Some Examples of Quasi-Interpolants Constructed from Local Spline Projectors

Byung-Gook Lee, Tom Lyche, and Knut Mørken

Abstract. We give a recipe for deriving local spline approximation methods which reproduce the whole spline space. The methods are obtained by solving a series of local approximation problems. Examples of specific quadratic and cubic approximation methods are given.

§1. Introduction

Many applications of splines make use of some approximation method to produce a spline function from given discrete data. Popular methods include interpolation and least squares approximation. However, both of these methods require solution of a linear system of equations with as many unknowns as the dimension of the spline space, and are therefore not suitable for real-time processing of large streams of data. For this purpose local methods, which determine spline coefficients by using only local information, are more suitable. To ensure good approximation properties it is important that the methods reproduce polynomials and preferably the functions in the given spline space. A method based on derivative information was constructed in [rndeB], while a more general class was studied in [rnLS]. In order to reproduce the spline space, the local information of the methods in [rnLS] was restricted to lie in one knot interval. In this paper we remove this restriction. We then discuss some specific approximation methods for quadratic and cubic splines.

We use B-splines as a basis for splines and denote the i^{th} B-spline of degree d with knots $\boldsymbol{t}$ by $B_{i,d} = B_{i,d,\boldsymbol{t}}$, and the linear space spanned by these B-splines by $\mathbb{S}_{d,\boldsymbol{t}}$.

Mathematical Methods in CAGD: Oslo 2000
Tom Lyche and Larry L. Schumaker (eds.), pp. 243–252.

ISBN 0-8265-xxxx-x.

§2. A General Construction of Quasi-interpolants

Given a function f, the basic problem of spline approximation is to determine B-spline coefficients $(c_i)_{i=1}^n$ such that

$$Pf = \sum_{i=1}^{n} c_i B_{i,d}$$

is a reasonable approximation to f. The basic challenge is therefore to devise a procedure for determining the B-spline coefficients. We assume that f is defined on an interval $[a, b]$, and that we have selected a space of splines $\mathbb{S}_{d,\boldsymbol{t}}$ defined on $[a, b]$ (i.e., so that $\boldsymbol{t} = (t_j)_{j=1}^{n+d+1}$ is nondecreasing with $t_{d+1} = a$ and $t_{n+1} = b$). We fix k and propose the following procedure for determining c_k:

(i) Choose a local interval $I = (t_\mu, t_\nu)$ with the property that I intersects the (interior of the) support of $B_{k,d}$:

$$I \cap (t_k, t_{k+d+1}) \neq \emptyset.$$

Denote the restriction of the space $\mathbb{S}_{d,\boldsymbol{t}}$ to the interval I by $\mathbb{S}_{d,\boldsymbol{t},I}$, *i.e.*,

$$\mathbb{S}_{d,\boldsymbol{t},I} = \operatorname{span}\{B_{\mu-d,d}, \dots, B_{\nu-1,d}\}.$$

(ii) Choose some local approximation method P_I with the property that

$$P_I g = g, \qquad \text{for all } g \text{ in } \mathbb{S}_{d,\boldsymbol{t},I}. \tag{1}$$

(iii) Let f_I denote the restriction of f to the interval I. Then there exist B-spline coefficients $(b_i)_{i=\mu-d}^{\nu-1}$ such that $P_I f_I = \sum_{i=\mu-d}^{\nu-1} b_i B_{i,d}$. Note that $\mu - d \leq k \leq \nu - 1$ since $\operatorname{supp} B_{k,d}$ intersects I.

(iv) Set $c_k = b_k$.

When determining c_k, this procedure gives us the freedom to restrict our attention to a local subinterval $I = [t_\mu, t_\nu]$ of our choice. By doing this we may reduce the complexity of the problem. Secondly, we have the freedom to choose the local approximation method P_I. Typical choices will be interpolation, least squares approximation, or a smoothing spline. As we shall see in Lemma pnrepro, the local condition (1) ensures that if f is a spline in $\mathbb{S}_{d,\boldsymbol{t}}$, it will be reproduced by Pf. In certain situations, other conditions may be more natural, but we will not pursue this any further here.

