# A survey of blending methods that use parametric surfaces

Janos Vida, Ralph R Martin* and Tamas Varady

The paper discusses the blending problem in geometric modelling, and it provides a comprehensive review of solutions that use parametric surfaces A terminology and a classification are presented to help clarify the nature of blending, and the relationships between various parametric blending methods Several geometric techniques are evaluated, highlighting concepts which the authors feel to be important Topological issues are also discussed In conclusion, the applicability and efficiency of parametric techniques for general blending situations are emphasized, and open questions for future research are presented An up-to-date list of publications on blending, including parametric-surface methods and other methods, is provided as a key to the literature

**Keywords: geometric modelling, blends, parametric surfaces**

## What are blends?

Blends are, quite simply, intermediate surfaces which smoothly join other surfaces of the exterior of an object Often, some of the original surfaces of the object would have intersected in sharp edges, and, for a variety of reasons, technical, aesthetic or otherwise, it may be desired to replace some or all of the sharp edges by corresponding smooth faces, or blends These surfaces meet the rest of the object with (at least) $G^1$ continuity In general, we may not wish to smooth just separate edges, but also vertices where they meet, or even a whole region of the original surface of the object An edge blend and a vertex blend are shown in *Figure 1*, a region blend is shown in *Figure 2* Note that, in many cases, the ease of construction of blend surfaces is more important than the exact nature of the shape produced. This means that each blend must be able to be specified with a small number of parameters, and that the blend geometry created as a result must be readily predictable

These basic ideas may be taken further For example, it may be desired to create a smooth transition surface between a pair of surfaces which did not originally intersect (see *Figure 3* for a transition surface between two spheres) Such constructions will not be considered further here

Various other papers have discussed at length the nature of blending – what blends are and why they are used – and we do not propose to repeat that material here The interested reader should consult the thesis by Rockwood[1], and the papers by Pratt[2], Woodwark[3], and the authors[4] of this paper among others

## Aim of paper

The main purpose of this paper is to provide a survey of blending methods which produce parametric surfaces For conciseness, we shall often refer to such methods simply as *parametric-blending* methods It is probably fair to say that blend methods which produce implicit (or algebraic) surfaces have received more publicity in the past, and so, to some extent, we are aiming to redress the balance Nevertheless, in the interests of completeness, our references include papers on implicit blending, and we feel that the reference list in this paper is a reasonably complete survey of blending literature in general However, it should be noted that this survey does not attempt to describe implementations of blending that are available in commercial systems unless the methodology used is readily available in the published literature.

In the rest of this paper, we first consider various types

Computer and Automation Institute, Hungarian Academy of Sciences, PO Box 63, H-1518 Budapest, Hungary
*University of Wales College of Cardiff, PO Box 916, Cardiff CF2 4YN, UK
J Vida is currently at the Department of Information Technology and Automation, University of Veszprem, Egyetem u 10, H-8200 Veszprem, Hungary
*Paper received 5 November 1992 Revised 4 November 1993*

0010-4485/94/05/0341-25 © 1994 Butterworth-Heinemann Ltd

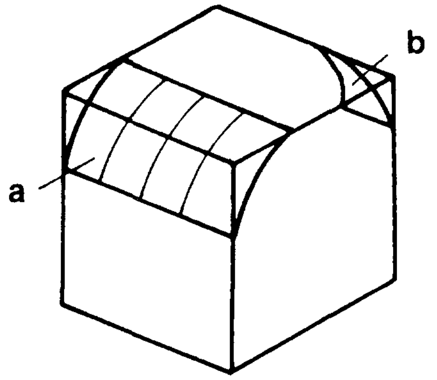



**Figure 1** Blends, (a) edge blend (b) vertex blend

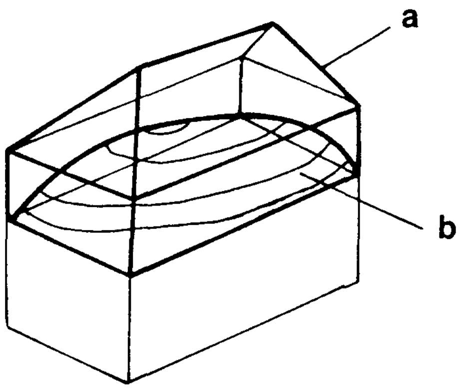



**Figure 2** Blending, (a) region being blended, (b) region blend

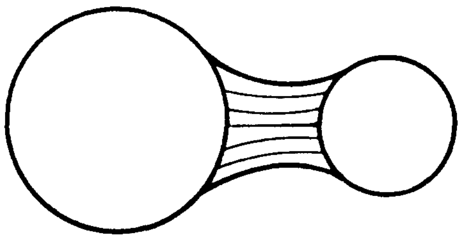


**Figure 3** Transition surface between two nonintersecting spheres

of blending, to show how parametric blending fits in more generally with other approaches to blending We then provide definitions for the terminology we use, and also review the terminology of other authors, as descriptions of blending work currently seem to use a variety of terms for similar or even the same concepts

The main body of the paper then goes on to consider how we can classify various approaches to the construction of parametric blends using rolling-ball-based, spine-based, trimline-based, polyhedral, and other methods We then review in detail the techniques published under each of these headings Finally, we consider to a certain extent various termination and topological issues, especially the problems that arise when blend surfaces encounter vertices in the original object model

## TYPES OF BLENDS

Various authors, including the authors of this paper, have attempted to classify blending techniques[3-5] Different aspects of blending may be considered to be important for various reasons, and obviously the choice of which are the important aspects will affect the hierarchy that is chosen for a particular classification Before we go on to consider parametric blends in more detail, we will give a brief overview of various blending techniques so that the reader may see how the parametric methods compare with other approaches As with any classification, some methods may not fit too well into the category to which they are assigned[†]

### Global and local blending

At the top level, we can classify blending methods as *global* or *local* A global blending method is one which affects the whole boundary of the object, rather than selected regions An example of such a technique is the method given in Reference 6 in which the whole object is outwardly and inwardly offset to produce smoothing Generally, though, most methods are local methods in which just some specified part of the object is blended

Rather loosely speaking, it can be remarked that global blends may well be preferred when CSG models are manipulated, because such models do not store faces, edges or vertices in explicit form Finding them, and also restricting the blend to a given local part of the model, may require quite some computation On the other hand, local blends are generally preferred in boundary-representation models, both because the structure is explicitly stored, and to avoid the cost of global computations Even so, blending operations which in principle are local may have global consequences if a rolling-ball blend is constructed with too large a size of ball, the ball may have unintended and unwanted interactions with faces other than the ones being blended. Such issues may necessitate global checks, even for local-blending methods

Note finally that the distinction between local and global blends is not as clear-cut as it may at first seem A global blend which affects all the primitives in a CSG model, for example, may be made local by applying it merely to a subtree of the model. On the other hand, a local method may be turned into a global one by iterating it over all the edges and/or vertices of an object as appropriate

### Edge, vertex and region blends

At the next level down, considering local blends only, we can distinguish between edge-based, vertex-based and region-based methods In the first category, edge blends basically replace a given edge between two faces by a new face which smoothly joins the two original (base) faces. Second, vertex blends are ones in which parts of three or more faces where they all meet at a common vertex are removed, and a new single face that smoothly joins the base faces replaces the vertex Edge and vertex blends are shown in *Figure 1*. The final category can be regarded as an extension of vertex blends, where now an arbitrary section of the original outer skin of the object is cut out, and a smoother piece is inserted in its place A region blend is shown in *Figure 2* Note that, after the region has been removed, the problem left to be solved (a hole to be filled with a new piece of surface) is the same as in the vertex-blend case The difference is that the region removed may consist of any combination of faces or parts of faces, together with associated edges and vertices However, in contrast, the problem to be solved in the edge-blend case is that of interpolating a new strip of surface between an opposing pair of surfaces (ignoring what happens at the ends of the edge for the moment)

It is probably fair to say that most blending methods in the literature describe edge blends, with there being rather fewer vertex- and region-blending techniques

Considering vertex blends a little further, we can see that, if some solid object is to be smoothed in a meaningful way, isolated edges or vertices are not blended, but rather whole collections of them Assuming that this is not done by using a region blend, there are two basic ways in which this process can (conceptually) be carried out In the first approach, initially, vertex blends are created at each appropriate vertex, and then afterwards edge blends are created which smooth the edges between these vertices In the second approach, the edge blends are considered as being of primary importance, and are created first, with special methods then being used at the vertices to ensure suitable transitions from one edge blend to the next In practice, the second of these approaches is much more typical Note that, as well as modifying the edge-blend construction method near vertices, it may be necessary to extend or cut back other faces which are not being blended but which meet at the vertex

Generally speaking, such topological, as well as geometric, issues relating to what to do when several edge blends meet at a vertex are quite complex Later in this survey, we review such problems and attempts at their solution Note that, to deal with such issues successfully, a full geometric model is required which stores the necessary connectivity information showing the adjacency of the faces, edges and vertices Unfortunately, the authors of many edge-blending papers only assume that they are dealing with a trimmed-face surface model, and they offer no suggestion about what to do at the ends of each edge

Finally, at this stage, we note that vertex blends correspond to the use of an $n$-sided patch to replace the region near the vertex, such patches are obviously also of use when terminating edge blends However, we do not propose to review $n$-sided patches in this paper. They have an extensive literature of their own, and the reader is referred to, for example, the survey by Hermann *et al.*[7].

## Blend representations

Finally, we can also classify blends according to the type of surface that they produce as a result On this basis, we can divide blends into *superficial*, *implicit*, *parametric* and *procedural* categories.

*Superficial blends*[8] are blends which are not explicitly represented in the modeller as a surface at all, but that rather exist as a flag belonging to a given edge which, if set, marks that edge as having been blended This is perhaps appropriate if, for example, the object is to be manufactured with a ball-ended cutter In this case, the manufacturing method creates the blend as a side effect of its operation Note that, even with such a simple method of representing blends, it is possible to use various rough methods to draw the blend, or to make approximate allowances for the blend when computing the volume of the object, for example

However, in many cases, particularly if the blend surfaces occupy more than a trivial amount of the surface area of the shape, or they have a complicated form, it becomes necessary to represent blends explicitly as surfaces in the object model, either in algebraic (implicit) or parametric form Generally speaking, then, such blends have, in principle, the same status as any other surfaces in the object model, and thus, for example, they can be intersected with other objects, the object volume can be calculated accurately, and so on However, in practice, particularly in the parametric case, the blend-surface representation used may not always be the same as that used for other surfaces in the model, and further modelling operations may be limited This is discussed at length further below

In particular, then, *implicit-blending*, and *parametric-blending* methods create, as their names suggest, blending surfaces which are implicit or parametric surfaces, respectively

As mentioned above, we will not further discuss implicit blends, but interested readers can consult the related references in the literature index References to individual parametric-blending methods are given at the appropriate places in the paper

*Procedural-blending* methods are ones in which an explicit mathematical form for the surface (such as a B-spline or quadratic equation) is not produced directly by the method, but, instead, some procedure is required to find points on the surface and to interrogate it This procedure may involve iteration, rules, or other computation Procedural-surface representations can be viewed as *indirectly* providing parametric or implicit surfaces, although not of the conventional type with definite equations and/or coefficients On the one hand, it may be possible to approximate such a surface by means of a conventional parametric (or implicit) representation Even if it is not, on the other hand, for the purposes of this review we include procedural surfaces which can be considered to be of a parametric type Examples of procedural surfaces which are described below include ones generated by partial differential equations and recursive-subdivision methods

An important practical computational consideration here is whether the blend data can be exchanged between different geometric-modelling systems or not (see for example the STEP standard[9]) The most frequent Bézier and NURBS representations satisfy this expectation, but, in other cases, some approximation to a standard format is necessary In the latter cases, further problems may emerge when numerical evaluation in the target modeller is needed Another practical point of view is the ability to compute robustly geometric interrogations and intersections The ease of computation of properties such as the convex hull and the applicability of techniques such as subdivision make certain surface types preferable to others

## Relation of blend type to base-surface type

Let us now consider the relation of the type of blending surface to the type of *base surface* (the surfaces being blended) Generally speaking, the methods which produce implicit-blend surfaces do so by starting from implicit base surfaces Although parametric base surfaces can, in principle, be implicitized, the resulting implicit surfaces are of far too high a degree (for other than very simple surfaces) for this idea to be of practical use Thus implicit blending methods are generally of little help when the original surfaces are supplied in parametric form, or when they are made up of several pieces A counterexample is worth mentioning here In References 10 and 11, the use of implicit-blending techniques for parametric surfaces is suggested In both cases, the computation of quasi distance functions on the basis of the approximate offsetting of parametric surfaces is needed (See also Reference 12)

On the other hand, at least some of the methods which produce parametric-blending surfaces can be made to work for either parametric or implicit base surfaces Even for those methods which do require parametric base surfaces, it is worth noting that many simple implicit surfaces of interest, such as quadrics, the torus and cyclides, can be parameterized, even though, in general, higher-order implicit surfaces cannot be parameterized We feel that this important point, that some methods which produce parametric-blend representations will also work for implicit base surfaces, has not been made sufficiently clear by other authors in the past, and that this relative advantage in comparison with implicit-blending methods has not been sufficiently stressed For a discussion of other relative advantages and disadvantages of implicit and parametric blends, the reader is referred to References 4 and 13

## TERMINOLOGY FOR PARAMETRIC-BLENDING TECHNIQUES

In the following section, a brief summary of the most important ideas in parametric blending is given This is felt to be necessary because different terms have been used at different times for identical or similar concepts in the blending literature Hopefully, the definitions below will help the reader to follow the forthcoming classification and comparison of the various techniques *Figures 4–7* should be consulted as this section is read

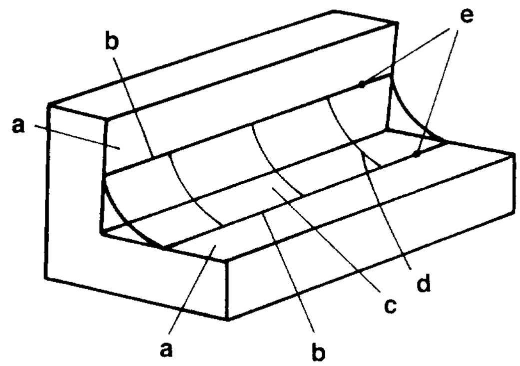



**Figure 4** Terminology, (a) base surfaces, (b) trimlines, (c) blending surface, (d) profile curve, (e) pair of points in assignment

