

Variable thickness ruled edge slice generation and three-dimensional graphical error visualization

Peter B. Chamberlain, Mark D. Van Roosendaal, Charles L. Thomas
University of Utah

Abstract

This paper describes a simple variable thickness ruled edged slicing algorithm that produces slices with zero integrated error. In its present state, the algorithm requires pre-sampled data to be taken from the STL file. Data points are extracted in a cylindrical coordinate system giving values of radius at regular intervals $\Delta\theta$ and Δz . Using this data, the algorithm creates slices based on averaging of the data points and the slope of line segments connecting them. Error based constraints are used to determine slice thickness. A three-dimensional visualization technique using color contour plots (representing error) on the surface of the prototyped model provides a means of evaluating the accuracy of the prototyped part.

Introduction

Most of the rapid prototyping systems currently available use vertical (zero order) edge layers to build parts (de Jager, 1997). Vertical edged layers result in parts with stair-stepping surfaces, see Figure 1. Vertical edge rapid prototyping systems use relatively thin layers to improve surface accuracy. Very large parts, on the order of feet or meters, become difficult and expensive to build from thin layers. Thick layers enable the building of very large parts without expensive increases in time. Higher order edge types should be used for thick layers to maintain as much accuracy as possible. Figure 2 shows a ruled (first order) edge, which is the simplest of the higher order edge types. Note that the surface quality of the ruled layer is much better than the quality of the vertical layer.

Several research groups from around the world are working on large-scale rapid prototyping with ruled edges. The groups have developed several different types of ruled edge definitions. Three basic types of edges will be discussed in this paper. Hope discussed two of the basic types (Hope, 1997b), which are derived from the vertical edge types discussed by Dutta (Dutta, 1996). He labels the two types tolerance all outside and tolerance all inside. The two edge types will be referred to as outside and inside ruled edges respectively. Outside ruled edges, lines 3 and 6 in Figure 3, completely contain the original volume of the model and can be sanded down to get a better approximation. Inside ruled edges, lines 1 and 4 in Figure 3, fit entirely within the original surface so that a prototyped part could physically fit inside the original part. Material could be added to the prototyped part to

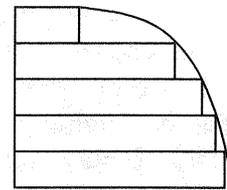


Figure 1
Stair-stepping effect of vertical edges

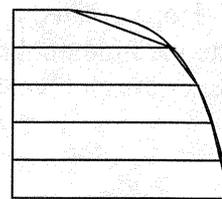


Figure 2
Improved surface quality of ruled edges

make it identical to the original part. A third type of ruled edge definition is the zero integrated error ruled edge slice (ZIGER slice) definition, lines 2 and 5 in Figure 3. A ZIGER surface represents the surface where positive and negative surface deviations sum to zero producing a surface that is both inside and outside the original part.

Shape Maker II, an RP system developed by Lee and Thomas at the University of Utah, uses a very simple algorithm to create ruled edge slices (Lee, 1997). Surface contours are extracted from an STL model at regular intervals of height. Layers are then defined by connecting successive contours with ruled edges. This method creates both outside and inside edges represented by lines 1 and 6 in Figure 3. TrueSurf, an RP system developed by Hope at the University of Queensland (Hope, 1997a), uses a different slice generation algorithm. Slices of regular thickness are defined by a line tangent to the slice cross section at the slice center, represented by lines 3 and 4. As with Shape Maker II, TruSurf creates outside and insided ruled edges. Some preliminary work on ZIGER edges has been conducted at the University of Utah using wavelets and quasi-wavelets (Lee, 1997). An important characteristic of the different slice types is edge matching. Connecting interpolated loops (Lee and Thomas) leads to layer edges that match from layer to layer. Slices defined by tangents and ZIGER edges will not meet unless slice boundaries are chosen so that the edges line up. That requires an adaptive thickness algorithm.