We first ascertain that the local reproduction condition leads to global reproduction of $\mathbb{S}_{d,\boldsymbol{t}}$.

Lemma 1. *The spline approximation Pf determined by steps (i)–(iv) above has the property that $Pf = f$ for all f in the spline space $\mathbb{S}_{d,\boldsymbol{t}}$.*

Proof: Suppose that $f = \sum_{i=1}^{n} \hat{c}_i B_{i,d}$ for certain coefficients $(\hat{c}_i)_{i=1}^{n}$; we must show that if $Pf = \sum_{i=1}^{n} c_i B_{i,d}$ then $c_i = \hat{c}_i$. We note that $f_I = \sum_{i=\mu-d}^{\nu-1} \hat{c}_i B_{i,d}$, so f_I is clearly in $\mathbb{S}_{d,\boldsymbol{t},I}$. Therefore, by assumption, we have

$$\sum_{i=\mu-d}^{\nu-1} b_i B_{i,d} = P_I f_I = f_I = \sum_{i=\mu-d}^{\nu-1} \hat{c}_i B_{i,d},$$

so $b_i = \hat{c}_i$ for $i = \mu - d, \ldots, \nu - 1$, and in particular $b_k = \hat{c}_k$. But remember that c_k is chosen equal to b_k so we therefore have $c_k = \hat{c}_k$, as required. □

To emphasize the dependence on f, the coefficient c_k is often written $c_k = \lambda_k f$, with λ_k some linear functional. The following lemma gives an explicit formula for the coefficient $\lambda_k f$ in the case where it is a combination of given linear functionals $\lambda_{k,1}, \ldots, \lambda_{k,\nu-\mu+d}$.

Lemma 2. *Suppose that the coefficient c_k of Pf is chosen as*

$$c_k = \frac{\det\bigl(\boldsymbol{\lambda} B_{\mu-d}, \ldots, \boldsymbol{\lambda} B_{k-1}, \boldsymbol{\lambda} f, \boldsymbol{\lambda} B_{k+1}, \ldots, \boldsymbol{\lambda} B_{\nu-1}\bigr)}{\det\bigl(\boldsymbol{\lambda} B_{\mu-d}, \ldots, \boldsymbol{\lambda} b_{\nu-1}\bigr)}, \tag{2}$$

where $\boldsymbol{\lambda} B_j$ denotes the column vector

$$\boldsymbol{\lambda} B_j = (\lambda_{k,1} B_{j,d}, \ldots, \lambda_{k,\nu-\mu+d} B_{j,d})^T$$

and $\lambda_{k,1}, \ldots, \lambda_{k,\nu-\mu+d}$ are linear functionals defined on $\mathbb{S}_{d,\boldsymbol{t}}$ such that the denominator in (2) is nonzero. Then $Pf = f$ for all f in $\mathbb{S}_{d,\boldsymbol{t}}$.

Proof: We choose a special local approximation operator in step (ii) of the construction procedure, namely the one that maps f_I to the spline that solves the generalized interpolation problem

$$\lambda_{k,i}(P_I f_I) = \lambda_{k,i} f_I, \qquad \text{for } i = 1, \ldots, \nu - \mu + d. \tag{3}$$

Expressing $P_I f_I$ in terms of B-splines as $P_I f_I = \sum_{j=\mu-d}^{\nu-1} b_j B_{j,d}$ and inserting this in (3) leads to the linear system of equations

$$\sum_{j=\mu-d}^{\nu-1} (\lambda_{k,i} B_{j,d}) b_j = \lambda_{k,i} f_I, \qquad i = 1, \ldots, \nu - \mu + d. \tag{4}$$

Solving this system for b_k by Cramer's rule and setting $c_k = b_k$ yields the formula (2), and the solution is unique since the denominator in (2) is nonzero. The uniqueness also implies that (1) holds. □

