + u_{1,1}^{[m]}[x](dy[m+1] + \tfrac{1}{720}\, d^5 y[m+1]) + O\,\delta^5
\end{aligned} \tag{3.2.10}$$

The $O\,\delta^5$ term on the right of equation 3.2.10 is negligible – it will be absorbed by the $O\,\delta^5$ residual in equation 3.2.5. Thus, the bias-corrected interpolation can be defined as

$$\begin{aligned}
yFit'^{[m]}[x] = {} & u_{0,0}^{[m]}[x]\, yComp[m] + u_{0,1}^{[m]}[x]\, yComp[m+1] + \\
& u_{1,0}^{[m]}[x]\, dyComp[m] + u_{1,1}^{[m]}[x]\, dyComp[m+1]
\end{aligned} \tag{3.2.11}$$

wherein $yComp$ is a bias-compensated function defined as

$$yComp[x] = y[x] + \tfrac{1}{720}\, d^4 y[x] \tag{3.2.12}$$

With bias compensation, the interpolation error is

$$yFit'^{[m]}[x] - y[x] = \left(-\tfrac{1}{24}(x-m)^2\,(m+1-x)^2 + \tfrac{1}{720}\right) d^4 y[m] + O\,\delta^5 \tag{3.2.13}$$

and the interval-averaged error is

$$\int_m^{m+1} (yFit'^{[m]}[x] - y[x])\, dx = O\,\delta^6 \tag{3.2.14}$$

Note that both the δ^4 and the δ^5 terms average out to zero in equation 3.2.14, leaving a residual bias of order δ^6.

The derivatives in equations 3.2.1 and 3.2.11 (dy and $dyComp$) can be approximated using a 4th -degree, symmetric Savitzky-Golay derivative estimator,

$$dy[x] = \tfrac{2}{3}(y[x+1] - y[x-1]) - \tfrac{1}{12}(y[x+2] - y[x-2]) + O\,\delta^5 \tag{3.2.15}$$

The following asymmetric derivative estimators can also be used to avoid out-of-range indexing in equation 3.2.15,

$$dy[x] = \pm\tfrac{1}{12}\, y[x\pm 3] \mp \tfrac{1}{2}\, y[x\pm 2] \pm \tfrac{3}{2}\, y[x\pm 1] \mp \tfrac{5}{6}\, y[x] \mp \tfrac{1}{4}\, y[x\mp 1] + O\,\delta^5 \tag{3.2.16}$$

$$dy[x] = \mp\tfrac{1}{4}\, y[x\pm 4] \pm \tfrac{4}{3}\, y[x\pm 3] \mp 3\, y[x\pm 2] \pm 4\, y[x\pm 1] \mp \tfrac{25}{12}\, y[x] + O\,\delta^5 \tag{3.2.17}$$

If there are only four sampling points, the following derivative estimators may be used,

$$dy[x] = \mp\tfrac{1}{6}\, y[x\pm 2] \pm y[x\pm 1] \mp \tfrac{1}{2}\, y[x] \mp \tfrac{1}{3}\, y[x\mp 1] + O\,\delta^4 \tag{3.2.18}$$

$$dy[x] = \pm\tfrac{1}{3}\, y[x\pm 3] \mp \tfrac{3}{2}\, y[x\pm 2] \pm 3\, y[x\pm 1] \mp \tfrac{11}{6}\, y[x] + O\,\delta^4 \tag{3.2.19}$$

The interpolation error and the uncompensated error bias are still of order δ^4 with these substitutions. With equations 3.2.15-17 the compensated error bias is still of order δ^6 (equation 3.2.14), but is of order δ^5 with equation 3.2.18 or 3.2.19.

The $d^4 y$ term in equation 3.2.12 can also be approximated using a finite-difference derivative estimator,

$$d^4 y[x] = 6\, y[x] - 4\,(y[x+1] + y[x-1]) + y[x+2] + y[x-2] + O\,\delta^6 \tag{3.2.20}$$

Out-of-range indexing can be avoided in equation 3.2.20 by applying constant extrapolation to $d^4 y$,

$$d^4 y[x \pm 1] = d^4 y[x] + O\,\delta^5, \quad d^4 y[x \pm 2] = d^4 y[x] + O\,\delta^5 \tag{3.2.21}$$

With this extrapolation, the residual bias (equation 3.2.14) will be of order δ^5.

Each error term of the form δ^n in the above equations implicitly includes a factor of $d^n y[x]$ (from equation 3.1.5). If the higher-order derivatives of y are not sufficiently bounded, it may be advantageous to use derivative estimators of lower degree; or lower-degree estimators may be used simply to improve computational efficiency. The first derivatives in equations 3.2.1 and 3.2.11 can be approximated using a 2nd-degree centered difference estimator,

$$dy[x] = \tfrac{1}{2}(y[x+1] - y[x-1]) + O\,\delta^3 \tag{3.2.22}$$

At the sampling range limits an asymmetric difference estimator may be used,

$$dy[x] = \pm\tfrac{1}{2}\, y[x \mp 2] \mp 2\, y[x \mp 1] \pm \tfrac{3}{2}\, y[x] + O\,\delta^3 \tag{3.2.23}$$

With these derivative estimators the interpolation accuracy order is only 2 (compared to 4 with higher-order estimators, but still better than the order-1 accuracy of linear interpolation),

$$yFit^{[m]}[x] - y[x] = O\,\delta^3 \tag{3.2.24}$$

In this case the interpolation bias is not significant in comparison to the error, so there no need for bias compensation,

$$\int_m^{m+1} (yFit^{[m]}[x] - y[x])\, dx = O\,\delta^4 \tag{3.2.25}$$

4. N-dimensional methods

4.1 Multilinear interpolation

Multilinear interpolation is defined by applying one-dimensional linear interpolation in each separate coordinate dimension. The interpolation has the following form,

$$yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] = \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\, y[\boldsymbol{m}+\boldsymbol{s}] \tag{4.1.1}$$

wherein the cardinal functions $u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ are defined so that

$$yFit^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}] = y[\boldsymbol{m}+\boldsymbol{s}]; \quad s_j \in \{0,1\}; \quad j \in \{1,\ldots,N\} \tag{4.1.2}$$

This implies that the cardinal functions satisfy the following defining conditions,

$$u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}'] = \begin{cases} 1, & \boldsymbol{s}=\boldsymbol{s}' \\ 0, & \boldsymbol{s}\neq\boldsymbol{s}' \end{cases} \quad s_j,\, s'_j \in \{0,1\}; \quad j \in \{1,\ldots,N\} \tag{4.1.3}$$

$u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ is multilinear in $\boldsymbol{x}$ (i.e., linear in each coordinate x_j), and is a product of factors having the form of equation 3.1.4,

$$u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \prod_{j\in\{1,\ldots,N\}} \big((1-s_j)+(2\,s_j-1)(x_j-m_j)\big) \tag{4.1.4}$$[4]

(Note: Section A.1 in the Appendix outlines derivations of the equations in this section.)

Making substitution 2.9 ($y[\boldsymbol{x}] = f[\delta\,\boldsymbol{x}]$) and expanding the interpolation error to 2nd order in δ, the result is a series of terms having the form of equation 3.1.6,

$$\begin{aligned} &yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = \\ &\quad \sum_{j\in\{1,\ldots,N\}} \tfrac{1}{2}(x_j-m_j)(m_j+1-x_j)\,\partial_j{}^2 y[\boldsymbol{m}] + O\,\delta^3 \end{aligned} \tag{4.1.5}$$

The average error over the interpolation grid cell ($m_j \le x_j \le m_j+1$) is

$$\begin{aligned} &\int_{x_N=m_N}^{m_N+1} \cdots \int_{x_1=m_1}^{m_1+1} (yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}])\, dx_1 \ldots dx_N = \\ &\quad \sum_{j\in\{1,\ldots,N\}} \tfrac{1}{12}\,\partial_j{}^2 y[\boldsymbol{m}] + O\,\delta^3 \end{aligned} \tag{4.1.6}$$

Generalizing from equations 3.1.11 and 3.1.12, the error bias can be substantially eliminated by applying bias compensation:

---

[4] An equivalent form of equation 4.1.4 (with $m_j \le x_j \le m_j+1$) is

$$u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \prod_{j\in\{1,\ldots,N\}} \big(1-\left|x_j-m_j-s_j\right|\big)$$

$$yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] = \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N), \\ s_j\in\{0,1\}}} u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\, yComp[\boldsymbol{m}+\boldsymbol{s}] \tag{4.1.7}$$

wherein the bias-compensated function $yComp$ is defined as

$$yComp[\boldsymbol{x}] = y[\boldsymbol{x}] - \sum_{j\in\{1,\ldots,N\}} \tfrac{1}{12}\partial_j{}^2 y[\boldsymbol{x}] \tag{4.1.8}$$

With bias compensation, the interpolation error is

$$\begin{aligned} &yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = \\ &\sum_{j\in\{1,\ldots,N\}} \left(\tfrac{1}{2}(x_j - m_j)(m_j + 1 - x_j) - \tfrac{1}{12}\right)\partial_j{}^2 y[\boldsymbol{m}] + O\,\delta^3 \end{aligned} \tag{4.1.9}$$

and the grid-cell-averaged error is

$$\int_{x_N=m_N}^{m_N+1}\cdots\int_{x_1=m_1}^{m_1+1} (yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}])\, dx_1 \ldots dx_N = O\,\delta^4 \tag{4.1.10}$$

Note that both the δ^2 and the δ^3 terms in equation 4.1.10 average out to zero, leaving a residual bias of order δ^4.

The $\partial_j{}^2 y$ terms in equation 4.1.8 can be approximated using a finite-difference estimator similar to equation 3.1.15, and – to avoid out-of-range indexing – constant extrapolation of $\partial_j{}^2 y$ as in equation 3.1.16,

$$\partial_j{}^2 y[\boldsymbol{x}] = y[\boldsymbol{x} - \boldsymbol{e}_j] - 2\,y[\boldsymbol{x}] + y[\boldsymbol{x} + \boldsymbol{e}_j] + O\,\delta^4 \tag{4.1.11}$$

$$\partial_j{}^2 y[\boldsymbol{x} \pm \boldsymbol{e}_j] = \partial_j{}^2 y[\boldsymbol{x}] + O\,\delta^3 \tag{4.1.12}$$

If extrapolation is used (equation 4.1.12), the residual bias is of order δ^3,

$$\int_{x_N=m_N}^{m_N+1}\cdots\int_{x_1=m_1}^{m_1+1} (yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}])\, dx_1 \ldots dx_N = O\,\delta^3 \tag{4.1.13}$$

4.2 Multicubic interpolation

Multicubic interpolation is defined by applying one-dimensional cubic interpolation in each separate coordinate dimension. The interpolation has the following form,

$$yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] = \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\}}} \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\,k_j} \right) y[\boldsymbol{m}+\boldsymbol{s}] \qquad (4.2.1)$$

wherein the cardinal functions $u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ are defined so that

$$\begin{aligned}&\left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\,k_j} \right) yFit^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}] = \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\,k_j} \right) y[\boldsymbol{m}+\boldsymbol{s}];\\ &k_j,\, s_j \in \{0,1\}; \quad j\in\{1,\ldots,N\}\end{aligned} \qquad (4.2.2)$$

This implies that the cardinal functions satisfy the following defining conditions,

$$\begin{aligned}&\left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\,k'_j} \right) u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}'] = \begin{cases} 1, & \boldsymbol{k}=\boldsymbol{k}' \text{ and } \boldsymbol{s}=\boldsymbol{s}' \\ 0, & \boldsymbol{k}\neq\boldsymbol{k}' \text{ or } \boldsymbol{s}\neq\boldsymbol{s}' \end{cases}\\ &k_j,\, k'_j,\, s_j,\, s'_j \in \{0,1\}; \quad j\in\{1,\ldots,N\}\end{aligned} \qquad (4.2.3)$$

$u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ is multicubic in $\boldsymbol{x}$ (i.e., cubic in each coordinate x_j), and is a product of factors having the form of equation 3.2.4,

$$u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \prod_{j\in\{1,\ldots,N\}} \begin{cases} s_j + (1-2\,s_j)(1+2\,(x_j-m_j))(1+m_j-x_j)^2 & \text{if } k_j = 0 \\ (x_j-m_j)(1+m_j-s_j-x_j)(1+m_j-x_j) & \text{if } k_j = 1 \end{cases}$$

(4.2.4)[5]

(Note: Section A.2 in the Appendix outlines derivations of the equations in this section.)

