WHICH CUBIC SPLINE SHOULD ONE USE?

by

R.K. Beatson and E. Chacko

Department of Mathematics, University of Canterbury, Christchurch, New Zealand.

Which cubic spline should one use ?

R.K. Beatson and E. Chacko *

November 30, 1990

Abstract

The aim of this paper is to provide a quantitative comparison of eight different C^1 and C^2 cubic spline interpolation schemes. The C^1 schemes discussed are local while the C^2 ones are global.

In practice cubic splines are often used when the smoothness of the function being interpolated/approximated is unknown. Also it is often necessary, or advantageous, to use a nonuniform mesh. Therefore we compare performance over a variety of smoothness classes, using uniform and also several thousand random meshes. The performance criteria used are the *quantitative* ones of exact operator and derived operator norms, and best possible pointwise error estimates.

Key words: Cubic splines, end conditions, operator norms, error estimates

AMS(MOS) subject classifications: 65D07, 65D10, 41A15.

1 Introduction

Our aim is to find an interpolant that will give good results when used to interpolate functions of unknown smoothness, on a possibly nonuniform mesh. We consider eight different C^1 or C^2 cubic spline interpolation schemes and compare their operator and derived operator norms as well as the pointwise error

$$\sup_{f \in \mathcal{F}} |(f - s)(x)|$$

for uniform, as well as several thousand random knot distributions, and several smoothness classes $\mathcal{F}$. It is worth emphasising that we do not seek the best interpolation method for a fixed smoothness class $\mathcal{F}$ – the problem of optimal interpolation.

All the schemes considered fit cubic splines with knots at the nodes of interpolation t_i —

$$s(t_i) = f(t_i), \qquad i = 0, 1, \dots, n.$$

*Mathematics Department, University of Canterbury, Private Bag, Christchurch 1, New Zealand. Email: rkb@math.canterbury.ac.nz, pec@math.canterbury.ac.nz

1.1 C^2 methods – not strictly local

In the first 6 methods the spline is chosen to be C^2, so that each scheme corresponds to a different choice of two *end-conditions*. Such schemes are not suitable for certain applications as they are not *strictly local*. Thus data far away from x can influence the value of $s(x)$. However, they are *semi-local*, meaning that the influence of data at point t_i on $s(x)$ falls off geometrically with the number of knots between x and t_i (see the discussion in section 2.1 below).

Method A

$$s^{(3)}(t_1-) = s^{(3)}(t_1+) \quad \text{and} \quad s^{(3)}(t_{n-1}-) = s^{(3)}(t_{n-1}+) \tag{1}$$

the well known *not-a-knot* end condition. (See Kershaw [7] and de Boor [2].) This end condition forces the restrictions of the spline to the first two and the last two intervals to be a single cubics. The maximum convergence rate, meaning the rate for a general C^∞ function, is $\mathcal{O}(\delta^4)$, where δ is the mesh size. Of course the limiting rate is actually achieved for C^3 functions with Lipschitz third derivative.

Method B

$$s'(t_0) = c'_l(t_0) \quad \text{and} \quad s'(t_n) = c'_r(t_n) \tag{2}$$

where c_l and c_r are cubic polynomials with

$$c_l(t_i) = f(t_i) \quad \text{and} \quad c_r(t_{n-i}) = f(t_{n-i}), \quad 0 \le i \le 3.$$

Thus, this spline is chosen to have the same first derivative as the local cubic interpolant through the first (last) four knots, at the first (last) knot. The maximum convergence rate is $\mathcal{O}(\delta^4)$.

Method C

$$s''(t_0) = c''_l(t_0) \quad \text{and} \quad s''(t_n) = c''_r(t_n) \tag{3}$$

with c_l and c_r as above. Thus, this spline is chosen to have the same second derivative as the local cubic interpolant through the first (last) four knots, at the first (last) knot. The maximum convergence rate is $\mathcal{O}(\delta^4)$.

Method D

$$d_1 = d_2 \quad \text{and} \quad d_{n-2} = d_{n-1} \tag{4}$$

where

$$d_i = s^{(3)}(t_i+) - s^{(3)}(t_i-).$$

Here the jump discontinuities in the third derivative at the second and third knots are forced to be the same, and similarly for the second and third to last knots. This method is known to minimize $\|f - s\|_\infty$ when the knots are equispaced and f is a quartic polynomial. The maximum convergence rate is $\mathcal{O}(\delta^4)$, and the method is not recommended for use with non-uniform meshes.

2

Method E

$$s''(t_0) = 0 \quad \text{and} \quad s''(t_n) = 0 \tag{5}$$

the so called *natural cubic spline* end condition. This *should not in general be used* for approximation purposes as it throws away any second derivative information in the data and, as a consequence, the order of approximation near the endpoints is restricted to $\mathcal{O}(\delta^2)$.

Method F

$$s'(t_0) = q_l'(t_0) \quad \text{and} \quad s'(t_n) = q_r'(t_n) \tag{6}$$

where q_l and q_r are quadratic polynomials with

$$q_l(t_i) = f(t_i) \quad \text{and} \quad q_r(t_{n-i}) = f(t_{n-i}), \quad 0 \le i \le 2.$$

Thus, this spline is chosen to have the same first derivative as the local quadratic interpolant through the first (last) three knots, at the first (last) knot. The maximum convergence rate is $\mathcal{O}(\delta^3)$.

1.2 C^1 methods – strictly local

The last two schemes are simple *strictly local* methods. This property allows easy modification of previously obtained fits, and is essential for some applications such as CAD. However, there is a cost in that the error estimates for smooth functions away from the ends of the interval are generally somewhat worse. The local methods we consider are C^1 rather than C^2.

