Rescalable Real-Time Interactive Computer Animations

John M. Sullivan

University of Illinois, Urbana, IL 61801, USA

Abstract. Animations are one of the best tools for communicating three-dimensional geometry, especially when it changes in time through a homotopy. For special-purpose animations, custom software is often necessary to achieve real-time performance. Luckily, in recent years, computer hardware has improved, and libraries have been standardized, to the point where such custom software can be easily ported across all common platforms, and the performance previously found only on high-end graphics workstations is available even on laptops.

Even the simplest surfaces in three dimensions are best explained with pictures or models; these might, for instance, explain the classification of regular polyhedra or that of smooth embedded surfaces by genus.

The illustrator's task becomes more complicated for self-intersecting surfaces (such as Boy's immersed projective plane), or triply periodic surfaces (including many interesting minimal surfaces). Here, the entire surface cannot be seen at once. George Francis has admirably described, in his Topological Picturebook [Fra], many ways to make drawings convey as much information as possible in such cases. However, neither static drawings nor three-dimensional models can easily show the complete structure of the surface.

Still more difficult is the visualization of homotopies, or deformations of surfaces through time. Here, the famous example of a sphere eversion (a smooth motion under which a sphere turns itself inside-out, see [Sul]) has been a benchmark of mathematical visualization for several decades.

For complicated surfaces or homotopies, animation becomes a good technique of illustration. As has been known for over a century, discrete images presented to our visual system in quick enough succession create the illusion of continuous motion.

The most practical way to create mathematical animations is by computer. Other kinds of animations often require complicated photorealistic renderings or articulated motions, but there is usually no need for these in mathematical visualization. Once a surface has been described to a computer and the first picture has been produced, it is often relatively easy to produce the whole animation, simply by adjusting a few parameters.

The first mathematical animations, like the famous rendition of a Morin-style sphere eversion by Nelson Max [Max], were produced on computers that could not even draw a simple wire-frame image in "real time". Images were collected and written to photographic film or, later, videotape.

In most contexts, real-time computations mean those which are done fast enough to keep up with some events in the real world. For a mathematical visualization, however, there is no sense of physical time. Thus, for us, "real-time" will describe rendering that is fast enough to present a convincing animation to the human eye. That is, each image should take no more than about 50ms to render, so that at least 20 frames per second can be presented to the viewer.

Of course, even with the vastly improved speed of today's computers, some renderings still cannot be done in real time. When producing a video, it is no problem to compute frames one at a time in order to get a special rendering effect. For instance, the transparent scenes (see Color Plate 1) in our recent sphere-eversion video "The Optiverse" [SFL] were rendered using a custom soap-film shader for Pixar's Renderman. Each frame took about 5 minutes to render, even on a fast modern workstation.

Precomputed animations, for instance in a video, have many advantages. The director of the video can carefully choose camera paths, rendering styles (including those impossible in real time), homotopy speeds, clipping planes, and so on, in order to convey particular information which can also be brought out in the narration.

For presentation in a lecture setting to a large audience, such prerendered videos are usually superior to interactive demonstrations. An interactive demo may fail to work properly, or the demonstrating lecturer may waste time finding the best views, or may forget to make the appropriate mathematical points.

But the audience members who see a mathematical video will often want to get their hands on the animation to explore further. For individuals (as opposed to large groups), an animation they can control is usually the best way to learn about a complicated surface or homotopy. Thus, the mathematics will be communicated best by supplementing a well-planned video with an interactive animation.

An interactive animation presents new frames to users in real time depending on the choices they have made. Some computer games work by choosing frames to display from a precomputed library on a DVD. But for mathematical animation, the choices the user might make are essentially infinite, and thus an interactive animation must use real-time rendering.

A decade ago, George Francis coined the term RTICA for Real-Time Interactive Computer Animation. He started producing such programs, illustrating various homotopies of projective planes [CFI,Fra], or the isotopy associated to the Dirac string trick from quantum mechanics [SFK$^+$,FK]. At the time, such RTICAs could be run only on high-end graphics workstations like those from Silicon Graphics. Therefore, they could not be disseminated widely, and were of limited value in communicating mathematics on a broad scale.

The continued recent progress in computer hardware and software, however, has changed this. Despite predictions of some theoretical limits, Moore's law about the exponential increase in processor speed over time has continued to hold. Thus a run-of-the-mill new CPU chip, with no special graphics hardware, can render images faster than a high-end graphics workstation from five or six years ago. And much faster graphics is now available simply by buying a relatively inexpensive 3D-graphics card.

Equally important are the advances in software which have been made possible by the open source movement. Whereas in the past, three-dimensional graphics had to be coded specially for each kind of hardware, the OpenGL library (thanks to being open) is now universal. It runs on all varieties of Unix workstations, on Intel hardware under Linux or Windows, or on Macintosh computers. It works both on machines with special purpose graphics hardware (which is now usually built to optimize performance of the OpenGL primitives) and on machines where all the rendering computations must be done by the CPU.

