Journal of Computational and Applied Mathematics 236 (2012) 4383–4397

Contents lists available at SciVerse ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A shape-preserving approximation by weighted cubic splines

Tae-wan Kim[a,*], Boris Kvasov[b]

[a] Department of Naval Architecture and Ocean Engineering, Research Institute of Marine Systems Engineering, Seoul National University, Seoul 151-744, Republic of Korea

[b] Department of Mathematical Modeling, Institute of Computational Technologies, Russian Academy of Sciences, Novosibirsk 630090, Russia

ARTICLE INFO

Article history:
Received 25 August 2011
Received in revised form 3 April 2012

Keywords:
Shape-preserving interpolation and approximation
Differential multipoint boundary value problem
Weighted cubic splines
Error bounds
Adaptive choice of shape control parameters
Recurrence relations for weighted B-splines

ABSTRACT

This paper addresses new algorithms for constructing weighted cubic splines that are very effective in interpolation and approximation of sharply changing data. Such spline interpolations are a useful and efficient tool in computer-aided design when control of tension on intervals connecting interpolation points is needed. The error bounds for interpolating weighted splines are obtained. A method for automatic selection of the weights is presented that permits preservation of the monotonicity and convexity of the data. The weighted B-spline basis is also well suited for generation of freeform curves, in the same way as the usual B-splines. By using recurrence relations we derive weighted B-splines and give a three-point local approximation formula that is exact for first-degree polynomials. The resulting curves satisfy the convex hull property, they are piecewise cubics, and the curves can be locally controlled with interval tension in a computationally efficient manner.

1. Introduction

C^2 cubic splines play a very important role in practical methods of spline approximation. However, such splines do not retain the shape properties of the data, a drawback known as the *shape-preserving approximation problem*. During the past few decades, different authors have developed various algorithms of spline approximation with both local and global shape control. They include exponential, hyperbolic, computationally more efficient rational splines [1–4], etc. The tension parameters are mainly viewed as an interactive design tool for manipulating the shape of a spline curve. A very detailed literature review of algorithms for passing a curve through data points so as to preserve the shape of the data is given in [5].

In this paper we consider the weighted cubic splines introduced in [6] (see also [7–12]). Such splines are C^1 piecewise cubic splines where weights are shape parameters. They are a natural generalization of cubic splines, describing from a physical point of view, an inhomogeneous elastic beam supported at some points. The idea is that the elastic property of the material is kept piecewise constant, and then it follows by variational arguments that C^2 continuity is lost, but is replaced by known jumps in second derivatives. The theory was steadily developed over years, and now weighted splines are known to possess a *B*-spline basis [10], optimal in a certain sense, and they are Chebyshev splines with sections in appropriate Extended Complete Chebyshev (ECC)-spaces [13,14].

To treat the weighted splines in a general setting we suggest using the approach of the second author (see [15–17]) in which such splines are defined as solutions of the differential multipoint boundary value problems. We give direct algorithms to construct the weighted cubic splines, prove error bounds, and show how to choose weight (tension) parameters automatically depending on the data monotonicity and convexity. Such algorithms for automatic selection of

[*] Corresponding author. Tel.: +82 10 2739 7364; fax: +82 2888 9298.
E-mail addresses: taewan@snu.ac.kr (T.-w. Kim), kvasovbi@gmail.com (B. Kvasov).
URL: http://caditlab.snu.ac.kr/ (T.-w. Kim).

0377-0427/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2012.04.001

the weight parameters are based on the sufficient conditions of monotonicity and convexity for C^2 cubic splines [15]. Due to the simplicity and the reliability of the corresponding algorithms, their use in CAD systems can be considered.

Normalized *B*-bases present optimal shape preserving properties for the representation of curves when control polygons are used [18,19]. Rational cubic *B*-spline bases with point and interval shape control parameters were suggested in [1,2]. The general approach in [20,15] is an alternative which permits to construct different kinds of tension *B*-splines, including rational, weighted, etc. This allowed us to give explicit formulas for normalized weighted *B*-splines in a simpler way than based on the Bernstein–Bézier representation in [10].

Each weighted *B*-spline is a non-negative cubic spline that is non-zero only on four intervals. The weighted *B*-splines form a partition of unity; that is, they sum to one. Curves generated by summing control points multiplied by the weighted *B*-splines have some desirable shape properties, including the local convex hull property. The different weights are built into the basis functions so that the resulting control point curve is a piecewise cubic with local control of interval tension. Recurrence formulas for weighted *B*-splines offer valuable insight into their geometric behavior. Knot insertion algorithms for weighted *B*-splines [21,22] produce numerically stable formulas for weighted *B*-splines.

This paper is divided into eight sections. In Section 2, we define weighted splines and give algorithms for their construction. Section 3 provides error bounds for weighted splines. In Section 4 a method for adaptive selection of weights is presented that allows the monotonicity and convexity of the data to be preserved automatically. In Section 5 by using recurrence relations we construct weighted *B*-spline basis with tension properties and give a three-point formula for local approximation. Section 6 uses a weighted *B*-spline basis to form a control point sum that will yield a curve with the convex hull property. We conclude with numerical examples of functional and curve interpolation and final comments in Sections 7 and 8.

2. Weighted splines

Suppose that we are given the data

$$(x_i, f_i), \quad i = 0, \dots, N+1, \tag{1}$$

where $a = x_0 < x_1 < \cdots < x_{N+1} = b$. Define

$$f[x_i, x_{i+1}] = (f_{i+1} - f_i)/h_i, \quad h_i = x_{i+1} - x_i,\ i = 0, \dots, N.$$

Data (1) are called monotonically increasing if

$$f[x_i, x_{i+1}] \geq 0, \quad i = 0, \dots, N,$$

and are called convex if

$$f[x_{i-1}, x_i, x_{i+1}] \geq 0, \quad i = 1, \dots, N.$$

The shape-preserving interpolation problem consists of constructing a sufficiently smooth function S such that $S(x_i) = f_i$ for $i = 0, \dots, N+1$ and S is monotonic and convex on the intervals of monotonicity and convexity of the input data.

The shape-preserving interpolation problem can be very efficiently solved by using weighted splines. Suppose that w is a function on $[a, b]$ satisfying $0 < m \leq w(x) \leq M$ for all $x \in [a, b]$. We will call w the *weight function*.

Definition 1. The weighted spline S is defined as the solution to the differential multipoint boundary value problem (DMBVP)

$$\frac{d^2}{dx^2}\left(w(x)\frac{d^2S}{dx^2}\right) = 0 \quad \text{for all } x \in (x_i, x_{i+1}),\ i = 0, \dots, N, S \in C^k[a, b],\ k \geq 1. \tag{2}$$

If $w(x) \equiv 1$ and $k = 2$, then we obtain a conventional C^2 cubic spline.

In the case when w is piecewise constant on the subdivision $(w(x) \equiv w_i$ for $x \in [x_i, x_{i+1}), i = 0, \dots, N)$ S'' is a piecewise linear function and thus S is a piecewise cubic function, but since w is discontinuous, the solution is only C^1 (see [6,8,9,7,10,11]). The second derivative satisfies the conditions

$$w_{i-1}S''(x_i^-) = w_iS''(x_i^+), \quad i = 1, \dots, N. \tag{3}$$

If we choose $w(x) = 1/q(x)$, where q is a continuous piecewise linear function on a given subdivision, then solution S will be a piecewise polynomial function of degree 4 belonging to C^2. This solution is called a q-spline and is investigated in [23].

For a more general form of the weight function w the DMVBP can be solved by using a finite-difference method (see [24,16,17]). An alternative and perhaps modern view is to refer to weighted cubic splines as to splines with cubic sections [25,26]. In this paper we shall study in detail the case $k = 1$ with w being a piecewise constant where the solution is a cubic spline belonging to C^1.

We assume that cubic spline S satisfies the interpolation conditions

$$S(x_i) = f_i, \quad i = 0, \dots, N+1. \tag{4}$$

To define a spline uniquely we also need boundary conditions. The most common are the following endpoint constraints:

1. First derivative endpoint conditions: $S'(a) = f_0'$ and $S'(b) = f_{N+1}'$.
2. Second derivative endpoint conditions: $S''(a) = f_0''$ and $S''(b) = f_{N+1}''$.
3. Periodic endpoint conditions: $S^{(r)}(a) = S^{(r)}(b)$, $r = 0, 1$, $w_0 S''(x_0^+) = w_N S''(x_{N+1}^-)$.
4. "Not-a-knot" endpoint conditions where adjacent polynomials nearest to the endpoints of the interval $[a, b]$ coincide: $S_0(x) \equiv S_1(x)$ and $S_{N-1}(x) \equiv S_N(x)$, that is, $S'''(x_i^-) = S'''(x_i^+)$, $w_{i-1} = w_i$, $i = 1, N$.