Making substitution 2.9 ($y[\boldsymbol{x}] = f[\delta\, \boldsymbol{x}]$) and expanding the interpolation error to 4th order in δ, the result is a series of terms having the form of equation 3.2.5,

$$\begin{aligned}&yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] =\\ &\quad \sum_{j\in\{1,\ldots,N\}} -\tfrac{1}{24}(x_j-m_j)^2\,(m_j+1-x_j)^2\,\partial_j^{\,4} y[\boldsymbol{m}] + O\,\delta^5\end{aligned} \qquad (4.2.5)$$

The average error over the interpolation grid cell ($m_j \le x_j \le m_j+1$) is

[5] An equivalent form of equation 4.2.4 (with $m_j \le x_j \le m_j+1$) is

$$u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \prod_{j\in\{1,\ldots,N\}} \begin{cases} \left(1+2\left|x_j-m_j-s_j\right|\right)\left(1-\left|x_j-m_j-s_j\right|\right)^2 & \text{if } k_j = 0 \\ \left(x_j-m_j-s_j\right)\left(1-\left|x_j-m_j-s_j\right|\right)^2 & \text{if } k_j = 1 \end{cases}$$

$$\int_{x_N=m_N}^{m_N+1} \cdots \int_{x_1=m_1}^{m_1+1} (yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}])\, dx_1 \ldots dx_N = \sum_{j\in\{1,\ldots,N\}} -\tfrac{1}{720} \partial_j{}^4 y[\boldsymbol{m}] + O\,\delta^5 \tag{4.2.6}$$

Generalizing from equations 3.2.11 and 3.2.12, the error bias can be substantially eliminated by applying bias compensation:

$$yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] = \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\}}} \; \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] \left(\prod_{j\in\{1,\ldots,N\}} \partial_j{}^{k_j} \right) yComp[\boldsymbol{m}+\boldsymbol{s}] \tag{4.2.7}$$

wherein the bias-compensated function $yComp$ is defined as

$$yComp[\boldsymbol{x}] = y[\boldsymbol{x}] + \sum_{j\in\{1,\ldots,N\}} \tfrac{1}{720} \partial_j{}^4 y[\boldsymbol{x}] \tag{4.2.8}$$

With bias compensation, the interpolation error is

$$yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = \sum_{j\in\{1,\ldots,N\}} \left(-\tfrac{1}{24} (x_j - m_j)^2\, (m_j + 1 - x_j)^2 + \tfrac{1}{720} \right) \partial_j{}^4 y[\boldsymbol{m}] + O\,\delta^5 \tag{4.2.9}$$

and the grid-cell-averaged error is

$$\int_{x_N=m_N}^{m_N+1} \cdots \int_{x_1=m_1}^{m_1+1} (yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}])\, dx_1 \ldots dx_N = O\,\delta^6 \tag{4.2.10}$$

Note that both the δ^4 and the δ^5 terms in equation 4.2.10 average out to zero, leaving a residual bias of order δ^6.

Many of the terms in equation 4.2.1 are superfluous because they are dominated by the interpolation error. Each derivative operator ∂_j, applied to y, implicitly includes a factor of δ (from equation 2.11), so the factor $\left(\prod_j \partial_j{}^{k_j}\right) y$ is of order $\delta^{\sum \boldsymbol{k}}$ (cf. equation 2.11). Thus, all terms in equation 4.2.1 with $\sum \boldsymbol{k} > 3$ can be omitted without reducing accuracy order.

Equation 4.2.1 will be generalized as follows to include only terms up to order L in δ,

$$yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] = \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum\boldsymbol{k}\le L}} \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] \left(\prod_{j\in\{1,\ldots,N\}} \partial_j{}^{k_j}\right) y[\boldsymbol{m}+\boldsymbol{s}] \qquad (4.2.11)$$

(This is the same as equation 4.2.1, except that the $\boldsymbol{k}$ sum is truncated to exclude terms with $\sum\boldsymbol{k} > L$.) With bias compensation, order truncation is applied by modifying equation 4.2.7 in a similar manner,

$$yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] = \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum\boldsymbol{k}\le L}} \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] \left(\prod_{j\in\{1,\ldots,N\}} \partial_j{}^{k_j}\right) yComp[\boldsymbol{m}+\boldsymbol{s}] \quad (4.2.12)$$

If $L \ge N$ equations 4.2.11 and 4.2.12 are identical to equations 4.2.1 and 4.2.7, respectively, and the total number of summation terms is 4^N. In general, the number of summation terms is

$$\text{term count} = 2^N \sum_{l\in\{0,\ldots,L\}} \frac{N!}{l!(N-l)!} \qquad (4.2.13)$$

For example, with $L = 3$ the interpolation accuracy is still 3rd order, but the number of terms is reduced from 4^N to $2^N\,(1 + N\,(5 + N^2)/6)$. With $L = 2$ the accuracy is only 2nd order (still better than the 1st-order accuracy of multilinear interpolation), but the number of summation terms is reduced to $2^N\,(1 + N\,(1+N)/2)$.

The interpolation error is described by equation 4.2.5 if $L > 3$. If $L = 3$ the error estimate, without bias compensation, is

$$\begin{aligned} &L = 3: \\ &\quad yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = \\ &\qquad \sum_{j\in\{1,\ldots,N\}} -\tfrac{1}{24}(x_j - m_j)^2\,(m_j + 1 - x_j)^2\,\partial_j{}^4 y[\boldsymbol{m}] \qquad (4.2.14) \\ &\qquad - \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum\boldsymbol{k}=4}} \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] \left(\prod_{j\in\{1,\ldots,N\}} \partial_j{}^{k_j}\right) y[\boldsymbol{m}+\boldsymbol{s}] \\ &\qquad + O\,\delta^5 \end{aligned}$$

With bias compensation, an extra factor of $\frac{1}{720}$ appears in the first sum,

$$
\begin{aligned}
&L = 3: \\
&\quad yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = \\
&\qquad \sum_{j \in \{1,\ldots,N\}} \left(-\tfrac{1}{24} (x_j - m_j)^2 \, (m_j + 1 - x_j)^2 + \tfrac{1}{720} \right) \partial_j{}^4 y[\boldsymbol{m}] \\
&\qquad - \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N), \\ k_j \in \{0,1\}, \\ \sum \boldsymbol{k} = 4}} \; \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N), \\ s_j \in \{0,1\}}} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] \left(\prod_{j \in \{1,\ldots,N\}} \partial_j{}^{k_j} \right) y[\boldsymbol{m} + \boldsymbol{s}] \\
&\qquad + O\,\delta^5
\end{aligned}
\tag{4.2.15}
$$

The second summation in these equations, representing the lowest-order truncated terms, is of order δ^4, and hence does not affect the accuracy order. The error bias is also still of order δ^4 without bias compensation (equation 4.2.6) and of order δ^6 with bias compensation (equation 4.2.10). (The latter result will be demonstrated in the Appendix, Section A.2.)

If $L = 2$ the interpolation error is 3rd order in δ,

$$
\begin{aligned}
&L = 2: \\
&\quad yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = \\
&\qquad - \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N), \\ k_j \in \{0,1\}, \\ \sum \boldsymbol{k} = 3}} \; \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N), \\ s_j \in \{0,1\}}} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] \left(\prod_{j \in \{1,\ldots,N\}} \partial_j{}^{k_j} \right) y[\boldsymbol{m} + \boldsymbol{s}] \\
&\qquad + O\,\delta^4
\end{aligned}
\tag{4.2.16}
$$

In this case the error bias is of order δ^4, which is not significant in relation to the order-δ^3 error, so no bias compensation is required,

$$
\begin{aligned}
&L = 2: \\
&\quad \int_{x_N = m_N}^{m_N + 1} \cdots \int_{x_1 = m_1}^{m_1 + 1} (yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}]) \, dx_1 \ldots dx_N = O\,\delta^4
\end{aligned}
\tag{4.2.17}
$$

The derivatives in equations 4.2.11 and 4.2.12 can be estimated using finite difference estimators. For $\sum \boldsymbol{k} = 1$, with $k_j = 1$ and all other components of $\boldsymbol{k}$ being zero, one of the following 4th –degree Savitzky-Golay derivative estimators can be used (from equations 3.2.15-17),

$$
\begin{aligned}
&\partial_j y[\boldsymbol{x}] = \tfrac{2}{3} (y[\boldsymbol{x} + \boldsymbol{e}_j] - y[\boldsymbol{x} - \boldsymbol{e}_j]) - \tfrac{1}{12} (y[\boldsymbol{x} + 2\boldsymbol{e}_j] - y[\boldsymbol{x} - 2\boldsymbol{e}_j]) \\
&\quad + O\,\delta^5
\end{aligned}
\tag{4.2.18}
$$

$$\partial_j y[\boldsymbol{x}] = \pm\tfrac{1}{12}\, y[\boldsymbol{x} \pm 3\boldsymbol{e}_j] \mp \tfrac{1}{2}\, y[\boldsymbol{x} \pm 2\boldsymbol{e}_j] \pm \tfrac{3}{2}\, y[\boldsymbol{x} \pm \boldsymbol{e}_j] \mp \tfrac{5}{6}\, y[\boldsymbol{x}] \mp \tfrac{1}{4}\, y[\boldsymbol{x} \mp \boldsymbol{e}_j] + O\,\delta^5 \tag{4.2.19}$$

$$\partial_j y[\boldsymbol{x}] = \mp\tfrac{1}{4}\, y[\boldsymbol{x} \pm 4\boldsymbol{e}_j] \pm \tfrac{4}{3}\, y[\boldsymbol{x} \pm 3\boldsymbol{e}_j] \mp 3\, y[\boldsymbol{x} \pm 2\boldsymbol{e}_j] \pm 4\, y[\boldsymbol{x} \pm \boldsymbol{e}_j] \mp \tfrac{25}{12}\, y[\boldsymbol{x}] + O\,\delta^5 \tag{4.2.20}$$

(With these approximations the compensated error bias, equation 4.2.10, is still of order δ^6.) If there are only four sampling points for $\boldsymbol{x}_j$, the following derivative estimators may be used (from equations 3.2.18-19),

$$\partial_j y[\boldsymbol{x}] = \mp\tfrac{1}{6}\, y[\boldsymbol{x} \pm 2\boldsymbol{e}_j] \pm y[\boldsymbol{x} \pm \boldsymbol{e}_j] \mp \tfrac{1}{2}\, y[\boldsymbol{x}] \mp \tfrac{1}{3}\, y[\boldsymbol{x} \mp \boldsymbol{e}_j] + O\,\delta^4 \tag{4.2.21}$$

$$\partial_j y[\boldsymbol{x}] = \pm\tfrac{1}{3}\, y[\boldsymbol{x} \pm 3\boldsymbol{e}_j] \mp \tfrac{3}{2}\, y[\boldsymbol{x} \pm 2\boldsymbol{e}_j] \pm 3\, y[\boldsymbol{x} \pm \boldsymbol{e}_j] \mp \tfrac{11}{6}\, y[\boldsymbol{x}] + O\,\delta^4 \tag{4.2.22}$$

(If equation 4.2.21 or 4.2.22 is used the compensated error bias is of order δ^5.)

