Modeling with Subdivision Surfaces

T. Ullrich, A. Schiefer, D. W. Fellner
Institut für ComputerGraphik und WissensVisualisierung, TU Graz, Austria
Fraunhofer Austria Research, Visual Computing Devision, Graz, Austria
Fraunhofer Institute for Computer Research and Technical University of Darmstadt, Germany

ABSTRACT

Subdivision surfaces are an established modeling tool in computer graphics and computer-aided design. While the theoretical foundations of subdivision surfaces are well studied, the correlation between a control mesh and its subdivided limit surface still has some open-ended questions: Which topology should a control mesh have? Where should control vertices be placed? A modeler – human or software – is confronted with these questions and has to answer them.

In this paper we analyze four characteristic situations. Each one consists of an analytical reference surface S and several variants of control meshes C_i. In order to concentrate on the topology of the control meshes, the geometrical positions of their control vertices have been determined and optimized automatically. As a result we identified the best topology of all C_i to represent the given surface S. Based on these results we derived heuristics to model with subdivision surfaces. These heuristics are beneficial for all modelers.

Keywords: Computer Graphics, Computer-Aided Design, Subdivision Surfaces, Modeling Techniques

1 INTRODUCTION

Subdivision surfaces have gained a lot of attention within the last decades. Especially artists and designers prefer subdivision surfaces to other free-form surface representations. The way how to model with subdivision surfaces is described in various tutorials and courses [4]. These tutorials pursue a plan which can be summarized by the sequence

idea → subdivision surface model → real 3d object,

whereas the last production step is omitted, if only the virtual object is of interest. Within the last few years subdivision surfaces are also used in reverse engineering at a progressive rate. Reverse engineering is the inverse situation of the sequence above. A real 3d object is the starting point and its corresponding subdivision surface model is the desired result. During the re-modeling phase the modeler – either human or algorithm – has to tackle two problems: geometry and topology. These are the two ingredients of a subdivision surface. While many articles are about geometric optimization, we concentrate on topological questions. Choosing the right topology to represent a given object by a subdivision surface reduces the number of needed control vertices significantly. The choice of a good topology is a distinction between an experienced, skilled modeler and a beginner. In this article we analyze some characteristic objects and test different, topological variants of control meshes. A geometrical optimization allows us to determine the best vertex positions automatically and to concentrate on topological aspects. Based on these optimizations we determine the appropriateness of each topological variant to represent a certain object. Consequently, we derive heuristics to model with subdivision surfaces.

WSCG'2010, February 1 – February 4, 2010
Plzen, Czech Republic.

2 RELATED WORK

Modeling with subdivision surfaces is the main topic of many tutorials and courses [16], [15]. A good overview on subdivision surfaces has been presented by WEIYIN MA [7] whereas details on the mathematical background can be found in the book *Subdivision Surfaces* by JÖRG PETERS and ULRICH REIF [11]. Furthermore a variety of articles is about topological robustness [1] or handling "special" situations such as filling n-sided regions [10].

In this article we concentrate on the more general situation from the reverse engineering point of view. From this perspective a subdivision surface reconstruction method has to tackle two problems, a topological problem (the layout of the control mesh) and a geometrical problem (the position of the control vertices in 3D). Unfortunately, in most articles about subdivision surface fitting [3], [2] the topological problem plays a minor role [8]. In this article we address this topological problem; i.e. we start with a topological setting; then we optimize the geometry of the control mesh by calculating the best vertex positions to approximate a given surface. The optimization results establish a relationship between topological layout and approximation quality, which allows to identify the best topology. In order to concentrate on the topological aspects the geometry is optimized automatically. The optimization uses a distance based error function [14], [12] which is minimized by a standard approach of numerical minimization [9], [5]. The minimization process is straight forward and operates on the vector of all vertex coordinates of all vertices to position. As the geometrical optimization is not within the scope of this article we

refer to RAINER STORN and KENNETH PRICE [13] for the algorithmic details of the minimization routine (see http://www.icsi.berkeley.edu/~storn/code.html for implementations).

3 MODELING

The case studies in the following section cover the four most important situations when modeling with Catmull-Clark subdivision surfaces:

Modeling Edges The first case examines smooth edges. While sharp edges can be modeled easily with special feature rules for subdivision surfaces, smooth edges offer at least two possibilities to be designed: with a row of control vertices along the edge or with two rows of control vertices alongside the edge.

Modeling Non-Quadrilateral Configurations Almost each type of subdivision surface has a topology for which it suits best. Loop subdivision prefers triangles, Catmull-Clark subdivision generates quads, etc. Unfortunately, not every object has a favorable geometric primitive to be modeled with. Several strategies for such a configuration are possible.

Modeling Curvature The third case inspects a surface with different curvatures: hyperbolic, parabolic and elliptic. It addresses the issue of quad orientation with respect to the surface's principle curvature directions.

Modeling Inflection Points The last case analyzes a surface which is defined by a cut curve. This is a standard situation in CAD modeling.