It is therefore appropriate to describe modern RTICAs written in OpenGL as rescalable. They may still work best on a high-end workstation, but they rescale to provide acceptable performance on almost any machine, even a laptop. With some care in the coding, these rescalable RTICAs can also work in immersive virtual-reality environments like the CAVE Automated Virtual Environment [CNSD+].

Most of the scenes in "The Optiverse" were rendered directly by our custom RTICA, called `avn`, which was written in C with OpenGL, and is available for various platforms on the CD-ROM accompanying this volume. (These scenes include views like the one shown in Color Plate 2.) In order to achieve real-time performance, transparency is not used. Instead other tricks enable the viewer to see the interior structure of the surface.

The minimax sphere eversion [FSK+] shown in "The Optiverse" is computed automatically in Brakke's `Surface Evolver` [Bra], making use of a triangulated approximation to the true smooth sphere. To emphasize that fact, we render the triangles with flat shading rather than using Gouraud shading or some other smooth technique. This also has the effect of leaving many sharp edges in the image; in the CAVE these edges produce a better stereoscopic effect. To make internal structure visible, we sometimes leave gaps around the edges of the triangles, or (as in Color Plate 2) in their centers.

The RTICA `avn`, although it does have some specific features for the sphere eversion, is actually a general purpose tool for animating homotopies. It reads a series of precomputed *topes*, or stages in the homotopy, and then displays them in sequence, letting the user control the motion forwards and backwards in the homotopy. Each tope is simply a triangulated surface in three-space, from which arbitrary views can be rendered, as the user also controls the camera motion.

Because the locus of self-intersection (or *double locus*) of the surface is important for understanding the stages in a sphere eversion, each of our precomputed topes includes information about its double locus, and `avn` can render a (smoothly shaded) tube around these curves.

"The Optiverse" uses a mixture of different rendering styles. These include transparency, solid surfaces, different kinds of gaps, and the tube around the double locus, as well as clipping boxes to zoom in on an area of particular interest. We have found that this variety is quite useful to show different aspects of the eversion and its internal structure. It is impressive that with today's computers, all but the soap-film-like transparent scenes can also be rendered in real time with an RTICA.

References

[Bra] Kenneth A. Brakke. The Surface Evolver. *Experimental Mathematics* **1**(1992).

[CFI] Donna Cox, George Francis, and Ray Idaszak. The Etruscan Venus. In T. DeFanti and M. Brown, editors, *Video Review Number 49*. ACM-SIGGRAPH, 1989.

[CNSD$^+$] Carolina Cruz-Neira, Daniel J. Sandin, Thomas A. DeFanti, R. V. Kenyon, and John C. Hart. The CAVE: Audio-visual experience automatic virtual environment. *Communications ACM* **35**(1992), 65–72.

[Fra] George Francis. *A Topological Picturebook*. Springer, New York, 1987.

[FSK$^+$] George Francis, John M. Sullivan, Rob B. Kusner, Ken A. Brakke, Chris Hartman, and Glenn Chappell. The minimax sphere eversion. In H.-C. Hege and K. Polthier, editors, *Visualization and Mathematics*, pages 3–20. Springer Verlag, Heidelberg, 1997.

[Fra] George K. Francis. The Etruscan Venus. In P. Concus, R. Finn, and D. Hoffman, editors, *Geometric Analysis and Computer Graphics*, pages 67–77. Springer Verlag, New York, 1991.

[FK] George K. Francis and Louis H. Kauffman. Air on the Dirac Strings. In W. Abikoff, J. Birman, and K. Kuiken, editors, *The Mathematical Legacy of Wilhelm Magnus*, volume 169, pages 261–276. Amer. Math. Soc., Providence, RI, 1994.

[Max] Nelson L. Max. *Turning a Sphere Inside Out*. International Film Bureau, Chicago, 1977. Narrated videotape (21 min).

[SFK$^+$] Daniel Sandin, George Francis, Louis Kauffman, Chris Hartman, and Glenn Chappell. Air on the Dirac Strings. In T. DeFanti and M. Brown, editors, *Electronic Theater*. ACM-SIGGRAPH, 1993.

[Sul] John M. Sullivan. "The Optiverse" and other sphere eversions. In R. Sarhangi, editor, *Bridges 1999*, pages 265–274, Winfield, Kansas, 1999. Bridges Conference, Southwestern College. ArXiv eprint math.GT/9905020.

[SFL] John M. Sullivan, George Francis, and Stuart Levy. The Optiverse. In H.-C. Hege and K. Polthier, editors, *VideoMath Festival at ICM'98*, page 16 plus 7–minute video, Heidelberg, 1998. Springer Verlag. See http://new.math.uiuc.edu/optiverse/.