

Ergun Akleman · Vinod Srinivasan · Jianer Chen · David Victor  
Morris · Stuart Tosten Tett

# TopMod3D

## An Interactive Topological Mesh Modeler

**Abstract** This paper presents *TopMod3D* our topologically robust mesh modeler. The system provides a large number of tools to manipulate the mesh structure of 2-manifold polygonal meshes. *TopMod3D* allows users to easily create a wide variety of very high genus polygonal meshes.

**Keywords** Shape Modeling · Topological Modeling · Application · System Development

---

### 1 Introduction

This paper introduces a topologically robust polygonal modeler, called *TopMod3D* (a.k.a. TopMod). The concepts and algorithms behind the tools in TopMod have been developed, implemented and published by our research group such as [1–3, 8, 9, 26, 10, 35]. Many of these tools are unique to TopMod.

The initial version of the software, TopMod 1.0, has been available as free software since 2003 from Ergun Akleman’s web-site. Since then, several talented artists created very interesting sculptures using TopMod 1.0. TopMod 1.0 was implemented in C++ using OpenGL and FLTK [18]. It runs on Mac, Linux and Windows platforms.

In August 2007, we released a new version, TopMod 2.0, with an improved user interface and scripting editor [39, 30]. For the interface of the new version, we switched from FLTK to Qt [41]. The new version also runs on Mac, Linux and Windows platforms. We have also developed a web-site to create a community around the software [40]. Since then, more than 15,000 people have downloaded the software.

---

E. Akleman, V. Srinivasan, D. V. Morris, S. T. Tett  
Visualization Department, Texas A&M University  
E-mail: ergun@viz.tamu.edu, vinod@viz.tamu.edu

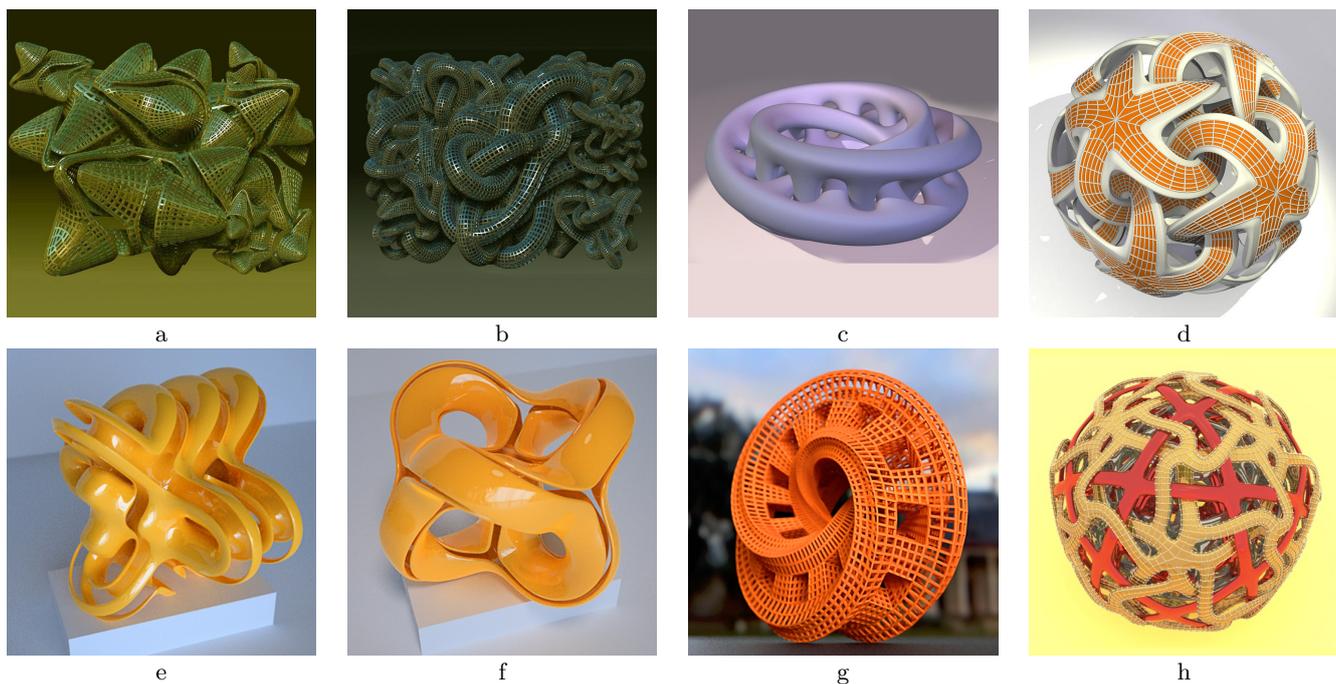
J. Chen  
Computer Science Department, Texas A&M University  
E-mail: chen@cs.tamu.edu



