STUDIA UNIV. "BABEŞ–BOLYAI", MATHEMATICA, Volume **LIII**, Number 4, December 2008

THE WEIGHTED SPLINE QUASI-INTERPOLANT OPERATORS

MAGNOLIA TILCA

Abstract. A new quasi-interpolant operator starting from the operator described by Sablonnière [1], [7] is presented here. The operator is a linear combination of some linear functionals and normalized B-spline functions. If Sablonièr uses the arithmetic mean of the consecutive given points, the linear functionals presented here use the mesh points chosen as the weighted arithmetic mean of given points from the interval $[a, b]$. The article describes the way of computing the quadratic and cubic weighted spline quasi-interpolant operators and underlines the good numerical approximation of these new operators using implemented Matlab functions. The fact that the cubic weighted spline quasi-intepolant operators are a completion of the cubic spline quasi-interpolant operators offering a better approximation, but only among some intervals, is proven in the last section of the paper.

1. Introduction

The general construction of quasi-interpolants, which were first developed by Carl de Boor and G. J. Fix [2] and generalized later by Lyche and Schumaker [4], starts from the following problem. Given a function f, the basic problem of spline approximation is to determine B-spline coefficients $(c_i)_{i=1}^n$ such that $Pf = \sum_{i=1}^n c_i N_{i,k}$ is a reasonable approximation to f. Let assume that f is defined on an interval $I = [a, b]$, and select the space of splines of order $k + 1$, $S_{k+1}(\Delta, I)$, $\Delta : a = x_1 < x_2 < ... < x_n = b$ defined on I (i.e., so that $\overline{\Delta} : x_{-k+1} = x_{-k+2} = ... = x_{-1} = x_0 = a, b = x_{n+1} = x_{n+2} = ... = x_{n+k})$. To emphasize the dependence on f,

Received by the editors: 20.01.2008.
2000 *Mathematics Subject Classification.* 41A15, 65D07, 47D30, 68W25.
Key words and phrases. B-splines, quasi-interpolant operators.

the coefficient c_i is written $c_i = \mu_i f$, with μ_i some linear functionals. Thus, a quasi-interpolant spline is an approximation operator obtained as a linear combination of functions with finite support (B-splines $N_{i,k}$) $Qf = \sum_{i=1}^{n+1} \mu_i(f)N_i$. There are known some different types of these linear functionals μ_i such as: differential type ($\mu_i(f)$ is a linear combination of values of derivatives of f) or discrete type (combination of discrete values of f). This paper treats the case of the coefficients of discrete type where the combination is formed by the weighted mean values of the given points from Δ weighted by the values of the function f.

2. The most important features

This section concerns upon the construction of the weighted quasi-interpolant operators. Two methods of obtaining the new operators are presented here: the first one involves the non-recurrent expressions of the normalized quadratic and cubic B-splines and the second one presents the exact formulations of the coefficients of the weighted spline quasi-interpolant.

2.1. **The construction of the weighted spline quasi-interpolant operators using the normalized B-spline expressions.** Let $n, k > 0$ with $n \geq k + 3$ be known integers and let $f \in C^k(I)$ be a function with the known values $f(x_i), x_i \in \Delta$ such as $f(x_i) + f(x_{i-1}) \neq 0$. We choose $n + 1$ weighted values points

$$t_s := \frac{x_{s-1}f(x_{s-1}) + x_s f(x_s)}{f(x_{s-1}) + f(x_s)}, s = 1, ..., n + 1. \tag{2.1}$$

It is obvious that $t_1 = a$ and $t_{n+1} = b$.

Definition 1. *Let $x \in [x_l, x_{l+1}] \subset [a, b]$, for $l = 1, ..., n - 1$, $a, b \in R$. We define the spline quasi-interpolant of degree k and order $k + 1$*

$$Q_k f(x) = \sum_{i=1}^{k+1} \mu_i^{\{l\}}(f) \cdot N_{-k-1+i+l}(x), \tag{2.2}$$

$N_i := N_{i,k+1}$, with the functionals $\mu_i^{\{l\}}$,

$$\mu_i^{\{l\}}(f) = \sum_{j=1}^{k+1} a_{i,j}^{\{l\}} \cdot f(t_{i+j-1}), i = 1, ..., k + 1 \tag{2.3}$$

where the mid points t_s are defined in (2.1) and $a_{i,j}^{\{l\}}$ are coefficients which depend on l, $\forall l = 1, ..., , n-1$.

