Free-Form Modeling with Implicit Surface Patches

Chandrajit L. Bajaj *
Department of Computer Science,
Purdue University,
West Lafayette, Indiana 47907
bajaj@cs.purdue.edu

1 Introduction

While it is possible to model a general closed surface of arbitrary genus as a single implicit surface patch, the geometry of such a global surface is difficult to specify, interactively control, and polygonize. The main difficulties stem from the fact that implicit representations are iso-contours which generally have multiple real sheets, self–intersections and several other undesirable singularities. In this chapter I shall present details of several implicit polynomial surface splines (one is termed the A-patch) which overcome the above difficulties, and show how these are used in C^1 and C^2 interpolation/approximation and interactive free-form modeling schemes. The potential of implicit patch splines remains largely latent and virtually all commercial and many research modeling systems are based on parametric spline representations. An exception is SHASTRA which allows modeling with both implicit and parametric splines [4].

The important issues in free-form patch modeling of shapes with arbitrary topology are:

1. the patch representation
2. the polynomial degree of the patches
3. the number of patches per face of some input or benchmarking polyhedron
4. functional connectivity and nonsingularity of the patches
5. conditions for the desired continuity between adjacent patches (How the patches "stitched" together to form a "smooth" surface)
6. curvature and higher derivative variation of the patches, especially around the "stitches"

A significant amount of recent research has focussed on these questions with varying emphasis on non-tensor product patches, multivariate generalizations B-splines, geometric continuity, approximation order, and the fairness of fit. Common free-form patch modeling patch schemes include convex combinations of blending functions, and local interpolation of a mesh of curves, simplex and box based schemes, and stationary / non-stationary subdivision. In this chapter I address these issues with different implicit surface patches in a variety of algorithms. There are three broad categories of implicit patch schemes : *curvlinear-mesh* based, *simplex or box* based, and *subdivision* based. These make up the three subsequent sections.

*Supported in part by NSF grant CCR 92-22467, AFOSR grant F49620-94-1-0080, and ONR grant N00014-94-1-0370

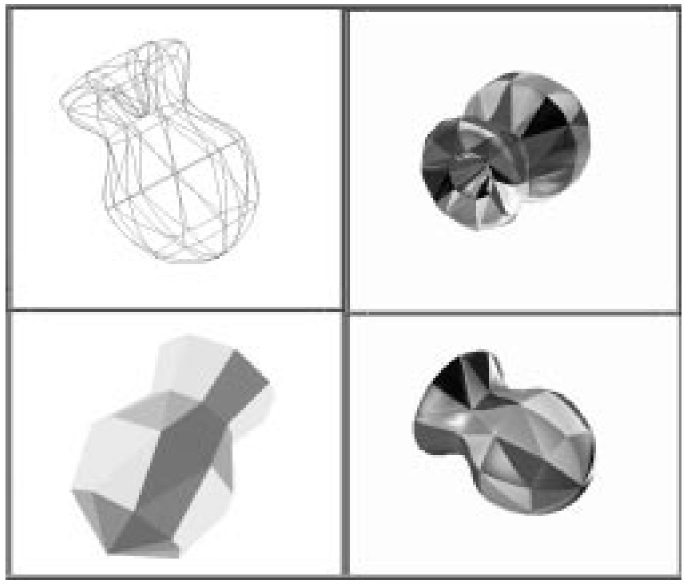


Figure 2.1: C^1 Implicit Splines over a Spatial Triangulation

2 Curvlinear Mesh Scheme

Bajaj, and Ihm [3, 12] construct implicit surfaces to solve the scattered data fitting problem. The resulting surfaces approximate or contain with C^1 continuity any collection of points and algebraic space curves with derivative information. Their Hermite interpolation algorithm solves a homogeneous linear system of equations to compute the coefficients of the polynomial defining the implicit surface. Bajaj, Ihm and Warren [14] extend this idea to C^k (rescaling continuity) interpolation or least squares approximation of a mesh of implicit or parametric curves in space. They also show that this problem can be formulated as a constrained quadratic minimization problem, where algebraic distance is minimized instead of geometric distance.

In a *curvlinear-mesh* based scheme, Bajaj and Ihm [13] construct low–degree implicit polynomial spline surfaces by interpolating a mesh of curves in space using the techniques of [3, 12, 14]. They consider an arbitrary spatial triangulation $\mathcal{T}$ consisting of vertices in $I\!R^3$ (or more generally a simplicial polyhedron $\mathcal{P}$ when the triangulation in closed) with possibly normal vectors at the vertex points. Their algorithm constructs a C^1 continuous mesh of real implicit polynomial surface patches over $\mathcal{T}$ or $\mathcal{P}$. The scheme is local (each patch has independent free parameters) and there is no local splitting. The algorithm first converts the given triangulation or polyhedron into a curvilinear wireframe with at most cubic parametric curves which C^1 interpolate all the vertices. The curvilinear wireframe is then fleshed to produce a single implicit surface patch of degree at most 7 for each triangular face $\mathcal{T}$ of $\mathcal{P}$. If the triangulation is convex then the degree is at most 5. Furthermore, the C^1 interpolation scheme is local in that each triangular surface patch has independent degrees of freedom which may be used to provide local shape control. Extra free parameters may be adjusted and the shape of the patch controlled by using weighted least squares approximation from additional points and normals, generated locally for each triangular patch. See also Figure 2.1. Similar techniques exist for parametrics[23, 36, 43] however the geometric degree of the solution surfaces tend to be very high.

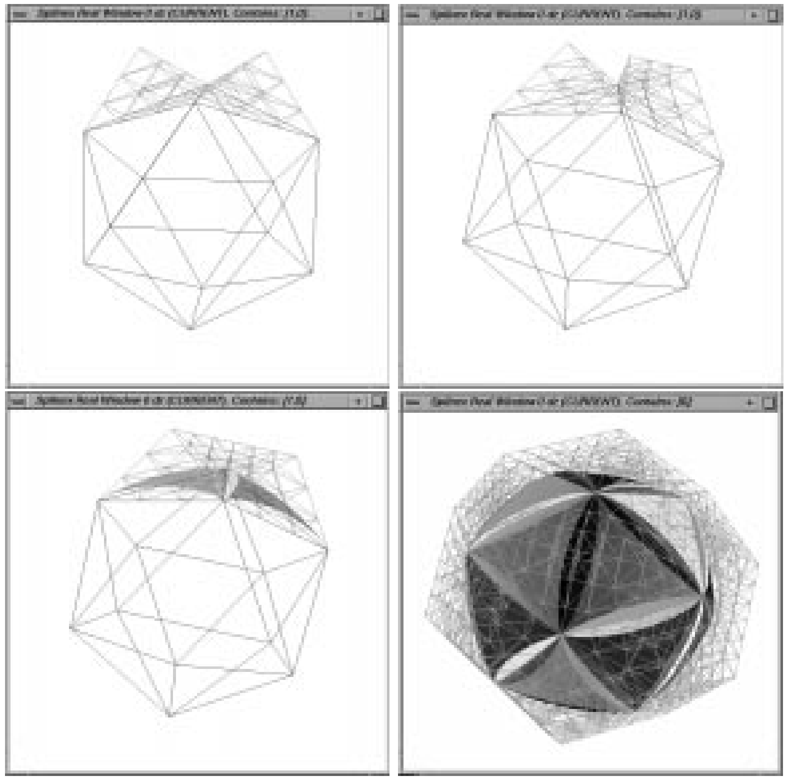


Figure 3.2: The construction of a simplicial hull and C^1 cubic A-patches interpolating the vertices of the input icosahedron.

