Software Testbed for Selective Laser Sintering

Richard H. Crawford, Assistant Professor
Suman Das, Research Assistant
J.J. Beaman, Professor
Department of Mechanical Engineering
The University of Texas at Austin

Abstract

Computer software plays an important role in the implementation of Solid Freeform Fabrication (SFF) technologies. This paper describes a software testbed for processing part geometry for a particular SFF technology, selective laser sintering (SLS), that is built around the separation of the slicing and rasterization operations to accommodate geometric information from a variety of sources. The paper also discusses the process control software being developed for a new high-temperature workstation for SLS of metal powders. This program features a high-resolution data file format, the ability to interpolate to achieve a desired resolution, and a menu-driven user interface with graphical feedback and process simulation capabilities.

Introduction

Solid Freeform Fabrication (SFF) technologies offer rapid prototyping capability by avoiding part-specific tooling and process planning. Computer software plays an important role in the implementation of these technologies. In this paper the characteristics of software for SFF will be described for a particular technology, selective laser sintering (SLS) [2, 4].

Producing parts by SFF requires the transformation of the geometric description of a part into a form suitable for processing by the particular technology. For SLS, this generally requires slicing the part geometry into layers, then rasterizing each layer to produce laser toggle points. By separating these two operations, geometric information from a variety of sources can be accommodated. In particular, geometric descriptions that do not require slicing, such as digitized data or CAT scans, can be processed directly by the rasterizing software. Separation of the operations allows research to improve either of the operations to be performed independently. The first part of the paper describes a software testbed for SLS that is built around this separation of slicing and rasterization operations.

Process control software for SLS interprets the toggle point data file and generates control signals for powder leveling and laser control. The second part of this paper describes the process control software being developed for a new high-temperature workstation for SLS of metal powders. The features of this software include a high-resolution binary toggle point file format, the ability to interpolate the data file to achieve the desired resolution, and a menu-driven user interface with graphical feedback and process simulation capabilities.

Geometry Processing for SLS

Slicing. The SLS process produces parts on a layer-by-layer basis. Geometric processing proceeds by first slicing the geometric description of the part into layers representing the sintering planes, typically at 0.005-0.010 in intervals (see Figure 1). The slicing operation generates the curves, or contours, that represent the boundaries of the part for each layer (see Figure 2). For the SLS testbed, slicing consists of computing the intersection of the geometry of the part with a series of planes oriented with normals in the positive *z* direction. The intersection calculation method depends upon the form of the input geometry; specific examples will be discussed below. The output, however, is independent of the input geometric form. A standard text file format has been established to provide contour information for subsequent laser toggle point generation. Currently, only polygonal contours are supported by the format (as well as the downstream processor). The format provides information on the part boundary at each layer in terms of a series of loops, and uses keywords to indicate the beginning and end of loops and contours. Each loop is described by listing the vertices that comprise the loop, ordered according to the right-hand rule (CCW for outer loops, CW for inner loops), as indicated in Figure 2.

Toggle point generation. The process of toggle point generation for each layer is akin to rasterization in computer graphics image generation [5], but more correctly might be called boundary discretization, since the goal for SLS is to generate discrete points on the boundary contours at which the energy beam must be toggled on or off, rather than shades for each pixel between such points. The process consists of intersecting mathematical rays directed along each scan line in the contour plane with the each segment of the boundary contours within each layer. These intersections are then ordered according to increasing distance from a given datum (the *x*-*z* plane for SLS). The algorithm used accounts for the special cases of vertices and boundary segments that lie entirely on a given scan line. Output from the program is a binary file of toggle points. The current format allows 8-bit scaled integers for each of the *x* and *y* coordinates of a toggle point, which supports resolution to 0.020 inches for a maximum size part.

Sources of geometric data. The motivation for dividing geometric processing for SLS into two distinct operations is to diversify the possible sources of geometric description while allowing reuse of common software where possible, and modular development of geometry-specific algorithms where necessary. As stated above, intersection calculations for slicing depend on the form of the geometry. Some sources of geometry require no slicing at all, but only reformatting into the standard contour representation described above.