**Fig. 1** The photographs of two sculptures that were interactively designed by Ergun Akleman using *TopMod 1.0* and built with a rapid prototyping machine. These sculptures were selected and exhibited in Siggraph [5] and Bridges 2006 Art Exhibitions [7]. Creating each shape in TopMod 1.0 took approximately 15 minutes. With the selection mechanisms provided in the latest version, it takes only a few minutes to create each shape.

This experience is a strong example of the importance of creation of a community for the useability of software. For instance, many people discovered ways to create unusually interesting shapes and shared their experiences by developing video tutorials. Other users, following video tutorials, created similar shapes. Having a community also helped to solve portability problems. For instance, the script editor was initially developed on the Mac platform and we had trouble compiling the code for Windows. One user from Italy provided a solution to this problem. Another user from France translated the user interface from English to French.

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 [36, 26, 35, 27, 37]. TopMod is a very dynamic and growing system. Its underlying data structure and minimal set of operations help to keep algorithms simple while guaranteeing 2-manifold property of resulting meshes. This



**Fig. 2** Example of high-genus manifold meshes created by international users of TopMod. The models shown in (a) and (b) were created by user frenchpilou from France, (c) and (d) were created by Jonathan Johanson from Germany, (e), (f) and (g) were created by Torolf Sauermann from Germany, and finally (h) was created by Ranjith Perumalil, who is one of the students of Visualization program in Texas A&M University. (Final renderings are not done in TopMod. Each person used a different commercial renderer. )

allowed us to add new modeling tools with minimal effort.

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 like those shown in Figure 1.

One of the reasons behind the popularity of TopMod is that it has a very easy learning curve. Even novice users can quickly learn to create a wide variety of polygonal meshes with large number of holes, handles and columns. Figure 2 shows several examples of manifold meshes created by users of 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 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 semi-regular mesh structures can be created. Based on these semi-regular mesh structures holes with complex shapes can be opened. The current version provides a powerful selection mechanism

that speeds up the design process. For instance, with *TopMod 1.0*, creating each shape in Figure 1 takes approximately 15 minutes. With the selection mechanisms provided in the latest version, it takes only a few minutes to design similar shapes. TopMod 2.0 also provides a script editor that allows users writing code to algorithmically create shapes.

## 2 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.

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 may not be guaranteed to be geometrically well-defined. When we allow geometrically ill-defined faces, checking self-intersection also 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 limitation 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 [29], SPLICE operator [20] and INSERTEDGE operator [1].