Method G s' agrees with the derivative of a local quadratic at every knot. Thus defining q_i as the quadratic that interpolates to f at t_i, t_{i+1} and t_{i+2},

$$s'(t_j) = \begin{cases} q_0'(t_0), & j = 0, \\ q_{j-1}'(t_j), & 0 < j < n, \\ q_{n-2}'(t_n), & j = n. \end{cases} \tag{7}$$

The maximum convergence rate is $\mathcal{O}(\delta^3)$.

Method H s' agrees with the derivative of a local cubic at every knot. Thus defining c_i as the cubic that interpolates to f at t_i, t_{i+1}, t_{i+2} and t_{i+3},

$$s'(t_j) = \begin{cases} c_0'(t_0), & j = 0, \\ c_{j-1}'(t_j), & 0 < j \le n/2, \\ c_{j-2}'(t_j), & n/2 < j < n, \\ c_{n-3}'(t_n), & j = n. \end{cases} \tag{8}$$

The maximum convergence rate is $\mathcal{O}(\delta^4)$.

1.3 Outline of criteria used

The present work is an extension of that in [1] where it was shown that, for methods A, B and C

$$\|f - s\|_\infty \le K\delta^j \omega(f^{(j)}, \delta) \quad 1 \le j \le 3. \tag{9}$$

3

where $\omega(f^{(j)}, \delta)$ is the modulus of continuity for $f^{(j)}$. For $j = 2, 3$ K is an absolute constant independent of the mesh t. For $j = 1$ it depends on the spacing of the first few and last few knots. Estimate (9) implies, in particular, $\mathcal{O}(\delta^4)$ approximation to C^4 functions. However, the paper [1] does not give any grounds for choosing a particular interpolant from amongst those methods with $\mathcal{O}(\delta^4)$ error estimates, and no information about strictly local methods, or those methods with maximum convergence rates less than $\mathcal{O}(\delta^4)$.

Our present aim was therefore to make a quantitative comparison of various commonly used cubic spline interpolants. Comparisons are made using uniform meshes and also several thousand nonuniform meshes. For each of these cases we computed the operator norms of the spline projector itself, and also of the first derived projectors $L' : f' \to s'$. For the C^2 methods we also computed the norm of the second derived projector $L'' : f'' \to s''$. These norms give a quantitative measure of the tendency of the particular spline operator to introduce *extraneous bumps and wiggles*. We also compute the *pointwise error multiplier*

$$K(j, x) = \sup_{f \in W_{j,\infty} \, : \, \|f^{(j)}\|_\infty \leq 1} |f(x) - s(x)|,$$

$1 \leq j \leq 4$. The utility of these functions lies in the associated "tight" bounds

$$|f(x) - s(x)| \leq K(j, x)\|f^{(j)}\|_\infty$$

and

$$\|f - s\|_\infty \leq C_j \|f^{(j)}\|_\infty$$

where $C_j = \sup_x K(j, x)$.

1.4 Summary of results

The results are presented in detail in section 2, and the mathematics underlying the calculations in section 3. While, we do reach a conclusion and recommend one method, namely method B, for general purpose use, the reader can use the results to analyze the pros and cons of other choices. For example, the results show the cost of using the *strictly local* method H, rather than a more conventional cubic spline interpolant, when approximating C^4 functions in the middle of the interval. For uniform meshes this is a worsening of the error bound by a factor of approximately 1.8, which may be quite acceptable. They also show that giving up fourth order convergence to smooth enough functions, can result in better convergence to functions with fewer derivatives.

When a simple *strictly local* method is required we would recommend instead method H. Clearly the norm of the latter interpolation operator can be bounded in terms of the local mesh ratio, $m_n = \max\{h_i/h_j : |i - j| = 1\}$, and independently of the number of knots. Marsden [8] has shown that this property fails to hold for several of the C^2 cubic spline interpolants we consider. We remind the reader that in an adaptive curve fitting setting the local mesh ratio may get very large, and the norm of all of the interpolation operators discussed here, including the strictly local ones, large with it. However the norm of the interpolation operator can be kept bounded if one chooses the nodes of interpolation after the knots of the spline, or replaces interpolation by quasi interpolation. (See de Boor [2, particularly pp.191-196 and pp.208-213].)

4

Method B fits a C^2 function and has fourth order convergence when the data comes from a sufficiently smooth function. Define

$$W_{j,\infty}[a,b] = \left\{ f \in C^{j-1}[a,b] : f^{(j-1)} \text{ absolutely continuous and} \right.$$

$$\left. f^{(j)} \in L_\infty[a,b] \right\}$$

$$= \left\{ C^{(j-1)} \text{ functions with } f^{(j-1)} \text{ Lipshitz} \right\}$$

Method B's performance for $W_{4,\infty}$ functions, is on average very slightly worse than the best of the other methods. However, it performs significantly better than the other C^2 fourth order methods on functions of lower smoothness. Of course, the various C^2 cubic interpolatory splines will differ greatly only in the first and last few intervals. However our contention is that the differences in those few intervals are important. This is particularly so since adapting a standard cubic spline code to end conditions B is trivial, and the extra computational expense is at worst five or six *flops*.

