# Sectional Curvature–Preserving Interpolation of Contour Lines

Bert Jüttler

**Abstract.** A sequence of given contour curves is interpolated by a surface composed of tensor–product B–spline patches. The interpolation scheme preserves the signs of the sectional curvature of the contours. Based on an appropriate linearization of the shape constraints we formulate this task as a quadratic programming problem which is solved with the help of an active set strategy.

## §1. Introduction

Methods for the generation of surfaces from given contour line data are required in several applications, ranging from the reconstruction of bone surfaces from medical images to the construction of ship hulls. For an overview over related literature the reader is referred to the survey articles by Schumaker [7] and Unsworth [8]. It is desirable that the generated surfaces preserve (at least approximately) the shape of the given contour data. In order to achieve this property, Kaklis and Ginnis [6] proposed to use interpolation by polynomial splines of non–uniform degree whereby the degrees of the spline segments act as tension parameters. If the degree of the spline segments is chosen high enough, then the surface preserves the shape of the contours. In the present paper we will outline a different approach; the shape–preserving property is guaranteed with the help of additional linear constraints to the control points.

We assume that B–spline representations of some contour curves of the surface are already known. They can be found with the help of a method for shape preserving least–square approximation by polynomial parametric spline curves which has been developed in [4]. We interpolate these curves by a surface which preserves their shape, i.e., all segments of level curves interpolating between two convex segments of contour lines are convex (sectional curvature–preserving interpolation). Based on an appropriate linearization of the shape constraints we are able to formulate this task as a quadratic programming (QP) problem. We then construct an initial solution which is very close to the optimum and solve the QP problem using an active set strategy. The method is illustrated by an example.

Curves and Surfaces with Applications in CAGD
A. Le Méhauté, C. Rabut, and L. L. Schumaker (eds.), pp. 203–210.


ISBN 0-8265-1293-3.

## §2. Sectional Curvature–Preserving Interpolation

The $C+1$ given contour curves, represented by open B–spline curves (see [3])

$$\mathbf{x}_i(t) = \sum_{j=0}^{D_i} \mathbf{d}_{i,j} N_{i,j}^d(t) \quad t \in [0,1], \quad i = 0, ..., C \tag{1}$$

of degree $d$, with the associated *heights* $(z_i)_{i=0,\ldots,C}$ (i.e., $x_{i,3}(t) \equiv z_i$), are to be interpolated by a $C^l$ $(l = 1, 2)$ surface $\mathbf{y}(z,t)$ with the parameter domain $(z,t) \in [z_0, z_C] \times [0,1]$. The contour curves are defined over possibly different knot sequences $\mathcal{T}_i$ with $d+1$–fold boundary knots 0 and 1 whereby all inner knots have multiplicity $d-l$. We assume that points with the same parameter $t$ on adjacent contours (1) correspond to each other. The third coordinate function of the interpolating surface will simply be equal to the $z$–coordinate, $y_3(z,t) \equiv z$. In addition to the interpolation of the given contours, $\mathbf{y}(z_i,t) \equiv \mathbf{x}_i(t)$ for $i = 0, ..., C$, the surface $\mathbf{y}(z,t)$ is to preserve the sectional curvature of the given contours. This notion has been introduced by Kaklis and Ginnis [6]:

**Definition 1.** *The surface $\mathbf{y}(z,t)$ is said to be a sectional curvature–preserving (sc–p) interpolant if it possesses the following property for any pair $\mathbf{x}_{i-1}(t)$, $\mathbf{x}_i(t)$ of adjacent contour curves ($i = 1, ..., C$): if both contours possess non–positive (resp. non–negative) curvatures at a point $t = t_0$, then also the curvatures of all interpolating contours $\mathbf{y}(z_0,t)$ with $z_0 \in [z_{i-1}, z_i]$, constant, are non–positive (resp. non–negative) at this point.*

