Topmod: Interactive Topological Mesh Modeler

ERGUN AKLEMAN,[*] Visualization Sciences Program, Department of Architecture
VINOD SRINIVASAN Visualization Sciences Program, Department of Architecture
ESAN MANDAL Visualization Sciences Program, Department of Architecture
JIANER CHEN Department of Computer Science
ZEKI MELEK Department of Computer Science
ERIC LANDRENEAU Department of Computer Science

Texas A&M University

Abstract

This paper presents *Topmod*, a topologically robust mesh modeler. The system provides a large number of tools to manipulate the mesh structure 2-manifold polygonal meshes. Topmod allows users to easily create a wide variety of very high genus polygonal meshes. The system also provide tools to seamlessly texture map these high genus meshes.

1 Executive Summary

This paper introduces a new topologically robust polygonal modeler, called *Topmod*. The system is implemented in C++ using OpenGL and FLTK [FLTK 2004] and runs on IRIX, Linux and Windows platforms. Our main achievement with this modeling system is the development of new ways and tools to design polygonal meshes with huge number of handles, holes and columns, i.e., very high genus 2-manifold meshes. Topmod is a very dynamic and growing system. Its open architecture allows many people continuously add new new modeling and texturing tools. The current version of the system already includes a wide variety of tools that provide a large number of ways to manipulate 2-manifold polygonal meshes. The system is compatible with commercial modeling systems; i.e., the models created in this system are portable, and can be manipulated in other systems. Using TOPMOD, it is very easy to construct very complicated watertight shapes that can directly be built using rapid prototyping machines as shown in Figure 1.

Figure 1: Two photographs of a shape that is interactively created by using TOPMOD and built with a rapid prototyping machine. With TOPMOD, to create this shape took approximately 15 minutes.

Topmod has a very easy learning curve. Even novice users can quickly learn to create a wide variety of polygonal meshes with high number of holes, handles and columns. Figure 2 shows such an example of seamlessly textured polygonal mesh that is created by one of our students using the system, as one of the biweekly assignments of a geometric modeling class.

[*]Address: C218 Langford Center, College Station, Texas 77843-3137. email: ergun@viz.tamu.edu. phone: +(409) 845-6599.

Figure 2: An example of textured high-genus manifold mesh created by using Topmod.

Although the most important differentiating feature of the system is the robust and easy modeling of very high genus manifold meshes, the system has many additional features which can complement the high genus modeling tools. For instance, it provides a wide variety of remeshing tools which can be applied to polygonal manifolds. Using these tools, all semiregular mesh structures can be created. Based on these semi-regular mesh structures interestingly shaped holes can be opened. The texture mapping feature of the system is also compatible with the high genus modeling tools. For instance, the high genus shape shown in Figure 2 is seamlessly texture mapped. In this example, the mapped image has a regular pattern, but the system allows seamless texture mapping of even abstract paintings. One such example, a seamlessly mapped Miro painting to a spherical shaped mesh, is shown in Figure 3.

2 Introduction and Background

Polygonal modeling is the most widely used modeling approach in computer graphics applications. With the advent of subdivision surfaces, most users have converted back to polygonal modeling to create control surfaces for subdivision schemes. It is interesting to note that this popularity of polygonal modeling happened despite the limited tools provided by commercial modeling systems (The users find effective uses of existing tools and share their experiences

(A) (B)

Figure 3: (A) A tiled texture automatically synthesized from a Miro painting that is seamlessly mapped to a spherical shaped mesh. (B) An image that shows the tiles and tile boundaries. Texture is initially applied to a cube. The textured cube, then, is smoothed by using Catmull-Clark Subdivision. For computing texture coordinates, we used linear vertex-insertion subdivision scheme.

with other users by publishing trade articles and books).

We believe that the popularity of polygonal modeling comes from one of its under-appreciated advantages over other modeling approaches. If the polygons are not triangles or quadrilaterals, the faces are not geometrically well-defined. With geometrically ill-defined faces, self-intersection becomes meaningless. So, any commercial system that allows general polygons does not check self-intersection and avoids the cost of self-intersection computation which can considerably slow down the application during interactive modeling.

The omission of automatic self-intersection avoidance is typically not of concern to most users, since they can easily avoid self-intersection manually. Given a choice users usually prefer interactivity and higher speed in their applications. On the other hand, when users become more advanced, their main complaint becomes the limitations of the tools. For instance, opening a hole or adding a handle can require huge amount of manual work. Therefore, modeling a very complicated shape with huge number of holes and handles can be an uphill task even for experienced users.

The limitations of the polygonal modeling tools in commercial systems stems from polygonal mesh representations. Most commercial systems, for convenience, allow many non-manifold representations and manifolds with boundaries. Several manifold representations that are particularly useful for algorithm development are not considered valid even if the underlying data structure can support them. These decisions make sense in the early stages of commercial system development but eventually become a burden for tool development.

We observe that avoiding non-manifolds and allowing all possible manifold meshes can greatly simplify the algorithms and help develop new tools for interactive modeling. Avoidance of non-manifolds is not really an issue. Starting from a simple set of manifold meshes, manifold property can easily be guaranteed by using only manifold preserving operators. Examples of manifold preserving operators are Euler operators [Mantyla 1988], SPLICE operator [L. Guibas 1985] and INSERTEDGE operator [Akleman and Chen 1999].

The only real concern is the representation of all manifolds. There are two issues related to this problem: (1) The underlying data structure must support all manifold meshes, (2) Manifold preserving operations must be complete; i.e. they must be able to create all manifold meshes.