If the derivative values at the endpoints are unknown one can employ cubic Lagrange polynomials by setting

$$f_0^{(r)} = L_{3,0}^{(r)}(a) \quad \text{and} \quad f_{N+1}^{(r)} = L_{3,N-2}^{(r)}(b), \quad r = 1, 2.$$

The derivative values at the endpoint conditions must be adjusted to the behavior of the data. Otherwise, we can obtain an incompatibility with the shape-preserving restrictions [15]. For example, in the case of the second derivative boundary conditions we can use the restrictions

$$f_0'' f[x_0, x_1, x_2] \geq 0, \qquad f_{N+1}'' f[x_{N-1}, x_N, x_{N+1}] \geq 0.$$

If we set $w_i = 1$ for all i in (3), then the solution to problem (2)–(4) is a cubic spline of the class C^2, which gives a smooth curve but does not always preserve the monotonicity and convexity of the input data. When by using the weighted spline, if w_i is large on one interval relative to the other intervals, then S'' is forced to be small in magnitude on that interval, hence S will be more linear on that interval. Similarly, smaller relative weights allow S'' to take on larger values. The term relative is used because if all weights are multiplied by a positive constant, then the resulting weighted spline would be the same.

Let us consider an algorithm for the construction of the weighted cubic spline S. We will use the notation

$$M_i = w_{i-1} S''(x_i^-) = w_i S''(x_i^+), \quad i = 1, \dots, N,$$
$$M_0 = w_0 S''(x_0^+), \qquad M_{N+1} = w_N S''(x_{N+1}^-).$$

For $x \in [x_i, x_{i+1}]$ one has

$$S(x) = f_i(1-t) + f_{i+1}t - t(1-t)\frac{h_i^2}{6w_i}\Big[(2-t)M_i + (1+t)M_{i+1}\Big], \tag{5}$$

where $t = (x - x_i)/h_i$, $h_i = x_{i+1} - x_i$.

To find the unknown coefficients M_i, $i = 0, \dots, N+1$, one must use the derivative of (5), which is

$$S'(x) = f[x_i, x_{i+1}] - \frac{h_i}{6w_i}\Big[(2 - 6t + 3t^2)M_i + (1 - 3t^2)M_{i+1}\Big].$$

As $S'(x_i^-) = S'(x_i^+)$, $i = 1, \dots, N$, we find

$$\frac{h_{i-1}}{w_{i-1}}M_{i-1} + 2\left(\frac{h_{i-1}}{w_{i-1}} + \frac{h_i}{w_i}\right)M_i + \frac{h_i}{w_i}M_{i+1} = 6\delta_i f, \quad i = 1, \dots, N, \tag{6}$$

where $\delta_i f = f[x_i, x_{i+1}] - f[x_{i-1}, x_i]$.

For simplicity, assume that system (6) is completed by second derivative boundary conditions (clearly other end conditions are also appropriate). Then (6) defines a diagonally dominant, tridiagonal linear system. Hence there exists a unique solution which can be easily calculated by use of the tridiagonal LU decomposition algorithm.

In some cases, it is more convenient to use a different algorithm for constructing weighted cubic interpolating splines. Such an algorithm is based on the representation of the spline through endpoint values of its first derivative.

Let us denote $m_i = S'(x_i)$, $i = 0, \dots, N+1$. On interval $[x_i, x_{i+1}]$, one can write the following formula for the cubic interpolating spline:

$$S(x) \equiv S_i(x) = f_i(1-t)^2(1+2t) + f_{i+1}t^2(3-2t) + m_i h_i t(1-t)^2 - m_{i+1}h_i t^2(1-t), \quad x \in [x_i, x_{i+1}]. \tag{7}$$

By differentiating formula (7) twice one finds

$$S_i''(x) = \frac{2}{h_i}\Big(3(1-2t)f[x_i, x_{i+1}] - (2-3t)m_i - (1-3t)m_{i+1}\Big).$$

Now condition (3) gives us

$$\lambda_i m_{i-1} + 2m_i + \mu_i m_{i+1} = 3\lambda_i f[x_{i-1}, x_i] + 3\mu_i f[x_i, x_{i+1}], \quad i = 1, \dots, N, \tag{8}$$

where

$$\lambda_i = \frac{w_{i-1}h_i}{w_{i-1}h_i + w_i h_{i-1}}, \quad \mu_i = 1 - \lambda_i. \tag{9}$$

To complete the system (8) we can use endpoint conditions. The corresponding systems of linear equations again have unique solutions providing the existence and uniqueness of a cubic weighted spline.

3. Error estimates

Let us consider the case when the initial data (1) are obtained from some smooth function f, that is, $f_i = f(x_i)$, $i = 0, \dots, N+1$. We would like to estimate the approximation error of this function by a cubic weighted spline.

We will use weights to obtain shape preserving interpolants. In general this gives us second order of approximation. One can raise the convergence order by a special choice of weights. If they are all equal, one obtains cubic C^2 splines with fourth order of approximation. We point also at weights which give rise to the third order. Because of this reason we compute the error of approximation once the data is second derivative continuous and again when it is C^4 respectively.

Theorem 1. *Let a cubic weighted spline $S \in C^1[a,b]$, with the first derivative boundary conditions $S'(x_0) = f_0'$ and $S'(x_{N+1}) = f_{N+1}'$, interpolate the values $f_i = f(x_i)$, $i = 0, \dots, N+1$, of some function $f \in C^2[a,b]$. Then the following error estimates hold:*

$$\|S^{(r)}(x) - f^{(r)}(x)\|_C \le C_r \bar{h}^{2-r} \|f''\|_C, \quad r = 0, 1, \tag{10}$$

where $C_0 = 13/48$, $C_1 = 0.86229$, and $\bar{h} = \max_i h_i$.

Proof. Let S_H be a cubic Hermite spline that satisfies the interpolation conditions

$$S_H(x_i) = f_i, \qquad S_H'(x_i) = f_i' = f'(x_i), \quad i = 0, \dots, N+1.$$

We have

$$S^{(r)}(x) - f^{(r)}(x) = \left[S^{(r)}(x) - S_H^{(r)}(x)\right] + \left[S_H^{(r)}(x) - f^{(r)}(x)\right], \quad r = 0, 1. \tag{11}$$

Spline S_H on the interval $[x_i, x_{i+1}]$ can be written in the form

$$S_H(x) = (1-t)^2(1+2t)f_i + t^2(3-2t)f_{i+1} + t(1-t)^2 h_i f_i' - t^2(1-t)h_i f_{i+1}',$$

where $t = (x - x_i)/h_i$.

By using a Taylor series to expand the values $f_i^{(r)}, f_{i+1}^{(r)}$, $r = 0, 1$, at the point $x \in [x_i, x_{i+1}]$, with the remainder in integral form, one obtains

$$\begin{aligned} R_H(x) &= S_H(x) - f(x) \\ &= h_i^2 \int_0^t \psi_1(t,\tau) f''(x_i + \tau h_i)d\tau + h_i^2 \int_t^1 \psi_2(t,\tau) f''(x_i + \tau h_i)d\tau, \end{aligned}$$

where

$$\begin{aligned} \psi_1(t,\tau) &= (1-t)^2[(1+2t)\tau - t], \\ \psi_2(t,\tau) &= t^2[(3-2t)(1-\tau) - (1-t)]. \end{aligned}$$

Applying the Hölder inequality, we find

$$|R_H(x)| \le \left[\int_0^t |\psi_1(t,\tau)|d\tau + \int_t^1 |\psi_2(t,\tau)|d\tau\right] h_i^2 \|f''\|_C \le \frac{h_i^2}{16}\|f''\|_C.$$

Analogously, we get the estimate

$$|R_H'(x)| \le 0.24750 h_i \|f''\|_C.$$

Taking into account formula (7), we find for the first term on the right side of the equality (11)

$$\begin{aligned} &|S(x) - S_H(x)| \le h_i t(1-t)\bar{q} \le h_i \bar{q}/4, \\ &|S'(x) - S_H'(x)| \le \left[(1-t)|1-3t| + t|2-3t|\right]\bar{q}, \end{aligned}$$

where $\bar{q} = \max_i |q_i|$, $q_i = m_i - f_i$.

Let us estimate q_i, $i = 0, \dots, N+1$. To this end, we rewrite system (8) with the first derivative boundary conditions in the form

$$\begin{aligned} &q_0 = 0, \\ &\lambda_i q_{i-1} + 2q_i + \mu_i q_{i+1} = c_i, \quad i = 1, \dots, N, \\ &q_{N+1} = 0, \end{aligned} \tag{12}$$

where $c_i = 3\lambda_i f[x_{i-1}, x_i] + 3\mu_i f[x_i, x_{i+1}] - \lambda_i f_{i-1}' - 2f_i' - \mu_i f_{i+1}'$.

By using the Taylor formula with the remainder in integral form one finds

$$c_i = \lambda_i h_{i-1} \int_0^1 (1-3\tau) f''(x_{i-1} + \tau h_{i-1})d\tau + \mu_i h_i \int_0^1 (2-3\tau) f''(x_i + \tau h_i)d\tau.$$

From here one obtains

$$\begin{aligned}|c_i| &\le \left[\lambda_i h_{i-1}\int_0^1 |1-3\tau|d\tau + \mu_i h_i \int_0^1 |2-3\tau|d\tau\right]\|f''\|_C \\ &= \frac{5}{6}(\lambda_i h_{i-1}+\mu_i h_i)\|f''\|_C \le \frac{5}{6}\overline{h}\|f''\|_C.\end{aligned}$$

Let $A\,q = c$ be a matrix form of system (12). Matrix A has a diagonal dominance and one can easily show (see, e.g., [27]) that $\|A^{-1}\|_\infty \le 1$. This gives us the estimate

$$\overline{q} = \max_i |q_i| = \|q\| \le \|A^{-1}\|_\infty \,\|c\| \le \max_i |c_i| \le \frac{5}{6}\overline{h}\|f''\|_C.$$

Now combining the obtained inequalities and applying them to the right side of equality (11), we come to the error bounds (10). This proves the theorem. □

It is worth noting that the estimates (10) are also valid in cases of second derivative, periodic, and "natural" boundary conditions when $S''(a) = S''(b) = 0$. In the case of "not-a-knot" boundary conditions the error bounds (10) are valid with larger values of the constants (see [27]).