Different cubic spline fits to RPN-14 data
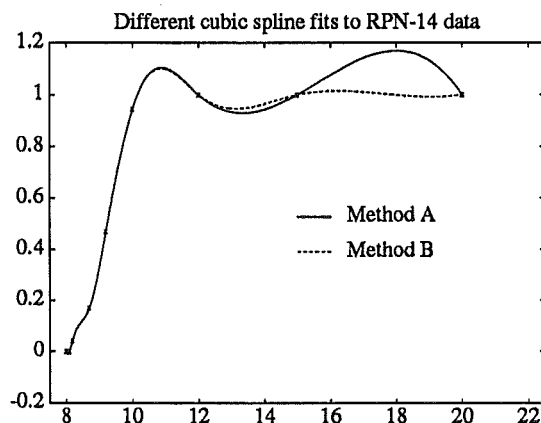
2 Detailed Results

In this section we present the results of our computations of exact error estimates and operator norms.

2.1 Differences between the C^2 and C^1 methods in the middle intervals.

We wish to emphasize the point made at the end of the previous section, that the differences between the various C^2 interpolants are generally only significant in the first and last few intervals. For example Kershaw [6] gives matrix estimates which, for a reasonably general set of end conditions, can be used to show the geometric decay of the influence of the end conditions, on the value of $s(x)$, as the number of knots between x and the endpoints grows.

To give a particular instance, if one considers the middle two intervals of a 30 subinterval uniform mesh then the calculated least constant C_4 for which

$$\sup_{14h \leq x \leq 16h} |f(x) - s(x)| \leq C_4 h^4 \|f^{(4)}\|_{[0,30h]},$$

for each of the C^2 methods, agrees to 8 significant digits with the value 5/384 appropriate for cardinal interpolation on an infinite mesh. (For the infinite problem see for example Powell [9]). Interestingly 5/384 is also the optimal constant in the bound $\|f - s\| \leq C h^4 \|f^{(4)}\|$ for the error in complete cubic spline interpolation where one has extra first derivative information at the end points (see Hall [4], and Hall and Meyer [5]). In contrast the strictly local methods can vary in their behavior in the middle intervals. Results are shown in table 1 below. The values under the optimal column are values that can be achieved by a, possibly non-cubic, spline interpolation method. That these values are lower bounds can be seen from the behaviour of the Euler splines $\mathcal{E}_k$ (see Schoenberg [10]).

Table 1: Error constant in middle of 30 interval uniform mesh

Derivative	Optimal	C^2 methods	Method G	Method H
1	$1/2 = .5$	.7745	.6250	.6875
2	$1/8 = .125$	.1623	.1406	.1517
3	$1/24 \approx .0416$	.0431	.0468	.0468
4	$5/384 \approx .0130$	.0130	undefined	.0234

It is apparent that the C^2 methods do somewhat better than the strictly local methods when f is smooth.

2.2   Results for pseudo-random meshes

Fix for the moment the mesh t and the smoothness class $C^j[t_0, t_n]$, $1 \leq j \leq 4$. Then for each $x \in [t_0, t_n]$, and each method α which reproduces π_{j-1}, we can compute the *pointwise error multiplier* $K(\alpha; j, x)$ which is the smallest number for which the bound

$$|f(x) - s(x)| \leq K(\alpha; j, x) \|f^{(j)}\|_\infty,$$

holds for all $f \in C^j[t_0, t_n]$. For the interpolants considered here the same bound holds for $W_{j,\infty}[t_0, t_n]$. Then the *error constant* $C_{\alpha,j}$ is defined to be the smallest number for which the relationship

$$\|f - s\|_\infty \leq C_{\alpha,j} \|f^{(j)}\|_\infty, \quad \forall f \in C^j[t_0, t_n],$$

holds. Clearly $C_{\alpha,j} = \sup_{x \in [t_0, t_n]} K(j, x)$. We can compare interpolation methods by comparing the error constants $C_{\alpha,j}$. Using method A as a standard we compute for each fixed mesh, and method α, the *relative error constant*
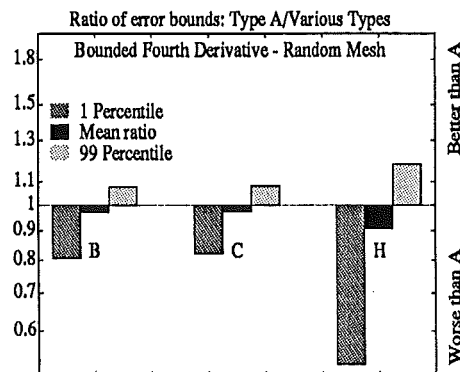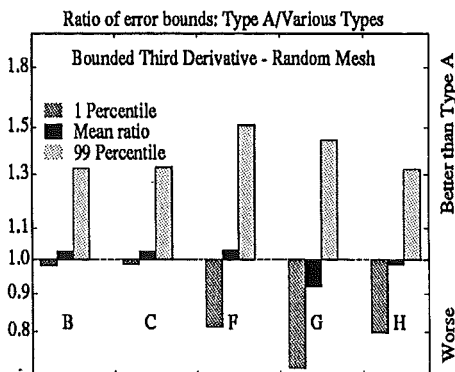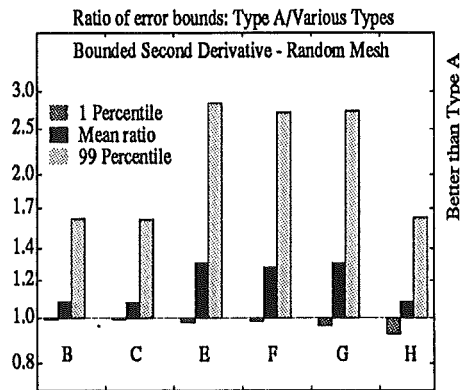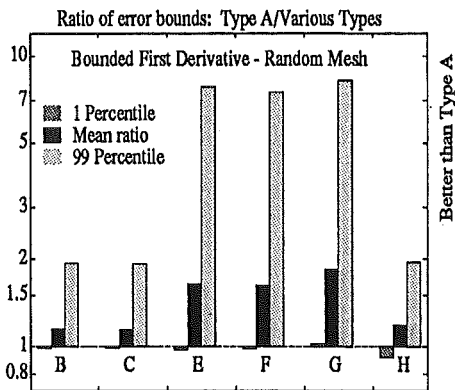
$$e_{\alpha,j} = C_{A,j}/C_{\alpha,j}.$$

