Computational Methods for Geometric Processing. Applications to Industry

A. Iglesias, A. Gálvez, and J. Puig-Pey

Department of Applied Mathematics and Computational Sciences, University of Cantabria, Avda. de los Castros, s/n, E-39005, Santander, Spain
iglesias@unican.es

Abstract. This paper offers a unifying survey of some of the most relevant computational issues appearing in geometric processing (such as blending, trimming, intersection of curves and surfaces, offset curves and surfaces, NC milling machines and implicitization). Applications of these topics to industrial environments are also described.

1 Introduction

Geometric processing is defined as the calculation of geometric properties of already constructed curves, surfaces and solids [5]. In its most comprehensive meaning, this term includes all the algorithms that are applied to already existing geometric entities [16]. As pointed out in [5], since geometric processing is intrinsically *hard* there is neither a unified approach nor "key developments" such as the Bézier technique [60] for design. On the contrary, the literature on geometric processing is much more disperse among different sources. The aim of the present paper is precisely to offer a unifying survey of some of the most relevant computational issues appearing in geometric processing as well as a description of their practical applications in industry. Obviously, this task is too wide to be considered in all its generality, and some interesting topics in geometric processing, such as curvature analysis, contouring, curve fairing, etc. have been omitted. We restrict ourselves to blending (Section 2.1), trimmed surfaces (Section 2.2), curve and surface intersection (Section 2.3), offset curves and surfaces (Section 2.4), NC milling technology (Section 2.5) and implicitization (Section 2.6).

2 Some Geometric Processing Topics

2.1 Blend Surfaces

We use the term *blending* to mean the construction of connecting curves and surfaces and the rounding off of sharp corners or edges. Thus, we talk about *superficial blending* to indicate that no explicit mathematical formula is available. It appears in the production process [87,88], in procedures such as round off a corner or edge with radius r. The blend described by additional surfaces connecting smoothly some given surfaces is usually referred to as *surface blending*,

V.N. Alexandrov et al. (Eds.): ICCS 2001, LNCS 2073, pp. 698–707, 2001.

while the *volumetric blending* is used to mean the combination of objects in a solid modeling system (see [34], Chapter 14).

The most interesting blend for our purposes is that in parametric form. To this aim, a number of methods are described, from *interactive methods* [4,56] to *automatic methods* based on calculation of intersections of offset surfaces to the two given surfaces [46,56]. Blending of tensor product B-spline or Bézier surfaces (see [18,20,34] for a definition) are analyzed, for example, in [4,12,24,45]. See also [86] for blending algebraic patches and [28,66] for implicit surfaces.

2.2 Trimmed Surfaces

Trimmed surfaces have a fundamental role in CAD. Most complex objects are generated by some sort of trimming/scissoring process, i.e. unwanted parts of the rectangular patch are trimmed away (see Fig. 1). Trimmed patches are also the result of Boolean operations on solid objects bounded by NURBS surfaces (see [19,61,68] for a definition). In the computer-aided design pipeline, the trimmed patch undergoes a number of processes such as rendering for visualization, cutter path generation, area computation or rapid prototyping, also known as *solid hard copy* [79]. For visualization, trimmed surfaces are rendered in two stages [67,77]: the surface is divided into a number of planar tesselants (triangles or other polygons), which are rendered using standard methods for planar polygons. Other algorithms for tessellation of trimmed NURBS surfaces can be found in [63] (and references 6-19 therein).

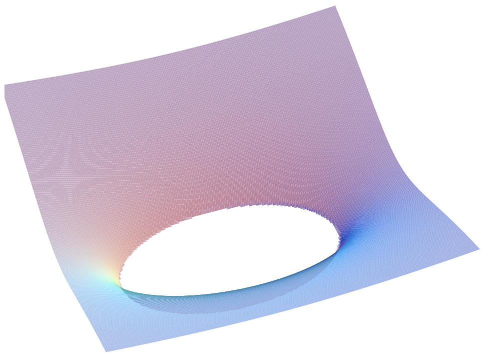

Fig. 1. Example of a trimmed NURBS surface

2.3 Intersection of Curves and Surfaces

In many applications, computation of the intersections of curves and surfaces is required. Among them, we quote smooth blending of curves and surfaces

(Section 2.1), the construction of contour maps to visualize surfaces, Boolean operations on solid bodies and determination of self-intersections in offset curves and surfaces (Section 2.4).