3 Simplex and Box Based Schemes

In a *simplex* based approach one first constructs a tetrahedral mesh (called the simplicial hull) conforming to a surface triangulation $\mathcal{T}$ of a polyhedron $\mathcal{P}$. See Figure 3.2. The implicit piecewise polynomial surface consists of the zero set of a Bernstein-Bézier polynomial defined within each tetrahedron (simplex) of the simplicial hull. A simplex based approach enforces continuity between adjacent patches by enforcing that vertex/edge/face-adjacent trivariate polynomials are continuous to each other.

Similar to the trivariate interpolation case, Powell-Sabin or Clough-Tocher splits are used to introduce degree-bounded vertices to prevent the continuity system from propagating globally. Such splitting, however, could result in a large number of patches. A full trivariate Clough-Tocher split would split a tetrahedron into 12 sub-tetrahedra. However, as only the zero set of the polynomial is of interest, one does not need a complete mesh covering the entire space. Furthermore, a vertex does not need to be fully covered by the trihedral corners of the incident tetrahedra. A "incomplete" vertex helps introduce degree-bounded vertices as well. An effective simplicial hull construction of this kind first appears in Dahmen [19], and subsequently developed and used by Guo, Lodha, Dahmen and Thamm-Scharr, Bajaj, Chen and Xu[9, 20, 24, 29].

Given a triangulated polyhedron, the simplicial hull construction builds a tetrahedron on each triangular face (sometimes a pair of tetrahedra one on each side of a face). See Figure 3.4. The surface is set to interpolate the three "bottom" vertices of the face tetrahedron. Hence a tetrahedron (over an edge) is enough to fill in the gap between tetrahedra built on adjacent faces. See Figure 3.2. The "top" of a face tetrahedron is thus degree-bound to just four. Hence, a face Powell-Sabin split or a face

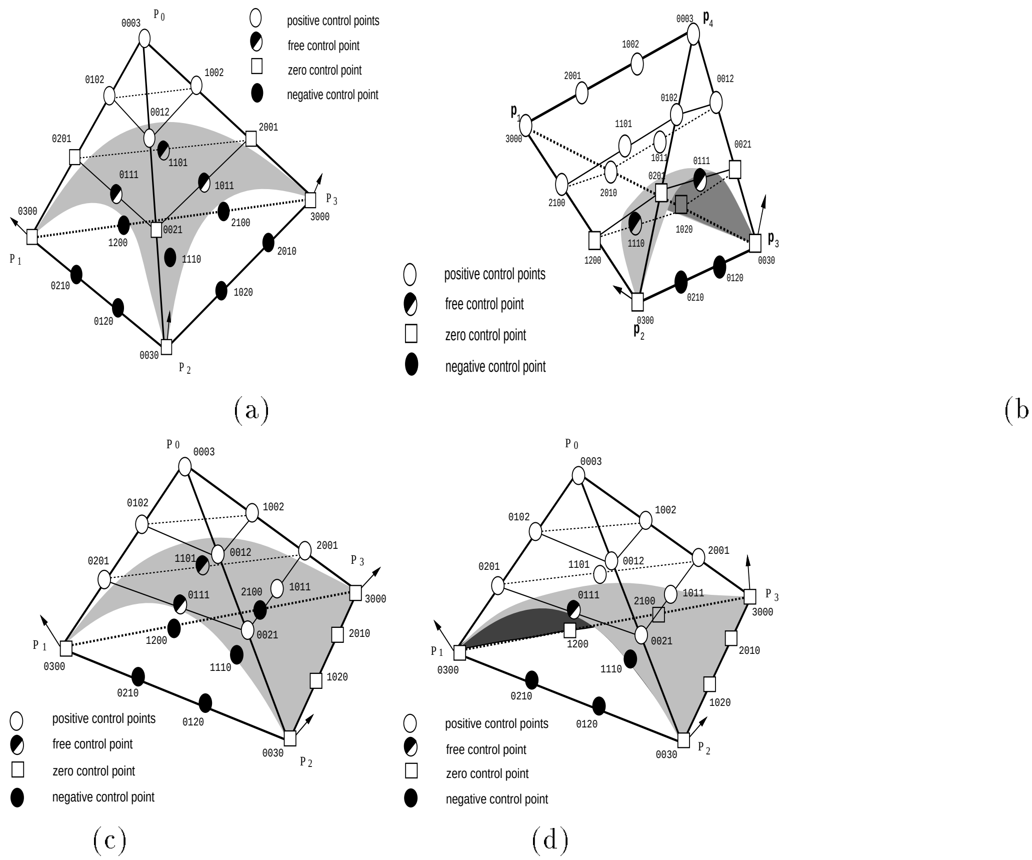


Figure 3.3: A-patches defined by a single change of coefficients in preferred directions. (a) A three sided A-patch tangent at p_1, p_2, p_3. (b) A degenerate four sided A-patch tangent to face $[p_1p_2p_4]$ at p_2 and $[p_1p_3p_4]$ at p_3. (c) A three sided A-patch interpolating edge $[p_2p_3]$. (d) A three sided A-patch interpolating edges $[p_2p_3]$ and $[p_1p_3]$.

Clough-Tocher split (the bivariate analogue to the trivariate case), suffices. In a face Clough-Tocher split, a tetrahedron is split into only 3 sub-tetrahedra and therefore the number of micro patches are greatly reduced. An edge tetrahedron is usually split into two in a C^1 or C^2 scheme.

A disadvantage of an interpolating simplicial hull is that, as the surface is pinned to the vertices of the hull, a modification of the surface would involve both changes of the coefficients and the simplicial hull, whereas if the surface does not have to pass through any hull vertex, surface modification can be done solely by changing the coefficients of the polynomial.