6

The performance of the interpolants on nonuniform meshes was compared by conducting 5000 pseudo-random trials. In each trial 9 numbers were generated uniformly at random in $[0, 1]$, then sorted and scaled to obtain a mesh t : $0 = t_0 < t_1 < \ldots < t_8 = 1$. Then *relative error constants* were computed numerically. The resulting 5000 relative error constants were then sorted and the 1 percentile, mean, and 99 percentile points displayed in a bar graph. A logarithmic scale was used so that *relative error constants* of κ and $1/\kappa$ have the same visual impact.

2.2.1 Comparison of the overall error bounds

In the first two graphs below we observe that the $\mathcal{O}(\delta^4)$ methods do not do as well as the lower order methods for these not very smooth functions. This is perhaps a consequence of the reproduction of cubics. Method A does particularly badly. Intuitively, lacking knots at t_1 and t_{n-1}, it cannot be as flexible or local as the other methods.

In the next two graphs we note that the global methods do better than the strictly local methods for these smoother functions. Method A, the not-a-knot spline, does the best of any of the methods on $W_{4,\infty}$ functions.

Ratio of error bounds: Type A/Various Types — Bounded First Derivative - Random Mesh

Ratio of error bounds: Type A/Various Types — Bounded Second Derivative - Random Mesh

Ratio of error bounds: Type A/Various Types — Bounded Third Derivative - Random Mesh

Ratio of error bounds: Type A/Various Types — Bounded Fourth Derivative - Random Mesh

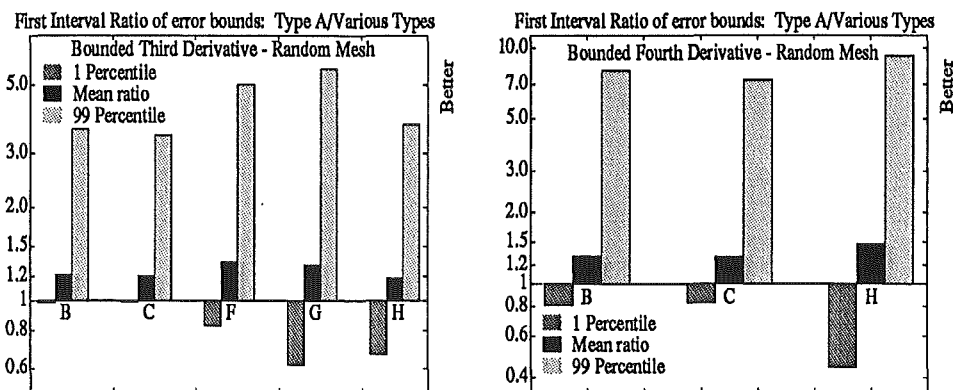### 2.2.2 Comparison of the first interval error bounds

In this section we compare the error bounds for the first interval only. Thus the first interval error constant $C'_{\alpha,j}$ is defined as the smallest number for which the bound

$$\sup_{[t_0,t_1]} |f(x) - s(x)| \leq C'_{\alpha,j} \|f^{(j)}\|_\infty, \quad \forall f \in C^j[t_0, t_n],$$

holds. Clearly

$$C'_{\alpha,j} = \sup_{[t_0,t_1]} K(\alpha; j, x).$$

We first compute these first interval error constants, and then the corresponding relative error constant $e'_{\alpha,j} = C'_{A,j}/C'_{\alpha,j}$. The results from numerical experiments were graphed as in the last section. For the first and second derivative bounds the first interval results were very much like the overall results. These graphs were therefore omitted. The first interval results for third and fourth derivative bounds are more extreme than the overall results and appear below.