The interpolating surface $\mathbf{y}(z,t)$ is an sc–p interpolant if the two inequalities $[\dot{\mathbf{x}}_{i-1}(t_0), \ddot{\mathbf{x}}_{i-1}(t_0)] \geq 0$ (resp. $\leq 0$) and $[\dot{\mathbf{x}}_i(t_0), \ddot{\mathbf{x}}_i(t_0)] \geq 0$ (resp. $\leq 0$) imply $[\dot{\mathbf{y}}(z,t_0), \ddot{\mathbf{y}}(z,t_0)] \geq 0$ (resp. $\leq 0$) for all $(z,t) \in [z_{i-1}, z_i] \times [0,1]$ with $1 \leq i \leq C$, whereby "$\cdot$" denotes the differentiation $\frac{\partial}{\partial t}$ with respect to $t$. The abbreviation $[\vec{\mathbf{p}}, \vec{\mathbf{q}}] = p_1 \cdot q_2 - p_2 \cdot q_1$ means the third component of the cross product of the vectors $\vec{\mathbf{p}}, \vec{\mathbf{q}} \in \mathbb{R}^3$. Moreover we denote by $\sphericalangle(\vec{\mathbf{p}}, \vec{\mathbf{q}}) \in (-\pi, \pi]$ the oriented angle between the top views (orthogonal projections onto the plane $z = 0$) of the two vectors $\vec{\mathbf{p}}, \vec{\mathbf{q}}$, whereas $\|\vec{\mathbf{p}}\| = \sqrt{(p_1^2 + p_2^2)}$ is the length of the top view of the vector $\vec{\mathbf{p}}$.

The construction of the interpolating surface is based on the following two additional assumptions.

($i$) The contour curves and their control polygons have coinciding shape. More precisely, if $\mathbf{x}_i(t)$ has non–negative (resp. non–positive) curvature for some of its polynomial spline segments, then the angles between adjacent legs of the corresponding control polygon $(\mathbf{d}_{i,j})_{\ j=p,\ldots,q}$ are also non–negative (resp. non–positive) but less than $\frac{\pi}{d-1}$ (greater then $-\frac{\pi}{d-1}$),

$$\begin{array}{c} 0 \leq \sphericalangle(\Delta_{[2]}\mathbf{d}_{i,j}, \Delta_{[2]}\mathbf{d}_{i,j+1}) < \frac{\pi}{d-1} \\ (\text{ resp. } \ -\frac{\pi}{d-1} < \sphericalangle(\Delta_{[2]}\mathbf{d}_{i,j}, \Delta_{[2]}\mathbf{d}_{i,j+1}) \leq 0 \ ) \end{array} \tag{2}$$

for $j{=}p, ..., q$ with $\Delta_{[2]}\mathbf{d}_{i,j} = \mathbf{d}_{i,j+1} - \mathbf{d}_{i,j}$. (Note that the lower index of the difference operator always refers to the number of the index where it applies to.)

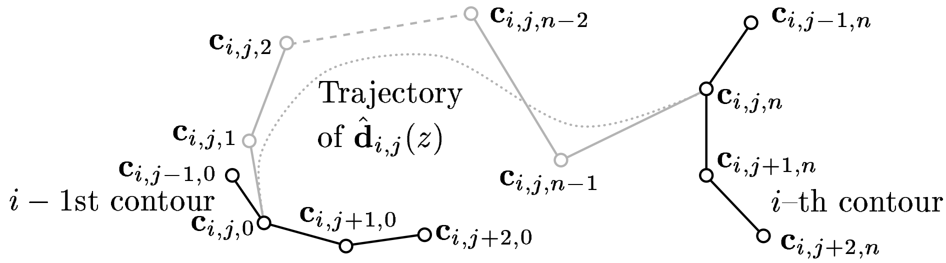


**Fig. 1.** The definition of the interpolating surface.

(*ii*) Inflection or flat points of the contours $\mathbf{x}_i(t)$ occur only at knots. In the case of $C^2$ spline curves, the presence of an inflection or flat point causes the three neighbouring control points to be collinear and we even assume to have $\ddot{\mathbf{x}}_i(t_{\text{infl}}) = \vec{\mathbf{0}}$ at this point.

The first assumption can always be made true by inserting additional knots into the knot vectors $\mathcal{T}_i$ of the contour curves. The upper (resp. lower) bounds for the angles are due to a sufficient convexity criterion by Goodman [2] which will be used in order to guarantee the property of sc–p interpolation. The second assumption is automatically satisfied if the contour curves have been constructed with the help of the algorithm for shape preserving least–square approximation presented in [4].