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 a hard problem. Most popular edge-based data structures such as half-edge, quad-edge and winged-edge [29, 20, 11] can represent all manifolds with a minor extension by including a manifold mesh with only one vertex and one face [29]. 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 [3]. 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 [19]. Ferguson, Rockwood and Cox used Bézier patches to create manifold surfaces with genus greater than two [21]. Welch and Witkins used handles to design triangulated free-form surfaces [42]. Using Morse operators and Reeb graphs, Takahashi, Shinagawa and Kunii developed a feature-based approach to create smooth high-genus 2-manifold surfaces [38]. 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 [22]. They use Euler operators [29] 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 artistic inspiration for high genus modeling came from Escher's drawings of rind shapes [17], and the intricate nested carved sculptures of Asia. 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. Some tools in the current system have been added by students with no formal computer science background, which is a 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) [1]. 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 popular. The users 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 Modeling Tools of TopMod

Modeling tools provided by TopMod can be classified under four categories. (1) Basic Operators, (2) Extrusions, (3) Remeshing Operations, and (4) High-genus operations. Most of these tools were already available in original TopMod but some are slightly improved in the new version.

1. **Basic Operators:** TopMod 1.0 provides INSERTEDGE, DELETEEDGE, a subset of SPLICE operator and subdivide edge operator for interactive modeling. CREATEVERTEX and DELETEVERTEX are available only in scripting mode.
2. **Generalized Extrusions:** These are the operators that are applied to only one face of the mesh without affecting the rest, i.e. boundary edges of the chosen face stay the same. These operators are a generalization of classical extrusion. TopMod includes extrusions such as dodecahedral or icosahedral extrusions that do not exist in any other software. These extrusions provides generalization of platonic solids [44] and can create platonic solids when they are applied to the native face of the related platonic solid. For instance, when a dodecahedral extrusion is applied to a pentagon, it gives a dodecahedron. TopMod provides planar and non-planar versions of tetrahedral, octahedral, cubical (classical), dodecahedral and icosahedral extrusions.
3. **Remeshing Operators:** Under the remeshing category, we consider all 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 [44], which is essential for creating dual meshes. Dual is also very useful operator for smoothing and allows creation of higher-order subdivision schemes that provide higher-degree continuity [45]. In addition to Dual operator, almost all existing subdivision schemes are included in TopMod as remeshing operators. We organized rest of the remeshing operators (i.e. except dual) under two categories: (1) Conversion operators that change any given mesh into a mesh that consists 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. The system provides three triangulation [4]

including  $\sqrt{3}$ -subdivision [24], two quadrilateralization including Catmull-Clark [13] and one pentagonalization scheme [10] and their duals [14,8,31] including Doo-Sabin [15] and Simplest [32]. Note that there is no conversion scheme that can convert any given mesh to a mesh consisting of only polygons with more than five sides. The system included six preservation schemes [16,33,31,10] including Loop subdivision [25]. Note that hexagonal preservation operators exists since hexagonal meshes can represent shapes with a genus higher than 1. Providing such 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.

4. **High Genus Modeling:** The 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 [9,6]. 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 developed a set of new tools to automatically create very high genus meshes. The current automatic methods includes three methods (1) Generalized & Connected Sierpinski Polyhedra [35], (2) Wire Modeling [26] and (3) Column Modeling [27,37]. 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 to create a wide variety of high genus meshes by applying several combinations of remeshing schemes before wire or column modeling as shown in Figures 3 and 4.
5. **Others:** The system currently includes a couple of experimental tools which are still under development. Planar modeling tool helps to create general planar polyhedra and developable surfaces like a sculptor. Multihandle tool allows 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 tool is the creation of generalized Menger sponges. This extends the Menger sponge [28] to shapes beyond a cube, producing similar shapes starting from a wide variety of initial surfaces.

## 6 Interface Improvements

The original user interface of TopMod was quite tedious and restrictive, and did not allow users to discover and



Fig. 3 Wired caricature of Humphrey Bogart.



Fig. 4 Wired rabbit.

learn how to use different operations quickly or easily. Moreover, original interface did not provide complicated selection mechanisms, which can greatly simplify creation of shapes such as the ones shown in Figure 1. The new version of the ToMod provided effective solutions to these interface problems and helped to popularize high-genus modeling paradigm among novice users.