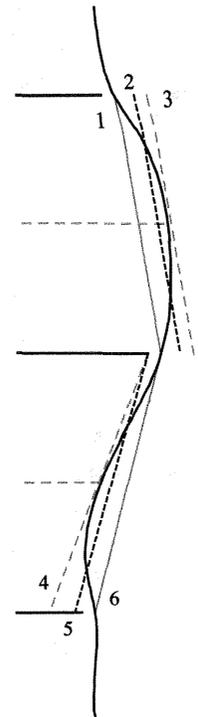


Figure 3
Ruled edge types

Adaptive slicing provides a means of improving the surface quality of large-scale rapid prototyping. With adaptive layer thickness it is possible to have thin layers where surface complexity is high and thick layers where there is little surface complexity. Most of the current adaptive thickness ruled edge algorithms are error based. Work by Dutta (REFERENCE) with vertical edges uses the concept of cusp height, shown as δ_D in Figure 4a. de Jager (de Jager, 1997) extended the concept of cusp height to ruled edges, shown as δ_J in Figure 4b. Hope created a dual cusp height, shown as ϵ and δ_H in Figure 4c (Hope, 1997b). In all cases the variable thickness algorithm works by performing a systematic search to maximize the thickness of each layer without exceeding a threshold cusp height value.

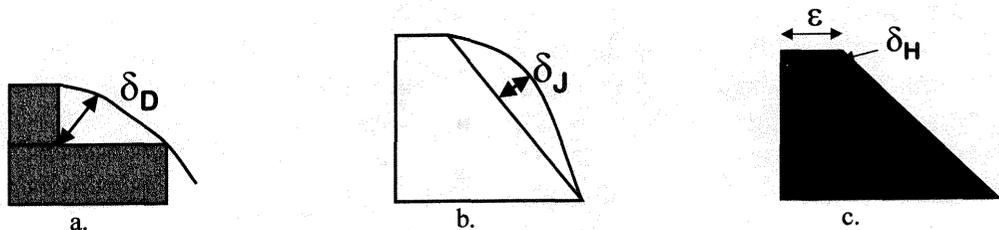


Figure 4
Variable thickness algorithm error measures

ZIGER Averaging Slicer

The focus of this paper is a slicing algorithm for the creation of ZIGER slices. The basic concepts and mathematics behind the slicer will be introduced here. For the initial discussion, assume that the geometry to be constructed is defined by a surface of revolution. Figure 5 shows a schematic diagram of how the slicer would work with such a surface. A ZIGER slice surface can be defined as a slice slope passing through a slice radius at the slice midline. The slice radius is simply the average value radius between the slice boundaries. The slice slope is the average value of slope between the slice boundaries. This procedure creates a ZIGER surface because the integral of the radial errors (using both positive and negative error) along the surface is zero. General equations for the ZIGER ruled slice are as follows:

$$r = \frac{\int_{z_0}^{z_1} r(z) dz}{z_1 - z_0} \quad \text{Equation 1}$$

$$s = \frac{\int_{z_0}^{z_1} \frac{dr(z)}{dz} dz}{z_1 - z_2} \quad \text{Equation 2}$$

where r is the slice radius, $r(z)$ is the radius at a given height z , and s is the slice slope. It is possible to generalize the method beyond the surface of revolution. Later in this paper one such method is tested as a first step toward developing a truly useful ZIGER slicing algorithm.