Let us consider now the case in which the interpolated function f is smoother.

Theorem 2. *Let a cubic weighted spline $S \in C^1[a,b]$, with the first derivative boundary conditions $S'(x_0) = f_0'$ and $S'(x_{N+1}) = f_{N+1}'$, interpolate the values $f_i = f(x_i)$, $i = 0,\dots,N+1$, of some function $f \in C^4[a,b]$. Then the following error estimates hold:*

$$\|S^{(r)}(x) - f^{(r)}(x)\|_C \le C_r \overline{h}^{2-r}\|f''\|_C + \tilde{C}_r\overline{h}^{4-r}\|f^{IV}\|_C, \quad r = 0, 1, \tag{13}$$

where $C_0 = 1/8$, $C_1 = 1/2$, $\tilde{C}_0 = 5/384$, and $\tilde{C}_1 = 1/24$.

Proof. For the second term on the right side of equality (11), by a Taylor series expansion we have

$$\begin{aligned}R_H(x) &= S_H(x) - f(x) \\ &= \frac{h_i^4}{6}\left\{\int_0^t \psi_1(t,\tau)f^{IV}(x_i+\tau h_i)d\tau + \int_t^1 \psi_2(t,\tau)f^{IV}(x_i+\tau h_i)d\tau\right\},\end{aligned}$$

where

$$\begin{aligned}&\psi_1(t,\tau) = (1-t)^2\tau^2[-3t+(1+2t)\tau], \\ &\psi_2(t,\tau) = t^2(1-\tau)^2[-3(1-t)+(3-2t)(1-\tau)].\end{aligned}$$

This gives us

$$|R_H(x)| \le \frac{t^2(1-t)^2}{4!}h_i^4\|f^{IV}\|_C \le \frac{h_i^4}{384}\|f^{IV}\|_C, \quad t \in [0,1].$$

Analogously, we obtain

$$\begin{aligned}&|R_H'(x)| \le \frac{1}{12}h_i^3 t(1-t)(1-2t)\|f^{IV}\|_C, \quad t \in [0,1/3], \\ &|R_H'(x)| \le \frac{h_i^3}{12}(1-t)\left[t(1-2t) + \frac{(1-3t)^4}{8t^3}\right]\|f^{IV}\|_C, \quad t \in [1/3,1/2].\end{aligned}$$

Let us consider again the system (12). By a Taylor series expansion we have

$$c_i = \frac{h_{i-1}h_i}{2}f_i''\frac{w_i - w_{i-1}}{w_i h_{i-1} + w_{i-1}h_i} + R_i, \tag{14}$$

where

$$R_i = \frac{1}{2}\lambda_i h_{i-1}^3\int_0^1 \tau^2(1-\tau)f^{IV}(x_{i-1}+\tau h_{i-1})d\tau - \frac{1}{2}\mu_i h_i^3\int_0^1 \tau(1-\tau)^2 f^{IV}(x_i+\tau h_i)d\tau.$$

By using the Hölder inequality, we get

$$|R_i| \le \frac{1}{24}(\lambda_i h_{i-1}^3 + \mu_i h_i^3)\|f^{IV}\|_C \le \frac{\overline{h}^3}{24}\|f^{IV}\|_C$$

and

$$|c_i| \le \frac{1}{2}(\lambda_i h_{i-1} + \mu_i h_i)\|f''\|_C + |R_i| \le \frac{\bar{h}}{2}\|f''\|_C + \frac{\bar{h}^3}{24}\|f^{IV}\|_C.$$

Thus, as before, one arrives at the estimate

$$\bar{q} \le \max_i |c_i| \le \frac{\bar{h}}{2}\|f''\|_C + \frac{\bar{h}^3}{24}\|f^{IV}\|_C.$$

Now collecting the obtained estimates, we come to the error bounds (13). This proves the theorem. □

The remark after Theorem 1 related to different types of the boundary conditions is also valid for Theorem 2.

The orders of approximation in the error bounds (13) can be increased by using special choice of the weights w_i. If they are all equal, then in (14) one has $c_i = R_i$ and the estimates (13) are valid with $C_0 = C_1 = 0$. Thus the convergence orders rise to $O(\bar{h}^{4-r})$, $r = 0, 1$. Let $w_i = (1 + \Delta_i^2)^{-1}$, $\Delta_i = f[x_i, x_{i+1}]$. Then, in (14) we have $c_i = p_i + R_i$, where

$$p_i = \frac{h_{i-1}h_i}{2} f_i'' \frac{w_i - w_{i-1}}{w_i h_{i-1} + w_{i-1} h_i} = \frac{h_{i-1}h_i}{2} f_i'' \frac{\Delta_{i-1}^2 - \Delta_i^2}{h_{i-1}(1 + \Delta_{i-1}^2) + h_i(1 + \Delta_i^2)}.$$

Because $\Delta_{i-1} - \Delta_i = -(h_{i-1} + h_i)f''(\xi)$ and $\Delta_{i-1} + \Delta_i = 2f'(\eta)$ with $\xi, \eta \in (x_{i-1}, x_{i+1})$ and

$$\frac{h_{i-1}h_i(h_{i-1} + h_i)}{h_{i-1}(1 + \Delta_{i-1}^2) + h_i(1 + \Delta_i^2)} \le \bar{h}^2$$

one obtains

$$|p_i| \le |f_i''|\,|f''(\xi)|\,|f'(\eta)|\bar{h}^2 \le \|f''\|_C^2 \|f'\|_C \bar{h}^2.$$

Thus, the orders of approximation in the estimates (13) are $O(\bar{h}^{3-r})$, $r = 0, 1$.

4. Adaptive choice of shape-control parameters

In this section, we shall describe the interpolation for points $\Delta :\ a = x_0 < x_1 \cdots < x_{N+1} = b$,

$$S(x_i) = f_i = f(x_i),$$

by weighted splines S that preserve the shape of the data (x_i, f_i), $i = 0, \ldots, N+1$. For example, if the function f is monotonic or convex on some interval $[x_j, x_k]$, we would like to have a spline S that also has these properties. To achieve this, we take spline S with knots at the points x_i, and we select the weights (shape control parameters) to ensure the desired properties of S. The main point, however, is to determine whether the error of approximation $\|S - f\|$ remains small under the proposed algorithms that describe S.

For definiteness we will assume that the data $f[x_i, x_{i+1}] \ge 0$, $i = 0, \ldots, N$, are monotonically increasing. Eqs. (8) for the slopes of a weighted cubic spline do not differ by form from the corresponding equations for a C^2 conventional cubic spline. This permits us to rewrite the sufficient conditions of monotonicity for a C^2 cubic spline (see [15, p. 112]) for the case of a weighted cubic spline. In particular, the following result holds.

Theorem 3. *Let a cubic weighted spline $S \in C^1[a, b]$, with the first derivative boundary conditions $S'(x_0) = f_0'$ and $S'(x_{N+1}) = f_{N+1}'$, interpolate the monotonic data $\{f_i\}$, $i = 0, \ldots, N + 1$. If the following inequalities are valid:*

$$\begin{aligned} &0 \le f_0' \le 3f[x_0, x_1], \quad 0 \le f_{N+1}' \le 3f[x_N, x_{N+1}], \\ &\lambda_i f[x_{i-1}, x_i] \le (1 + \lambda_i) f[x_i, x_{i+1}], \\ &\mu_i f[x_i, x_{i+1}] \le (1 + \mu_i) f[x_{i-1}, x_i], \quad i = 1, \ldots, N, \end{aligned} \tag{15}$$

then $S'(x) \ge 0$ for all $x \in [a, b]$, that is, S is monotonic on $[a, b]$.

Taking into account formulas (9) we can rewrite inequalities (15) in the form

$$\frac{w_{i-1}}{w_i}\frac{h_i}{h_{i-1}} \ge \frac{f[x_i, x_{i+1}]}{f[x_{i-1}, x_i]} - 2, \qquad \frac{w_i}{w_{i-1}}\frac{h_{i-1}}{h_i} \ge \frac{f[x_{i-1}, x_i]}{f[x_i, x_{i+1}]} - 2, \quad i = 1, \ldots, N. \tag{16}$$

It follows from here that by choosing a large value of the ratio w_{i-1}/w_i one can essentially reduce the restrictions on the data sufficient to obtain monotonicity of the weighted cubic spline. For usual C^2 cubic splines, we have $w_{i-1}/w_i = 1$.