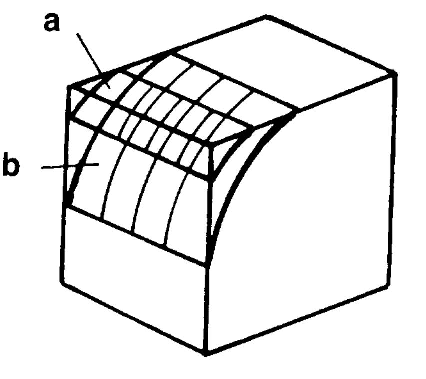



**Figure 5** Blends, (a) blend with small range, (b) blend with large range

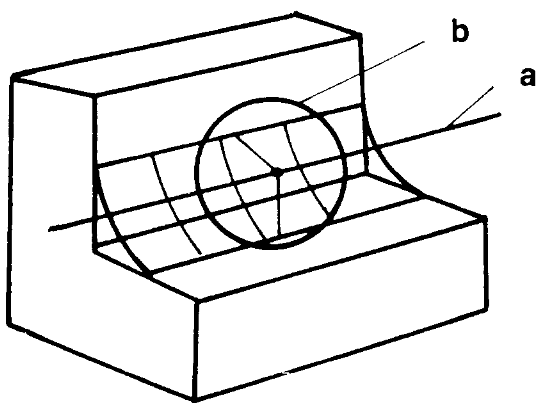



**Figure 6** Edge blending, (a) spine curve, (b) rolling ball

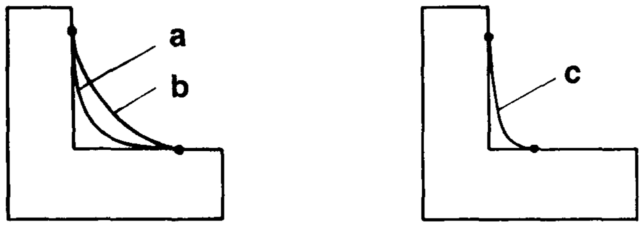



**Figure 7** Thumbweight profiles, (a) high thumbweight, (b) low thumbweight, (c) profile of asymmetric blend

The operation of creating smooth transitions between adjacent geometric elements is described using many different terms, such as 'blending', 'smoothing', 'rounding', 'filleting', and 'chamfering'. In the cases of the last three terms, as well as simple geometric meaning, manufacturing implications are sometimes (but not always) associated with these terms, which also imply the convexity or concavity of the transition surface In this paper, the term *blending* is used to cover all of these ideas Accordingly, the transition surface is simply called a *blend* or a *blending surface*

The surfaces which are to be joined smoothly, i.e the surfaces being blended, are called *base surfaces* Other terms used elsewhere are 'primitive' or 'primary' surfaces

Each curve which forms the common boundary of a base surface and the blend surface is called a *trimline* (see *Figure 4*) We use this term because the base surfaces are trimmed back along these curves Other authors use a different emphasis when discussing trimlines, using such terms as 'joining', 'set-off', 'linkage', 'contact', 'rail', 'connection curve' or 'curve of tangency' Going further, the extent of trimming back can be characterized by so-called *range* parameters or functions (see *Figure 5*) In principle, at least, such functions indicate how far each trimline is locally from the intersection curve of the base surfaces (in the case of edge blends). Depending on the actual method used, the trimlines themselves may be directly specified, or the range functions may indirectly provide equivalent information

In many cases, particularly when edge blends are generated (see the second section of the paper), the blending surface is created as a surface or volume swept along a given longitudinal trajectory, which is called the *spine curve* (see *Figure 6*) or 'directrix' Various possible choices of spine curve exist, and so, for example, it may be the intersection curve of the base surfaces or of their offset surfaces, or, more generally, it may lie in the vicinity of the base surfaces, or the spine may define some characteristic curve on the blend itself to be created At each point of the spine, a crosssectional *profile curve* (see *Figures 4* and *7*) is associated with it which locally defines the shape of the blend The profile curve is usually a planar curve, although it need not necessarily be so Alternative terms for the profile curve used elsewhere are 'blending arc', 'generator' or 'crossing curve' A profile is considered here to be a general curve, which can be constant or varying along the spine, can be symmetric or asymmetric, and can be defined as a circular or free-form arc One can control the shape of the profile by adjusting so-called 'fullness' or *thumbweight* (see *Figure 7*) parameters, or independently by changing the magnitudes of the start and end tangent vectors. The more closely the profile curve follows the shape of the original surfaces being blended, the higher the thumbweight is The terminology comes from the process in which pattern makers push filling material into a concave edge with their thumbs to create a blend manually

Profile curves can also be constructed without a spine Having two trimlines, a corresponding point pair, one point from each trimline, can be joined by a profile curve A major issue is that of how to establish correspondence between the points of the trimlines We refer to this process as *assignment* (see *Figure 4*) Assignment can be determined by using fractional arc length along each trimline, for example, or by various other geometric constructions, as is explained further below

## CLASSIFICATION OF PARAMETRIC-BLENDING METHODS

Although we restrict the subject of this review to parametric-blending methods, there is still a wide range of techniques published in the literature to consider To classify and compare these methods, the following set of considerations are taken into account

- What basic concept is used to construct the blend?
- What types of base surface can be blended using the given method?
- What sort of parametric representation is used to define the blending surface?
- What sort of continuity can be assured?
- Is the method capable of handing vertex blends?

### Basic categories of blend construction

We now consider basic approaches to the construction of parametric-blend surfaces. Note that here we will be mainly concerned with edge blends, and that most of our discussion of vertex blends will be deferred to the section on topological issues

The classification that we propose is based on the basic geometric entity which *a priori* determines the shape of the blend, and from which the other required geometric entities are derived We realize, of course, that this classification is somewhat artificial, and that, for some blending methods, it may be arguable as to which information provides the basic entities from which the others are derived Nevertheless, this approach leads to the following five categories

- rolling-ball blends,
- spine-based blends,
- trimline-based blends,
- polyhedral methods,
- other methods

In the first case, we notionally *roll a ball* so that it is simultaneously in contact with the pair of base surfaces (see *Figure 6*) In this way, a related *spine* (swept out by the centre of the ball) and two related *trimlines* (swept out by the points of contact of the ball with each surface) are generated simultaneously Both the shape of the *profile curves* and the *assignment* are determined automatically by the ball in a straightforward manner This category of blends may be subdivided into constant-radius blends, where the ball has a fixed size, and variable-radius methods, where the radius of the ball may vary In the latter case, methods may differ in how the radius function varies along the blend, and how this information is used to generate the actual surface representation

In the case of *spine-based* methods, some space curve is initially chosen as the spine, and then the *trimlines* are determined using the spine These methods can be distinguished in terms of the way in which the spine curve is defined Given a spine, several algorithms can be used to obtain the trimlines, but, in general, the points on them are automatically *assigned* according to the spine curve For example, from a given point on the spine, the current point on each trimline may be defined as the nearest point on each base surface After this step, the *profile curves* can be freely chosen, although, in most practical cases, these are forced to lie in the plane formed by the assigned point pair and the spine point

As far as *trimline-based* methods are concerned, there are many different ways of defining trimlines on the base surfaces In some cases, an 'auxiliary' *spine* is generated afterwards from the two trimlines, mainly for the purposes of *assignment* and the creation of a *profile* plane In other cases, no spine is generated, and there is additional freedom for both assignment and profiling

*Polyhedral* methods are ones in which the object is initially typically defined by a polyhedral model, and then some process is used to produce a new model by modifying the polyhedron in some way, either by producing a new, smoother polyhedron, or by replacing some of the straight edges and planar faces by curved edges and faces

Our final category of method is a catch-all category designed to include a wide variety of methods which do not fall into any of the preceding categories Methods which we consider here are based on the use of partial differential equations or Fourier methods to carry out the smoothing, or use the fact that certain surfaces, cyclides, which have implicit quartic and parametric biquadratic representations at the same time, solve various particular blending problems that occur frequently

## Base-surface type

The most commonly used surface types in geometric modelling are *implicit* and *parametric* surfaces Implicit surfaces define half spaces in the form $f(x, y, z) \leqslant 0$ If the function $f$ is a polynomial, then the implicit surface is called an *algebraic* surface Point-membership classification and ray casting can be performed on these surfaces relatively easily

Parametric patches are given in the form $\mathbf{r}(u, v) = (x(u, v), y(u, v), z(u, v))$ It is easy to generate trajectories across these surfaces for visualization, NC manufacturing etc The parametric equations are usually defined by control points which directly determine the shape of the surface and make it possible to perform convex-hull tests

A base surface can be described either by a *single* equation or in a *piecewise* (composite) manner Implicit surfaces are mostly described by one single equation, while parametric surfaces are usually given in composite form Some parametric-blending methods in this review can be applied to both algebraic and parametric base surfaces, while others work exclusively for parametric ones However, the majority of implicit-blending techniques are suitable only for implicit base surfaces, as we have already remarked

A special form of parametric representation is the so-called *indirect* or *procedural* form (see above) Normally, surfaces of this type are unlikely to be base surfaces, although they may be produced as the result of a blending operation This means that surfaces of this type can act as base surfaces if 'blends on blends' are also required

## Parametric representation of blending surfaces

A parametric surface can be considered as a mapping from a rectangular area of $u$-$v$ parameter space into 3D space In most common cases, this mapping is given in the form of a (rational) polynomial equation with known coefficients, where the *substitution* of a single $(u, v)$ parameter pair into the equation results in a 3D point In the case of indirect or procedural methods, these coefficients are not stored in evaluated form, and *iterative* or other methods must be applied to obtain a 3D point

Another issue is that of whether a *single mapping* is necessary or a sequence of mappings is required, which we refer to as *multiple mapping* In the latter case, several intermediate quantities need to be determined for a point on the surface to be calculated Multiple mapping may also involve a sequence of substitutions or iterative steps, and it is again an indirect or procedural method of surface definition

Since, in most geometric-modelling systems, the parametric representation of a single mapping with an explicit equation is preferred, procedural representations are often replaced by an approximating form of this type

## Continuity conditions

When examining continuity conditions associated with blends, one has to distinguish between *continuity across trimlines*, which describes the continuity between the base surfaces and the blending surface, and *internal continuity*, i.e. the continuity between adjacent patches of the blending surface From the very nature of blends, it is a requirement that blends join the base surfaces not only with position continuity, but in a smooth way, which requires 1st- or even 2nd-order continuity

The continuity conditions ensured are basically related to the types of base surface and blending surface The base surface is not necessarily a parametric surface, or, even if it is, the trimlines are not necessarily constant parameter lines. This makes it difficult to define parametric continuity across trimlines, although it can be meaningful for internal continuity Visual or geometric continuity is thus preferred in most cases, because it reflects the geometric features of the pieces to be connected, unlike parametric continuity, where, even if it is possible, the actual parameterization must also be taken into account

Loosely speaking, 1st-order geometric continuity ($G^1$) between two surfaces means that, for each point of the boundary, there exists a unique plane which is tangential to both surfaces being joined $G^2$ continuity means that, for each point of the connection, there exists a unique, common 'curvature distribution', i e taking an arbitrary plane which contains the joint normal vector, the intersection curve is curvature-continuous as it passes from one surface to the other at the given point These definitions can be found in a mathematically rigorous form in Reference 14, as can conditions for joining different surface elements with geometric continuity

In many cases, it is practically impossible or computationally infeasible to ensure continuity in an exact *mathematical sense* Instead, continuity is satisfied *numerically*, that is, tolerances are given to limit the extent of actual discontinuities due to the representation and numerical errors For example, taking the two boundaries of the surfaces to be joined, 'numerical' position continuity is satisfied if the corresponding points from each surface at the boundary are the same within a given position tolerance for every point of the boundary

Referring to the definition given in the previous paragraph, let us consider an arbitrary curve which runs across the boundary For strict $G^1$ continuity, this curve would be $G^0$-continuous and have a continuous tangent direction at the boundary For approximate $G^1$ continuity, this curve is now allowed to have a small tangent-direction discontinuity at the boundary, but it must be no larger than some given tangent-tolerance value Note that, in the case of approximate $G^1$ continuity, the surface may have exact or only approximate $G^0$ continuity, as shown in *Figure 8* Extending these ideas, numerical $G^2$ continuity is satisfied if the two radii of curvature (in normal sections) on each side of the boundary agree to within a given curvature tolerance

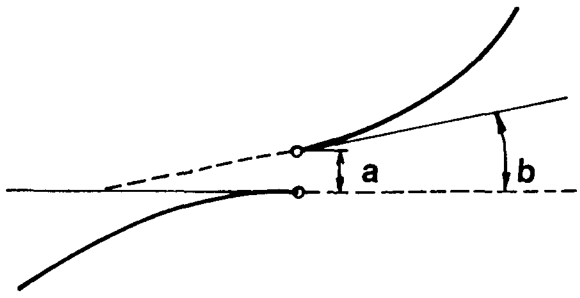



**Figure 8** Numerical continuity, (a) small positional step, (b) small angular deviation

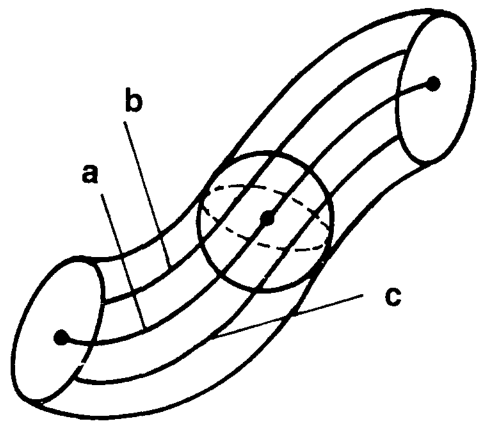



**Figure 9** Canal surface, (a) spine, (b) trimline, (c) trimline

# PARAMETRIC-BLENDING METHODS

## Fixed-radius rolling-ball blends

The main attraction of rolling-ball blending is that, since it is defined by a simple physical motion, the shape is generated in an intuitive way As mentioned above, it is also attractive from the modelling point of view that the spine, the trimlines, the assignment and the profile are 'automatically' generated. Unfortunately, the surface swept by the moving ball – a so-called *canal surface* (see *Figure 9*) – is of high algebraic degree, even in relatively simple cases[15] Owing to computational considerations, this excludes the representation of canal surfaces for practical purposes in exact form, and generally approximate methods are used. Note, nevertheless, that, at each point, the canal surface has a circular crosssection (profile), and so a natural parameterization exists in terms of distance along the spine and angle around each circular profile

As is well known, the centre of the rolling ball moves along a spine curve, which is the intersection curve of the surfaces formed by offsetting each base surface by the radius of the ball Thus, the calculation of rolling-ball blends is equivalent to offsetting and then solving a surface–surface intersection problem There are a great variety of methods of determining intersection curves While, earlier, the resulting nonlinear system of equations was solved mainly by eliminating variables, as thoroughly discussed in Reference 13, recently, the so-called dimensionality paradigm suggested in Reference 16 is gaining popularity In the former case, the number of variables is reduced, but very-high-degree equations are obtained, while, in the latter case, equations in a higher-dimensional space but of lower degree need to be solved

In the general case, iterative methods of, for example, the Newton–Raphson type are used In the authors' experience, the computation is more robust if it is based directly on quantities with geometric meaning in 3D space, instead of other algebraic quantities in some algebraic space being used These alternatives hereinafter are referred to as 'geometric' or 'algebraic' marching

Let us now consider various particular approaches to rolling-ball blending In Reference 17, general parametric surfaces are used as base surfaces, which we denote as $f(u, v)$ and $g(s, t)$ As shown below, the blends are represented in a parametric form where multiple mapping with substitution is applied These blends approximate the ideal canal surface produced by the rolling ball, and provide an additional degree of freedom to generalize the circular-arc profiles to arbitrary conics The basic steps of the algorithm are as follows

(1) By the intersection of the offsets of the base surfaces, a discrete sequence of assigned trimline points is generated in the $[u, v]$ and $[s, t]$ parameter spaces of the base surfaces

(2) By the construction of a cubic spline to interpolate each trimline point sequence, an approximate representation is obtained for the trimlines in the form $(u(\beta), v(\beta))$ and $(s(\beta), t(\beta))$, where $\beta$ is the joint parameter running along the blend to be created