Representing all manifolds with existing data structures is not really really a hard problem. Most popular edge-based data structures such as half-edge, quad-edge and winged-edge [Mantyla 1988; L. Guibas 1985; Baumgart 1972] can represent all manifolds with a minor extension by including a manifold mesh with only one vertex and one face [Mantyla 1988]. Face based data structures, including the simple Wavefront OBJ file structure, can represent all manifolds.

Forming a complete set of operators requires a careful study. We have identified that the operators, CREATEVERTEX, DELETEVERTEX, INSERTEDGE and DELETEEDGE form such a complete and also minimal set. In the development of our system we use this set since INSERTEDGE and DELETEEDGE are particularly suitable for use as interactive operators.

3 Inspiration and Motivation

Once we have a system that allows all and only manifold meshes, it is possible to add a wide variety of tools. In this system, our main focus has been to create tools for very high genus modeling where the genus can even exceed hundreds and thousands.

The creation of polygonal meshes with genus greater than two has always been a research interest in computer graphics and shape modeling [Fomenko and Kunii 1997]. Ferguson, Rockwood and Cox used Bézier patches to create manifold surfaces with genus greater than two [H. Ferguson and Cox 1992]. Welch and Witkins used handles to design triangulated free-form surfaces [Welch and Witkin 1994]. Using morse operators and Reeb graphs, Takahashi, Shinagawa and Kunii developed a feature-based approach to create smooth 2-manifold surfaces with genus-more-than-two [S. Takahashi and Kunii 1997]. The work most closely related to ours is a generative modeling system that was developed by Havemann et al. to reconstruct some of the Architectural heritage of Europe such as high-genus gothic windows [Havemann 2003]. They use Euler operators [Mantyla 1988] as manifold preserving operators. The main difference between the two systems is that their approach is procedural; in our system we focus on development of tools for interactive modeling.

Our inspiration for high genus modeling came from Escher's drawings of rind shapes [Escher 1994], and the intricate nested carved sculptures of Asia. Figure 4A is an example of a nested elephant sculpture from India. A second elephant can be seen inside the first through the holes on outer surface. Our system allows users create such high genus shapes; for example, the outer shape in Figure 4B is a rind model created using our system and rendered with bump mapped textures.

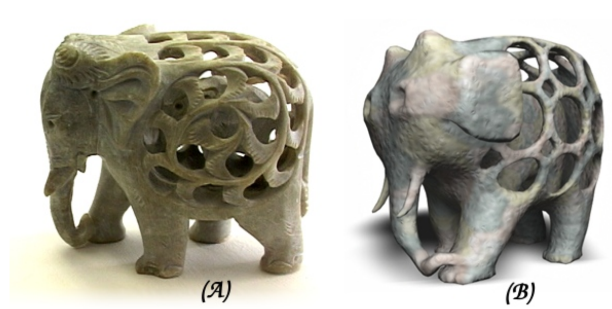


Figure 4: An example of nested sculptures from India. (A) shows a real nested elephant sculpture. (B) is a rind model created using our system and rendered with bump mapped textures.

As we have discussed earlier the design philosophy of most existing commercial systems does not allow the development of tools for creating such highly decorated structures with huge number of holes and handles. Many modeling programs allow rendering of wireframes or texture mapping with transparency that can be used to give an illusion of many holes; however such renderings lack the depth and richness of renderings of 3d models.

Although our inspiration came from art, high genus shapes are extremely common and useful in industrial applications. Many man-made objects have many holes and handles. Examples include teapots, masks, boxes and even houses. In some cases, the number of holes and handles goes beyond hundreds and even thousands, such as in the highly decorative embellishments used widely in architecture. In fact, the development phase of the system was influenced by the functional and particularly aesthetic needs of visual designers and architects. We continue to try to cater to the needs of design students with new tools and modeling solutions.

4 Design Philosophy and System Architecture

Our design philosophy was to develop a system architecture that will allow even novice programmers to easily add new capabilities and tools. The large number of tools in the current system that have been added by students with no formal computer science background is testament to the success of our approach.

As we have discussed earlier, only four operators, CREATEVERTEX, DELETEVERTEX, INSERTEDGE and DELETEEDGE are sufficient to create all and only orientable manifold meshes; any mesh operation can be implemented using only these four operators. Thus, once programmers understand the effects of these operators over manifold meshes, they can easily develop their own higher level mesh operations.

The underlying data structure used to represent meshes in our system is the Doubly Linked Face List (DLFL) [Akleman and Chen 1999]. The four fundamental operators have been directly implemented over DLFL. All other algorithms for higher level mesh operations are implemented using these fundamental operators. By de-coupling the implementation of higher level mesh operations from the underlying data structure we have been able to make the system very modular and easily extensible. Developers need not know the nuts and bolts of the DLFL data structure to be able to add new tools and features. Since access to the internal mesh representation is only through the four fundamental operators, maintaining topological validity of the mesh representation is also easy to guarantee. The higher level operations are also directly portable since they will work on any polygonal data structure which provides the four fundamental operators.

The system is modular and hierarchical. The DLFL data structure and the four fundamental operators form the core of the system. Commonly used sequences of fundamental operators have been grouped into high-level operators which form the next layer. The core components along with the next level of high-level operators are provided as a single library. Complex mesh operations such as remeshing and high genus modeling, can make use of these high-level operators, thereby simplifying code development.

An additional simplification comes from the development of the algorithms. Using these four operators, the algorithms for the implementation of operations can be as simple as drawing the steps of algorithm.