### 6.1 Interface Design

The new interface for TopMod features an intuitive system of iconography, organized menu items, a fully customizable interface with configurable shortcuts and style sheets that will allow the user to tailor their copy of the

application to his or her specific needs [30]. This has been accomplished through the use of various features of the Qt interface library. Every operation or function in the program is stored in the code as a Qt action. The actions are represented graphically in a variety of ways including toolbar buttons and menu items. This action driven design allows other TopMod developers to easily expand the functionality of the interface and expose more of the internal data structure to the user.

In the new version of TopMod the menus, submenus and toolbars can be detached from their default location and placed anywhere on the screen to allow for faster access to specific commands and tools. The style of the general user interface elements and the application window as a whole can be customized through the use of the TopMod stylesheet editor or a text editor window. We have also implemented an interactive shortcut customization feature which allows users to set custom shortcuts for any command available in TopMod.

The new TopMod provides a popup command line interface (CLI), similar to the command line and auto-completion functionality offered by Rhinoceros 3D, for accessing all the operations (from INSERTEDGE to SAVEFILEAS) by typing the command. The popup CLI can be invoked by hitting the space-bar and the command line input automatically completes the commands as the user types and displays a list of possible choices below the cursor based on the current input. This will essentially allow every command to be accessed in less than five keystrokes without touching the mouse.

## 6.2 Selection

The previous version of TopMod required the user to apply only one operation at a time, which required to apply same type of selection for each operation again and again. For example, the DELETEEDGE command requires a single edge as an input. Therefore, to delete multiple edges required the user to make one mouse click for each edge to be deleted. Similar to DELETEEDGE large number of TopMod operators require selection of one edge, one face, or one vertex. Thus, repetitive application of the same operator required a large number of mouse clicks which may take a long time to complete. In order to speed up the modeling process, we have also developed and implemented selection mechanisms that allow users to easily select multiple number of edges, faces, and vertices with a wide variety of ways; and we allowed the same operation to be applied to multiple sets of edges, faces and vertices. This approach speeded up the modeling process significantly. For instance, creating each shape shown in Figure 1 used to take at least 15 minutes. With the new approach, modeling time reduced to a few minutes.

In the new version, we have implemented standard selection mechanisms such as select multiple, select all,

select inverse, grow and shrink selection for edges, faces and vertices. For instance, the users can select all faces, or all edges, or all vertices, or all vertices and faces. They can grow a face selection and convert it to a vertex selection. In addition to these standard selection mechanisms we have developed some new selection mechanisms such as ring and loop selection, checkerboard selection and similarity selection. For instance, we define an edge loop as a line of edges in a quadrilateral mesh that are connected end to end; and clicking an edge in a select edge loop mode causes to select all the edges that are connected end to end to the clicked edge. All these selection mechanism are intuitive and therefore easy to learn with trial and error.

## 7 Scripting

In TopMod 1.0, most users could not program their own procedures. Even if the users have access to the source code, it was difficult for most of them to add their own procedures without our help. For those few users that do have the access and the ability to contribute to the TopMod source code, it is not practical to edit the code for every little task. It takes time to expose the new code to the graphical interface and also to compile the code. Adding several of these custom tasks would cause the code to become bloated very quickly.

This inability to write code significantly limited the users efficiency and creativity. The users could not perform repetitive or logical tasks efficiently for the 3D meshes that can be constructed/manipulated with a sequence of operations which can be condensed into a simple loop. Without having ability to code, a user used to perform each operation manually. Working this way was tedious especially when the user wanted to alter the operations parameters in each execution. Users could not also control the parameters of TopMod's operations with mathematical functions Cosine or Random. By using functions along with the standard arithmetic operators, modelers can use their creativity and mathematical knowledge to create interesting shapes.

Another problem was that the users did not have access to all minimal operators. While TopMod's graphical interface provides access to INSERTEDGE and DELETEEDGE, the other two minimal operators CREATEVERTEX and REMOVEVERTEX were not exposed to the users. Although, hiding access to these operations keeps the software interface simple, hiding these operations also limits the users' creativity.

