STL-based Finish Machining of Rapid Manufactured Parts and Tools

Xiuzhi Qu, Brent E. Stucker
Department of Industrial and Manufacturing Engineering
University of Rhode Island
Kingston, RI 02881

Abstract

Accuracy and surface quality problems when utilizing layered manufacturing technologies have limited its use in tooling areas. Therefore, in some situations a CNC machine is still necessary for finish machining of rapid manufactured parts and tools. This paper presents a STL-based CNC machining technique for automating the finishing of RP tools and parts to obtain CNC accuracies and surface finishes. Preprocess operations, such as rotate and scale, are used to change the part orientation and compensate for shrinkage in the whole process. An offset algorithm is developed to add "skin" to the original STL file to make sure enough material is left for finish machining after the rapid manufacturing process. The machining strategy of adaptive raster milling of the surface, plus hole drilling and sharp edge contour machining, is proposed to finish the RP parts and tools. Corresponding algorithms of adaptive tool path generation for raster milling, automatic hole recognition and drilling tool path generation, and automatic sharp edge detection and tool path generation are developed. Finally, a designed benchmark is machined successfully by using the above mentioned machining strategies and tool paths generated by developed software.

1. Introduction

In order to compete in today's global market, products must be developed and brought to market at an ever-increasing rate. This has led to the wide use of rapid prototyping and manufacturing technologies in industry. Until now, almost all rapid manufacturing techniques use layered manufacturing methods. Layered manufacturing processes, decomposing the 3D CAD model into 2D layers and building part layer by layer, have limitations on surface quality and accuracy. Therefore, in some situations CNC machining is still necessary for finish machining of rapid manufactured parts and tools to increase accuracy and to smooth the surface.

When using CNC machining to finish the parts, pre-processing of the original 3D model and quickly and correctly generating tool paths and machine code from a 3D model are very important issues. To a certain degree, this will determine the success of applying milling to Rapid Prototyping and Rapid Tooling (RP/RT). Thus, a CAM system is being developed to accomplish this purpose.

Today, many CAD systems are being used and each of them uses a different kernel to describe the geometry. It is not possible to read all of these different data formats using one software package. A standard format that can be output by most commercial CAD systems should be used as the input for this CAM system. When looking at standard formats for describing and transferring geometrical data, it was found that the STL format is readily available and can provide great accuracy when created properly. In addition, the STL format is being widely used in rapid prototyping systems that use layered manufacturing technology. It has been shown that this format works very well and reliably for tool path generation [1,2]. Thus, the STL format is a good choice for CAM systems focusing on the application of machining to RP/RT.

The accuracy and surface quality of parts manufactured by RP technology are material and process dependent. In order to obtain high quality parts or tools, some pre-processing of the 3D model is necessary. Translation and rotation operations are used to optimize the part position and orientation. Scaling of the model is used to compensate for the overall shrinkage during the manufacturing process and finish machining is used to improve part accuracy and smoothness. In order to make sure there is enough material left on the surface to be machined, a 3D offset method is developed to add “skin” to the original model.

Considering both accuracy and machine efficiency, the machining strategy of adaptive raster milling of the surface, plus hole drilling and sharp edge contour machining, is applied to finish the RP parts and tools. According to surface curvature information, a stepover distance is calculated adaptively to maintain a minimum cusp height during raster milling of the model surface. This adaptive stepover distance increases machining efficiency while maintaining accuracy and surface quality. Sharp edges are normally the definition curves of features. By machining along these edges, the dimensional accuracy of critical features can be improved. The strategy of recognizing holes from the STL model and drilling them using the appropriate drilling tools reduces machining time and improves part accuracy. For finish machining of rapid manufactured parts or tools, the degree of automation and the total time used to generate tool paths from the model are important criteria. It is more important to generate tool paths automatically and quickly than to generate high efficiency machine code. The latter is more important for mass production, but for this application one, or at most a few, parts will be produced with the same geometry. So an automatic tool-path generation algorithm is developed, which requires very little user interaction.