For efficient use of resources, we focus on only the tools that are not provided by commercial polygonal modeling systems. As a result of this approach, the current capabilities and tools of Topmod complement the capabilities of the existing commercial systems. Moreover, Topmod is compatible with commercial systems and one can easily transfer models back and forth between Topmod and a commercial package such as Maya, Softimage or 3D Studio Max. This design philosophy has made Topmod very popular among the students of our graduate program. They can still use commercial systems and whenever they need operations that are not provided in the commercial system, such as remeshing, hole opening or handle creation, they export the model to Topmod and apply the desired operations.

5 Extrusions and Stellations

The system provides a set of operators that are applied to only one face of the mesh. These operators do not affect the rest of the mesh, i.e. boundary edges of the chosen face stay the same. We have two types of such operators: (1) Extrusions that are generalized cylinders in which bottom and top polygons have the same number of sides, and (2) Stellations that are generalized pyramids, where there is a top vertex instead of top polygon.

Our extrusions and stellations are a generalization of classical extrusion shown in Figure 5. Classical extrusion has always been one of the most under-rated yet commonly-used operator in modeling. With the advent of Catmull-Clark subdivision, it became especially useful in modeling subdivision control meshes. This extrusion operator is a natural choice for quadrilateral meshes. For other mesh types and subdivision schemes such as triangular meshes and Loop subdivision, it is useful to have other types of operations that can be applied to only one face of the mesh.

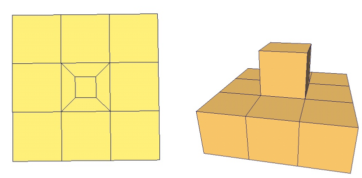


Figure 5: Classical (hexahedral) extrusion.

We have identified five new extrusions and stellations; four of them are based on platonic solids. When a classical extrusion is applied to a square, it can give a cube. Similarly, when a dodecahedral extrusion is applied to a pentagon, it should give a dodecahedron. Based on this observation, we have identified the four operators as tetrahedral, octahedral, dodecahedral and icosahedral extrusions. Among these, tetrahedral and octahedral extrusions are particularly useful for mesh manipulations.

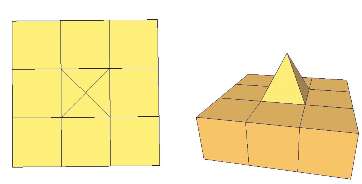


Figure 6: Stellation (Tetrahedral extrusion).

The Figures 6 and 7 show how the tetrahedral and octahedral extrusions affect a square. The left images are Schlegel diagrams of

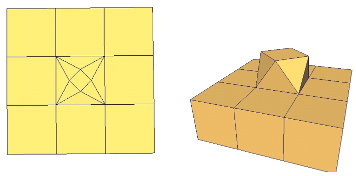


Figure 7: Octahedral extrusion.

extrusions [Williams 1972]. Among these operators, tetrahedral extrusion is also known as a stellation operation in polyhedral modeling [Williams 1972]; and it is used to create Kepler solids such as a small stellated dodecahedron. We also introduced a double stellation operator shown in Figure 8 to create star shaped ornaments. Double stellation is also useful to create faux-DePoinsot solids such as the great dodecahedron and great icosahedron. (These are not real DePoinsot solids, since the real ones have to be self-intersected.)

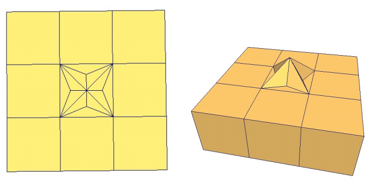


Figure 8: Double-stellation extrusion.

6 Remeshing Operations

The system also provides (global) operators that are applied to the whole mesh, or what we call remeshing operators. Under the remeshing category, we only consider global operators that do not change the topology of the mesh. These operators do not increase or decrease the genus of the surface nor do they connect or disconnect surface components.

The most important remeshing scheme in our system is the dual operator [Williams 1972], which is essential for creating dual meshes. Dual is also very useful operator for smoothing. If a dual operation is applied even number of times, in most cases it will create a smooth mesh. Zorin and Schröder recently showed that the dual operation allows creation of higher-order quadrilateral subdivision surfaces that provide higher-degree continuity [Zorin and Schröder 2002]. The same idea is also used to derive the rules for the schemes that create regular $(6,3)$ regions [J. Claes and Reeth 2002; Oswald and Schröder 2003].

All remeshing operators we provide (except dual) increase the number of vertices. Our remeshing operators are not probabilistic, i.e., they always give the same result for the given initial mesh. Although, we consider the subdivision schemes as remeshing schemes, a remeshing scheme does not have to be a subdivision – remeshing schemes may not necessarily be applied successively and may not smooth the mesh.

Providing a large variety of remeshing operators is particularly useful to making mesh structures ornamental or decorative. By applying provided remeshing operators to regular regions all major semiregular regions can be obtained. To show how to create particular semiregular regions we apply remeshing operators to three initial spherical shaped meshes shown in Figure 9. Among these three, spherical cube (SC) includes $(4,4)$ regular regions, and geodesic dome (GD) [Stewart 1991] includes $(3,6)$ regular regions, and finally Buckminsterfullerene (BF) [Stewart 1991] includes $(6,3)$ regular regions[1] (Note that these regions are only topologically regular, they are not regular geometrically. Geometric entities such as the angles and edge lengths are not the same and do not have to be the same.)

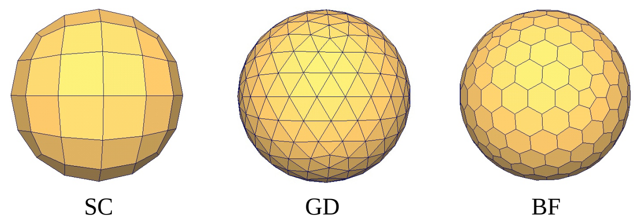


Figure 9: Initial meshes. Spherical cube (SC) is obtained by mapping vertices of a cube to a unit sphere. GD is a Geodesic dome or Pseudo-Icosahedron, and BF is a Buckminsterfullerene or generalized soccerball.

