

TR/14/91

October 1991

An Arbitrary Mesh Network Scheme  
using Rational Splines

by

John A. Gregory and P. K. Yuen

Submitted for publication in:

Mathematical Methods in CAGD and Image Processing

Tom Lyche and L L Schumaker (eds.)

(Per)

QA

1

B 78

z157747x

# An Arbitrary Mesh Network Scheme using Rational Splines

John A. Gregory and P.K. Yuen

**Abstract.** A  $C^1$  surface scheme is described which interpolates points defined on an arbitrary mesh network. The scheme involves the blending of strip ‘functions’ developed from a rational spline method. The rational spline provides interval and point tension weights which can be used to control the shape of the surface scheme.

## §1. Introduction

Most ‘free-form’ or ‘sculptured’ modelling systems for Computer Aided Design are based on the representation of the surface as a complex of ‘rectangular patches’. In this paper, however, we wish to give a progress report on some research into the construction of a  $C^1$  surface which interpolates points defined on an arbitrary 3-D mesh network. The construction is based on the blending of ‘strip functions’ defined on the mesh, which here are developed from a rational spline method.

The rational spline provides ‘interval’ and ‘point tension’ parameters to control the shape of the strip functions and hence of the blended surface. The strip functions contain position and tangent plane information and are blended together to form polygonal patches appropriate to the topological structure of the mesh.

We begin, in Section 2, by considering the topology and data structure of the mesh network, where for the purposes of this paper some simplifying assumptions are made. The rational spline strip functions are developed in Section 3 and the blending of these strip functions is considered in Section 4. The paper is completed in Section 5 with some examples and concluding remarks concerning the future direction of the research.

## §2. The Mesh Network

The mesh network is described by a set of 3-D directed *polygonal lines* whose *vertices* are a given set of interpolation points. The polygonal lines are assumed to intersect at certain of the vertices in such a way as to define a mesh network of *faces*. The faces are topologically equivalent to either triangular, rectangular or, in general, polygonal domains. Each *edge* of a face is defined by a signed subset of one of the polygonal lines, where the sign indicates whether it has the positive or negative direction of that line.

The mesh network is assumed equivalent to an orientable surface (thus having an inside and outside) and the edges of the faces, together with their vertices, are labelled in an anti-clockwise order when viewed from the outside. The data structure thus consists of:

- *Polygonal Lines*  $E_i$ ,  $i = 1, \dots, n$ , each defined by an ordered set of vertices denoted by  $P_{i,j} \in \mathbb{R}^3, j = 1, \dots, n_i$ .
- *Faces*  $M_i$ ,  $i = 1, \dots, m$ , defined by their anti-clockwise ordered edges, namely, signed subsets of the polygonal lines.

It can be observed that the vertices divide into the two classes of *face vertices* and *internal edge vertices*. Here, we make the simplifying assumptions that the face and internal edge vertices form mutually exclusive sets and that exactly two of the polygonal lines meet or intersect at any face vertex. (Thus every face vertex is labelled exactly twice with the above notation.) This allows the consistent definition of the strip functions of the following section, with respect to an appropriate local rectangular cartesian coordinate system. In particular, a consistent tangent plane will be defined at each face vertex.

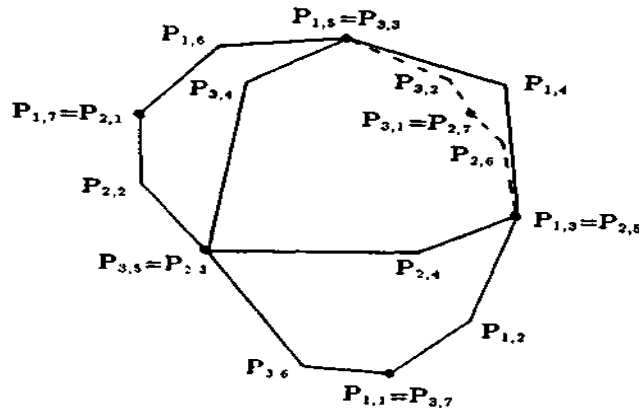


Figure 2.1. Example of a mesh network.

