September, 2003

Rapid Prototyping Using CNC Machining

Matthew C. Frank, *The Pennsylvania State University*
Richard A. Wysk, *Penn State University*
Sanjay B. Joshi, *Penn State University*

Proceedings of DETC'03
ASME 2003 Design Engineering Technical Conferences and
Computers and Information in Engineering Conference
Chicago, Illinois, USA, September 2-6, 2003

DETC2003/DFM-48157

Rapid Prototyping using CNC Machining

Matthew C. Frank, Richard A. Wysk and Sanjay B. Joshi

Department of Industrial and Manufacturing Engineering
The Pennsylvania State University
University Park, PA 16802, USA

Abstract

Although current rapid prototyping methods have had a significant impact on product and process design, they are often limited in both accuracy and choice of suitable materials. Also, the current methods share little similarity to typical manufacturing processes. In this paper, a method for using CNC machining as a Rapid Prototyping process is described in order to exploit the creation of functional prototypes in a wide array of materials. The method uses a plurality of simple 2½-D toolpaths from various orientations about an axis of rotation, in order to machine the entire surface of a part without refixturing. It is our goal to automatically create these tool paths for machining, and eliminate the complex planning traditionally associated with CNC machining. The current approach to process planning involves calculating all the necessary data from the slice information of an STL model. An overview of the CNC-RP process and the process planning methodology is presented.

Keywords: Rapid Prototyping, CNC Machining, Process Planning, Toolpath Generation

1. Introduction

While the current generation of Rapid Prototyping (RP) processes based on material addition have made significant impact, a large downside to these processes is the limited choice of materials and the limited functionality of parts produced by conventional RP processes. It is advantageous to have a functional prototype as early in the design process as possible. The ability to consider *functionality* and *manufacturability* as early as the conceptual design phase could be enhanced if an accurate prototype in the appropriate material is available. With the advent of high-speed CNC machines, machining is becoming a viable technology to compete with free-form RP. It has the benefit of being able to produce functional parts in the material of choice. However, unlike the traditional RP process, the use of CNC machines is hindered by complex process planning for the generation of tool paths, and fixturing issues, which has limited its development into an RP process [1].

Several researchers have explored the use of CNC machines for rapid prototyping. Chen and Song (2000) describe layer-based robot machining for rapid prototyping [2]. Their method is capable of machining complex geometries, however this is enabled by the fact that the layers are laminated in process. The process was demonstrated using laminated slabs of plastic, machined as individual layers upon gluing to the previous layers. Another approach is to use CNC machining for prototyping dies, an area generally called *Rapid Tooling*. Some approaches include using machined laminates, stacked to form dies [3,4]. Obviously, these approaches do not offer the flexibility or practicality of creating various parts, however short-run production of the same part is possible. A hybrid approach using both deposition and machining called *Shape Deposition Manufacturing* (SDM) is also being developed [5]. For each layer, both support and build material is deposited and machined in a combined additive and subtractive process. Sarma *et al* presented Reference Free Part Encapsulation (RFPE) as a new approach to using phase-change fixturing for machining [6]. RFPE, in combination with feature based CAD/CAM is proposed as an RP system in [7]. However, typical RP processes are feature-free approaches that can handle various complex and freeform geometries.

Lennings (2000) discusses the tradeoffs between layered manufacturing and CNC machining in [8]. In this work, the author describes DeskProto software by Delft Spline systems [9]. This system is similar in theory to the work of this paper. However, process planning in DeskProto provides no analysis of

the part to determine if in fact the part has complete visibility in some set of orientations. For example, consider the toy "jack" in Figure 1a. A set of toolpaths from one orientation leaves most of the surface unmachined. (Figure 1b) The Deskproto process uses a 4th axis to continually rotate the part while an end mill machines a set of parallel surface contours oriented orthogonal to the rotation axis. Unfortunately, their approach still leaves many uncut overhanging surfaces. (See Figure 1c).

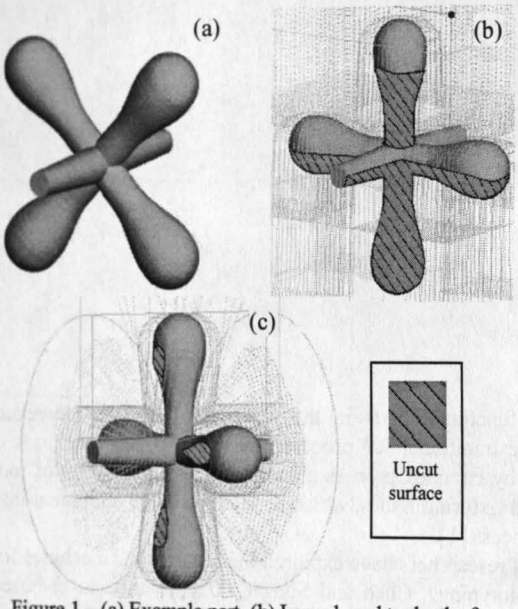


Figure 1 – (a) Example part, (b) Layer based toolpaths from one orientation, (c) Milling with a rotary axis

Previous approaches to CNC rapid prototyping have been limited in their ability to take advantage of the choice of materials and accuracy of CNC machining. Typically, the methods produce rapid tooling for creating parts in another operation such as casting or injection molding or are hybrid techniques that use additive or laminate techniques in combination with machining. The approach described in [8] does provide a means of machining from stock, some complex parts. However, it does not provide the critical geometric analysis needed, and is more limited in geometry versus the approach proposed in this paper.