We denote the quasi-interpolant operator with weighted values by QI_w.

Problem 1. *Construct an algorithm for weighted spline quasi-interpolant operator and implemet a corespondent routine using the Matlab application.*

`Step I.` Find the coefficients $c_{i,l-r},\ i, r = 0, ..., k$ of the normalized B-spline functions $N_{l-r,k+1}, r = 0, ..., k$ from the expression $N_{l-r,k+1}(x) = c_{k,l-r}x^k + c_{k-1,l-r}$ x^{k-1} $+... + c_{0,l-r}$. These coefficients are obtained from the well known recurrence formula (see, for example, [6])

$$N_{j,1} = \begin{cases} 1, & x \in [x_j, x_{j+1}); \\ 0, & else \end{cases}$$

$$N_{j,k+1} = \frac{(x - x_j)N_{j,k}(x)}{x_{j+k} - x_j} + \frac{(x_{j+k+1} - x)N_{j+1,k}(x)}{x_{j+1+k} - x_{j+1}}, j = -k + 1, ..., n - 1. \quad (2.4)$$

It is known from [8] that for any $x \in [x_l, x_{l+1}), l \in \{1, ..., n - 1\}$, there are only $k + 1$ nonzero B-splines $N_{j,k+1},\ j = l - k, ..., l$. Thus, an explicit non-recurrent expression for the spline coefficients can be deduced from the relations (2.4) in the cases of quadratic ($k = 2$) [11] and cubic ($k = 3$) [10] B-splines.

$$N_{l,k+1} = \frac{\sum_{i=0}^{k}(-1)^i \binom{k}{i} x_l^i}{\prod_{j=2}^{k}(x_{l+j} - x_l)} M_{l,1} \cdot x^{k-i}, k \geq 2,$$

$$N_{l-1,k+1} = \sum_{p=1}^{k} \frac{\sum_{s=0}^{k}(-1)^{s+1}(x_{l+p}P_{s,p} + P_{s+1,p})M_{l,1} \cdot x^{k-s}}{\prod_{j=1,p+j\leq k}^{k-1}(x_{l+k-j} - x_{l-1}) \prod_{j=1,p+j>k}^{k-1}(x_{l+k-j+1} - x_l)}, k \geq 2,$$

$$N_{l-2,k+1} = \sum_{p=1}^{k} \frac{\sum_{s=0}^{k}(-1)^{s}(x_{l+p-k}R_{s,p} + R_{s+1,p})M_{l,1} \cdot x^{k-s}}{\prod_{j=1,p+j\leq k}^{k-1}(x_{l+1} - x_{l-k+j+p}) \prod_{j=1,p+j>k}^{k-1}(x_{l+2} - x_{l+1-j})}, k = 3,$$

and finally

$$N_{l-k,k+1} = \frac{\sum_{i=0}^{k}(-1)^{i+k} \binom{k}{i} x_{l+1}^i}{\prod_{j=2}^{k}(x_{l+1} - x_{l-j+1})} M_{l,1} \cdot x^{k-i}, k \geq 2,$$

where $P_{s,p} = \sum_{j=0}^{s} \binom{p-1}{s-j-1} \binom{k-p}{j} x_l^{s-j-1} x_{l-1}^j$ and $R_{s,p} = \sum_{j=0}^{s} \binom{p-1}{s-j-1} \binom{k-p}{j} \cdot$ $x_{l+2}^{s-j-1} x_{l+1}^j$. We take $P_{0,p} = P_{k+1,p} := 0, \forall p = 1, ..., k$ and $\binom{k}{i} := 0$ if $i < 0$ or $i > k$ and we observe that $P_{1,p} = R_{1,p} = 1$, $\forall p = 1, ..., k$.