As a concrete example, consider the mesh network shown in Figure 2.1. Here, there are three directed polygonal lines  $E_i$ ,  $i = 1, \dots, 6$ , through a total of fifteen vertices. The vertices decompose into a set of six face vertices

and nine internal edge vertices, giving a set of four triangular faces  $M_i$ ,  $i = 1, \dots, 4$ . (The edges of the face hidden by the perspective view of the mesh network are shown with broken lines as an aid to understanding the three dimensional figure.)

### §3. The Rational Spline Strip Functions

We begin by constructing a set of rational spline curves, each associated with a given polygonal line  $E_i$ , and interpolating the ordered vertices which define  $E_i$ . This will provide the basic design tool for the construction of the arbitrary mesh network surface, since the surface will be constrained to contain the spline network.

The ordered vertices of each polygonal line decompose into face vertices and interior edge vertices. The ordered face vertices are associated with increasing integer knots and we choose to distribute the knots associated with the interior edge vertices between these integers, in proportion to their number between the subsequent face vertices. This gives a knot vector which may not be appropriate for standard parametric cubic spline interpolation and we thus wish to provide *shape control parameters* which allow easy manipulation of the spline curve. This is achieved here by the use of a  $C^2$  rational cubic spline method, see [6], which provides interval and point tension control of the curve. (An alternative choice would be the  $C^1$  weighted  $\nu$ -spline described in Foley [2], which also provides interval and point tension control.)

Given a set of values  $P_i \in \mathbb{R}^3$ ,  $i = 0, \dots, N$ , defined at knots  $s_i$ ,  $i = 0, \dots, N$ , with interval spacing  $h_i := s_{i+1} - s_i > 0$ , then a parametric  $C^1$  piecewise rational cubic function  $p : [s_0, s_n] \rightarrow \mathbb{R}^3$  is defined by

$$P|_{[s_i, s_{i+1}]}(s) := \frac{(1-\theta)^3 \alpha_i P_i + \theta(1-\theta)^2 (\gamma_i + \alpha_i) V_i + \theta^2 (1-\theta) (\gamma_i + \beta_i) W_i + \theta^3 \beta_i P_{i+1}}{(1-\theta)^2 \alpha_i + \theta(1-\theta) \gamma_i + \theta^2 \beta_i}.$$

Here, (3.1)

$$\theta|_{[s_i, s_{i+1}]} := (s - s_i) / h_i, \quad (3.2)$$

and

$$V_i := P_i + \frac{\alpha_i}{\gamma_i + \alpha_i} h_i D_i, \quad W_i := P_{i+1} - \frac{\beta_i}{\gamma_i + \beta_i} h_i D_{i+1}, \quad (3.3)$$

where the scalar weights  $\alpha_i, \beta_i, \gamma_i$  are assumed positive and  $D_i$ ,  $i = 0, \dots, N$  denote first derivative values defined at the knots. Thus  $p$  has the Hermite interpolation properties that

$$p(s_i) = P_i, \text{ and } p^{(1)}(s_i) = D_i, \quad i = 0, \dots, N. \quad (3.4)$$

Suppose the ‘end conditions’  $D_0$  and  $D_N$  are given. Then the unknowns  $D_i$ ,  $i = 1, \dots, N$  can be found by imposing the  $C^2$  consistency constraints

$$P^{(2)}(s_i^+) = P^{(2)}(s_i^-), \quad i = 1, \dots, N - 1. \quad (3.5)$$

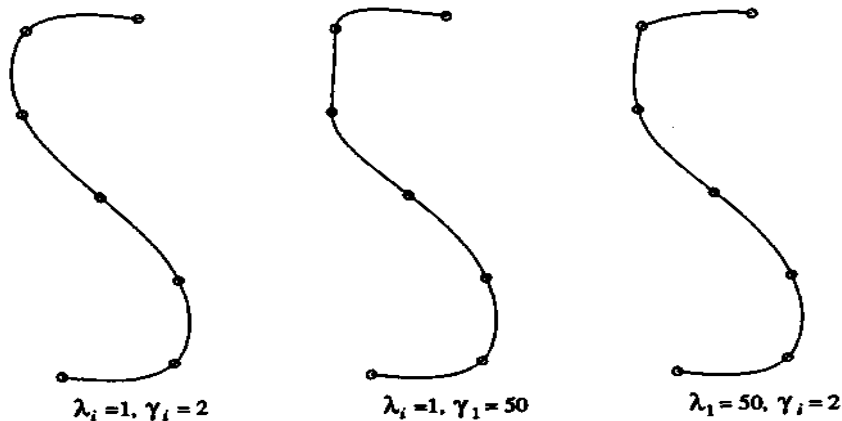
These constraints lead to a tridiagonal system of equations in the unknowns, which is diagonally dominant if  $\gamma_i > \alpha_i$ , and  $\gamma_i > \beta_i$ ,  $i = 1, \dots, N - 1$ . We thus have a sufficient condition for the existence of a unique, easily computed solution to the consistency constraints.