Each case examines different variants of control meshes that are adjusted by a number of parameters. These parameters are set by a numerical optimization routine based on *differential evolution* [13]. It minimizes the distance [14] between the nominal surface and the actual subdivision surface.

3.1 Smooth Edges

The first nominal surface is defined by the implicit equation

$$S_1 : \; x^6 + y^6 + z^6 = 1. \qquad (1)$$

The resulting surface looks like a cube with round edges and corners (see Figure 1). In order to analyze how to model beveled edges two variants are inspected.

Variant 1 The first variant of possible control meshes has a cube-like topological layout. It consists of 3×3 quads on each side. Due to symmetries the geometry is defined by only six parameters of three initial control points. Three parameters are used for the three coordinates (x_1, y_1, z_1) of the first control point. The fourth and the fifth parameter define the second control point.

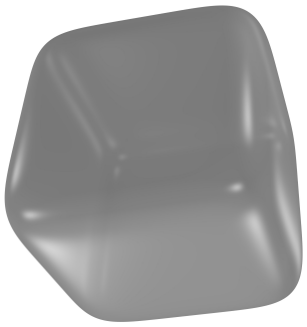

Figure 1: The implicit surface $x^6 + y^6 + z^6 = 1$ has a degree which cannot be reached by a cubic patch. Therefore, Catmull-Clark subdivision can only approximate it. The best way to approximate it is analyzed by testing different topologies.

In order to avoid self-intersections as well as optimizations with constraints, these parameters are defined as positive offsets to the first control point. Consequently, they have a fixed domain and the second control point is calculated via $(x_1 + x_2, y_2, x_1 + x_2)$. As the second control point defines an edge of the control mesh cube, and as the faces of the cube are connected at/over this edge, the x and z components have to be equal. Finally the last parameter x_3 again defines an offset from x_1. Due to symmetries this is the only value needed for the third control point: $(x_1 + x_3, x_1 + x_3, x_1 + x_3)$. As the third control point is at the corner of the control mesh cube, all coordinate components must be equal. Figure 2 shows the initial three control points that are generated by the six parameters.

The optimization routine minimizes the distance between the generated surface and the reference surface (Equation 1) based on uniformly-distributed samplings. The error function $f_{1,1}$ is a sum of point-to-subdivision surface distances. The optimum is

$$f_{1,1}(\; 0.33633, \; 0.33526, \; 0.99830, \; 0.66616, \; 0.32615, \; 0.66319 \;) = 7.45141.$$

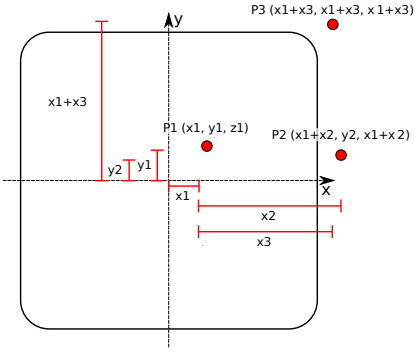

Figure 2: Due to symmetries six parameters are enough to define three different control points. The complete subdivision control mesh is composed of rotations and mirrorings of these points. Each of the six cube sides consists of 3×3 faces.

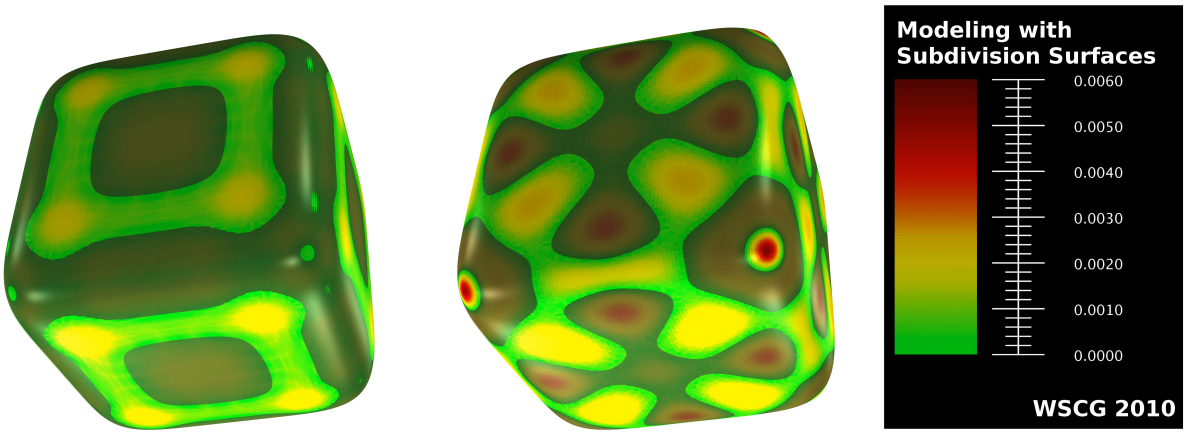

Figure 3: In comparison to each other the first variant performs better than the second one. Explicit modeling of beveled edges increases the risk of "over-modeling" – a high frequency fluctuation which is spread out from the beveled edge and which disturbs the low frequency parts of the surface.