(3) The profile curves are defined as rational quadratic segments with a crosssectional parameter denoted by $\alpha$ These profiles can be characterized by three control vertices $C_0(\beta)$, $C_1(\beta)$ and $C_2(\beta)$ and a thumbweight parameter, which determines the $w$ parameter of the equation below Obviously, $C_0(\beta) = f(u(\beta), v(\beta))$ and $C_2(\beta) = g(s(\beta), t(\beta))$, while $C_1(\beta)$ can be determined in a straightforward way using $C_0$ and $C_2$ and the $\mathbf{n}_f$, $\mathbf{n}_g$ surface normals to $f$ and $g$ at $\beta$ From these, the equation of the blend is the following

$$\mathbf{b}(\alpha, \beta) = \frac{(1-\alpha)^2 C_0(\beta) + 2w(1-\alpha)\alpha C_1(\beta) + \alpha^2 C_2(\beta)}{(1-\alpha)^2 + 2w(1-\alpha)\alpha + \alpha^2}$$

Note that this parametric representation of the blending surface assures exact $G^0$ continuity Because of the approximate nature of trimlines, only an approximate definition of the canal surface is obtained, and actually the blend meets the base surfaces with numerical $G^1$ continuity only Internal continuity of the blend is ensured by the cubic-spline interpolation method and the continuity of the base surfaces The construction in Reference 18 shows many similarities to the one above It is described in the trimline-based section below

The rolling-ball algorithms of References 19–21 and recently References 22 and 23 have mostly been developed independently, but they have several similar features In all cases, parametric approximating surfaces are generated which join the base surfaces with numerical position and tangent continuity In Reference 19, a 'geometric-marching' method is suggested that is applicable to the blending of general implicit and parametric surfaces A sequence of consecutive ball positions is generated which results in a sequence of blending arcs to be approximated by the blending surface Similar geometric iteration techniques can be used for the special cases arising at the ends of blends The blend surfaces are represented by a sequence of double-quadratic or bicubic patches fitted in an adaptive manner[4] In Reference 23, a more rigorous description of intersecting offset parametric surfaces is given, and the approximating surface is represented by a NURBS

Similar geometric marching methods are described that are based on parametric base surfaces in References 20–22 in which the computation of the tangents to the trimlines is also required at each ball position This is meant to improve the efficiency of the next approximating step There are variations in how the approximation is performed, in whether a proper step length is estimated, and in whether further sampling must be carried out to satisfy the prescribed tolerances In References 20 and 22, bicubic surfaces are generated A particular feature in Reference 21 is the use of the so-called RBG (rational-boundary Gregory) patches These are bounded by rational parametric curves, and independent cross-derivative functions in the $u$ and $v$ parametric directions are also allowed Note that RBG patches can represent circular-arc profiles exactly, unlike the other methods, in which approximating cubic arcs are used In Reference 22, details about overcoming certain self-intersection problems can also be found

As has been reported recently in Reference 24, the ACIS solid modeller uses a natural method for constructing rolling-ball blends The primary representation is an approximating spine curve, which holds pointers to the original primitive surfaces to be offset and intersected If necessary, a point on the spine can be determined as accurately as is required Simultaneously, an approximating canal surface is also created which provides a basis for interrogating and intersecting the blend surface Another interesting feature is the way in which ACIS incorporates the blend in the solid model A sheet object is created at first on the basis of the individual components of the blend, i e the sequence of edge and vertex blends In the next phase, the sheet object is combined with the original object by a subset of the steps required for performing Boolean operations

An interesting rolling-ball method was suggested in Reference 6 which produces surfaces which can easily be parameterized. The original CSG concept with classical Boolean operations is extended by the introduction of a special offsetting operator for solids The combination of offsetting inwards and outwards provides a method for blend definition The rolling-ball blends which are generated as a result are represented in approximate form as a sequence of sphere, cylinder, and torus pieces, and thus no new types of geometry other than the original set of solid primitives are required

## Variable-radius rolling-ball blends

The physical analogy of the rolling ball works well in the constant-radius case, and its advantages have led to the basic idea being extended These extensions are aimed at increasing the flexibility of this method of blending as a design tool, and allowing it to cover more practical cases Note that the parameters of blending can be varied in a number of ways Not all variable-blend or variable-radius-blend methods use the rolling-ball analogy, and so not all of them are dealt with in this section

With the extension to variable-radius rolling-ball (VRRB) blends, new problems arise. An obvious question is how far constant-radius techniques and thinking remain valid. It is not easy to determine the way in which the radius should change along a blend, and this problem is compounded by the desire for a compact mathematical formulation Several new concepts are required when taking a variable-radius approach

The centre of the rolling ball now has to move on the surface equidistant from the base surfaces, which is called the Voronoi surface. A point on the Voronoi surface determines both the position and the radius of a ball touching both base surfaces Choosing a spine curve on the Voronoi surface is, in some ways, the best method of defining a variable-radius rolling-ball blend

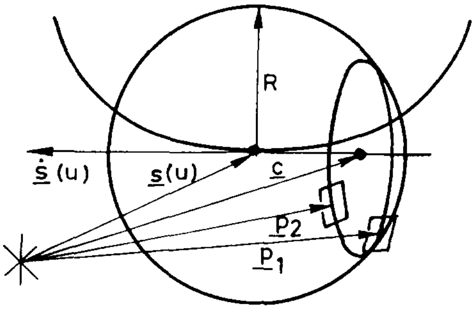



**Figure 10** Kinematic model of variable-radius rolling ball

Pegna analysed the variable-radius rolling-ball blend in his thesis[25]. *Figure 10* shows how to construct that circle on the ball which is simultaneously tangent to both base surfaces and that contributes to the variable-radius canal surface In this case, the spine curve $\mathbf{s}(u)$ and the radius function $R(u)$ are supposed to be given, and the base surfaces are indicated by their tangent planes at $\mathbf{p}_1$ and $\mathbf{p}_2$ It should be noted that, in the case of blending between two given surfaces, $R(u)$ and $\mathbf{s}(u)$ cannot be chosen arbitrarily Further investigating this model, Pegna and Wilde[26] found that the condition for the ball to contribute to the canal surface at all is

$$|R(u)|/|\mathbf{s}(u)| \leqslant 1$$

(Note that a similar but signed restriction seems to be needed for $\ddot{R}(u)$ as well, to avoid self intersection ) The contributing circle is centred at

$$\mathbf{c}(u) = \mathbf{s}(u) - \mathrm{sign}(\dot{R}(u))(1 - (\dot{R}(u)/|\dot{\mathbf{s}}(u)|)^2)^{1/2}\dot{\mathbf{s}}(u)/|\dot{\mathbf{s}}(u)|$$

However, Pegna does not explicitly consider the Voronoi surface at all Instead, the basic concept of Pegna's VRRB method is that a general 3D space curve $\mathbf{s}^*(u)$ is created initially as a spine curve, and one of the base surfaces is chosen as a reference surface to determine the radius $R^*(u)$ as the distance to this surface Then, $\mathbf{s}^*$ is adjusted so that it becomes equidistant between the two base surfaces This equidistant curve $\mathbf{s}(u)$ defines the VRRB blend uniquely, as suggested by *Figure 10* The method does not depend on the type of base surface, but it requires procedures for computing the projection of a point onto a surface (see details in Reference 27), and for performing the above adjustment of $\mathbf{s}^*$ pointwise The procedural parametric representation is the $\mathbf{s}(u)$ curve itself, together with the algorithm Quite independently of this method, but within the scope of blending, Reference 28 also investigated conditions for assuring the curvature continuity of parametric surfaces

Chandru, Dutta, and Hoffmann were the first to introduce and explicitly use the concept of Voronoi surfaces in References 29 and 30 In the previous VRRB blend, the centre of the variable-radius rolling ball moves on the Voronoi surface along the spine curve of the variable-radius canal surface In the current method, to define the spine, the Voronoi surface of the base surfaces is intersected with some reference surface chosen for this purpose This approach leads to a complicated system of equations As mentioned above, the dimensionality paradigm[16], also originating from the same authors, and devised for complex geometric problems of this type, offers the possibility of solving the system in higher-dimensional spaces numerically, instead of eliminating variables The authors also consider another numerical approach in which the spine is approximated by double elliptical arcs, which become spines of cyclide patches approximating the blending surface Using the special feature of cyclides by which they can be connected smoothly along their circular lines of curvature, a $C^1$-continuous piecewise cyclide surface can be defined in this way The continuity across trimlines is numerical The method does not impose strict requirements on the base surfaces, except that it must be possible to set up the systems of equations describing the Voronoi surface The cyclide pieces generated have compact formulations both as implicit and parametric surfaces An interesting extension of the Voronoi-surface concept is that of the skeleton of an object[31], which, in connection with blends, may be useful for solving topological problems

Choi suggests an extension to his previously mentioned rolling-ball blending method for variable-radius cases[32] Currently, he makes the radius steadily change in a monotonic way between the two radii chosen by the user at the ends of the blend, although he notes that his method works for other cases too The trimlines of an average-radius rolling ball are constructed first, and they are then adjusted pointwise within the normal plane of the constant-radius spine so that a circular arc in this plane touches both surfaces at the new trimline points The radius of this circular arc has the appropriate intermediate value, and this arc locally defines the blending surface This method can be regarded as a practical way of generating variable-radius blends, but the surface generated is not a close approximation to any particular variable-radius rolling-ball blend surface It is an advantage of this method that it allows one to define the blend easily

## Spine-based blends

Blending methods which use a spine curve as *a priori* information can be arrived at from several points of view In one case, we start from the idea that the parts of objects that we want to replace during blending are basically edges The blending surfaces generated from them should follow the direction of this edge, and so the blending-surface definition should be derived from the edge, which is thus used as the spine curve Starting from the idea of sweeping also leads to a spine-based method The directrix becomes the spine curve here, while the concept of the generatrix can be extended to a more general profile definition Extensions of rolling-ball blends also lead directly to spine-based methods Note that rolling-ball methods themselves, especially the variable-radius ones, can also be regarded as a subclass of spine-based methods.

A spine-based method generally proceeds as follows First, some spine curve is defined One possibility is to use a 3D-space curve, which allows plenty of freedom, but it may be rather awkward for the user to define the spine in this form Alternatively, the spine curve can be defined as the intersection curve of some surfaces instead Two obvious possibilities to use are the pair of base surfaces, or their pair of offset surfaces Finally, the spine curve may be defined as some curve lying on the Voronoi surface of the base surfaces Such a curve can be used to define both general spine-based blends and variable-radius rolling-ball blends

The next step is to generate trimlines, or at least points lying on notional trimlines, on the base surfaces Thus, this can be done pointwise or by using marching algorithms At the same time, tangents along the trimlines can be calculated either exactly or numerically, if they are required Usually, an assigned pair of trimline points is derived simultaneously from a single spine point, and so the assignment problem here is solved automatically

The third component of these methods is the definition of the profile curves, which, generally speaking, can be constructed using any suitable method

Pegna's approach illustrates well the close connection of this type of method to the VRRB methods As shown in *Figure 10*, those arcs between $\mathbf{p}_1$ and $\mathbf{p}_2$ which are centred at $\mathbf{c}$ define the VRRB blend that we have already considered, whereas the slightly different blend generated by the arcs between $\mathbf{p}_1$ and $\mathbf{p}_2$ centred at $\mathbf{s}$ is no longer a rolling-ball blend, but is rather just some more-general spine-based method Apart from this minor difference, the basic concept and the algorithm used are the same Comparing the two methods, Pegna states that the spine-based version produces a smoother transition between the base surfaces than does the variable-radius rolling ball

The overall structure of the method of Harada, Konno, and Chiyokura[33] shows some similarity to that of the variable-radius blend of Choi[32] The basic differences are that they introduce the concept of the spine explicitly, calling it the control curve $\mathbf{R}(t)$, and allow more general functions for both $\mathbf{R}(t)$ and the $r(t)$ radius function First, the 3D spine curve $\mathbf{R}(t)$ is generated or designed, and $r(t)$ has to be specified also Then, circular arcs of radius $r(t_i)$ are constructed which touch both base surfaces in the planes normal to $\mathbf{R}(t)$ at the $t_i$ parameter values Finally, these circular arcs are approximated by quartic segments, and those are interpolated by biquartic Gregory patches to generate the blend surface In this process, exact trimline tangents at the corners of the patches are also calculated and interpolated The method approximates the surface swept out by the sliding variable-radius circle in this way This parametric representation makes it possible to ensure exact $G^1$ continuity between the patches of the blending surface

In References 12 and 34, a special spine-curve generation technique is presented which is applicable to any parametric surface and can also be useful for blending quadrics Instead of offset surfaces being intersected, quasioffsets are defined and then intersected The spine points are in tight correspondence with trimline points on the base surfaces, thus determining the assignment Freedom of choice for profile information is also available The quasioffsets for quadrics are quadrics themselves, and consequently the complexity of spine-curve generation is reduced to that of quadric–quadric intersection

## Trimline-based blends

Blending replaces parts of base surfaces with blending surfaces One obvious way of specifying such an operation is to decide explicitly which parts are to be substituted by choosing where the trimlines should lie on the base surfaces Other necessary information is then constructed using the trimline geometry.

Trimlines can be defined in several ways One possibility is to determine them as intersection curves, for example, of pairs of base and offset surfaces In the case of a parametric base surface, the trimline can be designed in the parameter plane Methods which retain the 2D parameter-plane representation as the final description of the trimlines, and represent the blending surfaces in terms of them, generally lead to indirect- or procedural-blending surfaces

Once a pair of trimlines has been chosen, there are basically two possibilities for the next step Either a spine curve is used to choose corresponding points on the trimlines to be assigned together, for example by finding where the local normal plane to the spine curve meets each trimline, or assignment is done without the aid of a spine. The spine curve can, if required, be defined by using either the two trimlines, or any other method

The final important phase of trimline-based methods is some kind of method of generating profile information that makes it possible to define the profile curves which connect assigned pairs of trimline points and contribute to the blending surface. A number of methods use approximate circular-arc profiles, or derive profile definitions as generalizations of circular arcs A thumbweight parameter and an asymmetry factor, or ranges (tangent-vector magnitudes), possibly varying along the blend, seem to be sufficient for defining the profile in most practical cases

An early commercial trimline-based solution was provided by DUCT[35] This system defines a duct as a lofted parametric surface, and allows the creation of a connecting duct between a 'main' duct and a 'branch' duct. The curve bounding the end of the branch duct is used simultaneously both as a trimline and as a spine for generating the blending duct The blending surface is defined as a series of circular arcs of appropriate radii such that they (a) lie in the normal plane of the trimline of the branch duct, (b) meet the branch duct tangentially across the trimline, and (c) also meet the main duct tangentially As an alternative, if each circular arc in the above construction is replaced by a circular arc of fixed radius joined to a straight-line segment, a connecting duct can be created which consists of a ruled extension to the branch duct connecting smoothly with a rolling-ball blending surface

The method published by Bardis and Patrikalakis[36] generates trimlines by taking the intersection curve of the base surfaces and offsetting it across each of the base surfaces along geodesics Arbitrarily designed trimlines are also allowed The trimlines are first represented as NURBS curves in the 2D parameter plane of the NURBS base surfaces, and then they are converted into (nonrational) cubic B-splines in 3D If both the trimlines have been generated by geodesic offsets, they are put in assignment by correspondence with points of the original intersection curve In some cases, or in the case of more-general trimline-generation methods, reparameterization is required to ensure an appropriate assignment Finally, a bicubic (nonrational) B-spline blending surface is fitted that interpolates the position and normal vectors of the base surfaces at discrete trimline points Numerical $G^1$ continuity is achieved across the trimlines This method is a successor to that of Hansmann[37] in which parameterized analytic surfaces are also considered as base surfaces, although only designed trimlines are used

The above-mentioned Reference 37 was the first publication that investigated how to construct curvature-continuous transitions In a recent publication[38], numerically curvature-continuous NURBS blends are generated The two trimlines are defined in parameter space or procedurally, and then their 3D approximations within a specified tolerance are generated. These are represented as integral B-splines whose two knot vectors need to be merged during surface construction. The profile curves have six control points set in such a way that they match the normal curvatures computed at appropriate points of the base surfaces along the trimlines It is also possible to adjust the fullness of the blend. Assignment is carried out by a so-called directional curve The profile curves are lofted to obtain the final B-spline Error checks for numerical tangent-plane and curvature continuity are performed halfway between each consecutive pair of nodes, if any of these are unsatisfactory, a new knot value is inserted into the longitudinal knot vector of the B-spline.

