Evaluation of tension splines*

IVO BEROŠ† AND MILJENKO MARUŠIĆ‡

Abstract. *Tension spline of order k is a function that, for a given partition $x_0 < x_1 < \ldots < x_n$, on each interval $[x_i, x_{i+1}]$ satisfies differential equation $(D^k - \rho_i^2 D^{k-2})u = 0$, where ρ_i's are prescribed nonnegative real numbers.*

Most articles deal with tension splines of order four, applied to the problem of convex (or monotone) interpolation or to the two-point boundary value problem for ordinary differential equations. Higher order tension splines are described in several papers, but no application is given. Possible reason for this is a lack of an appropriate algorithm for their evaluation. Here we present an explicit algorithm for evaluation of tension splines of arbitrary order. We especially consider stable and accurate computation of hyperbolic-like functions used in our algorithm.

Key words: *tension spline, B-spline, evaluation*

AMS subject classifications: 65D07, 65-04

1. Preliminaries

Tension spline of order k is a function that, for given partition $x_0 < x_1 < \ldots < x_n$, on each interval $[x_i, x_{i+1}]$ satisfies differential equation

$$(D^k - \rho_i^2 D^{k-2})u = 0,$$

where ρ_i's are prescribed nonnegative real numbers. In other words, tension splines of order k are functions whose restrictions on non-empty interval (x_i, x_{i+1}) lie in span$\{1, x, \ldots, x^{k-1}\}$, for $\rho_i = 0$, or in span$\{e^{\rho_i x}, e^{-\rho_i x}, 1, x, \ldots, x^{k-3}\}$, otherwise.

Almost all papers dealing with tension splines consider tension spline of order four. Applications of such splines are mostly shape preserving approximation [9, 10] and a numerical solution of singularly perturbed two-point boundary value problem for ODE [1, 5]. Higher order tension splines are described in several papers [2, 4], but no application is given. Their application may be foreseen in a problem of

*This paper was presented at the Special Section: Appl. Math. Comput. at the 7^{th} *International Conference on OR, Rovinj, 1998.*

†Department of Mathematics, University of Zagreb, Bijenička 30, HR-10 000 Zagreb, Croatia, e-mail: `ivo.beros@math.hr`

‡Department of Mathematics, University of Zagreb, Bijenička 30, HR-10 000 Zagreb, Croatia, e-mail: `miljenko.marusic@math.hr`

approximation by functions with positive higher derivative, as well as in singularly perturbed boundary value problems of higher degree. A possible reason for the absence of their application is a lack of an appropriate algorithm for their evaluation. Here we present an explicit algorithm for the evaluation of tension splines of arbitrary order.

2. B-splines

As for the case of polynomial splines, we use a B-spline representation for tension splines (cf. [2, 3, 4, 6, 7]).

Suppose m, n are positive integers and $m \geq 2$. Let $t_1 \leq t_2 \leq \ldots \leq t_{n+m}$ be a non-decreasing sequence of knots, and for each non-empty interval (t_i, t_{i+1}) let ρ_i be a given non-negative number. Tension B-splines of order k are defined recursively [2, 8]

$$B_{j,k} := \frac{\int_{t_j}^{x} B_{j,k-1}(y)dy}{\sigma_{j,k-1}} - \frac{\int_{t_{j+1}}^{x} B_{j+1,k-1}(y)dy}{\sigma_{j+1,k-1}}\,,\ j = m-k+1, \ldots n, \tag{1}$$

where

$$\sigma_{j,k-1} := \int_{t_j}^{t_{j+k-1}} B_{j,k-1}(y)dy\,.$$

The recursion starts with functions $B_{i,2}$ defined by

$$B_{i,2}(x) := \begin{cases} \sinh(\rho_i(x-t_i))/\sinh(\rho_i\Delta_i)\,; & t_i \leq x < t_{i+1}\,, \\ \sinh(\rho_{i+1}(t_{i+2}-x))/\sinh(\rho_{i+1}\Delta_{i+1})\,; & t_{i+1} \leq x < t_{i+2}\,, \\ 0\,; & \text{elsewhere,} \end{cases}$$

$$\Delta_i = t_{i+1} - t_i\,.$$

In [2] it is proved that B-splines $B_{i,k}$ have the following properties:

1. $\operatorname{supp} B_{i,k} = [t_i, t_{i+k}]$ (local support property)
2. $B_{i,k}(x) > 0$ for $t_i < x < t_{i+k}$ (positivity)
3. $\sum_i B_{i,k} \equiv 1$, $k \geq 3$ (partition of unity)
4. $B_{i,k}$ has $k-1-l$ continuous derivatives at a knot of multiplicity l.

Moreover, B-splines $B_{j,k}$, $j = m-k+1, \ldots n$, are linearly independent and they form a basis for the space $\mathcal{T}_k = \operatorname{span}\{B_{m-k+1,k}, \ldots, B_{n,k}\}$. $\mathcal{T}_k$ is a space of tension splines of order k, with smoothness at knots t_i defined by multiplicity of the knots.

3. B-spline representation

Now we consider a local representation of the B-splines on the subinterval (t_i, t_{i+1}). We suppose that $t_i \neq t_{i+1}$ and that ρ_i is given.

Restriction of each B-spline on subinterval (t_i, t_{i+1}) can be represented in the form

$$B_{j,k}(x) = \beta^{j,k}_{-2,i}\varphi_k(x-t_i) + \beta^{j,k}_{-1,i}\varphi_k(t_{i+1}-x) + \sum_{l=0}^{k-3}\beta^{j,k}_{l,i}\left(\frac{x-t_i}{\Delta_i}\right)^l, \qquad (2)$$

where $\varphi_k(x) := \varphi_k(x;\rho_i,\Delta_i)$ is defined by

$$\varphi_k(x;\rho,\Delta) = \begin{cases} \frac{x^{k-1}}{(k-1)!\Delta}, & \rho = 0\,, \\ \frac{F_k(\rho x)-P_k(\rho x)}{\rho^{k-2}\sinh(\rho\Delta)}, & \rho > 0, \end{cases} \qquad (3)$$

and

$$F_k(t) = \begin{cases} \sinh(t), & k = 2l, \\ \cosh(t), & k = 2l+1, \end{cases} \qquad \text{and} \quad P_k(t) = \begin{cases} \sum_{j=0}^{l-2}\frac{t^{2j+1}}{(2j+1)!}, & k = 2l, \\ \sum_{j=0}^{l-1}\frac{t^{2j}}{(2j)!}, & k = 2l+1. \end{cases} \qquad (4)$$

We may note that

$$\int_0^u \varphi_k(y)dy = \varphi_{k+1}(u)\,, \quad u < \Delta_i\,. \qquad (5)$$

Now, we are going to find a representation of $\int_{t_j}^x B_{j,k}(y)dy/\sigma_{j,k}$ on subinterval (t_i, t_{i+1}). Because of the local support property, $\int_{t_j}^x B_{j,k}(y)dy/\sigma_{j,k} = 0$ for $t_{i+1} < t_j$, and it is equal to 1 for $t_i > t_{j+k}$. Otherwise, for $x \in (t_i, t_{i+1})$,

$$\begin{aligned} \frac{\int_{t_j}^x B_{j,k}(y)dy}{\sigma_{j,k}} &= \left(\int_{t_i}^x \left(\beta^{j,k}_{-2,i}\varphi_k(y-t_i) + \beta^{j,k}_{-1,i}\varphi_k(t_{i+1}-y)\right)dy\right. \\ &+ \left.\int_{t_i}^x \sum_{l=0}^{k-3}\beta^{j,k}_{l,i}\left(\frac{y-t_i}{\Delta_i}\right)^l dy + \int_{t_j}^{t_i} B_{j,k}(y)dy\right) \Big/ \sigma_{j,k}\,. \end{aligned} \qquad (6)$$