2. Pre-process of 3D model

In most situations, pre-processing of the original 3D model is required before utilizing a rapid manufacturing process. These operations include changing the part orientation for optimization of building time or surface quality, scaling and offsetting the original model to compensate for part shrinkage and for other kinds of dimensional variations during the manufacturing process.

2.1 Translate, rotate and scale

For most layered manufacturing processes, the building position and orientation will influence part accuracy, surface smoothness and total build time. Thus translate and rotate operations are applied to the original model to optimize the part position and orientation. In addition, shrinkage usually occurs at some point during the manufacturing process. For example, using a SLS machine and a furnace to make cermet parts, it was found that shrinkage would occur during furnace cycles as well as during the SLS process. Preprocess operations on the original model will allow a scale factor to be used to compensate for the overall shrinkage.

The implementation algorithms for these basic operations are first simply translating, rotating or scaling each vertex of triangles of the 3D model, and then recalculating the model size information. For rotate operations, one more calculation is needed to modify the unit normal values of all triangles.

2.2 Offset

In order to make sure there is enough material left on the surface to be machined, adding "skin" to the original model is necessary. From experiments performed in the Rapid Manufacturing Center (RMC) at the University of Rhode Island (URI) [3], it was shown that the shrinkage level varied, even for the same SLS and furnace parameter settings. This variation also requires offsetting of the original model to guarantee that even the features with the largest shrinkage levels still have material left for machining.

The algorithm developed for offset includes first offsetting all of the individual vertices of the model by using the normal information of all triangles, then reconstructing the triangles by using new vertex values. Normally each vertex is shared by several triangles whose unit normal vectors are different. When offsetting the model, the new value of each vertex is determined by the unit normal values of its connected triangles.

Suppose $\vec{V}_{Offset}$ is the unit vector from the original position to the new position of the vertex, and $N_1, N_2, ...N_n$ are the unit normal vectors of corresponding triangles. $\vec{V}_{Offset}$ can be calculated by the weighted mean of those unit normal vectors,

$$\vec{V}_{Offset} = \sum_{i=1}^{n} W_i * \vec{N}_i \tag{1}$$

where W_i are coefficients whose values are determined to satisfy the equation,

$$\vec{V}_{Offset} \bullet \vec{N}_i = 1 \qquad (i = 1,2...n) \tag{2}$$

After solving for $\vec{V}_{Offset}$, the new position of the vertex is given by the equation,

$$P_{new} = P_{original} + \vec{V}_{Offset} * d_{Offset} \tag{3}$$

where $\boldsymbol{d_{Offset}}$ is the offset dimension.

The above procedure is repeated until the new position values for all vertices are calculated. The model is then reconstructed using the new triangle information.

3. Machining Strategy

Machining strategy is very important for the finishing of rapid manufactured parts and tools. Considering both accuracy and machine efficiency, adaptive raster milling of the surface, plus hole drilling and sharp edge contour machining is applied in this paper.

3.1 Adaptive Raster Milling

Isoparametric curve machining, parallel plane-surface intersection curve machining, constant cusp height machining and space-filling curve machining are several very popular machining strategies used for freeform surface machining [4-6]. Through analysis, it was found that parallel plane-surface intersection machining (raster machining) is the most suitable for 3D models represented by triangular planar facets (STL format).

When raster machining is used for milling operations, stepover distance is a very important parameter that controls the machining accuracy and surface quality. It is known that higher accuracy and surface quality require a smaller stepover distance. Normally the cusp height of material left after the model is machined is used as a measurement of the surface quality.