3.1 Smooth Interpolation of a Polyhedron with C^1 A-patches

Dahmen [19] presents a simplicial hull scheme for constructing C^1 continuous piecewise quadric surface patches for a triangulation $\mathcal{T}$ of a polyhedron $\mathcal{P}$. In his simplicial hull construction, each triangular face is covered by a tetrahedron. Similar to the Powell–Sabin split [40], the tetrahedron is then face-split and replaced by six subtetrahedra, each of which defines a micro quadric triangular patch. Additional simplices are then used to fill in the gaps between the simplices built on adjacent triangular faces. Dahmen's technique however works only if the original triangulation of the data set allows a transversal system of planes, and hence is quite restricted. A major contribution of this scheme, however, is the

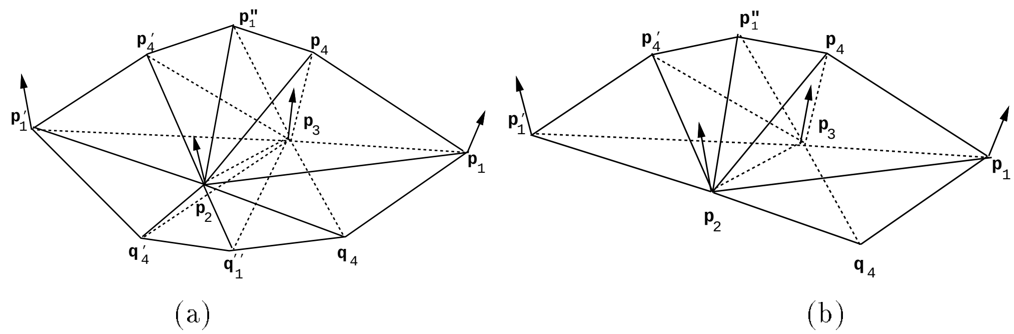


Figure 3.4: The construction of double tetrahedra for (a) adjacent 'non-convex'/'non-convex' faces and (b) 'convex'/'non-convex' faces

construction of the simplicial hull.

Guo [24] uses cubics in an interpolating simplicial hull approach, to create free–form geometric models and enforces monotonicity conditions on a cubic polynomial along the direction from one vertex to a point of the opposite face of the vertex. This condition is difficult to satisfy in general, and even if this condition is satisfied, one still cannot avoid singularities on the zero contour. A face Clough-Tocher split is used to subdivide each tetrahedron of the simplicial hull. Dahmen and Thamm-Scharr [20] independently develop a similar scheme utilizing a single cubic patch per tetrahedron at the cost of less locality and extra perturbation when adjacent faces of $\mathcal{P}$ are coplanar.

Moore and Warren [33] extend the marching-cubes box based scheme to a generic simplex-based scheme, and compute a C^1 piecewise triquadratic B-spline approximation. An effective technique called "signed distance" is used to prevent the multiple-sheeted problem with implicit surfaces when employing least square approximation. Auxiliary data points are evenly added in the domain dpace with values being the signed distances to the input data points. The technique, although a heuristic rather than a guarantee, works quite well in preventing unwanted branches. However, the auxiliary data points with values of signed distances forces the surface patch to be monotonic in one direction, a condition which is not necessary and overly constraining.

Lodha [29] constructs low degree surfaces with dual parametric and implicit representations and investigates their properties. A method is described for creating a quadratic triangular parametric Bézier surface patch which is the parametric dual of an implicit quadric surface. Another method is described for creating a biquadratic tensor product Bézier surface patches which is the parametric dual of an implicit cubic surface. The resulting patches satisfy all the standard properties of parametric Bézier surfaces, including interpolation of the corners of the control polyhedron and the convex hull property.

Papers [13, 19, 20, 24, 25, 33], propose heuristics based on trivariate coefficient monotonicity, and least square approximation of convex quadrics to circumvent the multiple sheeted and singularity problems of implicit patches. Bajaj, Chen and Xu [7, 9] construct 3- and 4-sided A–patches that are implicit surfaces in Bernstein–Bézier(BB) form that are guaranteed smooth and single-valued. See Figure 3.3. For a three–sided A-patch any line segment passing through a vertex of the tetrahedron and its opposite face intersects the surface patch at most once ; and for a four–sided A-patch where any line segment connecting two points on opposite edges intersects the patch at most once. Instead of having patch coefficients be monotonically increasing or decreasing there is now only a single sign change condition. There are also free parameters for both local and global shape modification of the patch. Papers [7, 9] also show how A-patches of cubic degree can be used to interpolate the vertices of

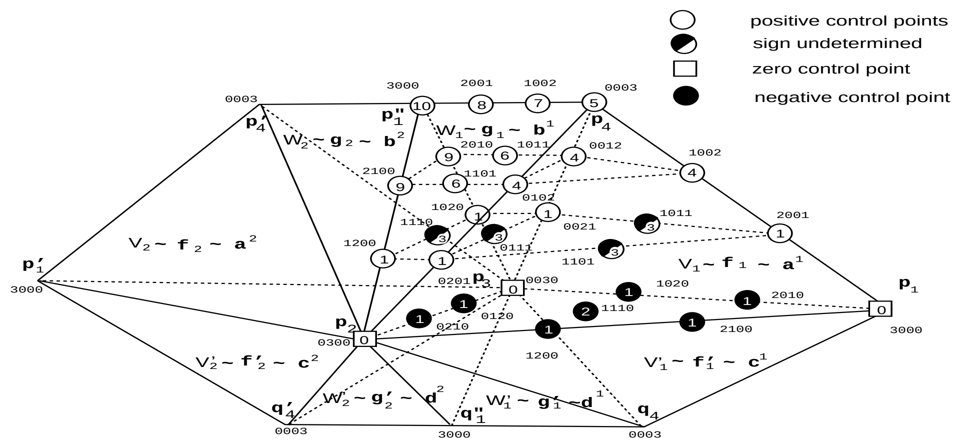


Figure 3.5: Adjacent double tetrahedra, coefficient signs and C^1 continuity relations for two cubic A-patches defined on non-convex adjacent faces

any polyhedron $\mathcal{P}$ and yield a globally C^1 surface. This scheme is also extended to a C^2 version using A-patches of degree five [8]. Details are provided in the next subsection.

In these algorithms [9, 7] they first specify unique "normals" (tangent planes) on the vertices of $\mathcal{P}$, then build a simplicial hull surrounding the surface triangulation T of $\mathcal{P}$ and satisfying vertex tangent plane containment, and finally construct cubic A-patches within each tetrahedron of the simplicial hull. Different configurations of vertex "normals" for edges and faces of $\mathcal{T}$ are categorized as 'convex' and 'non-convex' . The edges and faces together with their normals are thus tagged as 'convex' and 'non-convex' . As part of the simplicial hull, a single tetrahedron is constructed for a 'convex' face while a pair of tetrahedra are constructed (one on each side of the face) for a 'non-convex' face (Figure 3.4). C^1 continuity conditions are next set up and satisifed between coefficients (control points) of all adjacent tetrahedra in the simplicial hull. Figure 3.5 shows the control points, their signs and their relations (like numbers) by C^1 continuity conditions in neighboring edge and face tetrahedra for the most difficult of cases, viz. two adjacent 'non-convex' faces. The C^1 continuity conditions are all linear and shown to be always satisfiable while maintaining the single sign change conditions of the A-patches. Details are in [9]. The resulting mesh of A-patches is thus guaranteed to be globally C^1 continuous. They also show how to adjust the free parameters of the A-patches to achieve both local and global shape control (bottom of Figure 3.6).