Although Varady, Martin and Vida suggested a spine-based trimline definition[4], their solution was fundamentally trimline-based The trimline points are determined as intersections of the base surfaces with circular arcs lying in normal planes of the edge and centred on the edge The radius can vary, either producing a pseudo-rolling-ball effect, or, in principle, arbitrarily along the edge Other methods leading to 3D trimline curves on the base surfaces can also be applied. Either the normal plane of the edge is used to assign points on the trimlines, or approximate fractional arc lengths along the trimlines are used instead To define the surface, a set of control polygons of crosssectional curves is generated, and then the process of fitting approximate surfaces is reduced to the adaptive, simultaneous approximation of longitudinal curves (see *Figure 11*) The base surfaces can be either implicit or parametric ones, and the blending surface is represented by the explicit parametric approximation Numerical $G^1$ continuity is ensured across the trimlines Consecutive patches along the blend may be connected with either exact or numerical $G^1$ continuity The associated trimline-based

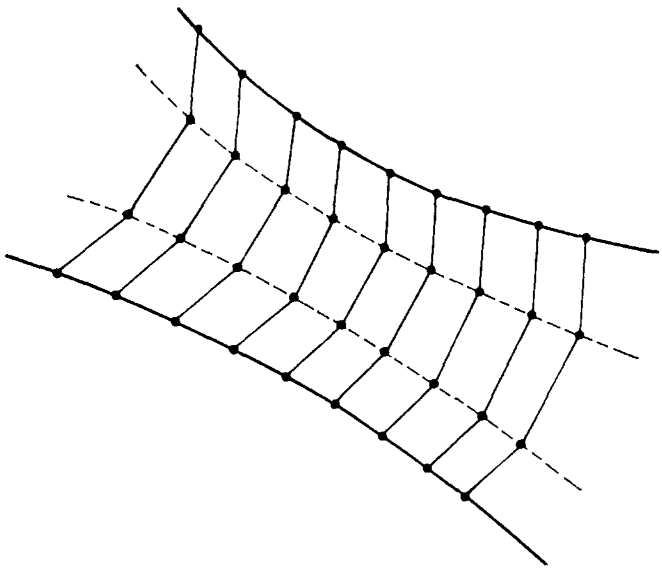


**Figure 11** Fitting a surface reduced to fitting four curves

topological model[4,39] is considered below in the section on topological issues Any parametric curves satisfying the end constraints can be used as profile curves, the authors report an implementation in which both the curves and surfaces used are double quadratics[40]

Filip[18] generates procedural-blending surfaces based on trimlines The trimlines are either parameter lines or designed and interpolated curves in the parameter plane Typically, he uses cubic B-spline curves and bicubic B-spline base surfaces, but the method does not depend on the type of the parametric representation The trimlines are assigned by using the same parameter interval for both of them The equation of the blending surface $\mathbf{B}(s, t)$ is defined in the form of a cubic span

$$\mathbf{B}(s, t) = H_1(s)\mathbf{C}_1(t) + H_2(s)\mathbf{C}_2(t) + H_3(s)\mathbf{T}_1(t) + H_4(s)\mathbf{T}_2(t)$$

where the $H_j(s)$, $j = 1 \ldots 4$, are the cubic Hermite polynomials, the $\mathbf{C}_i(t)$, $i = 1, 2$, are the trimlines, and the $\mathbf{T}_i(t)$ are the tangent vectors of the span The direction of $\mathbf{T}_i(t)$ is determined by projecting $\mathbf{C}_2(t) - \mathbf{C}_1(t)$ into the tangent plane of each base surface at $\mathbf{C}_i(t)$ The remaining two scalar degrees of freedom of the profile are determined by using a circular-arc approximation principle and an asymmetry factor for the profile curves Exact $G^1$ continuity across the trimlines is ensured by the procedural method, and $G^2$ continuity is also studied in terms of beta constraints For the blending surface to be $C^1$-continuous, the base surfaces must be $C^2$-continuous, and the trimline equations in the parameter space must be $C^1$-continuous This is necessary because the blending surface is expressed using derivatives of the base surfaces in a parameterization-dependent way

One variation of Filip's method was suggested in Reference 41, in which the $T_i$ functions were explicitly derived as a linear combination of the related $u$ and $v$ derivatives in a quasiorthogonal direction to the trimline curves This makes it possible to assure $C^1$ continuity to the base surfaces at the sample points, but the profile-curve control becomes somewhat more restricted

An original method of assignment is suggested by Koparkar[42,43] His approach uses 'fanout' surfaces, i e ruled surfaces which are normal to the base surfaces along the trimlines (see *Figure 12*) The spine is defined as the intersection curve of the two fanout surfaces A spine point puts in assignment the two trimline points whose surface normals meet at the given spine point The profile planes are determined by the triples of the spine and the corresponding trimline points The trimlines are represented in the parameter planes of the base surfaces, and a procedural-blending surface is defined which is assumed to be joined to the parametric base surfaces with $G^1$ continuity

In unfortunate cases, the fanout technique shows certain deficiencies, as was reported in Reference 41 As an alternative, the construction can also be performed using a pair of ruled surfaces, called canopy surfaces Each canopy surface is constructed so that it is tangential to the base surface along the corresponding trimline, the rulings of this surface are straight lines which lie in a quasiorthogonal direction to the trimline at each point The intersection of the pair of canopy surfaces determines the assignment of the trimlines, as each point of the intersection curve arises from a pair of generators, one for each ruled surface These can be traced back to a single point on each trimline

Bien and Cheng[44] reduce the blending of parametric base surfaces to the mixing of surfaces defined on a shared rectangular parametric domain by using blending functions In their method, a typical parametric (e g curve or surface) equation is defined as $\mathbf{F}(\mathbf{u}) = \sum_i b_i(\mathbf{u})\mathbf{P}_i$, where the $\mathbf{P}_i$ are control points and the $b_i(\mathbf{u})$ are blending functions Replacing the control points by parametrically defined objects $\mathbf{Q}_i(\mathbf{u})$, a more general surface equation $\mathbf{G}(\mathbf{u}) = \sum_i b_i(\mathbf{u})\ \mathbf{Q}_i(\mathbf{u})$ is derived which is a procedural definition according to our current terminology They use this concept in two ways in their blending method On the one hand, reparameterized rectangular domains of parametric surfaces can be mixed on the basis of this

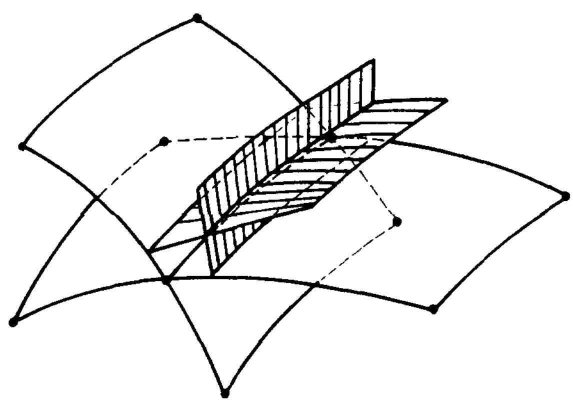


**Figure 12** Spine defined as intersection curve of fanout surfaces erected along trimlines

idea, and this process can produce a blending surface. On the other hand, however, it cannot be generally expected that the trimlines should be parameter lines To overcome this problem, a 2D parametric patch ($\mathbb{R}^2 \rightarrow \mathbb{R}^2$) is defined in the parameter plane of each base surface, to force the trimline to become a parameter line The assignment is solved by the shared parameter values, automatically, and the profile is formed according to the current blending functions. The chosen degree of the Bernstein polynomials used in the blending functions determines the continuity across the trimlines, from the procedural-surface formulation The method can also be extended to the case in which three base surfaces meet at a corner

## Polyhedral blends

There are several possible ways in which we can start from some polyhedral definition of an object and produce a new object which is blended ('blending' here really refers to the fact that an object smoother than the original polyhedron is produced as a result). Two basically different types of *polyhedron-based* methods are distinguished here

Taking the original polyhedron as defining the overall approximate form of the shape required, the first approach constructs a sequence of polyhedra, the new polyhedron at each step having more faces, edges and vertices, where the faces and edges in each successive polyhedron become smaller. The new geometric elements are constructed from the existing ones in such a way that each new polyhedron is of a shape similar to that of the previous one, but is smoother These are called *subdivision* methods. Subdivision blending surfaces formed by taking the infinite limit of the subdivision process can be expressed in explicit parametric form, or can at least be approximated by parametric patches.

The other group of methods builds a *wireframe* model of the object first (which is typically, but not necessarily, of polyhedral shape), and then substitutes some of the original edges in it with new edges lying on profile curves and trimlines which surround the regions to be blended. Finally, the whole wireframe model is covered by patches or composite free-form surfaces, and a blended solid is defined in this way. As noted, the simplest wireframes originally contain only straight edges, which is why this method is also referred to as being polyhedron-based

Let us start by considering subdivision methods As an extension to a rendering technique, Catmull[45] invented a subdivision method which is suitable not only for generating polyhedral approximations to B-spline surfaces, but also for designing free-form surfaces as well Doo also published similar work at about the same time[46] A practical implementation of these ideas is described by Veenman[47] Techniques are described in these papers which show how the generation of bicubic or biquadratic surfaces on quadrilateral meshes can be extended to polyhedra of arbitrary topology. In this way, the bulk of the surface is covered by rectangular patches, while a finite set of extraordinary points remains, where tangent-plane and curvature continuity ae not ensured in general by the method Doo and Sabin[48] showed analytically that there are generally discontinuities at extraordinary points in the Catmull–Clark method, but that they can be avoided by a particular choice of the weightings in the subdivision formulae in the quadratic case Ball and Storry[49] extended this work further to show in general what choices of weighting would give tangent-plane continuity at the extraordinary points They also showed[50] that it is not possible to choose the weightings to ensure curvature continuity too, but they did show how to choose the weightings so that the jumps in curvature at extraordinary points are minimized

Other recursive-subdivision work includes that by Nasri[51], who described how to convert the original polyhedron into a modified one that is capable of interpolating given points or lines after subdivision He also considers the problem of intersecting subdivision surfaces On the other hand, Dyn *et al*[52] give a recursive-subdivision method for polyhedra bounded by triangular facets which produces a $C^1$ surface which directly *interpolates* the original polyhedron vertices. The earlier methods produce new surfaces which do not have this property, which is why Nasri has first to construct a modified polyhedron Dyn *et al* also provide a tension parameter to control the shape of the final surface Tan and Chan[53] show how subdivision methods can be used to produce higher-order (than cubic) $C^1$ surfaces, and they suggest using the surface order to control the amount of smoothing provided. They are able to guarantee tangent-plane continuity at extraordinary points by using a method of subdividing the parameter space and adjusting the mesh points at each level of subdivision

Finally, it should be mentioned that Storry and Ball[54] show how recursive-subdivision patches can be used to solve the problem of producing a vertex blend in the special case of a bicubic surface containing an *n*-sided hole The blend has $C^1$ continuity with the rest of the shape, one internal point is used to control the shape of the blend

Unlike the subdivision methods, the wireframe methods do not define the blended shape as the limit of a process Instead, they create it by deforming the original wireframe, or otherwise replacing some of the straight edges by curved ones, and by interpolating the resulting curve network with smooth surfaces. This initial curved-wireframe representation includes topological information, and is intended to describe a solid without determining the bounding surfaces

The method of Chiyokura and Kimura[55] for designing free-form solids consists of three phases First, a polygonal *wireframe* is created by using local operators that are similar to Euler operators This model contains straight edges, which are expected to surround convex but not necessarily planar faces Edges of this polygonal structure are then marked as being required to be rounded off or not From this information, new curved and straight edges are created which surround the regions around the edges and vertices to be blended We can call these curves trimlines, using our terminology, as they bound the blend faces These new edges (cubic segments) depend on the average positions of the vertices along each edge and surrounding each face, and so the range of the blends created depends on the sizes of the original faces Finally, the curved wireframe model that represents the blended object and which may contain $n$-sided face regions is covered with surfaces using Gregory patches to complete the solid model

Again starting from a polyhedron, Fjallstrom introduced a weighting method[56] to control the shape of the blend in a more sophisticated way, without changing the polyhedron Instead of a flag, he uses a thumbweight parameter He also introduces original methods for fitting convex combination linear and higher-degree Taylor interpolants to the $n$-sided regions which are of general interest as $n$-sided techniques for vertex blends

Chiyokura[57] later extended his method to allow some curved edges to be included in the original wireframe before the specified roundings are evaluated Other new methods of specifying blends allow the user to choose radii or distances along the neighbouring edges at the ends of each edge to be blended, and they thus allow control over the range of the blend independently of the size of neighbouring faces

Pratt[5] and Beeker[58] published methods whose underlying philosophies are, to some extent, similar, both to each other and to the wireframe methods described above Pratt's suggestion is based on B-rep solid thinking Initial simple solids are bounded by planes and possibly simple quadric faces usually represented in implicit form These may also be described using simple parametric representations, together with a control polyhedron structure If these lower-order surfaces are regarded as degenerate cases of higher-order surfaces, this control polyhedron can be modified to provide a smooth transition to neighbouring faces while also meeting other constraints Remaining degrees of freedom can also be used to give the user some control over the process Beeker[58] creates a quadrilateral polygonal wireframe structure first, and regards it as a control polyhedron of Bernstein patches Then, he allows the user to modify the control points interactively, while applying continuity constraints between selected adjacent patches One last similar simple method for defining objects bounded by planes and bicubic blending surfaces is described in Reference 59

## Other methods

Our final class covers a range of unrelated methods suggested for solving the blending problem Under this title, we include blending methods which use cyclides for special cases of blends, and methods which solve partial differential equations to obtain blending surfaces, and which remove high frequencies from Fourier transforms of object boundaries to obtain smoother shapes

Cyclides are important as a method of blend construction, particularly as, among the cases they can handle, they provide a solution for some frequently occurring simple blends, such as a cylinder obliquely meeting a plane Such constructions, however, do not generally provide enough degrees of freedom to represent constant-radius rolling-ball blends, which are of practical importance for many applications The *special-cyclide* solutions published by Pratt and Boehm use several features of cyclides (see, for example, Reference 60), some of which are especially important when blends are modelled Cyclides can be regarded as generalizations of the torus and can be described by implicit quartic equations or in parametric form using trignonometrical parameterization or rational biquadratic Bézier equations Other descriptions of similar complexity (in terms of geometric constructions) are also possible An important feature of cyclides is that they have circular lines of curvature (see *Figure 12*), along which the angle between the planes of these circles and the surface normals is constant Thus, any cyclide can be cut at and rotated about these lines of curvature, and can be connected with tangent-plane continuity to some other surface along a circular boundary On the other hand, when used for blending, the required cyclide pieces can easily be constructed by determining circles on them, especially if these lie in the planes of symmetry

Pratt[60–62] presented ways of constructing single-cyclide blends between a cyclide and a sphere, between a cyclide and a plane, and also between a pair of cones that both

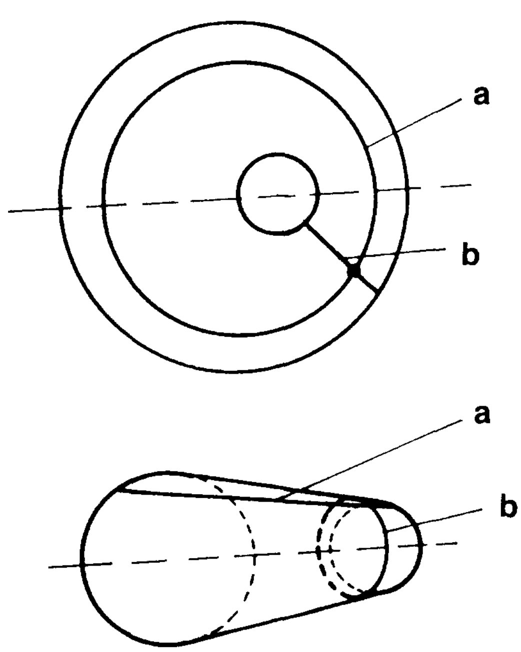



**Figure 13** Lines of curvature of cyclide

touch a common sphere Note that cyclides also include tori and the natural quadrics, i e planes, cylinders, cones, and spheres. (Planes and spheres are special cases, because they have infinitely many circles at every point which all possess the above-mentioned angle property, while points of more general cyclides satisfy this property only along the two lines of curvature) A blend construction using a pair of cyclides can be derived from a blend using a single cyclide section by dividing the section along one of its lines of curvature, and rotating one of the resulting pieces relative to the other piece about the axis of the circular line of curvature In the case of blending two cylinders of different radii, however, a direct double-cyclide construction is needed. Pratt also suggests a method of extending his ideas to more general cases, creating trimlines based on the angle property of lines of curvature on both base cyclides This approach, however, cannot solve, for example, the symmetrical torus–cone blend on the well known Cranfield object[63]

Boehm[64,65] uses a different terminology, and extends the coverage of known special constructions, giving a double-cyclide blend of two cones, a solution for the Cranfield-object problem, and another one for three cones touching a sphere Note that the latter two solutions are really quite specific, as they also contain pieces of planes and spheres to perform the construction, and are probably not of much help in finding other general cyclide constructions Although cyclides can provide simple blending methods in some special cases, no more-general construction algorithm has yet been presented.