There exists a significant body of literature on the calculation of intersections of two parametric surfaces [1,6,18,23,30,76] (see also [17] for a more exhaustive bibliography). Recent developments include the possibility of handling intersection singularities [10,49]. Intersections of offsets (see Section 2.4) of parametric surfaces are analyzed in [85]. This problem is often of great interest: for instance, a blend surface (see Section 2.1) of two surfaces can be constructed by moving the center of a sphere of given radius along the intersection curve of two surfaces that are offset from the base surfaces by the radius of the sphere.

However, there has been no known algorithm that can compute the intersection curve of two arbitrary rational surfaces accurately, robustly and efficiently [34]. In addition, it is known that two surface patches intersect in a curve whose degree is much higher than the parametric degree of the two patches. Thus, two bicubic patches intersect in a curve of degree 324!!! Fortunately, the situation is better when we restrict the domain of input surfaces to simple surfaces (planes, quadrics and tori, i.e. the so-called *CSG primitives*) [43,53,78]. These surfaces are important in conventional solid modeling systems for industry, since they can represent a large number of mechanical parts of a car, ship, plane, etc.

As noticed in the previous paragraph, algorithms for intersections strongly depend on the general form of the curves and surfaces we are dealing with. If both objects are given in implicit form, such an intersection is found by solving a system of nonlinear equations. This can be achieved through numerical methods [23], differential geometry [3] or a combination of geometric and analytic methods [54]. If the objects are described as free-form curves and surfaces [18,20,23,34, 61,68], methods can be grouped into several categories: *algebraic methods*, based on implicitization (Section 2.6), *subdivision methods*, which divide the objects to be intersected into many pieces and check for intersections of the pieces [6,9,13, 26,27,42,47,91], *discretization methods*, which reduce the degrees of freedom by discretizing the surface representation in several ways, such as contouring [14,58, 81] or parameter discretization [6,35], *hybrid methods*, which combine subdivision and numerical methods [82,90], etc.

2.4 Offset Curves and Surfaces

Offsetting is a geometric operation which expands a given object into a similar object to a certain extent. In general, we deal with offset curves and surfaces, which are also curves and surfaces at a constant distance d from a given initial curve or surface. Several methods for the computation of the offsets for curves are compared in [15]. As pointed out in [59], offsetting general surfaces is more complicated, and an offset surface is often approximated [21], although this approximation becomes inaccurate near its selfintersecting area [2,59]. Another approach for computing offsets of NURBS curves and surfaces is given in [62].

Offsetting has various important applications [69]. For example, if the inner surface of a piece is taken as the reference surface, the outer surface can be

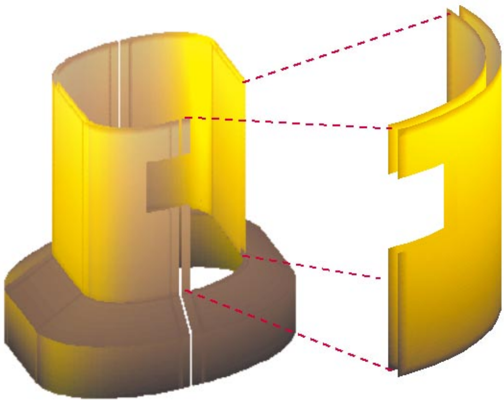

Fig. 2. Application of the offset operation: the outer surface of the piece is the offset of the inner trimmed NURBS surface

mathematically described by an offset surface corresponding to a distance equal to the thickness of the material (see Fig. 2). Offsets also appear in cutter-path generation for numerical control machine tools: pieces of a surface can be cut, milled or polished using a laser-controlled device to follow the offset. In the case of curves, they can be seen as the envelope corresponding to moving the center of a circle of radius d along the initial curve. This allows to define both the inside and outside offset curves, with applications in milling. Finally, they are fundamental tools (among others) in the constant-radius rounding and filleting of solids or in tolerance analysis, for definition of tolerance zones, etc.