The three control points, which generate the complete subdivision control mesh by rotations and mirrorings, have the coordinates

$$(0.33633, 0.33526, 0.99830),$$
$$(1.00249, 0.32615, 1.00249),$$
$$(0.99952, 0.99952, 0.99952).$$

Variant 2 Also the second variant is a cube-like control mesh derived from three initial control points but with five parameters. This time every side of the cube consists of four quads, the six sides of the cube are connected through beveled faces at the edges and with triangles at the corners. Figure 4 illustrates this control mesh.

The first parameter, z_1, is the z component of the first control point, which is located at $(0,0,z_1)$. Due to symmetries the first control point is centered on each side of the control mesh cube. The second and third parameters, x_2 and z_2, define the second control point at $(x_2,0,x_2+z_2)$. This time the sides of the cube are connected via beveled faces, therefore the offset z_2 is added to the z component of the second control point. Using the last two parameters, x_3 and z_3, the third control point is (x_3,x_3,x_3+z_3). The error of the best control mesh of this variant – according to the optimization routine – is 7.79202:

$$f_{1,2}(\ 0.99686,\ 0.83273,\ 0.14743,\ 0.74912,\ 0.33157) = 7.79202.$$

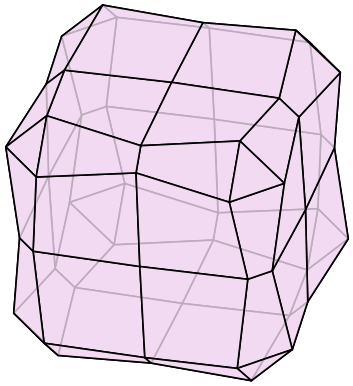

Figure 4: This variant uses a control mesh with explicitly beveled edges to approximate an implicit surface of degree 6.

Comparison Figure 3 illustrates both variants. Each variant is rendered with a color scheme indicating its distance to the nominal surface which is included in each rendering. It is visualized with a transparent, grayish style. The illustration shows that the first variant performs better than the second one. The second variant uses beveled edges to model the reference surface. This topology is not suitable to approximate the given surface. Even its geometrically optimized version shows "over-modeling" effects – a high frequency fluctuation which is spread out from the over-modeled parts. In this case the beveled edges disturb the low frequency parts of the surface. An in-depth analysis of the distances confirms the visualization. The average and the maximum of all distances between the nominal surface and points on the actual surface are:

	avg. distance	max. distance
Variant #1	0.00079	0.00204
Variant #2	0.00130	0.00519

3.2 Modeling Non-Quadrilateral Configurations

The second surface analysis investigates non-quadrilateral configurations. The reference surface is a so-called *monkey saddle*. It is a height field defined by

$$S_2(x,y) = \frac{x^3 - 3xy^2}{2}. \qquad (2)$$

This surface is axially symmetric at a degree of 120°; i.e. after a $\frac{2}{3}\pi$ rotation the surface is congruent to itself. The point at the origin is a parabolic, umbilic point. Within this analysis the parameters x and y may range from -1.0 to 1.0.

As the following variants have different numbers of control vertices, the area of every control mesh is chosen to be proportional to the number of its vertices. So control meshes with more vertices have to approximate a larger area of the reference surface.

Variant 1 The first variant has 19 vertices and seven parameters, which define their heights (z component). The symmetric configuration is illustrated in Figure 5 which shows the topology of the control mesh and the parameters (height) of each vertex. All vertices with $x = 0$ have a fixed height of 0.0. The best optimized

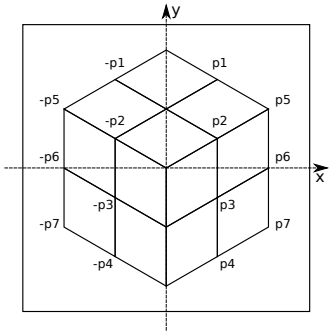

Figure 5: The topology of this control mesh is arranged in a 120° symmetric layout to adapt the subdivision surface to the reference.

version of this variant has an error of 0.44914. Its parameters are

$$f_{2,1}(\ \begin{matrix} -0.27195, & -0.00007, & 0.00001, \\ -0.27193, & -0.00007, & 0.27181, \\ -0.0001) & = 0.44914. & \end{matrix}$$

Variant 2 For the second variant a predefined mesh with 27 vertices is used as control mesh. The heights of its vertices are controlled by twelve parameters. Figure 6 shows the parameters and the topology which is (in the inner part) dual to the first variant. Again, all vertices with $x = 0$ have a fixed height of 0.0.

The smallest possible error of this topological configuration is

$$f_{2,2}(\ \begin{matrix} 0.00148, & -0.07632, & -0.25658 \\ -0.37829, & -0.07681, & 0.07507, \\ 0.00052, & -0.37787, & -0.25613, \\ 0.25710, & 0.37745, & -0,00072) \\ = 0.69362. & & \end{matrix}$$

Variant 3 The third variant for this surface does not adapt to the reference's axial symmetry of $\frac{2}{3}\pi$ in any way. It uses a regular quadratic grid centered on the reference surface. There are five vertices along each side of the grid, so 25 vertices in total. Fixing the vertices at $x = 0$ six parameters are enough to describe all vertices. Figure 7 illustrates this topology. The optimization returns the optimum of this topology with an error of $f_{2,3} = 0.80376$.