For $\sum \boldsymbol{k} = 2$, with $k_{j_1} = 1$ and $k_{j_2} = 1$, the mixed derivative $\partial_{j_1}\partial_{j_2} y[\boldsymbol{x}]$ in equations 4.2.11 and 4.2.12 can be estimated by applying two 2nd-degree derivative estimators, each having one of the following forms,

$$\partial_j y[\boldsymbol{x}] = \tfrac{1}{2}(y[\boldsymbol{x} + \boldsymbol{e}_j] - y[\boldsymbol{x} - \boldsymbol{e}_j]) + O\,\delta^3 \tag{4.2.23}$$

$$\partial_j y[\boldsymbol{x}] = \mp\tfrac{1}{2}\, y[\boldsymbol{x} \pm 2\boldsymbol{e}_j] \pm 2\, y[\boldsymbol{x} \pm \boldsymbol{e}_j] \mp \tfrac{3}{2}\, y[\boldsymbol{x}] + O\,\delta^3 \tag{4.2.24}$$

(from equations 3.2.22 and 3.2.23). The combination of two such estimators has a residual error of order δ^4. For example, using equation 4.2.23 for both derivatives, the following estimate is obtained,

$$\partial_{j_1}\partial_{j_2} y[\boldsymbol{x}] = \frac{1}{4}\begin{pmatrix} y[\boldsymbol{x} + \boldsymbol{e}_{j_1} + \boldsymbol{e}_{j_2}] - y[\boldsymbol{x} - \boldsymbol{e}_{j_1} + \boldsymbol{e}_{j_2}] \\ - y[\boldsymbol{x} + \boldsymbol{e}_{j_1} - \boldsymbol{e}_{j_2}] + y[\boldsymbol{x} - \boldsymbol{e}_{j_1} - \boldsymbol{e}_{j_2}] \end{pmatrix} + O\,\delta^4 \tag{4.2.25}$$

The 2nd-degree estimators can also be used for $\sum \boldsymbol{k} = 3$, e.g.,

$$\partial_{j_1}\partial_{j_2}\partial_{j_3} y[\boldsymbol{x}] = \frac{1}{8}\left(\begin{array}{l} y[\boldsymbol{x}+\boldsymbol{e}_{j_1}+\boldsymbol{e}_{j_2}+\boldsymbol{e}_{j_3}] - y[\boldsymbol{x}-\boldsymbol{e}_{j_1}+\boldsymbol{e}_{j_2}+\boldsymbol{e}_{j_3}] \\ \quad - y[\boldsymbol{x}+\boldsymbol{e}_{j_1}-\boldsymbol{e}_{j_2}+\boldsymbol{e}_{j_3}] + y[\boldsymbol{x}-\boldsymbol{e}_{j_1}-\boldsymbol{e}_{j_2}+\boldsymbol{e}_{j_3}] \\ \quad - y[\boldsymbol{x}+\boldsymbol{e}_{j_1}+\boldsymbol{e}_{j_2}-\boldsymbol{e}_{j_3}] + y[\boldsymbol{x}-\boldsymbol{e}_{j_1}+\boldsymbol{e}_{j_2}-\boldsymbol{e}_{j_3}] \\ \quad + y[\boldsymbol{x}+\boldsymbol{e}_{j_1}-\boldsymbol{e}_{j_2}-\boldsymbol{e}_{j_3}] - y[\boldsymbol{x}-\boldsymbol{e}_{j_1}-\boldsymbol{e}_{j_2}-\boldsymbol{e}_{j_3}] \end{array}\right) + O\,\delta^5 \tag{4.2.26}$$

(None of the above mixed-derivative estimators affects the order of the error bias.) Equations 4.2.11 and 4.2.12 contain no mixed derivatives with $\sum \boldsymbol{k} > 3$.

The $\partial_j{}^4 y$ term in equation 4.2.8 can also be approximated using a finite-difference derivative estimator,

$$\partial_j{}^4 y[\boldsymbol{x}] = 6\,y[\boldsymbol{x}] - 4\,(y[\boldsymbol{x}+\boldsymbol{e}_j] + y[\boldsymbol{x}-\boldsymbol{e}_j]) + y[\boldsymbol{x}+2\,\boldsymbol{e}_j] + y[\boldsymbol{x}-2\,\boldsymbol{e}_j] + O\,\delta^6 \tag{4.2.27}$$

(cf. equation 3.2.20). Out-of-range indexing can be avoided in equation 4.2.27 by applying constant extrapolation to $\partial_j{}^4 y$,

$$\partial_j{}^4 y[\boldsymbol{x} \pm \boldsymbol{e}_j] = \partial_j{}^4 y[\boldsymbol{x}] + O\,\delta^5, \quad \partial_j{}^4 y[\boldsymbol{x} \pm 2\,\boldsymbol{e}_j] = \partial_j{}^4 y[\boldsymbol{x}] + O\,\delta^5 \tag{4.2.28}$$

(The compensated error bias will be of order δ^5 if equation 4.2.28 is applied.)

If the higher-order derivatives of y are not sufficiently bounded, it may be advantageous to use derivative estimators of lower degree; or lower-degree estimators may be used simply to improve computational efficiency. The first derivatives in equations 4.2.11 and 4.2.12 can be approximated using equation 4.2.23 or 4.2.24 when $\sum \boldsymbol{k} = 1$. In this case the interpolation accuracy is only order 2,

$$yFit^{[m]}[\boldsymbol{x}] - y[\boldsymbol{x}] = O\,\delta^3 \tag{4.2.29}$$

so this approximation can be used with $L = 2$ in equation 4.2.11.

4.3 Reduced-cubic interpolation

Reduced-cubic interpolation is similar to equation 4.2.1, except that the mixed derivatives of y are not used; only the gradient terms are used. The interpolation has the form,

$$yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] = \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} \left(u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\, y[\boldsymbol{m}+\boldsymbol{s}] + \sum_{k\in\{1,\ldots,N\}} v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\, \partial_k y[\boldsymbol{m}+\boldsymbol{s}] \right) \quad (4.3.1)$$

wherein the cardinal functions $u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ and $v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ are defined so that

$$\left.\begin{array}{l} yFit^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}] = y[\boldsymbol{m}+\boldsymbol{s}] \\ \partial_k yFit^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}] = \partial_k y[\boldsymbol{m}+\boldsymbol{s}] \end{array}\right\} \quad s_j \in \{0,1\}; \quad k, j \in \{1,\ldots,N\} \quad (4.3.2)$$

This implies that the cardinal functions satisfy the following defining conditions,

$$\left.\begin{array}{l} u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}'] = \begin{cases} 1, & \boldsymbol{s}=\boldsymbol{s}' \\ 0, & \boldsymbol{s}\neq\boldsymbol{s}' \end{cases} \\ \partial_i u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}'] = 0 \\ v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}'] = 0 \\ \partial_i v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{m}+\boldsymbol{s}'] = \begin{cases} 1, & \boldsymbol{s}=\boldsymbol{s}' \text{ and } k=i \\ 0, & \boldsymbol{s}\neq\boldsymbol{s}' \text{ or } k\neq i \end{cases} \end{array}\right\} \quad s_j, s'_j \in \{0,1\}; \quad j,k,i \in \{1,\ldots,N\}$$

$$(4.3.3)$$

The following functions satisfy the above conditions,

$$\begin{array}{l} u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \left(\prod_{j\in\{1,\ldots,N\}} \big((1-s_j)+(2s_j-1)(x_j-m_j)\big) \right) \\ \quad \left(1 + \sum_{k\in\{1,\ldots,N\}} (s_k-(x_k-m_k))(2(x_k-m_k)-1) \right) \end{array} \quad (4.3.4)^{6}$$

$$\begin{array}{l} v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \left(\prod_{j\in\{1,\ldots,N\}} \big((1-s_j)+(2s_j-1)(x_j-m_j)\big) \right) \\ \quad (2s_k-1)(x_k-m_k)(x_k-m_k-1) \end{array} \quad (4.3.5)^{7}$$

---

[6] An equivalent form of equation 4.3.4 (with $m_j \le x_j \le m_j+1$) is

$$u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \left(\prod_{j\in\{1,\ldots,N\}} \big(1-|x_j-m_j-s_j|\big) \right) \left(1 + \sum_{k\in\{1,\ldots,N\}} |x_k-m_k-s_k| \big(1-2|x_k-m_k-s_k|\big) \right)$$

[7] An equivalent form of equation 4.3.5 (with $m_j \le x_j \le m_j+1$) is

$$v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \left(\prod_{j\in\{1,\ldots,N\}} \big(1-|x_j-m_j-s_j|\big) \right) (x_k-m_k-s_k)\big(1-|x_k-m_k-s_k|\big)$$

(Note: Section A.3 in the Appendix outlines derivations of the equations in this section.)

The reduced-cubic interpolation function is continuous across the grid cell boundaries, but as in the case of multilinear interpolation the gradient is not continuous. However, the gradient is continuous at the sampling grid points, and the discontinuity of the $yFit$ gradient between grid cells is only of order δ^4. By contrast, the gradient discontinuity with multilinear interpolation is of order δ^2. (The function f defined by equation 2.9, as estimated by the interpolation function, has gradient discontinuities of order δ^3 or δ using reduced-cubic or multilinear interpolation, respectively.)

Making substitution 2.9 ($y[\boldsymbol{x}] = f[\delta\,\boldsymbol{x}]$) in equation 4.3.1 and expanding the interpolation error to 4th order in δ, the result is

$$
\begin{aligned}
&yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = \\
&\quad \sum_{j\in\{1,\ldots,N\}} -\tfrac{1}{24}(x_j - m_j)^2\,(m_j + 1 - x_j)^2\,\partial_j{}^4 y[\boldsymbol{m}] \\
&\quad + \sum_{j_1\in\{1,\ldots,N-1\}}\ \sum_{j_2\in\{j_1+1,\ldots,N\}} \begin{pmatrix} -\tfrac{1}{4}(x_{j_2} - m_{j_2})(m_{j_2} + 1 - x_{j_2})(x_{j_1} - m_{j_1})(m_{j_1} + 1 - x_{j_1}) \\ \partial_{j_2}{}^2\partial_{j_1}{}^2 y[\boldsymbol{m}] \end{pmatrix} \\
&\quad + O\,\delta^5
\end{aligned}
\tag{4.3.6}
$$

The interpolation error is of order δ^4, the same as multicubic interpolation (cf. equations 4.2.5 and 4.2.14), even though the interpolation function (equation 4.3.1) has only $2^N\,(1+N)$ terms (compared to the 4^N terms of full multicubic, or the $2^N\,(1+N\,(5+N^2)/6)$ terms for truncated multicubic with $L=3$, cf. equation 4.2.13).

The average error over the interpolation grid cell ($m_j \le x_j \le m_j + 1$) is

$$
\begin{aligned}
&\int_{x_N=m_N}^{m_N+1}\cdots\int_{x_1=m_1}^{m_1+1} (yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}])\,dx_1\ldots dx_N = \\
&\quad \sum_{j\in\{1,\ldots,N\}} -\tfrac{1}{720}\partial_j{}^4 y[\boldsymbol{m}] + \sum_{j_1\in\{1,\ldots,N-1\}}\ \sum_{j_2\in\{j_1+1,\ldots,N\}} -\tfrac{1}{144}\partial_{j_2}{}^2\partial_{j_1}{}^2 y[\boldsymbol{m}] + O\,\delta^5
\end{aligned}
\tag{4.3.7}
$$

The error bias can be substantially eliminated by applying bias compensation:

$$
yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] = \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} \left(u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\,yComp[\boldsymbol{m}+\boldsymbol{s}] + \sum_{k\in\{1,\ldots,N\}} v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\,\partial_k yComp[\boldsymbol{m}+\boldsymbol{s}] \right)
\tag{4.3.8}
$$

wherein the bias-compensated function $yComp$ is defined as

$$yComp[\boldsymbol{x}] = y[\boldsymbol{x}] + \sum_{j\in\{1,\ldots,N\}} \tfrac{1}{720} \partial_j{}^4 y[\boldsymbol{x}] \quad + \sum_{j_1\in\{1,\ldots,N-1\}} \; \sum_{j_2\in\{j_1+1,\ldots,N\}} \tfrac{1}{144} \partial_{j_2}{}^2 \partial_{j_1}{}^2 y[\boldsymbol{x}] \tag{4.3.9}$$

With bias compensation, the interpolation error is

$$\begin{aligned} & yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}] = \\ & \quad \sum_{j\in\{1,\ldots,N\}} \left(-\tfrac{1}{24}(x_j - m_j)^2 (m_j + 1 - x_j)^2 + \tfrac{1}{720} \right) \partial_j{}^4 y[\boldsymbol{m}] \\ & \quad + \sum_{j_1\in\{1,\ldots,N-1\}} \; \sum_{j_2\in\{j_1+1,\ldots,N\}} \begin{pmatrix} \left(-\tfrac{1}{4}(x_{j_2} - m_{j_2})(m_{j_2} + 1 - x_{j_2})(x_{j_1} - m_{j_1})(m_{j_1} + 1 - x_{j_1}) + \tfrac{1}{144} \right) \\ \partial_{j_2}{}^2 \partial_{j_1}{}^2 y[\boldsymbol{m}] \end{pmatrix} \\ & \quad + O\,\delta^5 \end{aligned} \tag{4.3.10}$$

and the grid-cell-averaged error is

$$\int_{x_N = m_N}^{m_N+1} \cdots \int_{x_1 = m_1}^{m_1+1} (yFit'^{[\boldsymbol{m}]}[\boldsymbol{x}] - y[\boldsymbol{x}])\, dx_1 \ldots dx_N = O\,\delta^6 \tag{4.3.11}$$

Note that both the δ^4 and the δ^5 terms in equation 4.3.11 average out to zero, leaving a residual bias of order δ^6.

