Interproximation: interpolation and approximation using cubic spline curves

F Cheng and B A Barsky*

An algorithm for the construction of a cubic spline curve with relatively good shape that interpolates specified data points at some knots and passes through specified regions at some other knots is presented. The curve constructed by the algorithm has minimum energy on each of its components. This algorithm has applications in various fields, such as the reconstruction of natural phenomena where data points cannot be sampled exactly, or computer-aided modeling where some of the fitting points cannot be explicitly specified.

splines, interpolation, uncertain data

The need for curve and surface interpolation arises in computer graphics, image processing, computer vision, pattern recognition, computer-aided design and numerous other fields[1–7]. The task is to construct a curve or surface that interpolates, or passes through, a given set of points such that the shape of the curve or surface can be used to model some desired image or object.

Interpolation using spline curves and surfaces has been studied extensively[8–10]. Efficient algorithms have been proposed for both sequential and parallel environments[1,2,4,10,11]. Several different approaches to the removal of undesirable oscillations of the interpolating curves or surfaces when the data vary rapidly have also been proposed[12–18]. For the situation when interpolation of the given points is not required, Reinsch[19] proposed an algorithm for the construction of a cubic spline curve that approximates the given points with the *energy* of the curve being minimized. However, a different case has not yet been explored: what if a spline curve is required to interpolate several given points, but is only required to pass through some specified regions at some other points (see Figure 1)? How then should such a curve with relatively good shape be constructed?

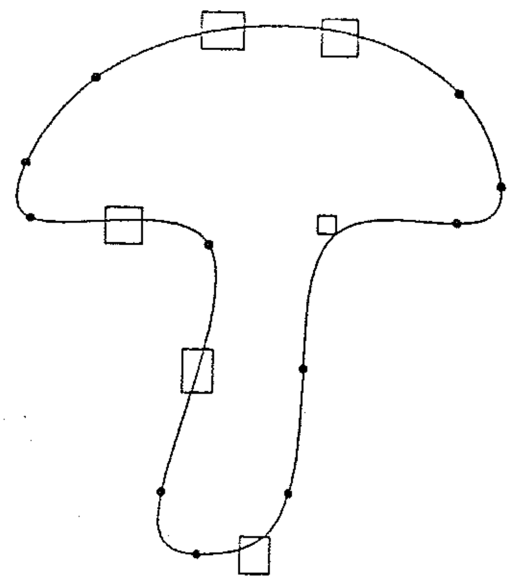

Figure 1. Spline curve that interpolates several given points and passes through some specified regions

This approach has applications in various fields, such as the reconstruction of some natural phenomena where some of the data points cannot be sampled exactly (and only a range of the points are known), or computer-aided design using fitting techniques, where, in the search for an optimal design, a CAD designer may not be sure where some of the fitting points should be (having only a rough idea about the range of the possible positions of these points). The approach finds applications even when all the data points are given explicitly; regions can be specified between consecutive data points to bound the behaviour of the fitting curve so that the curve has a desired shape.

It is certainly possible, in either case, to estimate these uncertain points, and to use interactive techniques to improve gradually their positions, and, hence, the shape of the curve. The problem with this approach is

Department of Computer Science, University of Kentucky, Lexington, KY 40506, USA
*Computer Science Division–EECS Department, University of California, Berkeley, CA 94720, USA
Paper received: 13 March 1990. Revised: 10 January 1991

that it takes too many trial-and-error iterations. It would be better to obtain a satisfactory curve at the outset, or at least a curve that would not require excessive subsequent work.

This paper presents an answer to this question for 2D curves. Specifically, an algorithm is presented for the construction of a cubic spline curve with relatively good shape that interpolates specified points at some *knots*, and passes through specified rectangular regions at some other knots. The cubic spline curve constructed by the algorithm has minimum energy at each of its components. Two examples are given to show how this algorithm can be used to remove undesired oscillations generated on an ordinary cubic spline interpolating function/curve by the specification of a few intervals/regions between interpolation points. These examples demonstrate that, by the use of this approach, a spline function/curve with the desired shape can be constructed in just a few steps.

The approach is based on the observation that this problem can be abstracted as a minimization problem for some quadratic form, and, therefore, techniques for quadratic forms can be used to solve this problem. First, the problem is formally defined.

DEFINITIONS AND PROBLEM FORMULATIONS

Let $\tau = \{u_0, u_1, \ldots, u_n, v_1, \ldots, v_m\}$ be a set of distinct *knots* contained in the parameter interval $[a, b]$ with $u_0 = a$. Let $\mathbf{H}$ be the set of 2D cubic spline curves on $[a, b]$ for the *knot sequence* τ, i.e. $\mathbf{H} = \{\mathbf{S} = (\mathbf{S}_1, \mathbf{S}_2)\}$, where $\mathbf{S}_1$ and $\mathbf{S}_2$ are real-valued cubic spline functions on $[a, b]$ for the *knot sequence* τ.