The current state-of-the-art for describing geometry for most SFF technologies consists of tessellating the surfaces of the geometric model into a mesh of non-overlapping triangular facets. The resulting geometry is transmitted in a standard file format, the so-called STL file format, established by 3D Systems, Inc. [1] This format has been adopted by many CAD vendors. The slicing algorithm for STL geometries used in the SLS testbed is based on building a rich topological data structure that removes redundant data present in the input file and explicitly represents facet adjacency through a face-edge-vertex structure. The algorithm begins by determining the intersection of the current slicing plane with a given triangular facet. The most common case will result in an intersection with two of the facet's edges and generate one segment of the contour. The next facet to intersect is then chosen based on the explicit adjacency information in the geometric data structure. The contour edge generated by this facet is known to connect to the previous contour edge by virtue of the adjacency of their respective facets. This approach greatly simplifies the process of building the final contours, since the contours are built incrementally as the intersections are computed. The degenerate cases of facet-in-plane, edge-in-plane, and vertex-in-plane are handled separately, and require slightly more complicated decisions. The final step for each layer is determining the correct orientation for each contour. This operation currently is based on in-plane ray-tracing to determine which contours are

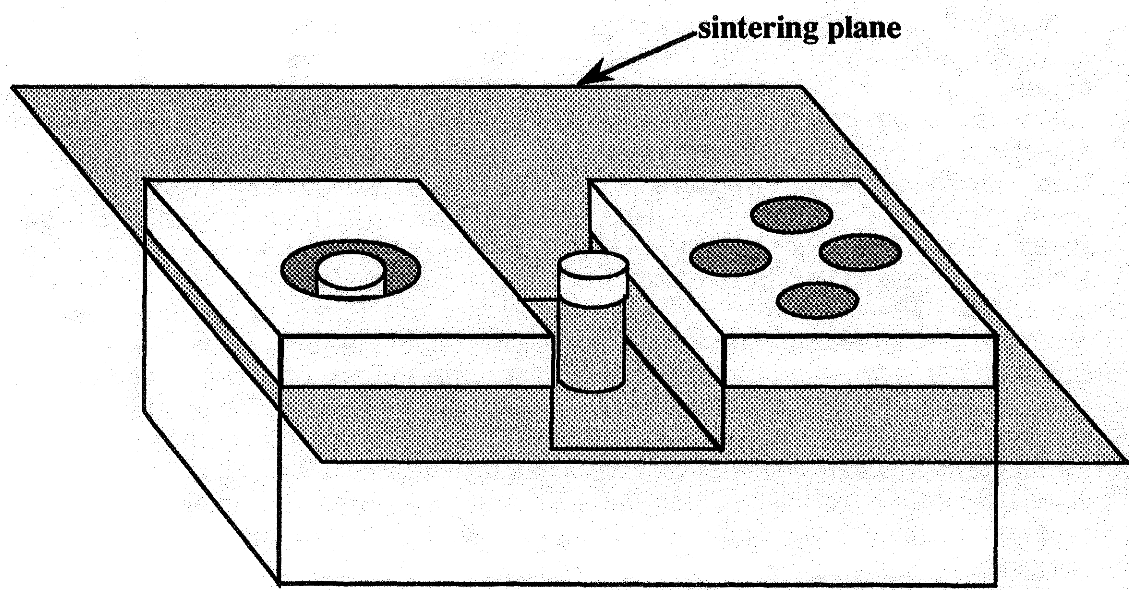

Figure 1. Slicing operation for a typical sintering plane.

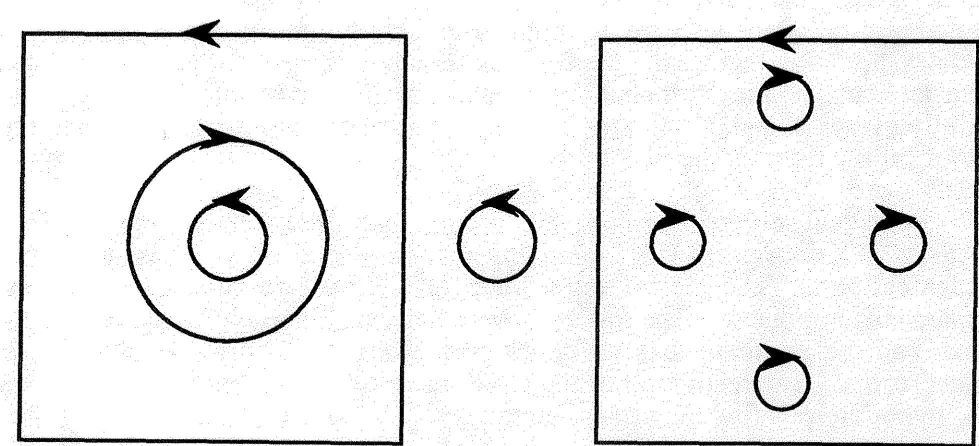

Figure 2. Oriented contours from slicing operation of Figure 1.

contained by others. Orientation is then alternated between CCW and CW, with the outermost contours being oriented CCW.