The derivatives in equations 4.3.1 and 4.3.8 can be estimated using equations 4.2.18-22. The $\partial_j{}^4 y$ term in equation 4.3.9 can also be approximated using equations 4.2.27 and 4.2.28. The mixed derivative $\partial_{j_2}{}^2 \partial_{j_1}{}^2 y[\boldsymbol{x}]$ can be estimated by applying two 2nd-degree derivative estimators of the form given by equations 4.1.11 and 4.1.12.

If the higher-order derivatives of y are not sufficiently bounded, it may be advantageous to use derivative estimators of lower degree; or lower-degree estimators may be used simply to improve computational efficiency. The derivatives in equations 4.3.1 and 4.3.8 can be approximated using equation 4.2.23 or 4.2.24. In this case the interpolation accuracy is only order 2, as in equation 4.2.29.

Appendix. Derivations

A.1 Multilinear interpolation

The equations in Section 4.1 can be derived by induction on N. We define $yFit^{[\boldsymbol{m},n]}[\boldsymbol{x}]$ to be the interpolation function obtained by interpolating $y[\boldsymbol{x}]$ on dimensions $1,\ldots,n$; and we define

$$yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] = yFit^{[\boldsymbol{m},N]}[\boldsymbol{x}] \tag{A.1.1}$$

The induction conditions are

$$yFit^{[\boldsymbol{m},0]}[\boldsymbol{x}] = y[\boldsymbol{x}] \tag{A.1.2}$$

$$\begin{aligned} yFit^{[\boldsymbol{m},n]}[\boldsymbol{x}] = {} & u_0^{[\boldsymbol{m},n]}[\boldsymbol{x}]\, yFit^{[\boldsymbol{m},n-1]}[x_1,\ldots,x_{n-1},m_n,x_{n+1},\ldots,x_N] + \\ & u_1^{[\boldsymbol{m},n]}[\boldsymbol{x}]\, yFit^{[\boldsymbol{m},n-1]}[x_1,\ldots,x_{n-1},m_n+1,x_{n+1},\ldots,x_N] \end{aligned} \tag{A.1.3}$$

wherein

$$u_s^{[\boldsymbol{m},n]}[\boldsymbol{x}] = (1-s) + (x_n - m_n)(2s-1) \quad (s \in \{0,1\}) \tag{A.1.4}$$

(Equations A.1.3 and A.1.4 are obtained by applying equations 3.1.1 and 3.1.4 to dimension n.) The following expression satisfies these conditions,

$$\begin{aligned} & yFit^{[\boldsymbol{m},n]}[\boldsymbol{x}] = \\ & \quad \sum_{s_1,s_2,\ldots s_n \in \{0,1\}} \left(\prod_{j \in \{1,\ldots,n\}} u_{s_j}^{[\boldsymbol{m},j]}[\boldsymbol{x}] \right) y[m_1+s_1,\ldots,m_n+s_n,x_{n+1},\ldots,x_N] \end{aligned} \tag{A.1.5}$$

For $n = N$, the right side of equation A.1.5 matches equation 4.1.1, with

$$u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \prod_{j \in \{1,\ldots,N\}} u_{s_j}^{[\boldsymbol{m},j]}[\boldsymbol{x}] \tag{A.1.6}$$

(This is equivalent to equation 4.1.4.)

The interpolation error and error bias can be estimated by using a truncated Taylor series for $y[\boldsymbol{x}]$ (equation 2.10). The series comprises monomial terms of the form $\prod_j (x_j - m_j)^{k_j}$, which are multiplicatively separable in the x_j coordinates. The interpolation function (equation 4.1.1) corresponding to any such monomial is similarly separable,

$$y[\boldsymbol{x}] = \prod_{j \in \{1,\ldots,N\}} (x_j - m_j)^{k_j} \quad \to$$

$$yFit[\boldsymbol{x}] = \prod_{j \in \{1,\ldots,N\}} \sum_{s_j \in \{0,1\}} \big((1 - s_j) + (2 s_j - 1)(x_j - m_j)\big) s_j^{\,k_j} \qquad \text{(A.1.7)}$$

$$= \prod_{j \in \{1,\ldots,N\}} (x_j - m_j)^{\min[k_j, 1]}$$

An interpolator’s “invariant monomials” are the monomials whose associated interpolation error is zero. For multilinear interpolation, it is evident from equation A.1.7 that the invariant monomials are multilinear (i.e., $k_j \le 1$ for all j).

The Taylor series for a general function $y[\boldsymbol{x}]$ can be partitioned into partial series, each comprising monomials of a particular order:

$$y[\boldsymbol{x}] = \sum_{k_1,\ldots,k_N \in \{0,1,2,\ldots\}} \left(\left(\prod_{j \in \{1,\ldots,N\}} \partial_j^{\,k_j} \right) y[\boldsymbol{m}] \right) \prod_{j \in \{1,\ldots,N\}} \frac{(x_j - m_j)^{k_j}}{k_j!}$$

$$= \sum_{\substack{k_1,\ldots,k_N \in \{0,1,2,\ldots\}; \\ \sum \boldsymbol{k} = 0}} \cdots \; + \sum_{\substack{k_1,\ldots,k_N \in \{0,1,2,\ldots\}; \\ \sum \boldsymbol{k} = 1}} \cdots \; + \sum_{\substack{k_1,\ldots,k_N \in \{0,1,2,\ldots\}; \\ \sum \boldsymbol{k} = 2}} \cdots \; + \sum_{\substack{k_1,\ldots,k_N \in \{0,1,2,\ldots\}; \\ \sum \boldsymbol{k} = 3}} \cdots$$

$$+ O\,\delta^4$$

(A.1.8)

wherein each of the sums after the second “=” covers terms of monomial order $\sum \boldsymbol{k}$, and the summands (“ $\cdots$ ”) are all the same as the first summand. (With substitution of equaltion 2.9, $y[\boldsymbol{x}] = f[\delta\,\boldsymbol{x}]$, each summand includes an implicit factor of $\delta^{\sum \boldsymbol{k}}$.) The sum over $\sum \boldsymbol{k} = 0$ is just $y[\boldsymbol{m}]$,

$$\sum_{\substack{k_1,\ldots,k_N \in \{0,1,2,\ldots\}; \\ \sum \boldsymbol{k} = 0}} \cdots = y[\boldsymbol{m}] \qquad \text{(A.1.9)}$$

In each term of the sum over $\sum \boldsymbol{k} = 1$, one of the k_j ’s is 1 and the others are all zero. Denoting the subscript of the non-zero k_j as j_1, this sum takes the form

$$\sum_{\substack{k_1,\ldots,k_N \in \{0,1,2,\ldots\}; \\ \sum \boldsymbol{k} = 1}} \cdots = \sum_{j_1 \in \{1,\ldots,N\}} (\partial_{j_1} y[\boldsymbol{m}])(x_{j_1} - m_{j_1}) \qquad \text{(A.1.10)}$$

In the sum over $\sum \boldsymbol{k} = 2$ there are two possibilities: Two distinct k_j ’s are 1 and all others are zero, or one of the k_j ’s is 2 and the others are all zero. In the former case, the subscripts of the first and second non-zero k_j 's will be denoted respectively as j_1 and j_2, and in the latter case the subscript of the non-zero k_j will be denoted as j_1,

$$\sum_{\substack{k_1,\ldots,k_N\in\{0,1,2,\ldots\};\\ \sum\boldsymbol{k}=2}} \cdots = \sum_{\substack{j_1,j_2\in\{1,\ldots,N\};\\ j_1<j_2}} (\partial_{j_1}\partial_{j_2}\, y[\boldsymbol{m}])(x_{j_1}-m_{j_1})(x_{j_2}-m_{j_2}) + \sum_{j_1\in\{1,\ldots,N\}} (\partial_{j_1}{}^2\, y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})^2}{2!} \tag{A.1.11}$$

With $\sum\boldsymbol{k}=3$ there are three possibilities: There are three non-zero k_j 's, all equal to 1; there are two non-zero k_j 's, one equal to 1 and the other equal to 2; or there is one non-zero k_j, which is equal to 3,

$$\begin{aligned}\sum_{\substack{k_1,\ldots,k_N\in\{0,1,2,\ldots\};\\ \sum\boldsymbol{k}=3}} \cdots = {} & \sum_{\substack{j_1,j_2,j_3\in\{1,\ldots,N\};\\ j_1<j_2<j_3}} (\partial_{j_1}\partial_{j_2}\partial_{j_3}\, y[\boldsymbol{m}])(x_{j_1}-m_{j_1})(x_{j_2}-m_{j_2})(x_{j_3}-m_{j_3}) \\ & + \sum_{\substack{j_1,j_2\in\{1,\ldots,N\};\\ j_1\neq j_2}} (\partial_{j_1}\partial_{j_2}{}^2\, y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})(x_{j_2}-m_{j_2})^2}{2!} \\ & + \sum_{j_1\in\{1,\ldots,N\}} (\partial_{j_1}{}^3\, y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})^3}{3!}\end{aligned} \tag{A.1.12}$$

The interpolation error for a general $y[\boldsymbol{x}]$ are estimated by summing the errors associated with each of the monomial terms in equations A.1.9-12. The monomial error calculations can be simplified by considering the special case where $\boldsymbol{m}=\boldsymbol{0}$ (i.e., $m_j=0$ for all j) and the x_j coordinate ordering is chosen so that $j_1=1$, $j_2=2$ and $j_3=3$. Table A.1.1 tabulates the $y[\boldsymbol{x}]$ functions corresponding to such monomials up to order 2, and shows the associated interpolation error and error bias for each monomial. Equations 4.1.5 and 4.1.6 are obtained from Table A.1.1 by substituting $x_{j_k}-m_{j_k}$ for x_k in each monomial (k = 1 or 2), and combining the monomials in linear combinations having the form of equations A.1.9-11.

Table A.1.1. Multilinear interpolation error $\left(yFit^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}]\right)$ and error bias $\left(\int_{x_N=0}^{1}\cdots\int_{x_1=0}^{1}(yFit^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}])\,dx_1\ldots dx_N\right)$ for monomial functions up to order 2 (based on equation 4.1.1).			
order	$y[\boldsymbol{x}]$	error	bias
0	1	0	0
1	x_1	0	0
2	$x_1\,x_2$	0	0
	$x_1{}^2$	$x_1\,(1-x_1)$	$\frac{1}{6}$

With bias compensation, the monomial functions up to order 3 are tabulated in Table A.1.2 along with each monomial's bias-compensated form ($yComp[\boldsymbol{x}]$, equation 4.1.8) and its interpolation error and bias. Equations 4.1.9 and 4.1.10 are obtained from this tabulation. (Fourth-order monomials such as $x_1{}^2\,x_2{}^2$ and $x_1{}^4$ would exhibit a non-zero error bias, so the bias would generally be of order δ^4.)