`Step II.` Find the coefficients $a_{i,j}^{\{l\}}$ of the linear functionals (2.3) as the solution of the system obtained by applying the conditions of exactness of the quasi-interpolant operator in the space of polynomial of degree at most k. So, the conditions of the exactness of Q_k operator in the set of polynomials of degree at most k, $Q_k p = p, p \in P_k$, leads to the identities $Q_k(e_i) = e_i$ where $e_i(x) = x^i, i = 0, ..., k$. Rearranging after the powers of x and equalizing both sides we obtain a system with $(k+1)(k+1)$ equations and $(k+1)^2$ unknowns, $A \cdot C = B$, where A is the matrix of the unknown coefficients $a_{i,j}^{\{l\}}, i = 1, ..., k+1,\ j = 1, ..., k+1$, C is the matrix of the coefficients of the normalized B-spline functions and the values ξ_j, and B is the line matrix of the unity vector $u_i, i = 1, ..., k+1$ as presented below. The matrix C is of the form

$$C = \left[\begin{array}{l|l|l|l} c_{l-k} & c_{l-k}X_1 & ... & c_{l-k}X_1^k \\ & & & \\ c_{l-k+1} & c_{l-k+1}X_2 & ... & c_{l-k+1}X_2^k \\ ... & & & \\ c_l & c_l X_{k+1} & ... & c_l X_{k+1}^k \end{array} \right], \tag{2.5}$$

where c_{l-k} is the square bloc of the coefficients of the B-spline $N_{l-k,k+1}$

$$c_{l-k} = \begin{array}{llll} c_{k,l-k} & c_{k-1,l-k} & \cdots & c_{0,l-k} \\ ... & & & \\ c_{k,l-k} & c_{k-1,l-k} & \cdots & c_{0,l-k} \end{array}.$$

The X_i, $i = 1, ..., k+1$ vectors are defined as $X_i^j = [t_i^j, t_{i+1}^j, ..., t_{i+k}^j]^t$, $i = 1, ..., k+1$ for $j = 0, ..., k$. Maintaining the above notations, vector B is defined as $B = [u_{k+1} \,|\, u_k \,|\, ... \,|u_1]$ where u_i is a vector with 1 on the position of i, $i = 1, ..., k+1$, and the rest of k elements are zero. We can observe that for every value of $x \in [x_l, x_{l+1}], l = 1, ..., n-1$, it is necessary to solve $n-1$ systems with $(k+1)^2$ equations. In order to obtain the compatibility of the systems and the uniqueness of the solution, we impose the knots condition

$$n \geq (k+1) + 2.$$

`Step III.` Compute the values of the linear functionals $\mu_i^{\{l\}}$, $\forall x \in [x_l, x_{l+1})$.

`Step IV.` Compute the values of the weighted spline quasi-interpolant operator $Q_k(f)$, $\forall x \in [x_l, x_{l+1})$.

In order to solve this system, we have implemented an algorithm and by using the Matlab application we will obtain these practical results.

Problem 2. *Implement a routine which calculates*

1. *the joint values* t_j, $j = 1, ..., n+1$;

2. *the coefficients of the normalized B-spline functions* N_j, $j = -k+l, ..., l$, $l = 1, ..., n-1$;

3. *the values of the functionals* $\mu_i^{\{l\}}(f)$, $i = 1, ..., k+1$ *for some given functions* f

4. $Q_k(f)(x)$, $k \in \{2, 3\}$, *the value of the quasi-interpolant operator* Q_k *for a given number* $x \in (a, b)$

5. *the values of the quasi-interpolant operator* Q_k *for all the equidistant numbers* $x \in [a, b]$ *with step* 0.1.

The algorithm calculates the coefficients and the values of the quadratic and cubic weighted spline quasi-interpolant Q_k for any partition X of the interval $I = [a, b]$ and any $x \in (a, b)$.