We should note, however, that offset curves and surfaces lead to several practical problems. Depending on the shape of the initial curve, its offset can come closer than d to the curve, thus causing problems with *collisions*, for instance, when steering a tool. These collision problems also arise in other applications, as path-planning for robot motions, a key problem in the current industry. To avoid this, we need to remove certain segments of the curve which start and end at self-intersections [29,70]. Special methods for the case of interior offsets (as used in milling holes or *pockets*) can be found in [29] and [57]. In the case of surfaces, the scenario is, by large, much more complicated: singularities at a point can arise when the distance d of the smallest value of the principal curvature is attained at the point. In addition, these singularities can be of many different types: cusps, sharp edges or self-intersections [21]. Finally, the set of rational curves and surfaces is not closed under offsetting [18]. Therefore, considerable attention has been paid to identify the curves and surfaces which admit rational offsets [22,59,64]. The case of polynomial and rational curves with rational offsets is analyzed in [48]. We also recommend [50] for a more recent overview of offset curves and surfaces.

Other recent developments are *geodesic offsets* [55] and *general offsets*, first introduced in [7] and extended in [65]. Both kinds of offsets exhibit applications in manufacture. For example, geodesic offset curves are used to generate tool paths on a part for zig-zag finishing using 3-axis machining (see Section 2.5) with ball-end cutter so that the scallop-height (the cusp height of the material removed by the cutter) will become constant. This leads to a significant reduction in size of the cutter location data and hence in the machining time. On the other hand, not only ball-end but also cylindrical and toroidal cutters are used in 3-axis NC machining. When the center of the ball-end cutter moves along the offset surface, the reference point on the cylindrical and toroidal cutters move along the general offset.

2.5 NC Milling

Numerical controlled (NC) milling technology is a process where a rotating cutter is sequentially moved along prescribed tool paths in order to manufacture a free-form surface from raw stock. NC milling is an essential tool for manufacturing free-form surfaces. For example, dies and injection molds for automobile parts are manufactured by using milling machines, which can be classified as a function of the number of axis in two (used to cut holes [29,57]), two-and-one-half, three, four and five axis (to mill free-form surfaces) (see [34], Chapter 16). These tasks have given rise to a number of different problems [44], such as those related to the determination of the milling coordinates and axis relative to the desired surface depending on the type of milling, transformation of control curves to machine coordinates, displacement of the tool along special surface curves or collision checking, etc. In general, these problems can be summarized as the determination of which parts of the surface are effected as the milling tool moves.

At first sight, two different approaches for the simulation of the process can be considered [25]: the exact, analytical approach [41,80] (which is computationally expensive) and the approximation approach. The cost of the simulation for the first approach (when using Constructive Solid Geometry) is reported to be $O(n^4)$ (n being the number of tool movements) by $O(n)$ for the approximation approach [38]. Since a complex NC program might consist of ten thousand movements, the first approach is computationally unapproachable and only approximate techniques are applied [32,36,37,38,72].

2.6 Implicitization

In the last years, implicit representations are being used more frequently in CAGD, allowing a better treatment of several problems. As one example, the point classification problem is easily solved with the implicit representation: it consists of a simple evaluation of the implicit functions. This is useful in many applications, as solid modeling for mechanical parts, for example, where points must be defined inside or outside the boundaries of an object, or for calculating intersections of free-form curves and surfaces (see Section 2.3). Through implicit representation, the problem is reduced to a trivial sign test. Other advantages are that the class of implicit surfaces is closed under such operations as offsetting, blending and bisecting. In other words, the offset (see Section 2.4) of an algebraic curve (surface) is again an algebraic curve (surface) and so on. In addition, the intersection (see Section 2.3) of two algebraic surfaces is an algebraic curve. Furthermore, the implicit representation offers surfaces of desired smoothness with the lowest possible degree. Finally, the implicit representation is more general than the rational parametric one [30]. All these advantages explain why the implicit equation of a geometric object is of importance in practical problems.