They use a single cubic A-patch per face of T except for the following two special cases. For a 'non-convex' face, if additionally the three inner products of the face normal and its three adjacent face normals have different signs, then in this case one needs to subdivide the face using a single face Clough-Tocher split, yielding C^1 continuity with the help of three cubic A-patches for that face. Furthermore for coplanar adjacent faces of T, they show that the C^1 conditions cannot be met using a single cubic A-patch for each face. Hence for this case they again use face Clough-Tocher splits for the pair of coplanar faces yielding C^1 continuity with the help of three cubic A-patches per face.

3.2 Smooth Interpolation with C^2 A-patches

Bajaj, Chen and Xu [8]. present a scheme for building a C^2 patch complex with quintic A-patches. Similar to the C^1 scheme, a simplicial hull Σ is constructed conforming to a face triangulated polyhedron $\mathcal{P}$ and a quintic A-patch is defined within each tetrahedron of Σ and made C^2 continuous across their

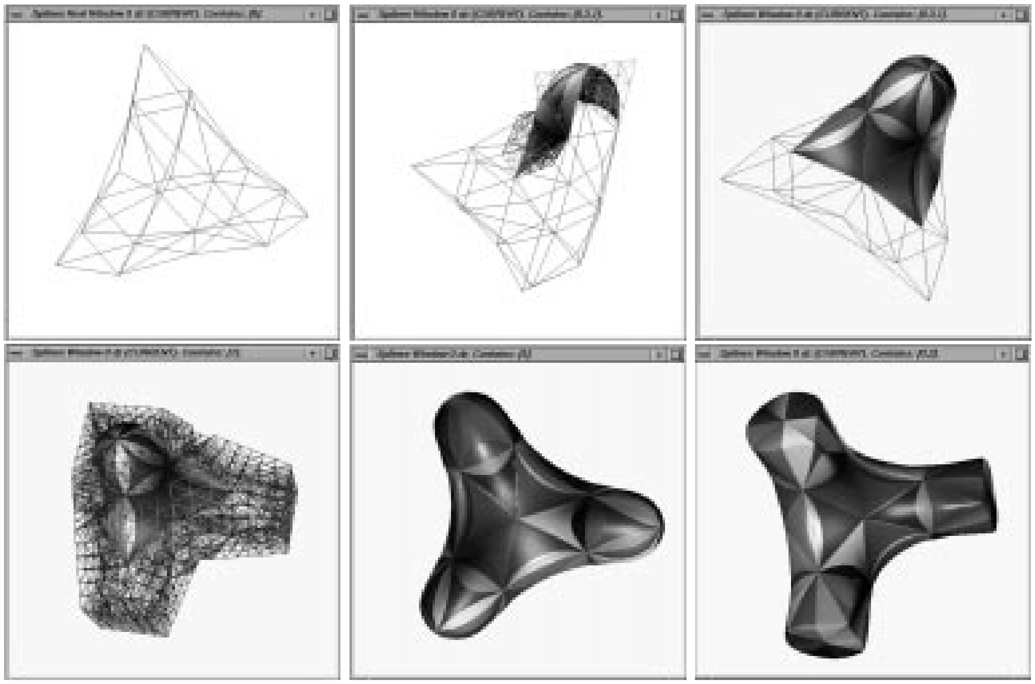


Figure 3.6: A face triangulated polyhedron, the simplicial hull and different interpolating C^1 cubic A-patches. Shape modification is achieved by adjusting the free parameters of the A-patches.

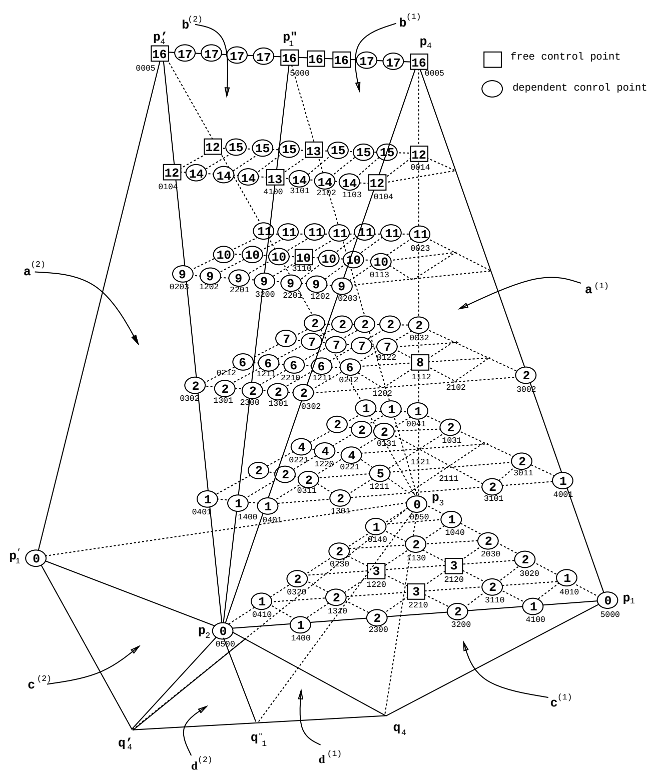


Figure 3.7: Adjacent double tetrahedra, coefficient signs and C^2 continuity relations for two quintic A-patches defined on 'non-convex' adjacent faces

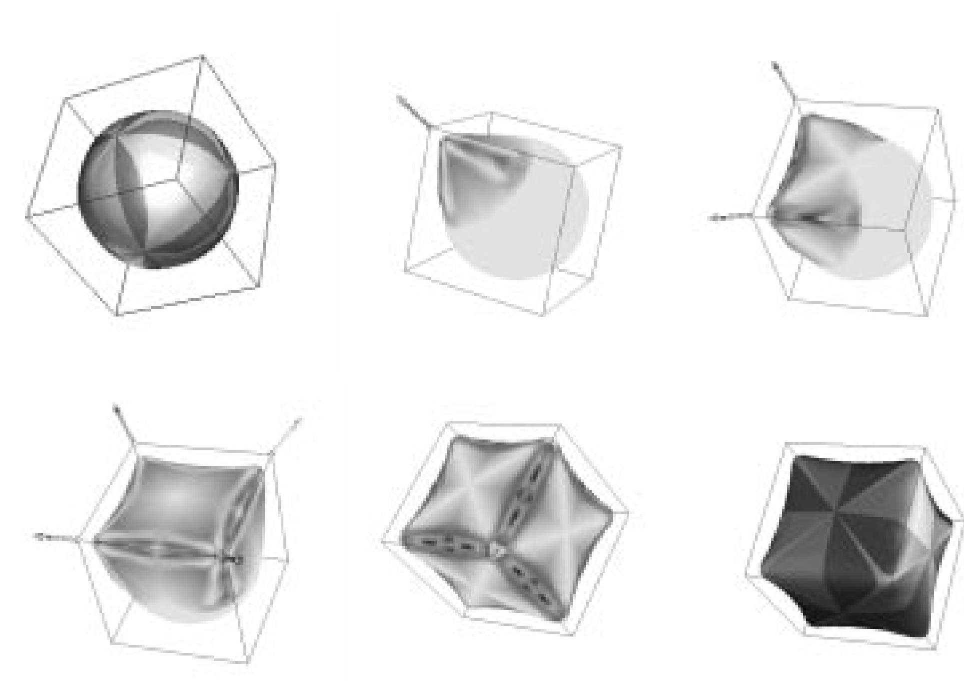


Figure 3.8: Interactive deformation of a sphere defined by C^2 continuous quintic A-patches.

share boundaries. The one-sign change condition, which is used in the C^1 scheme, turns out to be too constraining, and is thus relaxed by subdivisions for the C^2 case. The C^2 continuity conditions are all linear and involve groups of coefficients (control points) across common tetrahedral vertices, edges and faces. Figure 3.7 shows the control points, their signs and their relations (like numbered control points) by C^2 continuity conditions in neighboring edge and face tetrahedra for the most difficult of cases, viz. two adjacent 'non-convex' faces.