First, we note that for each i one of the inequalities (16) is fulfilled. Let, for example, the inequality $f[x_i, x_{i+1}] > f[x_{i-1}, x_i]$ hold. Then the second inequality in (16) is valid. The first inequality can be satisfied by choosing w_i directly from this inequality. We suggest the following recurrence algorithm.

Let the parameter w_{i-1} be known. If inequalities (16) are valid for $w_i = w_{i-1}$ then we set $w_i = w_{i-1}$. Otherwise, we calculate w_i from inequalities (16), which are violated in case $w_i = w_{i-1}$, by replacing the inequality sign by the equality sign. We begin this algorithm with $w_0 = 1$ and easily find all parameters $\{w_i\}$ providing the monotonicity of the weighted cubic spline for any monotonic data. If on some step $w_i < \varepsilon$ ($w_i > \varepsilon^{-1}$) then we assign $w_i = \varepsilon$ ($w_i = \varepsilon^{-1}$), where ε is a small enough positive number to avoid underflow (overflow).

This algorithm has a couple of important features. On segments where the data are changing slowly the weighted spline will have two continuous derivatives, inheriting all good properties of the C^2 cubic spline. The second derivative will be discontinuous only at knots of the mesh Δ where sharp changes in the slopes are needed.

Let us note that in case of the "natural" boundary conditions, when $S''(a) = S''(b) = 0$, one can use conditions (16) as sufficient conditions of monotonicity (see [15, p. 112]).

Let us consider the following choice of the weight parameters:

$$w_i = \left[1 + C_i(f[x_i, x_{i+1}])^2\right]^{-\alpha_i}, \quad C_i \ge 1,\ \alpha_i \ge 0,\ i = 0, \dots, N. \tag{17}$$

If $\alpha_i = 0$ for all i then this gives us $w_i = 1$ for all i and we obtain a C^2 cubic spline. We will assume that the mesh is uniform and $f[x_i, x_{i+1}] > 1$. The latter condition can be achieved by scaling in x or y. If $f[x_i, x_{i+1}] > f[x_{i-1}, x_i]$ then the second inequality in (16) is fulfilled. The first inequality takes the form

$$\frac{(1 + C_i(f[x_i, x_{i+1}])^2)^{\alpha_i}}{(1 + C_{i-1}(f[x_{i-1}, x_i])^2)^{\alpha_{i-1}}} \ge \frac{f[x_i, x_{i+1}]}{f[x_{i-1}, x_i]} - 2. \tag{18}$$

In the particular case when $C_{i-1} = C_i = 1$ and $\alpha_{i-1} = \alpha_i = 1$ inequality (18) is equivalent to the condition

$$g(\Delta_{i-1}, \Delta_i) = (1 + \Delta_i^2)\Delta_{i-1} - (1 + \Delta_{i-1}^2)\Delta_i + 2\Delta_{i-1}(1 + \Delta_{i-1}^2) \ge 0.$$

Function g takes a minimal value with Δ_i when $g'_{\Delta_i} = 0$. This gives us $\Delta_i^* = (\Delta_{i-1} + \Delta_{i-1}^{-1})/2$ and $g(\Delta_{i-1}, \Delta_i^*) = (6\Delta_{i-1} - \Delta_{i-1}^{-1} + 7\Delta_{i-1}^3)/4 > 0$. Thus, inequality (18) is valid and the cubic weighted spline will preserve the monotonicity of the data. Obviously, this can be done by using the more general formula (17).

Let us consider now an algorithm for selecting the weight parameters to preserve convexity of the data. We will assume that the data are convex, that is, $f[x_{i-1}, x_i, x_{i+1}] \ge 0$, $i = 1, \dots, N$.

Eqs. (6) together with the second derivative boundary conditions can be rewritten in the form

$$\begin{aligned} &M_0 = w_0 f_0'', \\ &\mu_i M_{i-1} + 2M_i + \lambda_i M_{i+1} = d_i, \quad i = 1, \dots, N, \\ &M_{N+1} = w_N f_{N+1}'', \end{aligned} \tag{19}$$

where μ_i and λ_i are given in (9) and

$$d_i = \frac{6 w_{i-1} w_i}{w_i h_{i-1} + w_{i-1} h_i} \delta_i f. \tag{20}$$

Eqs. (19) differ from the corresponding equations for a conventional C^2 cubic spline by their right sides only. This allows us to write the sufficient conditions of convexity for a weighted cubic spline (see [15, p. 10]).

Theorem 4. *Let a cubic weighted spline $S \in C^1[a, b]$, with the second derivative boundary conditions $S''(x_0) = f_0''$ and $S''(x_{N+1}) = f_{N+1}''$, interpolate the convex data $\{f_i\}$, $i = 0, \dots, N + 1$. If the following inequalities are valid:*

$$f_0'' \ge 0, \qquad f_{N+1}'' \ge 0, \qquad 2d_i - \mu_i d_{i-1} - \lambda_i d_{i+1} \ge 0, \quad i = 1, \dots, N, \tag{21}$$

where $d_0 = 2w_0 f_0''$ and $d_{N+1} = 2w_N f_{N+1}''$, then $S''(x) \ge 0$ for all $x \in [a, b]$, that is, S is convex on $[a, b]$.

Let us suppose that $\delta_i f > 0$, $i = 1, \dots, N$. One can strengthen inequalities (21) by splitting them into two parts. We have

$$2d_i - \mu_i d_{i-1} - \lambda_i d_{i+1} = \mu_i(2d_i - d_{i-1}) + \lambda_i(2d_i - d_{i+1}), \quad i = 1, \dots, N.$$

Hence, it follows that the inequalities in (21) are valid provided that the following conditions hold:

$$\begin{aligned} &0 \le w_0 f_0'' \le d_1, \quad 0 \le w_N f_{N+1}'' \le d_N, \\ &d_{i-1}/2 \le d_i \le 2d_{i-1}, \quad i = 2, \dots, N. \end{aligned} \tag{22}$$

Taking into account formula (20) one can readily show that inequalities (22) will be satisfied if the following restrictions on the boundary conditions,

$$0 < f_0'' < 6\delta_1 f / h_0, \qquad 0 < f_{N+1}'' < 6\delta_N f / h_N,$$

and on the weight parameters,

$$\begin{aligned}
&\frac{w_0}{w_1}\frac{h_1}{h_0} \le \frac{6\delta_1 f}{h_0 f_0''} - 1,\\
&\frac{1}{2}\frac{\delta_i f}{\delta_{i-1} f} - 1 \le \frac{w_{i-1}}{w_i}\frac{h_i}{h_{i-1}} \le 2\frac{\delta_i f}{\delta_{i-1} f} - 1, \quad i = 2, \ldots, N,\\
&\frac{w_N}{w_{N-1}}\frac{h_{N-1}}{h_N} \le \frac{6\delta_N f}{h_N f_{N+1}''} - 1,
\end{aligned} \tag{23}$$

are fulfilled. By using these inequalities, we can suggest an algorithm for automatic selection of the shape control parameters w_i to preserve convexity of the data that is just a repetition of the one described above for the case of monotonicity (except that in case of negative values of the weight parameters w_i we replace them by $\varepsilon > 0$). However, one can easily see that if $\delta_{i-1} f \delta_i f < 0$ then inequalities (23) cannot be satisfied for any weights $w_i > 0$. This means that one should consider separately segments of the data convexity and concavity.

5. Construction of basis splines

Let us construct a basis for the space of weighted cubic splines S_4^W by using functions that have local supports of minimum length. Since

$$\dim(S_4^W) = 4(N+1) - 3N = N + 4$$

we extend the grid Δ by adding the points $x_j, j = -3, -2, -1, N+2, N+3, N+4$, such that $x_{-3} < x_{-2} < x_{-1} < a$ and $b < x_{N+2} < x_{N+3} < x_{N+4}$.

We demand that the weighted cubic B-splines B_i, $i = -1, \ldots, N+2$, have the following properties:

$$\begin{aligned}
&B_i(x) > 0, \quad x \in (x_{i-2}, x_{i+2}),\\
&B_i(x) \equiv 0, \quad x \notin (x_{i-2}, x_{i+2}),\\
&\sum_{j=-1}^{N+2} B_j(x) \equiv 1, \quad x \in [a, b].
\end{aligned} \tag{24}$$

The formula for a weighted cubic spline (5) in the interval $[x_i, x_{i+1}]$ can be rewritten in the form

$$S(x) = [S(x_i) - \Phi_i(x_i)M_i](1-t) + [S(x_{i+1}) - \Psi_i(x_{i+1})M_{i+1}]t + \Phi_i(x)M_i + \Psi_i(x)M_{i+1}, \tag{25}$$

where

$$\Phi_i(x) = \frac{(x_{i+1} - x)^3}{6h_i w_i}, \qquad \Psi_i(x) = \frac{(x - x_i)^3}{6h_i w_i}.$$