The *partial differential equation* (PDE) method suggested by Bloor, Wilson, and others[66–70] views blending as a boundary-value problem A boundary of a domain together with some derivatives at it are specified, and a smooth solution is required over that domain The authors of these papers suggest that this can be achieved by constructing an elliptic PDE, the solution of which will be the blend. They define a 2nd-order differential operator as

$$D^2=\left(\frac{\partial^2}{\partial u^2}+a^2\frac{\partial^2}{\partial v^2}\right)$$

and seek for the solutions of equations $D^2\mathbf{r}(u,v)=0$, or $(D^2)^2\mathbf{r}(u,v)=0$, or $(D^2)^3\mathbf{r}(u,v)=0$, according to whether position, tangent-plane or curvature continuity is required across the boundary (trimlines). The parameter $a$ controls the shape; it plays the role of thumbweight in the examples they give. Simple examples can be solved analytically, while more-general cases require numerical treatment. B-spline patches can be fitted to the result of such a numerical method The PDE method is felt to be a powerful means of surface generation One shortcoming is that the question of parameterization and assignment is not dealt with in these papers, while the result seems to depend strongly on how these are chosen. It seems that the PDE method does not really offer more functionally than other trimline-based methods described in this paper, but requires more computation

The *Fourier*-based approach proposed in Reference 71 is a region-blending method The basic concept is that some region of the surface of the object is chosen, and then point samples are taken from that region and placed into a rectangular 2D array The Fourier transform of the point set is then taken, and then the high frequencies of this data are attenuated Finally, an inverse Fourier transform of the data is computed to produce a new set of points in real space Because high frequencies correspond to regions of rapid change in the shape (such as occur at edges and vertices), this attenuation process smooths the shape

In practice, choosing the attenuation function directly in Fourier space is not a good idea[72], and so the idea is better expressed as a convolution, which corresponds to using the sample points to construct splines Either a discrete convolution can be performed which produces a new point set (which may be useful for viewing the results of smoothing, for example), or a continuous convolution can be performed which produces a parametric-surface representation of the smoothed region as a spline surface

Various advantages are expected from this technique On the one hand, unlike the Liming-type implicit methods, surface degrees do not increase if repeated blending of blends is performed On the other hand, as sampling is done, the amount of data does not increase either when blends on blends are constructed

However, it should be remarked that, although limited success has been obtained with constructing blend surfaces which are smoothly connected to the rest of the original object, serious topological issues remain to be solved before this method can be considered to be of widespread applicability The major issue is that of how sensibly to sample the point data from an arbitrary region of the surface of an object Other work on the use of Fourier methods for blending is described in Reference 73, broadly similar ideas are to be found there

A related approach which also uses convolutions for smoothing is described in Reference 74, although a 3D rather than 2D convolution is used The object is taken, and a sphere of a given size. The sphere is then positioned at a series of points (chosen by the user, lying on rectangular grids) around the boundary of the object, and adjusted until the sphere is half full of solid and half empty The final position of the centre of the sphere gives some point on the surface of the smoothed object. In practice, numerical integrals are computed together with iterative adjustment of the position of the sphere along the surface-normal direction to determine each new surface point, which is a very slow process. Finally, the author claims that parametric patches can be fitted to the resulting point data, although no description is given of how this can usefully be done in practice

## JOINING AND TERMINATING BLENDS: TOPOLOGICAL CONSIDERATIONS

Edge blends give rise to a problem which generally does not occur when simple isolated vertex or region blends are carried out This problem is that each edge has two ends, and so every edge blend has to be terminated in some way Unfortunately, many authors of papers on edge blending only discuss how to construct surfaces along the length of the blend, and they ignore the issue of what happens at the ends This is a serious problem, as the geometric methods used for the main part of the blend may not work at the ends, because of the need to take into account faces other than the two base surfaces Instead, we may need to create vertex blends at the ends, or use some other method to ensure a satisfactory shape for the ends of the edge blend

It should be noted that, to terminate edge blends, the topology of the geometric model needs to be considered, as well as the local geometry The topological restructuring along the length of the blend is straightforward, as an edge is replaced by a new face (the blend) and two new edges (along the trimlines) However, the topological restructuring of the model at the ends is typically rather more complex, and indeed a variety of topological (and geometric) solutions may be necessary to cope with the wide range of configurations that may be present at the ends of the edge being blended We first discuss approaches to this issue, and then review specific pieces of work in this area Further discussion of the source and nature of these topological and termination issues can also be found in Reference 4

We base our blend-termination discussion around the idea of trimlines Even if the method of construction of the edge blend is not directly based on them, they are eventually created as the new edges between the blend surface and the base surfaces We start by considering what to do in the case of a single edge blend, and we then proceed to consider the case in which several edge blends meet at a vertex Let us first, however, see why vertex- and region-blending methods do not generally pose problems of this type

### Vertex and region blends

The topological issue is almost trivial in the case of vertex or region blends Typically, a ring of 'trimlines' is drawn around the vertex or region to be blended showing where existing geometry is to be removed and replaced, and then some new geometry is constructed to fill the hole This smoothly meets the surrounding faces with a certain continuity A complete ring of trimlines is given as part of the specification of the blend, existing faces only need to be cut back (and not extended), and usually only a single face is inserted. If the method uses more than one face to fill the hole, as does recursive subdivision, then the method itself generally provides a (straightforward) specification of how the new faces are topologically (as well as geometrically) related Thus, there are no difficult topological problems to solve in vertex and region blending, although the resulting geometrical problem may be rather complex

### Single edge blends

The simplest form of termination issues arises when just a single edge blend is being created Basically, what has to be done is that the two trimlines have to be joined in some way at the end of the blend (see *Figure 14*) to construct the missing boundary of the blend face Note that the user will probably specify that the blend should terminate at the existing end of the edge being blended (see *Figure 15*, lower blend), but it is also possible in general to allow blends to terminate part of the way along the edge (see *Figure 15*, upper blend)

There are several ways in which the trimlines can be joined In the simplest case, the two trimlines can be made to converge to a single point, as in the example at the left-hand end of the upper blend in *Figure 15* This point will probably be a point of the original edge being blended, as shown It is also possible to join the trimlines by means of a curve (or curves) running in free space, as shown at the right-hand end of the upper blend, or as shown at the ends of the lower blend, in *Figure 15* Finally,

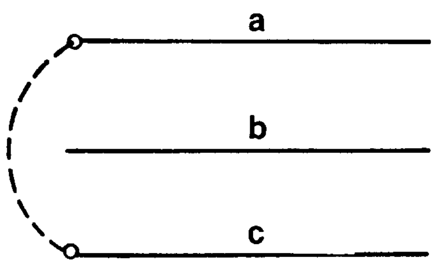



**Figure 14** Trimlines must be joined at end of an edge, (a), (b) trimlines, (c) edge

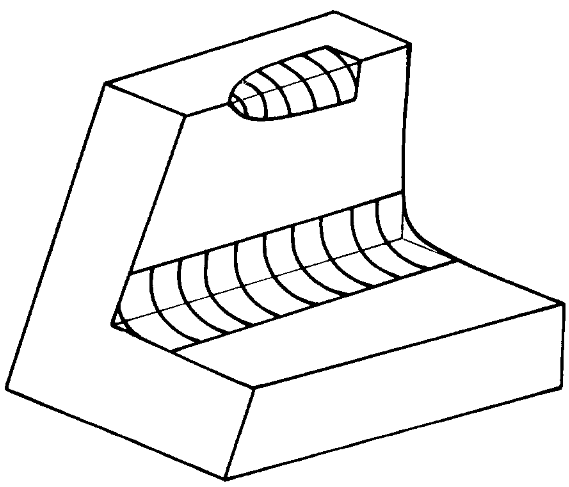


**Figure 15** Single edge blend and terminations part of way along edge

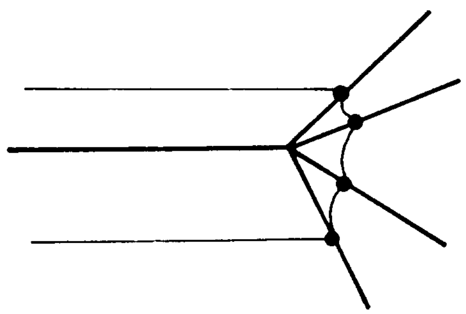


**Figure 16** Trimlines joined by curves running across existing faces

the trimlines may be joined by a curve (or curves) running across existing faces, as shown in *Figure 16*, or indeed by some combination of free space curves and curves across existing faces. More complex cases requiring combinations of these ideas may also occur[4]

Note that, when a curve (or curves) are constructed to join the ends of the trimlines, they may arise logically in two separate ways. First, the two points on the trimlines at the end of the blend may be in assignment, and a profile curve may be constructed in the usual way This type of approach might be appropriate for simple cases (see *Figure 15*). However, another possibility is to use some method of constructing the blend surface so that it extends past the vertex, and then to trim it back by intersection with existing faces of the model (or their extensions if necessary). The blend shown in *Figure 16* might well have been produced in such a manner Using this method, the endpoints of each trimline will not generally be a pair of assigned points

Note that a range of geometric and topological constructions may be required in these various cases For example, existing faces (other than the base surfaces), such as the L-shaped faces in *Figure 15*, may need to be extended, or they may need to be intersected with the blend surface and cut back, as shown in *Figure 16* Whole new small facets may need to be created, together with appropriate edges joining them to existing faces, as shown at the right-hand end of the upper blend in *Figure 15*

In summary, deciding what to do at the ends of blends is a difficult problem Both the local topology and geometry need to be taken into account in the region of the end of the blend so that it can be decided what types of topological solution are feasible or acceptable, we must also consider what geometric constructions are acceptable or even possible Generally, different treatments are appropriate for different cases, and the information used to decide on which solution to adopt generally involves the classification of edges as convex, concave or smooth

## Multiple edge blends

Typically, a single blend is not created in isolation, but often several edge blends are required when an object is blended In many ways, to simplify the constructions required at vertices, it is more sensible to create these blends simultaneously. If blends meeting at a vertex are treated in isolation, then work may be done to create facets or edges which are later thrown away when subsequent blends are considered, for example. Also, it is more likely that a simpler blend at the vertex (with fewer small edges and facets), and probably a more symmetrical one too, will be created if the edge blends are not done one-by-one. Note that it is almost impossible to fix the radius of a blend at a vertex at which two or more blends meet, and, typically, all the user requires in such cases is a smooth transition from one blend surface to the other Thus, for several reasons, it is desirable to consider how to deal simultaneously with two or more blends meeting at a given vertex Again, the problem can be described in terms of how to join the trimlines of the blends meeting at the vertex, as shown in *Figures 17* and *18* for two and many blends meeting at a vertex, respectively We return to these cases in detail further below

As well as the methods described for single edge-blend termination, we may now need further constructions, such as the finding of the intersection between two blend surfaces, for example Generally, when more than one blend meets at the same vertex, the already difficult set of cases to consider becomes much more complex

When several blends meeting at a given vertex are

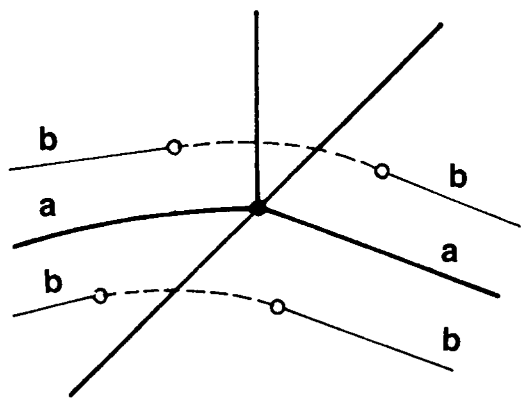



**Figure 17** Two blends meeting at vertex (a) edges, (b) trimlines

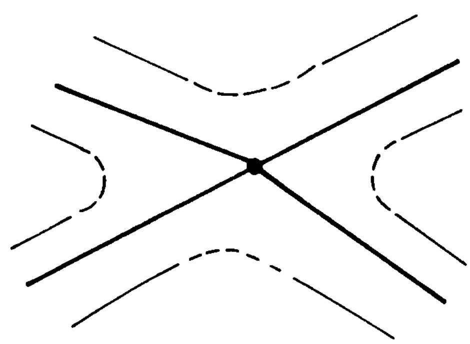


**Figure 18** Many blends meeting at vertex

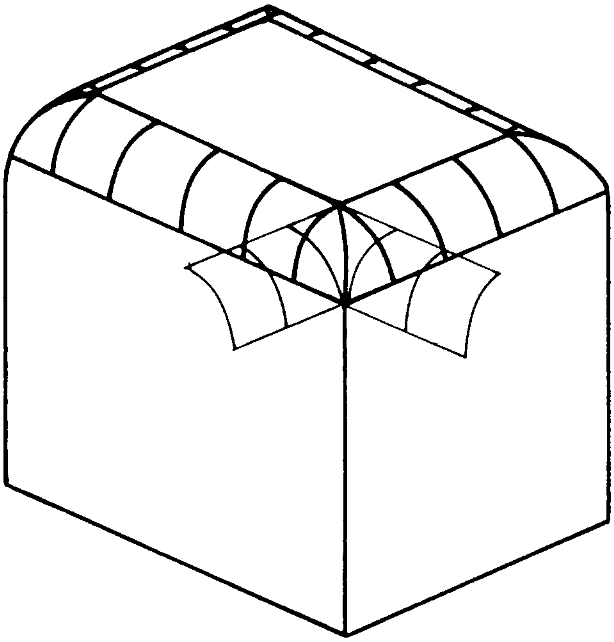


Figure 19 Stripwise blending around top of cube

considered, two basic philosophies have been proposed, as shown in *Figures 17* and *18* The first can be called *stripwise* blending, when the blend to be created consists of a linear sequence of edge blends, with the specific restriction that only two edge blends should meet at any given vertex Such a strip is shown for the ring of faces running around the top of a cube in *Figure 19*, note that the start and end of the strip may be at the same point, as in this case The second possibility can be called *starwise* blending, when more than two edge blends meet at a given vertex Note that, in either case, other edges may meet at the vertex as well as the ones being blended

