A free form spline software package[1]
— Documentation for version 1.1 —

S. Bischoff, Th. Müller, H. Prautzsch[2]

Universität Karlsruhe
Fakultät für Informatik
Am Fasanengarten 5
D - 76128 Karlsruhe
Germany

Karlsruhe, 3rd September 1999

Abstract

This software package can be used to build spline surfaces of arbitrary topological shape. The resulting surfaces are of geometric continuity order 1 and bidegree 4 or 5 or of geometric continuity order 2 and bidegree 6 or 13.

Keywords: G-splines, n-sided fillings, n-sided patches.

1 The method

This free form spline package can be used to construct G^r-spline surfaces of bidegree $2r+2$ or $2r^2+2r+1$, for $r = 1, 2$. Especially, it can be used to fill n-sided holes of a G^r-surface for $n = 3, 5, 6, 7, 8$.

The general idea behind our free form splines is described in [Prautzsch '97] and [Prautzsch & Boehm '99, Chapter 14.4]. Here we only recall their properties, which are necessary and sufficient to work with these splines.

Every free form spline has an affinely invariant control net, i.e., an affine image of the net controls the corresponding affine image of the spline surface.

[1]Supported by DFG grant # PR 565/1-1

[2]Corresponding author, e-mail: prau@ira.uka.de

The control net of a G^1-free form spline is a quadrivalenced net, i.e., all vertices have valence four as illustrated in Figure 1 for a simple example.

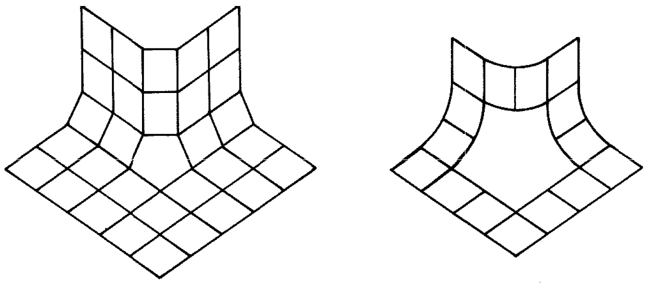

Figure 1: Control net of an n-sided G^1-free form spline (left) and the corresponding $3n$ ordinary patches (right) for $n = 5$.

The control net of a G^2-free form spline is a quadrilateral net, i.e., all meshes have four vertices as illustrated in Figure 2.

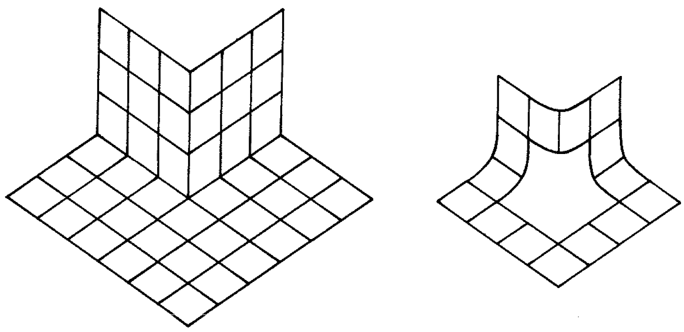

Figure 2: Control net of an n-sided G^2-free form spline (left) and the corresponding $3n$ ordinary patches (right) for $n = 5$.

Each $r+1 \times r+1$ regular subnet of such a control net defines an **ordinary patch** of the surface. If $\mathbf{c}_{ij}, i, j \in \{-(r+1), \ldots, 0\}$, denote the vertices of any such subnet, the corresponding patch is given by

$$\sum_{i=-(r+1)}^{0} \sum_{j=-(r+1)}^{0} \mathbf{c}_{ij} N_i^{(r+1)}(x) N_j^{(r+1)}(y), \quad x, y \in [0,1],$$

where N_i^{r+1} is the B-spline of degree $r+1$ over the equidistant knots $i, i+1, \ldots, i+r+2$.

The control net is assumed to be such that the ordinary patches form a connected surface. If the surface has n-sided holes surrounded by $3n$ patches as illustrated in Figures 1 and 2, then **extra patches** of bidegree $2r+2$ or $2r^2+2r+1$ can be constructed by the programs below. Together, the ordinary and the extra patches form one G^r-surface.