The number 0, 1, and 2 coefficients are given by the C^2 data values at the vertices. The number 4, 5, 6, 7, 9 , 11, and 17 coefficients are determined by C^2 continuity conditions and involve all the tetrahedra surrounding the vertex or edge. The number 3, 8, 12, 13, and 16 coefficients can be freely specified and adjusted for local or global shape control.

The number 0, 1, and 2 coefficients are given by the C^2 data values at the vertices. The number 4, 5, 6, 7, 9 , 11, and 17 coefficients are determined by C^2 continuity conditions and involve all the tetrahedra surrounding the vertex or edge. The number 3, 8, 12, 13, and 16 coefficients can be freely specified and adjusted for local or global shape control. One approach to setting default values for all the free coefficients is to first construct a desirable C^1 surface with cubic A-patches. Next one degree-raises the entire simplicial hull to quintic patches. These values become default for all the control points, some of which are subsequently modified by the C^2 continuity conditions as specified above. Figure 3.8 shows the modeling and interactive deformation of a sphere defined by C^2 continuous quintic A-patches by adjusting groups of free control points. Starting from the free-form model of the sphere, Figure 3.8 (a) the surface is first pulled towards a vertex of the cube (Figure 3.8 (b)). The Mean curvature map is displayed. The surface is next pulled towards an edge of the cube (Figure 3.8 (c)). The Gaussian curvature map is displayed. The surface is subsequently pulled towards a face of the cube showing the Mean curvature map and then towards all the other faces (Figure 3.8 (e)) showing

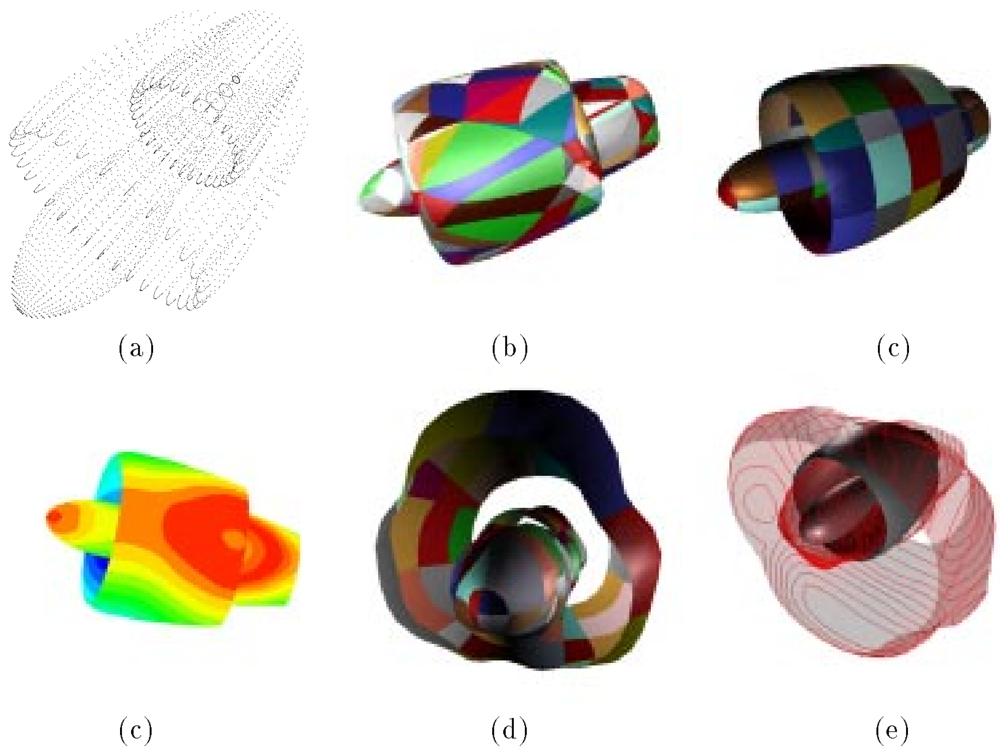


Figure 3.9: Jet engine model and associated pressure (scalar) field reconstruction from scattered data (a) Input point data (b) reconstructed model with cubic A-patches within a 3D triangulation (c) reconstructed model with bicubic A-patches over a box decomposition (d) isocontours of a pressure field displayed on the jet engine surface (e) reconstructed model of the pressure field with bicubic A-patches over a box decomposition (f) pressure field with iso-contours displayed surrounding the jet engine

the Gaussian curvature map. Figure 3.8 (f) show the final deformed surfaces with shades highlighting the C^2 continuous A-patches.

3.3 Smooth Reconstruction from Scattered Data

The problem here is the reconstruction of surfaces and scalar fields defined over it (surface-on-surface), from scattered trivariate data. The data points are assumed sampled from the surface of a 3D object, and the sampling is assumed to be *dense* for unambiguous reconstruction. Laser range scanners are able to produce a dense sampling, usually organized in a rectangular grid, of an object surface. Some 3D scanners are also able to measure the RGB components of the object color (i.e. three scalar fields) at each sampled point. When the object has a simple shape, this grid of points can be a sufficient representation. However, multiple scans are needed for objects with more complicated geometry, e.g. objects with holds, handles, pockets cannot be scanned in a single pass. Other applications, for example recovering the shape of a bone from contour data extracted from a CT scan, require reconstruction of a surface from data points organized in slices. The approach of considering the input points as unorganized has the advantage of generating cross-derivatives by a uniform treatment of all spatial directions.

Figure 4.10: Corner Cuts, Inner Simplicial Hull and C^1 smooth A-patches

Bajaj, Bernardini and Xu [6] reconstruct the sampled surface using A-patches. Their scheme effectively utilizes an incremental Delaunay 3D triangulation for a more adaptive fit; the dual 3D Voronoi diagram for efficient point location in signed distance computations and cubic implicit surface patches. Furthermore, in the same time they also compute a C^1 smooth approximation of the sampled surface-on-surface. Bajaj, Bernardini and Xu [5] have also developed a similar method based on tensor-product Bernstein-Bézier patches.

A different, three-step solution is given by Hoppe et al.[27, 28, 26]. In the first phase, a triangular mesh that approximates the data points is created. In a second phase, the mesh is optimized with respect to the number of triangles and the distance from the data points. A third step constructs a smooth surface from the mesh.