## §3. Definition of the Surface and Continuity Constraints

The interpolating surface $\mathbf{y}(z,t)$ is defined as a composition of $C$ tensor–product B–spline surfaces $(\mathbf{y}_i(z,t))_{i=1..C}$. The degree of the parameter lines $t = \text{const}$ is equal to $n \geq 2l+2$, whereas the contour curves $z = \text{const}$ are of degree $d$. The $i$–th surface patch $\mathbf{y}_i(z,t)$ is defined over the parameter domain $(z,t) \in [z_{i-1}, z_i] \times [0,1]$ and it possesses the parametric representation

$$\mathbf{y}_i(z,t) = \sum_{j=0}^{\hat{D}_i} \hat{\mathbf{d}}_{i,j}(z) \cdot \hat{N}_{i,j}^d(t) \qquad (i = 1, ..., C) \tag{3}$$

with the $\hat{D}_i + 1$ *contour control points*

$$\hat{\mathbf{d}}_{i,j}(z) = \sum_{k=0}^{n} \mathbf{c}_{i,j,k} \cdot B_k^n(\frac{z - z_{i-1}}{\Delta_{[1]} z_{i-1}}), \quad j = 0, ..., \hat{D}_i; \quad \Delta_{[1]} z_i = z_{i+1} - z_i, \tag{4}$$

running on Bézier curves with control points $(\mathbf{c}_{i,j,k})_{k=0,\ldots,n}$, cf. Figure 1.
The B–spline basis functions $(\hat{N}_{i,j}^d(t))_{j=0,\ldots\hat{D}_i}$ in (3) are defined over the union $\hat{\mathcal{T}}_i$ of the knot vectors $\mathcal{T}_{i-1}$ and $\mathcal{T}_i$ of the adjacent contours. The blending functions $B_k^n(u) = \binom{n}{k} u^k (1-u)^{n-k}$ are the Bernstein polynomials of degree $n$. The third components of the control points $\mathbf{c}_{i,j,k} \in \mathbb{R}^3$ of the surface $\mathbf{y}_i(z,t)$ are chosen according to $c_{i,j,k,3} = (1 - \frac{k}{n}) \cdot z_{i-1} + \frac{k}{n} \cdot z_i$ which implies $y_{i,3}(z,t) \equiv z$. Moreover, the first and last control points $\mathbf{c}_{i,j,0}, \mathbf{c}_{i,j,n}$ $(j = 0, ..., \hat{D}_i)$ of the

trajectories of the contour control points (4) result immediately from the interpolation conditions. They are obtained by representing the adjacent contour curves $\mathbf{x}_{i-1}(t)$ and $\mathbf{x}_i(t)$ as B-spline curves over the knot vector $\hat{\mathcal{T}}_i$ with the help of the knot insertion algorithm, cf. [3].

The first and second components of the remaining control points are unknown yet. They will be computed by solving an appropriate optimization problem. Due to the required order $l = 1, 2$ of differentiability they are subject to the *continuity constraints*

$$\left(\frac{\partial}{\partial z}\right)^{\lambda} \mathbf{y}_i(z,t)\Big|_{z=z_i} \equiv \left(\frac{\partial}{\partial z}\right)^{\lambda} \mathbf{y}_{i+1}(z,t)\Big|_{z=z_i} \qquad (5)$$

for $\lambda = 1, ..., l$ and $i = 1, ..., C-1$. Note that the B–spline basis functions of adjacent surface patches are defined over the possibly different knot vectors $\hat{\mathcal{T}}_i$ and $\hat{\mathcal{T}}_{i+1}$. After representing both sides of (5) over the union of these knot vectors we obtain a set of linear equations for the control points $\mathbf{c}_{i,j,k}$ by comparing the coefficients. The set of linear equations obtained from (5) is denoted by $\mathcal{CC}_i$. Due to the different knot vectors of adjacent surface patches, it includes certain not–a–knot–type conditions for the unknown control points. Resulting from the choice of the polynomial degree $n \geq 2l+2$ of the parameter lines $t$=const, each control point $\mathbf{c}_{i,j,k}$ is subject to one set $\mathcal{CC}_i$ of continuity constraints at most.