The scalar weights of the piecewise rational form provide shape control parameters for the  $C^2$  curve which results from this interpolatory process, see [6], where a local support basis representation is also discussed. In particular, we choose

$$\beta_{i-1} = \alpha_i := 1/\lambda_i, \quad (3.6)$$

giving weights  $\lambda_i$ ,  $i = 0, \dots, N$ , and  $\gamma_i$ ,  $i = 0, \dots, N - 1$ , for the rational spline scheme, where the diagonal dominance condition is now  $\gamma_i \lambda_i > 1$ ,  $i = 1, \dots, N - 2$ .

The weight  $\lambda_i$  is called a point tension parameter, associated with the knot  $s_i$ , since a tangent discontinuity is introduced into the spline curve at  $P_i$  as  $\lambda_i \rightarrow \infty$ . The weight  $\gamma_i$  is called an *interval tension parameter*, associated with the interval  $[s_i, s_{i+1}]$ , since the curve is tightened to the line segment joining  $P_i, P_{i+1}$  as  $\gamma_i \rightarrow \infty$ . These properties are illustrated in Figure 3.1, where the cubic spline interpolant of the first curve is given by the use of the default parameters  $\lambda_i := 1$ ,  $i = 0, \dots, N$ , and  $\gamma_i := 2$ ,  $i = 0, \dots, N - 1$ .



**Figure 3.1. Tension behaviour of the rational spline..**

Application of the rational spline method to the vertex data along each polygonal line  $E_i$  of the arbitrary mesh network, using the previously defined knot distribution, results in a network of spline curves. These curves define a *wire frame model* of a surface which has the same topological structure as the mesh network and the shape of the wire frame model can be manipulated by varying the control parameters which govern the shape of each curve. At each face vertex of the mesh network there is a unique tangent plane for the wire frame surface, defined by the derivative parameters (i.e. tangent vectors) of the two spline curves which meet or intersect at that vertex. In particular, at

any integer knot of a rational spline curve, a ‘*cross boundary*’ tangent vector is defined.

We now define a bivariate ‘*strip function*’ extension of the rational spline method on  $[s_0, s_N] \times [\varepsilon, \varepsilon]$ , for some  $\varepsilon > 0$ , by the Taylor form

$$P^{ext}(s,t) := p(s) + tp_t(s), \quad (3.6)$$

where  $p_t(s)$  denotes the piecewise cubic Hermite interpolant to a set of cross boundary tangent and ‘*twist*’ vectors, defined on a subset of the rational spline knots. Here, the cross boundary tangent denotes a derivative along the  $t$  direction and the twist vector denotes a (1,1) derivative parameter along the  $s$  and  $t$  directions. The twist vectors can be considered as derivative unknowns to be determined by imposing  $C^2$  constraints across the knot subset. In this case  $p_t(s)$  is the cubic spline interpolant to the cross boundary tangent vectors, where we assume that appropriate end conditions have been imposed. It can be observed that

$$p^{ext}(s,0) = p(s) \text{ and } \partial_t p^{ext}(s,0) = p_t(s), \quad (3.7)$$

and hence the strip function extension contains position and tangent plane information about the rational spline curve.