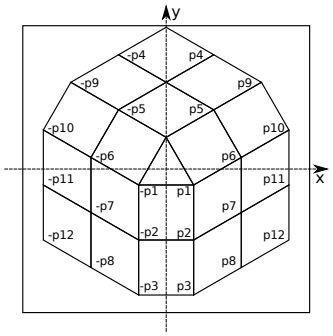

Figure 6: The topology of the second variant has a configuration which is dual to the first one (except for the border).

Comparison Taking the number of control vertices into account, each surface has been normalized; i.e. the area of every control mesh is proportional to the number of its vertices. Therefore, all subsequent error values are normalized and comparable to each other. In this situation the case study does not reveal a best topology but a worst one. The regular gird does not approximate the given surface well. The two variants whose topologies reflect the nominal surface's symmetry perform much better (see Figure 8):

	avg. distance	max. distance
Variant 1	0.00241	0.00624
Variant 2	0.00153	0.01037
Variant 3	0.00330	0.01555

3.3 Modeling Curvature

The third modeling study inspects a torus due to its characteristic curvatures: hyperbolic, parabolic, and elliptic [6]. The reference is described by the formula:

$$S_3(u,v) = \begin{pmatrix} (a + b\cos(2\pi v)) \cdot \cos(2\pi u) \\ (a + b\cos(2\pi v)) \cdot \sin(2\pi u) \\ b\sin(2\pi v) \end{pmatrix} \quad (3)$$

with major radius $a = 5.0$ and minor radius $b = 3.0$. The parameters u and v are within the range 0.0 to 1.0.

Variant 1 For the first variant the parameter domain is sampled at a fixed, regular grid to get the control point positions. Two parameters, that can be set by

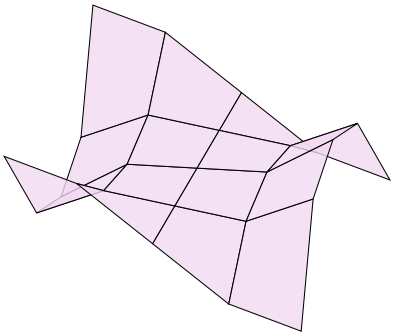

Figure 7: A rectangular topology leads to a very high error. Topologies that reflect the nominal surface's symmetry perform much better.