We organized remeshing operators (except dual) under two categories: (1) Conversion operators that change any given mesh into a mesh that consist of the same type of faces (e.g. all triangles or all pentagons) or vertices (e.g., all valent-3) and (2) preservation operators that preserve the property (e.g. all quadrilaterals) of the mesh when applied to a mesh that consists of the same type of faces or vertices. Operators are also classified into two additional categories: (1) Primary operators that create the same type of faces, and (2) dual operators that create the same type of vertices. Although the dual of an operator can simply be obtained as dual + operator + dual, in most cases we have implemented dual operators separately to achieve faster computational speed.

6.1 Triangulization and Valent-3 Conversion

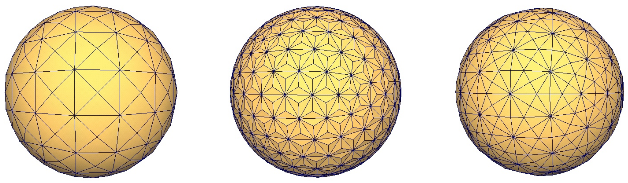
The system provides three triangulization and three valent-3 conversion operators.

- $\sqrt{3}$ *and its dual.* The remeshing scheme of $\sqrt{3}$ subdivision introduced by Kobbelt [L. 2000] is the only subdivision scheme that converts all faces to triangles. Dual of $\sqrt{3}$ has been discovered by three groups simultaneously [J. Claes and Reeth 2002; Akleman and Srinivasan November 2002; Oswald and Schröder 2003]. It converts any manifold mesh to a mesh with only 3-valent vertices.

- *Vertex Trunctation (Trunc) and its dual (DualTrunc).* Polyhedral vertex truncation is a widely used operator in polyhedral modeling to obtain Archimedean solids [Williams 1972]. We extended polyhedral vertex truncation by adopting Chaikin's algorithm [Sabin 2000] to vertex truncation. Vertex truncation can be used to create semi-regular regions 4.8.8 and 3.12.12 as shown in Figure 11. Vertex tructation is a valent-3 conversion scheme. Dual of Vertex truncation, a triangulization scheme, is also implemented and shown in Figure 10.

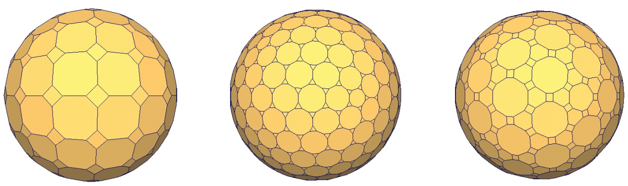
[1] The symbol (n,m) is called the Schlafli symbol for regular regions where n is the number of the sides of faces and m is the valence of vertices. For instance, $(6,3)$ means regular regions that consist of hexagons with 3-valent vertices. Schlafli symbol for semi-regular meshes is different. For example, it is given as 3.3.4.3.4, this symbol means a semi-regular region that consists of either pentagons with vertex valences 3,3,4,3 and 4 in the same order (for primary meshes) or its dual (for dual meshes).

- *1264 and its dual.* We have recently developed these two remeshing operators to create semi-regular region 12.6.4 when applied to the regular region (3,6) as shown in Figure 10 and 11.



SC + DualTrunc
Primary: 4.8.8 | GD + DualTrunc
Primary: 3.12.12 | GD + 1264
Primary: 12.6.4

Figure 10: Triangulizations that create semiregular regions.



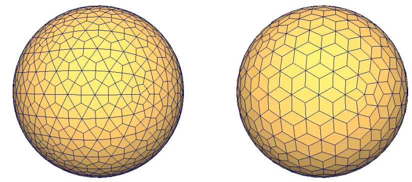
SC + Trunc
Dual: 4.8.8 | BF + Trunc
Dual: 3.12.12 | BF + Dual1264
Dual: 12.4.6

Figure 11: Valent-3 conversions that create semiregular regions.

6.2 Quadrilateralization and Valent-4 Conversion

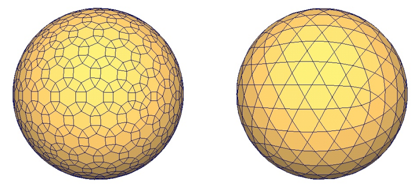
The system provides two quadrilateralization and two valent-4 conversion operators.

- *Vertex Insertion (VertIns) and its dual, Corner Cutting (CorCut).* Vertex Insertion is remeshing scheme of Catmull-Clark [Catmull and Clark 1978] subdivision. The system provides both Catmull-Clark and linear vertex insertion. This subdivision creates $(4,4)$ regular regions. But, if it is applied once to a mesh with a $(6,3)$ region, it creates semi-regular regions 6.4.3.4 as shown in Figure 12. The corner cutting scheme is the dual of vertex insertion. An example of corner cutting is remeshing algorithm of Doo-Sabin subdivision [Doo and Sabin 1978]. After one application of corner cutting the valence of all the vertices becomes 4. Corner cutting is a very useful remeshing operator that can create interesting structures such as the one shown in Figure 12.
- *Simplest (Simp) and its dual (DualSimp).* The remeshing scheme of simplest subdivision [Peters and Reif 1997] converts any mesh to a mesh with only 4-valent vertices. Simplest creates 6.3.6.3 semi-regular regions when it is applied to (6,3) regular regions as shown in Figure 13. We have also implemented the dual of simplest. In the program, We call it stellate with edge removal. Dual of simplest is not a known subdivision scheme and it is not useful as a smoothing scheme. However, it allows the creation of appealing 6.3.6.3 semi-regular regions as shown in Figure 12.