Another common representation for mechanical parts is constructive solid geometry (CSG). In this representation the geometry of a part is modeled as a binary tree whose leaf nodes are scaled and oriented instances of primitive shapes (e.g., spheres, cylinders, parallelepipeds, etc.) and whose non-terminal nodes are the regularized Boolean set operations union, intersection, and difference. Evaluating the CSG tree with a geometric modeler results in an explicit boundary representation (B-rep) of the part's surfaces. The B-rep can then be sliced to produce contours for the SLS process. In the SLS software testbed, TWIN, an experimental solid modeling package [7], is being used both to generate B-rep's from parts described as CSG trees, and to perform the slicing operation as well. Unlike most commercially available geometric modeling packages, which are designed to run as stand-alone systems, TWIN is implemented as a library of C subroutines and was designed to be imbedded in other applications. Slicing in this case is based on a TWIN function that sections a solid object. The algorithm consists of first building the B-rep of the part, then successively sectioning the part with planes corresponding to the SLS sintering layers. The top face from the resulting sectioned object in each step is the contour for that layer. The final step then consists of translating the TWIN face description into the standard contour format. Since the sectioning operation is imbedded in the slicing program, and does not require interfacing to a stand-alone program, the execution time is reasonable.

Geometry for some applications is naturally represented as contours. By providing the facility to process contour data directly, the problem of developing a surface model to be sliced and subsequently rasterized is avoided. The SLS project has worked with contour data from two sources: a laser digitizer and CAT scans of mechanical parts.

The first application is the design and production of prosthetic sockets at the University of Texas Health Science Center at San Antonio [9]. In this process, a residual limb is first digitized with a laser scanner, typically in layers of 0.125 in. The resulting scanned geometry then serves as input to a CAD program, which allows input of wall thickness, a scaling factor, and the addition of an attachment fitting to complete the design of the socket. The program also interpolates the digitized contours to produce the higher resolution (typically 0.005 in) required for SLS. The output from this program is in the contour format used by the SLS testbed.

A second source of contour data is computer-aided tomography (CAT). SMS, Inc. of Austin, TX fabricates CAT scanning machines and provides scanning services for mechanical parts. The typical application for these services is flaw detection in critical mechanical components. The SMS software has the ability to provide output data in many forms, and the company has written a post-processor to provide contours in the SLS testbed format. The resolution of the CAT scanning equipment is below that required for SLS, so no interpolation between contours is required for this application.

Research directions. The SLS project at UT is actively pursuing research in several directions to expand its geometric processing capability. One goal is to develop more accurate slicing and rasterization algorithms for CSG-based geometries. The algorithm described above uses a polygonal representation of the part, since TWIN is a polygonal modeler. Also, the sectioning operation in TWIN always produces a valid solid object. This operation can be optimized for SLS slicing purposes, since the desired output is a single face, not a complete solid. One approach being investigated involves slicing the primitives in a CSG individually, then performing Boolean operations on the slices. If the CSG primitives are limited to quadrics, then exact intersections can be generated for the contours, rather than polygonal approximations. Since the resulting contours will consist of quadratic curves, the rasterization software must also be suitably modified.

Another area of research is the investigation of rational bicubic parametric surfaces as the primary geometric description for SLS. The slicing operation for such a description consists of intersecting a rational bicubic surface with a slicing plane, resulting in a set of polynomial curves. This operation amounts to solving the inversion problem, *i.e.*, determining the surface parameter values that correspond to a particular value of z. While it is theoretically possible to solve this intersection problem in closed form, we are focusing on developing algorithms that approximate this intersection as rational cubic spline curves by using a combination of analytical techniques from elimination theory as well as numerical techniques [8]. Development of a toggle point generation algorithm involves the generalization of the scan conversion algorithm to non-linear contours. The algorithm must account for multiple intersections of contour curves, as well as intersections at inflection points. For both the slicing operation and toggle point generation, measures for the approximation errors must be developed to control the accuracy of the SLS process.

Process Control Software

Current facilities for SLS research at UT include a laser sintering workstation equipped with a 100W Nd:YAG laser and a 25W CO_2 laser. This workstation [6], known as Bambi, has been used to perform SLS of polymers, ceramic-binder mixtures, polymer-coated metal and ceramic powders, as well as initial research with metal powder mixtures . To further the research in SLS of metal powders, a new High Temperature Workstation (HTW) equipped with a 1.1kW CO_2 laser is being constructed [3]. Improvements and additional features incorporated in this workstation require that the process control software be redesigned. This section of the paper compares the existing control program in Bambi with selected improvements and features that will be incorporated in the control software for the new workstation.

Platform. The process control computer used in Bambi is an IBM PS-2 running MS-DOS. A Sun IPC workstation will be used on the HTW. This choice combines the advanced software development environments of Unix and the X11 window system, multitasking capabilities, and widely accepted networking and file transfer protocols. These capabilities will allow real-time fabrication of parts as geometry files are sent over the network from remote sites and the potential to operate the HTW from a remote terminal over the network.