The representation (25) allows us to consider weighted cubic splines as generalized tension splines [20,15]. With the notation

$$z_j^r \equiv z_j^{(r)}(x_j) = \Psi_{j-1}^{(r)}(x_j) - \Phi_j^{(r)}(x_j), \quad r = 0, 1; \qquad y_j = x_j - \frac{z_j}{z_j'},$$

by using the approach of [15], we obtain the formula

$$B_i(x) = \begin{cases} \Psi_{i-2}(x)M_{i-1,B_i}, & x \in [x_{i-2}, x_{i-1}),\\ \dfrac{x - y_{i-1}}{y_i - y_{i-1}} + \Phi_{i-1}(x)M_{i-1,B_i} + \Psi_{i-1}(x)M_{i,B_i}, & x \in [x_{i-1}, x_i),\\ \dfrac{y_{i+1} - x}{y_{i+1} - y_i} + \Phi_i(x)M_{i,B_i} + \Psi_i(x)M_{i+1,B_i}, & x \in [x_i, x_{i+1}),\\ \Phi_{i+1}(x)M_{i+1,B_i}, & x \in [x_{i+1}, x_{i+2}),\\ 0, & \text{otherwise}, \end{cases} \tag{26}$$

where

$$M_{j,B_i} = \frac{y_{i+1} - y_{i-1}}{z_j' \omega_{i-1}'(y_j)}, \quad j = i-1, i, i+1,$$

$$\omega_{i-1}(x) = (x - y_{i-1})(x - y_i)(x - y_{i+1}).$$

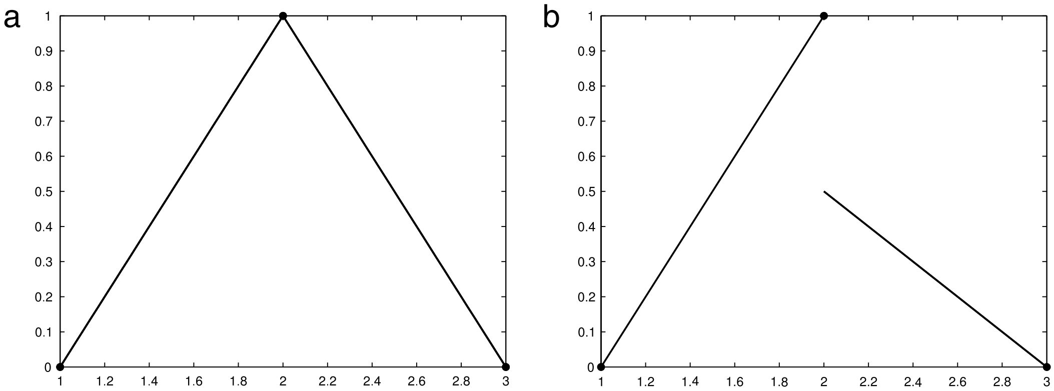

Fig. 1. Normalized weighted linear basis spline B_1 with weights (a) $w_0 = w_1 = 1$ and (b) $w_0 = 1$ and $w_1 = 2$.

One can obtain the formula (26) by using recurrence relations for weighted B-splines. Let us define functions

$$B_{j,1}(x) = \begin{cases} \Psi_j''(x), & x_j \le x < x_{j+1}, \\ \Phi_{j+1}''(x), & x_{j+1} \le x < x_{j+2}, \\ 0, & \text{otherwise}, \end{cases}$$

$j = i-2, i-1, i$. Here Ψ_j'' and Φ_{j+1}'' are linear functions in (x_j, x_{j+2}) and $B_{j,1}(x_{j+1}^-) = 1/w_j$, $B_{j,1}(x_{j+1}^+) = 1/w_{j+1}$, and $B_{j,1}(x_l) = 0$ for $l \neq j+1$. Discontinuous splines $B_{j,1}$ are the generalization of "hut functions" for polynomial B-splines. They are positive on their supports and have the following property:

$$\sum_{j=0}^{N} w_j \big[B_{j-1,1}(x) + B_{j,1}(x) \big] \equiv 1 \quad \text{for all } x \in [a, b].$$

By using $B_{j,1}$, let us define recursively the splines

$$B_{j,k}(x) = \int_{x_j}^{x} \frac{B_{j,k-1}(\tau)}{c_{j,k-1}} d\tau - \int_{x_{j+1}}^{x} \frac{B_{j+1,k-1}(\tau)}{c_{j+1,k-1}} d\tau, \tag{27}$$

$$c_{j,k-1} = \int_{x_j}^{x_{j+k}} B_{j,k-1}(\tau) d\tau, \quad j = i-2, i-k+1, \ k = 2, 3. \tag{28}$$

Simple calculations provide

$$c_{j,1} = z_{j+1}', \quad j = i-2, i-1, i; \qquad c_{j,2} = y_{j+2} - y_{j+1}, \quad j = i-2, i-1,$$

clearing the geometric value of these quantities.

By differentiating (27) we have also

$$B_{j,k}'(x) = B_{j,k-1}(x)/c_{j,k-1} - B_{j+1,k-1}(x)/c_{j+1,k-1}, \quad k = 2, 3. \tag{29}$$

The splines $B_{j,k}$, $k = 2, 3$, can be written in explicit form. By virtue of (27) and (28) for $j = i-2, i-1$ we obtain

$$B_{j,2}(x) = \begin{cases} \Psi_j'(x)/c_{j,1}, & x_j \le x < x_{j+1}, \\ 1 + \Phi_{j+1}'(x)/c_{j,1} - \Psi_{j+1}'(x)/c_{j+1,1}, & x_{j+1} \le x < x_{j+2}, \\ -\Phi_{j+2}'(x)/c_{j+1,1}, & x_{j+2} \le x < x_{j+3}, \\ 0, & \text{otherwise}. \end{cases}$$

The expressions for $B_{j,3}$ differ from B_j in (26) only in the index numeration with respect to the center of the support interval ($B_j(x) \equiv B_{j-2,3}(x)$).

Figs. 1–3 show the graphs of weighted B-splines $B_{j,k}$, $k = 1, 2, 3$ on a uniform mesh with a step $h = 1$ and with different weights. In Figs. 1(a), 2(a), and 3(a), the weights are all equal one, and thus, $B_{j,k}$ are simply $C^{k-1}B$-splines. In Fig. 1(b), the weights are $w_0 = 1$ and $w_1 = 2$. In Fig. 2(b) the weights are $w_0 = 0.1$, $w_1 = 1$, and $w_2 = 10$. In Fig. 3(b), the weights are $w_0 = 0.1$, $w_1 = w_2 = 1$, and $w_3 = 10$. The Bézier points are denoted by dots.

The functions $B_{j,k}$, $k = 1, 2, 3$, have most of the usual polynomial B-spline characteristics. By using the approach of [20], it is easy to show that functions B_j, $j = -1, \ldots, N+2$, have supports of minimal length, are linearly independent, and form a basis in the space S_4^W. So any weighted cubic spline $S \in S_4^W$ can be uniquely represented in the form

$$S(x) = \sum_{j=-1}^{N+2} b_j B_j(x) \quad \text{for } x \in [a, b], \tag{30}$$

with some constant coefficients b_j.

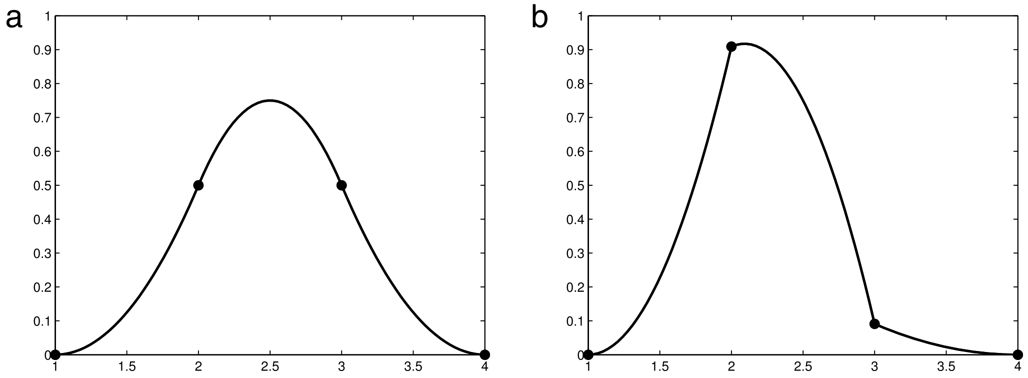

Fig. 2. Normalized weighted quadratic basis spline B_2 with weights (a) $w_i = 1$ for all i and (b) $w_0 = 0.1$, $w_1 = 1$, and $w_2 = 10$.

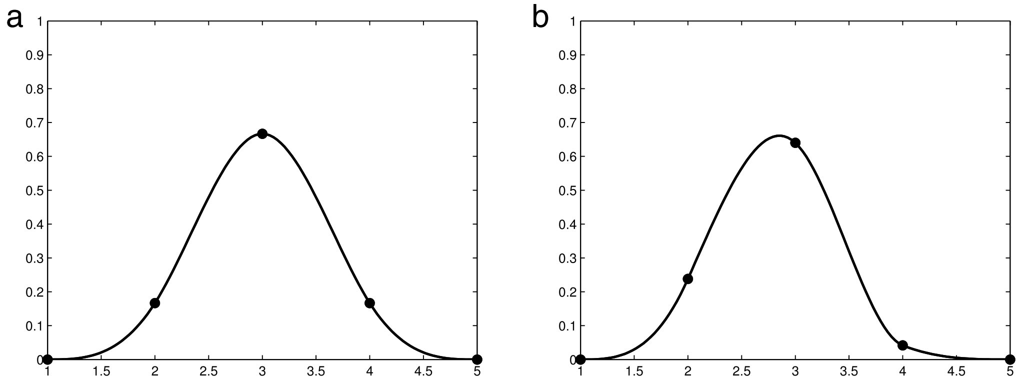

Fig. 3. Normalized weighted cubic basis spline B_3 with weights (a) $w_i = 1$ for all i and (b) $w_0 = 0.1$, $w_1 = w_2 = 1$, $w_3 = 10$.