`Input:`

a) X, the vector of the extended partition $\overline{\Delta}$;

b) $x \in (a, b)$;

c) f, a function which may be chosen from the set of functions

$\{ax^2 + bx + c, a/(b + cx), a \cdot (sin(bx))^c, a \cdot (cos(bx))^c, (ae^{b \cdot x})^c, (ax^2 + 1)/(bx + c)\}$.

`Output:`

i) CC, the vector of the coefficients of the quasi-interpolant Q_k;

ii) $Q_k(f)(x)$, the value of the quasi-interpolant operator Q_k for a given number $x \in (a, b)$;

iii) q, the vector values of the quasi-interpolant operator Q_k for all the equidistant numbers $x \in [a, b]$ with the step 0.1.

`Step 1:` The computation of the matrix denoted with n of order $k+1$ of the B-spline $N_j, j = l-k, ..., l$ coefficients $c_{k-s,l-r}$, $s = 0, ..., k, r = 0, ..., k, l = 1, ..., n-1, k \in \{2, 3\}$

using the non-recurrent expressions of the coefficients from the equalities mentioned above.

`Step 2:` The elements of the vector T of the $t_j, j = 1, ..., n+1$ from (2.1) ;

`Step 3:` The construction of the matrix C of the form (2.5);

`Step 4:` The construction of the vector B of the form $B = [u_{k+1} \mid u_k \mid ... \mid u_1]$;

`Step 5:` The computation of the solution of the matrix equation $A \cdot C = B$, A as $A = B \cdot C^{-1}$;

`Step 6:` The computation of the values of the functionals $\mu_i^{\{l\}}, i = l, ..., k+1$. (It is not necessary to compute all the values $\mu_i^{\{l\}}$, $i = 1, ...n-1, n \geq k$, because only $N_{l-k}, ..., N_l$ are nonzero).

In what follows, we will exemplify our results on the non polynomial case.

Example 1. *(Numerical results)*
Let $I = [0, 1.25]$, $\Delta : 0 < 0.25 < 0.5 < 0.75 < 1 < 1.25$ *and* $0.1 := x \in [x_1, x_2] := [0, 0.25]$. *For a given function, say* $f(x) = (x^2+1)/(x+1)$, *find the value of the cubic quasi-interpolant operator* Q.

Solution: Applying the Matlab function

```
[Q]=coef_w_cubic(X,x,a,b,c,functia),
```

the vector of the extended partition of the cubic (order=4, degree=3) B-spline function is $X = [0\ 0\ 0\ 0\ 0.25\ 0.5\ 0.75\ 1\ 1.25\ 1.25\ 1.25\ 1.25]$ and the coefficients of the function are $a = 1, b = 1, c = 1$. The Matlab function

```
coef_w_cubic(X,0.1,1,1,1,'(a*x.^2+1)/(b*x+c)')
```

generates the following results:

1. the joint values t_j, $j = 1, ..., 7$ are

$T = 0 \quad 0.1149 \quad 0.3738 \quad 0.6293 \quad 0.8821 \quad 1.1331 \quad 1.2500$;

2. the coefficients of the normalized B-spline functions N_j, $j = -2, ..., 1$ are contained in the following matrix

$n =$

$$\begin{array}{c|cccc} N_1 & 10.6667 & 0 & 0 & 0 \\ N_0 & -58.6667 & 24.0000 & 0 & 0 \\ N_{-1} & 112.0000 & -72.0000 & 12.0000 & 0 \\ N_{-2} & -64.0000 & 48.0000 & -12.0000 & 1.0000 \end{array},$$

from where we can construct the expressions of the B-spline functions:

$N_{1,4} = 10.(6)x^3$

$N_{0,4} = -58.(6)x^3 + 24x^2$

$N_{-1,4} = 112x^3 - 72x^2 + 12x$

$N_{-2,4} = -64x^3 + 48x^2 - 12x + 1;$

3. the values of the functionals $\mu_i^{\{1\}}(f)$, $i = 1, ..., 4$ are the elements of the vector

$CC = 1.0000 \quad 0.9193 \quad 0.8302 \quad 0.8216;$

4. the value of the quasi-interpolant operator Q_3 is

$Q = 0.9195$

Remark 1. *The value of the function f for $x = 0.1$ is 0.9182 which means that the weighted spline quasi-interpolant operator offers a good approximation.*

Furthermore, for computing the values of the quasi-interpolant operator Q_3 for all the equidistant numbers $x \in [a, b]$ with the step 0.1, we implemented another Matlab function

$$[q] = table\_val\_Q\_cubic(T, X, a, b, c, functia).$$

Thus, for $T = [0 \quad 1.25]$,

$q = 1.0000$ 0.9195 0.8679 0.8400 0.8400 0.8313 0.8484 0.8753 0.9128 0.9539 1.0000 1.0525.