The most commercial computer graphics software allow the users writing codes for repetitive tasks, using functions and accessing to low-level operations with a scripting language. In this way, although high-level operations can still be hidden from basic users who wants simpler interface, experienced or interested users can utilize the full potential of the software.

Some computer graphics packages such as Autodesk Maya or Autodesk 3D Studio Max have their own specific scripting language, Maya Embedded Language (MEL) [43] and MaxScript respectively. With the increase in quantity and power of existing scripting languages, more recent packages make use of an external scripting language. Scripting languages usually have much extensibility with which developers can create bindings to that package. Rhinoceros, a 3D modeling tool for Windows, uses VB-Script [23]. In Blender [12], users script with Python [34], an increasingly popular language for scripting. Python is also used in the computer graphics package Softimage XSI, and has been adopted recently in Maya 8.5, which gives Maya users a choice in their scripting language.

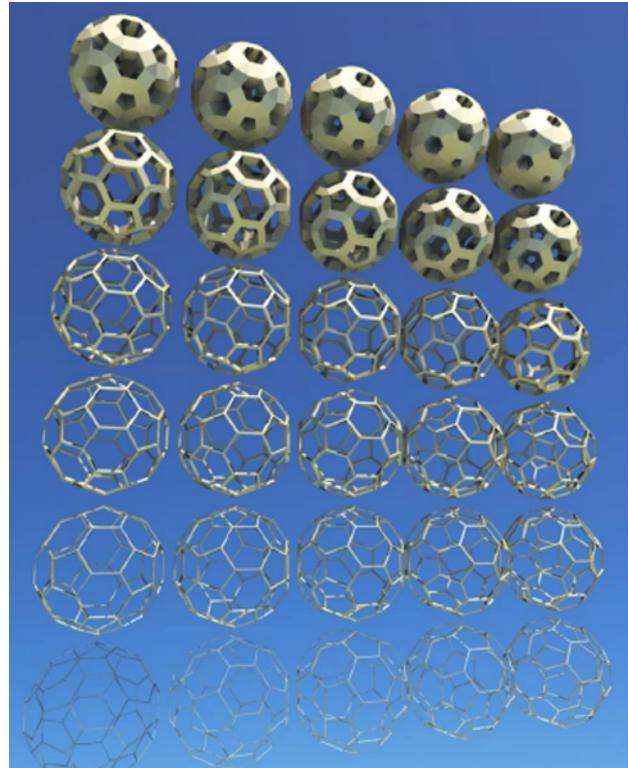


**Fig. 5** A tower shape created by controlling the distance and scale parameters of an extrude with the sine of the current time. Running the same code at a different time yielded different results.

Similar to Blender, we have developed a scripting engine [39] based on Python. Our scripting engine provides all core and auxiliary TopMod operations to users. It allows users to quickly and easily perform repetitive tasks as well as extending the functionality of TopMod without editing the source code. Figures 5 and 6 show shapes that are created using the script engine. Using an external scripting language like Python has many benefits. Users that are already familiar with Python would not have the hurdle of learning a new language. Users that are new to Python would be learning a popular language that they could use with many other programs.

In order to allow users to write scripts interactively within the TopMod interface, we have also developed a custom script editor interface in the new version of TopMod. The scripting interface is developed using Qt's in-

terface widgets, and made sure the code was consolidated into the final version.



**Fig. 6** The thirty versions of a wireframe soccerball. Each has a different thickness. The script also organizes them into this grid arrangement.

## 8 Conclusion and Future Work

This paper first time introduces TopMod as a complete modeling system. The functions and tools of *TopMod* can be utilized in a variety of design disciplines. In architecture, it can be used to enhance basic forms with high genus structures [6]. 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.

TopMod has already become a phenomenal success among virtual sculptors. 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 the system. Although topology change is considered a challenging mathematical concept, the users of our system can easily learn it and create complicated meshes.