Taking into account that

$$\begin{aligned} \int_{t_i}^x \varphi_k(y-t_i)dy &= \varphi_{k+1}(x-t_i), \\ \int_{t_i}^x \varphi_k(t_{i+1}-y)dy &= \varphi_{k+1}(\Delta_i) - \varphi_{k+1}(t_{i+1}-x)\,, \\ \int_{t_i}^x \left(\frac{y-t_i}{\Delta_i}\right)^l dy &= \frac{\Delta_i}{l+1}\left(\frac{x-t_i}{\Delta_i}\right)^{l+1} \end{aligned}$$

and defining

$$\sigma^i_{j,k} = \int_{t_j}^{t_i} B_{j,k}(y)dy\,,$$

from (6) we obtain

$$\frac{\int_{t_j}^x B_{j,k}(y)dy}{\sigma_{j,k}} = \frac{1}{\sigma_{j,k}}\Big(\beta^{j,k}_{-2,i}\varphi_{k+1}(x-t_i) - \beta^{j,k}_{-1,i}\varphi_{k+1}(t_{i+1}-x)$$

$$+ \quad \sigma^i_{j,k} + \beta^{j,k}_{-1,i}\varphi_{k+1}(\Delta_i) + \Delta_i \sum_{l=1}^{k-2} \frac{\beta^{j,k}_{l-1,i}}{l} \left(\frac{x-t_i}{\Delta_i} \right)^l \Bigg) . \qquad (7)$$

Using (1) and (7), we establish a relation among coefficients $\beta^{j,k+1}$ of B-spline $B_{j,k+1}$ and coefficients $\beta^{j,k}$ and $\beta^{j+1,k}$ of $B_{j,k}$ and $B_{j+1,k}$, respectively:

$$\begin{aligned} \beta^{j,k+1}_{-2,i} &= \frac{\beta^{j,k}_{-2,i}}{\sigma_{j,k}} - \frac{\beta^{j+1,k}_{-2,i}}{\sigma_{j+1,k}} \\ \beta^{j,k+1}_{-1,i} &= \frac{\beta^{j+1,k}_{-1,i}}{\sigma_{j+1,k}} - \frac{\beta^{j,k}_{-1,i}}{\sigma_{j,k}} \\ \beta^{j,k+1}_{0,i} &= \frac{\sigma^i_{j,k} + \varphi_{k+1}(\Delta_i)}{\sigma_{j,k}} - \frac{\sigma^i_{j+1,k} + \varphi_{k+1}(\Delta_i)}{\sigma_{j+1,k}} \\ \beta^{j,k+1}_{l,i} &= \Delta_i \frac{\beta^{j,k}_{l-1,i}}{l\sigma_{j,k}} - \Delta_i \frac{\beta^{j+1,k}_{l-1,i}}{l\sigma_{j+1,k}}, \; l = 1, \ldots, k-2 \, , \end{aligned} \qquad (8)$$

for $k \geq 3$ and $j = i-k+1, \ldots, i$. The recursion starts with

$$\beta^{i-1,2}_{-2,i} = 0, \; \beta^{i-1,2}_{-1,i} = 1, \; \beta^{i,2}_{-2,i} = 1 \text{ and } \beta^{i,2}_{-1,i} = 0 \, . \qquad (9)$$

Coefficients $\sigma^i_{j,k}$ and $\sigma_{j,k}$ are obtained from

$$\begin{aligned} \sigma^i_{j,k} &= \sum_{l=j}^{i-1} \int_{t_l}^{t_{l+1}} B_{j,k}(y)dy \\ &= \sum_{l=j}^{i-1} \left((\beta^{j,k}_{-2,l} + \beta^{j,k}_{-1,l})\varphi_{k+1}(\Delta_l) + \Delta_l \sum_{s=0}^{k-3} \frac{\beta^{j,k}_{s,l}}{s+1} \right) , \end{aligned}$$

noting that $\sigma_{j,k} = \sigma^{j+k}_{j,k}$.