## §4. Shape Constraints

Now we consider the conditions on one segment $\mathbf{y}_i(z,t)$ of the interpolating surface which are implied by the desired shape of the contour curves. According to the assumptions made in §2, the shape of the given contours $\mathbf{x}_{i-1}(t) = \mathbf{y}_i(z_{i-1}, t)$ and $\mathbf{x}_i(t) = \mathbf{y}_i(z_i, t)$ coincides with the shape of the control polygons $(\hat{\mathbf{d}}_{i,j}(z_{i-1}))_{i=0,\dots,\hat{D}_i}$ and $(\hat{\mathbf{d}}_{i,j}(z_i))_{i=0,\dots,\hat{D}_i}$. We denote by

$$\Delta_{[2]}\hat{\mathbf{d}}_{i,j}(z) = \sum_{k=0}^{n} B_k^n\left(\frac{z - z_i}{\Delta_{[1]} z_{i-1}}\right) \cdot \Delta_{[2]}\mathbf{c}_{i,j,k} \quad (j = 0, ..., \hat{D}_i - 1) \qquad (6)$$

the difference vectors $\hat{\mathbf{d}}_{i,j+1} - \hat{\mathbf{d}}_{i,j}$ of adjacent contour control points. The difference vectors at $z$=$z_{i-1}$ and $z$=$z_i$ are already known from the interpolation conditions. The following conditions are sufficient for the desired property of sectional curvature–preserving interpolation.

1.) If for two adjacent difference vectors of contour control points the inequality

$$\begin{gathered} 0 \leq \sphericalangle\, (\Delta_{[2]}\hat{\mathbf{d}}_{i,j}(z), \Delta_{[2]}\hat{\mathbf{d}}_{i,j+1}(z)) < \tfrac{\pi}{d-1} \\ (\text{resp. } 0 \geq \sphericalangle\, (\Delta_{[2]}\hat{\mathbf{d}}_{i,j}(z), \Delta_{[2]}\hat{\mathbf{d}}_{i,j+1}(z)) > -\tfrac{\pi}{d-1}) \end{gathered} \qquad (7)$$

holds for both boundaries $z$=$z_i$ and $z$=$z_{i-1}$ whereby the angle $\sphericalangle$ (...) vanishes once at most, then we ensure that it is even true for *all* $z \in [z_{i-1}, z_i]$.

2.) If the angle in (7) vanishes for $z{=}z_{i-1}$ *and* $z{=}z_i$ then the control points $\hat{\mathbf{d}}_{i,j+1}(z_{i-1})$ and $\hat{\mathbf{d}}_{i,j+1}(z_i)$ are an affine combination of their neighbours,

$$\begin{aligned}\hat{\mathbf{d}}_{i,j+1}(z_{i-1}) &= (1-\rho)\cdot\hat{\mathbf{d}}_{i,j}(z_{i-1}) + \rho\cdot\hat{\mathbf{d}}_{i,j+2}(z_{i-1})\\ \hat{\mathbf{d}}_{i,j+1}(z_i) &= (1-\sigma)\cdot\hat{\mathbf{d}}_{i,j}(z_i) + \sigma\cdot\hat{\mathbf{d}}_{i,j+2}(z_i)\end{aligned} \tag{8}$$

with some constants $\rho, \sigma \in \mathbb{R}$. Resulting from the assumption (*ii*) made in §2, these numbers are equal, $\sigma = \rho$. So we can add the linear equations

$$\mathbf{c}_{i,j+1,k} = (1-\rho)\cdot\mathbf{c}_{i,j,k} + \rho\cdot\mathbf{c}_{i,j+2,k} \text{ for } k = 0,...,n \tag{9}$$

to the set of shape constraints. Note that these equations are compatible with the continuity conditions obtained from (5).

The second case may happen for $C^2$ surfaces if both contour curves have an inflection with the same parameter value $t_{\text{infl}}$. The two sets of constraints obtained from 1.) and 2.) guarantee the following property: for each sub–polygon $(\hat{\mathbf{d}}_{i,j}(z))_{j=p,\dots,q}$ $(0 \le p < q \le \hat{D}_i)$ of the contour control polygons the angles between adjacent legs are always non–negative (resp. non–positive) for all $z \in [z_{i-1}, z_i]$ and smaller than $\frac{\pi}{d-1}$ (resp. greater than $-\frac{\pi}{d-1}$), provided that this is true for the boundaries $z{=}z_{i-1}$ and $z{=}z_i$. So it is possible to apply Goodman's sufficient convexity criterion [2]. Therefore the constraints imply that the interpolating contour curves preserve the curvature signs of the given contours.