Implicitization is the process of determining the implicit equation of a parametrically defined curve or surface. One remarkable fact is that this parametric-implicit conversion is *always* possible [11,75]. Therefore, for any parametric curve or surface there exists an implicit polynomial equation defining exactly the same curve or surface. The corresponding algorithm for curves is given in [73] and [74]. In addition, a parametric curve of degree n has an implicit equation of also degree n. Further, the coefficients of this implicit equation are obtained from those of the parametric form by using only multiplication, addition and subtraction, so conversion can be performed through symbolic computation, with no numerical error introduced. Implicitization algorithms also exist for surfaces [51,73, 74]. However, a triangular parametric surface patch of degree n has an implicit equation of degree n^2. Similarly, a tensor product parametric patch of degree (m, n) has an implicit equation of degree $2mn$. For example, a bicubic patch has an implicit equation of degree 18 with 1330 terms!!!

In general, the implicitization algorithms are based on *resultants*, a classical technique [71], Gröbner bases techniques [8] and on the Wu-Ritt method [89]. Resultants provide a set of techniques [39] for eliminating variables from systems of nonlinear equations. However, the derived implicit equation may have extraneous factors: for example, surfaces can exhibit additional sheets. On the other hand, symbolic computation required to obtain the implicit expression exceeds the resources in space and time, although parallel computation might, at least partially, solve this problem. On the other hand, given an initial set of two or three polynomials defining the parametric curve or surface as a basis for an ideal [30], the Gröbner basis will be such that it contains the implicit form of the curve or surface. In the rational case, additional polynomials are needed to account for the possibility of base points [40]. Finally, the Wu-Ritt method consists of transforming the initial set into a triangular system of polynomials. This transformation involves rewriting the polynomials using pseudo-division and adding the remainders to the set. The reader is referred to [39] and [89] for more details. With respect to implementation, hybrid symbolic/numerical methods have been proposed in [52]. Also, in [31] atractive speed-ups for Gröbner based implicitization using numerical and algebraic techniques have been obtained.

Finally, we remark that implicitization can be seen as a particular case of conversion between different curve or surface forms (see, for example, [83,84]). See also [33] (and references therein) for a survey on approximate conversion between Bézier and B-spline surfaces, which are also applied to offsets.

Acknowledgements

The authors would like to acknowledge the CICYT of the Spanish Ministry of Education (project TAP98-0640) and the European Fund FEDER (Contract 1FD97-0409) for partial support of this work. They also thank the referees for their careful reading of the initial version of the manuscript and their helpful suggestions which allowed a substantial improvement of the paper.

References

1. K. Abdel-Malek and H.J. Yeh: On the determination of starting points for parametric surface intersections. CAD **29** (1997) 21-35
2. S. Aomura and T. Uehara: Self-intersection of an offset surface. CAD **22** (1990) 417-422
3. C. Asteasu: Intersection of arbitrary surfaces. CAD **20** (1988) 533-538
4. L. Bardis and N.M. Patrikalakis: Blending rational B-spline surfaces. Eurographics'89 (1989) 453-462
5. R.E. Barnhill: Geometry Processing for Design and Manufacturing, SIAM, Philadelphia, PA (1992)
6. R.E. Barnhill and S.N. Kersey: A marching method for parametric surface/surface intersection. CAGD **7** (1990) 257-280
7. E.L. Brechner: General tool offset curves and surfaces. In: R.E. Barnhill (ed.): Geometry Processing for Design and Manufacturing, SIAM (1992) 101-121
8. B. Buchberger: Gröbner bases: an algorithmic method in polynomial ideal theory. In: N.K. Rose (ed.): Multidimensional Systems theory, Reidel Publishing Co. (1985) 184-232
9. W.R. Carlson: An algorithm and data structure for 3D object synthesis using surface patch intersections. Computer Graphics **16** (1982) 255-263
10. E.W. Chionh and R.N. Goldman: Using multivariate resultants to find the implicit equation of a rational surface. The Visual Computer **8** (1992) 171-180
11. K.P. Cheng: Using plane vector fields to obtain all the intersection curves of two general surfaces. In: W. Strasser and H.P. Seidel (ed.): Theory and Practice in Geometric Modeling, Springer, New York (1989) 187-204
12. B.K. Choi and S.Y. Ju: Constant-radius blending in surface modeling. CAD **21** (1989) 213-220
13. E. Cohen, T. Lyche and R.F. Riesenfeld: Discrete B-splines and subdivision techniques in CAGD and computer graphics. Computer Graphics and Image Processing **14** (1980) 87-111
14. D.P. Dobkin, S.V.F. Levy, W.P. Thuston and A.R. Wilks: Contour tracking by piecewise linear approximations. ACM Trans. on Graph. **9** (1990) 389-423
15. G. Elber, I. Lee and M.S. Kim: Comparing offset curve approximation methods. IEEE Comp. Graph. and Appl. **17**(3) (1997) 62-71
16. G. Farin: Trends in curve and surface design. CAD **21**(5) (1989) 293-296
17. G. Farin: An ISS bibliography. In: R.E. Barnhill (ed.): Geometry Processing for Design and Manufacturing, SIAM (1992) 205-207
18. G. Farin: Curves and Surfaces for Computer Aided Geometric Design, Fourth Edition, Academic Press, San Diego (1996)
19. G. Farin: NURB Curves and Surfaces: from Projective Geometry to Practical Use, Second Edition, AK Peters, Wellesley, MA (1999)