4. Computing functions φ_k

Functions φ_k, defined by (3) depend on three parameters, and for each parameter we only know that it is nonnegative. It is more convenient to use functions

$$\tilde{\varphi}_k(p,t) = \begin{cases} \frac{t^{k-1}}{(k-1)!}, & p = 0 \, , \\ \frac{F_k(pt) - P_k(pt)}{p^{k-2}\sinh(p)}, & p > 0 \, . \end{cases} \qquad (10)$$

By substituting

$$p = \rho\Delta \text{ and } t = \frac{x}{\Delta}$$

we obtain the relation:

$$\varphi_k(x; \rho, \Delta) = \Delta^{k-2} \tilde{\varphi}_k(\rho\Delta, \frac{x}{\Delta}) \, . \qquad (11)$$

Function $\tilde{\varphi}_k$ depends on two parameters. Parameter t is from interval $[0, 1]$ and p is nonnegative.

Computing hyperbolic-like functions $\tilde{\varphi}_k$ by using explicit formula (10) is very sensitive. For p or pt small (10) is numerically unstable, while for p or pt large (e.g. over 700 for double precision arithmetic), machine overflow occurs. So we have to consider evaluation of $\tilde{\varphi}_k$ in the different situations. All calculation presented in the following text is done in FORTRAN 77 for HP 9000.

4.1. Case when p is small

When p is close to 0, there are two possible sources of error in evaluation of $\tilde{\varphi}_k$ by (10). The first is a loos of significant digits in subtraction $F_k(pt) - P_k(pt)$ and the second is a possible underflow in evaluation of $p^{k-2}\sinh p$. *Figure 1* shows relative errors in evaluation of $\tilde{\varphi}_6$ in this domain (small p). For example, we can see that (10) is inappropriate for $p \leq 0.5$.

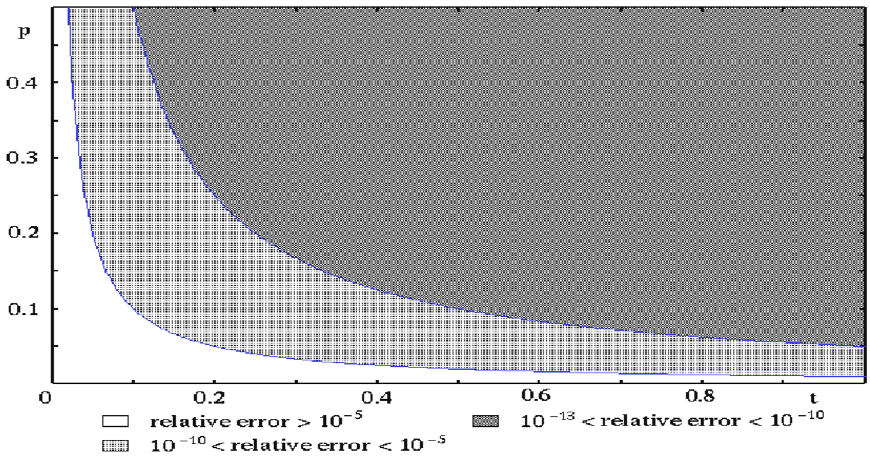

Figure 1. *Relative error for evaluating* $\tilde{\varphi}_6$ *by (10), on* $[0, 0.5] \times [0, 1]$*, in double precision arithmetic.*

To solve this problem, we expand the numerator and the denominator in Taylor series. For k even we obtain:

$$\begin{aligned}
\tilde{\varphi}_k(p,t) &= \frac{F_k(pt) - P_k(pt)}{p^{k-2}\sinh(p)} = \frac{\sum_{j=l-1}^{\infty} \frac{(pt)^{2j+1}}{(2j+1)!}}{p^{k-1}\frac{\sinh(p)}{p}} = \\
&= t^{k-1}\frac{\frac{1}{(k-1)!} + \sum_{j=l}^{\infty} \frac{(pt)^{2(j-l+2)}}{(2j+1)!}}{1 + \sum_{j=1}^{\infty} \frac{p^{2j}}{(2j+1)!}} \\
&= t^{k-1}\frac{\frac{1}{(k-1)!} + \sum_{j=1}^{\infty} \frac{(pt)^{2j}}{(k+2j-1)!}}{1 + \sum_{j=1}^{\infty} \frac{p^{2j}}{(2j+1)!}}\,.
\end{aligned}$$