## Stripwise blending

As described above, the main topological issue here is that two trimlines (from one edge blend) go into the vertex, and two other trimlines leave the vertex (from the other edge blend), and these need to be appropriately joined somehow across the vertex (see *Figure 17*) Different approaches or cases may require us to consider these joining edge pieces first, and then span them with surfaces, or, alternatively, appropriate surfaces may be constructed (or extended) if required, and intersected with other surfaces to find these edge pieces Generally, several edge pieces and several surface pieces may be needed at the vertex

The main philosophy of the stripwise approach is that, by restricting the number of blend surfaces meeting at a given vertex to two, things are kept relatively simple The number of possibilities to consider can hopefully be kept reasonably small, and so a set of methods can be given which will cover these possibilities Also, it should be relatively easy to visualize what is desirable behaviour at vertices when considering how to deal with vertex crossing for stripwise blends

However, if we wish to blend all the edges of a cube, for example, using the stripwise technique, then blends on blends have to be used First, say, the top ring of edges and the bottom ring of edges are blended independently, perhaps with each blend surface being intersected with its neighbour to produce a new sharp edge in place of each old vertex, as shown in *Figure 19* Finally, four vertical edge strips are each blended independently, each strip consisting of what remains of each vertical edge, plus the two intersection edges previously created at each end Thus, stripwise methods still have some disadvantages They share the problem of single edge blends in that extra work may be done which is not required in the final result, as shown by the intersection curves in the cube-blending case which are not required in the final result The final blends created at the corners of the cube are almost certain to be asymmetrical, as the edges are treated differently Further, a burden is placed on the user, who must decide in what order blending is to be done (or some arbitrary automatic method must be used)

Other problems may also arise with this approach For example, if a rolling-ball method is used to generate blends, and some very small edges are created as the result of earlier blends (again consider the cube example), then it is almost certain that the method will break down if the length of some edge being blended is smaller than the size of the ball The ball will not touch the faces which are adjacent to that edge, for example, and any assumptions which are made about trimlines being created on those faces will break down In such cases, the system must either report a failure of the method, which is quite unsatisfactory, or more sophisticated methods must be developed to decide what to do in such regions

## Starwise blending

The alternative to stripwise blending is to use starwise blending Here, all edge blends meeting at a given vertex are created at the same time, and the region near a vertex is blended with a single vertex blend The basic method used here is that edge blends are created along the main length of the edges being blended, which are then cut back (or otherwise stopped) at some point, the ends of adjacent trimlines are joined across existing faces, and finally an *n*-sided patch is inserted which smoothly joins all the edge blends and existing faces This process is shown in *Figure 20*, note that only some of the edges meeting at the vertex are being blended in this case

In principle, there is no reason why this process should be limited to dealing with the problems arising at a single vertex, and, indeed, more generally, a region blend can be used to replace a section of the original object's outer surface which contains more than one vertex The method

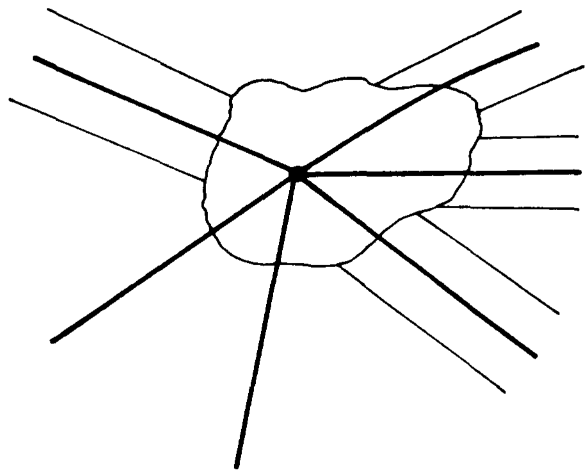


**Figure 20** Starwise blending

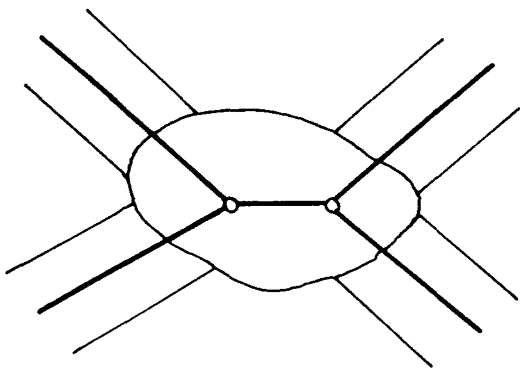


**Figure 21** Joining edge blends running into closely adjacent vertices

of construction essentially remains the same, and it is just the nature of what is removed from the original object which differs This process might well be the natural one to carry out when two vertices in the original object are close together, for example This idea is illustrated in *Figure 21*

The main advantage of the starwise approach is that the complex topological issues which must be considered in the stripwise approach are, to a large extent, avoided, but there is a price to pay There is now a more complex geometric problem to be solved, which is that of constructing an $n$-sided patch which meets the existing surfaces and blend surfaces with the desired continuity.

Taking the ideas a little further, in practice, we may use one or more patches to fill the $n$-sided region A single $n$-sided patch may be used as described above, but this will have the disadvantage of probably being of a different patch type to the other patches used by the modeller This means that it will be necessary for extra software to be written to interrogate this patch type, to intersect such patches with other patches, and so on. One alternative possibility is to use several 4-sided patches which all meet at a (new) vertex, as shown in *Figure 22* Here, a new edge is constructed that goes from the midpoint of each edge of a 5-sided region to the new vertex, resulting in five 4-sided patches. Some control over the shape of the blend can be exercised by adjusting the position of the new vertex in this case Although there is still a continuity problem to be solved (both between the blend patches themselves, and between the blend patches and the base surfaces), the patches may now in principle be of the same type as any other patches used by the modeller For a review of $n$-sided-patch methods see Reference 7

As a separate issue, the way in which adjacent trimlines are joined across the original faces may also differ. They may be joined by a single smooth curve, as typified by the examples in *Figures 20, 21* and *23*a Alternatively, they may be joined by 'sharp', angled pieces, possibly but not necessarily following the direction of the trimlines to their 'natural' meeting point, as shown in *Figure 23b* Alternatively, the trimlines may be allowed to intersect naturally where appropriate, and then no joining pieces will be necessary, as shown in *Figure 23c* Depending on which of these possibilities is chosen, the number of sides $n$ of the $n$-sided patch will be different

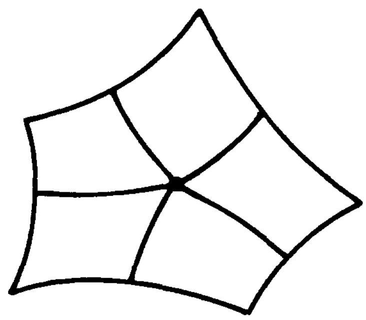


**Figure 22** Several 4-sided patches filling $n$-sided region

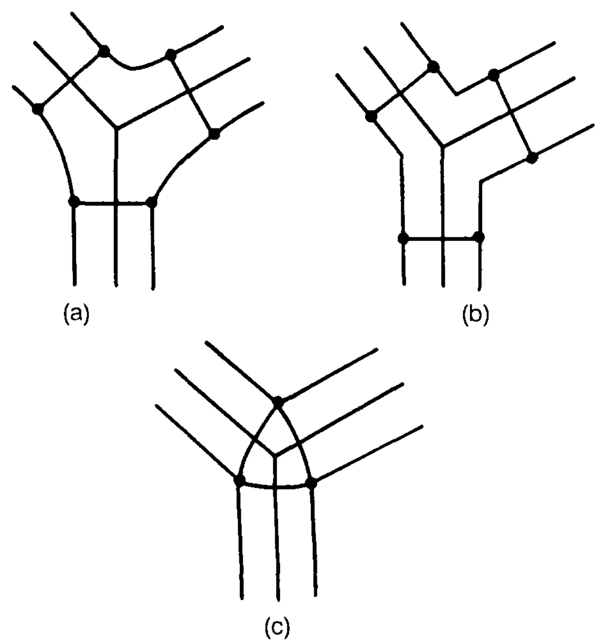



**Figure 23** Joining adjacent trimlines, (a) by single smooth curves, (b) by sharp pieces, (c) by finding their intersection points

Note also as a final possibility that, as with stripwise blending, the adjacent trimlines may be joined by a space curve rather than one lying on the base surfaces In such a case, as before, small facets will also have to be inserted It is unlikely, however, that such a technique will be of much general use in starwise blending

In summary, it may be said that, while the starwise approach is quite appealing because the topological problem is rendered quite simple, unfortunately, no good general methods are known for creating $n$-sided patches for general $n$ which meet the likely desired continuity requirements, and generally only numerical continuity is achieved

## Individual methods

Let us start by noting that several *implicit-blending methods* also deal with topological problems, but that the issues arising in such cases tend to be of a nature different from those of the topological issues in parametric blending Part of the reason for this difference is that it is expected that implicit-blending methods will be used in set-theoretic models, while parametric ones can be associated with a local definition, and thus with a trimmed face/surface or B-rep scheme We will not consider the termination and topological issues arising from implicit blends further here, but the interested reader may wish to consult various papers covered by Woodwark's survey[3], as well as several more recent ones[1,75-81]

Blending *isolated vertices* and *regions* independently of edge blending seems at present to be addressed only by the Fourier and related methods described above

Using the method described in Reference 82, topological problems of *multiple edge blending* can be converted into geometric ones by interpolating a curve network of 3- and 4-sided meshes $C^1$ or $C^2$ continuity can be ensured by adding compensating patches to the base interpolating patches This process requires the introduction of a new surface type, but it allows the topological structure of the curve network to remain unchanged

The *stripwise* alternative of multiple edge blending is used by the methods of Varady, Martin and Vida Here, blending surfaces are approximated by free-form surface strips, which allows the blending of a B-rep solid model bounded by planar, quadric and free-form surfaces (see above) The basic concept used here[4,39,83] for considering topological issues is that blending an object consists in creating blending edge strips (or sequences), and that blending is a local operation which changes only the faces that are neighbours to the edges (or vertices) being blended $n$-sided-patch techniques are not used, but methods of handling the sharp edges meeting the blending surface are considered instead, and generally the edge blends meeting at vertices are joined smoothly by the sharp edges between them being blended Martin[39] presents a series of rules for constructing the new topology From the basic assumptions of the blending model, algorithms for increasingly complex cases of edges meeting the sequence of edges being blended are expounded, also, several alternatives for termination are suggested The topological-restructuring rules are based on convexity, types of connections of trimlines, and user parameters

The implementation of rolling-ball edge blends[19] in the above stripwise framework led to the perception that rolling-ball blending can be extended beyond blending between a pair of faces meeting at an edge to blending between any combination of faces, edges and vertices[84,85] (see *Figure 24*) The rules and algorithm used for topological restructuring can be simplified in this way, and fewer vertex-blend cases remain to be solved by constructive or $n$-sided-patch techniques This model also makes it easier to work out what to do when the rolling ball contacts elements in a wider local neighbourhood of the edge (sequence) being blended than just the faces directly adjacent to these edges It should be noted that, in the current model, all the incoming edge blends must be generated by a ball with the same radius

We note here that such an approach could, in principle, be taken further to make use of an extended B-rep which includes a description of the Voroni surface or skeleton structure[30,31] as well as the existing description of the object This would be useful, because most degenerate cases and all unwanted global interactions are proximity problems, which are not always found by local algorithms However, it is still an area for further research to show how to create and maintain such additional structures, to show how much this costs computationally,

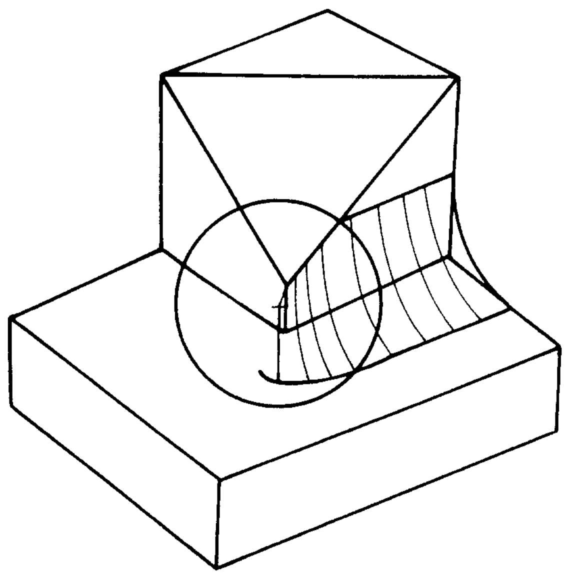


**Figure 24** Rolling ball between vertex and face

and to determine whether this can be done for a reasonable range of cases.

As already discussed, *n-sided-patch techniques* can be used in solutions for vertex blending, region blending, and models using *starwise* topology for joining multiple edge blends We review only those *n*-sided-patch methods here which have been specifically considered in the context of blending.