The conditions obtained from 1.) are guaranteed with the help of linear inequalities which are constructed using the following observation.

**Lemma 2.** *Let a constant* $\lambda \in \mathbb{R}$ *and four vectors* $\vec{\mathbf{u}}_0, \vec{\mathbf{u}}_1, \vec{\mathbf{v}}_0, \vec{\mathbf{v}}_1 \in \mathbb{R}^3$ *satisfying* $\|\vec{\mathbf{u}}_0\|=\|\vec{\mathbf{u}}_1\|=\|\vec{\mathbf{v}}_0\|=\|\vec{\mathbf{v}}_1\|=1$ *and* $0 \le \sphericalangle(\vec{\mathbf{u}}_0, \vec{\mathbf{u}}_1) < \pi$, $0 \le \sphericalangle(\vec{\mathbf{v}}_0, \vec{\mathbf{v}}_1) < \frac{\pi}{d-1}$ *be given. If the control points fulfill the linear inequalities*

$$\begin{aligned}[-\vec{\mathbf{u}}_1, \Delta_{[2]}(\mathbf{c}_{i,j,k} - \lambda\mathbf{c}_{i,j+1,k})] \ge 0, \quad & [\vec{\mathbf{u}}_0, \Delta_{[2]}\mathbf{c}_{i,j+1,k}] \ge 0,\\ [\Delta_{[2]}(\mathbf{c}_{i,j,k} - \lambda\mathbf{c}_{i,j+1,k}), \vec{\mathbf{u}}_0] \ge 0, \quad & [\Delta_{[2]}\mathbf{c}_{i,j+1,k}, \vec{\mathbf{u}}_1] \ge 0,\end{aligned} \tag{10}$$

*and*

$$[\vec{\mathbf{v}}_0, \Delta_{[2]}\mathbf{c}_{i,j,k}] \ge 0, \quad [\Delta_{[2]}\mathbf{c}_{i,j+1,k}, \vec{\mathbf{v}}_1] \ge 0 \tag{11}$$

*for* $k = 0,...,n$, *then the trajectories of the contour control points satisfy the relation in the first line of (7) for all* $z \in [z_{i-1}, z_i]$.

**Proof:** The inequalities (10) and (11) imply that the difference vectors of the control points $\mathbf{c}_{i,j,k}$ can be separated as shown in Figures 2a and b. Due to the convex hull property, this is also true for the difference vectors $\Delta_{[2]}\mathbf{d}_{i,j}(z)$ of the contour control points for $z \in [z_{i-1}]$.

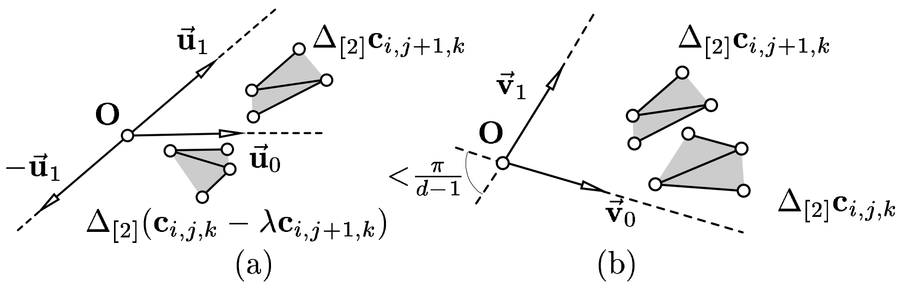


**Fig. 2.** Linearization of the shape constraints (top view).

As evident from Figure 2a the inequalities (10) lead to

$$[\Delta_{[2]}(\mathbf{d}_{i,j}(z)-\lambda\mathbf{d}_{i,j+1}(z)), \Delta_{[2]}\mathbf{d}_{i,j+1}(z)]=[\Delta_{[2]}\mathbf{d}_{i,j}(z), \Delta_{[2]}\mathbf{d}_{i,j+1}(z)] \geq 0, \tag{12}$$

thus they guarantee the left–hand side of (7). Similarly the inequalities (11) imply the right–hand side of (7). ■

We introduced the constant $\lambda$ (based on the identity (12)) in order to keep the number of required inequalities as small as possible. Similarly one can also modify the second argument of the bracket product [.,.]; this yields a mirrored version of Lemma 2.