20. G. Farin and D. Hansford: The Essentials of CAGD, AK Peters, Wellesley, MA (2000)
21. R.T. Farouki: The approximation of non-degenerate offset surfaces. CAGD **3** (1986) 15-43
22. R.T. Farouki: Pythegorean-hodograph curves in practical use. In: R.E. Barnhill (ed.): Geometry Processing for Design and Manufacturing, SIAM (1992) 3-33
23. I.D. Faux and M.J. Pratt: Computational Geometry for Design and Manufacture, Ellis Horwood, Chichester (1979)
24. D.J. Filip: Blending parametric surfaces. ACM Trans. on Graph. **8**(3) (1989) 164-173
25. G. Glaeser and E. Gröller: Efficient volume-generation during the simulation of NC-milling. In: H.C. Hege and K. Polthier (ed.): Mathematical Visualization. Algorithms, Applications and Numerics, Springer Verlag, Berlin (1998) 89-106
26. R.N. Goldman: Subdivision algorithms for Bézier triangles. CAD **15** (1983) 159-166
27. J.G. Griffiths: A data structure for the elimination of hidden surfaces by patch subdivision. CAD **7** (1975) 171-178
28. E. Hartmann: Blending of implicit surfaces with functional splines. CAD **22** (1990) 500-506
29. M. Held: On the computational geometry of pocket machining. Lectures Notes in Computer Science, 500, Springer Verlag, Berlin, New York (1991)
30. C.M. Hoffmann: Geometric and Solid Modeling, Morgan Kaufmann, San Mateo, CA (1989)
31. C.M. Hoffmann: Algebraic and numerical techniques for offsets and blends. In: S. Micchelli, M. Gasca and W. Dahmen (ed.): Computations of Curves and Surfaces, Kluwer Academic (1990) 499-528
32. T. van Hook: Real time shaded NC milling display. Computer Graphics **20**(4) (1986) 15-20 (Proc. SIGGRAPH'86)
33. J. Hoschek and F.J. Schneider: Approximate spline conversion for integral and rational Bézier and B-spline surfaces. In: R.E. Barnhill (ed.): Geometry Processing for Design and Manufaturing, SIAM (1992) 45-86
34. J. Hoschek and D. Lasser: Fundamentals of Computer Aided Geometric Design, A.K. Peters, Wellesley, MA (1993)
35. E.G. Houghton, R.F. Emnett, J.D. Factor and C.L. Sabharwal: Implementation of a divide-and-conquer method for intersection of parametric surfaces. CAGD **2** (1985) 173-183
36. Y. Huang and J.H. Oliver: NC milling error assessment and tool path correction. Computer Graphics Proceedings (1994) 287-294 (Proc. SIGGRAPH'94)
37. K.C. Hui: Solid sweeping in image space-application in NC simulation. The Visual Computer **10** (1994) 306-316
38. R.B. Jerard, S.Z. Hussaini, R.L. Drysdale and B. Schaudt: Approximate methods for simulation and verification on NC machining programs. The Visual Computer **5** (1989) 329-348
39. D. Kapur and Y.N. Lakshman: Elimination methods. In: B. Donald, D. Kapur and J. Mundy (ed.): Symbolic and Numerical Computing for Artificial Intelligence, Academic Press (1992)
40. M. Kalkbrener: Implicitization of rational parametric curves and surfaces. Technical Report, Kepler Universität, Linz, Austria, RISC, Linz (1990)
41. Y. Kawashima, K. Itoh, T. Ishida, S. Nonaka and K. Ejiri: A flexible quantitative method for NC machining verification using a space-division based solid model. The Visual Computer **7** (1991) 149-157