If coefficients b_j in (30) are known then by virtue of formula (26) we can write the expression for the weighted cubic spline S on the interval $[x_i, x_{i+1}]$, which is convenient for calculations:

$$S(x) = b_i + \tilde{\Delta}_i b(x - y_i) + c_i \Phi_i(x) + c_{i+1} \Psi_i(x), \tag{31}$$

where

$$c_j = \frac{\tilde{\Delta}_j b - \tilde{\Delta}_{j-1} b}{c_{j-1,1}}, \quad j = i, i+1, \qquad \tilde{\Delta}_j b = \frac{b_{j+1} - b_j}{c_{j-1,2}}.$$

Representations (30) and (31) allow us to find a simple and effective way to approximate a given function f from its samples.

By setting

$$b_j = f_j - \frac{1}{c_{j-1,1}} \left[\Psi_{j-1}(x_j) \frac{f_{j+1} - f_j}{h_j} - \Phi_j(x_j) \frac{f_j - f_{j-1}}{h_{j-1}} \right]$$

in (30), we obtain a formula for a three-point local approximation, which is exact for polynomials of the first degree. The proof of this result does not differ from the one given in [15].

It is worth noting that the knot insertion formula in [26] expresses weighted B-splines as a linear combination of polynomial cubic B-splines with positive, explicit coefficients. It is therefore very accurate and computationally inexpensive.

6. Control point approximations

Given control points $P_i = (x_i, y_i)$, $i = 0, \ldots, N+1$, define the parametric curve $S(t) = (S_x(t), S_y(t))$ for $t \in [t_1, t_N]$ by

$$S(t) = \sum_{i=0}^{N+1} P_i B_i(t). \tag{32}$$

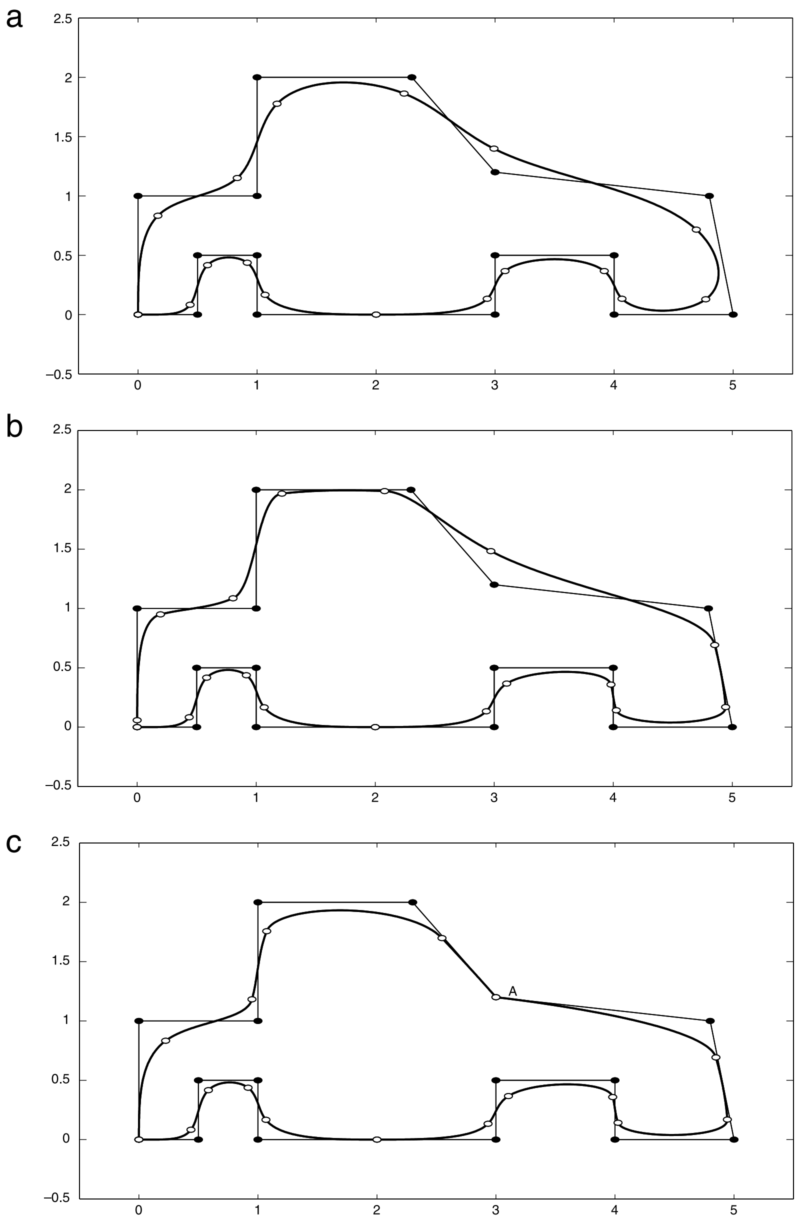

Fig. 4. Weighted spline basis approximations in (32) to the control points denoted by dots by using weights (a) $w_i = 1$ for all i (same as by using B-splines) and (b) $w_8 = 8, w_{10} = w_{13} = 40, w_{15} = 8$, and $w_i = 1$ otherwise; (c) same as (b) except that the control point A is counted twice.

Since $B_i(t) \equiv 0$ outside the support interval $[t_{i-2}, t_{i+2}]$, then for $t \in [t_i, t_{i+1}]$ one has

$$S(t) = P_{i-1}B_{i-1}(t) + P_iB_i(t) + P_{i+1}B_{i+1}(t) + P_{i+2}B_{i+2}(t).$$

The curve in (32) is locally controlled in that if the point P_i is changed, then the only part of the curve that changes is when $t \in (t_{i-2}, t_{i+2})$. If one weight w_i is changed, then only the basis functions $B_j, j = i-1, \ldots, i+2$, are changed, and the control point curve will change only in the parts where $t \in (t_{i-2}, t_{i+3})$.

In our example, we have used Foley's data [10]. Fig. 4(a) shows the parametric curve (32) with all weights equal to 1, and thus it is a B-spline approximation, and Fig. 4(b) is the parametric curve in (32) with the weights $w_8 = 8, w_{10} = w_{13} = 40, w_{15} = 8$, and $w_i = 1$ otherwise. We have used the accumulated chord length parametrization in Fig. 4(a)–(c). The $N = 18$ control points $P_i, i = 1, \ldots, N$, are denoted by dots and are connected by line segments. There is one control point at (2.0, 0.0) that is obscured by a circle symbol, and the first and last points are in the lower left at (0.0, 0.0). The points $P_0 = (0.0, -1.0)$ and $P_{N+1} = (-0.5, 0.0)$, which are the linear extensions of the first and last line segments, are not displayed. The circles represent where one cubic segment of the parametric curve ends and the next begins. More precisely,

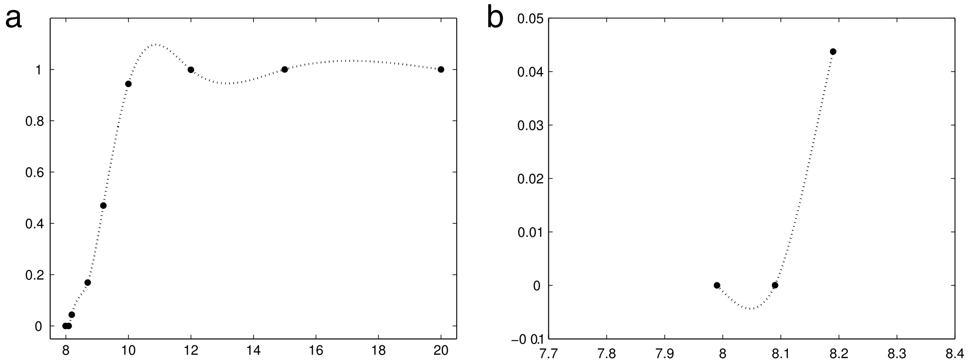

Fig. 5. The radiochemical data with end conditions $M_0 = M_{N+1} = 0$. (a) Interpolation by cubic C^2 spline. (b) A magnification of the lower left corner.

Table 1
Radiochemical data.

x_i	7.99	8.09	8.19	8.7	9.2	10	12	15	20
f_i	0	2.76429E−5	4.37498E−2	0.169183	0.469428	0.943740	0.998636	0.999916	0.999994

they represent points on the curve $S(t)$, where

$$t_i = t_{i-1} + \frac{|P_i - P_{i-1}|}{\sum_{j=1}^{N+1} |P_j - P_{j-1}|}(t_{N+1} - t_0), \quad i = 1, \ldots, N+1,$$

with $|\cdot|$ denoting the usual Euclidean distance. The circle points are also interpolation points, in that the curve $S(t)$ is the parametric weighted spline interpolant to the circle data by using appropriate second derivative end conditions at P_1 and P_N.