Input data processing. The process control software used in Bambi is limited to interpreting toggle-point data in 8-bit format, with the default spacing between successive scan-lines being 0.020 in. Numerical toggle-point data on the HTW will have as its default a 16-bit format that provides a higher resolution. The process control software will possess the capabilities of resolution selection independent of model size, interpolation of low resolution files to higher resolution, local transformation of data, and backward compatibility. A scan of the data file will inform the user whether it is possible to achieve the desired resolution between successive scan-lines. If not, the data file will be interpolated to obtain the desired resolution. Toggle-point files generated from contours will incorporate the option of vector scanning a layer's contours. Vector scanning promises improved part quality, integrity, and surface finish. Also, given a contour file, it will be possible to create cut-away sections of a part.

Process Control System. The process control system on Bambi uses a separate microprocessor for powder delivery and leveling control, a General Scanning DG series scan head controller, and a separate on-off temperature controller. The process control program communicates directly only with the scan head controller. The increased sophistication of the HTW and the advanced nature of the research require a more complex array of control and data acquisition devices, such as a CNC laser scanner controller, a data logger, a gas analyzer, and a safety interlock. These devices provide significant specialized functionality. The process controller in the HTW will act as the system integrator by handling all communication with the various control and measurement devices. This design centralizes the feedback from these devices and allows user control over them through the menu-driven interface described next.

Menu Driven Interface. The process controller in Bambi has a command-line type of user interface with a fixed sequence of commands that requires some information to be entered manually from tables and manual downloading of leveler control software, and has little feedback from the process. The process control software in the HTW will incorporate a menu-driven user interface, based on X11. This will allow the user to set process parameters such as chamber temperature, sintering layer thickness, scan delay between successive layers, scan speed, part scaling, and offsets for making multiple parts simultaneously. An interrupt-restart facility will be incorporated to allow changes to process parameters during the fabrication of a given part. The interface will also provide the user with a graphical feedback of process status and simulation of the SLS process on screen. Simulation of the SLS process will provide a test environment for other software development (*e.g.*, geometric processing, scanning algorithms, etc.) without the overhead and expense of actually fabricating test parts.

Conclusion

Software for processing geometric data and controlling part fabrication is an important factor in the development of Solid Freeform Fabrication techniques. This paper describes ongoing research at the University of Texas that focuses on software development to support Selective Laser Sintering. As SLS matures and the research focus moves toward developing the ability to fabricate functional mechanical parts, the demands on the software will grow. Future work will focus on expanding the variety of sources for geometric data, inclusion of more sophisticated control algorithms, the development of mathematical models of tolerance and surface finish in terms of process parameters, and developing geometric reasoning capabilities for orienting parts for fabrication based on final part strength, geometric accuracy, and process efficiency.

References

1. 3D Systems, Inc., "Stereolithography Interface Specification", Valencia, CA, June 1988.

2. Ashley, S., "Rapid Prototyping Systems", *Mechanical Engineering*, Vol. 113, no. 4, 1991, pp. 34-43.

3. Das, S., McWilliams, J., Wu, B., and Beaman, J. J., "Design of a High Temperature Workstation for the Selective Laser Sintering Process", Solid Freeform Fabrication Symposium, The University of Texas at Austin, Aug 12-14, 1991.

4. Deckard, C. R. and Beaman, J. J., "Recent Advances in Selective Laser Sintering", *Proceedings of the Fourteenth Conference on Production Research and Technology*, sponsored by the National Science Foundation, Ann Arbor, MI, October 1987, pp. 447-452.

5. Foley, J. D., van Dam, A., Feiner, S. K., and Hughes, J. F., *Computer Graphics Principles and Practice*, Addision-Wesley, Reading, MA, 2nd edition, 1990, pp. 92-99.

6. Forderhase, P.F., "Design of a Selective Laser Sintering Machine Intended for Academic Research", M.S. Thesis, The University of Texas at Austin, May 1989.

7. Mashburn, T. A., *A Polygonal Solid Modeling Package*, M.S. Thesis, School of Mechanical Engineering, Purdue University, December 1987.

8. Waggenspack, W. N. and Anderson, D. C., "Piecewise Parametric Approximations for Algebraic Curves", *Computer-Aided Geometric Design*, Vol. 6, 1989, pp. 33-53.

9. Walsh, N. E., Lancaster, J. L., Faulkner, V. W., and Rogers, W.E., "A Computerized System to Manufacture Prostheses for Amputees in Developing Countries", *Journal of Prosthetics and Orthotics*, Vol. 1, no. 3, April 1989.