BF + VertIns
Primary: 6.4.3.4 | BF + DualSimp
Primary: 6.3.6.3

Figure 12: Quadrilateralization.



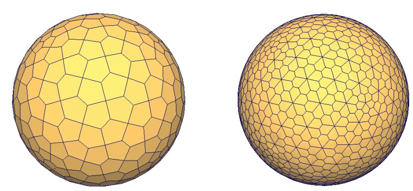
BF + CorCut
Dual: 6.4.3.4 | BF + Simp
Dual: 6.3.6.3

Figure 13: Valent-Four Conversion.

6.3 Pentagonalization and Valent-5 Conversion

We have recently shown that it is possible to convert any 2-manifold to a pentagonal mesh. We also showed that there is no conversion method that can convert any given mesh to a mesh consisting of only polygons with more than five sides.

- *Pentagonalization and its dual.* This is the only known pentagonalization algorithm, which we have introduced recently This scheme can create two types of semi-regular regions: 3.4.3.3.4 and 3.3.3.3.6. We have also implemented its dual which converts every vertex of a mesh to valent-5 vertex.



SC + Pent
Primary: 3.4.3.3.4 | BF + Pent
Primary: 3.3.3.3.6

Figure 14: Pentagonalization.

6.4 Preservation Schemes

We have included six preservation schemes. Note that hexagonal preservation operators exists since hexagonal meshes can represent shapes with a genus higher than 1.

- *Triangle and Valent-3 Preservation.* Remeshing algorithm of the Loop subdivision [Loop 1987] is not a conversion scheme. If it is applied to a triangular mesh, it creates a triangular mesh. Loop is also a valent-6 preservation scheme. However, if it is applied to a mesh that has non-triangular faces, they continue to exist. Dual of Loop, which has recently been discovered by two groups [N. Dyn and Simoens 2002; Prautzsch and Boehm 2000; Oswald and Schröder 2003], is both valent-3 and hexagon preservation scheme.
- *Quadrilateral and Valent-4 Preservation.* We have developed and included a new scheme called checkerboard, which is a

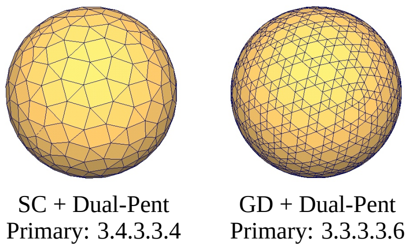


Figure 15: Valent-5 conversion.

generalization of the vertex insertion scheme. We also included the dual of checkerboard scheme.

- *Pentagon and Valent-5 Preservation.* We have recently developed a pentagon preservation scheme and its dual.
- *Hexagon and Valent-6 Preservation.* (See Triangle and Valent-3 Preservation.)

7 High Genus Modeling

Our system provides a wide variety of methods and tools to create high genus meshes. We can classify these tools into three categories: (1) Interactive, (2) Semi-Automatic and (3) Automatic methods.

The system provides several interactive methods including Multi-Segment Curved Handles, and one semi-automatic method, Rind Modeling. In both these methods, the users increase the genus one-by-one, controlling where to open holes and where to add handles. For creating very high genus shapes, it is hard to open holes and add handles one-by-one. We have been developing a set of new tools to automatically create very high genus meshes. The current automatic methods includes three methods (1) Generalized & Connected Sierpinski Polyhedra, (2) Wire Modeling and (3) Column Modeling.

7.1 Multi-Segment Curve Handles

This method allows users to simply and easily create multi-segment, curved handles between two star-shaped faces of an orientable 2-manifold mesh, or to connect two 2-manifold meshes along such faces. The handle creation algorithm combines a very simple 2D morping algorithm with a Hermite interpolation to construct the handle. The tool can be used for handle creation (by adding a handle to a surface) and for surface blending (by connecting two distinct surfaces and smoothing). Both applications of the algorithm are useful to designers for creating manifolds of high genus.

A handle is not an extrusion (or lofting) and handle creation is not simply an extrusion method. Handle creation is a topological operation and it requires topological consistency. Therefore, handle creation methods are different from extrusion methods since they are required to guarantee topological consistency. This method not only guarantees 2-manifold property of final mesh, but in every stage of our handle creation meshes continue to be 2-manifold.

The meshes constructed with multi-segment curve handles can be smoothed by any subdivision scheme, since the manifold property is preserved after handle creation. Figure 16 shows two views of a genus-6 shape created by our approach combined by subdivision schemes. To create this object, we first created 6 multi-segment curved handles by connecting pairs of neighboring faces of a dodecahedron, and then smoothed the resulting mesh using one iteration of the Doo-Sabin and two iterations of the Catmull-Clark subdivision schemes [Catmull and Clark 1978; Doo and Sabin 1978; Zorin and P. Schröder 2000].

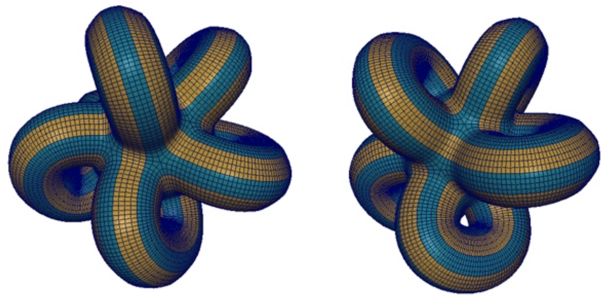


Figure 16: Two views of a genus-6 manifold mesh created by using multi-segment curve handles.