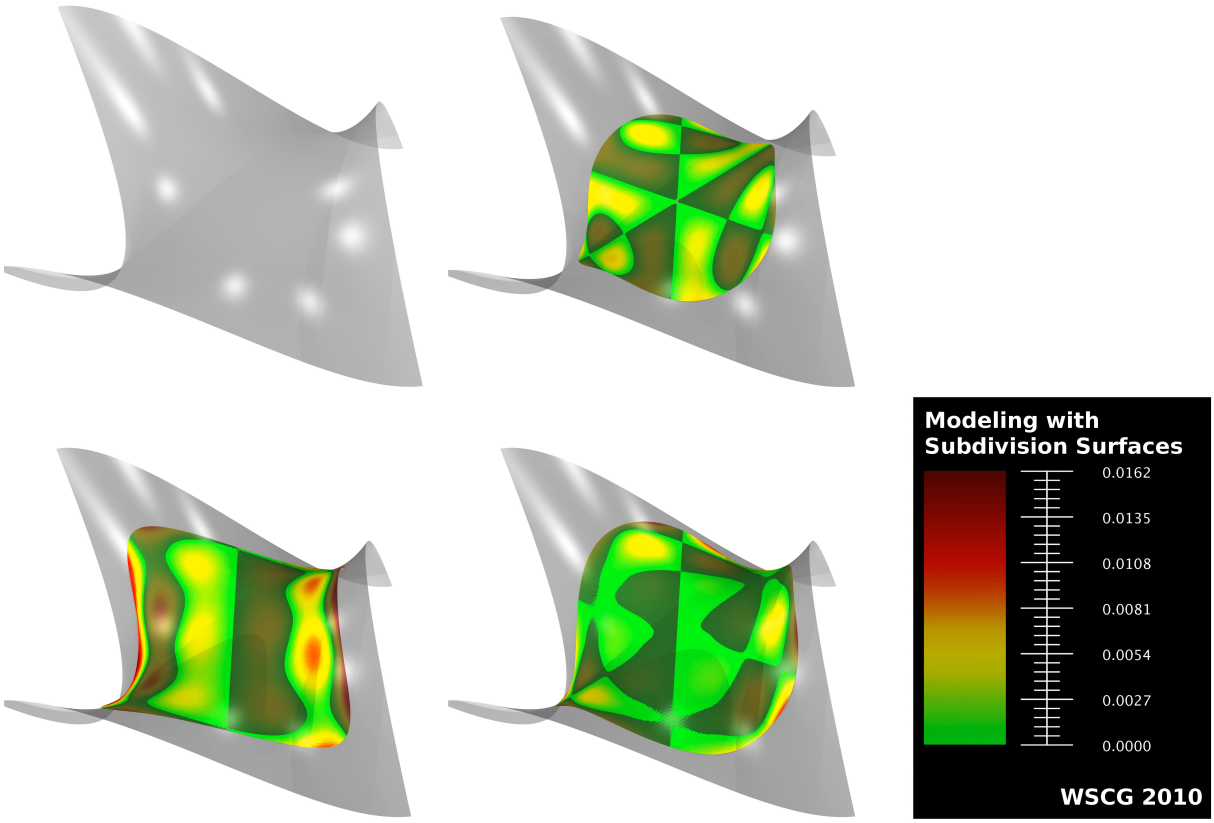

Figure 8: The nominal surface (upper left) is invariant to 120° rotations. The topology of variant #1 (upper right) and #2 (lower right) reflect this property whereas variant #3 (lower left) uses a simple rectangular grid. In this case topologies, which reflect the main symmetries of the reference surface, perform better.

the optimization, define the major and the minor radius of the control mesh torus. This is probably the most commonly used parameterization of a torus in modeling software.

The error of this variant after the optimization is 824.82012. The two parameters defining the major and minor radius have their optimum at 5.42375 respectively 3.38278. This variant is compared to slanted torus control mesh.

Variant 2 A slanted torus uses a slightly different parameterization than the first one. Again, the values u and v are a regular grid in the parameter domain, but this time an offset depending on u is added to v each time. For this variant, the offset is defined as $\frac{2}{3}u$. This introduces a slant in the control mesh.

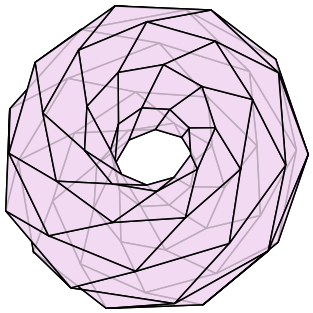

Figure 9: A slanted torus control mesh is a "non-standard" way to model a torus. This variant is compared to the most commonly used parameterization with a rectangular grid layout of the parameter domain.

The optimization routine calculates a minimum error of 1474.60738 for this variant. The parameters defining the resulting control mesh are $a = 5.62499$ and $b = 3.43444$. Figure 9 shows the slanted control mesh created with these parameters.

Variant 3 This variant of the torus control mesh is basically the same as variant 2, except that the offset defining the slant is defined as $\frac{4}{3}u$. Even the optimized geometry ($a = 6.24999, b = 3.56744$) leads to a large error $f_{3,3} = 4290.40374$.

Variant 4 As it is difficult to determine appropriate slant offsets, the fourth variant also optimizes this parameter; i.e. in addition to the two parameters defining the radii of the torus control mesh, it has a third parameter defining the slant offset. The optimization process returns

$$f_{3,4}(5.42399, 3.38277, 0.00000) = 826.65427.$$

Comparison The easiest way to model a torus seems to be the best way. All variations in topology increase the approximation error (see Figure 10). The in-depth analysis of distances confirms this heuristic:

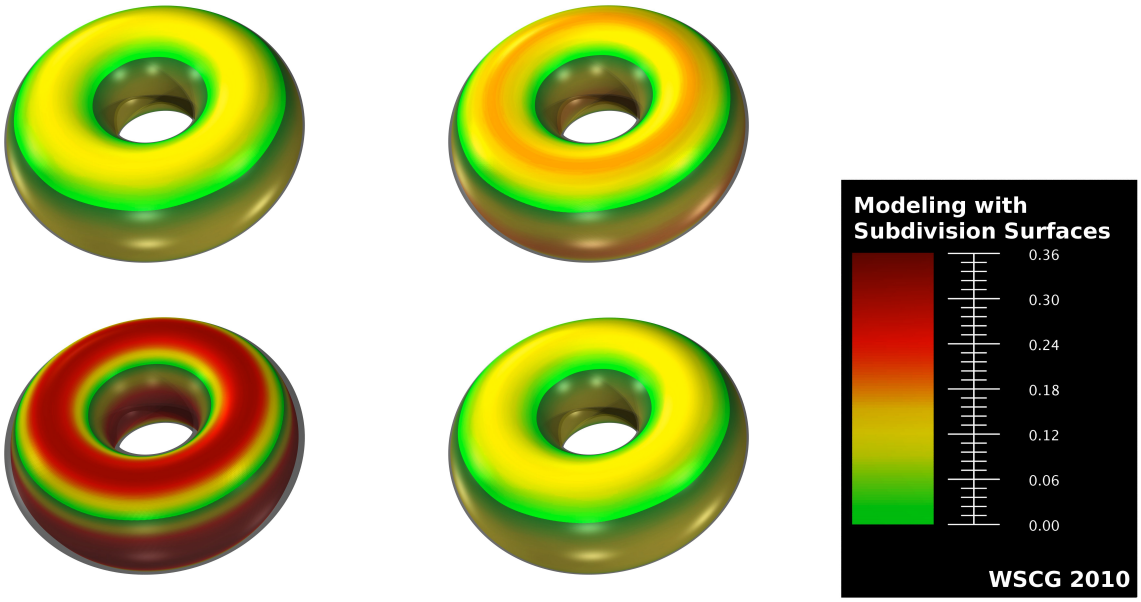

Figure 10: A torus can be modeled with a slant offset and without one (upper left). With increasing slant offset (upper right, lower left) the approximation error increases. If the optimization routine is allowed to set the slant offset by itself, it is set to zero (lower right).