Table A.1.2. Multilinear interpolation error $\left(yFit'^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}]\right)$ and error bias $\left(\int_{x_N=0}^{1}\cdots\int_{x_1=0}^{1}(yFit'^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}])\,dx_1\ldots dx_N\right)$ for monomial functions up to order 3, with bias compensation (based on equation 4.1.7).				
order	$y[\boldsymbol{x}]$	$yComp[\boldsymbol{x}]$	error	bias
0	1	1	0	0
1	x_1	x_1	0	0
2	$x_1\,x_2$	$x_1\,x_2$	0	0
	$x_1{}^2$	$x_1{}^2-\frac{1}{6}$	$x_1\,(1-x_1)-\frac{1}{6}$	0
3	$x_1\,x_2\,x_3$	$x_1\,x_2\,x_3$	0	0
	$x_1\,x_2{}^2$	$x_1\,(x_2{}^2-\frac{1}{6})$	$x_1\,(x_2\,(1-x_2)-\frac{1}{6})$	0
	$x_1{}^3$	$x_1\,(x_1{}^2-\frac{1}{2})$	$x_1\,(\frac{1}{2}-x_1{}^2)$	0

A.2 Multicubic interpolation

The equations in Section 4.2 can be derived by induction on N. We define $yFit^{[\boldsymbol{m},n]}[\boldsymbol{x}]$ to be the interpolation function obtained by interpolating $y[\boldsymbol{x}]$ on dimensions $1,\ldots,n$; and we define

$$yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]=yFit^{[\boldsymbol{m},N]}[\boldsymbol{x}] \tag{A.2.1}$$

The induction conditions are

$$yFit^{[\boldsymbol{m},0]}[\boldsymbol{x}] = y[\boldsymbol{x}] \tag{A.2.2}$$

$$\begin{aligned}
&yFit^{[\boldsymbol{m},n]}[\boldsymbol{x}] = \\
&\quad u_{0,0}^{[\boldsymbol{m},n]}[\boldsymbol{x}]\, yFit^{[\boldsymbol{m},n-1]}[x_1,\ldots,x_{n-1},m_n,x_{n+1},\ldots,x_N] + \\
&\quad u_{0,1}^{[\boldsymbol{m},n]}[\boldsymbol{x}]\, yFit^{[\boldsymbol{m},n-1]}[x_1,\ldots,x_{n-1},m_n+1,x_{n+1},\ldots,x_N] + \\
&\quad u_{1,0}^{[\boldsymbol{m},n]}[\boldsymbol{x}]\, \partial_n yFit^{[\boldsymbol{m},n-1]}[x_1,\ldots,x_{n-1},m_n,x_{n+1},\ldots,x_N] + \\
&\quad u_{1,1}^{[\boldsymbol{m},n]}[\boldsymbol{x}]\, \partial_n yFit^{[\boldsymbol{m},n-1]}[x_1,\ldots,x_{n-1},m_n+1,x_{n+1},\ldots,x_N]
\end{aligned} \tag{A.2.3}$$

wherein

$$\left.\begin{aligned}
u_{0,s}^{[\boldsymbol{m},n]}[\boldsymbol{x}] &= s + (1-2s)(1+2(x_n-m_n))(1+m_n-x_n)^2 \\
u_{1,s}^{[\boldsymbol{m},n]}[\boldsymbol{x}] &= (x_n-m_n)(1+m_n-s_n-x_n)(1+m_n-x_n)
\end{aligned}\right\} \quad (s \in \{0,1\}) \tag{A.2.4}$$

(Equations A.2.3 and A.2.4 are obtained by applying equations 3.2.1 and 3.2.4 to dimension n.) The following expression satisfies these conditions,

$$yFit^{[\boldsymbol{m},n]}[\boldsymbol{x}] = \sum_{k_1,k_2,\ldots k_n \in \{0,1\}} \; \sum_{s_1,s_2,\ldots s_n \in \{0,1\}} \left(\prod_{j\in\{1,\ldots,n\}} u_{k_j,s_j}^{[\boldsymbol{m},j]}[\boldsymbol{x}] \right) \left(\prod_{j\in\{1,\ldots,n\}} \partial_j^{\,k_j} \right) y[m_1+s_1,\ldots,m_n+s_n,x_{n+1},\ldots,x_N] \tag{A.2.5}$$

For $n = N$, the right side of equation A.2.5 matches equation 4.2.1, with

$$u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}] = \prod_{j\in\{1,\ldots,n\}} u_{k_j,s_j}^{[\boldsymbol{m},j]}[\boldsymbol{x}] \tag{A.2.6}$$

(This is equivalent to equation 4.2.4.)

The interpolation function (equation 4.2.1) has the following form for a monomial function $y[\boldsymbol{x}]$,

$$\begin{aligned}
&y[\boldsymbol{x}] = \prod_{j\in\{1,\ldots,N\}} (x_j-m_j)^{k_j} \quad \rightarrow \\
&\quad yFit[\boldsymbol{x}] = \prod_{j\in\{1,\ldots,N\}} \begin{cases} (x_j-m_j)^{k_j}, & k_j \le 3 \\ (x_j-m_j)^2\big((k_j-2)(x_j-m_j)+3-k_j\big), & k_j > 3 \end{cases}
\end{aligned} \tag{A.2.7}$$

Thus, the interpolator's invariant monomials are multicubic (i.e., $k_j \le 3$ for all j).

The multicubic interpolation error and error bias for a general function $y[\boldsymbol{x}]$ can be estimated by expanding its Taylor series to 5th order in δ,

$$
\begin{aligned}
y[\boldsymbol{x}] &= \sum_{k_1,\ldots,k_N\in\{0,1,2,\ldots\}} \left(\left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\,k_j} \right) y[\boldsymbol{m}] \right) \prod_{j\in\{1,\ldots,N\}} \frac{(x_j - m_j)^{k_j}}{k_j!} \\
&= (\text{invariant terms}) + \sum_{\substack{k_1,\ldots,k_N\in\{0,1,2,\ldots\};\\ \sum \boldsymbol{k}=4}} \cdots \; + \sum_{\substack{k_1,\ldots,k_N\in\{0,1,2,\ldots\};\\ \sum \boldsymbol{k}=5}} \cdots \\
&\quad + O\,\delta^6
\end{aligned}
\tag{A.2.8}
$$

wherein the “invariant terms” comprise monomials of order 3 or less. These are invariant monomials, which do not contribute to the interpolation error. Furthermore, in the sum over $\sum \boldsymbol{k} = 4$ the only non-invariant monomials are those for which one of the k_j ’s is 4 and all others are zero. Denoting the subscript of the non-zero k_j as j_1, this sum takes the form

$$
\begin{aligned}
\sum_{\substack{k_1,\ldots,k_N\in\{0,1,2,\ldots\};\\ \sum \boldsymbol{k}=4}} \cdots &= \sum_{j_1\in\{1,\ldots,N\}} (\partial_{j_1}^{\,4}\, y[\boldsymbol{m}]) \frac{(x_{j_1} - m_{j_1})^4}{4!} \\
&\quad + \quad (\text{invariant terms})
\end{aligned}
\tag{A.2.9}
$$

In the sum over $\sum \boldsymbol{k} = 5$ the non-invariant monomials are of the following form: There are either two non-zero k_j ’s, one equal to 1 and the other equal to 4; or there is one non-zero k_j, which is equal to 5. In the former case, the subscripts of the first and second non-zero k_j ’s will be denoted respectively as j_1 and j_2, and in the latter case the subscript of the non-zero k_j will be denoted as j_1,

$$
\begin{aligned}
\sum_{\substack{k_1,\ldots,k_N\in\{0,1,2,\ldots\};\\ \sum \boldsymbol{k}=5}} \cdots &= \sum_{\substack{j_1, j_2\in\{1,\ldots,N\};\\ j_1\neq j_2}} (\partial_{j_1} \partial_{j_2}^{\,4}\, y[\boldsymbol{m}]) \frac{(x_{j_1} - m_{j_1})(x_{j_2} - m_{j_2})^4}{4!} \\
&\quad + \sum_{j_1\in\{1,\ldots,N\}} (\partial_{j_1}^{\,5}\, y[\boldsymbol{m}]) \frac{(x_{j_1} - m_{j_1})^5}{5!} \quad + \quad (\text{invariant terms})
\end{aligned}
\tag{A.2.10}
$$

The error and error bias for the 4th-order monomials in equation A.2.9 can be determined by considering the representative monomial $x_1^{\,4}$ (i.e., $\boldsymbol{m} = \boldsymbol{0}$, $j_1 = 1$), as outlined in Table A.2.1. Equations 4.2.5 and 4.2.6 are obtained from Table A.2.1 by substituting $x_{j_1} - m_{j_1}$ for x_1 and combining the monomials in linear combinations having the form of equation A.2.9.

Table A.2.1. Multicubic interpolation error $\left(yFit^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}]\right)$ and error bias $\left(\int_{x_N=0}^{1}\cdots\int_{x_1=0}^{1}(yFit^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}])\,dx_1\ldots dx_N\right)$ for non-invariant monomial functions of order 4 (based on equation 4.2.1).			
order	$y[\boldsymbol{x}]$	error	bias
4	${x_1}^4$	$-{x_1}^2\,(1-x_1)^2$	$-\frac{1}{30}$

With bias compensation, the non-invariant monomial functions up to order 5 are tabulated in Table A.2.2 along with each monomial's bias-compensated form ($yComp[\boldsymbol{x}]$, equation 4.2.8) and its interpolation error and bias. Equations 4.2.9 and 4.2.10 are obtained from this tabulation.

Table A.2.2. Multicubic interpolation error $\left(yFit'^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}]\right)$ and error bias $\left(\int_{x_N=0}^{1}\cdots\int_{x_1=0}^{1}(yFit'^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}])\,dx_1\ldots dx_N\right)$ for non-invariant monomial functions of order up to 5, with bias compensation (based on equation 4.2.7).				
order	$y[\boldsymbol{x}]$	$yComp[\boldsymbol{x}]$	error	bias
4	${x_1}^4$	${x_1}^4+\frac{1}{30}$	$-{x_1}^2\,(1-x_1)^2+\frac{1}{30}$	0
5	$x_1\,{x_2}^4$	$x_1\,({x_2}^4+\frac{1}{30})$	$x_1\,(-{x_2}^2\,(1-x_2)^2+\frac{1}{30})$	0
	${x_1}^5$	$x_1\,({x_1}^4+\frac{1}{6})$	$x_1\,(-x_1\,(2+x_1)\,(1-x_1)^2+\frac{1}{6})$	0

With truncated multicubic interpolation (equation 4.2.11), the invariant monomials are multicubic and are of order L or less. (The truncation has no effect on such monomials.) With $L\le 3$, all order-L monomials are multicubic; hence the invariant monomials are simply the monomials of order L or less.

Considering the case $L=3$, the truncated terms in equation 4.2.11 have the form

$$\text{truncated terms} = \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum\boldsymbol{k}\ge 4}}\ \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\left(\prod_{j\in\{1,\ldots,N\}}{\partial_j}^{k_j}\right)y[\boldsymbol{m}+\boldsymbol{s}] \quad \text{(A.2.11)}$$

(With bias compensation, equation 4.2.12 applies and y is replaced by $yComp$ in equation A.2.11.) The incremental error bias induced by the truncation is equal to the grid-cell average of the above expression. Denoting the grid-cell average as $\langle\ldots\rangle$, the grid-cell average of $u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ is (from equation 4.2.4)

$$\left\langle u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle = \int_{x_N=m_N}^{m_N+1}\cdots\int_{x_1=m_1}^{m_1+1} u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\, dx_1 \ldots dx_N = (\tfrac{1}{2})^N \; (\tfrac{1}{6})^{\sum \boldsymbol{k}} \; (-1)^{\sum_j k_j s_j} \tag{A.2.12}$$

The truncated terms' grid-cell average is

$$\begin{aligned}\left\langle \text{truncated terms}\right\rangle &= \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum \boldsymbol{k}\ge 4}} \; \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} \left\langle u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\;k_j}\right) y[\boldsymbol{m}+\boldsymbol{s}] \\ &= (\tfrac{1}{2})^N \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum \boldsymbol{k}\ge 4}} (\tfrac{1}{6})^{\sum \boldsymbol{k}} \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} (-1)^{\sum_j k_j s_j} \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\;k_j}\right) y[\boldsymbol{m}+\boldsymbol{s}]\end{aligned} \tag{A.2.13}$$

In the above expression, the summation over any s_j for which $k_j = 1$ represents a difference term, which can be approximated by a derivative. For example, if $k_1 = 1$ then the sum over s_1 can be approximated as follows,

$$\begin{aligned} k_1 = 1 \quad &\rightarrow \quad \sum_{s_1\in\{0,1\}} (-1)^{k_1 s_1} \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\;k_j}\right) y[\boldsymbol{m}+\boldsymbol{s}] \\ &= \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\;k_j}\right) y[m_1, m_2+s_2, \ldots, m_N+s_N] - \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\;k_j}\right) y[m_1+1, m_2+s_2, \ldots, m_N+s_N] \\ &= -\left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\;k_j}\right) \partial_1 (y + O\,\delta^2)[m_1+\tfrac{1}{2}, m_2+s_2, \ldots, m_N+s_N]\end{aligned} \tag{A.2.14}$$

(The "$O\,\delta^2$" term in this expression represents higher-order approximation terms, $\frac{1}{24}\partial_1^{\;2} y + \ldots$) Applying a similar operation to all sums over s_j with $k_j = 1$, the following result is obtained,

$$\begin{aligned}&\left\langle \text{truncated terms}\right\rangle = \\ &\quad (\tfrac{1}{2})^N \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum \boldsymbol{k}\ge 4}} (-\tfrac{1}{6})^{\sum \boldsymbol{k}} \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in \begin{cases}\{0,1\} & \text{if } k_j=0\\ \{\frac{1}{2}\} & \text{if } k_j=1\end{cases}}} \left(\prod_{j\in\{1,\ldots,N\}} \partial_j^{\;2k_j}\right) (y + O\,\delta^2)[\boldsymbol{m}+\boldsymbol{s}]\end{aligned} \tag{A.2.15}$$

The derivative term in the above expression is of order $\delta^{2\sum \boldsymbol{k}}$; thus with $\sum \boldsymbol{k} \geq 4$ the expression is of order δ^8 and the order-6 error bias (equation 4.2.10) is not affected by order truncation.