The same formula holds for k odd, so $\tilde{\varphi}_k$ is of the form

$$\tilde{\varphi}_k(p,t) = t^{k-1} \frac{\frac{1}{(k-1)!} + \sum_{j=1}^{\infty} \frac{(pt)^{2j}}{(k+2j-1))!}}{1 + \sum_{j=1}^{\infty} \frac{p^{2j}}{(2j+1)!}}\,, \tag{12}$$

for all k. The above representation does not have singularity for $p = 0$.
To approximate (12), we use finite instead of infinite sums. In that case, (12) is approximated by

$$G_k^m(p,t) = t^{k-1} \frac{\frac{1}{(k-1)!} + \sum_{j=1}^{m} (pt)^{2j}/(k+2j-1)!}{1 + \sum_{j=1}^{m} p^{2j}/(2j+1)!}\,. \tag{13}$$

We define numbers p_k^m:

$$p_k^m := \max \left\{ p \geq 0 : \left| \frac{\tilde{\varphi}_k(p,t) - G_k^m(p,t)}{\tilde{\varphi}_k(p,t)} \right| < \epsilon \,, t \in [0,1] \right\}, \tag{14}$$

defining the region where a relative error in evaluation of $\tilde{\varphi}_k$ is less than ϵ. The numbers p_k^m are found experimentally calculating $\tilde{\varphi}_k(p,t)$ according to formula (10) in quadri precision arithmetic, while G_k^m is calculated in double precision arithmetic. We have used tolerance $\epsilon = 10^{-15}$. *Table 1* shows values of p_k^m for different k and m. It is noteworthy that p_k^m almost does not depend on k.

	m					
k	4	5	6	7	8	9
2	0.08	0.21	0.41	0.68	1.02	1.41
3	0.08	0.21	0.41	0.68	1.02	1.41
4	0.08	0.21	0.41	0.68	1.02	1.42
5	0.08	0.21	0.41	0.68	1.02	1.42
6	0.08	0.21	0.41	0.68	1.02	1.42
7	0.10	0.21	0.41	0.69	1.02	1.42
8	0.10	0.21	0.40	0.68	1.01	1.40

Table 1. *Numbers p_k^m defined by (14), for $\epsilon = 10^{-15}$*

4.2. Case when pt is small

Figures 1 and *2* show a large relative error in evaluation of $\tilde{\varphi}_k$ not only for small p but also for p relatively large and t small. When product pt is small, but p relatively large (e.g. $p \geq p_k^m$), we do not have a problem with underflow in the denominator (as in 4.1), but it is still possible to loose significant digits in evaluation of the numerator. *Figure 2* shows relative errors in evaluation of $\tilde{\varphi}_6$ in this situation.

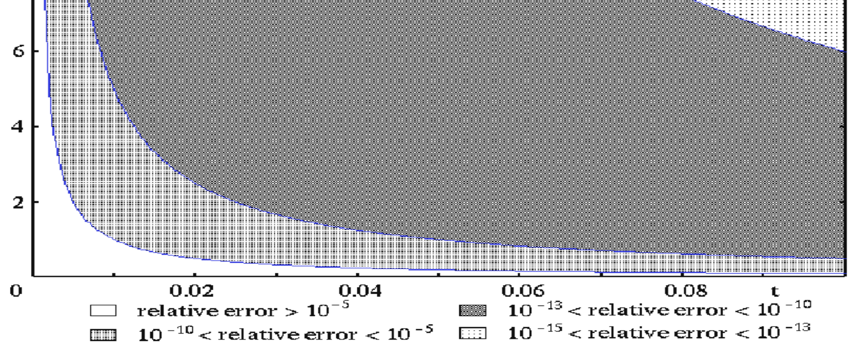

Figure 2. *Relative error for evaluating $\tilde{\varphi}_6$ by (10), on $[1,10]\times[0,0.1]$ in double precision arithmetic.*