An *n*-sided interpolation technique called RISP (Rim Specified Patch) was given in Reference 86 to allow the manufacture of blending gaps between conventional surface elements The specification of the problem is to some extent similar to that used by the PDE method described above Position vectors along the contour to be filled in are given together with slopes across this patch rim Discrete boundary values are then taken and averaged to define the internal shape of the RISP, which is constructed in a way that results in a polar parameterization for the patch

A comprehensive *starwise* blending framework is presented in Reference 20 (see also above) which has evolved from the polyhedral methods of Chiyokura It permits arbitrary curved base surfaces, and supports rolling-ball and other trimline definitions The basic approach is that trimlines are generated first on the faces that are neighbours to the edges being blended. Then, three types of special point are distinguished, as shown in *Figure 25* An end of a trimline inside a face is called a C point Intersections of trimlines, if any, with the winged edges of the edge being blended are termed E points Existing intersections of consecutive trimlines on faces are called I points. New vertices are created at C, E and I points At a single C vertex, the blend terminates, while consecutive C vertices on the same face are connected with an extra edge Circular-arc edges connect assigned vertices where applicable, in the same way as circular arcs connect assigned trimline points Other

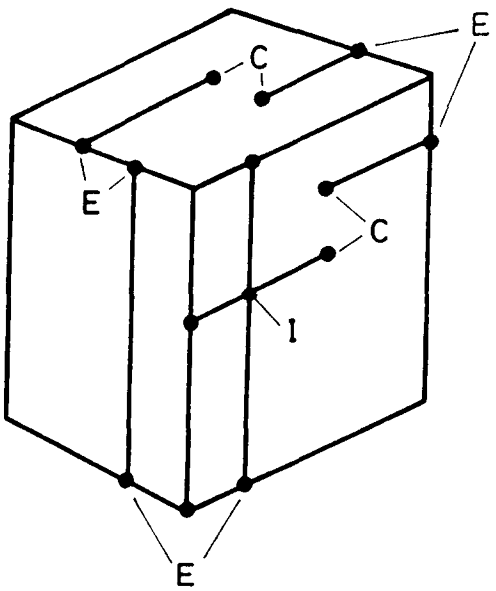



**Figure 25** C, E and I points

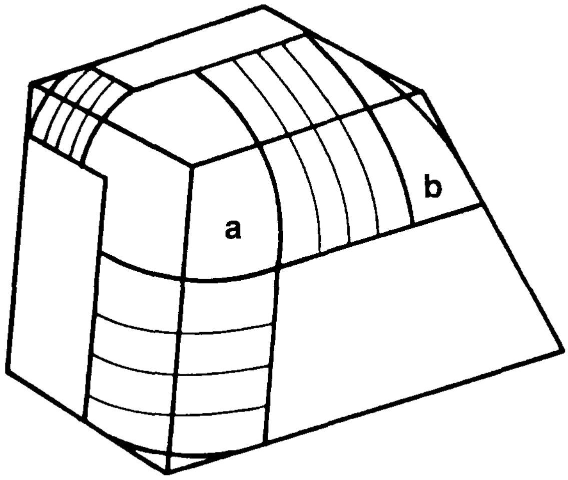



**Figure 26** *n*-sided regions, (a) vertex, (b) at termination

endpoints are connected with more general free-form curves *n*-sided regions may appear at vertices and terminations (see *Figure 26*). As a refinement, sharp corners of trimlines at I vertices are rounded within faces, which can be considered as converting the case shown in *Figure 23c* to that in *Figure 23a* Finally, a surface-interpolation algorithm is used to fit rational-boundary Gregory patches[94] to this data, which results in a patch structure of 4-sided patches

The constant-radius rolling-ball blending method of Choi and Ju (see above) provides a solution for joining edge blends meeting at a vertex The base surfaces can be parametric surfaces, and the edge blends are procedural surfaces which can be evaluated by substitution into the base surfaces A 3-sided corner also has a procedural representation which is expressed in terms of the edge blends The corner patch is defined as a convex combination of the linear Taylor interpolants[87] of the edge blends along the final profile curves as shown in *Figure 23c*, and provides one degree of freedom for each edge blend to change the shape of the corner The 3-sided method is extended in a straightforward way to *n*-sided cases in Reference 32

One very important issue is that of providing a pleasing interior for vertex blends. In Reference 88, an energy-minimization approach is described The vertex-blend surface is subdivided into triangular interpolants, and it is assumed that the subdividing curve segments form a $G^2$-continuity compatible curve network which is optimal with respect to minimizing the curvature variation along the network The blend interpolates position vectors and optionally surface normals and surface curvatures to ensure the desired degree of continuity with the existing surface In Reference 89, a more-general surface-interpolation scheme with 4-sided patches is introduced which is also directly applicable to the vertex-blend problem Here also the curvature variation is minimized numerically for joining quintic Bézier patches with $G^1$ continuity These approaches are somewhat computationally expensive, producing quite good results for typical input data, although, if deliberately difficult cases are chosen, the methods can produce less predictable results

Zhao and Rockwood developed a special *n*-sided-patch formulation called the convolution patch for constructing vertex blends to join edge blends in solid models[90] The edge blends are cut back to some extent from the vertex, and then 'spring' curves are generated to join the adjacent trimlines lying on one face, as shown in *Figure 23a* Thus, if *n* edges run together, a 2*n*-sided patch is created The patch is formed by the convolution of geometric primitives and so-called weight spline patches, which provide $G^1$ continuity internally The representation surprisingly leads to 2*n* tensor-product subpatches of degree 5 (A parametric singularity occurs at the midpoint ) The edge-profile curves and the spring curves can be represented by cubic polynomial or rational curves, and so smooth connection to rolling-ball edge blends is also provided

A special scheme for vertex blending using standard patches has been developed in Reference 91 This scheme also creates 'spring-curve'-type vertex blends, as above The composite patch is constructed mostly by bicubic and biquartic patches excluding parametric singularities, while, at the same time, means for interior shape control are also offered

Special cases for constructing 3- to 6-sided patches to join edge blends are given in Reference 92 The basic idea of this method is to some extent similar to that of References 5 and 58 (see above), but it is applied to the more complex problem of joining blends

An interesting step-by-step rolling-ball vertex blend is presented in Reference 93 which seems to be a mixture of stripwise and starwise blending The rolling-ball blends are generated in decreasing order of radii until two remain which are connected by a variable-radius blend (Unfortunately, the technical details of this method are currently available only in Japanese )

## CONCLUSIONS

As we have already noted, we feel that the use of parametric surfaces has certain advantages over the use of implicit surfaces in blending In particular, the two main advantages are (a) parametric surfaces can be used to blend parametric or implicit base surfaces, while, in practice, implicit blends seem only to be useful for implicit base surfaces, and (b) the fact that parametric surfaces have a parameterization is of considerable benefit when using and interrogating such surfaces

As this survey shows, there is a very wide range of approaches to the creation of parametric blends, and it is difficult to compare them, as they use some very different basic ideas for blend construction Nevertheless, we feel that we have shown that such methods can be classified by considering in particular how the blend is defined, how it is represented, and how it is interrogated. An important issue is the continuity of the blend, between both the blend surface and the base surfaces, and successive patches of the blend surface if several patches are used Within the classification given here, there may be several methods of a certain type, but it may not always be obvious which methods offer an advantage over others, or when

Nevertheless, we hope that, by showing the current state of the art of parametric blending, and by helping to draw together the underlying concepts, we will stimulate further research in this important area We feel that the major outstanding topics requiring further research are the construction of variable-radius blends, the achievement of blends of higher-order continuity than simply tangent-plane continuity, and the termination/topological issues

## ACKNOWLEDGEMENTS

This work was carried out in the Computer and Automation Institute of the Hungarian Academy of Sciences, Hungary, and it was supported by the National Science Foundation of the Hungarian Academy of Sciences Dr Martin wishes to acknowledge the financial support of the Nato Visiting Expert programme and the Hungarian Academy of Sciences, which made his visit possible

All the authors would like to thank the many persons whose work is described in this review who have provided details of their work and other assistance. Special thanks are due to the referees of this paper for calling the authors' attention to various references and concepts that they had missed

Finally, the authors would like to apologise to any persons affected if they have inadvertently omitted any significant work in this area, or if they have misrepresented any other person's work

## REFERENCES

1 Rockwood, A P 'Blending surfaces in solid geometric modelling' *PhD Thesis* University of Cambridge, UK (1987)

2 Pratt, M J 'Surfaces in solid modelling' *in* Creasy, C F M and Craggs, C (Eds ) *Applied Surface Modelling* Ellis Horwood (1990) pp 85–103

3 Woodwark, J R 'Blends in geometric modelling' *in* Martin, R R (Ed ) *The Mathematics of Surfaces II* Oxford University Press (1987) pp 225–297

4 Varady, T, Vida, J and Martin R R 'Parametric blending in a boundary representation solid modeller' *in* Handscomb, D C (Ed ) *The Mathematics of Surfaces III* Oxford University Press (1989) pp 171–197

5 Pratt, M J 'Automatic blending in solid modelling – an approach based on the use of parametric geometry' *Papers 4th Joint Anglo-Hungarian Sem CAD* Budapest, Hungary (1985) pp 81–97

6 Rossignac, J R and Requicha, A A G 'Constant-radius blending in solid modeling' *Comput Mech Eng* Vol 3 (Jul 1984) pp 65–73

7 Hermann, T, Renner, G and Varady, T 'Mathematical techniques for interpolating surfaces with general topology' *GML-1992/1* Computer and Automation Institute, Hungarian Academy of Sciences, Hungary (1992)

8 Braid, I C 'Superficial blends in geometric modelling' *CAD Group Document No 105* University of Cambridge, UK (1980)

9 Owen, J 'STEP Part 1 Overview and fundamental principles' *Document ISO TC 184/SC4/WGI N467* National Institute of Standards and Technology, USA (1990)

10 Vaishnav, H and Rockwood, A 'Blending parametric objects by implicit techniques' *Proc 2nd ACM Solid Modeling Symp* Montreal, Canada (1993) pp 165–168

11 Lee, T, Bedi, S and Dubey, R N 'A parametric surface blending method for complex engineering objects' *Proc 2nd ACM Solid Modeling Symp* Montreal, Canada (1993) pp 179–188

12 Sanglikar, M A, Koparkar, P and Joshi, V N 'Quasi parallel derivatives for efficient blending of parametric surfaces' *Pres Int Conf Manuf Automat* Hong Kong (1992)

13 Hoffmann, C M *Geometric and Solid Modeling* Morgan Kaufmann (1989)

14 Farin, G E *Curves and Surfaces for Computer Aided Geometric Design A Practical Guide* (2nd Ed ) Academic Press (1990)

15 Sanglikar, M A, Koparkar, P and Joshi, V N 'Modelling rolling ball blends for computer aided geometric design' *Comput Aided Geom Des* Vol 7 (1990) 399–414

16 Hoffmann, C M 'A dimensionality paradigm for surface interrogations' *Comput Aided Geom Des* Vol 7 (1990) pp 517–532

17 Choi, B K and Ju, S Y 'Constant-radius blending in surface modelling' *Comput-Aided Des* Vol 21 No 4 (1989) pp 213–221

18 Filip, D J 'Blending parametric surfaces' *ACM Trans Graph* Vol 8 No 3 (1989) pp 164–173

19 Varady, T 'Rolling ball blends in solid modelling' *Proc 3rd Int IFIP Conf Computer Applications in Production & Engineering* North-Holland (1989) pp 295–308

20 Harada, T, Toriya, H and Chiyokura, H 'An enhanced rounding operation between curved surfaces in solid modeling' *Proc CG International '90* Springer-Verlag (1990) pp 563–588

21 Klass, R and Kuhn, B 'Fillet and surface intersections defined by rolling balls' *Comput Aided Geom Des* Vol 9 (1992) pp 185–193

22 Hermann, T 'Rolling ball blends, self intersections' *in* Warren, J D (Ed ) *Curves and Surfaces in Computer Vision and Graphics III* SPIE (1992) pp 204–209

23 Barnhill, R E, Farin, G E and Chen, Q 'Constant-radius blending of parametric surfaces' *in* Farin, G, Hagen, H and Noltemeier, H (Eds ) *Geometric Modelling* Springer-Verlag (1993)

24 Braid, I C 'A history of geometric modeling' *Proc 1st ACIS TECH-EX Forum* Spatial Technology, Chicago, IL, USA (Nov 1991) pp 1–17

25 Pegna, J 'Variable sweep geometric modeling' *PhD Dissertation* Stanford University, USA (1987)

26 Pegna, J and Wilde, D J 'Spherical and circular blending of functional surfaces' *Trans ASME J Offshore Mechanics & Arctic Eng* Vol 112 (1990) pp 134–142

27 Pegna, J and Wolter, F-E 'Designing and mapping trimming curves on surfaces using orthogonal projection' *Proc 16th ASME Design Automation Conf Advances in Design Automation – Vol I* ASME, USA (1990) pp 235–245

28 Pegna, J and Wolter, F-E 'Geometrical criteria to guarantee curvature continuity of blend surfaces' *ASME Trans J Mech Des* Vol 114 (Mar 1992) pp 201–210

29 Dutta, D 'Variable-radius blends and Dupin cyclides' *PhD Thesis* Purdue University, USA (1989)

30 Chandru, V, Dutta, D and Hoffmann, C M 'Variable radius blending using Dupin cyclides' *in* Wozny, M, Preiss, K and Turner, J (Eds ) *Geometric Modeling for Product Engineering* North-Holland (1990) pp 39–57

31 Dutta, D and Hoffmann, C M 'A geometric investigation of the skeleton of CSG objects' *in* Ravani, B (Ed ) *Advances in Design Automation* ASME, USA (1990) pp 67–75

32 Choi, B K *Surface Modeling for CAD/CAM* Elsevier (1991)

33 Harada, T, Konno, K and Chiyokura, H 'Variable-radius blending by using Gregory patches in geometric modeling' *Proc Eurographics '91* North-Holland (1991) pp 507–518

34 Sanglikar, M A, Koparkar, P and Joshi, V N 'Locally invertible topological map for parametric blending' *ICCG Int Conf Computer Graphics Interaction, Design, Modelling & Visualisation* Bombay, India (22–26 Feb 1993)

35 Welbourn, D B 'The use of DUCT' Wolfson Cambridge Industrial Unit, University of Cambridge, UK (1981)

36 Bardis, L and Patrikalakis, N M 'Blending rational B-spline surfaces' *Proc Eurographics '89* North-Holland (1989) pp 453–462

37 Hansmann, W 'Interactive design and geometric description of smooth transitions between curved surfaces' *Trans ASME J Comput Offshore & Arctic Eng* (1987) pp 19–26

38 Filkins, P C, Tuohy, S T and Patrikalakis, N M 'Computational methods for blending surface approximation' *Eng Comput* (1993) pp 49–62

39 Martin, R R 'Notes on the topological aspects of the FFSOLID blending model' *GMS-1989/2* Geometric Modelling Studies, Computer and Automation Institute, Hungarian Academy of Sciences, Hungary (1989)

40 Varady, T 'Integration of free-form surfaces into a volumetric modeller' *Dissertation* Computer and Automation Institute, Hungarian Academy of Sciences, Hungary (1985)

41 Koparkar, P 'Canopy surface technique for parametric blending' *Visual Comput* Vol 9 No 2 (1992) pp 84–90

42 Koparkar, P 'Designing parametric blends surface model and geometric correspondence' *Visual Comput* Vol 7 (1991) pp 39–58

43 Koparkar, P 'Parametric blending using fanout surfaces' *Proc ACM Symp Solid Modeling Foundations & CAD/CAM Applications* Austin, TX, USA (1991) pp 317–327

44 Bien, A-P and Cheng, F 'A blending model for parametrically defined geometric object' *Proc ACM Symp Solid Modeling Foundations & CAD/CAM Applications* Austin, TX, USA (1991) pp 339–347

45 Catmull, E E and Clark, J H 'Recursively generated B-spline surfaces on arbitrary topological meshes' *Comput-Aided Des* Vol 10 No 6 (1978) pp 350–355

46 Doo, D W H 'A subdivision algorithm for smoothing down irregular shaped polyhedrons' *Proc Int Conf Interactive Techniques in CAD* Bologna, Italy (Sep 1978) pp 157–165

47 Veenman, P 'Design of sculptured surfaces using recursive subdivision techniques' *Proc Conf CAD/CAM Technology in Mechanical Engineering* MIT Press (1982) pp 54–63

48 Doo, D and Sabin, M 'Behavior of recursive division surfaces near extraordinary points' *Comput-Aided Des* Vol 10 No 6 (1978) pp 356–360

49 Ball, A A and Storry, D J T 'Conditions for tangent plane continuity over recursively generated B-spline surfaces' *ACM Trans Graph* Vol 7 No 2 (1988) pp 83–102

50 Ball, A A and Storry, D J T 'An investigation of curvature variations over recursively generated B-spline surfaces' *ACM Trans Graph* Vol 9 No 4 (1990) pp 424–437

51 Nasri, A H 'Polyhedral subdivision methods for free-form surfaces' *ACM Trans Graph* Vol 6 No 1 (1987) pp 29–73

52 Dyn, N, Levin, D and Gregory, J A 'A butterfly subdivision scheme for surface interpolation with tension control' *ACM Trans Graph* Vol 9 No 2 (1990) pp 160–169

53 Tan, S T and Chan, K C 'Generation of high order surfaces over arbitrary polyhedral meshes' *Comput-Aided Des* Vol 18 No 8 (1986) pp 411–423