The rational spline extension method is considered along each curve of the wire frame model, where the cross boundary tangents are given on integer subsets of the spline knots. Since these cross boundary tangents are defined with respect to the local parameterizations of their corresponding spline curves, it is necessary to change certain of their signs for them to be consistent with the  $t$  parameterization on the strip function domain. This sign information can be obtained from the face table data structure. The twist vectors for the rational spline extensions are initially determined by imposing  $C^2$  spline constraints on the piecewise cubic Hermite cross boundary tangent function, see above. This will give incompatible twist vector values at each face vertex, corresponding to the two rational spline extensions which meet or intersect at that vertex. This incompatibility must be allowed for in the construction of the surface scheme of the following section or, alternatively, the strip functions can be redefined by taking the average of the two twist vector parameters at each face vertex. In either case, the surface model is now one of rational spline strip functions which can be blended to form a  $C^1$  surface.

#### §4. Blending the Strip Functions on Polygonal Domains

Consider a face of the arbitrary mesh network. At a face vertex, two edges meet and along each edge a piece of strip function is defined, consistent with a  $C^1$  surface. More precisely, our definition and parameterization of the strip functions implies that the two strip function pieces can be written in terms of a  $(u, v)$  rectangular cartesian coordinate system about each vertex as

$$p(0,v) + up_u(0,v), \quad 0 \leq u \leq 1, \quad \text{and} \quad p(u,0) + vp_v(u,0), \quad 0 \leq v \leq 1, \quad (4.1)$$

cf (3.6). Here, after appropriate simple affine transformations,  $p(0, v)$  and  $p(u, 0)$  represent the pieces of the rational spline curves, and  $p_u(0, v)$  and  $p_v(u, 0)$  represent the cross boundary tangent functions, between the face vertices of each edge. Moreover, these Taylor functions are consistent with a  $C^1$  surface  $p(u, v)$ . In particular, a  $C^1$  extension of the two strip functions is defined for  $(u, v) \in [0, 1]^2$  by the ‘rationally corrected’ Boolean sum Taylor interpolant, see Gregory [3],

$$p(u, v) := p(0, v) + up_u(0, v) + p(u, 0) + vp_v(u, 0) - [1 \ u] \begin{bmatrix} p(0,0) & p_u(0,0) \\ p_u(0,0) & t(u,v) \end{bmatrix} \begin{bmatrix} 1 \\ v \end{bmatrix}, \quad (4.2)$$

where

$$t(u, v) := \frac{uP_{v,u}(0,0) + vP_{u,v}(0,0)}{u + v}, \quad (4.3)$$

and  $p_{u,v}(0,0) := \partial_u \partial_v p(0,0)$ ,  $p_{v,u}(0,0) := \partial_u \partial_v p(0,0)$  denote the twist vectors at the vertex. If  $p_{u,v}(0,0) \neq p_{v,u}(0,0)$ , the rational term introduces singularities which are removable to first order at the vertex. If  $p_{u,v}(0,0) = p_{v,u}(0,0)$ , then

$$t(u, v) := p_{u,v}(0,0), \quad (4.3)$$

giving a standard Boolean sum form.

Now each face of the mesh network is topologically equivalent to a polygonal domain and the final surface is constructed by blending the Boolean sum Taylor interpolants on each of these polygons in the following manner. This follows the approach of Charrot and Gregory [1], see also [3] and [5].

Let  $\Omega$ , denote a generic regular polygon with vertices  $X_i$ ,  $i = 0, \dots, N - 1$ . Associated with each vertex  $X_i$  there is a Boolean sum Taylor interpolant  $P_i(u, v)$  say, which must be reparameterized onto the polygonal domain. This is achieved by ‘central projections’. Thus let

$$(1 - u_i)X_i + u_i X_{i+1}, \text{ and } (1 - v_i)X_i + v_i X_{i-1} \quad (4.4)$$

define the points of intersection of the two central projections through  $X \in \Omega$  with the sides of the polygon as shown in Figure 4.1. Then

$$\phi_i(X) := (u_i, r_i) := \left( \frac{d_{i-1}}{d_{i-1} + d_{i+1}}, \frac{d_i}{d_{i-2} + d_i} \right), \quad (4.5)$$

defines the central projection coordinate chart about  $X_i$ , where  $d_i(X)$  denotes the perpendicular distance from  $X$  to the side  $X_i X_{i+1}$ .