Problem 1: Given $\mathbf{P}_i = (\mathbf{x}_i, \mathbf{y}_i)$, $i = 0, 1, \ldots, n$, and $\mathbf{A}_j = [a_j, b_j] \times [c_j, d_j]$, $j = 1, 2, \ldots, m$, find $\hat{\mathbf{S}} = (\hat{\mathbf{S}}_1, \hat{\mathbf{S}}_2) \in \mathbf{H}$ such that

$$\begin{aligned} \hat{\mathbf{S}}(u_i) &= \mathbf{P}_i \qquad i = 0, 1, \ldots, n \\ \hat{\mathbf{S}}(v_j) &\in \mathbf{A}_j \qquad j = 1, 2, \ldots, m \end{aligned} \tag{1}$$

and the following condition is satisfied

$$\int_a^b [\hat{\mathbf{S}}^{(2)}(u)]^2\, du = \min\left\{ \int_a^b [\mathbf{S}^{(2)}(u)]^2\, du \,|\, \mathbf{S} \in \mathbf{H}, \; \mathbf{S} \text{ satisfies Equation 1} \right\}$$

where

$$[\mathbf{S}^{(2)}(u)]^2 \equiv \left(\frac{d^2\mathbf{S}_1(u)}{du^2}\right)^2 + \left(\frac{d^2\mathbf{S}_2(u)}{du^2}\right)^2 \qquad \blacksquare$$

That is, among all the 2D cubic spline curves that interpolate $\mathbf{P}_i$ at u_i, $i = 0, 1, \ldots, n$, and pass through $\mathbf{A}_j$ at v_j, $j = 1, 2, \ldots, m$, the one with the smoothest shape (i.e. the one with minimum 'energy'*) is searched for.

Although each 2D parametric cubic spline curve has two components, as energy will be computed on the basis of individual components, and the technique involved for each component is actually the same, it is sufficient to consider this problem for curves with only a single component, i.e. cubic spline functions. Therefore, by the definition of $\mathbf{F}$ as the set of cubic spline functions on $[a, b]$ for the knot sequence τ, Problem 1 can be rewritten as follows.

Problem 1′: Given $\mathbf{x}_i$, $i = 0, 1, \ldots, n$, and $\mathbf{A}_j = [a_j, b_j]$, $j = 1, 2, \ldots, m$, find $\mathbf{f} \in \mathbf{F}$ such that

$$\begin{aligned} \mathbf{f}(u_i) &= \mathbf{x}_i \qquad i = 0, 1, \ldots, n \\ \mathbf{f}(v_j) &\in \mathbf{A}_j \qquad j = 1, 2, \ldots, m \end{aligned} \tag{2}$$

and the following condition is satisfied

$$\int_a^b [\mathbf{f}^{(2)}(u)]^2\, du = \min\left\{ \int_a^b [\mathbf{g}^{(2)}(u)]^2\, du \,|\, \mathbf{g} \in \mathbf{F}, \mathbf{g} \text{ satisfies Equation 2} \right\}$$

where $\mathbf{f}^{(2)}$ is the second derivative of $\mathbf{f}$ with respect to u. $\blacksquare$

Obviously, $\mathbf{F}$ can be decomposed as the direct sum of Π_1 and $\tilde{\mathbf{F}}$

$$\mathbf{F} = \Pi_1 \oplus \tilde{\mathbf{F}}$$

where Π_1 is the set of real polynomials of degree $\leqslant 1$ and

$$\tilde{\mathbf{F}} = \{\mathbf{g} | \mathbf{g} \in \mathbf{F}, \mathbf{g}(a) = \mathbf{g}^{(1)}(a) = 0\}$$

Hence $\mathbf{f}$ (if it exists) can be expressed as

$$\mathbf{f}(u) = \tilde{\mathbf{f}}(u) + \mathbf{x}_0 + \mathbf{x}'_0(u - a)$$

for some $\tilde{\mathbf{f}} \in \tilde{\mathbf{F}}$ such that

$$\begin{aligned} \tilde{\mathbf{f}}(u_i) &= \mathbf{x}_i - \mathbf{x}_0 - \mathbf{x}'_0(u_i - a) \qquad i = 1, 2, \ldots, n \\ \tilde{\mathbf{f}}(v_j) &\in \mathbf{A}_j - \mathbf{x}_0 - \mathbf{x}'_0(v_j - a) \qquad j = 1, 2, \ldots, m \end{aligned} \tag{3}$$

and

$$\int_a^b [\tilde{\mathbf{f}}^{(2)}(t)]^2\, dt = \min\left\{ \int_a^b [\mathbf{g}^{(2)}(t)]^2\, dt \,|\, \mathbf{g} \in \tilde{\mathbf{F}}, \mathbf{g} \text{ satisfies Equation 3} \right\} \tag{4}$$

where $\mathbf{x}'_0$, representing the derivative of $\mathbf{f}$ at u_0, is given.