With $L = 2$ a similar argument applies. In this case, $\sum \boldsymbol{k} \geq 3$ and the contribution of the truncated terms to the error bias is of order 6. Hence, the order-4 error bias (equation 4.2.17) is preserved.

The first-order derivative approximations, equations 4.2.18-20, induce an error of order δ^5 in the interpolation function. The error can be estimated by expanding the equations to 5$^{\text{th}}$ order in δ,

$$\begin{aligned}\partial_j y[\boldsymbol{x}] &= \tfrac{2}{3}(y[\boldsymbol{x}+\boldsymbol{e}_j] - y[\boldsymbol{x}-\boldsymbol{e}_j]) - \tfrac{1}{12}(y[\boldsymbol{x}+2\boldsymbol{e}_j] - y[\boldsymbol{x}-2\boldsymbol{e}_j]) \\ &\quad + \tfrac{1}{30}\partial_j{}^5 y[\boldsymbol{x}] + O\,\delta^7\end{aligned} \tag{A.2.16}$$

$$\begin{aligned}\partial_j y[\boldsymbol{x}] &= \pm\tfrac{1}{12} y[\boldsymbol{x}\pm 3\boldsymbol{e}_j] \mp \tfrac{1}{2} y[\boldsymbol{x}\pm 2\boldsymbol{e}_j] \pm \tfrac{3}{2} y[\boldsymbol{x}\pm \boldsymbol{e}_j] \mp \tfrac{5}{6} y[\boldsymbol{x}] \mp \tfrac{1}{4} y[\boldsymbol{x}\mp \boldsymbol{e}_j] \\ &\quad - \tfrac{1}{20}\partial_j{}^5 y[\boldsymbol{x}] + O\,\delta^6\end{aligned} \tag{A.2.17}$$

$$\begin{aligned}\partial_j y[\boldsymbol{x}] &= \mp\tfrac{1}{4} y[\boldsymbol{x}\pm 4\boldsymbol{e}_j] \pm \tfrac{4}{3} y[\boldsymbol{x}\pm 3\boldsymbol{e}_j] \mp 3\, y[\boldsymbol{x}\pm 2\boldsymbol{e}_j] \pm 4\, y[\boldsymbol{x}\pm \boldsymbol{e}_j] \mp \tfrac{25}{12} y[\boldsymbol{x}] \\ &\quad + \tfrac{1}{5}\partial_j{}^5 y[\boldsymbol{x}] + O\,\delta^6\end{aligned} \tag{A.2.18}$$

The effect of these approximations on the interpolation error bias can be estimated by using the following expression for the grid-cell-average interpolation function (from equations 4.2.11 and A.2.12),

$$\begin{aligned}\left\langle yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle &= \int_{x_N=m_N}^{m_N+1} \cdots \int_{x_1=m_1}^{m_1+1} yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\, dx_1 \ldots dx_N \\ &= \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum\boldsymbol{k}\leq L}} \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} \left\langle u^{[\boldsymbol{m}]}_{\boldsymbol{k},\boldsymbol{s}}[\boldsymbol{x}]\right\rangle \left(\prod_{j\in\{1,\ldots,N\}} \partial_j{}^{k_j}\right) y[\boldsymbol{m}+\boldsymbol{s}] \\ &= (\tfrac{1}{2})^N \sum_{\substack{\boldsymbol{k}=(k_1,\ldots,k_N),\\ k_j\in\{0,1\},\\ \sum\boldsymbol{k}\leq L}} (\tfrac{1}{6})^{\sum\boldsymbol{k}} \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} (-1)^{\sum_j k_j s_j} \left(\prod_{j\in\{1,\ldots,N\}} \partial_j{}^{k_j}\right) y[\boldsymbol{m}+\boldsymbol{s}]\end{aligned} \tag{A.2.19}$$

(With bias compensation, equation 4.2.12, $yComp$ replaces y in equations A.2.16-19.) For example, the approximation of $\partial_1 y[\boldsymbol{x}]$ by equation A.2.16 will only affect the $\boldsymbol{k}$

summation term $\boldsymbol{k} = (1,0,0,\ldots)$. Thus, the approximation-induced increment in $\left\langle yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle$, denoted as $\left\langle \Delta\, yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle$, would have the following form for this case,

$$\begin{aligned} &\partial_1 y[\boldsymbol{x}] \cong \tfrac{2}{3}(y[\boldsymbol{x}+\boldsymbol{e}_1]-y[\boldsymbol{x}-\boldsymbol{e}_1]) - \tfrac{1}{12}(y[\boldsymbol{x}+2\boldsymbol{e}_1]-y[\boldsymbol{x}-2\boldsymbol{e}_1]) \quad \rightarrow \\ &\quad \left\langle \Delta\, yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle = (\tfrac{1}{2})^N\,(\tfrac{1}{6})(-\tfrac{1}{30}) \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} (-1)^{s_1}\,\partial_1{}^5\, y[\boldsymbol{m}+\boldsymbol{s}] \quad + \quad O\,\delta^6 \end{aligned} \tag{A.2.20}$$

The summation over s_1 represents a difference term, which can be approximated by a 6th-order derivative,

$$\begin{aligned} &\sum_{s_1\in\{0,1\}} (-1)^{s_1}\,\partial_1{}^5\, y[\boldsymbol{m}+\boldsymbol{s}] = \\ &\quad \partial_1{}^5\, y[m_1, m_2+s_2,\ldots,m_N+s_N] - \partial_1{}^5\, y[m_1+1, m_2+s_2,\ldots,m_N+s_N] \\ &\quad = -\partial_1{}^6\, y[m_1, m_2+s_2,\ldots,m_N+s_N] + O\,\delta^7 = O\,\delta^6 \end{aligned} \tag{A.2.21}$$

Thus, the order-6 error bias (equation 4.2.10) is not affected by the $\partial_1 y[\boldsymbol{x}]$ approximation (equation 4.2.18), and a similar result follows using either of equations 4.2.19-20.

The 4-point derivative estimators, equations 4.2.21-22, have errors of order δ^4. Following are 4th-order expansions of these equations,

$$\begin{aligned} &\partial_j y[\boldsymbol{x}] = \mp\tfrac{1}{6}\, y[\boldsymbol{x}\pm 2\boldsymbol{e}_j] \pm y[\boldsymbol{x}\pm\boldsymbol{e}_j] \mp \tfrac{1}{2}\, y[\boldsymbol{x}] \mp \tfrac{1}{3}\, y[\boldsymbol{x}\mp\boldsymbol{e}_j] \\ &\quad + \tfrac{1}{12}\partial_j{}^4 y[\boldsymbol{x}] + O\,\delta^5 \end{aligned} \tag{A.2.22}$$

$$\begin{aligned} &\partial_j y[\boldsymbol{x}] = \pm\tfrac{1}{3}\, y[\boldsymbol{x}\pm 3\boldsymbol{e}_j] \mp \tfrac{3}{2}\, y[\boldsymbol{x}\pm 2\boldsymbol{e}_j] \pm 3\, y[\boldsymbol{x}\pm\boldsymbol{e}_j] \mp \tfrac{11}{6}\, y[\boldsymbol{x}] \\ &\quad - \tfrac{1}{4}\partial_j{}^4 y[\boldsymbol{x}] + O\,\delta^5 \end{aligned} \tag{A.2.23}$$

Using equation 4.2.21 or 4.2.22, the approximation-induced error bias is of order δ^5.

The 2nd-order, mixed-derivative operator in equation 4.2.25 can be expanded to 4th order as follows,

$$\begin{aligned} &\partial_{j_1}\partial_{j_2} y[\boldsymbol{x}] = \frac{1}{4}\left(\begin{aligned} &y[\boldsymbol{x}+\boldsymbol{e}_{j_1}+\boldsymbol{e}_{j_2}] - y[\boldsymbol{x}-\boldsymbol{e}_{j_1}+\boldsymbol{e}_{j_2}] \\ &\quad - y[\boldsymbol{x}+\boldsymbol{e}_{j_1}-\boldsymbol{e}_{j_2}] + y[\boldsymbol{x}-\boldsymbol{e}_{j_1}-\boldsymbol{e}_{j_2}] \end{aligned} \right) \\ &\quad + \tfrac{1}{6}(\partial_{j_1}{}^3\partial_{j_2} y[\boldsymbol{x}] + \partial_{j_1}\partial_{j_2}{}^3 y[\boldsymbol{x}]) + O\,\delta^6 \end{aligned} \tag{A.2.24}$$

Considering, for example, the approximation for $\partial_1 \partial_2 y[\boldsymbol{x}]$, the associated increment in $\left\langle yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] \right\rangle$ (equation A.2.19) has the form

$$
\begin{aligned}
&\partial_1 \partial_2 y[\boldsymbol{x}] \cong \tfrac{1}{4}\left(y[\boldsymbol{x}+\boldsymbol{e}_1+\boldsymbol{e}_2] - y[\boldsymbol{x}-\boldsymbol{e}_1+\boldsymbol{e}_2] - y[\boldsymbol{x}+\boldsymbol{e}_1-\boldsymbol{e}_2] + y[\boldsymbol{x}-\boldsymbol{e}_1-\boldsymbol{e}_2] \right) \quad \to \\
&\left\langle yFit^{[\boldsymbol{m}]}[\boldsymbol{x}] \right\rangle = \\
&\quad -(\tfrac{1}{2})^N (\tfrac{1}{6})^3 \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} (-1)^{s_1+s_2} \left(\partial_1{}^3 \partial_2 y[\boldsymbol{m}+\boldsymbol{s}] + \partial_1 \partial_2{}^3 y[\boldsymbol{m}+\boldsymbol{s}] \right) \quad + \quad O\,\delta^6
\end{aligned}
\tag{A.2.25}
$$

This expression is of order δ^6, e.g.,

$$
\begin{aligned}
&\sum_{s_1,s_2\in\{0,1\}} (-1)^{s_1+s_2}\, \partial_1{}^3 \partial_2 y[\boldsymbol{m}+\boldsymbol{s}] = \\
&\quad \partial_1{}^3 \partial_2 y[m_1, m_2, m_3+s_3, \ldots] - \partial_1{}^3 \partial_2 y[m_1+1, m_2, m_3+s_3, \ldots] \\
&\quad - \partial_1{}^3 \partial_2 y[m_1, m_2+1, m_3+s_3, \ldots] + \partial_1{}^3 \partial_2 y[m_1+1, m_2+1, m_3+s_3, \ldots] \\
&\quad = \partial_1{}^4 \partial_2{}^2 y[m_1, m_2, m_3+s_3, \ldots] + O\,\delta^7 = O\,\delta^6
\end{aligned}
\tag{A.2.26}
$$

Thus, application of equation 4.2.25 preserves the order-6 error bias (equation 4.2.10). The mixed-derivative estimator in equation 4.2.25 is obtained by combining two symmetric, first-order derivative estimators (equation 4.2.23), but the same result applies if asymmetric derivatives (equation 4.2.24) are used. Furthermore, a similar analysis can be applied to higher-order mixed-derivative estimators (e.g., equation 4.2.26) that are constructed from symmetric or asymmetric first-order derivative estimators.