42. T.L. Kay and J.T. Kajiya: Ray tracing complex scenes. Computer Graphics **20** (1986) 269-278
43. K.J. Kim and M.S. Kim: Torus/sphere intersection based on configuration space approach. Graphical Models and Image Processing **60**(1) (1998) 77-92
44. R. Klass and P. Schramm: NC milling of CAD surface data. In: H. Hagen and D. Roller (ed.): Geometric Modeling. Methods and Applications, Springer Verlag, Berlin Heidelberg (1991) 213-226
45. R. Klass and B. Kuhn: Fillet and surface intersections defined by rolling balls. CAGD **9** (1992) 185-193
46. P.A. Koparkar: Designing parametric blends: surface model and geometric correspondence. The Visual Computer **7** (1991) 39-58
47. D. Lasser: Intersection of parametric surfaces in the Bernstein-Bézier representation. CAGD **3** (1986) 186-192
48. W. Lü: Offset-rational parametric plane curves. CAGD **12** (1995) 601-616
49. W. Ma and Y.S. Lee: Detection of loops and singularities of surface intersections. CAD **30** (1998) 1059-1067
50. T. Maekawa: An overview of offset curves and surfaces. CAD **31** (1999) 165-173
51. D. Manocha and J. F. Canny: Algorithm for implicitizing rational parametric surfaces. CAGD **9** (1992) 25-50
52. D. Manocha and J. F. Canny: Implicit representations of rational parametric surfaces. J. of Symbolic Computation **13** (1992) 485-510
53. J. Miller and R.N. Goldman: Geometric algorithms for detecting and calculating all conic sections in the intersection of any two natural quadric surfaces. Graphical Models and Image Processing **57**(1) (1995) 55-66
54. J.C. Owen and A.P. Rockwood: Intersection of general implicit surfaces. In: G.E. Farin (ed.): Geometric Modeling: Algorithms and New Trends, SIAM (1987) 335-345
55. N.M. Patrikalakis and L. Bardis: Offsets of curves on rational B-spline surfaces. Engineering with Computers **5** (1989) 39-46
56. J. Pegna and D.J. Wilde: Spherical and circular blending of functional surfaces. Trans. of ASME, Journal of Offshore Mechanics and Artic Engineering **112** (1990) 134-142
57. H. Persson: NC machining of arbitrarily shaped pockets. CAD **10** (1978) 169-174
58. G. Petrie and T.K.M. Kennie: Terrain modeling in surveying and civil engineering. CAD **19** (1987) 171-187
59. B. Pham: Offset curves and surfaces: a brief survey. CAD **24** (1992) 223-229
60. L. Piegl: Key developments in Computer-Aided Geometric Design, CAD **21**(5) (1989) 262-273
61. L. Piegl and W. Tiller: The NURBS Book, Second Edition, Springer Verlag, Berlin Heidelberg (1997)
62. L. Piegl and W. Tiller: Computing offsets of NURBS curves and surfaces. CAD **31** (1999) 147-156
63. L. Piegl and W. Tiller: Geometry-based triangulation of trimmed NURBS surfaces. CAD **30** (1998) 11-18
64. H. Pottmann: Rational curves and surfaces with rational offsets. CAGD **12** (1995) 175-192
65. H. Pottmann: General offset surfaces. Neural, Parallel and Scientific Computations **5** (1997) 55-80
66. A. Rockwood: The displacement method for implicit blending of surfaces in solid modeling. ACM Trans. on Graph. **8**(4) (1989) 279-297
67. A. Rockwood, K. Heaton and T. Davis: Real-time rendering of trimmed surfaces. Computer Graphics **23** (1989) 107-116 (Proc. SIGGRAPH'89)