Now, for any $\mathbf{f}, \mathbf{g} \in \mathbf{F}$, if

$$\langle \mathbf{f}, \mathbf{g} \rangle \equiv \int_a^b \mathbf{f}^{(2)}(u)\mathbf{g}^{(2)}(u)\, du$$

is defined, then

$$\|\mathbf{f}\| \equiv (\langle \mathbf{f}, \mathbf{f} \rangle)^{1/2} \tag{5}$$

defines a norm on $\tilde{\mathbf{F}}$. $\tilde{\mathbf{F}}$ is a *separable Hilbert space*[22] with this norm. Note that

$$K(u, t) \equiv \frac{(t - u)_+^3}{6} - \frac{(t - a)^3}{6} + \frac{(t - a)^2}{2}(u - a)$$

is a *reproducing kernel of* $\tilde{\mathbf{F}}$, i.e.

$$\langle \mathbf{f}, K(\cdot, t) \rangle = \int_a^b \mathbf{f}^{(2)}(u)(t - u)_+^1\, du = \mathbf{f}(t)$$

* The energy definition used in this paper follows that of Kjellander[20]. For more discussion on the definition of energy, see Lee[21].

for any $\mathbf{f} \in \tilde{\mathbf{F}}$. Therefore, if

$$\mathbf{g}_i(u) = K(u, u_i) \qquad i = 1, 2, \ldots, n$$
$$\mathbf{h}_j(u) = K(u, v_j) \qquad j = 1, 2, \ldots, m$$

is set, and

$$\tilde{\mathbf{x}}_i \equiv \mathbf{x}_i - \mathbf{x}_0 - \mathbf{x}_0'(u_i - a) \qquad i = 1, 2, \ldots, n$$
$$\tilde{\mathbf{A}}_j \equiv \mathbf{A}_j - \mathbf{x}_0 - \mathbf{x}_0'(v_j - a) \qquad j = 1, 2, \ldots, m$$
$$\tilde{\mathbf{F}}(\tilde{\mathbf{x}}, \tilde{\mathbf{A}}) \equiv \{\mathbf{f} \in \tilde{\mathbf{F}} \mid \langle \mathbf{f}, \mathbf{g}_i \rangle = \tilde{\mathbf{x}}_i,\ i = 1, 2, \ldots, n,$$
$$\langle \mathbf{f}, \mathbf{h}_j \rangle \in \tilde{\mathbf{A}}_j,\ j = 1, 2, \ldots, m\}$$

is defined, then Problem 1′ is equivalent to the following Problem 1″.

Problem 1″: Find $\tilde{\mathbf{f}} \in \tilde{\mathbf{F}}(\tilde{\mathbf{x}}, \tilde{\mathbf{A}})$ such that

$$\|\tilde{\mathbf{f}}\| = \min\{\|\mathbf{f}\| \mid \mathbf{f} \in \tilde{\mathbf{F}}(\tilde{\mathbf{x}}, \tilde{\mathbf{A}})\} \qquad \blacksquare$$

It is this problem that is solved in this paper in the next section.

SOLUTION

It is first proved that such an $\tilde{\mathbf{f}}$ is a linear combination of $\mathbf{g}_i$ and $\mathbf{h}_j$.

Lemma 1: The solution $\tilde{\mathbf{f}}$ of Problem 1″ satisfies

$$\tilde{\mathbf{f}} = \sum_{i=1}^{n} \alpha_i \mathbf{g}_i + \sum_{j=1}^{m} \beta_j \mathbf{h}_j \tag{6}$$

for some constants α_i, $i = 1, 2, \ldots, n$, and β_j, $j = 1, 2, \ldots, m$.

Proof: The proof is standard, and is presented for completeness only. The existence of $\tilde{\mathbf{f}}$ follows from the fact that $\tilde{\mathbf{F}}(\tilde{\mathbf{x}}, \tilde{\mathbf{A}})$ is a nonempty, closed subset of a separable Hilbert space. For it to be shown that Equation 6 is true, observe that $\tilde{\mathbf{f}}$ can be expressed as

$$\tilde{\mathbf{f}} = \mathbf{h} + \sum_{i=1}^{n} \alpha_i \mathbf{g}_i + \sum_{j=1}^{m} \beta_j \mathbf{h}_j$$

where α_i, $i = 1, 2, \ldots, n$, and β_j, $j = 1, 2, \ldots, m$, are constants, and $\mathbf{h}$ is an element in $\tilde{\mathbf{F}}$ that is orthogonal to the *linear span* of $\{\mathbf{g}_1, \ldots, \mathbf{g}_n, \mathbf{h}_1, \ldots, \mathbf{h}_m\}$. As

$$\tilde{\mathbf{f}} - \mathbf{h} \in \tilde{\mathbf{F}}(\tilde{\mathbf{x}}, \tilde{\mathbf{A}})$$

and

$$\langle \tilde{\mathbf{f}}, \tilde{\mathbf{f}} \rangle = \langle \tilde{\mathbf{f}} - \mathbf{h}, \tilde{\mathbf{f}} - \mathbf{h} \rangle + \langle \mathbf{h}, \mathbf{h} \rangle$$

it follows that $\mathbf{h} = 0$. Therefore, Equation 6 is true. ■

The following lemma shows that, to find $\tilde{\mathbf{f}}$, it is sufficient to find β_j, $j = 1, 2, \ldots, m$, in Equation 6 only.