Adaptive Thickness Error Definition

The ZIGER averaging slicer lends itself to a new and sophisticated definition of slice error. The error definitions described in the introduction considered only one or two measures of error at points on the slice surface. It would be more desirable to have some measure of error representing the entire surface. One such measure will be proposed here. Once again, consider the special case of a surface of revolution. Figure 6 shows how the concept of error can be defined using the ZIGER slice. In the figure the original surface is shown as a gray line and the reconstructed surface is shown as a black line. Error is defined as the radius of the original surface subtracted from the radius of the original surface. Note that if the errors are integrated along the height of the slice, the result

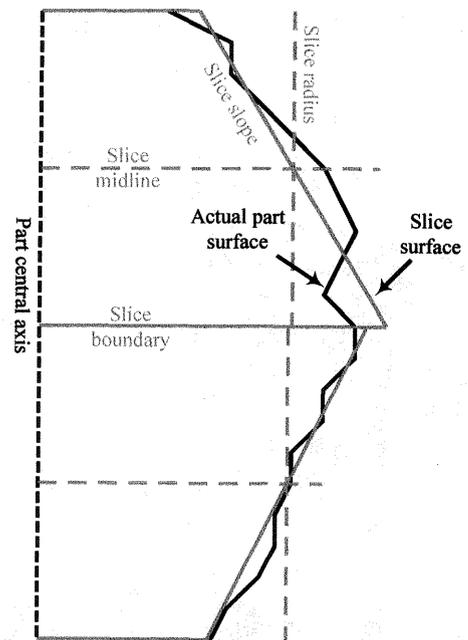


Figure 5
Concept of radial slicer applied to a surface of revolution

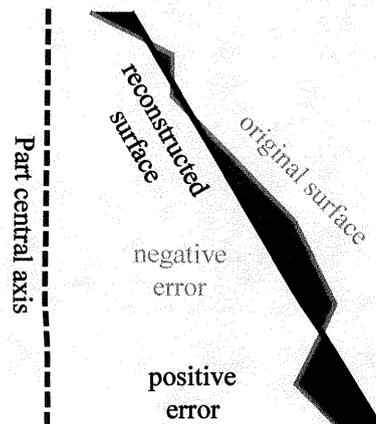


Figure 6
Error of ZIGER ruled surface

is zero(positive and negative error cancel), see Eq. 3. A useful definition of error is the average absolute error, \bar{E}_{abs} , which provides a meaningful measure of error over the entire surface of the slice, see Eq. 4. Another useful indication of error is the maximum absolute error, E_{abs}^{max} , that occurs anywhere within the slice, see Eq. 5.

$$ZIGE = \int_{z_0}^{z_1} (r_{re}(z) - r_o(z)) dz = 0 \quad \text{Equation 3}$$

$$\bar{E}_{abs} = \frac{\int_{z_0}^{z_1} |r_{re}(z) - r_o(z)| dz}{z_1 - z_0} \quad \text{Equation 4}$$

$$E_{abs}^{max} = \max(|r_{re}(z) - r_o(z)|) \quad \text{Equation 5}$$

The adaptive ZIGER edge slicer works like any other adaptive slicer. In this case the slicer will attempt to maximize the thickness of each layer without exceeding the specified average absolute error and maximum absolute error thresholds.

An Adaptive ZIGER Slicer

It is always useful to begin with a simplified model to test a new algorithm. For this paper a simple algorithm for adaptive slicing of any general single valued surface will be described. The algorithm contains three distinct components, a pre-sampler, a ZIGER slicer, and an adaptive thickness algorithm.

Pre-sampling

A surface of revolution provides a simple implementation of the algorithm, but a surface of revolution is not very interesting. A slightly more interesting model can be defined as a single valued surface in a cylindrical coordinate system. It is possible to use a coordinate transform to convert the triangle data of a single surface STL file from Cartesian to cylindrical coordinates. A further useful simplification is supply the algorithm with values of radius sampled at regular