Remark 2. *We have also implemented a function*

$$[tabel] = final(X, T, a, b, c, functia)$$

to compare the values of the given function with the values of the weighted cubic spline quasi interpolant operator and the values of the cubic spline quasi-interpolant operator described by Sablonière. Thus, applying

$$final(X, [0 \ \ 1.25], 1, 1, 1, '(a * x.^2 + 1)/(b * x + c)')$$

we get

tabel =

x	0	0.1	0.2	0.3	0.4	0.5	0.6
f	1.000	0.9182	0.8667	0.8385	0.8286	0.8333	0.8500
$Q_3(f)$	1.0000	0.9203	0.8675	0.8378	0.8275	0.8325	0.8494
$Q_{w3}(f)$	1.0000	0.9195	0.8679	0.8400	0.8298	0.8313	0.8484
$Q-f$	0.0000	0.0022	0.0008	-0.0006	-0.0011	-0.0008	-0.0006
Q_w-f	0.0000	0.0013	0.0012	0.0016	0.0012	-0.0020	-0.0016

x	0.7	0.8	0.9	1	1.1
f	0.8765	0.9111	0.9526	1.0000	1.0524
$Q_3(f)$	0.8760	0.9108	0.9524	1.0000	1.0526
$Q_{w3}(f)$	0.8753	0.9128	0.9539	1.0000	1.0525
$Q-f$	-0.0004	-0.0003	-0.0002	-0.0000	0.0002
Q_w-f	-0.0012	0.0017	0.0012	0.0000	0.0001

Analyzing the errors expressed in the last two rows of the table, we can notice that the operators described by Sablonière are better approximations than the operators presented in this paper, with the exception of the edged values. Making further investigations, the weighted cubic quasi-interpolant operators are more convenient for approximation of this function in values contained on the interval $[0; 0.17] \cup [1; 1.24]$. *This better approximation can be visualized in the graphical error representation, Fig.1, where the errors generated by quasi-interpolated operators with mean values are larger than the errors generated by quasi-interpolated operators with weighted values.*

2.2. **The construction of the weighted cubic spline quasi-interpolant operators which does not require the normalized B-spline expressions.** As we could see, the construction of the weighted spline quasi-interpolant operators requires the solution of linear systems. To avoid this volume of computation, following the idea presented in [7], we can generate the exact expressions of the coefficients $a_{i,j}^{\{l\}}$, $\forall i, j = 1, ...k+1$ and $\forall l = 1, ..., n-1$.

Let $\overline{\Delta}$ be the extended partition of the interval $[a, b]$. We recall the definition of Greville's points (mentioned in [5] and [9])

$$\xi_j = \frac{x_{j+1} + x_{j+2} + ... + x_{j+k}}{k}, j = -k+1, ..., n-1,$$

$$\xi_j^{(2)} = \frac{x_{j+1}x_{j+2} + x_{j+1}x_{j+3} + ... + x_{j+k-1}x_{j+k}}{\binom{k}{2}}, j = -k+1, ..., n-1,$$