Lemma 2: $\boldsymbol{\alpha}^t = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ depends on $\boldsymbol{\beta}^t = (\beta_1, \beta_2, \ldots, \beta_m)$ only. Specifically,

$$\boldsymbol{\alpha} = \mathbf{Z}^{-1}(\tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta}) \tag{7}$$

where

$$\mathbf{Z} = (\mathbf{z}_{k,i})_{n \times n} = (\langle \mathbf{g}_i, \mathbf{g}_k \rangle)_{n \times n}$$
$$\mathbf{W} = (\mathbf{w}_{k,j})_{n \times m} = (\langle \mathbf{h}_j, \mathbf{g}_k \rangle)_{n \times m} \tag{8}$$
$$\tilde{\mathbf{x}}^t = (\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \ldots, \tilde{\mathbf{x}}_n)$$

Proof: As $\langle \tilde{\mathbf{f}}, \mathbf{g}_k \rangle = \tilde{\mathbf{x}}_k$, for $k = 1, 2, \ldots, n$, it follows that

$$\sum_{i=1}^{n} \langle \mathbf{g}_i, \mathbf{g}_k \rangle \alpha_i + \sum_{j=1}^{m} \langle \mathbf{h}_j, \mathbf{g}_k \rangle \beta_j = \tilde{\mathbf{x}}_k \qquad k = 1, 2, \ldots, n$$

or

$$\mathbf{Z}\boldsymbol{\alpha} = \tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta}$$

where $\mathbf{Z}$, $\mathbf{W}$ and $\tilde{\mathbf{x}}$ are defined in Equation 8. Obviously, as it is a Gram matrix, $\mathbf{Z}$ is symmetric and positive-definite[23]. Therefore, Equation 7 is true. ■

To find $\boldsymbol{\beta}$, observe that

$$\langle \tilde{\mathbf{f}}, \mathbf{h}_l \rangle \in \tilde{\mathbf{A}}_l \qquad l = 1, 2, \ldots, m$$

Therefore,

$$\sum_{i=1}^{n} \langle \mathbf{g}_i, \mathbf{h}_l \rangle \alpha_i + \sum_{j=1}^{m} \langle \mathbf{h}_j, \mathbf{h}_l \rangle \beta_j \in \tilde{\mathbf{A}}_l \qquad l = 1, 2, \ldots, m$$

that is

$$\mathbf{W}^t\boldsymbol{\alpha} + \mathbf{N}\boldsymbol{\beta} \in \tilde{\mathbf{A}} \tag{9}$$

where

$$\mathbf{N} = (\mathbf{n}_{l,j})_{m \times m} = (\langle \mathbf{h}_j, \mathbf{h}_l \rangle)_{m \times m}$$
$$\tilde{\mathbf{A}} = \mathop{\times}_{l=1}^{m} \tilde{\mathbf{A}}_l$$

By the use of Equation 7,

$$\begin{aligned}\mathbf{W}^t\boldsymbol{\alpha} + \mathbf{N}\boldsymbol{\beta} &= \mathbf{W}^t\mathbf{Z}^{-1}(\tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta}) + \mathbf{N}\boldsymbol{\beta} \\ &= \mathbf{W}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} - \mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W}\boldsymbol{\beta} + \mathbf{N}\boldsymbol{\beta} \\ &= (\mathbf{N} - \mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W})\boldsymbol{\beta} + \mathbf{W}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}}\end{aligned} \tag{10}$$

Hence, from Equations 9 and 10,

$$(\mathbf{N} - \mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W})\boldsymbol{\beta} \in \tilde{\mathbf{A}} - \mathbf{W}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} \tag{11}$$

Now it can be shown that minimizing $\|\tilde{\mathbf{f}}\|$ is equivalent to minimizing a *quadratic form* subject to some condition.

Theorem 1: Minimizing $\|\tilde{\mathbf{f}}\|$ is equivalent to minimizing $\boldsymbol{\beta}^t\mathbf{V}\boldsymbol{\beta}$ subject to the condition that

$$\mathbf{V}\boldsymbol{\beta} \in \tilde{\mathbf{A}} - \mathbf{W}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} \tag{12}$$

where

$$\mathbf{V} = \mathbf{N} - \mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W} \tag{13}$$

Proof: By the use of Lemmas 1 and 2,

$$\begin{aligned}\|\tilde{\mathbf{f}}\|^2 &= \boldsymbol{\alpha}^t\mathbf{Z}\boldsymbol{\alpha} + \boldsymbol{\beta}^t\mathbf{N}\boldsymbol{\beta} + 2\boldsymbol{\alpha}^t\mathbf{W}\boldsymbol{\beta} \\ &= (\tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta})^t\mathbf{Z}^{-1}\mathbf{Z}\mathbf{Z}^{-1}(\tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta}) + \boldsymbol{\beta}^t\mathbf{N}\boldsymbol{\beta} \\ &\quad + 2(\tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta})^t\mathbf{Z}^{-1}\mathbf{W}\boldsymbol{\beta} \\ &= (\tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta})^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} + \boldsymbol{\beta}^t\mathbf{N}\boldsymbol{\beta} \\ &\quad + (\tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta})^t\mathbf{Z}^{-1}\mathbf{W}\boldsymbol{\beta} \\ &= \tilde{\mathbf{x}}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} - \boldsymbol{\beta}^t\mathbf{W}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} + \boldsymbol{\beta}^t\mathbf{N}\boldsymbol{\beta} + \tilde{\mathbf{x}}\mathbf{Z}^{-1}\mathbf{W}\boldsymbol{\beta} \\ &\quad - \boldsymbol{\beta}^t\mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W}\boldsymbol{\beta}\end{aligned}$$