The problem of modeling and visualizing function-on-surface arises in several physical analysis application areas: characterizing the rain fall on the earth, the pressure on the wing of an airplane and the temperature on the surface of a human body. A number of methods have been developed for dealing with this problem.

Currently known approaches for approximating function-on-surface data however possess restrictions either on the domain surfaces or the surface-on-surface. The domain surfaces are usually assumed to be spherical, convex or genus zero. The function-on-surface is not always polynomial [16, 35], or rather higher order polynomial [42], or a large number of pieces [1] compared to the approach of [6]. The method of [1] is a C^1 Clough-Tocher scheme that splits a tetrahedron into 4 subtetrahedra, uses quintic polynomials and requires C^2 data on the vertices of each subtetrahedron. Another Clough-Tocher scheme [44] requires only C^1 data at the vertices, for again constructing a C^1 function which is a cubic polynomial over each subtetrahedron, however splits the original tetrahedron into 12 pieces. A C^1 scheme [42] that does not split each tetrahedron uses degree 9 polynomials and requires C^4 data at the vertices. In extending the method of [42] to a C^2 scheme, requires degree 17 polynomials and C^8 data at the vertices of each tetrahedron. Compared to these approaches, the C^1/C^2 construction of [15] has no splitting and uses much lower degree polynomials (cubic/quintic) requiring only C^1/C^2 data respectively, at the vertices of each tetrahedron.

4 Subdivision Based Schemes

The third class of popular approaches of free-form modeling of shapes are in conjunction with subdivision methods. The input mesh is averaged and split before surface patches are fit in. The subdivision steps are sophistically designed so that the surface patches are bereft of difficulties such as *curve compatibility* and/or *vertex enclosure*. The idea behind subdivision is somewhat similar to that of

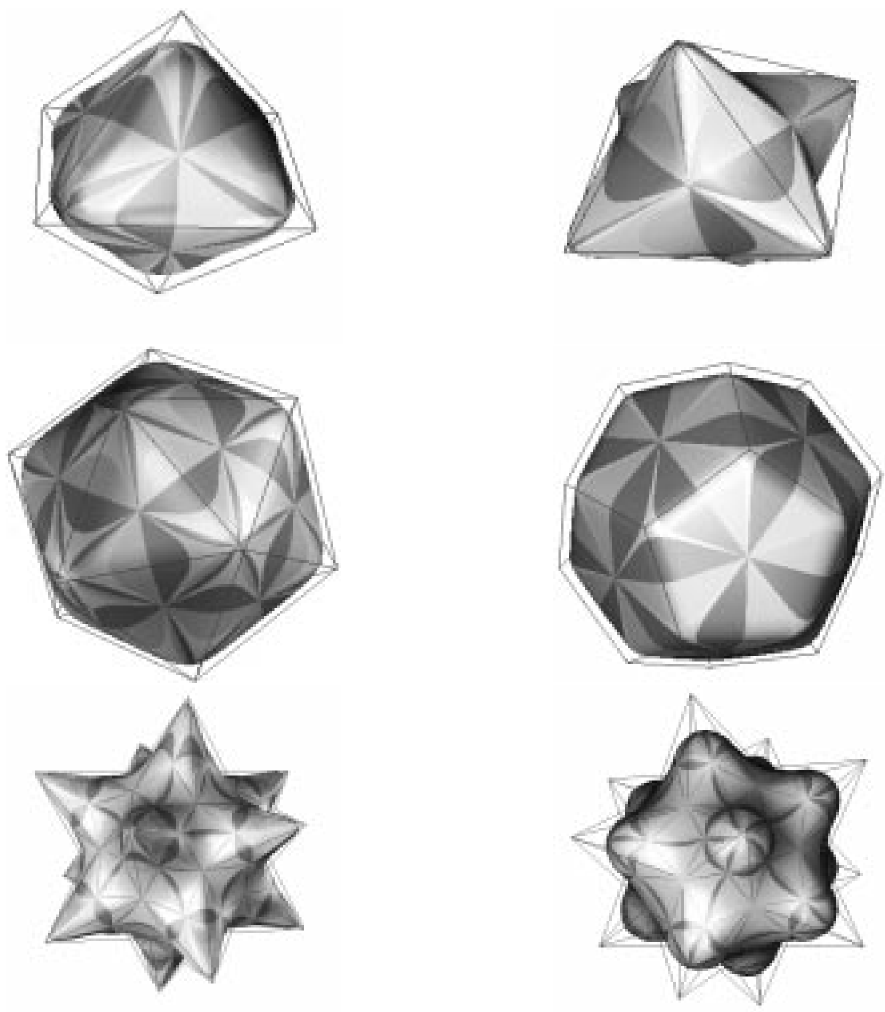


Figure 4.11: Smoothed Octahedron, icosahedron, dodecahedron and stellated dodecahedra using cubic A-patches. Corners of the dodecahedron are trihedral. Those of the others are non-trihedral.

Clough-Tocher and Powell-Sabin interpolants: introducing new vertices that are degree-bounded. The earliest of these approaches are the recursive subdivision schemes of Chaikin, Doo, Sabin, Catmull and Clark [17, 18, 21, 22]. These algorithms generate C^1 surfaces that interpolate the centroids of all faces at every step of subdivision.

Nasri [34] describes a recursive subdivision surface scheme that is capable of interpolating points on irregular networks as well as normal vectors given at these points. The subdivision scheme developed by Loop [30] splits each triangle of a triangular mesh into four triangles. Each new vertex is positioned using a fixed convex combination of the vertices of the original mesh. The final limit surface is tangent plane continuous. Hoppe et al. [26] extends Loop's method to incorporate sharp edges into the final limit surface. The vertices of the initial polyhedron are tagged as belonging on a face, edge, or vertex of the final limit surface. Based on this tag different averaging masks are used to produce new polyhedra. Reif [41] presents a unified approach to subdivision algorithms for meshes with arbitrary topology and gives a sufficient condition for the regularity of the surface. The existence of a smooth regular parameterization for the generated surface near the point is determined from the leading eigenvalues

