Title: Artificial Evolution of Implicit Surfaces
Category: Technical

Primary Author: Edward J. Bedwell
Affiliation: University of Maryland Baltimore County
Department of Computer Science and Electrical Engineering
Address: UMBC - CSEE Department
1000 Hilltop Circle
Baltimore, MD 21250
Phone: (410) 455-6339
Fax: (410) 455-3969
E-mail: ebedwe@csee.umbc.edu

Second Author: David S. Ebert
Affiliation: University of Maryland Baltimore County
Department of Computer Science and Electrical Engineering
Address: UMBC - CSEE Department
1000 Hilltop Circle
Baltimore, MD 21250
Phone: (410) 455-3541
Fax: (410) 455-3969
E-mail: ebert@csee.umbc.edu

Note: The quality of the images has been compromised by the conversion to PDF. Photo ready images can be supplied. The original images can be found at http://www.csee.umbc.edu/~ebedwe/s98`

Summary: Through the application of genetic programming, the complexity of implicit surface generation can be significantly reduced if not done away with altogether. This sketch provides an overview of the techniques we employed to achieve this goal.

Artificial Evolution of Implicit Surfaces

Edward J. Bedwell and David S. Ebert
Department of Computer Science and Electrical Engineering
University of Maryland Baltimore County

Procedural techniques allow algorithmic and mathematical methods to generate complexity to a nearly infinite level of detail. By giving the user parametric control of procedural methods, extremely complex models can be created in a fraction of the time it would require to build them by hand. This methodology has become widely accepted throughout the computer graphics community, as evidenced by the work of Lindenmayer and Prusinkiewicz in procedurally simulated plant growth **[PRU88]**, and the vast work that has been done in the realm of implicit surfaces **[HAR93][BLO97][etc.]**.

However, procedural techniques introduce new levels of difficulty. Creating these procedural tools often requires a strong understanding of complicated mathematics. Furthermore, even if one does have the mathematical knowledge necessary, providing parametric controls to one who does not is a non-trivial task. Karl Sims presented genetic programming as a viable solution to this problem with his procedural texture generation system **[SIM91]**. Genetic programming has also been proven useful to computer graphics. Richard Dawkins made use of the genetic metaphor to generate 2D branching structures **[DAW86]**. Gritz and Hahn applied genetic programming techniques to generate various procedural animation methods **[GRI95]**.

This sketch presents a system that combines implicit surfaces, as modeling primitives, with genetic programming to facilitate automated generation of exceedingly complex models. The class of implicit surfaces this system utilizes is characterized by a function $\mathbf{f} : \mathbf{R}^3\text{-}\mathbf{R}$ that assigns some value to each point in three-space. The surface is the set of points such that $\mathbf{f}(x,y,z) = 0$ **[HAR93]**. For initial testing, five of these surfaces form our primitive community. This community is the population from which the system allows the user to select the initial individuals to be mated. The mating process then generates offspring consisting of new surfaces. From this new generation the user can again select two individuals to mate, creating the next generation. This process is repeated until a surface is generated that fits the user's needs.

Implementation

To mate two surfaces, the user begins by selecting two parent individuals from the primitive community that exhibit traits wanted by the user. The functional representations are manipulated as LISP expressions. A series of LISP routines generates hierarchical parse trees from each expression. Mating is facilitated by a "cross-over" procedure in which subtrees are pseudo-randomly selected from each parent and swapped. Thus, two children are "born" to the user selected parents. This process is repeated until the generation reaches a suitable size. In the initial implementation this size is set to ten.

The newly generated expressions are converted to infix notation and embedded in ASCII scene description files. These scenes are then rendered using a modified version of POV-Ray, a publicly available ray-tracing package.

Results and Application

In biological systems, offspring created through the mating process will exhibit traits from both parents. This holds true in the artificial mating of implicit surfaces. The system has shown that even several generations will maintain traits from the original parents. Given a sufficiently large and diverse population of surfaces from which to begin, the user should be able to create a surface that meets their needs.

The system does not restrict mating simply from within a single generation. Individuals can be selected from different generations, or even different family trees. If an individual were to be generated that was close to one's target, another individual that exhibits the lacking trait could be mated with it to generate adequate progeny.

Conclusion & Future Work

This system successfully demonstrates that genetic programming greatly simplifies the generation of complex implicit surfaces. Furthermore, the genetic metaphor allows for the transmission of desirable traits from each parent to the child. Thus the user maintains some control of the evolutionary process. Future work includes an extended primitive set, differing probabilities for crossover points in the parse trees, mutation during the mating process, and blending multiple implicit surfaces through genetically determined blending functions.

Acknowledgments

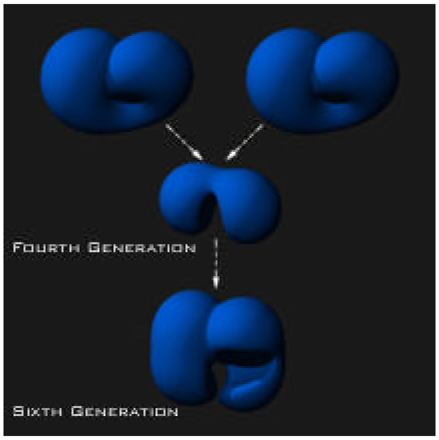
Ryoichi Suzuki, of the Electrotechnical Laboratory in Japan, for providing the isosurface modifications to the Persistece of Vision raytracer.

Andrew Guy and Brian Wyvill, of the University of Calgary in Canada, for making their software available.

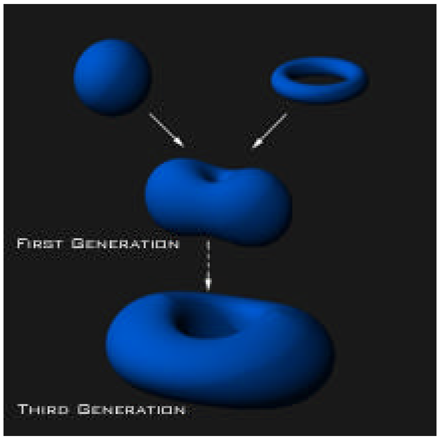
References

[BLO97] - J. Bloomenthal, C. Bajaj, J. Blinn, M. Cani-Gascuel, A. Rockwood, B. Wyvill, G. Wyvill, Introduction to Implicit Surfaces, Morgan-Kaufmann, 1997.
[DAW87] - R. Dawkins, The Blind Watchmaker, Harlow Logman 1986.
[GRI95] - L. Gritz, J. K. Hahn, "Genetic Programming for Articulated Figure Motion", The Journal of Visualization and Computer Animation, Vol.6, 1995, pp. 129-142.
[HAR93] - J. C. Hart, "Ray Tracing Implicit Surfaces", WSU Technical Report EECS-93-014, 1993.
[PRU88] - P. Prusinkiewicz, A. Lindenmayer, and J. Hanan. "Developmental Models of Herbaceous Plants for Computer Imagery Purposes", Computer Graphics, Vol.22 No.4, 1988, pp.141-150.
[SIM91] - K. Sims, "Artificial evolution for computer graphics", ACM Computer Graphics, 25(4), July 1991, pp. 319-328.

Visual Material



Two Klein bottles are mated. This image shows example offspring from the fourth and sixth generations of that mating.



A sphere and a torus are mated. This image shows a direct child of the mating, as well as one from the third generation

These three surfaces are examples of highly complex children generated from relatively simple parents. All of these surfaces are direct descendants of the sphere and torus