To avoid a loss of significant digits, we expand the numerator in Taylor series:

$$\tilde{\varphi}_k(p,t) = 2e^{-p}pt^{k-1}\frac{\sum_{j=0}^{\infty}\frac{(pt)^{2j}}{(k+2j-1)!}}{(1-e^{-p})(1+e^{-p})}\,. \tag{15}$$

Substitution

$$p^{k-2}\sinh p = p^{k-2}\frac{(1-e^{-p})(1+e^{-p})}{2e^{-p}}\,,$$

is used to avoid a potential overflow in the denominator for large p.

As in (12), we use finite instead of infinite sum to approximate (15). In that case, (15) is approximated by

$$\tilde{G}_k^m = 2e^{-p}pt^{k-1}\frac{\sum_{j=0}^{m}(pt)^{2j}/(k+2j-1)!}{(1-e^{-p})(1+e^{-p})}\,. \tag{16}$$

Region where a relative error for $\tilde{G}_k^m$ is less than ϵ is described by $pt \le u_k^m$, where

$$u_k^m = \max\left\{pt>0 : \left|\frac{\tilde{\varphi}_k(p,t)-\tilde{G}_k^m(p,t)}{\tilde{\varphi}_k(p,t)}\right| < \epsilon\,, t\in(0,1], p>0\right\}. \tag{17}$$

As the numbers p_k^m (14), the numbers u_k^m are found by a numerical experiment for $\epsilon = 10^{-15}$.

	m					
k	4	5	6	7	8	9
2	0.07	0.20	0.40	0.67	1.00	1.40
3	0.09	0.24	0.47	0.77	1.14	1.58
4	0.10	0.27	0.52	0.87	1.27	1.74
5	0.12	0.31	0.59	0.97	1.40	1.92
6	0.13	0.34	0.65	1.06	1.53	2.08
7	0.14	0.37	0.71	1.14	1.65	2.23
8	0.16	0.42	0.79	1.25	1.81	2.41

Table 2. *Numbers u_k^m, defined by (17), for $\epsilon = 10^{-15}$*

4.3. Case when p is large

When p is large, we have the problem of machine overflow. For example, in calculating $\tilde{\varphi}_6$ by (10) overflow occurs for $p \ge 710$ in double precision arithmetic, and for $p \ge 89$ in single precision arithmetic.

For large p, we rewrite (10) as

$$\begin{aligned}\tilde{\varphi}_k(p,t) &= \frac{\frac{e^{pt}-(-1)^k e^{-pt}}{2} - P_k(pt)}{p^{k-2}\frac{e^p - e^{-p}}{2}} \\ &= e^{-p(1-t)}\frac{1-2e^{-pt}P_k(pt)-(-1)^k e^{-2pt}}{p^{k-2}(1-e^{-p})(1+e^{-p})}\,. \qquad (18)\end{aligned}$$

Because of

$$\lim_{x\to\infty} x^k e^{-x} = 0\,,\ \text{for all } k\,,$$

there exist numbers V_k such that

$$2P_k(y)e^{-y} < \varepsilon \text{ for } y > V_k,$$

where ε is machine precision. Since $e^{-p} < 2P_k(pt)e^{-pt}$ holds for $pt > V_k$, $\tilde{\varphi}_k(p,t)$ can be approximated by an asymptotic formula:

$$H_k(p,t) = \frac{e^{-p(1-t)}}{p^{k-2}}\,. \qquad (19)$$

By using (19) we avoid the problem of machine overflow.

Numbers V_k are found experimentally. We found the smallest integer y satisfying

$$fl(1+2P_k(y)e^{-y}) = 1. \qquad (20)$$

Here $fl(C)$ stands for “floating point result of C”. Values for V_k in double precision arithmetic ($\varepsilon \approx 10^{-16}$) are presented in *Table 3*.

k	2	3	4	5	6	7	8
V_k	19	38	42	45	48	50	54

Table 3. *Numbers V_k, defined by (20), for double precision arithmetic*

4.4. Regular case

Outside the regions defined by 4.1–4.3 we use formula (10).