Remark 1
Some pictures illustrating our free form splines are available at our homepage:

`http://i33www.ira.uka.de`

Remark 2
The extra patches minimize certain quadratic fairness functionals. A detailed account is given in [Müller '97] and [Bischoff '98].

2 How to use the programs

The package contains two shell scripts,

`compile` and `off2bez`,

five programs,

`bez2list`, `fill_hole`, `off2invec`, `outvec2bez` and `subdivide`,

40 data files used by `fill_hole` and 10 example .off-files.

`compile`
To use the software, compile the C++ source code by the shell script

`compile`

A single file *name*.`c++` can be compiled by

`compile` *name*

Note to edit the shell script `compile` in order to adapt the C++ compiler and the include- and library-path to you system-setup.

`off2bez`
The second shell script provides the simplest way to obtain a free form surface from an n-sided control net of the form illustrated in Figures 1 and 2, where $n = 3, 5, 6, 7$ or 8. The output is a shaded image generated by `geomview`. The second shell script is used as follows:

`off2bez` *off-file* m

The *off-file* contains the control net in the Geomview .off-format.

The parameter $m \in \{0, 1, 2, 3\}$ specifies a certain type of the free form spline.

For $m = 1, 2$ the free form spline surface consists of $3n$ boundary and $4n$ interior patches of degree $2r + 2, r = 1, 2$, as illustrated in Figure 3 (right). The $3n$ boundary patches are ordinary patches as explained in Section 1.

For $m = 0, 3$ the surface consists of $3n$ boundary and n interior patches of degree $2r + 2$ or $2r^2 + 2r + 1$, respectively, as illustrated in Figure 3 (left). For $m = 0$ the boundary and cross boundary derivatives up to order 1 (G^1-case) or 2 (G^2-case) are different from the ones for $m = 1, 2$, in general. But everywhere else the $3n$ boundary patches are the same. For $m = 3$ the $3n$ boundary patches are everywhere the same as for $m = 1, 2$.

`off2invec`
The program `off2invec` converts a control net as in Figure 1 or 2 given in the Geomview .off-format to the here so-called *invec* format. It is used as follows:

`off2invec` *off-file invec-file*

The *invec-file* is an ASCII-file containing all $9n$ (G^1-case) or $12n+1$ (G^2-case) vertices of the control net in the ordering given by Figure 4. Each line of this file contains the 3 coordinates of one vertex as in the following example:

0.707107	2	0
0.707107	2.70711	0.717107
0	2	0.707107
1.70711	1	0
	⋮	

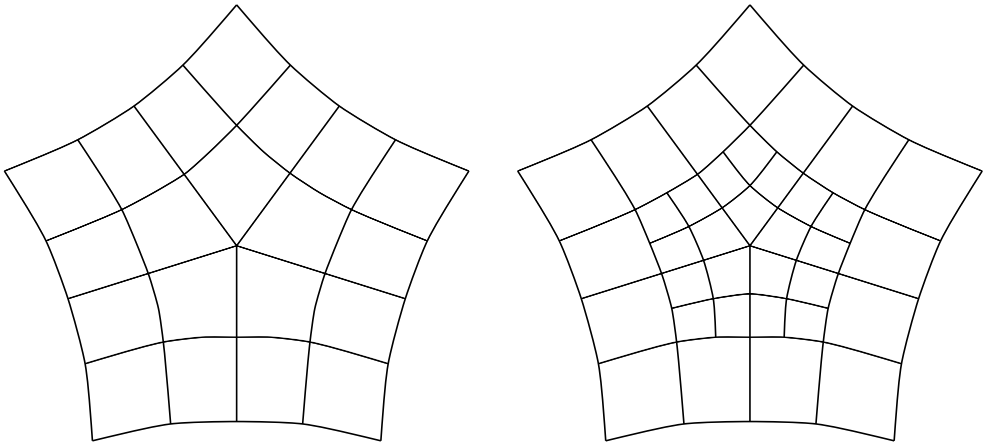

Figure 3: Schematic illustration of a free form spline surface consisting of $3n$ boundary and n interior patches (left) as for $m = 0, 3$ and a free form spline surface consisting of $3n$ boundary and $4n$ interior patches (right) as for $m = 1, 2$.