A general class of approximation methods are obtained by letting P_I be given as point functionals of the form

$$\lambda_{k,j} f = f(x_{k,j}), \quad j = 1, \ldots, m_k, \tag{5}$$

where $m_k = \nu - \mu + d$ and $x_{k,1}, \ldots, x_{k,m_k}$ are given points. With this choice, it is well known (see page 200 of [rndeBbook]) that if

$$B_{\mu-d-1+j,d}(x_{k,j}) > 0, \qquad j = 1, \ldots, m_k, \tag{6}$$

then the denominator in (2) is nonzero and Lemma 2 can be applied. Expanding the numerator in (2), we obtain c_k in the form

$$c_k = \lambda_k f = \sum_{j=1}^{m_k} w_{k,j} f(x_{k,j}), \tag{7}$$

for some vector $\boldsymbol{w}_k = (w_{k,j})$. Equivalently, we can find $\boldsymbol{w}_k$ by solving the linear system

$$\delta_{i,k} = \lambda_k(B_{i,d}) = \sum_{j=1}^{m_k} w_{k,j} B_{i,d}(x_{k,j}) \qquad \text{for } i = \mu - d, \, \ldots, \, \nu - 1, \tag{8}$$

where $\delta_{i,k} = 1$ if $i = k$ and zero otherwise, as usual. In practice one would usually determine c_k numerically, either from (2), (4), or (8), except in special cases where the formulas are particularly simple.

Quasi-interpolants of this kind were studied in [rnLS]. However, there the data points $\{x_{k,j}\}_{j=1}^{m_k}$ are restricted to all lie in one subinterval $[t_l, t_{l+1}]$ of $[t_k, t_{k+d+1}]$.

There are standard ways to obtain error estimates for the kind of approximation methods developed here. Let us denote the total approximation by Pf, and suppose we have found a constant C (that may depend on the knots, but not on f) such that

$$\|Pf\| \le C\|f\|. \tag{9}$$

Here $\|f\|$ denotes the uniform norm on the interval $[a, b]$,

$$\|f\| = \max_{x\in[a,b]} |f(x)|.$$

¿From (9) it follows by a standard argument that

$$\|f - Pf\| \le (1 + C)\,\mathrm{dist}(f, \mathbb{S}_{d,\boldsymbol{t}}), \tag{10}$$

where $\mathrm{dist}(f, \mathbb{S}_{d,\boldsymbol{t}})$ denotes the quantity

$$\mathrm{dist}(f, \mathbb{S}_{d,\boldsymbol{t}}) = \inf_{g\in\mathbb{S}_{d,\boldsymbol{t}}} \|f - g\|.$$

§3. Examples: Projection Into a Given Spline Space

In this section we consider some examples in the case where the knots and the degree of the spline are given.

Example 3. *A quadratic spline.* Suppose $d = 2$ and the knots $\boldsymbol{t} = (t_j)$ are given. To determine c_k we choose a point $(x_{k,2})$ in the middle subinterval of the B-spline $B_{k,2}$. Thus $I = [t_\mu, t_\nu] = [t_{k+1}, t_{k+2}]$ so that we need $m_k = \nu - \mu + d = 3$ local interpolation points in I. The space $\mathbb{S}_{d,\boldsymbol{t},I}$ (quadratic polynomials on I) is spanned by the three B-splines $(B_{i,2})_{i=k-2}^{k}$ and the local operator P_I is the interpolation operator at the three points $(x_{k,j})_{j=1}^{3}$. Since $t_{k+1} = x_{k,1} < x_{k,2} < x_{k,3} = t_{k+2}$, it is clear that (6) holds and the system (4) becomes a simple 3×3 linear system of equations. The coefficient c_k can either be determined as the second of the three resulting coefficients, which is b_k with our labelling, or from (2). The result is

$$c_k = \lambda_k f = \frac{1}{2}\Big(-\theta_k^{-1} f(x_{k,1}) + \theta_k^{-1}(1+\theta_k)^2 f(x_{k,2}) - \theta_k f(x_{k,3})\Big), \qquad (11)$$