68. D.F. Rogers: An Introduction to NURBS: with Historical Perspective, Morgan Kaufmann, San Mateo, CA (2000)
69. J.R. Rossignac and A.A.G. Requicha: Offsetting operations in solid modeling. CAGD **3** (1986) 129-148
70. S.E.O. Saeed, A. de Pennington and J.R. Dodsworth: Offsetting in geometric modeling. CAD **20** (1988) 67-74
71. G. Salmon: Lessons Introductory to the Modern Higher Algebra, G.E. Stechert & Co., New York (1885)
72. T. Saito and T. Takahashi: NC machining with G-buffer method. Computer Graphics **25**(4) (1991) 207-216 (Proc. SIGGRAPH'91)
73. T.W. Sederberg: Implicit and parametric curves and surfaces for computer aided geometric design. Ph.D. thesis, Purdue Univ., West Lafayette, IN (1983) 29-42
74. T.W. Sederberg, D.C. Anderson and R.N. Goldman: Implicit representation of parametric curves and surfaces. Computer Vision, Graphics and Image Processing **28** (1984) 72-74
75. T.W. Sederberg: Algebraic geometry for surface and solid modeling. In: G.E. Farin (ed.): Geometric Modeling: Algorithms and New Trends, SIAM (1987) 29-42
76. T.W. Sederberg and R.J. Meyers: Loop detection in surface patch intersections. CAGD **5** (1988) 161-171
77. M. Shantz and S.L. Chang: Rendering trimmed NURBS with adaptive forward differences. Computer Graphics **22** (1988) 189-198 (Proc. SIGGRAPH'88)
78. C.K. Shene and J. Johnstone: On the lower degree intersections of two natural quadrics. ACM Trans. on Graphics **13**(4) (1994) 400-424
79. X. Sheng and B.E. Hirsch: Triangulation of trimmed surfaces in parametric space. CAD **24**(8) (1992) 437-444
80. A.I. Sourin and A.A. Pasko: Function representation for sweeping by a moving solid. IEEE Trans. on Visualization and Computer Graphics **2**(2) (1996) 11-18
81. D.C. Sutcliffe: Contouring over rectangular and skewed rectangular grids. In: K. Brodlie (ed.): Mathematical Methods in Computer Graphics and Design, Academic Press (1980) 39-62
82. M. Sweeney and R. Bartels: Ray tracing free-form B-spline surfaces. IEEE Comp. Graph. and Appl. **6** (1986) 41-49
83. A.E. Vries-Baayens: Conversion of a Composite Trimmed Bézier Surface into Composite Bézier Surfaces. In: P.J. Laurent, Le Mehaute and L.L.Schumaker (ed.): Curves and Surfaces in Geometric Design, Academic Press, Boston, USA (1991) 485-489
84. A.E. Vries-Baayens and C.H. Seebregts: Exact Conversion of a Composite Trimmed Nonrational Bézier Surface into Composite or Basic Nonrational Bézier Surfaces. In: H. Hagen (ed.): Topics in Surface Modeling, SIAM, Philadelphia, USA (1992) 115-143
85. Y. Wang: Intersections of offsets of parametric surfaces. CAGD **13** (1996) 453-465
86. J. Warren: Blending algebraic surfaces. ACM Trans. on Graph. **8**(4) (1989) 263-278
87. D.B. Welborun: Full three-dimensional CAD/CAM. CAE Journal **13** (1996) 54-60, 189-192
88. J.R. Woodwark: Blends in geometric modeling. In: R.R. Martin (ed.): The Mathematics of Surfaces II, Oxford Univ. Press (1987) 255-297
89. W.T. Wu: Basic principles of mechanical theorem proving in geometries. J. of Systems Sciences and Mathematical Sciences **4** (1986) 207-235
90. C.G. Yan: On speeding up ray tracing of B-spline surfaces. CAD **19** (1987) 122-130
91. J. Yen, S. Spach, M Smith and R. Pulleyblank: Parallel boxing in B-spline intersection. IEEE Comp. Graph. and Appl. **11** (1991) 72-79