First Interval Ratio of error bounds: Type A/Various Types

First Interval Ratio of error bounds: Type A/Various Types

2.3 Comparison of operator norms

We also calculated the norm of L, and of the derived projector L', for each of the methods and random 8 interval meshes. Bar graphs showing the 1 percentile, mean and 99 percentile points over 20,000 trials are shown below. We remind the reader that the norm of L cannot be bounded independent of the of the mesh ratio (see de Boor [2, pp.209-214] and the references there); the norm of L' (at least for methods A, B and C) can be bounded in terms of the mesh ratio in the first and last two intervals (see [1]), and that the norm of L'' can be bounded independent of the mesh. Following the bar graphs is a table of the extreme values of $\|L''\|$ seen over 20,000 random trials with 28 interval meshes.
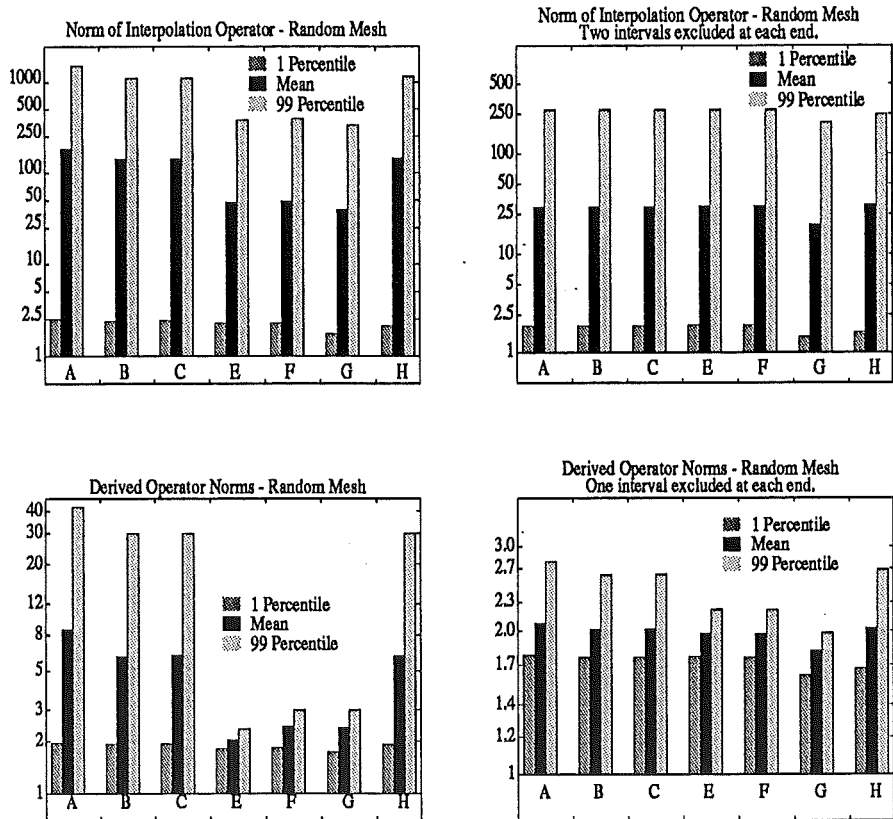
Norm of Interpolation Operator - Random Mesh

Norm of Interpolation Operator - Random Mesh
Two intervals excluded at each end.

Derived Operator Norms - Random Mesh

Derived Operator Norms - Random Mesh
One interval excluded at each end.

Table 2: Norm of second derived projector–random mesh
Maximum value observed over 20,000 trials

Method A	Method B	Method C	Method E
7.8919	4.9699	4.9652	2.6599

These norms represent a quantitative measure of the tendency of the various methods to introduce *spurious bumps and wiggles* in the fitted curve. We emphasise that, for the C^2 methods, the fitted curves will differ very little in the middle intervals, so that any large difference in operator norms corresponds to differing behaviour in the first few or last few intervals.
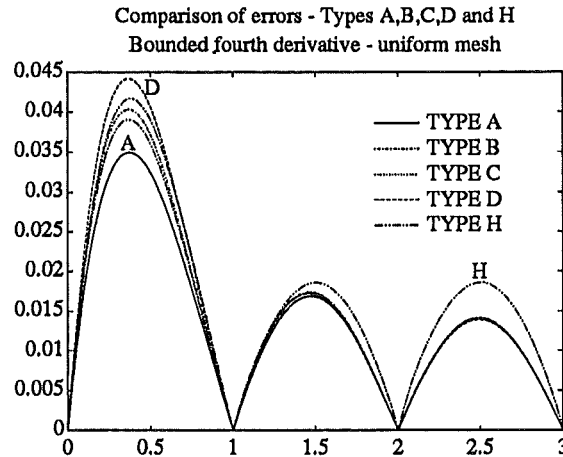
2.4   Results for a uniform mesh

The four graphs in figure 1 below show the pointwise error multipliers $K(j, x)$ in the first 3 intervals of a uniform mesh of 29 intervals, for each of the methods and the four smoothness classes. We do not show the plot for the interior subintervals where all the C^2 methods are practically indistinguishable. This is

9

Figure 1: Pointwise error multipliers K(j,x)

Comparison of error bounds - Types A,B,C, and D
Bounded first derivative - uniform mesh

Comparison of error bounds - Types A,E,F,G and H
Bounded first derivative - uniform mesh

Comparison of error bounds - Types A,B,C, and D
Bounded second derivative - uniform mesh

Comparison of error bounds - Types A,E,F,G and H
Bounded second derivative - uniform mesh

Comparison of error bounds - Types A,B,C, and D
Bounded third derivative - uniform mesh

Comparison of error bounds - Types A,F,G and H
Bounded third derivative - uniform mesh

10

to be expected because of the semi-local nature of cubic splines.

Comparison of errors - Types A,B,C,D and H
Bounded fourth derivative - uniform mesh

We also calculated the norms of s and the derived projectors s' and s'' for uniform meshes of various sizes, with end intervals included and excluded. The results are shown in tables 3-8 below

The entries in the first of each pair of tables are the usual operator norms,

$$\|L\| = \sup_{f \neq 0} \|Lf\|_{[t_0,t_n]}/\|f\|_{[t_0,t_n]}$$

while for those in the second the end intervals are excluded in the numerator. Thus,

$$\|L\|' = \sup_{f \neq 0} \|Lf\|_{[t_1,t_{n-1}]}/\|f\|_{[t_0,t_n]}.$$

Our interpretation of these results is as follows. For these uniform meshes the various C^2 methods differ very little when the end intervals are excluded[1]. Thus once again we see that the end-conditions make a difference only near the end-points. Also, as expected, the operator norms for a particular method hardly change as the the number of intervals in the uniform mesh is increased beyond 8. The error curves for $W_{4,\infty}$ functions show the C^2 methods doing better than the strictly local method H in the interior subintervals.