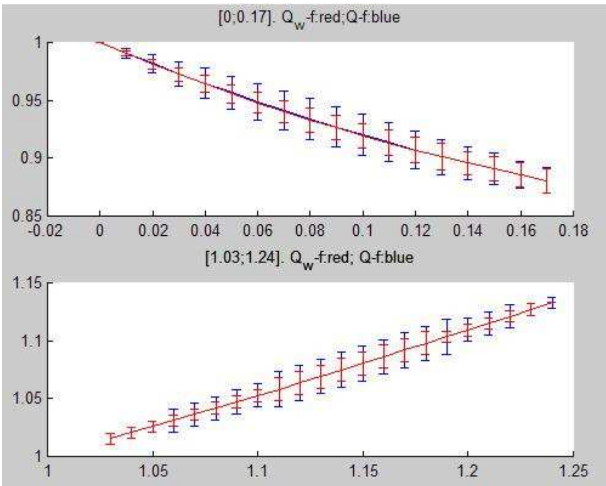

FIGURE 1. The graphical error representation

$$\xi_j^{(3)} = \frac{x_{j+1}x_{j+2}x_{j+3} + x_{j+1}x_{j+2}x_{j+4} + \dots + x_{j+k-2}x_{j+k-1}x_{j+k}}{\binom{k}{3}}, j = -k+1, \dots, n-1.$$

Theorem 1. *For $k = 3$, the exact expressions of the coefficients $a_{i,j}^{\{l\}}$ of the linear functionals $\mu_i^{\{l\}}$ from (2.3) are given by the formula*

$$a_{i,j}^{\{l\}} = (-1)^{j+1} \frac{P_{i,j} - \xi_{-4+i+l} \cdot (SP)_{i,j} + \xi_{-4+i+l}^{(2)} \cdot S_{i,j} - \xi_{-4+i+l}^{(3)}}{\prod_{i \le s < p \le i+k}^{(s=i+j-1)\vee(p=i+j-1)} (t_p - t_s)} \tag{2.6}$$

where P, SP and S are respectively the product of all the elements t where t_{i+j-1} is omitted, SP denotes the sum of all combinations of the products of two elements t from P and, finally, S is the sum of all elements from P:

$$P_{i,j} := \frac{\prod_{s=i}^{i+k} t_s}{t_{i+j-1}},$$

$$(SP)_{i,j} := \sum_{i \le s < p \le i+k} t_s t_p - \sum_{i \le s < p \le i+k}^{(s=i+j-1)\vee(p=i+j-1)} t_s t_p,$$

$$S_{i,j} := \sum_{s=i}^{i+k} t_s - t_{i+j-1}.$$

Proof. We begin by imposing the conditions of exactness for the quasi-interpolant operators Q_k in the set of polynomials of degree at most k, $Q_k p = p, p \in P_k$ which lead to the identities $Q_k(e_i) = e_i$ where $e_i(x) = x^i, i = 0, ..., k$. The e_i functions can be rewrites using Marsden's equalities [5]

$$x^j = \sum_{i=1}^{k+1} \xi^{(j)}_{-4+i+l} N_{-4+i+l}(x), j = 0, ..., k,$$

with $\xi_i^{(0)} := 1$ and $\xi_i^{(1)} := \xi_i$. Equalizing the coefficients $a_{i,j}^{\{l\}}$ from the equations $(Q_k e_i)(x) = e_i(x), i = 0, ..., k$ we obtain a system with $(k+1)^2$ equations and $(k+1)^2$ unknows for every $l \in \{1, ..., n-1\}$. The system being a separable variables one, we can rearrange the equations obtaining $(k+1)$ systems with $(k+1)$ equations by the form $\sum_{j=1}^{k+1} a_{i,j}^{\{l\}} t_j^s = \xi^{(s)}_{-4+i+l}$, for fixed $i \in \{1, ..., k+1\}$ and $s = 0, ..., k$ with $\xi_i^{(0)} := 1$ and $\xi_i^{(1)} := \xi_i$. The determinant of these systems is Vandermonde determinant, thus the computation is quite simple and each solution $a_{i,j}$ of the systems can be generalized by the (2.6). $\square$

3. **The evaluation of the error**

In this section a comparison of the norm of the cubic quasi-interpolant operator with weighted values (QI_w) and the norm of the cubic quasi-interpolant operator with mean values (QI) is presented. In general, it is difficult to minimize the true norm of the operators. In order to avoid this direct minimization we use the idea of Sablonniére [1] of the minimization problem:

Let $Qf = \sum_i \mu_i(f) N_i$ be the general form of the spline quasi-interpolant of f, with $\mu_i(f) = \sum_i a_i f(x_i)$. Find $a_i \in R^n$ solution of the problem

$$\|a_i^*\|_1 = min\left\{\|a_i\|_1, a_i \in R^n, V_i a_i = b_i\right\}$$

where $\|Q\|_\infty \leq \sum_i |\mu_i(f)| N_i \leq max_i |\mu_i(f)| \leq max_i \|a_i\|_1$. The notation V_i denotes the Vandermonde matrix.