A.3 Reduced-cubic interpolation

The reduced-cubic interpolation function $yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]$ (equation 4.3.1) is a polynomial that is constructed to match y and the gradient of y at the grid points $\boldsymbol{x} = \boldsymbol{m} + \boldsymbol{s}$ (equations 4.3.2). These conditions define $2^N\,(N+1)$ constraints, and the interpolating polynomial must comprise an equivalent number of basis monomials and corresponding coefficients to satisfy the matching conditions.

The basis monomials for reduced cubic interpolation comprise the multilinear monomials $\prod_{j\in\{1,\ldots,N\}} (x_j - m_j)^{t_j}$, $t_j = 0$ or 1 (of which there are 2^N), and also the products of all multilinear monomials with $(x_i - m_i)^2$ for each $i \in \{1,\ldots,N\}$ (defining $2^N\,N$ additional distinct monomials, for a total of $2^N\,(N+1)$). In other words, the basis monomials are of the form $\prod_{j\in\{1,\ldots,N\}}(x_j-m_j)^{t_j}$, wherein $t_j=0,1,2,\text{ or }3$ for all j, and $t_j=2\text{ or }3$ for at most one j.

The cardinal functions $u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ and $v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]$ defined by equations 4.3.4 and 4.3.5 are linear combinations of the reduced-cubic basis monomials. This can be confirmed by expanding the product factor in these equations into a sum of multilinear basis monomials as follows,

$$\begin{aligned}&\prod_{j\in\{1,\ldots,N\}}\big((1-s_j)+(2s_j-1)(x_j-m_j)\big)=\\&\quad\sum_{t_1,\ldots,t_N\in\{0,1\}}\prod_{j\in\{1,\ldots,N\}}(1-s_j)^{1-t_j}\ ((2s_j-1)(x_j-m_j))^{t_j}\end{aligned}\tag{A.3.1}$$

Equations 4.3.3 can be verified by direct substitution from equations 4.3.4 and 4.3.5. (For example, with $x_j-m_j=s'_j=0\text{ or }1$, the product term in equation 4.3.4 will be zero if $s_j\neq s'_j$ for any j; hence the first of equations 4.3.3 is established.) It then follows from equations 4.3.3 and 4.3.1 that the interpolation function $yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]$ satisfies the matching conditions, equations 4.3.2.

The interpolating function is a linear combination of the basis functions, of the general form

$$\begin{aligned}&yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]=\\&\sum_{t_1,\ldots,t_N\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N\}}(x_j-m_j)^{t_j}\right)\left(a_{(t_1,\ldots,t_N)}+\sum_{i\in\{1,\ldots,N\}}b_{i,(t_1,\ldots,t_N)}(x_i-m_i)^2\right)\right)\end{aligned}\tag{A.3.2}$$

If $y[\boldsymbol{x}]$ is similarly a linear combination of basis monomials of the form

$$y[\boldsymbol{x}]=\sum_{t_1,\ldots,t_N\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N\}}(x_j-m_j)^{t_j}\right)\left(a'_{(t_1,\ldots,t_N)}+\sum_{i\in\{1,\ldots,N\}}b'_{i,(t_1,\ldots,t_N)}(x_i-m_i)^2\right)\right)\tag{A.3.3}$$

then $yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]$ will match $y[\boldsymbol{x}]$ identically and the interpolation error will be identically zero, provided that the monomial coefficients $a_{(t_1,\ldots,t_N)}$ and $b_{i,(t_1,\ldots,t_N)}$ are uniquely determined by conditions 4.3.2 (i.e., $a_{(t_1,\ldots,t_N)}=a'_{(t_1,\ldots,t_N)}$ and $b_{i,(t_1,\ldots,t_N)}=b'_{i,(t_1,\ldots,t_N)}$). To establish uniqueness of the coefficients, it suffices to show that if $y[\boldsymbol{x}]$ is identically zero, then the coefficients are all zero (i.e., we can consider the difference of two $yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]$

functions corresponding to the same $y[\boldsymbol{x}]$, and establish that the expansion coefficients for the difference must all be zero). Setting $y[\boldsymbol{x}]=0$, and taking advantage of the following three identities,

$$(s_i)^2 = s_i \tag{A.3.4}$$

$$(s_k)^{t_k+1} = s_k \tag{A.3.5}$$

$$\frac{d}{dx_k}(x_k - m_k)^{t_k} = t_k \tag{A.3.6}$$

(because $s_i = 0 \text{ or } 1$ and $t_k = 0 \text{ or } 1$), the first of equations 4.3.2 (with substitution of equation A.3.2 and $y[\boldsymbol{x}]=0$) reduces to

$$\sum_{t_1,\ldots,t_N\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N\}}(s_j)^{t_j}\right)\left(a_{(t_1,\ldots,t_N)} + \sum_{i\in\{1,\ldots,N\}} b_{i,(t_1,\ldots,t_N)}\, s_i\right)\right) = 0 \tag{A.3.7}$$

and the second equation reduces to

$$\sum_{t_1,\ldots,t_N\in\{0,1\}}\left(\left(\prod_{\substack{j\in\{1,\ldots,N\};\\ j\neq k}}(s_j)^{t_j}\right)\left(\begin{array}{l}\left(a_{(t_1,\ldots,t_N)} + \sum\limits_{\substack{i\in\{1,\ldots,N\};\\ i\neq k}} b_{i,(t_1,\ldots,t_N)}\, s_i\right)t_k \\ \quad + (2+t_k)\, b_{k,(t_1,\ldots,t_N)}\, s_k\end{array}\right)\right) = 0; \quad k\in\{1,\ldots,N\} \tag{A.3.8}$$

The sum over t_N in equation A.3.7 can be expanded to obtain the following equivalent expression,

$$\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)\left(\begin{array}{l}\left(a_{(t_1,\ldots,t_{N-1},0)} + \sum\limits_{i\in\{1,\ldots,N\}} b_{i,(t_1,\ldots,t_{N-1},0)}\, s_i\right) + \\ \quad s_N\left(a_{(t_1,\ldots,t_{N-1},1)} + \sum\limits_{i\in\{1,\ldots,N\}} b_{i,(t_1,\ldots,t_{N-1},1)}\, s_i\right)\end{array}\right)\right) = 0 \tag{A.3.9}$$

This equivalence holds for $s_N = 0 \text{ or } 1$; hence the following two conditions are obtained,

$$\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)\left(a_{(t_1,\ldots,t_{N-1},0)}+\sum_{i\in\{1,\ldots,N-1\}}b_{i,(t_1,\ldots,t_{N-1},0)}\,s_i\right)\right)=0 \tag{A.3.10}$$

$$\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)\left(\begin{array}{l}a_{(t_1,\ldots,t_{N-1},1)}+b_{N,(t_1,\ldots,t_{N-1},0)}+b_{N,(t_1,\ldots,t_{N-1},1)}\\ \quad+\displaystyle\sum_{i\in\{1,\ldots,N-1\}}b_{i,(t_1,\ldots,t_{N-1},1)}\,s_i\end{array}\right)\right)=0 \tag{A.3.11}$$

With $k=N$ equation A.3.8 takes the form

$$\sum_{t_1,\ldots,t_N\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)\left(\begin{array}{l}\left(a_{(t_1,\ldots,t_N)}+\displaystyle\sum_{i\in\{1,\ldots,N-1\}}b_{i,(t_1,\ldots,t_N)}\,s_i\right)t_N\\ \quad+(2+t_N)\,b_{N,(t_1,\ldots,t_N)}\,s_N\end{array}\right)\right)=0 \tag{A.3.12}$$

Expanding the t_N sum and considering the two cases $s_N=0$ or 1, this implies the following two equations,

$$\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)\left(a_{(t_1,\ldots,t_{N-1},1)}+\sum_{i\in\{1,\ldots,N-1\}}b_{i,(t_1,\ldots,t_{N-1},1)}\,s_i\right)\right)=0 \tag{A.3.13}$$

$$\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)(2b_{N,(t_1,\ldots,t_{N-1},0)}+3b_{N,(t_1,\ldots,t_{N-1},1)})\right)=0 \tag{A.3.14}$$

Subtracting equation A.3.13 from A.3.11 yields

$$\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)(b_{N,(t_1,\ldots,t_{N-1},0)}+b_{N,(t_1,\ldots,t_{N-1},1)})\right)=0 \tag{A.3.15}$$

and equations A.3.14 and A.3.15 imply

$$\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)b_{N,(t_1,\ldots,t_{N-1},s_N)}\right)=0;\quad s_N=0\text{ or }1 \tag{A.3.16}$$

Expanding the t_{N-1} sum in this expression yields

$$\sum_{t_1,\ldots,t_{N-2}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-2\}}(s_j)^{t_j}\right)(b_{N,(t_1,\ldots,t_{N-2},0,s_N)}+s_{N-1}\,b_{N,(t_1,\ldots,t_{N-2},1,s_N)})\right)=0 \tag{A.3.17}$$

and considering the two cases $s_{N-1}=0 \text{ or } 1$ yields

$$\sum_{t_1,\ldots,t_{N-2}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-2\}}(s_j)^{t_j}\right)b_{N,(t_1,\ldots,t_{N-2},s_{N-1},s_N)}\right)=0;\quad s_{N-1}=0 \text{ or } 1 \tag{A.3.18}$$

The steps leading from equation A.3.16 to A.3.18 can be repeated inductively to obtain

$$\sum_{t_1,\ldots,t_{N-n}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-n\}}(s_j)^{t_j}\right)b_{N,(t_1,\ldots,t_{N-n},s_{N-n+1},\ldots,s_N)}\right)=0;\quad n\in\{1,\ldots,N\} \tag{A.3.19}$$

With $n=N$ this reduces to

$$b_{N,(s_1,\ldots,s_N)}=0 \tag{A.3.20}$$

Equations A.3.10 and A.3.13 have the form

$$\begin{gathered}\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{j\in\{1,\ldots,N-1\}}(s_j)^{t_j}\right)\left(a_{(t_1,\ldots,t_{N-1},s_N)}+\sum_{i\in\{1,\ldots,N-1\}}b_{i,(t_1,\ldots,t_{N-1},s_N)}\,s_i\right)\right)=0;\\ s_N=0 \text{ or } 1\end{gathered} \tag{A.3.21}$$

With $k<N$ the t_N sum in equation A.3.8 can be expanded and equation A.3.20 can be applied to obtain

$$\begin{gathered}\sum_{t_1,\ldots,t_{N-1}\in\{0,1\}}\left(\left(\prod_{\substack{j\in\{1,\ldots,N-1\};\\ j\neq k}}(s_j)^{t_j}\right)\left(\left(a_{(t_1,\ldots,t_{N-1},s_N)}+\sum_{\substack{i\in\{1,\ldots,N-1\};\\ i\neq k}}b_{i,(t_1,\ldots,t_{N-1},s_N)}\,s_i\right)t_k \atop +(2+t_k)b_{k,(t_1,\ldots,t_{N-1},s_N)}\,s_k\right)\right)=0;\\ k\in\{1,\ldots,N-1\}\end{gathered} \tag{A.3.22}$$

The steps leading from equations A.3.7 and A.3.8 to equations A.3.20, A.3.21 and A.3.22 can be repeated inductively to obtain the following three relations,

$$b_{N-n+1,(s_1,\ldots,s_N)} = 0; \quad n \in \{1,\ldots,N\} \tag{A.3.23}$$

$$\sum_{t_1,\ldots,t_{N-n}\in\{0,1\}} \left(\left(\prod_{j\in\{1,\ldots,N-n\}} (s_j)^{t_j} \right) \left(\begin{array}{l} a_{(t_1,\ldots,t_{N-n},s_{N-n+1},\ldots,s_N)} \\ \quad + \displaystyle\sum_{i\in\{1,\ldots,N-n\}} b_{i,(t_1,\ldots,t_{N-n},s_{N-n+1},\ldots,s_N)}\, s_i \end{array} \right) \right) = 0;$$
$$n \in \{1,\ldots,N\} \tag{A.3.24}$$

$$\sum_{t_1,\ldots,t_{N-n}\in\{0,1\}} \left(\left(\prod_{\substack{j\in\{1,\ldots,N-n\};\\ j\neq k}} (s_j)^{t_j} \right) \left(\begin{array}{l} \left(\begin{array}{l} a_{(t_1,\ldots,t_{N-n},s_{N-n+1},\ldots,s_N)} \\ \quad + \displaystyle\sum_{\substack{i\in\{1,\ldots,N-1\};\\ i\neq k}} b_{i,(t_1,\ldots,t_{N-n},s_{N-n+1},\ldots,s_N)}\, s_i \end{array} \right) t_k \\ \quad + (2+t_k)\, b_{k,(t_1,\ldots,t_{N-n},s_{N-n+1},\ldots,s_N)}\, s_k \end{array} \right) \right) = 0;$$
$$k \in \{1,\ldots,N-n\}, \quad n \in \{1,\ldots,N\} \tag{A.3.25}$$

Substituting $n = N + 1 - i$ in equation A.3.23, and $n = N$ in A.3.24, yields

$$b_{i,(s_1,\ldots,s_N)} = 0 \tag{A.3.26}$$

$$a_{(s_1,\ldots,s_N)} = 0 \tag{A.3.27}$$

Thus, if $y[\boldsymbol{x}]$ is identically zero the monomial coefficients in equation A.3.2 are all zero. It follows that for a general function $y[\boldsymbol{x}]$ the coefficients are uniquely determined by equations 4.3.2, and hence if $y[\boldsymbol{x}]$ is a linear combination of the basis monomials (equation A.3.3) the interpolation error is identically zero. This establishes that the basis monomials are the invariant monomials for reduced-cubic interpolation.