where

$$\theta_k = \frac{x_{k,3} - x_{k,2}}{x_{k,2} - x_{k,1}}.$$

The spline approximation $P_2 f = \sum_j \lambda_j f B_{j,2}$ reproduces the quadratic spline space, and from (10) we obtain

$$||f - P_2 f|| \le (3 + \rho)\,\mathrm{dist}(f, \mathbb{S}_{2,\boldsymbol{t}}),$$

where

$$\rho = \max_k \{\theta_k, \theta_k^{-1}\}.$$

This holds for any function f and with the special choice $x_{k,2} = (x_{k,1} + x_{k,3})/2$ then $\|f - P_2 f\| \le 4\,\mathrm{dist}(f, \mathbb{S}_{2,\boldsymbol{t}})$. With this special choice of $x_{k,2}$, this operator is classical, and the corresponding approximation method has been used to approximate functions on the sphere ([[redacted]rnST[redacted]]).

Example 4. *A quadratic spline based locally on 5 points.* Another possibility is to choose $[t_\mu, t_\nu] = [t_k, t_{k+3}]$. Then the local spline space $\mathbb{S}_{2,\boldsymbol{t},I}$ has dimension 5 and is spanned by the five B-splines $(B_{i,d})_{i=k-2}^{k+2}$. If we choose three extra points

$$x_{k,1} \in (t_k, t_{k+1}), \qquad x_{k,3} \in (t_{k+1}, t_{k+2}), \qquad x_{k,5} \in (t_{k+2}, t_{k+3})$$

in addition to the two interior knots $x_{k,2} = t_{k+1}$ and $x_{k,4} = t_{k+3}$, we can take P_I to be the operator corresponding to interpolation at the five points $(x_{k,i})_{i=1}^{5}$. Again it is easy to see that (6) holds and we choose c_k as the middle coefficient in (4).

Example 5. *A cubic spline based locally on 5 points.* Similar constructions are possible in the cubic case ($d = 3$). With k fixed, we choose the interval $I =$

$[t_{k+1}, t_{k+3}]$ which means that the local spline space has dimension 5. Again we determine $P_I f$ by interpolation, this time at the three knots $x_{k,1} = t_{k+1}$, $x_{k,3} = t_{k+2}$ and $x_{k,5} = t_{k+3}$ plus the two additional points $x_{k,2} \in (t_{k+1}, t_{k+2})$ and $x_{k,4} \in (t_{k+2}, t_{k+3})$. As before, (6) holds and c_k is chosen as the middle coefficient.

In most cases the formulas for the B-spline coefficients are rather complicated, but in the following case the expressions become quite simple: We choose $x_{k,2}$ as the midpoint between t_{k+1} and t_{k+2}, and $x_{k,4}$ as the midpoint between t_{k+2} and t_{k+4}. If we introduce the knot ratio $\theta = (t_{k+3} - t_{k+2})/(t_{k+2} - t_{k+1})$, the formula for $c_k = \lambda_k f$ becomes

$$\begin{aligned} c_k =& \frac{1}{9}\Big(\frac{1+2\theta}{\theta(1+\theta)} f(x_{k,1}) - 8\frac{1+2\theta}{\theta(1+\theta)} f(x_{k,2}) + (16 + 7\theta^{-1} + 7\theta) f(x_{k,3}) \\ & - 8\frac{\theta(2+\theta)}{1+\theta} f(x_{k,4}) + \frac{\theta(2+\theta)}{1+\theta} f(x_{k,5})\Big). \end{aligned}$$

§4. Approximation of Discrete Data by Quasi-interpolants

In the examples of quasi-interpolants in the previous section it was assumed that a spline space was given, and that we were to construct an approximation method that would project functions onto the spline space. In practice, a more common situation is that a set of discrete data points are given, and the challenge is to construct a spline approximation. In this case, we need to determine a suitable spline space (spline degree and knot vector) before we can compute the approximation, and the knots must be chosen somehow. We consider a class of methods which illustrate how quasi-interpolants of the above type can be used to solve approximation problems of this kind.