The blended polygonal surface patch is now defined by

$$p(X) := \sum_{i=0}^{N-1} w_i(X) p_i(\phi_i(X)), \quad (4.6)$$



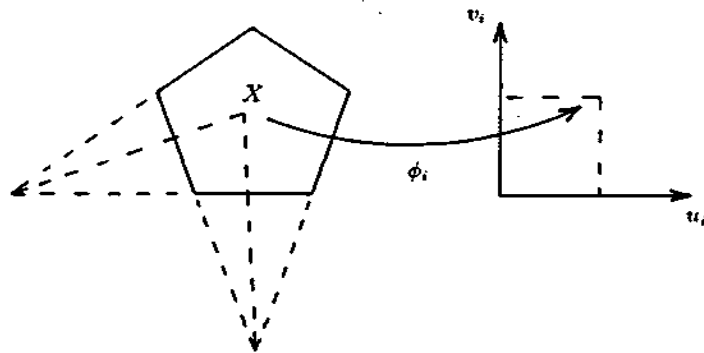


Figure 4.1. The central projection coordinate chart.

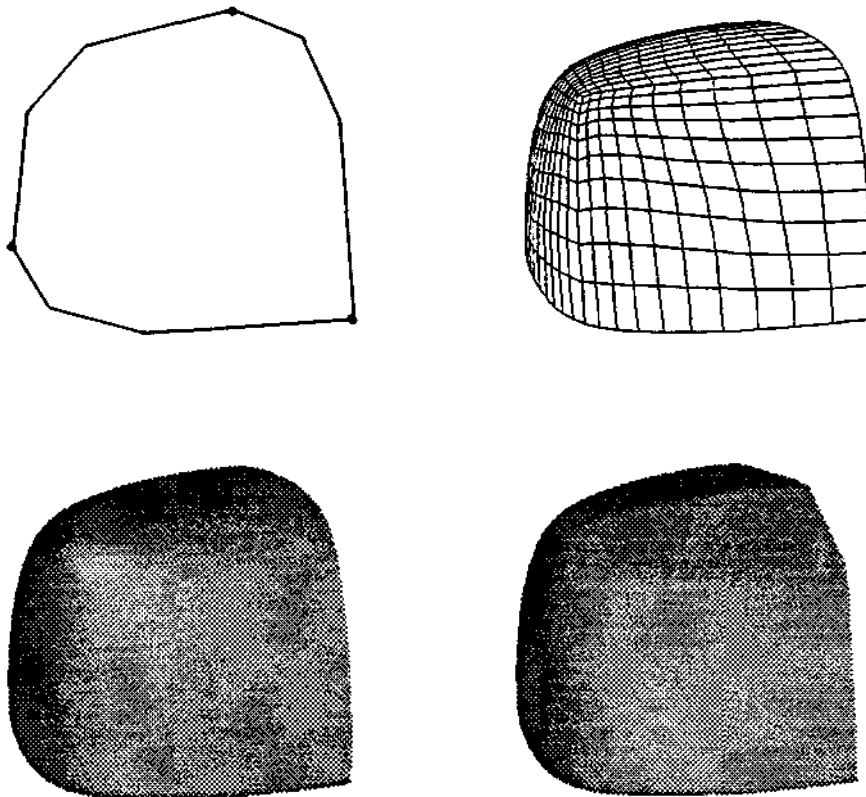


Figure 5.1. A single triangular patch.

where

$$w_i(X) := \frac{\prod_{j \neq i-1, i} d_j^2}{\sum_{k=0}^{N-1} \prod_{j \neq k-1, k} d_j^2} \quad (4.7)$$

defines a weight function which is constructed such that  $p_i$  contributes, in position and tangent plane, only on those edges where it interpolates a strip function.

### §5. Examples and Concluding Remarks

We conclude by giving two simple examples of the arbitrary mesh network scheme. The first case, shown in Figure 5.1, is that of a single triangular patch, defined with zero twists, and illustrates the effect of increasing the interval tension parameter between two internal edge vertices.