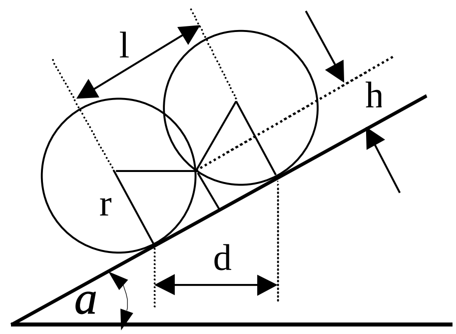

Figure 1. Illustration of determining stepover distance

Fig. 1 shows a triangle face being machined with a ball endmill. The relationship between cusp height h, cutter radius r, stepover distance d, and inclining angle a can be given in the following equation:

$$d = 2.0 * \sqrt{r^2 - (r-h)^2}\cos a \qquad (4)$$

a is determined by the triangle surface normal and stepover direction. Suppose $\vec{N}_{Triangle}$ is the unit normal vector of triangle surface, $\vec{N}_{Stepover}$ is the unit vector along stepover direction, then

$$\cos(\frac{p}{2} - a) = \sin a = \left|\vec{N}_{Triangle} \bullet \vec{N}_{Stepover}\right| \qquad (5)$$

From Equations (4) and (5), the following equation is derived,

$$d = 2.0 * \sqrt{h(2r-h)*(1-(N_{Triangle} \bullet N_{Stepover})^2)} \qquad (6)$$

When machining the model, the cutter radius and milling direction are the same for all triangle surfaces. If given the required cusp height h, d is only related to the triangle normal vector. For surfaces with different normal vectors, the stepover distance obtained will be different. In order to guarantee a machining tolerance, a minimum d should be chosen for the whole raster milling step if a constant stepover distance is used.

Smaller stepover distances, however, will lead to longer programs and machining times. Therefore, an adaptive stepover distance for milling operations according to local geometry is used to allow for both accuracy and machine efficiency. It means that stepover distances are calculated dynamically for each just-finished tool pass, using the maximum cusp height to determine the stepover distance for the next tool pass.

3.2 Sharp Edge Contour Machining

Sharp edges are often the intersection curves between features and surfaces. Normally these edges define the critical dimensions. When using raster milling, the edges parallel to the milling direction can be missed and cause large errors. As shown in Fig.2, the stepover distance d is used to machine a part with a slot. Even when the CNC machine is perfectly aligned (i.e. ignoring machine positioning errors), the slot width error will be at least,

$$W_{Error} = 2d - d_1 - d_2 \tag{5}$$

When d_1, d_2 become 0, $W_{Error} = 2d$. It means that the possible maximum error is approximately two times the stepover distance. A machining experiment performed running tool paths generated by using constant stepover distance on a 3-axis CNC machine to make wax benchmark parts, when measured, also showed this result.

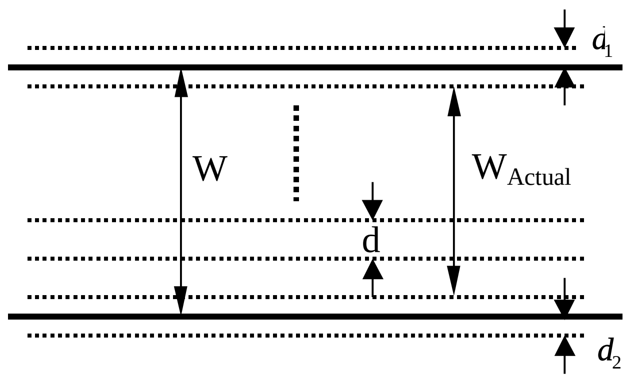

Figure 2. Influence of stepover distance on dimensional accuracy

For complicated edges not parallel to the milling direction, raster milling is ineffective for creating smooth edges. In addition, this will make the adaptive tool path generation algorithm less efficient. A method of milling along recognized sharp edges (contours) is developed to solve this problem.

3.3 Hole drilling

Circular holes are common features in parts and tools. Using milling tools to create holes is inefficient and the circularity of the holes is poor. Therefore, a machining strategy of drilling holes using the correct sized drilling tool is developed. The most challenging aspect is to recognize holes in all directions automatically. In the STL data format, the 3D geometry is represented by a collection of unordered triangular planar facets. Thus all feature information is lost. The algorithms used in the next section allow for reconstructing the holes and determining their diameters, orientations and depths.