For generating the linear inequalities which ensure the constraints obtained from 1.) one has to choose a couple of constants and bounding vectors. This is done automatically with the help of algorithms described in the report [5]. Due to space limitations we are not able to describe these algorithms in more detail. They are based on so–called *reference curves* $(\Delta_{i,j}^{\text{sprl}}(z))_{j=0,\ldots,\hat{D}_i-1}$ which represent the expected turns of the difference vectors of the contour control points. For example, the reference curves can be chosen as segments of Archimedean spirals which interpolate the difference vectors $\Delta_{[2]}\mathbf{d}_{i,j}(z_{i-1})$, $\Delta_{[2]}\mathbf{d}_{i,j}(z_i)$ of the control points of the given contour curves.

Lemma 2 can also be applied to subsegments $z \in [z_a, z_e] \subseteq [z_{i-1}, z_i]$ of the trajectories $\hat{\mathbf{d}}_{i,j}(z)$ of the contour control points. We then have to replace the control points $(\mathbf{c}_{i,j,k})_{k=0,\ldots,n}$ by those of the subsegments which result from the de Casteljau scheme. Sometimes it is necessary to use this idea in order to obtain the linearized constraints, see [5].

## §5. Computing the Control Points

The unknown components of the control points $\mathbf{c}_{i,j,k}$ of the interpolating spline surface are found by minimizing an appropriate objective function subject to the linear shape and continuity constraints. The objective function is chosen such that the transition of the control polygons of adjacent contours $(\mathbf{x}_i(t))_{i=0,\ldots,C}$ becomes as smooth as possible. For this we take sample points from the reference curves and minimize the corresponding least–square sum. Moreover we add a "tension term" for the trajectories of the first and the last contour control points, e.g. the sum of the squared lengths $\|\Delta_{[3]}\mathbf{c}_{i,0,k}\|^2$ and

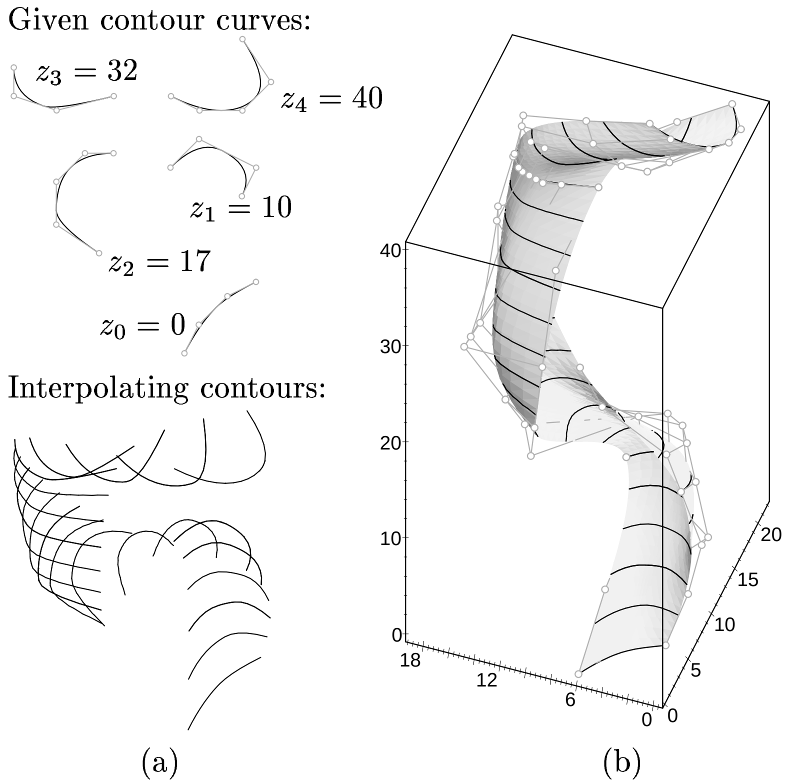


**Fig. 3.** A spline surface which interpolates five given contours.

$\|\Delta_{[3]}\mathbf{c}_{i,\hat{D}_i,k}\|^2$ of their control polygons ($i$=1, ..., $n$−1). This guarantees the uniqueness of the solution, see [5].