As $\boldsymbol{\beta}^t\mathbf{W}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} = \tilde{\mathbf{x}}^t\mathbf{Z}^{-1}\mathbf{W}\boldsymbol{\beta}$, it follows that

$$\|\tilde{\mathbf{f}}\|^2 = \tilde{\mathbf{x}}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} + \boldsymbol{\beta}^t(\mathbf{N} - \mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W})\boldsymbol{\beta}$$
$$= \tilde{\mathbf{x}}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} + \boldsymbol{\beta}^t\mathbf{V}\boldsymbol{\beta}$$

where **V** is defined in Equation 13. The theorem then follows from the observation that $\tilde{\mathbf{x}}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}}$ is a constant, and the fact that $\boldsymbol{\beta}$ must satisfy Equation 11. ■

To minimize $\boldsymbol{\beta}^t\mathbf{V}\boldsymbol{\beta}$ subject to the condition in Equation 12, first note that the matrix **V** is symmetric and positive-definite. This follows from the observation that **V** is a Gram matrix.

Theorem 2: The matrix **V** defined in Equation 13 is the Gram matrix of $\tilde{\mathbf{h}}_j$, $j = 1, 2, \ldots, m$, where

$$\tilde{\mathbf{h}}_j = (\mathbf{I} - \mathbf{P_g})\mathbf{h}_j \qquad j = 1, 2, \ldots, m$$

and $\mathbf{P_g}$ is the *orthogonal projection* on the linear span of $\{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_n\}$.

Proof: It is well known[22] that

$$\mathbf{P_g}\mathbf{f} = \sum_{l=1}^{n} \langle \mathbf{f}, \mathbf{g}_l \rangle \sum_{k=1}^{n} \mathbf{Z}_{l,k}^{-1}\mathbf{g}_k$$

where

$$(\mathbf{Z}_{l,k}^{-1}) = \mathbf{Z}^{-1} = (\langle \mathbf{g}_l, \mathbf{g}_k \rangle)^{-1}$$

Hence,

$$\langle \mathbf{P_g}\mathbf{h}_i, \mathbf{h}_j \rangle = (\mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W})_{i,j}$$

Note that, as

$$\langle \mathbf{P_g}\mathbf{f}, \mathbf{h} \rangle = \langle \mathbf{f}, \mathbf{P_g}\mathbf{h} \rangle \qquad \mathbf{f}, \mathbf{h} \in \mathbf{F}$$

and

$$\mathbf{P_g}\mathbf{P_g} = \mathbf{P_g}$$

it follows that

$$\langle \tilde{\mathbf{h}}_i, \tilde{\mathbf{h}}_j \rangle = \langle \mathbf{h}_i, \mathbf{h}_j \rangle - 2\langle \mathbf{P_g}\mathbf{h}_i, \mathbf{h}_j \rangle + \langle \mathbf{P_g}\mathbf{h}_i, \mathbf{P_g}\mathbf{h}_j \rangle$$
$$= \langle \mathbf{h}_i, \mathbf{h}_j \rangle - \langle \mathbf{P_g}\mathbf{h}_i, \mathbf{h}_j \rangle$$

Therefore,

$$(\langle \tilde{\mathbf{h}}_i, \tilde{\mathbf{h}}_j \rangle) = (\langle \mathbf{h}_i, \mathbf{h}_j \rangle) - \mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W}$$
$$= \mathbf{N} - \mathbf{W}^t\mathbf{Z}^{-1}\mathbf{W} = \mathbf{V}$$

■

Hence $\boldsymbol{\beta}$ can be replaced by $\mathbf{V}^{-1}\boldsymbol{\gamma}$ in $\boldsymbol{\beta}^t\mathbf{V}\boldsymbol{\beta}$ and Equation 12, and it can be found that minimizing $\boldsymbol{\beta}^t\mathbf{V}\boldsymbol{\beta}$ subject to the condition in Equation 12 is equivalent to minimizing $\boldsymbol{\gamma}^t\mathbf{V}^{-1}\boldsymbol{\gamma}$ subject to the condition

$$\boldsymbol{\gamma} \in \hat{\mathbf{A}} \equiv \tilde{\mathbf{A}} - \mathbf{W}^t\mathbf{Z}^{-1}\tilde{\mathbf{x}} \qquad (14)$$

The minimum always exists, because $\mathbf{f}(\boldsymbol{\gamma}) \equiv \boldsymbol{\gamma}^t\mathbf{V}^{-1}\boldsymbol{\gamma}$ is a continuous function on the compact subset $\hat{\mathbf{A}}$ of the m-dimensional Euclidean space. Actually, as $\mathbf{V}^{-1}$ is symmetric and positive-definite, the minimum is unique.