	avg. distance	max. distance
Variant 1	0.07806	0.12925
Variant 2	0.11145	0.19450
Variant 3	0.19887	0.35382
Variant 4	0.07806	0.12904

3.4 Modeling Inflection Points

The last case analyzes a surface which is defined by a cut curve. Blended with a straight line the result has the formula:

$$S_4(x,y) = \frac{y}{10} \cdot \sin\left(\frac{2\pi}{3}\arctan x\right) \qquad (4)$$

The parameter x describes the cut curve in the range from -5.0 to 5.0 whereas the parameter y blends between the straight line and the curve from 0.0 to 10.0. The surface is plotted in Figure 11.

Variant 1 The control mesh of the first variant gets 20 parameters – 10 pairs of x position and height value z. The range of the x values (-5.0 to 5.0) has been partitioned into 10 equally-sized intervals. In each interval one and only one parameter pair is allowed. For each pair $p_i = (x_i, z_i)$ two control points are generated: one at $(x_i, 10, z_i)$ to approximate the cut curve and one at $(x_1, 0, 0)$ on the straight line. The error of the optimized variant is 6.73931:

$$\begin{aligned} f_{4,1}(\quad & -4.35546, \quad -0.30532, \\ & -3.05017, \quad -0.48656, \\ & -2.93309, \quad -0.50592, \\ & -1.58539, \quad -0.82026, \\ & -0.37453, \quad -1.29349, \\ & 0.38362, \quad 1.32794, \\ & 1.71154, \quad 0.78575, \\ & 2.30383, \quad 0.65748, \\ & 3.00657, \quad 0.48136, \\ & 4.02755, \quad 0.35095 \quad) = 6.73931. \end{aligned}$$

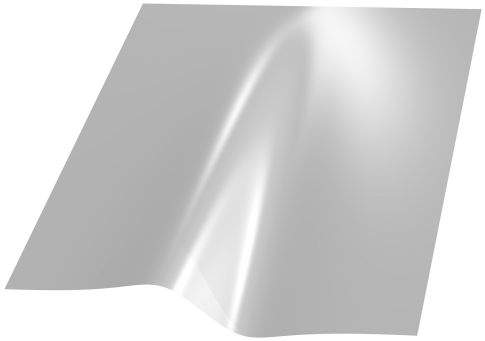

Figure 11: This surface is defined by a cut curve which is blended with a straight line. This is a standard situation in CAD modeling.

Variant 2 This variant uses a regular 10×3 grid covering the whole reference surface. All heights are specified as a parameter and the (x,y)-positions are fixed. Consequently, this variant operates on 30 parameters. The optimization calculates an error of 14.70749 for the best geometry. The best parameters for the control mesh are:

$$\begin{array}{lll} -0.00257, & -0.13120, & -0.27108, \\ 0.00131, & -0.17258, & -0.33910, \\ -0,00099, & -0.28033, & -0.55752, \\ 0,00048, & -0.36468, & -0.74336, \\ -0,00057, & -0.72072, & -1.40922, \\ -0,00022, & 0.72074, & 1.41115, \\ 0.00191, & 0.36363, & 0.74169, \\ -0.00244, & 0.28386, & 0.55652 \\ 0.00267, & 0.16798, & 0.34212, \\ -0.00105, & 0.13555, & 0.26656. \end{array}$$