We obtain a quadratic function of the control points $\mathbf{c}_{i,j,k}$ ($i$=1, ..., $C$; $j$=0, ..., $\hat{D}_i$; $k$=1, ..., $n$−1). The minimization of the objective function under the linear equality and inequality constraints ensuring the desired shape and continuity properties therefore leads to a *quadratic programming problem* which is solved with the help of an active set strategy as described in the textbook by Fletcher [1]. This strategy requires an initial solution which has to be constructed first with the help of linear programming, i.e., with the simplex algorithm. We choose the initial solution as close as possible to the optimal one. This is achieved by choosing the objective function of the auxiliary linear programming problem as the $l^1$ norm (taken in the linear space of the unknown components of the control points) of the difference to the solution of the unconstrained problem. The latter one is obtained by minimizing the quadratic objective function under equality constraints (which arise from the continuity and interpolation conditions) only. It can be computed using Lagrangian multipliers leading to a system of linear equations for the unknown components of the control points.

Note that the existence of solutions is not automatically guaranteed. In our examples, the feasible region of the LP problem was always non–empty. It can be shown that solutions exist, provided that the polynomial degree $n$ of the parameter lines $t = \text{const}$ has been chosen high enough [5].

As an example we show a spline surface which has been obtained by interpolating five contour curves. The given contours are described by B–spline curves of order three which are defined over different knot vectors. The interpolating $C^1$ spline surface of degree $(2,3)$ preserves the signs of the sectional curvature. Figure 3a shows the top view of the contour curves, whereas the resulting spline surface and its control points have been drawn in Figure 3b. The construction of the surface led to a quadratic programming problem with 92 unknowns, 48 equality constraints and 201 inequality constraints. In our implementation we use the equality constraints for eliminating a part of the unknowns from the problem; this yielded a QP problem with only 44 unknowns. Only four inequalities are active for the final solution.

**Acknowledgments.** The author thanks Professor Tim Goodman for his interest in the presented work. The financial support by the European Union through the network FAIRSHAPE is also gratefully acknowledged.

## References

1. Fletcher, R., *Practical Methods of Optimization*, Wiley, Chichester 1991.
2. Goodman, T. N. T., Inflections on curves in two and three dimensions, Comput. Aided Geom. Design **8** (1991), 37–50.
3. Hoschek, J., and D. Lasser, *Fundamentals of Computer Aided Geometric Design*, A. K. Peters, Wellesley MA, 1993.
4. Jüttler, B., Shape preserving least–square approximation by polynomial parametric spline curves, University of Dundee, Applied Analysis Report 965, 1996, to appear in Comput. Aided Geom. Design.
5. Jüttler, B., Sectional Curvature Preserving Approximation of Contour Lines, University of Dundee, Applied Analysis Report 966, 1996.
6. Kaklis, P. D. and A. I. Ginnis, Sectional–Curvature Preserving Skinning Surfaces, National Technical University of Athens, Ship–Design Laboratory Technical Report, to appear in Comput. Aided Geom. Design.
7. Schumaker, L. L., Reconstructing 3D objects from cross–sections, in *Computation of Curves and Surfaces,* W. Dahmen, M. Gasca and C. A. Micchelli (eds.), Kluwer, Dordrecht, 1990, 275–309.
8. Unsworth, K., Recent developments in surface reconstruction from planar cross–sections, in *Computer Aided Geometric Design,* C. A. Micchelli and H. B. Said (eds.), Annals of Numerical Mathematics **3** (1996), 401–422.

Bert Jüttler
Technische Hochschule Darmstadt
Fachbereich Mathematik, AG 3
Schloßgartenstraße 7
64289 Darmstadt, GERMANY
juettler@mathematik.th-darmstadt.de