Since the functions B_i are non-negative and they sum to one for all $t \in [t_1, t_N]$, the functions $S_x(t)$ and $S_y(t)$ are convex sums and hence the parametric curve $S(t)$ satisfies the local convex hull property as described in [18]. This is experimentally shown in Fig. 4, where the curve defined on $[t_i, t_{i+1}]$ does not exceed the convex hull of the four points $P_j, j = i-1, \ldots, i+2$. We do not prove theoretically that the curve has the minimum number of inflections consistent with the data.

The parametric curve $S(t)$ in (32) has C^1 continuity as long as four consecutive control points P_i are not all equal and the tangent vector is non-zero. Fig. 4(c) exhibits this behavior when there are two control points at A. The weights and control points in Fig. 4(c) are the same as in Fig. 4(b) with the exception that point A is counted twice. The curve comes closer to point A because the convex hull of the neighboring four points is narrower near A because of the repeated point. As with usual B-splines, if three consecutive control points are equal, then the convex hull of the four nearest points is the point itself. In this case, the curve must pass through the control point with C^0 continuity. A common approach in forming the control point curve in (32) is to define the first and last points with a multiplicity of three, and thus the curve will interpolate the first and last points as Bézier curves do. If four consecutive control points are collinear, then the curve in (32) is linear on the middle segment and equal to the line segment joining the two interior control points. This follows from the local convex hull property.

7. Graphical examples

The aim of this section is to illustrate the shape features of weighted cubic interpolation splines with some popular examples. We want to notice that, before, to choose weight parameters we have used our algorithms described in Section 4 and the formula

$$w_i = \left[1 + C_i\big(f[x_i, x_{i+1}]\big)^2\right]^{-\alpha_i}, \quad C_i \geq 1, \ \alpha_i \geq 0, \ i = 0, \ldots, N. \tag{33}$$

In the first example, we have interpolated the radiochemical data reported in Table 1. These data are unequally spaced. The effects of changing the weight values w_i are depicted in Figs. 5 and 6. We have imposed the end conditions $M_0 = 0$ and $M_{N+1} = 0$. Fig. 5 is obtained by setting all weights $w_i = 1$, that is, considering the C^2 cubic spline interpolating the data. In Fig. 6, a new interpolant, obtained by our algorithm described in Section 4, is displayed for the same data, and the stretching effect of the weight parameters is evident. By using formula (33) with different values of α_i and C_i we could not preserve the data monotonicity in the lower left corner.

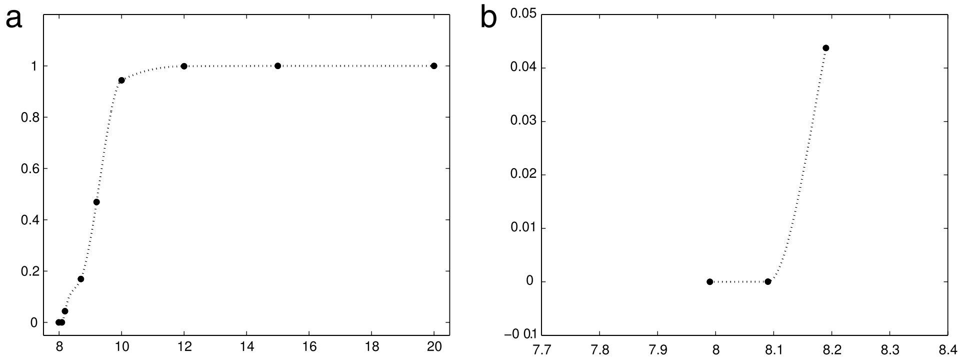

Fig. 6. The same as Fig. 5 with weights by our algorithm described in Section 5.

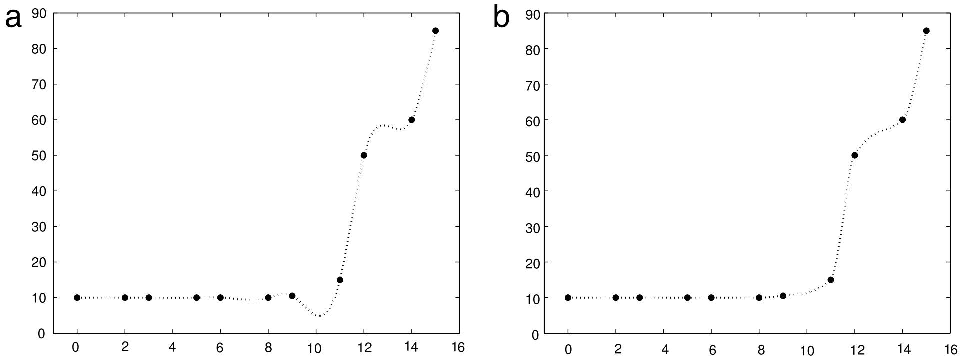

Fig. 7. Akima's data with boundary conditions $M_0 = M_{N+1} = 0$. (a) Interpolating C^2 cubic spline ($w_i = 1$). (b) Weighted cubic spline with shape parameters chosen by our algorithm.

Table 2
Akima's data [28].

x_i	0	2	3	5	6	8	9	11	12	14	15
f_i	10	10	10	10	10	10	10.5	15	50	60	85

In the second example we have taken Akima's data [28] of Table 2 and constructed weighted interpolants with end conditions $M_0 = M_{N+1} = 0$. Fig. 7(a) shows the plot produced by a uniform choice of tension factors, namely $w_i = 1$. Fig. 7(b) shows a second solution, which perfectly reproduces the data shape, where we have set weights by our algorithm.

The data for the next example were taken from [27]. We consider interpolating the function

$$f(x) = 1 - \frac{\exp(100x) - 1}{\exp(100) - 1}, \quad x \in [0, 1],$$

on the uniform mesh $x_i = i/10, i = 0, 1, \dots, 10$. In Fig. 8(a) the graph of the C^2 cubic spline is shown ($w_i = 1$ for all i). The spline gives unacceptable oscillations. Decreasing their amplitude is only possible by either introducing a non-uniform mesh that concentrates the knots in the domain having large gradient or choosing an appropriate parametrization. The weighted spline with shape parameters computed by our algorithm, shown in Fig. 8(b), exhibits the same monotonicity and convexity as f. In the first case the end conditions $S'(x_0) = 0$ and $S'(x_{10}) = -100$ are used. In the second case, we have assigned $M_0 = M_{10} = 0$.

As one more example, we have considered an interpolation of the function $f(x) = 2 - \sqrt{x(2-x)}, 0 \le x \le 2$, that defines a semicircle. This function is interpolated on a mesh uniform in x (Fig. 9(a)) and along an arc length (Fig. 9(b)). In both cases 11 interpolation points and the boundary conditions $S'(x_0) = -50, S'(x_{10}) = 50$ (C^2 cubic spline) and $m_0 = 2f[x_0, x_1], m_{10} = 2f[x_9, x_{10}]$ (weighted C^1 spline) were used. The dotted and solid lines show the graphs of the interpolating cubic C^2 spline and weighted cubic spline with weights by our algorithm. We see that the transition to a mesh with constant steps along the arc length enables us to reduce the oscillations of the C^2 spline but it does not remove them. The weighted C^1 spline again retains the monotonicity property of the initial data.

The algorithm for automatic selection of weight parameters to preserve the convexity of the data was also tested on this example with the convex data. It gives the same results as shown in Fig. 9. We have used the boundary conditions $M_0 = 3w_0\delta_1 f/h_0$ and $M_{N+1} = 3w_N\delta_N f/h_N$.

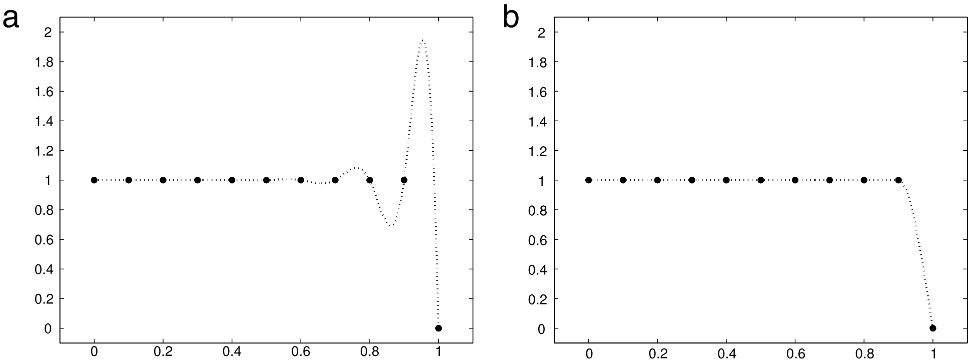

Fig. 8. Exponential boundary-layer-type data [27]. (a) Interpolating C^2 cubic spline ($w_i = 1$). (b) Weighted C^1 cubic spline with weights by our algorithm.