3 Mathematics underlying the computations

3.1 Calculation of the optimal error bounds.

Fix for the moment the mesh t, the cubic spline interpolant L, and $\xi \in [t_0, t_n] = [a, b]$. Let

$$E(f, \xi) = f(\xi) - L(f, \xi)$$

Suppose that this error functional anihilates π_j for some $0 \leq j \leq 3$. The application of the Peano kernel theorem (Davis [3], Powell [9]) shows that if

---

[1] In fact the end-intervals deleted operator norm rapidly approaches the value $(1+3\sqrt{3})/4 \approx 1.549038..$ valid for C^2 cubic spline interpolation on an infinite uniform mesh (see Powell [9].)

Table 3: Norm of spline operator end intervals included

n	A	B	C	D	E	F	G	H
8	1.97098	1.67836	1.71712	2.72960	1.53579	1.53345	1.25000	1.63113
12	1.97164	1.67843	1.71725	2.73294	1.54808	1.54793	1.25000	1.63113
16	1.97164	1.67843	1.71725	2.73296	1.54897	1.54896	1.25000	1.63113
20	1.97164	1.67843	1.71725	2.73296	1.54903	1.54903	1.25000	1.63113

Table 4: Norm of spline operator end intervals excluded

n	A	B	C	D	E	F	G	H
8	1.51768	1.52316	1.52243	1.54745	1.53579	1.53345	1.25000	1.38490
12	1.54666	1.54719	1.54712	1.54903	1.54808	1.54793	1.25000	1.38490
16	1.54887	1.54890	1.54890	1.54904	1.54897	1.54896	1.25000	1.38490
20	1.54902	1.54903	1.54903	1.54904	1.54903	1.54903	1.25000	1.38490

Table 5: Norm of first derived projector end intervals included

n	A	B	C	D	E	F	G	H
8	4.30769	3.33333	3.46392	6.78788	1.73196	2.00000	2.00000	3.33333
12	4.30939	3.33333	3.46410	6.79738	1.73205	2.00000	2.00000	3.33333
16	4.30940	3.33333	3.46410	6.79743	1.73205	2.00000	2.00000	3.33333
20	4.30940	3.33333	3.46410	6.79743	1.73205	2.00000	2.00000	3.33333

Table 6: Norm of first derived projector end intervals excluded

n	A	B	C	D	E	F	G	H
8	1.73120	1.69759	1.69643	2.27352	1.71428	1.71134	1.50000	1.58333
12	1.73205	1.72958	1.72949	2.27669	1.73077	1.73057	1.50000	1.58333
16	1.73205	1.73187	1.73187	2.27671	1.73196	1.73194	1.50000	1.58333
20	1.73205	1.73204	1.73204	2.27671	1.73204	1.73204	1.50000	1.58333

Table 7: Norm of second derived projector end intervals included

n	Method A	Method B	Method C	Method D	Method E
8	3.05846	2.31680	2.33333	5.53712	1.99244
12	3.05920	2.31689	2.33333	5.54195	1.99946
16	3.05921	2.31689	2.33333	5.54198	1.99996
20	3.05921	2.31689	2.33333	5.54198	2.00000

Table 8: Norm of second derived projector end intervals excluded

n	Method A	Method B	Method C	Method D	Method E
8	1.97675	1.98322	1.98235	1.96058	1.99244
12	1.99838	1.99879	1.99874	1.99735	1.99946
16	1.99988	1.99991	1.99990	1.99981	1.99996
20	1.99999	1.99999	1.99999	1.99999	2.00000

$f \in C^{j+1}[a,b]$

$$E(f,\xi) = \int_a^b f^{(j+1)}(t)K(t)dt \tag{10}$$

where

$$K(t) = \frac{1}{j!}E_x((x-t)_+^j, \xi) \tag{11}$$

and the notation $E_x((x-t)_+^j, \xi)$ means that the functional $E(\cdot, \xi)$ is applied to $(x-t)_+^j$ considered as a function of x. Because *in the case we consider* K will have only a finite number of sign changes it follows easily from (10), (11) and smoothing arguments that

$$\sup_{\{f \in C^{j+1}[a,b]: \|f^{(j+1)}\|_\infty \le 1\}} |E(f,\xi)| = \|K\|_1$$

or what is equivalent that the least value of C for which the relationship

$$|E(f,\xi)| \le C\|f^{(j+1)}\|_\infty$$

holds for all $f \in C^{j+1}[a,b]$ is $C = \|K\|_1$. These conclusions also hold when $C^{j+1}[a,b]$ is replaced by $W_{j+1,\infty}$.

It only remains to discuss how one can calculate K and $\|K\|_1$ numerically. Firstly we let $\{\ell_i\}_{i=0}^n$ be the cardinal splines corresponding to the interpolant L. That is ℓ_i is the cubic spline interpolant $L(f)$ when $f(t_k) = \delta_{ik}$. Then for any f, ξ

$$L(f,\xi) = \sum_{i=0}^N f(t_i)\ell_i(\xi)$$

and

$$E(f,\xi) = f(\xi) - \sum_{i=0}^n f(t_i)\ell_i(\xi)$$

Substituting from (11) it follows that

$$j!K(t) = (\xi - t)_+^j - \sum_{i=0}^n (t_i - t)_+^j \ell_i(\xi). \tag{12}$$

Hence for each $0 \le j \le 3$, $K(t)$ is a spline of degree j with possible knots at ξ and the t_i's, whose coeficients we can easily calculate. $\|K\|_1$ can then be calculated numerically by using a cubic root finder to find the zeroes of K, and Simpson's rule to integrate $|K(t)|$ exactly over each subinterval within which it reduces to a cubic polynomial.

3.2 Computation of $\|L\|_\infty$ and $\|L'\|_\infty$

The methods of computing $\|L\|_\infty$ and $\|L'\|_\infty$ are essentially the same so we will only give the details of the computation of $\|L'\|_\infty$.