Although our system can only work on orientable surfaces, non-orientable appearing surfaces can easily be obtained with this method.Figure 17 shows a non-orientable look-a-like shape, a Möbius band. Klein-bottle look-a-like can also be constructed.

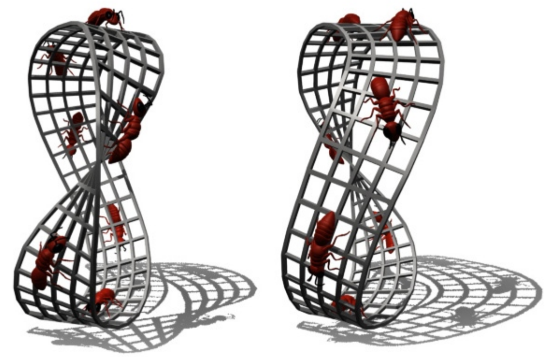


Figure 17: An homage to Escher: *Band Van Möbius II*. (Ant model was created separately.)

7.2 Rind Modeling

Rind modeling provides for the easy creation of surfaces resembling peeled and punctured rinds. Rind modeling consists of two steps:

1. *Creation of a shell or crust like the rind of an orange:* in this (automatic) step, for any given 2-manifold mesh surface an offset surface is created based on a user defined thickness parameter. As a result of this step, from one surface two similar surfaces, which can be considered as a shell or a crust are created.
2. *Opening holes in the crust by punching or peeling:* This interactive step consists of two modes, hole punching and peeling, similar to punching holes in a coconut husk or peeling an orange rind.

The rind modeling method can also be used to create shapes that do not necessarily look like rind shapes. An example of such uses is shown in earlier Figure **??**. The surface in this figure looks more like an extruded surface than a rind surface, but it cannot be created as a simple extrusion because of its branching structure.Being able to create a wide variety of semi-regular regions, are useful for opening interesting holes as shown in Figure 18.

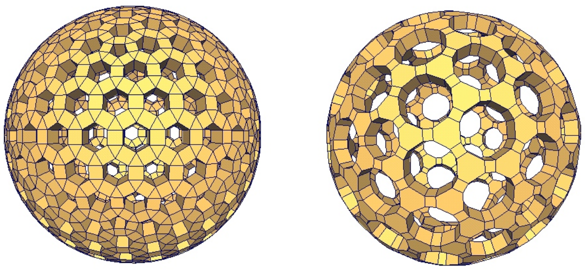


Figure 18: Rind modeling over semi-regular meshes.

7.3 Generalized & Connected Sierpinski Polyhedra

Our system provides a subdivision scheme to construct generalized Sierpinski polyhedra. Unlike usual Sierpinski polyhedra construction schemes [Mandelbrot 1980; Barnsley 1988], which create either an infinite set of disconnected tetrahedra or a non-manifold polyhedron, our robust construction scheme creates a single connected and manifold polyhedron as is evident in Figure 19. Moreover, unlike the original schemes, this new scheme can be applied to any manifold polyhedral mesh and based on the shape of this initial polyhedra a large variety of Sierpinski polyhedra can be obtained. The simplicity of this algorithm comes directly from using manifold preserving operators.

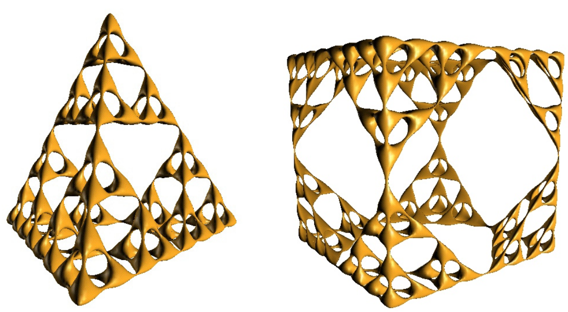


Figure 19: An example of smoothed (A) Sierpinski tetrahedron and (B) Sierpinski cube. We have applied 3 iterations of our Sierpinski subdivision to a (A) tetrahedron and (B) cube then smoothed them using Doo-Sabin subdivision.

7.4 Wire and Column Modeling

These two tools allow users to model shapes with large number of holes that are made from wires and columns. They allows users to replace each edge of a given mesh with a *"3D pipe"*. The system guarantees that the pipes are connected and the resulting shape can be physically constructed.

Although, both wire and column modeling are essentially same types of operations, the resulting mesh structures are very different. In case of wire modeling, cross-sections of 3D pipes are squares and their connections in each vertex are created by a corner cutting method. On the other hand, in column modeling cross-sections of pipes can be any regular *n*-gon, where *n* is given by the users.

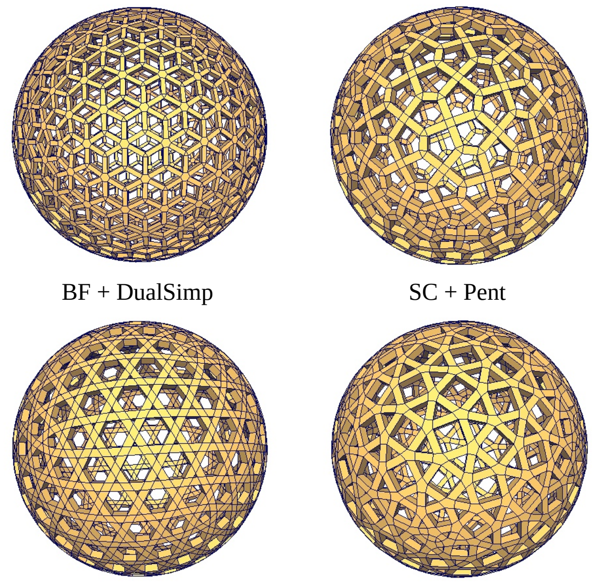


Figure 20: Wire modeling over semi-regular regions.