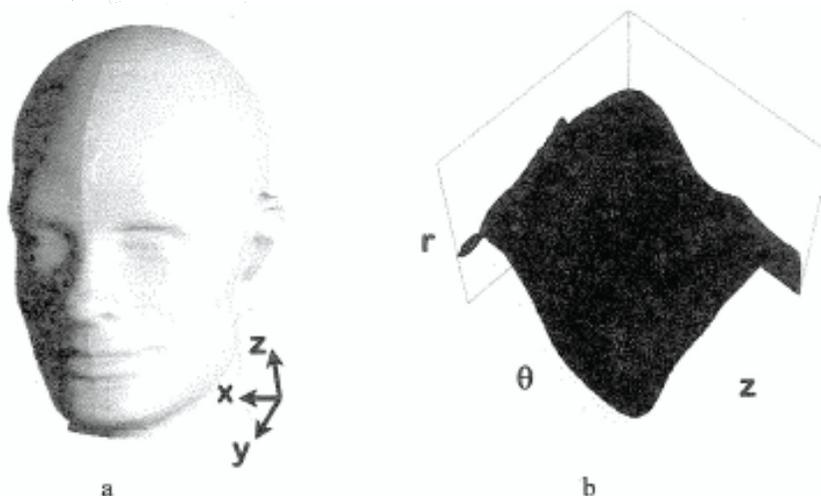


Figure 7
Head in STL form and as regularly sampled points

intervals $\Delta\theta$ and Δz from the STL model in cylindrical coordinates. Figure 7a shows a single valued STL model of a human head. Figure 7b shows a single valued set of points representing the surface of the STL model which were sampled at regular intervals in a cylindrical coordinate system.

There are several issues involved in sampling points from triangles. The most important issue is the loss of surface detail. It is difficult to precisely locate a specific feature with a regular grid, but it is possible to sample data points so that the details are partially represented. A good STL sampling algorithm has two parts, a triangle interpolator and a point averager.

Figure 8a shows how the process of triangle interpolation works. Figure 8b shows how point averaging works.

When the density of triangles in the STL file is higher than the density of points to be sampled, it is best to use some kind of averaging sampler. An averaging sampler works by averaging the values of points within the neighborhood of the point to be sampled. Weightings based on the distance of a point from the point to be sampled can be easily applied. If there are fewer than four points in the neighborhood of the point to be sampled, triangle interpolation should be used. It is always desirable to over-sample data points. The averaging sampler is just a fallback for situations where over-sampling is not practical for computational reasons.

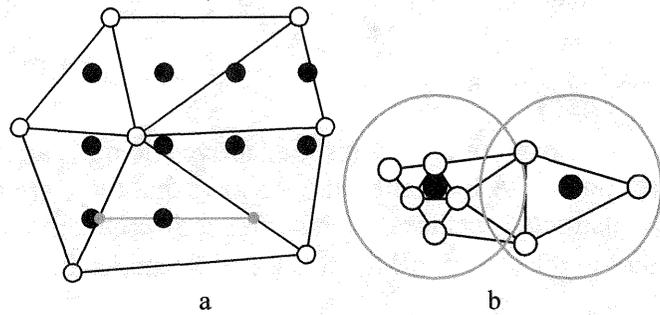


Figure 8
Data sampling

Slicer

Using a simplified set of regularly sampled data makes implementation of the ZIGER slicing algorithm trivial. Figure 9 shows a schematic of data points along a column of data representing the cross section of a surface at a particular angle theta. The gray points and their connecting lines represent the surface of the original part. The black points and their connecting lines represent the surface of a reconstructed slice at the particular angle. It is possible to create the entire slice by individual calculation and slope and radius for each cross sectional angle. For example, if a model were sampled with an incremental angle of one degree, 359 cross sections would have to be processed for each slice. Then each of the cross sections would be laced together to form the actual slice geometry. Calculation of a single slice cross section at a particular angle is carried out with the following formulas:

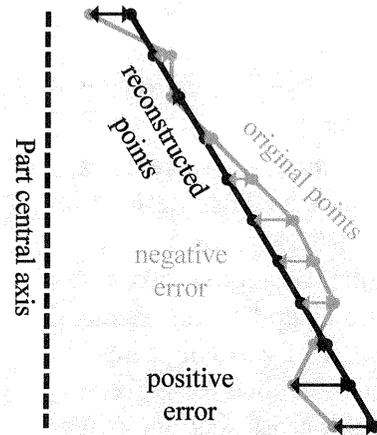


Figure 9
Regularly sampled surface points representing slice cross section a particular angle theta

$$\bar{r} = \frac{\sum_{i=1}^N r_i}{N}$$

Equation 10

$$\bar{s} = \frac{\sum_{i=2}^N \frac{r_i - r_{i-1}}{\Delta z}}{N - 1} \quad \text{Equation 11}$$

where \bar{r} is the sectional slice radius, \bar{s} is the sectional slice slope, Δz is the slice thickness, and N is the number of points in the slice. One \bar{r} and one \bar{s} would be calculated for each angle of each slice.

Adaptive Slicing

Error computation for adaptive slicing is also easy because of the regularly sampled input data points. A look at Figure 9 shows that there are discrete points in the original and reconstructed surfaces, each having a discrete error. The slice errors are calculated using every point in the slice at all heights z and all angles θ .

$$ZIGE = \sum_{i=1}^{N_\theta} \left(\sum_{j=1}^{N_z} r_{re,i,j} - r_{O,i,j} \right) = 0 \quad \text{Equation 12}$$

$$\bar{E}_{abs} = \frac{\sum_{i=1}^{N_\theta} \left(\sum_{j=1}^{N_z} |r_{re,i,j} - r_{O,i,j}| \right)}{N_\theta N_z} \quad \text{Equation 13}$$

$$E_{abs}^{\max} = \max_{i=1}^{N_\theta} \left(\max_{j=1}^{N_z} (|r_{re,i,j} - r_{O,i,j}|) \right) \quad \text{Equation 14}$$

where the error variables are the same as those defined in Eqs. 3-5. N_θ represents the number of angular cross sections, and N_z represents the number of vertical points inside the slice for a given angular cross section. Slice thickness is determined by increasing the number of vertical points used to define a layer without exceeding the dual error criteria, \bar{E}_{abs} and E_{abs}^{\max} . A sample of the slices generated at one radial cross section is shown in Figure 10.