54 Storry, D J T and Ball, A A 'Design of an *n*-sided surface patch from Hermite boundary data' *Comput Aided Geom Des* Vol 6 (1989) pp 111–120

55 Chiyokura, H and Kimura, F 'Design of solids with free-form surfaces' *Comput Graph* Vol 17 No 3 (1983) pp 289–298 (*Proc SIGGRAPH*)

56 Fjallstrom, P-O 'Smoothing of polyhedral models' *Proc 2nd Symp Computational Geometry* ACM (1986) pp 226–235

57 Chiyokura, H 'An extended rounding operation for modeling solids with free-form surfaces' *IEEE Comput Graph & Applic* Vol 7 No 6 (1987) pp 27–36

58 Beeker, E 'Smoothing of shapes designed with free-form surfaces' *Comput-Aided Des* Vol 18 No 4 (1986) pp 224–232

59 Szilvasi-Nagy, M 'Flexible rounding operation for polyhedra' *Comput-Aided Des* Vol 23 No 9 (1991)

60 Pratt, M J 'Cyclides in computer aided geometric design' *Comput Aided Geom Des* Vol 7 No 1–4 (1990) pp 221–242

61 Pratt, M J 'Applications of cyclide surfaces in geometric modelling' *in* Handscombe, D C (Ed ) *The Mathematics of Surfaces III* Oxford University Press (1989) pp 405–428

62 Pratt, M J 'Cyclide blending in solid modelling' *in* Strasser, W and Seidel, H-P (Eds ) *Theory and Practice of Geometric Modeling* Springer-Verlag (1989) pp 235–245

63 Faux, I D (Ed ) *2nd CAM-I Geometric Modelling Sem* Computer Aided Manufacturing International, Cambridge, UK (1983)

64 Boehm, W 'Some remarks on cyclides in geometric modeling' *in* Strasser, W and Seidel, H-P (Eds ) *Theory and Practice of Geometric Modeling* Springer-Verlag (1989) pp 247–252

65 Boehm, W 'On cyclides in geometric modeling' *Comput Aided Geom Des* Vol 7 (1990) pp 243–255
66 Bloor, M I G and Wilson, M J 'Generating blend surfaces using partial differential equations' *Comput-Aided Des* Vol 21 No 3 (1989) pp 165–171
67 Bloor, M I G and Wilson, M J 'Blend design as a boundary-value problem' *in* Strasser, W and Seidel, H-P (Eds) *Theory and Practice of Geometric Modeling* Springer-Verlag (1989) pp 221–234
68 Bloor, M I G and Wilson, M J 'Representing PDE surfaces in terms of B-splines' *Comput-Aided Des* Vol 22 No 6 (1990) pp 324–331
69 Brown J M, Bloor, M I G, Bloor, M S and Wilson, M J 'Generation and modification of non-uniform B-spline surface approximations to PDE surfaces using the finite element method' *in* Ravani, B (Ed) *Advances in Design Automation* ASME, USA (1990) pp 265–272
70 Cheng, S Y, Bloor, M I G, Saia, A and Wilson, M J 'Blending between quadric surfaces using partial differential equations' *in* Ravani, B (Ed) *Advances in Design Automation* ASME, USA (1990) pp 265–272
71 Roach, P A and Martin, R R 'Production of blends and fairings by Fourier methods' *in* Warren, J D (Ed) *Curves and Surfaces in Comoputer Vision and Graphics III* SPIE (1992) pp 162–173
72 Roach, P A 'Fourier methods for curve and surface smoothing in computer aided design' *PhD Thesis* University of Wales College of Cardiff, UK (1992)
73 Fowler, E 'Global blending and Fourier based CAD' *MSc Thesis* Cranfield Institute of Technology, UK (1990)
74 Colburn, S 'Method for global blending of computer modeled solid objects using a convolution integral' *United States Patent 4791583* (1988)
75 Bajaj, C L and Ihm, I 'Algebraic surface design with Hermite interpolation' *ACM Trans Graph* Vol 11 No 1 (1992) pp 61–91
76 Hartmann, E 'Blending of implicit surfaces with functional splines' *Comput-Aided Des* Vol 22 No 8 (1990) pp 500–506
77 Hoffmann, C M and Hopcroft J 'The geometry of projective blending surfaces' *Artif Intell* Vol 37 (1988) pp 357–376
78 Holmstrom, L, Laakko, T, Mantyla, M and Rekola, P 'Ray tracing of boundary models with implicit blend surfaces' *in* Strasser, W and Seidel, H-P (Eds) *Theory and Practice of Geometric Modeling* Springer-Verlag (1989) pp 253–271
79 Holmstrom, L 'Piecewise quadric blending of implicitly defined surfaces' *Comput Aided Geom Des* Vol 4 (1987) pp 171–189
80 Kosters, M 'Quadratic blending surfaces for complex corners' *Visual Comput* Vol 5 (1989) pp 134–146
81 Ohkura, K and Kakazu, Y 'Generalization of the potential method for blending three surfaces' *Comput-Aided Des* Vol 24 No 11 (1992) pp 599–610
82 Saitoh, T and Hosaka, M 'Interpolating curve networks with new blending patches' *Proc Eurographics '90* North-Holland (1990) pp 137–146
83 Varady, T and Vida, J 'Blending surfaces in geometric modelling' *TA/IH-V 3/86* Computer and Automation Institute, Hungarian Academy of Sciences, Hungary (1986) (in Hungarian)
84 Vida, J 'Integration of blends into a solid modeller' *Dissertation* Computer and Automation Institute, Hungarian Academy of Sciences, Hungary (1992) (in Hungarian)
85 Vida, J 'Rolling a ball around a B-rep solid model' (in preparation)
86 Bloor, M S, de Pennington, A and Woodwark, J R 'RISP bridging the gap between conventional surface elements' *Proc CAD-78 Conf* Butterworths (1978) pp 398–405
87 Gregory, J A $C^1$ rectangular and non-rectangular surface patches' *in* Barnhill, R E and Boehm, W *Surfaces in CAGD* North-Holland, Netherlands (1983)
88 Moreton, H and Sequin, C 'Surface design with minimum energy networks' *Proc ACM Symp Solid Modeling Foundations & CAD/CAM Applications* Austin, TX, USA (1991) pp 291–330
89 Moreton, H and Sequin, C 'Functional optimization for fair surface design' *Comput Graph (Proc SIGGRAPH '92)* Vol 26 No 2 (1992) pp 167–176
90 Zhao, Y and Rockwood, A A convolution approach to *n*-sided patches and vertex blending' *in* Sapidis, N (Ed) *Fair Surfaces* (to appear)
91 Varady, T and Rockwood, A P 'Control frame patches for vertex blending' (in preparation)
92 Hosaka M *Modeling of Curves and Surfaces in CAD/CAM* Springer-Verlag (1992)
93 Higashi, M 'High-quality solid-modelling system with free-form surfaces' *Comput-Aided Des* Vol 25 No 3 (1993) pp 172–183
94 Chiyokura, H, Takamara, T, Konno, K and Harada, T '$G^1$ surface interpolation over irregular meshes with rational curves' *in* Farin, G *NURBS for Curve and Surface Design* SIAM (1991) pp 15–34


## BIBLIOGRAPHY


A Blechsmidt, J L and Nagasura, D 'The use of algebraic functions as a solid modeling alternative an investigation' *in* Ravani, B (Ed) *Advances in Design Automation* ASME, USA (1990) pp 265–272
B Chiyokura, H 'An extended rounding operation for modeling solids with free-form surfaces' *Proc Computer Graphics 1987* Springer-Verlag (1987) pp 249–268
C Chiyokura, H *Solid Modeling with DESIGNBASE* Addison-Wesley (1988)
D Dickens, N A 'A survey of literature on blending in solid modelling' *Internal Report* Siemens Cad Development Unit, Cambridge, UK (1989)
E Dickens, N A 'User requirements for blending in parasolid' Siemens Cad Development Unit, Cambridge, UK (1989) (unpublished MS)
F Engeli, M 'Geometric modelling of complex forms with EUKLID' *Manuf Syst* Vol 10 No 2 (1981) (*Proc CIRP Sem*)
G Hansmann, W 'Interaktiver Entwurf und geometrische Beschreibung glatter Übergange zwischen raumlich genkrummten Flachenstrukturen' *Doctoral Thesis* Fachbereich Informatik der Universitat Hamburg, Germany (1985)
H Higashi, M, Mori, T and Hosaka, M 'Interference calculation of surfaces based on their geometric properties' *Proc 3rd Int IFIP Conf Computer Applications in Production & Engineering* North-Holland (1989) pp 275–282
I Hoffmann, C M 'Algebraic and numerical techniques for offsets and blends' *in* Dahmen, W, Gasca, M and Michelli, C (Eds) *Computation of Curves and Surfaces* Kluwer (1990) pp 499–528
J Hoffmann, C M and Hopcroft, J 'Automatic surface generation in computer aided design' *Visual Comput* Vol 1 No 2 (1985) pp 92–100
K Hoffmann, C M and Hopcroft, J 'Quadratic blending surfaces' *Comput-Aided Des* Vol 18 No 6 (1986) pp 301–306
L Hoffmann, C M and Hopcroft, J 'The potential method for blending surfaces and corners' *in* Farin, G E (Ed) *Geometric Modeling Algorithms and New Trends* SIAM, USA (1987) pp 347–365
M Hoffmann, C M and Vermeer, P J 'Eliminating extraneous solutions in curve and surface operations' *Int J Comput Geom & Applic* Vol 1 No 1 (1991) pp 47–66
N Holmstrom, L and Laakko, T 'Rounding facility for solid modelling of mechanical parts' *Comput-Aided Des* Vol 20 No 10 (1988) pp 605–614
O Hoschek, J and Hartmann, E '$G^{n-1}$-functional splines for modeling' *in* Hagen, H and Roller, D (Eds) *Geometric Modeling, Methods and Applications* Springer-Verlag (1991) pp 185–211
P Kosters, M 'An extension of the potential method to higher-order blendings' *Proc ACM Symp Solid Modeling Foundations & CAD/CAM Applications* Austin, TX, USA (1991) pp 329–337
Q Lee, D C-K and Pegna, J 'Orthogonal projection as a tool for surface blending' *Technical Report 90021* Rensselaer Polytechnic Institute, USA (1990)
R Liming, R A *Practical Analytical Geometry with Applications to Aircraft* Macmillan, USA (1944)
S Middleditch, A E and Sears, K H 'Blend surfaces for set theoretic volume modelling systems' *Comput Graph* Vol 19 No 3 (1985) pp 161–170 (*Proc SIGGRAPH '85*)
T Patrikalakis, N M and Bardis, L 'Offsets of curves on rational B-spline surfaces' *Eng Comput* Vol 5 (1989) pp 39–49
U Pegna, J 'Interactive design of curvature continuous fairing surfaces' *Proc 8th OMAE Conf – Vol VI Computer Technology* ASME, USA (1989) pp 191–198
V Ricci, A 'A constructive geometry for computer graphics' *Comput J* Vol 16 No 2 (1973) pp 157–160
W Roach, P A and Martin, R R 'Fourier methods for surface smoothing in computer aided design' *in* Bowyer, A (Ed) *The Mathematics of Surfaces IV* Oxford University Press (to be published)

X Rockwood, A P *Blended Sculptured Surfaces for ROMULUS* Shape Data (1983)

Y Rockwood, A P 'Introducing sculptured surfaces into a geometric modeller' *in* Pickett, M S and Boyse, J W (Eds) *Solid Modeling by Computers From Theory to Applications* Plenum Press, UK (1984) pp 237–258

Z Rockwood, A P 'The displacement method for implicit blending surfaces in solid models' *ACM Trans Graph* Vol 8 No 4 (1989) pp 279–297

AA Rockwood A P and Owen, J C 'Blending surfaces in solid modeling' *in* Farin, G E (Ed) *Geometric Modeling Algorithms and New Trends* SIAM, USA (1987) pp 367–383

AB Rossignac, J R and Requicha, A A G 'Offsetting operations in solid modelling' *Comput Aided Geom Des* Vol 3 No 2 (1986) pp 129–148

AC Sanglikar, M A 'Designing parametric blend surfaces using constant, relaxed and perturbed offsetting' *PhD Thesis* University of Bombay, India (Apr 1992)

AD Varady, T 'Survey and new results in *n*-sided patch generation' *in* Martin, R R (Ed) *The Mathematics of Surfaces II* Oxford University Press (1987)

AE Varady, T, Martin, R R and Vida, J 'Topological considerations in blending boundary representation solid models' *in* Strasser, W and Seidel, H-P (Eds) *Theory and Practice of Geometric Modeling* Springer-Verlag (1989) pp 205–220

AF Warren, J 'Blending quadric surface with quadric and cubic surfaces' *Proc 3rd Symp Computational Geometry* ACM, USA (1987) pp 341–347

AG Warren, J 'Blending algebraic surfaces' *ACM Trans Graph* Vol 8 No 4 (1989) pp 263–278

AH Zhang, D 'CSG solid modelling and automatic NC machining of blend surfaces' *PhD Thesis* University of Bath, UK (1986)

AI Zhang, D and Bowyer, A 'CSG set-theoretic solid modelling and NC machining of blend surfaces' *Proc 2nd Ann ACM Symp Computational Geometry* (1986) pp 236–245


## LITERATURE INDEX

### Implicit blends

1, 15, 31, 75–81, J–L, N–P, S, V, X–Z, AA, AF–AI

### Parametric blends

4–6, 11, 12, 17–19, 20–23, 25, 26, 29, 30, 32–38, 41–62, 64–73, 83, 84, 92, 93, B, C, F, G, U, W, AC

### Termination and topology

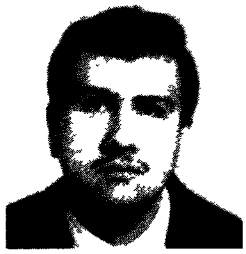
6, 20, 32, 39, 82, 85, 86, 88–92, AE

### Reviews

1–5, D

### Other papers

7–9, 16, 24, 27, 28, 40, 74, 87, A, E, H, I, M, Q, R, T, AB, AD

*Janos Vida is a research fellow From 1985 to 1992, he carried out research and development work in the field of free-form surfaces and B-rep solid modelling at the Computer and Automation Institute of the Hungarian Academy of Sciences, Hungary He received a DiplIng in mechanical engineering and mathematics from the Technical University of Budapest, Hungary, in 1985, and he has submitted a dissertation on blending B-rep solid models to the Academy for the degree of candidate of technical science (PhD)*

*Ralph Martin has been working in the field of CAD/CAM since 1979 He obtained a PhD in 1983 from Cambridge University, UK, for a dissertation on principal patches Since then, he has published on such topics as surface modelling, intelligent sketch input, vision-based geometric inspection, algorithms for containment and sweeping, and geometric reasoning*

*Tamas Varady received a DiplIng in electrical engineering from the Technical University of Budapest, Hungary, in 1976 He obtained a doctorate degree in 1977 He was awarded an advanced degree of technical science (PhD) from the Hungarian Academy of Sciences, Hungary, in 1986 with a dissertation on the integration of free-form surfaces into a volumetric modeller He led the CAD Department at the Computer and Automation Institute of the Academy between 1984 and 1990, where he directed several research projects on geometric modelling and software developments for industrial applications In 1990, he was the cofounder of CADMUS Consulting and Development, of which he is president Since January 1991, he has headed the Geometric Modelling Laboratory at the Institute His recent research interests include general-topology surface modelling, and vertex blending in B-rep solid modellers*