We start with an odd number m of data points $(x_j, y_j)_{j=1}^{m}$, where $m \geq 2d - 1$. We assume that the points are sampled from a function f defined on an interval $[a, b]$, and that the abscissae $\boldsymbol{x} = (x_j)_{j=1}^{m}$ satisfy $a = x_1 < x_2 < \ldots < x_m = b$. From $\boldsymbol{x}$ we form the knot vector

$$\boldsymbol{t} = (t_j)_{j=1}^{n+d+1} = (\overbrace{x_1, \ldots, x_1}^{d+1}, x_3, x_5, \ldots, x_{m-2}, \overbrace{x_m, \ldots, x_m}^{d+1}),$$

where $n = (m-1)/2 + d$. Note that the knots are related to the abscissae of the data points via the formula $t_j = x_{2(j-d)-1}$ for $j = d+1, \ldots, n+1$. We shall design a quasi-interpolant

$$P_d f = \sum_{k=1}^{n} \lambda_k f B_{k,d}$$

such that each $\lambda_k f$ depends on at most $2d - 1$ data points. To compute $c_k = \lambda_k f$ for k in the range $d \leq k \leq n - d + 1$, we set $I = [t_{k+1}, t_{k+d}]$. The restriction $\mathbb{S}_{d,\boldsymbol{t},I}$ of $\mathbb{S}_{d,\boldsymbol{t}}$ to this interval has dimension $m_k = 2d - 1$, but we also have $2d-1$ data points in I, namely the points $x_{k,j} = x_{2(k-d)+j}$ for $j = 1, \ldots, 2d-1$. We can therefore determine a local approximation by forcing a spline in $\mathbb{S}_{d,\boldsymbol{t},I}$ to interpolate these points and then choose the middle coefficient as c_k. The ends must be treated specially. One possibility is to use derivative information, but we do not consider this here. Instead, for $k = 1, \ldots, d-1$ we use the same local interval as for $k = d$, i.e., $I = [t_{d+1}, t_{2d}]$, which means that $\mathbb{S}_{d,\boldsymbol{t},I}$ has dimension $2d-1$ and we have $2d-1$ data points in I. We solve the local interpolation problem and choose $(c_k)_{k=1}^{d-1}$ as the first $d-1$ coefficients of the local interpolant. The right end can be handled similarly.

To be more specific, we work out some of the details in the cubic case (the quadratic case corresponds to the construction in Example 3). We determine the first two coefficient functionals by considering the interval $I = [t_4, t_6] = [x_1, x_5]$. The nonzero B-splines on this interval are $(B_{i,3})_{i=1}^{5}$ so we can enforce five local interpolation conditions, namely interpolation at the data points $(x_i)_{i=1}^{5}$. To make sure that we reproduce the local spline space, it is sufficient to require

$$\delta_{i,k} = \lambda_k(B_{i,3}) = \sum_{j=1}^{5} w_{k,j} B_{i,3}(x_j), \qquad \text{for } i = 1, \ldots, 5 \text{ and } k = 1, 2.$$

Since $B_{1,3}(x_1) = 1$ and all other B-splines are 0 at x_1, we see immediately that $\lambda_1 f = f(x_1)$. The other system of equations can be solved numerically to determine the $w_{k,j}$. The right end of the interval $[a, b]$ is treated similarly.

For $k = 3, \ldots, n-2$, we use the five data points $\{x_{k,j}\}_{j=1}^{5}$ belonging to the subinterval $[t_{k+1}, t_{k+3}]$ to calculate $\lambda_k f$, where $x_{k,j} = x_{2k+j-6}$ for $j = 1, 2, \ldots, 5$. Proceeding as above, we end up with a simple 5×5 system of equations for each set of unknowns $(w_{k,j})_{j=1}^{5}$.