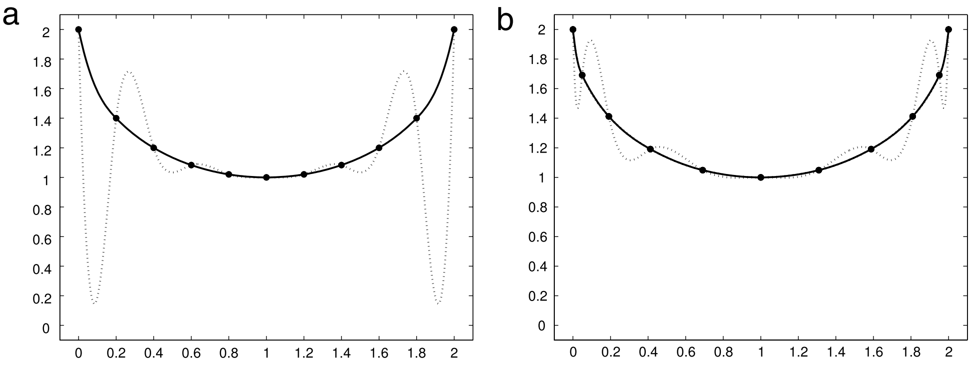

Fig. 9. Interpolation of a semicircle's data on a mesh: (a) uniform in the x coordinate and (b) uniform in the arc length. The dotted and solid lines show the graphs of C^2 cubic ($w_i = 1$) and weighted C^1 cubic (our algorithm) splines.

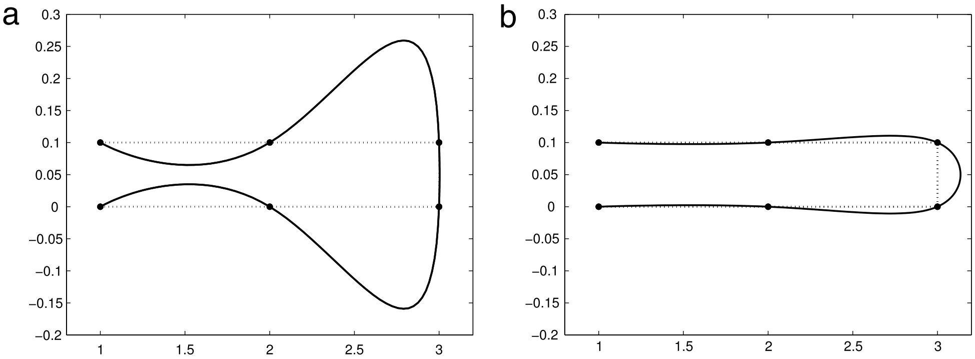

Fig. 10. The parametric C^2 cubic spline (solid curve) and weighted C^1 spline (dashed curve) to the same data: (a) the cord-length parametrization; (b) the unit parametrization.

The purpose of the last example is to observe the effects of the shape parameters w_i when different parametrizations are used. The solid curves in Fig. 10 are the C^2 cubic spline interpolants by using the cord-length (a) and the unit (b) parametrizations with the boundary conditions $(S_x'(t_i), S_y'(t_i)) = (0, 0), i = 1, 6,$ applied to the data $\{x_i\} = \{1, 2, 3, 3, 2, 1\}, \{y_i\} = \{0, 0, 0, 0.1, 0.1, 0.1\}$. The dashed curves in Fig. 10 are the weighted C^1 cubic spline interpolants to the same data and parametrizations where the weights are chosen by our algorithm from Section 4 with $\varepsilon = 0.0001$. Fig. 10 illustrates that the shape of the interpolating curve by using the cord-length parametrization may not be what the user has in mind. However, the weighted spline produces the expected results.

8. Conclusion

Shape preservation is one of the fundamental topics in computer aided geometric design. During the past few decades, different authors have developed various algorithms of shape preserving spline approximation. However the shape parameters are mainly viewed as an interactive design tool for manipulating shape of a spline curve [5]. The main challenge of this paper is to present algorithms that select shape control parameters (weights) automatically. We give two such algorithms: one to preserve the data monotonicity and other to retain the data convexity. These algorithms adapt the spline curve to the data geometric behavior. The main point, however, is to determine whether the error of approximation remains small under the proposed algorithms. To this end we prove two theorems to estimate error bounds. We show that by using special choice of shape parameters one can rise the order of approximation.

There have been important advances in the study of shape preserving representations of curves and surfaces [19]. The curves inherit shape properties of the control polygon when the basis is totally positive and normalized. We construct such basis of weighted cubic B-splines and show experimentally that it has the local convex hull property.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NSF) (Grant Nos. 2012-0005043 and 2012-0000176), funded by the Korean government (MEST). The second author was also supported by the Russian Foundation for Basic Research under Grant No. 09-01-00186. We also wish to express our gratitude to the referees for their invaluable remarks and suggestions for improvements.

References

[1] J.A. Gregory, M. Sarfraz, A rational cubic spline with tension, Comput. Aided Geom. Design 7 (1990) 1–13.
[2] M. Sarfraz, A C^2 rational cubic spline alternative to the NURBS, Comput. Graph. 16 (1) (1992) 69–77.
[3] M. Sarfraz, Interpolatory rational cubic spline with biased, point and interval tension, Comput. Graph. 16 (4) (1992) 427–430.
[4] H. Späth, One Dimensional Spline Interpolation Algorithms, A.K. Peters, Natick, MA, 1995.
[5] T.N.T. Goodman, Shape preserving interpolation by curves, in: J. Levesley, I. Anderson, J. Mason (Eds.), Algorithms for Approximation IV, University of Huddersfield, Huddersfield, UK, 2002, pp. 24–35.
[6] Ph. Cinquin, Splines unidimensionnelles sous tension et bidimensionnelles paramétrées: deux applications médicales, Thése, Université de Saint-Etienne, 28 October 1981.
[7] L. Bos, K. Salkauskas, Limits of weighted splines based on piecewise constant weight functions, Rocky Mountain J. Math. 23 (1993) 483–493.
[8] K. Salkauskas, C^1 splines for interpolation of rapidly varying data, Rocky Mountain J. Math. 14 (1984) 239–250.
[9] K. Salkauskas, L. Bos, Weighted splines as optimal interpolants, Rocky Mountain J. Math. 22 (1992) 705–717.
[10] T.A. Foley, Local control of interval tension using weighted splines, Comput. Aided Geom. Design 3 (1986) 281–294.
[11] T.A. Foley, Interpolation with interval and point tension control using cubic weighted v-splines, ACM Trans. Math. Softw. 13 (1987) 68–96.
[12] T.A. Foley, A shape preserving interpolant with tension controls, Comput. Aided Geom. Design 5 (1988) 105–118.
[13] T. Bosner, M. Rogina, Variable degree polynomial splines are Chebyshev splines, Adv. Comput. Math. (2011) http://dx.doi.org/10.1007/s10444-011-9242-z.
[14] L.L. Schumaker, Spline Functions: Basic Theory, Wiley, New York, 1981.
[15] B.I. Kvasov, Methods of Shape-Preserving Spline Approximation, World Scientific, Singapore, 2000.
[16] B.I. Kvasov, Hyperbolic spline interpolation algorithms, Comput. Math. Math. Phys. 51 (2011) 722–740.
[17] B.I. Kvasov, Parallel mesh methods for tension splines, J. Comput. Appl. Math. 236 (2011) 843–859.
[18] G. Farin, Curves and Surfaces for Computer Aided Geometric Design, Academic Press, San Diego, 2002.
[19] J.M. Peña (Ed.), Shape Preserving Representations in Computer-Aided Geometric Design, Nova Science Publishers, Inc., New York, 1999.
[20] B.I. Kvasov, GB-splines and their properties, Ann. Numer. Math. 3 (1996) 139–149.
[21] T. Bosner, Knot insertion algorithms for weighted splines, in: Z. Drmač, et al. (Eds.), Proceedings of the Conference on Applied Mathematics and Scientific Computing, Springer, 2005, pp. 151–160.
[22] M. Rogina, A knot insertion algorithm for weighted cubic splines, in: A. Le Méhauté, C. Rabut, L.L. Schumaker (Eds.), Curves and Surfaces with Applications in CAGD, Vanderbilt University Press, Nashville, London, 1997, pp. 387–395.
[23] R.P. Kulkarni, P.-J. Laurent, Q-splines, Numer. Algorithms 1 (1991) 45–73.
[24] P. Costantini, B.I. Kvasov, C. Manni, On discrete hyperbolic tension splines, Adv. Comput. Math. 11 (1999) 331–354.
[25] M.-L. Mazure, P.-J. Laurent, Polynomial Chebyshev splines, Comput. Aided Geom. Design 16 (1999) 317–343.
[26] M. Rogina, T. Bosner, On calculating with lower order Chebyshev splines, in: P.-J. Laurent, P. Sablonnière, L.L. Schumaker (Eds.), Curves and Surfaces Design: Saint-Malo 1999, Vanderbilt University Press, Nashville, TN, 2000, pp. 343–352.
[27] Yu.S. Zavyalov, B.I. Kvasov, V.L. Miroshnichenko, Methods of Spline Functions, Nauka, Moscow, 1980 (in Russian).
[28] H. Akima, A new method of interpolation and smooth curve fitting based on local procedures, J. Assoc. Comput. Mach. 17 (1970) 589–602.