Therefore, the solution of Problem 1″ can be found in three steps:

- Find $\boldsymbol{\gamma} \in \hat{\mathbf{A}}$ such that $\boldsymbol{\gamma}^t\mathbf{V}^{-1}\boldsymbol{\gamma}$ is minimal.
- Compute $\boldsymbol{\beta} = \mathbf{V}^{-1}\boldsymbol{\gamma}$.
- Compute $\boldsymbol{\alpha} = \mathbf{Z}^{-1}(\tilde{\mathbf{x}} - \mathbf{W}\boldsymbol{\beta})$.

Step 1 above is the well known *quadratic programming problem* in nonlinear programming; standard techniques can be used to solve this problem (see, for example, Reference 24, pp 437–441). Step 2 and Step 3 are straightforward. The matrices **Z**, **N** and **W** are constructed using the following formulas.

$$\langle \mathbf{g}_i, \mathbf{g}_j \rangle = \begin{cases} \frac{1}{6}(u_i - a)^2(3(u_j - a) - (u_i - a)) & u_i < u_j \\ \frac{1}{3}(u_i - a)^3 & u_i = u_j \end{cases}$$

$$\langle \mathbf{h}_i, \mathbf{h}_j \rangle = \begin{cases} \frac{1}{6}(v_i - a)^2(3(v_j - a) - (v_i - a)) & v_i < v_j \\ \frac{1}{3}(v_i - a)^3 & v_i = v_j \end{cases}$$

$$\langle \mathbf{g}_i, \mathbf{h}_j \rangle = \begin{cases} \frac{1}{6}(u_i - a)^2(3(v_j - a) - (u_i - a)) & u_i < v_j \\ \frac{1}{6}(v_j - a)^2(3(u_i - a) - (v_j - a)) & u_i > v_j \end{cases}$$

IMPLEMENTATION

The algorithm has been implemented in PASCAL for both cubic spline functions and 2D cubic spline curves. Some examples are shown in Figures 2 and 3. In Figure 2a, the profile of a girl taken from Reference 25, p 164, is shown. A nonuniform cubic spline function that interpolates 20 points taken from the profile is shown in Figure 2b. As some of the fitting points vary rapidly, there is an undesirable oscillation between points b and d. By the specification of three intervals [6.0, 7.0], [5.0, 5.5] and [5.0, 5.2] between a and b, b and c, and c and d, respectively, and the construction of a function that interpolates the given points and passes through the three specified intervals, a curve close to the given profile is obtained (see Figure 2c). The shape of the interpolating curve can be further improved by reducing the intervals to smaller intervals, or shifting the intervals to more appropriate locations (see Figure 2d). Note that, if an interval is reduced to a single point, it simply becomes an interpolation point of the curve (see Figure 2d).

Examples for 2D cubic spline curves are shown in Figure 3. A uniform 2D cubic spline curve interpolating 23 points taken from the profile shown in Figure 3a is shown in Figure 3b. Several severe oscillations occur. By the specification of four rectangular regions in the related areas, a much better curve is obtained (see Figure 3c). The curve is then further improved by the reduction of the intervals to either smaller intervals, or single points (see Figure 3d). Note that, for an interval to be reduced to a point, a simple procedure must be written so that the value of the curve at a particular knot can be displayed on the screen.

CONCLUSIONS

The problem of fitting uncertain data using cubic spline curves has been studied. An algorithm is presented for the construction of a cubic spline curve that interpolates specified data points at some knots, and passes through specified rectangular regions at some other knots, with minimum energy at each of its components. This approach allows the user to construct a fitting curve with satisfactory shape when some of the fitting points are uncertain, without the need of many trial-and-error iterations.

varying data' *Rocky Mountain J. Math.* Vol 14 No 1 (1984) pp 239–250

18 **Schweikert, D G** 'An interpolation curve using a spline in tension' *J. Math. & Physics* Vol 45 (1966) pp 312–317

19 **Reinsch, C H** 'Smoothing by spline functions' *Numer. Math.* Vol 10 No 3 (1967) pp 177–183

20 **Kjellander, J A P** 'Smoothing of cubic parametric splines' *Comput.-Aided Des.* Vol 15 No 5 (1983) pp 175–179

21 **Lee, E T Y** 'Energy, fairness and a counterexample' *Comput.-Aided Des.* Vol 22 No 1 (1990) pp 37–40

22 **Strang, G** *Linear Algebra and Its Applications* Academic Press, USA (1980)

23 **Natanson, I P** *Constructive Function Theory, Vol II – Approximation in Mean* Frederick Ungar, USA (1965)

24 **Avriel, M** *Nonlinear Programming: Analysis and Methods* Prentice-Hall, USA (1976)

25 **Rice, J R** *The Approximation of Functions. Vol II – Nonlinear and Multivariate Theory* Addison–Wesley, USA (1969)

26 **Ritter, K** 'Generalized spline interpolation and nonlinear programming' *in Approximation with Special Emphasis on Spline Functions* Academic Press, UK (1969) pp 75–118

BIBLIOGRAPHY

Yosida, K *Functional Analysis* Springer-Verlag, Germany (1965)