It is possible to find explicit expressions for the weights by using a computer algebra system, and this can be useful for an analysis of the resulting approximation method, see the next section. When implementing the approximation method however, it is usually more efficient to solve the equations numerically, as the explicit formulas for the weights are rather complicated and expensive to evaluate. However, in certain special cases it is worth precomputing the weights. One such case is when the data points are uniformly spaced when we find

$$\begin{aligned}
\lambda_1(f) &= f(x_1), \\
\lambda_2(f) &= \frac{1}{18}\big(-5f(x_1) + 40f(x_2) - 24f(x_3) + 8f(x_4) - f(x_5)\big), \\
\lambda_3(f) &= \frac{1}{6}\big(f(x_1) - 8f(x_2) + 20f(x_3) - 8f(x_4) + f(x_5)\big).
\end{aligned}$$

The weights for $k = 4, \ldots, n-2$ are the same as those for $k = 3$, while the weights for $k = n-1$ agree with those for $k = 2$, but in reverse order.

Note that if x lies in an interval $[t_\mu, t_{\mu+1})$ for some integer μ in the range

$4 \le \mu \le n$, then $P_3 f$ reduces to

$$(P_3 f)(x) = \sum_{j=\mu-3}^{\mu} (\lambda_j f) B_{j,3}(x),$$

and therefore depends on (at most) 11 data values in the vicinity of $[t_\mu, t_{\mu+1})$.

§5. Example of Error Analysis

We give the result of an error analysis for the cubic quasi-interpolant P_3 that we considered in Section 4; the other operators can be analyzed similarly. To state the results, we use the following mesh-ratios:

$$\theta_j = \frac{x_{j+1} - x_j}{x_j - x_{j-1}}, \quad j = 2, \ldots, m-1.$$

Proposition 6. *For each positive integer j with $2j+1 \le m$, we have*

$$||P_3 f||_{\infty,[x_{2j-1},x_{2j+1}]} \le K(\rho) ||f||_{\infty,[x_{2j-5},x_{2j+5}]\cap[x_1,x_m]},$$

where $K(\rho)$ is a polynomial in $\rho = \max_k\{\theta_k, \theta_k^{-1}\}$.

Proof: We deduce the estimate on the first interval $[x_1, x_3]$; the treatment of the other intervals is similar. For $x \in [x_1, x_3]$ we have

$$P_3 f(x) = \sum_{j=1}^{4} \lambda_j f B_{j,3}(x),$$

where $\lambda_1 f = f(x_1)$, $\lambda_2 f = \sum_{j=1}^{5} w_{2,j} f(x_j)$, $\lambda_k f = \sum_{j=1}^{5} w_{k,j} f(x_{2k-6+j})$ for $3 \le k \le m-2$, and where the $w_{k,j}$ can be computed from $\lambda_k f = c_k$ given by (2). Note that only the 7 first x-values are used to define $P_3 f(x)$ for $x \in [x_1, x_3]$. Therefore, since the B-splines form a nonnegative partition of unity, we obtain

$$|(P_3 f)(x)| \le \max_{1\le k\le 4} \sum_{j=1}^{5} |w_{k,j}| \, ||f||_{\infty,[x_1,x_7]}, \quad x \in [x_1, x_3].$$

Since the $w_{k,j}$ for fixed k are computed by forming minors in the numerator in (2), it is clear that they must alternate in sign. This follows since all minors are nonnegative by total positivity properties of B-splines, see *e.g.* page 201 of [rndeBbook]. Moreover, since P_3 reproduces constants, we have $\sum_{j=1}^{5} w_{k,j} = 1$ for all k. Combining this identity with the fact that the $w_{k,j}$ oscillate in sign, we find that

$$\sum_{j=1}^{5} |w_{k,j}| \le 2C_k + 1,$$

where

$$C_k = |w_{k,2}| + |w_{k,4}|.$$

It then follows that

$$|(P_3 f)(x)| \le (2 \max_{1\le k\le 4} C_k + 1)||f||_{\infty,[x_1,x_7]}, \quad x \in [x_1, x_3].$$