## References

- Akleman, E., Chen, J.: Guaranteeing the 2-manifold property for meshes with doubly linked face list. *International Journal of Shape Modeling* **5**(2), 149–177 (1999)
- Akleman, E., Chen, J., Srinivasan, V.: A new paradigm for changing topology during subdivision modeling. In: *Proc. 8th Pacific Conference on Computer Graphics and Applications (PG'00)*, pp. 192–200 (2000)
- Akleman, E., Chen, J., Srinivasan, V.: A minimal and complete set of operators for the development of robust manifold mesh modelers. *Graphical Models* **65**(5), 286–304 (2003)
- Akleman, E., Edmundson, P., Ozener, O.: A vertex truncation subdivision scheme to create intriguing polyhedra. In: *Proceedings of Bridges: Mathematical Connections in Art, Music, and Science*, Winfield, Kansas, August, pp. 117–124 (2004)
- Akleman, E., Ozener, O.: Symmetric sculptures. In: *Visual Proceedings of ACM SIGGRAPH'2006 (Siggraph Art Exhibition Sketch)* (2006)
- Akleman, E., Ozener, O., Srinivasan, V.: Rind architecture. *International Journal of Architectural Computing (Selected from ECAADE 2003)* **3**(1), 93–106 (2004)
- Akleman, E., Ozener, O., Yuksel, C.: On a family of symmetric, manifold and high genus sculptures. In: *Proc. Bridges: Mathematical Connections in Art, Music, and Science*, pp. 145–150 (2006)
- Akleman, E., Srinivasan, V.: Honeycomb subdivision. In: *Proceedings of ISCIS'02, 17th International Symposium on Computer and Information Sciences*, vol. 17, pp. 137–141 (November 2002)
- Akleman, E., Srinivasan, V., Chen, J.: Interactive rind modeling. In: *Proceedings of Shape Modeling International 2003*, Seoul, Korea, May, pp. – (2003)
- Akleman, E., Srinivasan, V., Melek, Z., Edmundson, P.: Semi-regular pentagonal subdivisions. In: *Proceedings of Shape Modeling International 2004*, Genoa, Italy, June, pp. 110–118 (2004)
- Baumgart, B.J.: Winged-edge polyhedron representation. Master's thesis, Technical Report CS-320, Stanford University (1972)
- Blender: Official website. Internet: [www.blender.org](http://www.blender.org) (2007)
- Catmull, E., Clark, J.: Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer Aided Design* (10), 350–355 (1978)
- Claes, J., Beets, K., Reeth, F.V.: A corner-cutting scheme for hexagonal subdivision surfaces. In: *Proceedings of Shape Modeling International'2002*, Banff, Canada, pp. 13–17 (2002)
- Doo, D., Sabin, M.: Behavior of recursive subdivision surfaces near extraordinary points. *Computer Aided Design* (10), 356–360 (1978)
- Dyn, N., Levin, D., Simoons, J.: Face-value subdivision schemes on triangulations by repeated averaging. In: *Curve and Surface Fitting: Saint-Malo 2002*, pp. 129–138 (2002)
- Escher, M.C.: *The graphic works: introduced and explained by the artist (image plates 41 and 43)*. Barnes and Nobles Books, New York (1994)
- FLTK: The fast light toolkit home page. <http://www.fltk.org/> (2004)
- Fomenko, A.T., Kunii, T.L.: *Topological Modeling for Visualization*. Springer-Verlag, New York (1997)
- Guibas, L., Stolfi, J.: Primitives for the manipulation of general subdivisions and computation of voronoi diagrams. *ACM Transaction on Graphics* (4), 74–123 (1985)