4. Automatic Tool Path Generation

One of the essential tasks of finish machining is to generate the tool paths automatically from the 3D CAD model. Based on the machining strategies proposed above, the corresponding algorithms for adaptive tool path generation for raster milling, automatic hole recognition and drilling tool path generation, and automatic sharp edge detection and tool path generation have been developed.

4.1 Tool Path Generation for Raster Milling

For tool path generation using raster milling, the most widely used method is to first calculate the cutter contact point on the model, then use normal direction and cutter size information to find the cutter location. This method normally requires gouge detection algorithms to adjust cutter location data in order to avoid overcuts for some complex surfaces. Gouge detection development is time consuming and complex. An alternative, offset surface method can generate gouge-free tool paths [7], and offset algorithms were developed for the above-mentioned skin addition operation. Therefore, offset surface milling algorithms are used for tool path generation.

Fig. 3 shows a flowchart for this tool path generation program. It is designed for use with ball endmills. The original facet model is offset by the cutter radius. This new model represents the cutter location points. Because not all triangular surfaces can be accessed by the cutter in a 3 axis machine using one setup, only the triangles who have positive z-components of their normal vectors are picked to generate a machinable triangles list. Taking into account the milling direction given and relevant 3D model information, a starting cutting plane is defined. For example, if the milling direction is along the x axis and the minimum y value of model is $y_{\min}$, then the plane $y = y_{\min}$ is defined as the starting plane. By calculating the intersections between the cutting plane and the triangle facets in the machinable list, many line segments will be defined. These segments are sorted, checked and linked to identify the top envelope lines that will become the tool paths. By using equation (6), the required stepover distance can be calculated for each line segment, or tool pass. The minimum distance is used as the stepover distance to update the cutting plane. The above procedure is repeated until the cutting plane lies outside of the model.

4.2 Tool Path Generation for Sharp Edge Contour Machining

In order to machine along sharp edges, all sharp edges must be identified from the STL model first. The normal vector information of each triangle is used to check the property of the edge. The angle between normal vectors of two neighboring triangles is calculated. If this angle is larger than a specified angle, the edge shared by these two triangles will be marked as a sharp edge. If edges are not accessible by a cutter, they are eliminated, and leaving only the “nearly-straight” walls of interest for machining. Thus, hidden edges and edges not belonging to vertical surface are removed from the sharp edges list. Overlaps of edge projections in the XY plane will cause redundant tool paths, they are also eliminated before tool paths are calculated. By offsetting the edges by the cutter radius, the x, y location of the endmill is obtained. The z value is determined by calculating the intersection with the 3D model and finding the corresponding maximum z value. Fig. 4 shows a flowchart of the process.