The remeshing schemes we have discussed earlier are very useful for creating artistically interesting mesh structures that can be used in wire and column modeling. It is possible create a wide variety of mesh structures by applying several combinations of remeshing schemes (For examples of wire modeling see Figures 20, 21 and 22). Figure 23 shows an example to column modeling.

Figure 21: Wired caricature of Humphrey Bogart.

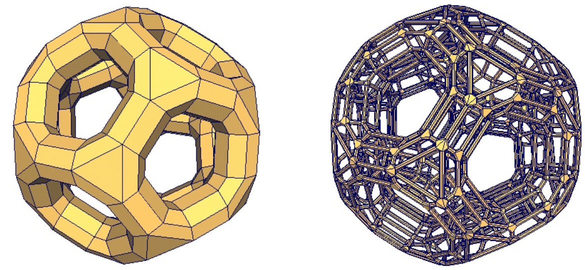
Figure 22: Wired rabbit.

8 Texturing

Our system also provides a simple and practical texturing technique for seamlessly texturing quadrilateral meshes. Our technique is an extension of Wang tiling techniques recently introduced by Stam [Stam 1997], Neyret and Cani [Neyret and Cani 1999] and Cohen et al. [M. Cohen and Deussen 2003]. Earliest work is [Stam 1997]. Stam uses quadrilateral tiles and requires genus-1 surfaces (i.e., planar tilings) Neyret and Cani has the most general mapping method which uses triangular tiles and can can be mapped to any triangular mesh without any discontinuity or singularity. Among the three, only [M. Cohen and Deussen 2003] provides a texture synthesis method, but it also requires genus-1 surfaces.

Topmod provides a texture mapping method based on quadrilateral tiles. It can texture map any quadrilateral mesh regardless of its genus. We also provide texture synthesis using a separate system. With this system any image can be converted to a ”tiled” texture that can be mapped to any quadrilateral mesh without any discontinuity or singularity. Users interactively synthesize texture image files from existing images with a simple interface. These texture image files are called tiled textures since they consist of a set of square tiles. Consistent organization of these tiles in the texture image yields an extremely simple texture mapping. Topmod assigns texture coordinates to each corner of the mesh while providing seamless and continuous boundaries across the surface. Once the texture coordinates are assigned, the resulting textured mesh can be used in any commercial system such as Maya or Softimage. Using only one texture file is crucial for compatability with commercial systems since most commercial renderers (like those in Maya, Softimage etc), allow only one single rectangular texture image to be read for each parameter of a shader.

Figure 3 shows a tiled texture automatically synthesized from a Miro painting that is seamlessly mapped to a spherical shaped mesh. The texture is initially applied to a cube. The textured cube, is then smoothed by using Catmull-Clark subdivision [Catmull and Clark 1978] while the texture coordinates are computed using a linear vertex-insertion remsshing scheme. [Sabin 2000; Zorin and P. Schröder 2000; Zorin and Schröder 2002].



Column Modeling once Column Modeling twice

Figure 23: Column Modeling.

9 Conclusion and Future Work

In this paper, we have introduced a new topologically robust polygonal modeler, called *Topmod*. Topmod is being widely used in our graduate program as a high-genus modeling package. We observe that the functions and tools of Topmod can be utilized in variety of design disciplines. In architecture, it can be used to enhance basic forms with high genus structures. It can also be used for modeling new forms especially in early design phase and conceptual design. In visual communication design, Topmod can be used for modeling real and conceptual shapes for graphic design and communication. It can also be utilized in design education. It can enhance basic design courses with a user friendly and flexible digital tool. It can help understand and create complex geometries and relations with hands-on experience.

Our system has already become a phenomenal success among our students. From the programmer's point of view, using the provided operator set, it is very easy to extend the capabilities of our system by writing simple subroutines for new high level user operators. From the user's point of view, it is extremely easy to learn and use our system. Although topology change is considered a challenging mathematical concept, the users of our system can easily learn it and create complicated meshes. Many high-genus 2-manifold meshes have been created by our students as their weekly projects in a regular graduate level shape modeling course, using our software.

The system currently includes a couple of experimental features which are still under development. One is the ability to create handles between multiple faces. This will allow the user to select three or more faces and connect them in a single step to produce a clean handle (without too much skew or asymmetry) connecting all the faces. Another experimental high-genus modeling operation is the creation of generalized Menger sponges. This extends the Menger sponge [Mandelbrot 1980] to shapes beyond a cube, producing similar shapes starting from a wide variety of initial surfaces.

References

AKLEMAN, E., AND CHEN, J. 1999. Guaranteeing the 2-manifold property for meshes with doubly linked face list. *International Journal of Shape Modeling 5*, 2, 149–177.

AKLEMAN, E., AND SRINIVASAN, V. November 2002. Honeycomb subdivision. In *Proceedings of ISCIS'02, 17th International Symposium on Computer and Information Sciences*, vol. 17, 137–141.

BARNSLEY, M. 1988. *Fractals Everywhere*. Academic Press, Inc. San Diego Ca.

BAUMGART, B. J. 1972. *Winged-edge polyhedron representation*. Master's thesis, Technical Report CS-320, Stanford University.