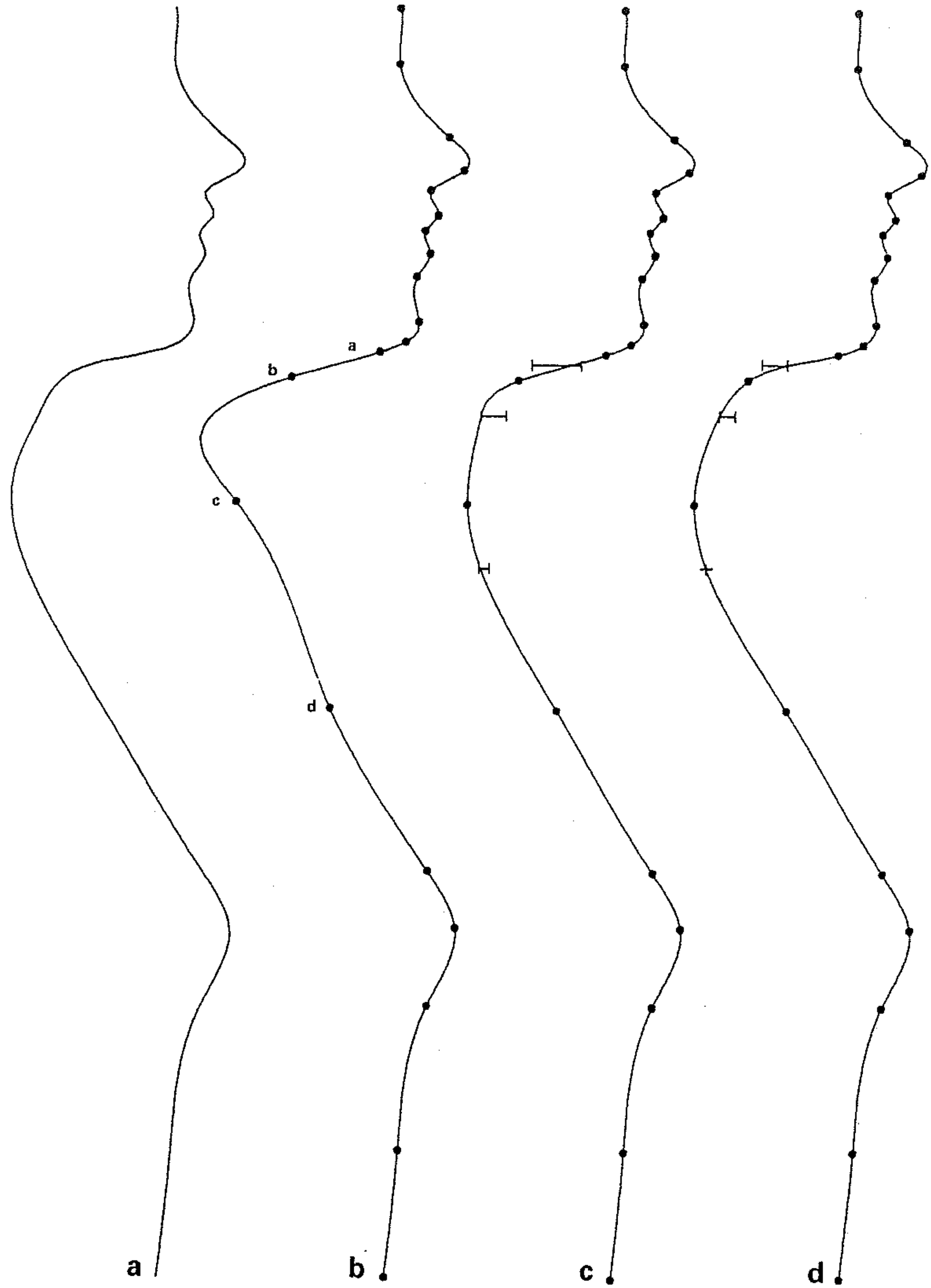

Figure 2. Examples of fitting using cubic spline functions

ACKNOWLEDGEMENTS

The authors would like to thank Professor Gregory M Nielson for pointing out the work of Reinsch to them.

After completion of the work, the authors learned that similar work had been previously carried out[26].

F Cheng was partially supported by a grant from the Center for Robotics and Manufacturing Systems, University of Kentucky, USA. B A Barsky was supported in part by US NSF Presidential Young Investigator Award CCR-84-51997.