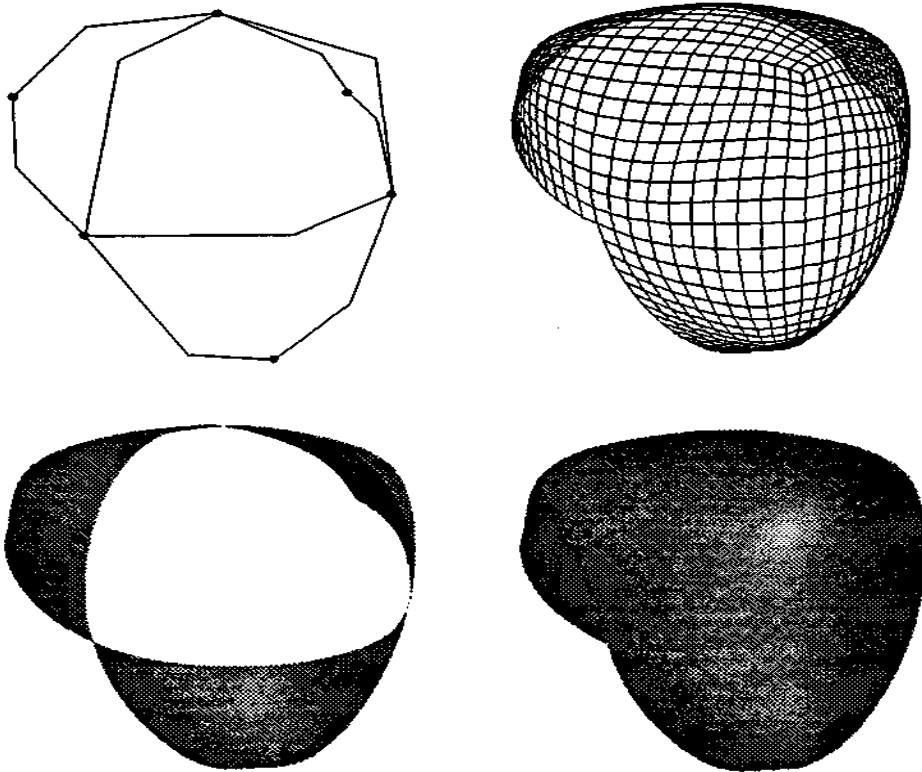


Figure 5.2. A mesh network of triangular patches.

The second case, shown in Figure 5.2, is of the mesh network of four triangular patches, given as the concrete example of Section 2. In the figure, the rational spline network is that defined by the special case of cubic spline interpolation. The figure shows that the blending approach proposed here gives a smooth, tangent plane continuous surface.

Our current research is directed towards allowing more than two curves to meet or intersect at any face vertex, thus removing a major restriction on the topology of the network proposed here. This involves modification of the derivative parameters of the rational spline network at such face vertices, in order that a consistent tangent plane is defined there. It is also proposed to investigate shape control of the interior of a patch, other than through the control of its boundary. Finally, the method can be extended to the construction of higher order continuous surfaces, although more care is now needed in the construction of the blends of the strip functions, as is discussed in [4] and [5].

**Acknowledgements:** This work was supported by the SERC ACME Grant GR/F80838, in collaboration with Siemens Nixdorf Information Systems Ltd.

### References

1. Charrot, P., and J. A. Gregory, A pentagonal surface patch for computer aided geometric design, *Computer-Aided Geom. Design* 1 (1984), 87-94.
2. Foley, T. A., Interpolation with interval and point tension controls using weighted  $v$ -splines, *ACM Trans. Math. Software* 13 (1987), 68-96.
3. Gregory, J. A.,  $C^1$  rectangular and non-rectangular surface patches, in *Surfaces in Computer Aided Geometric Design*, R. E. Barnhill and W. Boehm (eds.), North Holland, Amsterdam, 1983, 25-33.
4. Gregory, J. A., and J. Hahn, Geometric continuity and convex combination patches, *Computer-Aided Geom. Design* 4 (1987), 79-89.
5. Gregory, J. A., J. Hahn, and V.K.H. Lau, High order continuous polygonal patches, TR/06/91, Mathematics & Statistics Department, Brunel University.
6. Gregory, J. A., M. Sarfraz, and P. K. Yuen, A rational spline with interval and point tension, in preparation.

John A Gregory and P.K. Yuen  
Department of Mathematics and Statistics  
Brunel University  
Uxbridge, UB8 3PH, U.K.  
John.Gregory@brunel.ac.uk  
Pak-Kay.Yuen@brunel.ac.uk

**NOT TO BE  
REMOVED**  
FROM THE LIBRARY

XB 2327798 X