The multicubic interpolation error and error bias for a general function $y[\boldsymbol{x}]$ can be estimated by using its 5th-order Taylor series, equation A.2.8. The 3rd- and lower-order terms in the series (indicated as "invariant terms") comprise invariant monomials, which do not contribute to the interpolation error. In the sum over $\sum \boldsymbol{k} = 4$ the only non-invariant monomials are those for which one of the k_j 's is 4 and all others are zero, or two of the k_j 's are 2 and all others are zero. Thus, this sum takes the form

$$\sum_{\substack{k_1,\ldots,k_N\in\{0,1,2,\ldots\};\\ \sum\boldsymbol{k}=4}} \cdots = \sum_{\substack{j_1,j_2\in\{1,\ldots,N\};\\ j_1<j_2}} (\partial_{j_1}{}^2\,\partial_{j_2}{}^2\,y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})^2\,(x_{j_2}-m_{j_2})^2}{2!\cdot 2!}$$
$$+\sum_{j_1\in\{1,\ldots,N\}} (\partial_{j_1}{}^4\,y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})^4}{4!} \quad + \quad \text{(invariant terms)}$$

(A.3.28)

Considering only non-invariant monomials, the sum over $\sum\boldsymbol{k}=5$ takes the form,

$$\sum_{\substack{k_1,\ldots,k_N\in\{0,1,2,\ldots\};\\ \sum\boldsymbol{k}=5}} \cdots =$$
$$\sum_{\substack{j_1,j_2,j_3\in\{1,\ldots,N\};\\ j_1\neq j_2,\ j_1\neq j_3,\ j_2<j_3}} (\partial_{j_1}\,\partial_{j_2}{}^2\,\partial_{j_3}{}^2\,y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})(x_{j_2}-m_{j_2})^2\,(x_{j_3}-m_{j_3})^2}{2!\cdot 2!}$$
$$+\sum_{\substack{j_1,j_2\in\{1,\ldots,N\};\\ j_1\neq j_2}} (\partial_{j_1}{}^2\,\partial_{j_2}{}^3\,y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})^2\,(x_{j_2}-m_{j_2})^3}{2!\cdot 3!}$$
$$+\sum_{\substack{j_1,j_2\in\{1,\ldots,N\};\\ j_1\neq j_2}} (\partial_{j_1}\,\partial_{j_2}{}^4\,y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})(x_{j_2}-m_{j_2})^4}{4!}$$
$$+\sum_{j_1\in\{1,\ldots,N\}} (\partial_{j_1}{}^5\,y[\boldsymbol{m}])\frac{(x_{j_1}-m_{j_1})^5}{5!} \quad + \quad \text{(invariant terms)}$$

(A.3.29)

Table A.3.1 tabulates the representative monomials $x_1{}^2\,x_2{}^2$ and $x_1{}^4$, and the associated errors and error bias corresponding to the 4th-order monomials in equation A.3.28. Equations 4.3.6 and 4.3.7 are obtained from this tabulation.

Table A.3.1. Reduced-cubic interpolation error $\left(yFit^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}]\right)$ and error bias $\left(\int_{x_N=0}^{1}\cdots\int_{x_1=0}^{1}(yFit^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}])\,dx_1\ldots dx_N\right)$ for non-invariant monomial functions of order 4 (based on equation 4.3.1).

order	$y[\boldsymbol{x}]$	error	bias
4	$x_1{}^2\,x_2{}^2$	$-x_1(1-x_1)\,x_2\,(1-x_2)$	$-\frac{1}{36}$
	$x_1{}^4$	$-x_1{}^2\,(1-x_1)^2$	$-\frac{1}{30}$

With bias compensation, the non-invariant monomial functions up to order 5 are tabulated in Table A.3.2 along with each monomial's bias-compensated form ($yComp[\boldsymbol{x}]$, equation 4.3.9) and its interpolation error and bias. Equations 4.3.10 and 4.3.11 are obtained from this tabulation.

Table A.3.2. Reduced-cubic interpolation error $\left(yFit'^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}]\right)$ and error bias $\left(\int_{x_N=0}^{1}\cdots\int_{x_1=0}^{1}(yFit'^{[\boldsymbol{0}]}[\boldsymbol{x}]-y[\boldsymbol{x}])\,dx_1\ldots dx_N\right)$ for non-invariant monomial functions of order up to 5, with bias compensation (based on equation 4.3.8).				
order	$y[\boldsymbol{x}]$	$yComp[\boldsymbol{x}]$	error	bias
4	$x_1{}^2\,x_2{}^2$	$x_1{}^2\,x_2{}^2+\frac{1}{36}$	$-x_1\,(1-x_1)\,x_2\,(1-x_2)+\frac{1}{36}$	0
	$x_1{}^4$	$x_1{}^4+\frac{1}{30}$	$-x_1{}^2\,(1-x_1)^2+\frac{1}{30}$	0
5	$x_1\,x_2{}^2\,x_3{}^2$	$x_1\,(x_2{}^2\,x_3{}^2+\frac{1}{36})$	$x_1\,(-x_2\,(1-x_2)\,x_3\,(1-x_3)+\frac{1}{36})$	0
	$x_1{}^2\,x_2{}^3$	$x_2\,(x_1{}^2\,x_2{}^2+\frac{1}{12})$	$x_2\,(-x_1\,(1-x_1)(1-x_2{}^2)+\frac{1}{12})$	0
	$x_1\,x_2{}^4$	$x_1\,(x_2{}^4+\frac{1}{30})$	$x_1\,(-x_2{}^2\,(1-x_2)^2+\frac{1}{30})$	0
	$x_1{}^5$	$x_1\,(x_1{}^4+\frac{1}{6})$	$x_1\,(-x_1\,(2+x_1)(1-x_1)^2+\frac{1}{6})$	0

The grid-cell-average cardinal functions have the following form (from equations 4.3.4 and 4.3.5),

$$\left\langle u_{\boldsymbol{k},\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle=\int_{x_N=m_N}^{m_N+1}\cdots\int_{x_1=m_1}^{m_1+1}u_{\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\,dx_1\ldots dx_N\;=(\tfrac{1}{2})^N \tag{A.3.30}$$

$$\left\langle v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle=\int_{x_N=m_N}^{m_N+1}\cdots\int_{x_1=m_1}^{m_1+1}v_{k,\boldsymbol{s}}^{[\boldsymbol{m}]}[\boldsymbol{x}]\,dx_1\ldots dx_N\;=(\tfrac{1}{2})^N\,(\tfrac{1}{6})(-1)^{s_k} \tag{A.3.31}$$

The effect of the derivative approximations on the interpolation error bias can be estimated by using the following expression for the grid-cell-average interpolation function (from equations 4.3.1, A.3.30 and A.3.31),

$$\begin{aligned}\left\langle yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle&=\int_{x_N=m_N}^{m_N+1}\cdots\int_{x_1=m_1}^{m_1+1}yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\,dx_1\ldots dx_N\\&=(\tfrac{1}{2})^N\sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}}\left(y[\boldsymbol{m}+\boldsymbol{s}]+(\tfrac{1}{6})\sum_{k\in\{1,\ldots,N\}}(-1)^{s_k}\,\partial_k y[\boldsymbol{m}+\boldsymbol{s}]\right)\end{aligned} \tag{A.3.32}$$

(With bias compensation, equation 4.3.8, $yComp$ replaces y in these equations.) If equation 4.2.18 is used to approximate $\partial_1 y[\boldsymbol{x}]$, then the error in the derivative is given approximately by the 5th-order term in equation A.2.16, and the resulting approximation-induced increment in $\left\langle yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle$, denoted as $\left\langle \Delta\, yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle$, would have the following form,

$$\partial_1 y[\boldsymbol{x}] \cong \tfrac{2}{3}(y[\boldsymbol{x}+\boldsymbol{e}_1]-y[\boldsymbol{x}-\boldsymbol{e}_1])-\tfrac{1}{12}(y[\boldsymbol{x}+2\boldsymbol{e}_1]-y[\boldsymbol{x}-2\boldsymbol{e}_1]) \quad \to$$
$$\left\langle \Delta\, yFit^{[\boldsymbol{m}]}[\boldsymbol{x}]\right\rangle = (\tfrac{1}{2})^N\,(\tfrac{1}{6})(-\tfrac{1}{30}) \sum_{\substack{\boldsymbol{s}=(s_1,\ldots,s_N),\\ s_j\in\{0,1\}}} (-1)^{s_1}\,\partial_1{}^5 y[\boldsymbol{m}+\boldsymbol{s}] \quad + \quad O\,\delta^6 \tag{A.3.33}$$

The summation over s_1 represents a difference term, which can be approximated by a 6th-order derivative,

$$\begin{aligned}&\sum_{s_1\in\{0,1\}}(-1)^{s_1}\,\partial_1{}^5\,y[\boldsymbol{m}+\boldsymbol{s}] = \\ &\quad \partial_1{}^5\,y[m_1,m_2+s_2,\ldots,m_N+s_N]-\partial_1{}^5\,y[m_1+1,m_2+s_2,\ldots,m_N+s_N] \\ &\quad = -\partial_1{}^6\,y[m_1,m_2+s_2,\ldots,m_N+s_N]+O\,\delta^7 = O\,\delta^6\end{aligned} \tag{A.3.34}$$

Thus, the order-6 error bias (equation 4.3.11) is not affected by the $\partial_1 y[\boldsymbol{x}]$ approximation (equation 4.2.18), and a similar result follows using either of equations 4.2.19-20. (However, with the 4-point derivative estimators, equations 4.2.21-22, the error bias would be of order δ^5.)

The reduced-cubic interpolation function is continuous across grid cell boundaries. For example, the continuity condition at the boundary $x_N = m_N$ is

$$yFit^{[m_1,\ldots,m_N]}[x_1,\ldots,x_{N-1},m_N]-yFit^{[m_1,\ldots,m_N-1]}[x_1,\ldots,x_{N-1},m_N]=0 \tag{A.3.35}$$

The function's gradient, however, is not continuous between cells, although based on the gradient-matching condition (the second of equations 4.3.2) it is continuous at the sampling grid points. The order of the gradient discontinuity between grid cells can be estimated by differentiating the interpolation error, equation 4.3.6. The interpolation fitting error is of order δ^4; and the gradient error and gradient discontinuity are both also of order δ^4. (By contrast, a similar calculation with equation 4.1.5 indicates that the multilinear interpolation's gradient error and discontinuity are of order δ^2.)