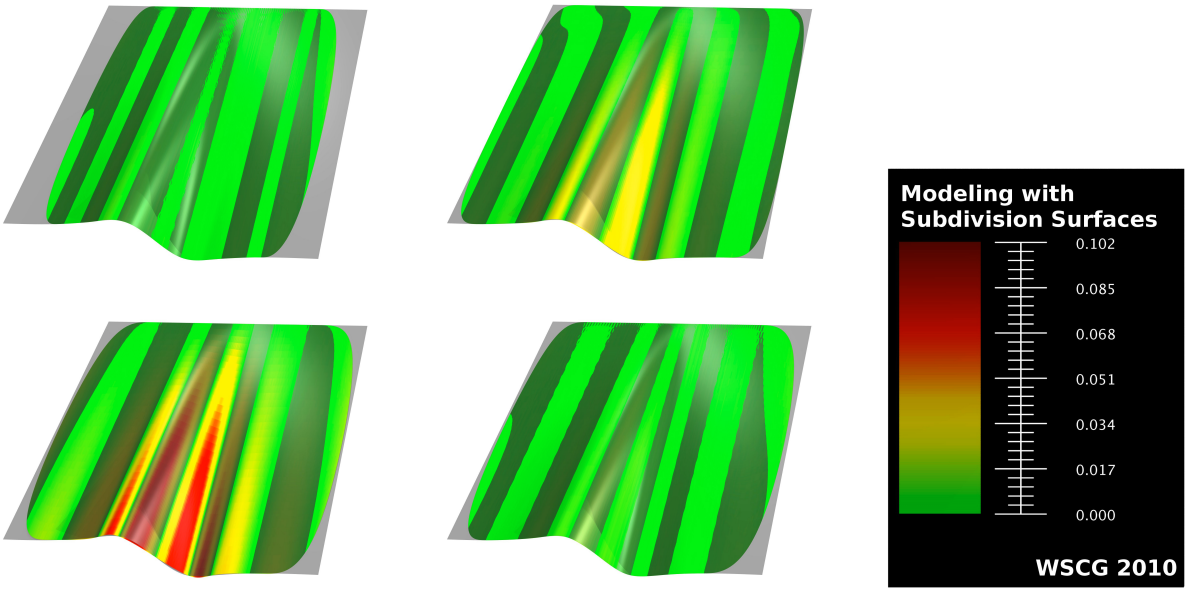

Figure 12: In this comparison approximations with fixed x positions are more erroneous than approximations with x positions as free parameters. The interval spacing (upper left) of the first variant and the prominent-values-of-the-cut-curve variant with additional offsets (lower right) are the best results, whereas the regular grid (upper right) and the fixed prominent-values-of-the-cut-curve (lower left) are the worst results.

Variant 3 The third variant has fixed x positions at prominent values of the cut curve: -5.00000 (end of range), -3.31764, -1.63528 (inflection point), -0.93159 (minimum), 0.00000 (inflection point, root), 0.93159 (maximum), 1.63528 (inflection point), 3.31764, 5.00000 (end of range). For each x position there is again one control point at $y = 0$ with height 0 and another one at $y = 10$ with the height value which has to be optimized. So this variant has nine height parameters, one for each x position. The result is

$$\begin{aligned} f_{4,3}(\quad & -0.22288, \quad -0.47240, \quad -0.67730, \\ & -1.37479, \quad -0.00353, \quad 1.37625, \\ & 0.67573, \quad 0.47305, \quad 0.22281) \\ & = 11.91346. \end{aligned}$$

Variant 4 The fourth variant is very similar to the third variant. It uses the same configuration but in contrast to fixed x positions, the optimization routine is allowed to modify these positions within an offset of $\pm\frac{1}{3}$. Therefore, this variant takes 18 parameters, nine for the height values at each x position and nine offsets for the initial x positions.

For this variant, the minimum error after the optimization is 6.17020. The parameters for this control mesh are

$$\begin{aligned} f_{4,4}(\quad & -0.25058, \quad -0.46790, \quad -0.74541, \\ & -1.18889, \quad -0.38835, \quad 1.29557, \\ & 0.76445, \quad 0.36971, \quad 0.26533, \\ \\ & 0.04882, \quad 0.33053, \quad -0.25836, \\ & 0.28352, \quad -0.17465, \quad -0.33332, \\ & -0.01570, \quad 0.32223, \quad -0.03218) \\ & = 6.17020. \end{aligned}$$

Comparison The in-depth distance analysis reveals the following values:

	avg. distance	max. distance
Variant 1	0.00264	0.01044
Variant 2	0.00718	0.04152
Variant 3	0.01826	0.09219
Variant 4	0.00323	0.01460

If the area of the approximated surface is taken into account, the fourth variant is even slightly better then the first one. The approximation with the highest error level is the variant which has fixed x positions at prominent values of the cut curve (see Figure 12 (lower left)). Consequently, the best result and the worst result can be created with the same topology and minor differences in geometry. The additional offsets which distinguish a good from a bad result are plotted in Figure 13. The figure shows some very important properties when modeling with subdivision surfaces.

- At the extreme values (minimum and maximum at ± 0.93159) the control points have been moved by the optimization routine towards the direction of higher absolute gradients.
- Having moved the control points, the control polygon has fewer intersections with the nominal curve than the fixed-x-positions version.
- All versions with a small error (also the first variant with interval spacing) intersect the reference curve very close to (at $x = -1.63528$ and at $x = 0.0$) or nearby (at $x = 1.63528$) inflection points.

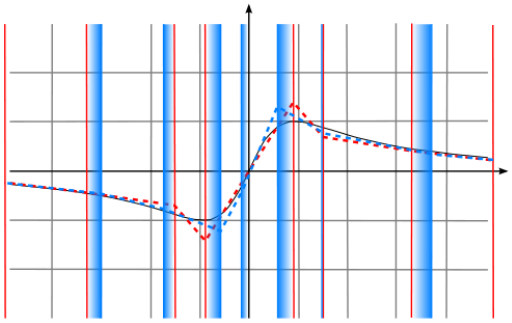

Figure 13: The cut curve which defines the fourth nominal surface is plotted in this diagram. Furthermore it shows the positions of the control vertices of the worst approximation (red) and the best approximation (blue). Their differences are visualized by gradients.