- H. Ferguson, A.R., Cox, J.: Topological design of sculptured surfaces. In: *Proceedings of SIGGRAPH 1992, Computer Graphics Proceedings, Annual Conference Series*, pp. 149–156. ACM, ACM Press / ACM SIGGRAPH (1992)
- Havemann, S.: Generative modeling. Internet: [graphics.tu-bs.de/gml](http://graphics.tu-bs.de/gml) (2003)
- Kingsley-Hughes, A., Kingsley-Hughes, K., Read, D.: *VBScript Programmer's Reference*. Wiley (2007)
- Kobbelt, L.:  $\sqrt{3}$ -subdivision. In: *Proceedings of SIGGRAPH 2000, Computer Graphics Proceedings, Annual Conference Series*, pp. 103–112. ACM, ACM Press / ACM SIGGRAPH (2000)
- Loop, C.: Smooth subdivision surfaces based on triangles. Master's thesis, University of Utah (1987)
- Mandal, E., Akleman, E., Srinivasan, V.: Wire modeling. In: *Visual Proceedings of ACM SIGGRAPH'2003*, San Diego, California, July, pp. – (2003)
- Mandal, E., Akleman, E., Srinivasan, V.: Column modeling. In: *Visual Proceedings of ACM SIGGRAPH'2004 (Siggraph Technical Sketch)* (2004)
- Mandelbrot, B.: *The Fractal Geometry of Nature*. W. H. Freeman and Co., New York (1980)
- Mantyla, M.: *An Introduction to Solid Modeling*. Computer Science Press, Rockville, Ma. (1988)
- Morris, D.V.: New graphical user interface for a 3d topological mesh modeler. Master's thesis, Texas A&M University (2007)
- Oswald, P., Schröder, P.: Composite primal/dual  $\sqrt{3}$ -subdivision schemes. *Computer Aided Geometric Design, CAGD* **20**(3), 135–164 (2003)
- Peters, J., Reif, U.: The simplest subdivision scheme for smoothing polyhedra. *ACM Transactions on Graphics* **16**(4), 420–431 (1997)
- Prautzsch, H., Boehm, W.: Chapter: Box splines. *The Handbook of Computer Aided Geometric Design* (2000)
- Python: Official website. Internet: [www.python.org](http://www.python.org) (2007)
- Srinivasan, V., Akleman, E.: Connected and manifold sierpinsky polyhedra. In: *Proceedings of Solid Modeling 2004*, Genoa, Italy, June, pp. 261–266 (2004)
- Srinivasan, V., Akleman, E., Chen, J.: Interactive construction of multi-segment curved handles. In: *Proc. Pacific Graphics 2002*, pp. 429–435 (2002)
- Srinivasan, V., Mandal, E., Akleman, E.: Solidifying frames. In: *Proc. Bridges: Mathematical Connections in Art, Music, and Science*, pp. 203–210 (2005)
- Takahashi, S., Shinagawa, Y., Kunii, T.L.: A feature-based approach for smooth surfaces. In: *Proceedings of Fourth Symposium on Solid Modeling*, pp. 97–110 (1997)
- Tett, S.T.: A sculpting interface for doubly linked face list based polygonal meshes. Master's thesis, Texas A&M University (2007)
- TopMod: Official website. Internet: [www.topmod3d.org](http://www.topmod3d.org) (2007)
- Trolltech: Qt website. <http://www.trolltech.com/> (2004)
- Welch, W., Witkin, A.: Free-form shape design using triangulated surfaces. In: *Proceedings of SIGGRAPH 1994, Computer Graphics Proceedings, Annual Conference Series*, pp. 247–256. ACM, ACM Press / ACM SIGGRAPH (1994)
- Wilkins, M.R., Kazmier, C.: *MEL Scripting for Maya Animators*. Morgan Kaufmann (2005)
- Williams, R.: *The Geometrical Foundation of Natural Structures*. Dover Publications, Inc. (1972)
- Zorin, D., Schröder, P.: A unified framework for primal/dual quadrilateral subdivision schemes. *Computer Aided Geometric Design, CAGD* **18**(5), 429–454 (2002)