Thus, the minimization of the norm $\|Q\|_\infty$ reduces to the operate with the coefficients a_i.

It is known from [1] that the norm of QI is not uniformly bounded independent of the partition.

Theorem 2. [1] *For the cubic spline quasi-interpolant operators* Q_3 *let the linear functionals be* $\mu_i(f) = a_i f(x_{i-1}) + b_i f(x_i) + c_i f(x_{i+1})$. *If there exists* $r > 0$ *such that the partition satisfies* $\frac{1}{r} \leq \frac{h_{i+1}}{h_i} \leq r$, $i \in Z$, *where* $h_i = x_i - x_{i-1}$, *than we obtain the following upper bounds* $|a_i|, |c_i| \leq \frac{1}{3}\frac{r^2}{1+r}$, $|b_i| \leq \frac{1}{3}(1+r)^2$, *from which* $\|Q_3\|_\infty \leq \frac{1}{3}\left((1+r)^2 + \frac{2r^2}{1+r}\right)$.

Thus, in the case of uniform partition, $r = 1$, the upper bound of the norm is ≈ 1.66.

The next result states that the cubic weighted spline quasi-interpolant operators QI_w for the uniform partition case also have the upper bound less than 1.66.

Theorem 3. *Let* $I = [a, b]$ *be an interval with the uniform partition* Δ *and* Q_3 *the cubic weighted spline quasi-interpolant operator given by*

$$Q_3 f(x) = \sum_{i=1}^{n+3} \mu_i(f) N_i(x)$$

with linear functionals

$$\mu_i(f) := a_i f(t_{i-1}) + b_i f(t_i) + c_i f(t_{i+1}),$$

$$\mu_1(f) = f(a), \mu_{n+3}(f) = f(b)$$

and the points t_i *defined as in (2.1). For* $f \in C[a, b]$ *smooth enough we have*

$$\|Q_3\|_\infty \leq 1.66.$$

Proof. We define the auxiliary points

$$t_i := (1 - m)x_i + m x_{i-1}, \tag{3.1}$$

$i = 1, ..., n+3, m \in [0, 1]$ which are a generalization of the points t_i taken as mean values of $x_i \in \Delta$ and taken as weighted values of $x_i \in \Delta$ and $f(x_i)$.

The idea of the proof is to express the coefficients a_i, b_i, c_i depending only on the parameter m.

To compute these coefficients, we follow the same idea presented in Subsection 2.2. Thus, after imposing the conditions of the exactness $Q_3(e_s) = e_s$, $s = 0, ..., 3$ and after using Marsden's equalities, rearranging after the powers of t_i, $i = 1, ..., n+3$ we get a 3×3 system for every i, from which the coefficients are

$$a_i = \frac{t_i t_{i+1} - \xi_i^{(1)}(t_i + t_{i+1}) + \xi_i^{(2)}}{(t_i - t_{i-1})(t_{i+1} - t_{i-1})}, \tag{3.2}$$

$$b_i = -\frac{t_{i+1} t_{i-1} - \xi_i^{(1)}(t_{i+1} + t_{i-1}) + \xi_i^{(2)}}{(t_i - t_{i-1})(t_{i+1} - t_i)},$$

$$c_i = \frac{t_{i-1} t_i - \xi_i^{(1)}(t_i + t_{i-1}) + \xi_i^{(2)}}{(t_{i+1} - t_i)(t_{i+1} - t_{i-1})},$$

$i = 1, ..., n+2$. It is obvious that $a_i + b_i + c_i = 1$, $\forall i = 1, ..., n+3$.