4 CONCLUSION

Based on the four case studies we arrive at several conclusions. The first case examines smooth edges and demonstrates the "over-modeling" effect. Modeling a local surface feature always takes the risk to disturb large parts of a model by spreading high-frequency fluctuations from the "over-modeled" parts. This effect can be avoided by a better topology, which does not model rounded edges explicitly, or by barrier lines – two or three consecutive lines of control vertices of low frequency, which suppress fluctuations due to the locally limited influence of a vertex to a subdivision surface.

While simple geometric properties – such as beveled edges – should not be considered in the topological layout of a control mesh, high-level geometric aspects play an important role. The second case study shows that global symmetries should be reflected in the control mesh. All control meshes, which did not reflect the global symmetry of the surface to approximate, caused higher errors than those with symmetrical layout. The third study approves this fact.

The last case analyzes a surface on the geometrical – not topological – level to explore the best positions for control vertices. Besides the conclusions already presented in the previous section the last study demonstrates the difficulties in predicting a good subdivision approximation without iterative optimization. All solutions with partly-fixed control vertices have a high error value. All heuristics derived from the last case reduce the error but do not reach the quality of the automatic optimization. As the optimization is time consuming (up to several hours for complex configurations) this is a problem for interactive modeling tools.

5 FUTURE WORK

Any surface can be approximated with a sufficient number of control points. Reducing this number keeps a subdivision surface model manageable. In the future we will concentrate on topological improvements and address the question whether it is possible to get accurate conclusions based on these case studies.

ACKNOWLEDGEMENTS

We gratefully acknowledge the generous support from the European Commission for the research project 3DCOFORM (3D COllection FORMation, 3D-coform.eu) under grant number FP7 ICT 231809.

REFERENCES

[1] Jianer Chen and Ergun Akleman. Topologically Robust Mesh Modeling: Concepts, Data Structures and Operations. *Technical Report*, 1:1–18, 2003.

[2] Kin-Shing Cheng, Wenping Wang, Hong Qin, Kwan-Yee K. Wong, Huaiping Yang, and Yang Liu. Design and Analysis of Optimization Methods for Subdivision Surface Fitting. *IEEE Transactions on Visualization and Computer Graphics*, 13(5):878–890, 2007.

[3] Kin-Shing D. Cheng, Wenping Wang, Hong Qin, Kwan-Yee Kenneth Wong, Huaiping Yang, and Yang Liu. Fitting Subdivision Surfaces to Unorganized Point Data using SDM. *Proceedings of 12th Pacific Conference on Computer Graphics and Applications*, 1:16–24, 2004.

[4] Tony DeRose, Michael Kass, and Tien Truong. Subdivision Surfaces in Character Animation. *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, 25:85 – 94, 1998.

[5] Nick Gould, Dominique Orban, and Philippe Toint. Numerical methods for large-scale nonlinear optimization. *Acta Numerica*, 14:299–361, 2005.

[6] Mark L. Irons. The Curvature and Geodesics of the Torus. *Technical Report, Raindrop Laboratories*, 20:1–19, 2005.

[7] Weiyin Ma. Subdivision surfaces for CAD - an overview. *Computer-Aided Design*, 37(7):693–709, 2005.

[8] Martin Marinov and Leif Kobbelt. Optimization methods for scattered data approximation with subdivision surfaces. *Graphical Models*, 67(5):452 – 473, 2005.

[9] J. C. Nash. *Compact Numerical Methods for Computers: Linear Algebra and Function Minimisation*. Adam Hilger, second edition edition, 1990.

[10] Ahmad H. Nasri, Malcom Sabin, and Yasseen. Filling N-Sided Regions by Quad Meshes for Subdivision Surfaces. *Computer Graphics Forum*, 28:1644–1658, 2009.

[11] Jörg Peters and Ulrich Reif. *Subdivision Surfaces*. Springer, 2008.

[12] Jorg Peters and Xiaobin Wu. The distance of a subdivision surface to its control polyhedron. *Journal of Approximation Theory*, to appear:to appear, 2009.

[13] Rainer Storn and Kenneth Price. Differential Evolution: A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359, 1997.

[14] Torsten Ullrich, Volker Settgast, Ulrich Krispel, Christoph Fünfzig, and Dieter W. Fellner. Distance Calculation between a Point and a Subdivision Surface. *Proceedings of 2007 Vision, Modeling and Visualization (VMV)*, 1:161–169, 2007.

[15] Denis Zorin, Peter Schröder, Tony DeRose, Leif Kobbelt, Adi Levin, and Wim Sweldens. Subdivision for Modeling and Animation. *SIGGRAPH 2000 Course Notes*, 1:1–116, 2000.

[16] Denis Zorin, Peter Schröder, and Wim Sweldens. Interpolating Subdivision for Meshes with Arbitrary Topology. *Proceedings of 1996 ACM Siggraph*, 23:189–192, 1996.