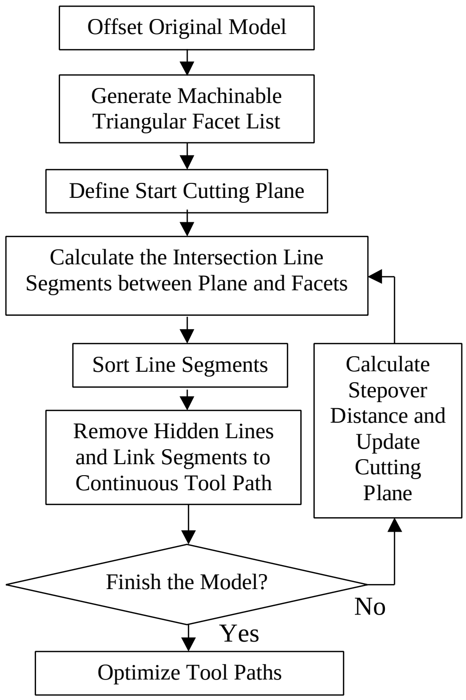

Figure 3. Flowchart of Tool Path Generation for Raster Milling

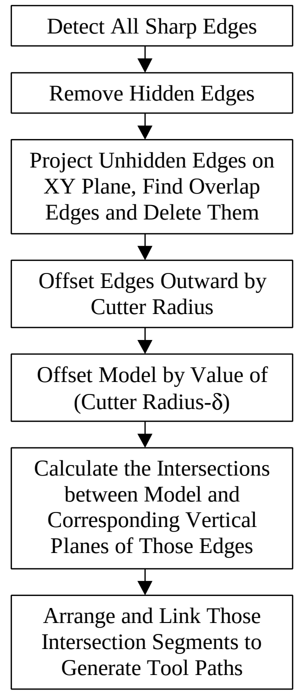

Figure 4. Flowchart of Tool Path Generation for Sharp Edges

4.3 Hole Drilling Tool Path Generation

The intersection curve between a hole and a surface is typically a closed loop. Thus, closed loops are constructed using sharp edges from the entire model. However, these closed loops will not necessarily be the intersection curves between holes and the model surface. A hole checking method is used to remove the loops that do not correspond to a drilled hole. The remaining loops and hole geometry are used to determine the diameter, axis position and direction, and depth for drilling. Finally, the tool path can be easily generated using information. Fig. 5 summarizes this procedure.

The hole checker algorithm shown in Fig 6 illustrates how to determine if loops represent drilled holes in the surface. In most situations, more than 10 triangular facets are needed to reconstruct hole surfaces in order to meet accuracy requirements. This criterion is used to quickly remove unsatisfactory loops. The cross product of two different normal vectors of the inside surface of the hole will give a vector that is parallel to the hole axis direction. Based on this information, 3 evenly distributed edges in each loop are selected. For each edge, there will be two corresponding triangular facets, only one of which belongs to the hole surface. By calculating the cross products of normal vectors of these triangles, the hole orientation is obtained. By projecting all of the edges inside the loop onto the plane that is perpendicular to the hole axis direction, passing through (0,0,0), the circularity of the loop is checked. Center and diameter information is obtained if it meets circularity conditions.