These expressions are easily computable when the relation (3.1) is used, $t_i = (1-m)x_i + mx_{i-1}$. Thus, we get

$$a_i = \frac{3m^2 + 9m + 5}{6}, b_i = -m^2 - 4m - \frac{8}{3}, c_i = \frac{3m^2 + 15m + 17}{6}$$

and again $a_i + b_i + c_i = 1$, $\forall i = 1, ..., n+3$. Now from the fact that $\|Q_3\|_\infty \leq |a_i| + |b_i| + |c_i|$ [1], $\forall i = 1, ..., n+3$ and using the Matlab application to evaluate this expression for every $m \in [0, 1]$, we have that $\|Q_3\|_\infty \leq 1.66$. $\square$

It is well known ([12], chapter 5) that for any subinterval $I_i = [x_{i-1}, x_i]$, $i = 1, ..., n$ and for any function f, $\|f - Q_k f\|_{\infty, I_i} \leq (1 + \|Q_k\|_\infty) d_{\infty, I_i}(f, \Pi_k)$ where the distance of f to polynomials is defined by $d_{\infty, I_i}(f, \Pi_d) = inf\{\|f - p\|_{\infty,, I_i}, p \in \Pi_k\}$, $\|f - p\|_{\infty,, I_i} = max_{x \in I_i} |f(x) - p(x)|$. Therefore, for $f \in C^4(I)$ the error estimated is

$$\|f - Q_3 f\|_{\infty, I_i} \leq 2.66 \cdot d_{\infty, I_i}(f, \Pi_3)$$

for $i = 1, ..., n$.

From these theoretical arguments and numerical computations, an approach between these two quasi-interpolant operators QI_w and QI can be observed. The QI_w operators complete the QI operators because they can provide better approximations on some subintervals of I.

References

[1] Berrera, D., Ibáñez, M.J., Sablonnière, P., Sbibih, D., *On two families of near-best spline quasi-interpolants on non-uniform partitions of the real line*, IRMAR, January 2006.

[2] de Boor, C., Fix, J.J., *Spline approximation by quasi-interploants*, J. Approx. Theory, **8**(1973), 19-45.

[3] Curry, H.B., Schoenberg, I.J., *On Polya frequency functions IV: The fundamental spline functions and their limits*, J. d'Analyse Math. **17**(1966), 71-107.

[4] Lyche, T., Schumaker, L.L., *Local spline approximation methods*, J. Approx. Theory, **25**(1979), 266-279.

[5] Marsden, J.M., Schoenberg, I.J., *An identity for spline functions with applications to variation diminishing spline approximation*, J. Approx. Theory, **3**(1970), 7-49.

[6] Micula, Gh., Micula, S., *Handbook of spline*, Kluwer Academic Publishers, Dordrecht-Boston-London, 1999.

[7] Sablonnière, P., *Univariate spline quasi-interpolants and applications to numerical analysis*, INSA and IRMAR, Rennes, 2005.

[8] Schumaker, L., *Spline functions, basic theory*, A Wiley-Interscience Publication John Wiley and Sons, New York, 1980.

[9] Tilca, M., *Upon an extensive set of knots for the normalized B-spline functions*, Proceedings of the International Conference on Numerical Analysis and Approximation Theory, Cluj-Napoca, 2006, 393-400.

[10] Tilca, M., *A cubic spline quasi-intepolant operator*, International Journal of Pure and Applied Mathematics, Vol. 42, No.1, (2008), 39-48.

[11] Tilca, M., *The spline quasi-interpolant operators*, International Conference "Educaţie şi Creativitate Pentru o Societate Bazată pe Cunoaştere", Bucureşti, 2007.

[12] DeVore, R.A., Lorentz, G.G., *Constructive approximation*, Springer-Verlag, Berlin, 1993.

Titu Maiorescu University,
Faculty of Information Sciences and Technology,
187, Văcăreşti Ave., sector 4, 040051, Bucharest
E-mail address: magnolia.tilca@utm.ro