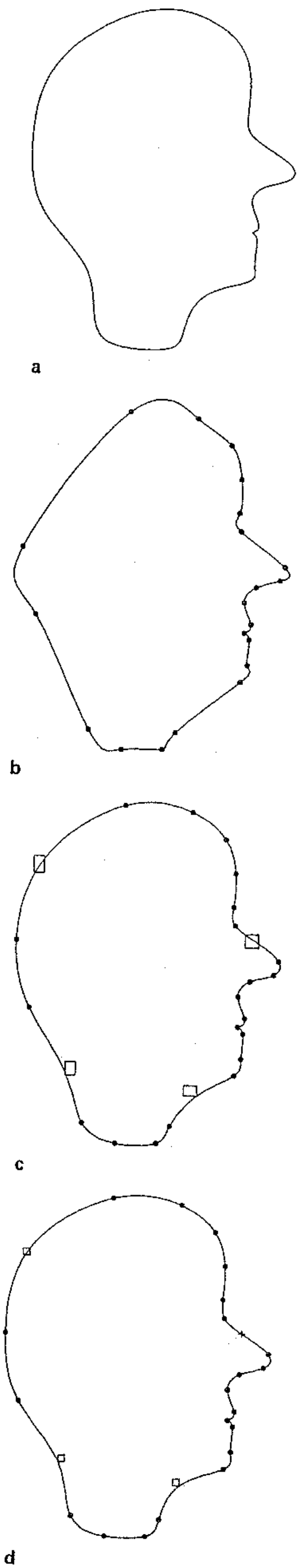

Figure 3. Examples of fitting using 2D cubic spline curves

REFERENCES

1 **Barsky, B A and Greenberg, D P** 'Determining a set of B-spline control vertices to generate an interpolating surface' *Comput. Graph. & Image Proc.* Vol 14 No 3 (1980) pp 203–226

2 **Barsky, B A and Greenberg, D P** 'An interactive surface representation system using a B-spline formulation with interpolation capability' *Comput.-Aided Des.* Vol 14 No 4 (1982) pp 187–194 (and corrigendum: Vol 15 No 3 (1983) p 174)

3 **Faux, I D and Pratt, M J** *Computational Geometry for Design and Manufacture* John Wiley, USA (1979)

4 **Goshtasby, A, Cheng, F and Barsky, B A** 'B-spline curves and surfaces viewed as digital filters' *Comput. Vision Graph. & Image Proc.* Vol 52 No 2 (1990) pp 264–275

5 **Knuth, D E** *TEX and METAFONT: New Directions in Typesetting* Digital Press and American Mathematical Society, USA (1979)

6 **Pavlidis, T** 'Curve fitting as a pattern recognition problem' *Proc. 6th Int. Conf. Pattern Recognition* IEEE Computer Society Press, USA (1982) pp 853–859

7 **Schmitt, F J M, Barsky, B A and Du, W-H** 'An adaptive method for surface-fitting from sampled data' *Siggraph '86 Conf. Proc.* Vol 20 No 4 ACM, USA (18–22 Aug 1986) pp 179–188

8 **Bartels, R H, Beatty, J C and Barsky, B A** *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling* Morgan Kaufmann, USA (1987)

9 **Boehm, W, Farin, G and Kahmann, J** 'A survey of curve and surface methods in CAGD' *Comput. Aided Geom. Des.* Vol 1 No 1 (1984) pp 1–60

10 **de Boor, C** *A Practical Guide to Splines* Springer-Verlag, USA (1978)

11 **Cheng, F and Goshtasby, A** 'A parallel B-spline surface fitting algorithm' *ACM Trans. Graph.* Vol 8 No 1 (1989) pp 41–50

12 **Barsky, B A** 'Exponential and polynomial methods for applying tension to an interpolating spline curve' *Comput. Vision Graph. & Image Proc.* Vol 27 No 1 (1984) pp 1–18

13 **Cline, A K** 'Scalar- and planar-valued curve fitting using splines under tension' *Commun. ACM* Vol 17 No 4 (1974) pp 218–220

14 **Foley, T A** 'Weighted bicubic spline interpolation to rapidly varying data' *ACM Trans. Graph.* Vol 6 No 1 (1987) pp 1–18

15 **Nielson, G M** *Computer-Aided Geometric Design* Academic Press, USA (1974) pp 209–235

16 **Nielson, G M** 'Rectangular ν-splines' *IEEE Comput. Graph. & Applic.* Vol 6 No 2 (1986) pp 58–67

17 **Salkauskas, K** 'C^1 splines for interpolation of rapidly

varying data' *Rocky Mountain J. Math.* Vol 14 No 1 (1984) pp 239–250

18 **Schweikert, D G** 'An interpolation curve using a spline in tension' *J. Math. & Physics* Vol 45 (1966) pp 312–317

19 **Reinsch, C H** 'Smoothing by spline functions' *Numer. Math.* Vol 10 No 3 (1967) pp 177–183

20 **Kjellander, J A P** 'Smoothing of cubic parametric splines' *Comput.-Aided Des.* Vol 15 No 5 (1983) pp 175–179

21 **Lee, E T Y** 'Energy, fairness and a counterexample' *Comput.-Aided Des.* Vol 22 No 1 (1990) pp 37–40

22 **Strang, G** *Linear Algebra and Its Applications* Academic Press, USA (1980)

23 **Natanson, I P** *Constructive Function Theory, Vol II – Approximation in Mean* Frederick Ungar, USA (1965)

24 **Avriel, M** *Nonlinear Programming: Analysis and Methods* Prentice-Hall, USA (1976)

25 **Rice, J R** *The Approximation of Functions. Vol II – Nonlinear and Multivariate Theory* Addison–Wesley, USA (1969)

26 **Ritter, K** 'Generalized spline interpolation and nonlinear programming' *in Approximation with Special Emphasis on Spline Functions* Academic Press, UK (1969) pp 75–118

BIBLIOGRAPHY

Yosida, K *Functional Analysis* Springer-Verlag, Germany (1965)