CATMULL, E., AND CLARK, J. 1978. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer Aided Design*, 10, 350–355.

DOO, D., AND SABIN, M. 1978. Behavior of recursive subdivision surfaces near extraordinary points. *Computer Aided Design*, 10, 356–360.

EFROS, A. A., AND FREEMAN, W. T. 2001. Image quilting for texture synthesis and transfer. In *Proceedings of SIGGRAPH 2001*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 341–346.

ESCHER, M. C. 1994. *The graphic works: introduced and explained by the artist (image plates 41 and 43)*. Barnes and Nobles Books, New York.

FLTK, 2004. The fast light toolkit home page:. http://www.fltk.org/.

FOMENKO, A. T., AND KUNII, T. L. 1997. *Topological Modeling for Visualization*. Springer-Verlag, New York.

GENERALIZED-SIERPINSKI. 2004. Connected and manifold sierpinski polyhedra. In *Unpublished Supplemental Materials - Chapter 4*, 43–51.

H. FERGUSON, A. R., AND COX, J. 1992. Topological design of sculptured surfaces. In *Proceedings of SIGGRAPH 1992*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 149–156.

HAVEMANN, S., 2003. Generative modeling,. Internet: graphics.tu-bs.de/gml.

HOFFMANN, C. M. 1989. *Geometric & Solid Modeling, An Introduction*. Morgan Kaufman Publishers, Inc., San Mateo, Ca.

INTERACTIVE-DEMOS, 2004. interactivedemos.mov,. Supplemental Video.

J. CLAES, K. B., AND REETH, F. V. 2002. A corner-cutting scheme for hexagonal subdivision surfaces. In *Proceedings of Shape Modeling International'2002, Banff, Canada*, 13–17.

L. GUIBAS, J. S. 1985. Primitives for the manipulation of general subdivisions and computation of voronoi diagrams. *ACM Transaction on Graphics*, 4, 74–123.

L., K. 2000. $\sqrt{3}$-subdivision. In *Proceedings of SIGGRAPH 2000*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 103–112.

LOOP, C. 1987. *Smooth Subdivision Surfaces Based on Triangles*. Master's thesis, University of Utah.

M. COHEN, J. SHADE, S. H., AND DEUSSEN, O. 2003. Wang tiles for image and texture generation. In *Proceedings of SIGGRAPH 2003*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 287–294.

MANDELBROT, B. 1980. *The Fractal Geometry of Nature*. W. H. Freeman and Co., New York.

MANIFOLD-CONVERSION. 2004. Topological construction of 2-manifold meshes from arbitrary polygonal data. In *Unpublished Supplemental Materials - Chapter 5*, 53–61.

MANTYLA, M. 1988. *An Introduction to Solid Modeling*. Computer Science Press, Rockville, Ma.

N. DYN, D. L., AND SIMOENS, J. 2002. Face-value subdivision schemes on triangulations by repeated averaging. In *Curve and Surface Fitting: Saint-Malo 2002*, 129–138.

NEYRET, F., AND CANI, M. 1999. Pattern based texturing revisited. In *Proceedings of SIGGRAPH 1999*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 235–242.

OSWALD, P., AND SCHRÖDER, P., 2003. Composite primal/dual $\sqrt{3}$-subdivision schemes. Computer Aided Geometric Design, CAGD.

PETERS, J., AND REIF, U. 1997. The simplest subdivision scheme for smoothing polyhedra. *ACM Transactions on Graphics 16*, 4, 420–431.

PRAUTZSCH, H., AND BOEHM, W., 2000. Chapter: Box splines. The Hanbook of Computer Aided Geometric Design.

S. TAKAHASHI, Y. S., AND KUNII, T. L. 1997. A feature-based approach for smooth surfaces. In *Proceedings of Fourth Symposium on Solid Modeling*, 97–110.

SABIN, M., 2000. Subdivision: Tutorial notes. Shape Modeling International 2001, Tutorial, May.

SEMI-REGULAR. 2004. Semi-regular pentagonal subdivision. In *Unpublished Supplemental Materials - Chapter 2*, 23–32.

SHAPES, 2004. shapes.mov,. Supplemental Video.

SMOOTH-FRACTAL. 2004. Smooth fractal polyhedra. In *Unpublished Supplemental Materials - Chapter 6*, 63–68.

STAM, J., 1997. Aperiodic texture mapping. Technical Report R046.

STEWART, I. 1991. *Game, Set and Math: Enigmas and Conundrums*. Penguin Books, London.

STUDENT-WORKS. 2004. Student works. In *Unpublished Supplemental Materials - Chapter 7*, 69–70.

THEORY. 2004. Topologically robust mesh modeling: Concepts, data structures and operations. In *Unpublished Supplemental Materials - Chapter 1*, 5–21.

TILED-TEXTURES. 2004. Tiled textures. In *Unpublished Supplemental Materials - Chapter 3*, 33–41.

WELCH, W., AND WITKIN, A. 1994. Free-form shape design using triangulated surfaces. In *Proceedings of SIGGRAPH 1994*, ACM Press / ACM SIGGRAPH, Computer Graphics Proceedings, Annual Conference Series, ACM, 247–256.

WILLIAMS, R. 1972. *The Geometrical Foundation of Natural Structures*. Dover Publications, Inc.

ZORIN, D., AND P. SCHRÖDER, E., 2000. Subdivision for modeling and animation,. ACM SIGGRAPH 2000 Course #23 Notes, July.

ZORIN, D., AND SCHRÖDER, P., 2002. A unified framework for primal/dual quadrilateral subdivision schemes. Computer Aided Geometric Design, CAGD.