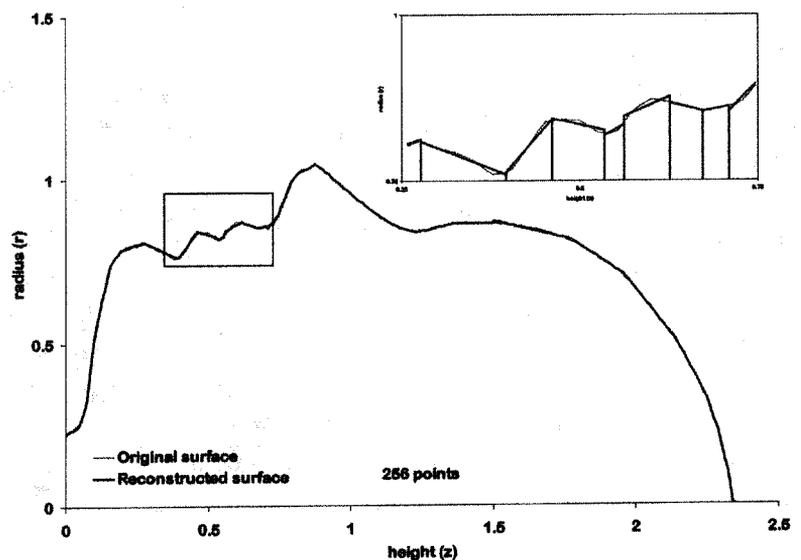


Figure 10
Plot of slices on a single radial cross section through head

Error Visualization

For further evaluation of the algorithm an error visualization system using color contour plots representing radial error on the surface of the reconstructed part were created. In its true form, the algorithm uses a color map. Areas that protrude from the actual surface are shaded blue, areas that are inside the actual surface are red, and areas that are close to the original are green. For this paper a grayscale colormap was used. Areas that are black represent reconstructed surfaces that extend radially beyond the original surface. Areas that are white represent reconstructed surfaces that are inside the original surface. Figures 11 and 12 show two of those plots using a grayscale colormap. Lighting of the surface caused problems with this particular visualization. Places where the light shines directly, the forehead and chin, have been whitened producing a misrepresentation of the actual error. A look at the head shows a banding effect where dark bands indicate boundary edges along the forehead. A look at Figure 12 shows a close up view of the head. It should be possible to pick out the slice boundaries from the plot, even in areas where banding is not pronounced.

Conclusions

The zero integrated error ruled edge slice definition provides an alternative to other slicing definitions. The advantages of the slice definition include the fact that it produces a slice with zero integrated error, it provides a sophisticated measure of error across the entire slice instead of at one point, and it can be used as an adaptive thickness slicing algorithm. As presented here, the slicer is only useful for single valued surfaces. It is a simple first step that proves that the method can work for adaptive thickness ruled edge slicing.

The sophisticated error measures afforded by the algorithm allow slicing with higher accuracy. The important measures of error are the average absolute error, and the maximum absolute error. As stated, these measures are easily adapted to adaptive thickness slicing by using a dual error threshold criteria to determine the thickness of each slice. These errors are representative of the entire slice, not just a point on the slice.

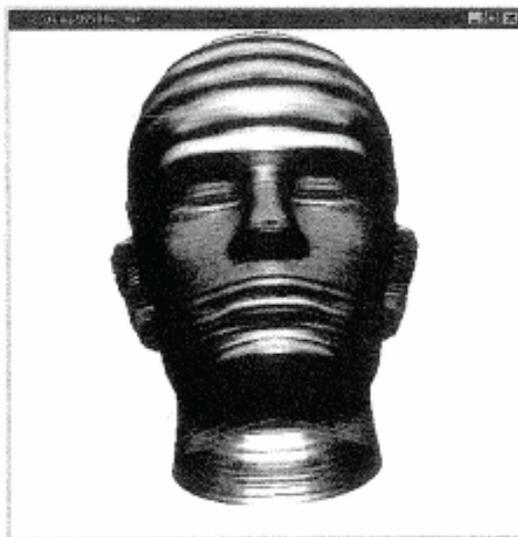


Figure 11
Gray mapped contour plot of error on head
reconstructed from slices
Black represents surfaces that protrude
outward, white represents recessed surfaces



Figure 12
Close up of error plot of reconstructed head

With more work the slicer can be modified to work with generalized geometry including multiple contour parts and sectional transition parts.

References

Dutta, D. and Kulkarni, P. An accurate slicing procedure for layered manufacturing. *Computer Aided Design*. Vol 28 No. 9. 1996. pp. 683-97.

de Jager, P.J., Broek, J.J, and Vergeest, J.S.M. A comparison between zero and first order approximation algorithms for layered manufacturing. *Rapid Prototyping Journal*. Vol. 3 No. 4. 1997. pp. 144-149.

Hope, R.L, Roth, R.N., and Jacobs, P.A. Rapid prototyping with sloping surfaces. *Rapid Prototyping Journal*. Vol. 3 No. 1. 1997. pp. 12-19.

Hope, R.L., Roth, R.N., and Jacobs, P.A. Adaptive slicing with sloping layer surfaces. *Rapid Prototyping Journal*. Vol. 3 No. 3. 1997. pp. 89-98.

Lee, Cheol H. and Thomas, Charles L. An analysis technique for layered manufacturing basee on quasi-wavelet transforms. *Proceedings of the Solid Freeform Fabrication Symposium*. 1997. pp. 577-584.

Lee, Cheol H. New Analysis Methods for Three-Dimensional Objects in Solid Freeform Manufacturing. Ph.D. Thesis. University of Utah. 1997.