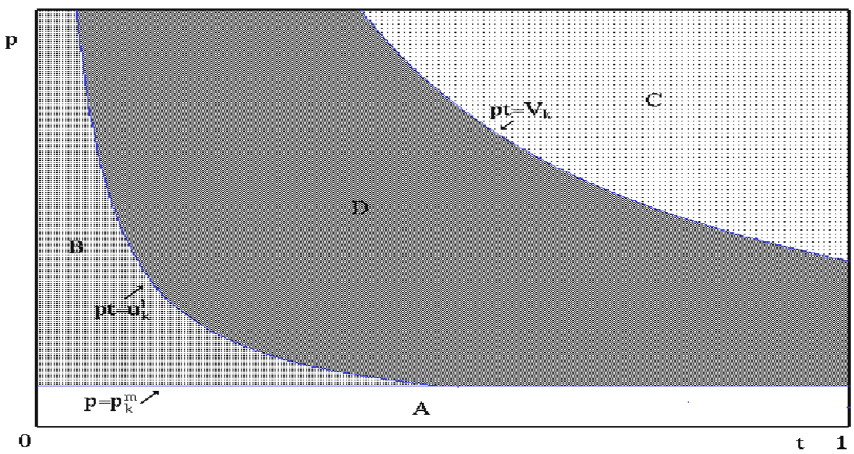

Figure 3. *Regions for evaluating of $\tilde{\varphi}_k$: A – formula (13), B – formula (16), C – formula (19), D – formula (10).*

To summarize, evaluation of $\tilde{\varphi}_k(p,t)$ is divided in four different cases:

A) When $p \leq p_k^m$, for arbitrary m, we approximate $\tilde{\varphi}_k(p,t)$ by $G_k^m(p,t)$ (13).

B) When $p > p_k^m$ and $pt \leq u_k^l$, for arbitrary m and l, we approximate $\tilde{\varphi}_k(p,t)$ by $\tilde{G}_k^l(p,t)$ (16).

C) When $pt > V_k$, we approximate $\tilde{\varphi}_k(p,t)$ by $H_k^m(p,t)$ (19).

D) Otherwise, we evaluate $\tilde{\varphi}_k(p,t)$ by (10).

Acknowledgments

Supported by grant 037011 by Croatian Ministry of Science and Technology.

References

[1] J. E. Flaherty, W. Mathon, *Collocation with polynomial and tension splines for singularly-perturbed boundary value problems*, SIAM J. Sci. Stat. Comput. **1**(1980), 260–289.

[2] P. E. Koch, T. Lyche, *Construction of exponential tension B-spline of arbitrary order*, in: Curves and Surfaces, (P. J. Laurent, A. Le Mehaute and L. L. Schumaker, Eds.), Academic Press, N.Y., 1991, 255–258.

[3] B. I. Kvasov, *Local bases for generalized cubic splines*, Russian J. Numer. Anal. Math. Modelling **10**(1995), 49–80.

[4] M. Marušić, *Stable calculation by splines in tension*, Grazer Math. Ber. **328**(1996), 65–76.

[5] M. Marušić, M. Rogina, *A collocation method for singularly perturbed two-point boundary value problems with splines in tension*, Adv. Comput. Math. **6**(1996), 65–76.

[6] M. Marušić, M. Rogina, *B-splines in tension*, In: Proceedings of VII Conference on Applied Mathematics, (R. Scitovski, Ed.), Osijek, 1989, 129–134.

[7] B. J. Mc Cartin, *Theory of exponential splines* J. Approx. Theory **66**(1991), 1–23.

[8] M. Rogina, *Basis of splines associated with some singular differential operators*, BIT **32**(1992), 496–505.

[9] D. G. Schweikert, *An interpolating curve using a spline in tension*, J. Math. Physics **45**(1966), 312–317.

[10] H. Späth, *Exponential spline interpolation*, Computing **4**(1969), 225–233.