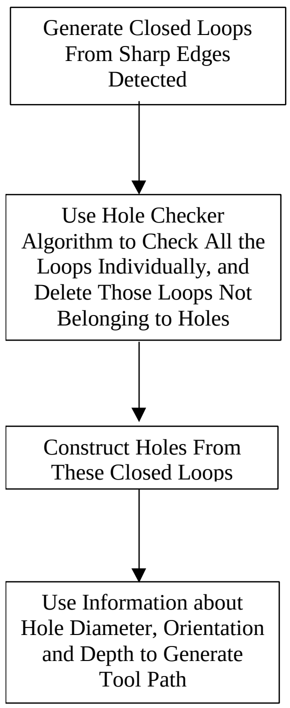

Figure 5. Hole Recognition and Tool Path Generation

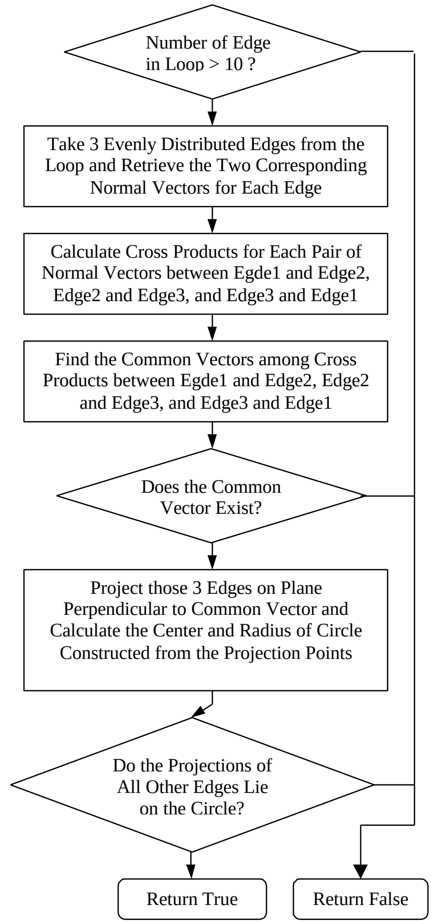

Figure 6. Flowchart of Hole Checker

5. Application

A benchmark shown in Fig.7 is designed to efficiently verify the proposed machining strategies and automatic tool-path generation algorithms developed. This benchmark includes a variety of features, such as different types of surfaces, complicated sharp edges and holes in different orientations. The part is successfully machined using G code generated by the developed software on a 3-axis CNC machine.

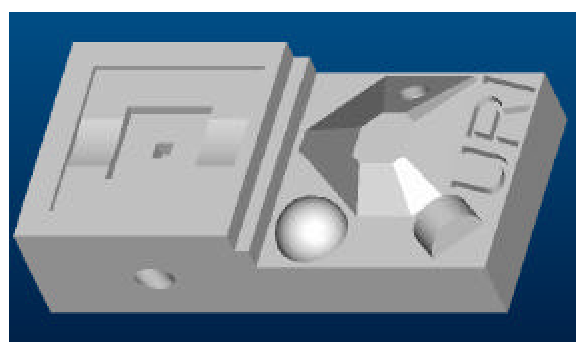

Figure 7. Designed Benchmark

6. Conclusion

The overall objective of this project is to develop techniques for finish machining of rapid manufactured parts and tools. The work focuses on the issues of original STL file preprocessing and STL-based automatic tool path generation for CNC machining.

Preprocess operations, such as rotate and scale, are used to change the part orientation and compensate for shrinkage in the entire process. An offset algorithm was developed to add “skin” to the original STL file to ensure material is left for finish machining after the rapid manufacturing process.

Layered manufacturing techniques have many benefits over traditional tooling manufacture methods. Accuracy and surface quality problems, however, have limited its use in tooling areas. Therefore, in some situations a CNC machine is still necessary for finish machining. Considering the popularity and reliability of the STL format, it was chosen as the input file type for CAM software development. The machining strategy of raster milling of the surface, plus hole drilling and sharp edge contour machining, is proposed and automatic tool path generation has been developed.

Future work on pre-processing will be aimed at adding reference fixture features to the original 3D model to facilitate easy and accurate mounting of the part on a CNC machine. To improve machining efficiency, a surface splitting algorithm will be developed to divide the model into several regions using sharp edge information. These different types of surfaces can be machined by using different strategies and cutters which are more optimum for each type of surface identified.

References

[1]. J.W. Gunnick, Multi-Axis High Speed Milling: How to Speed up Prototyping & Tooling Process by Using STL Technology, Proceedings TCT Conference 1998, Nottingham, Oct 1998:43-65.

[2]. Lex Lennings, Selecting Either Layered Manufacturing or CNC Machining to Build Your Prototype, Rapid Prototyping and Manufacturing Conference 2000, Rosemont(Chicago), April 2000.

[3]. Muni Malhotra, Rapid Manufacture of Zirconium Diboride/Copper & Dimensional Stability of RapidSteel® 2.0, MS thesis, University of Rhode Island, RI, May 2000.

[4]. C.G. Jensen, D.C. Anderson, A review of numerically controlled methods for finish-sculptured-surface machining, IIE Transaction, Vol. 28, pp. 30-39,1996.

[5]. D. Dragomatz and S. Mann, A classified bibliography of literature on NC milling path generation, Computer Aided Design, 29(3), pp. 239-247, 1997.

[6]. Chih-Ching Lo, CNC machine tool surface interpolator for ball-end milling of free-form surfaces, International Journal of Machine Tools and manufacture, 40(3), pp. 307-326, 2000.

[7]. K.I. Kim, K. Kim, A new machine strategy for sculptured surfaces using offset surface, International Journal of Production Research, 33(6), pp.1683-1697, 1995.