Let Lf denote one of the cubic spline interpolants under consideration applied to the function f at the nodes $t = t_0 < t_1 < \ldots < t_n$. Then assuming the map L is exact for polynomials of degree $j - 1$, the derived projector $L^{(j)}$ given by

$$L^{(j)}(D^j f) = D^j (Lf)$$

is well defined. We define, as is usual, the operator norm

$$\|L^{(j)}\|_\infty = \sup_{g \in C[t_0, t_n]: \|g\|_\infty \leq 1} \|L^{(j)}(g)\|$$

For convenience denote $(Lf)(x)$ by $s(x)$. Assume L is exact for constants. Then considering the matrix system expressing the first derivatives of s at the knots in terms of the data, we see that the linear map from g to $L'(g)$ may be expressed as the composition of three linear maps

$$L' = E \circ P \circ S \tag{13}$$

where $S : C[t_0, t_n] \to \mathbb{R}$ is the map taking $g = f'$ to the n-vector with i^{th} component

$$f[t_i, t_{i+1}] = \frac{1}{t_{i+1} - t_i} \int_{t_i}^{t_{i+1}} g(t) dt.$$

$P : \mathbb{R}^n \to \mathbb{R}^{2n+1}$ is the map taking the vector α to the $2n+1$ vector β with i^{th} component

$$\beta_i = \begin{cases} \alpha_i, & i < n, \\ s'(t_{i-n}), & i \geq n. \end{cases}$$

and finally,

$$E : \mathbb{R}^{2n+1} \to C[t_0, t_n]$$

is the map taking β to the piecewise quadratic, s', with specified endpoint and average values on each subinterval.

Because L' receives only the information about g given to it by the map S we can rewrite (13) in a form more useful for computation. More precisely, as g ranges over

$$\{g \in C[t_0, t_n] : \|g\|_\infty \leq 1\},$$

$S(g)$ ranges over a set in $\mathbb{R}^n$ whose closure is the closed ℓ^∞ unit ball in $\mathbb{R}^n$. Hence (13) implies

$$\|L'\|_\infty = \sup_{\Omega = \{\alpha : \|\alpha\|_\infty \leq 1\}} \|(E \circ P)(\alpha)\|_\infty \tag{14}$$

Finally since $E \circ P$ is a linear function and $\|\cdot\|_\infty$ a convex function the function in (14) is convex and achieves its supremum at an extreme point of the convex set Ω. This shows that for each fixed mesh t, $\|L'\|_\infty$ can be found by a process of exhaustive search over the 2^n extreme points of Ω, computing for each extreme point the corresponding value of $\|s'\|_\infty$. Indeed, since the sign of the first slope

14

can be fixed, without loss of generality, the search can be restricted to 2^{n-1} extreme points. We note that it is known [1] that if L reproduces cubics then $\|L'\|_\infty$ cannot be bounded independently of $\mathbf{t}$.

The computation of $\|L\|_\infty$ is analogous to that of $\|L'\|_\infty$. In this case Ω is the ℓ^∞ unit ball in $\mathbb{R}^{n+1}$, extreme points of which correspond to function values $[f(t_0), \ldots, f(t_n)]^T$ to be interpolated. $\|L\|_\infty$ is computed by computing $\|s\|_\infty$ for 2^n of these 2^{n+1} extreme points.

3.3 Computation of $\|L''\|_\infty$

When L is exact for linear polynomials the map L'', from f'' to $L''(f'')$, is a well defined linear map. It can be written as the composition of three linear maps

$$L'' = E \circ P \circ S.$$

Here S maps f'' to the $n-1$ vector of second divided differences $\mathbf{b}$ with

$$b_i = f[t_i, t_{i+1}, t_{i+2}] = \frac{1}{t_{i+2} - t_i} \int_{t_i}^{t_{i+2}} N_{i,2}(x) f''(x) dx.$$

P maps $\mathbf{b}$ to the vector of second derivatives $\sigma = (s''(t_i))_{i=0}^n$ and therefore has the form $\sigma = D\mathbf{b}$ for some $(n+1) \times (n-1)$ matrix D. Finally E maps the vector of second derivative values σ to the piecewise linear interpolant s''. [2]

Since s'' is piecewise linear $\|s''\|_\infty$ corresponds to a value of s'' at one of the knots. Hence

$$\|L''\|_\infty = \max_{0 \le i \le n} \sup_{\{f : f \in L^2_\infty \text{ and } \|f''\|_\infty \le 1\}} |\sum_j d_{ij} f[t_j, t_{j+1}, t_{j+2}]|$$

Since values of f'' in one interval can affect two of the second diffences b_i, we cannot simply choose the differences in a bang-bang way as we did in computing $\|L\|$ and $\|L'\|$. However, fixing i and writing e_j for $d_{i,j}$

$$
\begin{aligned}
s''(f; t_i) &= \sum_{j=0}^{n-2} e_j f[t_j, t_{j+1}, t_{j+2}] \\
&= \sum_{j=0}^{n-2} \frac{e_j}{t_{j+2} - t_j} \int_{t_j}^{t_{j+2}} N_{j,2}(x) f''(x) dx \\
&= \sum_{j=0}^{n-2} e_j \left[(\frac{t_{j+1} - t_j}{t_{j+2} - t_j}) \int_0^1 x f''(t_j + (t_{j+1} - t_j)x) dx + \right. \\
&\qquad \left. + (\frac{t_{j+2} - t_{j+1}}{t_{j+2} - t_j}) \int_0^1 (1 - x) f''(t_{j+1} + (t_{j+2} - t_{j+1})x) dx \right] \\
&= \omega_{0,2} + \sum_{j=1}^{n-2} (\omega_{j,1} \lambda_{j,1} + \omega_{j,2} \lambda_{j,2}) + \omega_{n-1,1} \lambda_{n-1,1}
\end{aligned}
$$

___
[2] For the C^2 methods considered here the system for σ takes the form $A\sigma = B\mathbf{b}$, with A^{-1} and B bounded in the infinity norm (see for example [1, pp.905-906], for explicit formulations). Hence $D = A^{-1}B$ is bounded and the relationship between second divided differences and second derivatives implies immediately that $\|L''\|_\infty \le \frac{1}{2}\|D\|_\infty < \infty$. This simple analysis provides for example the (not tight) upper bounds $\|L''_C\|_\infty \le 8$ and $\|L''_E\|_\infty \le 3$.