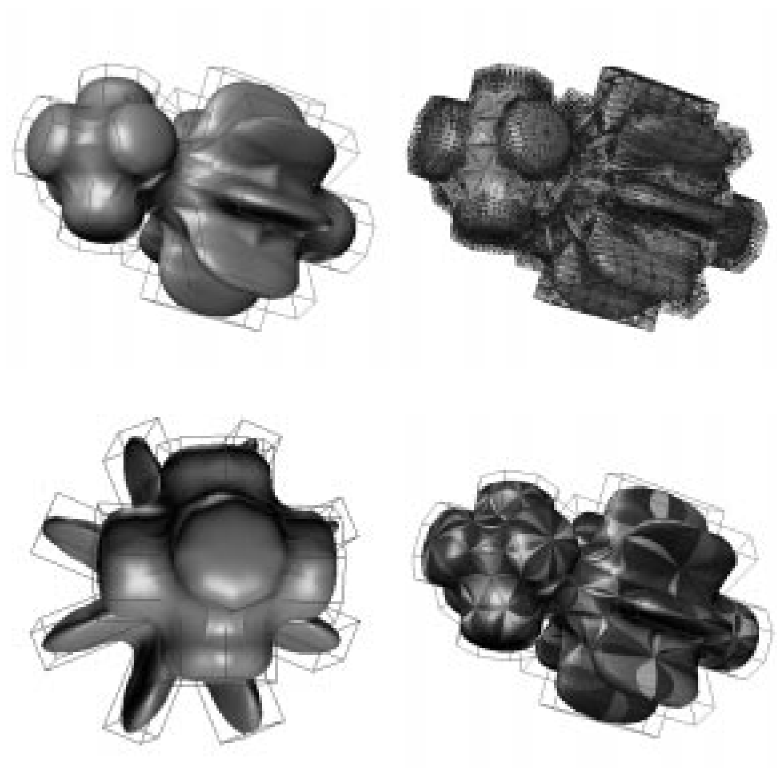


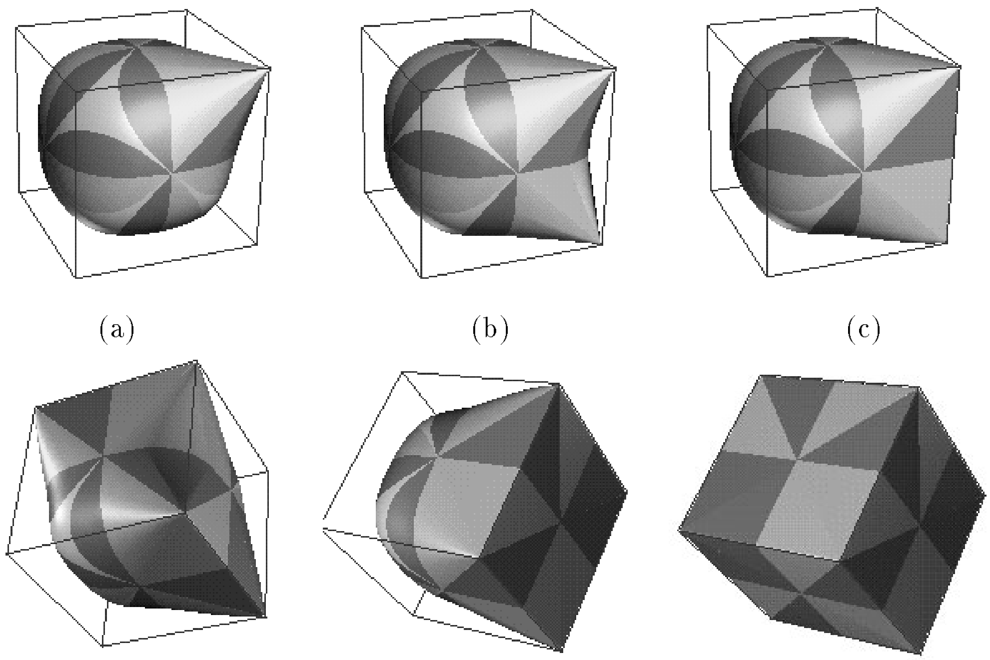
Figure 4.12: Subdivision based smoothing of a satellite-like object with cubic A-patches. Upper right also shows the simplicial hull, while bottom right shows the individual cubic A-patches.

of the subdivision matrix and an associated characteristic map.

As subdivision techniques do not necessarily yield surfaces with analytical representations. However, by careful arrangement, after initial steps of subdivision, the mesh can be used as control nets for piecewise parametric Bézier patches or B-splines [31, 32, 37, 38, 39] or using implicit surface A-patches[10, 11]. The subdivision flavor is thus "taken over" by the subdivision algorithms of the particular parametric representation.

Bajaj, Chen and Xu [10, 11] construct an "inner" simplicial hull after one step of subdivision of the input polyhedron $\mathcal{P}$. See Figure 4.10. Similar to traditional subdivision schemes, $\mathcal{P}$ is used as a control mesh for free-form modeling while an inner surface triangulation $\mathcal{T}$ of the hull can be considered as the second level mesh. Both a C^1 smooth mesh with cubic A-patches and a C^2 smooth mesh with quintic A-patches can be constructed to approximate the given polyhedron $\mathcal{P}$. See Figures 4.11, 4.12.

Also similar to traditional schemes, simple editing of $\mathcal{P}$ can impose some interesting constraints on the surface, such as interpolating a line or a region. Furthermore, the free wieghts of neighboring patches and cutting ratios control how deep a corner is smoothed. For a trihedral corner, we control the shape of the neighboring surface by changing only the free weights in the face patch built at the corner and the surrounding edge patches. For a non-trihedral corner, we control the shape of the neighboring surface by changing the corner cutting ratio. See Figures 4.11 and 4.12. C^0 and C^1 features can be mixed into the same model, by allowing zero cutting ratios and patches with singular vertices/edges (weights around the vertices/edges are all zero, compared to coincident control points in the parametric case). Figure 4.13 shows how C^0 and C^1 features can be mixed to approximate a cube in different shapes.



(a) (b) (c)

(d) (e) (f)

Figure 4.13: Modeling with singular A-patches. (a) Interpolating a vertex with a singular point. (b) Interpolating two vertices. (c) Interpolating an edge with a singular edge on the surface. (d) Interpolating two edges. (e) Interpolating a face of a cube. (f) The A-patch surface degenerates into the cube. All the edges are now singular.

5 Conclusion

All the algorithms of the previous sections have been implemented in the SPLINEX and SHILP toolkits of the distributed and collaborative geometric design environment SHASTRA [2]. SHILP is an X11 and Motif based, interactive solid modeling system, which is used to create a simplicial (face triangulated) polyhedral model of a desired shape. This model could also be the triangulation of an arbitrary surface in three dimensions. This triangulation is C^1 smoothed by client/server calls to SPLINEX processes using inter process communication. SPLINEX is an X11 and Motif based, interactive surface modeling toolkit for arbitrary algebraic surfaces (implicit or parametric) in BB form. It allows for the creation of simplex chains (for example, the simplicial hull of the triangulation) and the interactive change of control points of the A-patches for shape control. SPLINEX also has the ability to distribute its rendering tasks (for the display of the individual A-patches) on a network of workstations, to achieve maximal display parallelism.

References

[1] P. Alfeld. A trivariate Clough-Tocher scheme for tetrahedral data. *Computer Aided Geometric Design*, 1:169–181, 1984.

[2] V. Anupam and C. Bajaj. SHASTRA: Collaborative Multimedia Scientific Design. *IEEE Multimedia*, 1(2):39–49, 1994.

[3] C. Bajaj. Surface fitting with implicit algebraic surface patches. In H. Hagen, editor, *Topics in Surface Modeling*, pages 23–52. SIAM Publications, 1992.

[4] C. Bajaj. The Emergence of Algebraic Curves and Surfaces in Geometric Design. In R. Martin, editor, *Directions in Geometric Computing*, pages 1–29. Information Geometers Press, 1993.