Now $C_1 = 0$ and using *Mathematica* we find

$$C_2 = \frac{(1+\theta_2)^2 \big((1+\theta_4)^2 + \theta_3\theta_4(1+\theta_2)\nu_2\big)}{3\theta_2^2\theta_3\theta_4\nu_2},$$

while for $3 \le k+1 \le m-2$

$$C_{k+1} = \frac{(1+\theta_{2k-2})^2(1+\theta_{2k-1})(1+\theta_{2k})^2\big(1+\theta_{2k-1}\theta_{2k}(1+\theta_{2k-2}\theta_{2k-1})\big)}{3\theta_{2k-2}\theta_{2k-1}\theta_{2k}(\theta_{2k-1}(1+\theta_{2k})+\nu_k)},$$

where

$$\nu_k = 2 + \theta_{2k} + \theta_{2k-2}(2 + \theta_{2k} + 2\theta_{2k-1}(1+\theta_{2k})).$$

It follows that these expressions can be bounded by a polynomial in ρ. □

If the data are evenly spaced, or more generally, the even indexed datapoints are located midway between their neighbours, the estimate for the constant $K(\rho)$ simplifies.

Corollary 7. *Suppose that* $x_{2k} = (x_{2k-1} + x_{2k+1})/2$ *for* $k \le (m-1)/2$. *For any positive integer* j *with* $2j+1 \le m$ *we have*

$$||P_3 f||_{\infty,[x_{2j-1},x_{2j+1}]} \le K(\rho)||f||_{\infty,[x_{2j-5},x_{2j+5}]\cap[x_1,x_m]},$$

where

$$K(\rho) = (16\rho + 41)/9.$$

Proof: In this case $\theta_{2k} = 1$ for all k, and the expressions for C_2 and C_{k+1} in the proof of Proposition 6 simplify to

$$C_2 = \frac{8}{9}\left(2 + \frac{1}{\theta_3}\right),$$

$$C_{k+1} = \frac{8}{9}\left(\frac{1}{\theta_{2k-1}} + 1 + \theta_{2k-1}\right), \qquad \text{when } 3 \le k+1 \le m-2.$$

Since $\theta_{2k-1}^{-1} + \theta_{2k-1} \le 1 + \rho$ we find $K(\rho) \le 2C + 1$, where $C = \max_k C_k \le \frac{8}{9}(2+\rho)$ for all k. □

§6. Remarks

Remark 6.1. From Proposition 6 and (10) we can deduce that the error $P_3 f - f$ will be of the same order of magnitude as the error in best approximation by cubic splines. In fact if the data is uniform, then $\rho = 1$ and from Corollary 7 we find $K(\rho) = 19/3$. Of course $K(\rho)$ is larger for nonuniform data. In any case, if f has a bounded fourth derivative we see that $P_3 f$ is a fourth order accurate approximation to f.

Remark 6.2. The approximation procedure in this paper can be extended to trigonometric splines [rnLSS] and other more general classes of functions.

> *Remark 6.3.* We have given the three formulae (2), (4), and (8) for deriving quasi-interpolants reproducing the whole spline space. In addition formulas based on blossoming could be used. We refer to [rnLSS].

Acknowledgments. This work was done during the first authors's visit to Oslo under support of a Korea Science and Engineering Foundation (KOSEF) postdoctoral fellowships program.

References

1. de Boor, C., *A Practical Guide to Splines*, Springer Verlag, New York, 1978.
2. de Boor, C. and G. J. Fix, Spline approximation by quasiinterpolants, J. Approx. Theory **8** (1973), 19–45.
3. Lyche, T. and L. L. Schumaker, Local spline approximation methods, J. Approx. Theory **15** (1975), 294–325.
4. Lyche, T., L. L. Schumaker, and S. Stanley, Quasi-interpolants based on trigonometric splines, J. Approx. Theory **95** (1998), 280–309.
5. Schumaker, L. L. and C. Traas, Fitting scattered data on spherelike surfaces using tensor products of trigonometric and polynomial splines, Numer. Math. **60** (1991), 133–144.

Department of Applied Mathematics
Dongseo University
Pusan, 617-716 Korea
`lbg@dongseo.ac.kr`

Universitetet i Oslo
Istititt for informatikk
P.O. Box 1080, Blindern
0316 Oslo, Norway
`tom@ifi.uio.no`
`knutm@ifi.uio.no`