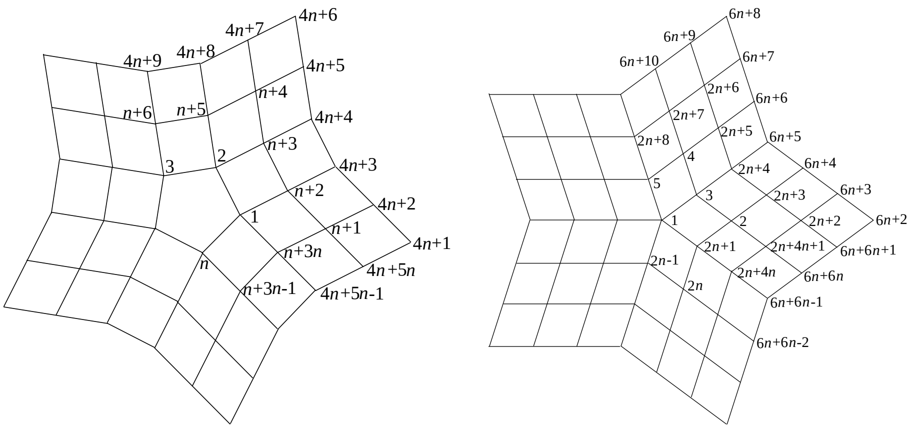

Figure 4: Labelling of the control points in the *invec-file* for $n = 5$. Left: G^1-case, right: G^2-case.

`fill_hole`

The kernel of the software package is the program `fill_hole`. Here the control net is "multiplied" by one of the 40 matrices stored in the data files. The program is used as follows:

`fill_hole` *invec-file outvec-file m*

The *outvec-file* is an ASCII-file created by `fill_hole`. It contains the spline control points of the patches of the free form spline surface. Let r denote the smoothness order of the spline surface and let $\mathbf{c}_0, \mathbf{c}_1 \ldots, \mathbf{c}_\mu$ be the points stored in the *outvec-file*. Then the k-th patch is given by

$$\sum_{i=0}^{d} \sum_{j=0}^{d} \mathbf{c}_{(d+1)^2 k + (d+1)i + j} N_i^d(x) N_j^d(y) ,$$

where d is either $2r+2$ or $2r^2+2r+1$ and $N_0^d, \ldots, N_d^d$ are the B-splines of degree d over the knots

$$\underbrace{0, \ldots, 0}_{r+1}, \underbrace{1, \ldots, 1}_{d-r}, \underbrace{2, \ldots, 2}_{d-r}, \underbrace{3, \ldots, 3}_{r+1} .$$

The parameter m is defined as for the shell script `off2bez`.

`outvec2bez`

The program `outvec2bez` converts an *outvec-file* created by `fill_hole` to an *ext-bez-file*. It is used as follows:

`outvec2bez` *outvec-file ext-bez-file*

An *ext-bez-file* is in the Geomview .bez-format but allows also for degrees $7, \ldots, 16$. The degree is hexadecimally coded in the range $0, \ldots, f$. Thus an *ext-bez-files* with degree ≤ 6 is an ordinary Geomview *bez-file* and can be visualized using `geomview` (activate `smooth shading` for best results):

`geomview` *ext-bez-file*

`subdivide`

The program `subdivide` is used to subdivide the Bézier net given in an *ext-bez-file1* generated by `outvec2bez`. It subdivides each patch using the de Casteljau algorithm at parameter value $(x, y) = (1/2, 1/2)$ and writes the result to an *ext-bez-file2*. It is used as follows:

`subdivide` *ext-bez-file1 ext-bez-file2* `bez2list`

The program `bez2list` converts an *ext-bez-file* to a *list-file* in the Geomview .list-format containing the Bézier nets of the patches of the free form spline surface. It is used as follows:

`bez2list` *ext-bez-file list-file*

The *list-file* can be visualized using `geomview` as described for `outvec2bez`.

Note to perform one or two subdivision steps using `subdivide` before visualizing the *list-file* to get a better approximating Bézier net.

References

S. Bischoff (1998). Konstruktion normalenstetiger Flächen beliebiger Form. IBDS, Universität Karlsruhe, 1998. Studienarbeit.

Th. Müller (1997). Glatte Splineflächen aus Kontrollnetzen beliebiger Topologie. Master's thesis, IBDS, Universität Karlsruhe, 1997.

H. Prautzsch (1997). Freeform splines. *Computer Aided Geometric Design*, 14(3):201–206.

H. Prautzsch and W. Boehm (1999). *Bézier and B-Spline Techniques.* To appear: Springer-Verlag.