[5] C. Bajaj, F. Bernardini, and G. Xu. Adaptive resconstruction of surfaces and surface-on-surface from dense scattered trivariate data. Technical Report Computer Science Technical Report, CS-95-028, Computer Sciences Department, Purdue University, 1994.

[6] C. Bajaj, F. Bernardini, and G. Xu. Automatic reconstruction of surfaces and scalar fields from 3D scans. In R. Cook, editor, *Annual Conference Series. Proceedings of SIGGRAPH 95*, pages 109–118. ACM SIGGRAPH, Addison Wesley, August 6-11 1995.

[7] C. Bajaj, J. Chen, and G. Xu. Free form surface design with A-patches. In *Proceedings of Graphics Interface '94, Banff, Canada*, pages 174–181, 1994.

[8] C. Bajaj, J. Chen, and G. Xu. Free form modeling with C^2 quintic A-patches. Presented in the Fourth SIAM Conference on Geometric Design, September 1995.

[9] C. Bajaj, J. Chen, and G. Xu. Modeling with cubic A-patches. *ACM Transactions on Graphics*, 14(2), April 1995.

[10] C. Bajaj, J. Chen, and G. Xu. *Smooth Low Degree Approximations of Polyhedra (Part 1)*. Manuscript, September 1995.

[11] C. Bajaj, J. Chen, and G. Xu. *Smooth Low Degree Approximations of Polyhedra (Part 2)*. Manuscript, September 1995.

[12] C. Bajaj and I. Ihm. Algebraic surface design with Hermite interpolation. *ACM Transactions on Graphics*, 11(1):61–91, January 1992.

[13] C. Bajaj and I. Ihm. C^1 Smoothing of Polyhedra with Implicit Algebraic Splines. *SIGGRAPH'92, Computer Graphics*, 26(2):79–88, July 1992.

[14] C. Bajaj, I. Ihm, and J. Warren. Higher order interpolation and least squares approximation using implicit algebraic surfaces. *ACM Transactions on Graphics*, 12(4):327–347, Oct. 1993.

[15] C. Bajaj and G. Xu. Modeling Scattered Function Data on Curved Surface. In J. Chen, N. Thalmann, Z. Tang, and D. Thalmann, editor, *Fundamentals of Computer Graphics*, pages 19 – 29, Beijing, China, 1994.

[16] R. E. Barnhill, K. Opitz, and H. Pottmann. Fat surfaces: a trivariate approach to triangle-based interpolation on surfaces. *Computer Aided Geometric Design*, 9:365–378, 1992.

[17] E. Catmull and J. Clark. Recursively Generated B-spline Surfaces on Arbitrary Topological Meshes. *Computer Aided Design*, 10(6):350–355, 1978.

[18] G. Chaikin. An algorithm for high-speed curve generation. *Computer Graphics and Image Processing*, 3:346–349, 1974.

[19] W. Dahmen. Smooth piecewise quadratic surfaces. In T. Lyche and L. Schumaker, editors, *Mathematical Methods in Computer Aided Geometric Design*, pages 181–193. Academic Press, Boston, Massachusetts, 1989.

[20] W. Dahmen and T-M. Thamm-Schaar. Cubicoids: modeling and visualization. *Computer Aided Geometric Design*, 10(2):89–108, Apr. 1993.

[21] D. Doo. A Subdivision Algorithm for Smoothing Down Irregular Shaped Polyhedrons. In *Proceedings of Interactive Techniques in Computer Aided Design, Bologna*, pages 157–165, 1978.

[22] D. Doo and M. Sabin. Behaviour of recursive division surfaces near extraordinary points. *Computer Aided Design*, 10(6):356–360, November 1978.

[23] G. Farin. Triangular Bernstein-Bézier Patches. *Computer Aided Geometric Design*, 3(00):83–127, 1986.

[24] B. Guo. Surface generation using implicit cubics. In N.M. Patrikalakis, editor, *Scientific Visualizaton of Physical Phenomena*, pages 485–530. Springer-Verlag,Tokyo, 1991.

[25] B. Guo. Non-splitting Macro Patches for Implicit Cubic Spline Surfaces. *Computer Graphics Forum*, 12(3):434 – 445, 1993.

[26] H. Hoppe, T. DeRose, T. Duchamp, M. Halstead, H. Jin, J. McDonald, J. Schweizer, and W. Stuetzle. Piecewise smooth surface reconstruction. *Computer Graphics*, 28:295–302, 1994.

[27] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *Computer Graphics*, 26(2):71–78, 1992.

[28] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Mesh optimzation. *Computer Graphics*, 27:19–26, Aug 1-6 1993.

[29] S. Lodha. *Surface Approximation with Low Degree Patches with Multiple Representations.* PhD thesis, Computer Science, Rice University, Houston, Texas, 1992.

[30] C. Loop. Smooth Subdivision based on triangles. Master's thesis, University of Utah, 1987.

[31] C. Loop. A G^1 triangular spline surface of arbitrary topological type. *Computer Aided Geometric Design*, 11:303–330, 1994.

[32] C. Loop. Smooth Spline Surfaces over Irregular Meshes. In *Proceedings of SIGGRAPH'94*, pages 303–310, July 1994.

[33] D. Moore and J. Warren. Approximation of dense scattered data using algebraic surfaces. In *Proceedings of the twenty-fourth Hawaii International Conference on System Sciences*, volume 1, pages 681–690, Kauai, Hawaii, 1991. IEEE, IEEE Computer Society Press, Los Alamitos, California.

[34] A. Nasri. Surface interpolation on irregular networks with normal conditions. *Computer Aided Geometric Design*, 8:89–96, 1991.

[35] G. Nielson, T. Foley, B. Hamann, and D. Lane. Visualizing and Modeling Scattered Multivariate Data. *IEEE Computer Graphics And Applications*, 11:47–55, 1991.

[36] J. Peters. Smooth interpolation of a mesh of curves. *Constructive Approximation*, pages 221–246, July 1991.

[37] J. Peters. Free-form surface splines. Technical Report CSD-TR-93-019, Computer Sciences Department, Purdue University, March 1993.

[38] J. Peters. Smooth free-form surface over irregular meshes generalizing quadratic splines. *Computer Aided Geometric Design*, pages 347–361, October 1993.

[39] J. Peters. C^1 surface splines. *SIAM Journal of Numerical Analysis*, 32(2):645–666, April 1995.

[40] M. Powell and M. Sabin. Piecewise quadratic approximations on triangles. *ACM Transactions on Mathematical Software*, 3:316–325, 1977.

[41] U. Reif. A unified approach to subdivision algorithms. Technical report, Mathematisches Institut A, Universität Stuttgart, 1992. Preprint 92-16.

[42] K. Rescorla. C^1 Trivariate Polynomial Interpolation. *Computer Aided Geometric Design*, 4:237–244, 1987.

[43] R. Sarraga. G^1 Interpolation of generally unrestricted Cubic Bezier Curves. *Computer Aided Geometric Design*, 4(00):23–29, 1987.

[44] A. Worsey and G. Farin. An n-dimensional Clough-Tocher element. *Constructive Approximation*, 3(2):99–110, 1987.