15

where

$$\omega_{j,2} = e_j \frac{(t_{j+1} - t_j)}{(t_{j+2} - t_j)}, \quad \omega_{j,1} = e_{j-1} \frac{(t_{j+1} - t_j)}{(t_{j+1} - t_{j-1})},$$

and

$$\lambda_j = \left[\begin{array}{c} \int_0^1 (1-x)f''(t_j + (t_{j+1} - t_j)x)dx \\ \int_0^1 x f''(t_j + (t_{j+1} - t_j)x)dx \end{array} \right] = \left[\begin{array}{c} \int_0^1 (1-x)g_j(x)dx \\ \int_0^1 x g_j(x)dx \end{array} \right].$$

Hence defining Λ as the closed convex set in $\mathbb{R}^2$

$$\Lambda = \left\{ \left[\begin{array}{c} \lambda_1 \\ \lambda_2 \end{array} \right] = \left[\begin{array}{c} \int_0^1 (1-x)g_j(x)dx \\ \int_0^1 x g_j(x)dx \end{array} \right] : \|g\|_\infty \leq 1 \right\}$$

the maximum of $s''(t_i)$ over functions f with $\|f''\| \leq 1$ equals

$$\max_{\lambda_j \in \Lambda} \omega_{0,2}\lambda_{0,2} + \left(\sum_{j=1}^{n-2} (\omega_{j,1}\lambda_{j,1} + \omega_{j,2}\lambda_{j,2}) \right) + \omega_{n-1,1}\lambda_{n-1,1} \qquad (15)$$

Here the numbers $\omega_{j,k}$ depend only on the mesh t and the matrix D. Hence the problem (15) separates into the sum of n two dimensional subproblems

$$\max_{\lambda \in \Lambda} \omega_{0,2}\lambda_2 + \left(\sum_{j=1}^{n-2} \max_{\lambda \in \Lambda} (\omega_{j,1}\lambda_1 + \omega_{j,2}\lambda_2) \right) + \max_{\lambda \in \Lambda} \omega_{n-1,1}\lambda_1$$

since Λ is convex the maxima in the subproblems occur at extreme points of Λ. Indeed the following lemma shows that the two dimensional subproblems are trivial to solve.

Lemma 1 *The set Λ defined above is the closed convex set with boundary the curves*

$$\gamma_1 = \left\{ \lambda = \left[\begin{array}{c} \alpha^2 - \frac{1}{2} \\ 2\alpha - \frac{1}{2} - \alpha^2 \end{array} \right] : 0 < \alpha < 1 \right\}$$

and

$$\gamma_2 = \left\{ \lambda = \left[\begin{array}{c} 2\alpha - \frac{1}{2} - \alpha^2 \\ \alpha^2 - \frac{1}{2} \end{array} \right] : 1 > \alpha > 0 \right\}$$

together with the points $A = \left(-\frac{1}{2}, -\frac{1}{2}\right)$ and $B = \left(\frac{1}{2}, \frac{1}{2}\right)$.

Proof:- Omitted.

Having defined Λ we see that for each non-zero ω, $\omega^T \lambda$ is maximized at a unique point on the boundary of Λ. One can easily calculate that point. For example when $\omega_1 > 0$, $\omega_2 < 0$ the maximum corresponds to the unique point in the interior of the lower boundary where ω is perpendicular to the tangent. Hence occurs at the point $\alpha = \frac{\omega_1}{(\omega_1 - \omega_2)}$. Other sign patterns for ω are similarly dealt with. This leads to a procedure for solving the two dimensional subproblems and hence, with extra code, for computing $\|L''\|$.

References

[1] R. K. Beatson, *On the convergence of some cubic spline interpolation schemes*, SIAM J. Numer. Anal., **23** (1986), pp. 903-912.

[2] C. de Boor *A Practical Guide to Splines*, Springer-Verlag, New York, 1978.

[3] P. J. Davis *Interpolation and Approximation*, Blaisdell, New York, 1963.

[4] C. A. Hall *On error bounds for cubic spline interpolation*, J. Approximation Theory, **1** (1968), 209-218.

[5] C. A . Hall and W. W. Meyer *Optimal error bounds for cubic spline interpolation*, J. Approximation Theory, **16** (1976),105-122.

[6] D. Kershaw, *Inequalities on the elements of a certain tridiagonal matrix*, Math. Comp., **24** (1970), pp.155-158.

[7] D. Kershaw, *Two interpolatory cubic splines*, J. Inst. Maths Applics **11**(1973),329-333.

[8] M. J. Marsden *Cubic spline interpolation of continuous functions*, J. Approximation Theory, **10** (1974),103-111.

[9] M. J. D. Powell *Approximation theory and methods*, Cambridge University Press, Cambridge, 1981.

[10] I. J. Schoenberg *The elementary cases of Landau's problem of inequalities between derivatives*. The American Mathematical Monthly, **80** (1973), 121-158.