A more complete approach to RP using 2½-axis CNC machining and a 4th axis indexer is described in this paper. Our approach extends the current state of the art use of CNC milling for RP to include the use of 2½-D milling paths from various orientations to completely machine a part *without* refixturing. The approach to fixturing involves adding sacrificial supports to the ends of the CAD model along the axis of rotation. The 2½-D milling paths are created in a manner similar to slicing in the RP processes. It is our goal to automatically create these tool paths for machining, and eliminate the complex planning traditionally associated with CNC machining. We suggest that the goal for a CNC rapid prototyping method is that the entire set of process planning tasks be completed automatically. In this manner, a designer or engineer can create the NC code for a part without technical expertise in CNC machining. There are several key problems for developing process plans for CNC-RP including; determining a set of orientations, creating layer-based toolpaths, and the creation of a fixture scheme. Methods for automating some of these tasks and a general methodology for CNC rapid prototyping are described in this paper. Specifically, this paper presents methods for 1) Determining a set of orientations needed to machine all surfaces of a part, 2) For each orientation, creating a set of layer-based toolpaths.

2. The CNC-RP Method

The methods presented in this paper are based on a general framework for a solution to CNC rapid prototyping problems. The approach involves machining a complex part from a plurality of tool access directions (orientations). For each orientation, layer-based 2½-D toolpaths are executed. A sufficient number of layer-based toolpaths oriented about an axis of rotation can be used to machine all surfaces of some complex parts. The toolpaths are created using similar "layering" principles as other RP methods, except that the part boundary on each layer or "slice" represents the area that will be left after machining, rather than the area that is to be added. The layer thickness is simply the depth of cut for each tool path routine. Since only 2½-D pockets are being machined, a flat-end mill cutter is used. In CNC-RP part surface contours are created with the same "staircase" effect seen in other RP methods. However, since machining is able to make very shallow depths of cut, CNC-RP can produce thin layer thicknesses. Machining time increases with reduction in layer thickness, but does not necessarily do so proportionally, since shallower depths of cut enable higher feed rates. CNC-RP can achieve layer thicknesses easily down to 0.003" or less. Most RP systems have difficulty with layer thicknesses as large as 0.005".

Consider the surface shown in Figure 2.1a. The freeform shape of the surface can be machined using layer-based toolpaths with a flat-end tool. Notice how the "shadow" region that is blocked by the overhanging surface is left unmachined (area in gray). This region can be machined by another set of toolpaths (Figure 2.1b) oriented counterclockwise from the set in Figure 2.1a.

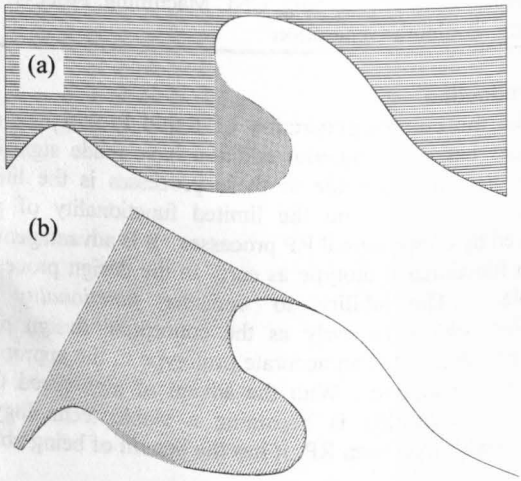


Figure 2.1– Freeform surface machined with multiple layer-based toolpaths

The method uses a 4th axis indexer on the CNC machine to reorient the part. A flexible and automated fixture solution is required. The current fixturing approach borrows from the concept of *sacrificial supports* used in current commercial RP methods. In these applications, thin web structures are created during the build process to both secure the part to the build platform and support overhanging features. The supports are removed after processing of the model is complete. In CNC RP, the requirements for the support structures are different, however the general intention is retained. The goal is to have a fixture solution that is created in the process and is customized for each part. Specific to this work, the fixture supports need to allow the part to be rotated about the axis while providing access to as much of the part surface as possible. Conventional fixturing methods for CNC often utilize vices, clamps, vacuum surfaces, etc. These approaches occlude visibility to a significant amount of the part.

Our concept for sacrificial supports in CNC RP is to add small diameter cylinders to the ends of the part along the axis of rotation. These cylinders are treated as new features on the part and are therefore included in toolpath planning. The part and fixture cylinders are machined from round stock material. The round stock is held between two opposing chucks, one on an indexer and one on a tailstock. This work-holding solution allows the stock to be rotated through the set of orientations. The fixture cylinders are created incrementally and retain the part to the stock material. Deflection due to cutting forces is always a concern in CNC machining. In our approach, cutting forces are both very small (cutting at most, 0.005" per layer) and more consistent (always cut the same layer depth). The round stock is clamped between two opposing chucks, which provides a rigid frame and the fixture cylinders act as beams with fixed supports rather than cantilevered beams. Round stock is appropriate since the tool and work offsets can be set once for all operations. For any orientation about the axis, the initial stock height for toolpath planning will be the same. The steps involved in the CNC-RP method are illustrated in Figure 2.2.

3. Process Planning Methodology

A common characteristic of the current commercialized RP methods is that the *process planning* tasks are almost completely automated. The user does not have to be a skilled technician or have a technical understanding of the particular model building process. For example, an SLA (Stereolithography Apparatus) user need not understand photo-polymerization in order to use the machine. In contrast, current CNC machine users must be familiar with tool selection, toolpath planning, feature selection and sequencing, setting feeds/speeds/depths of cut, and must understand how to implement and/or design a fixture. Currently, a semi-automated process planning method for CNC-RP has been developed. The goal is to develop a planning approach that is able to specify all of the necessary parameters for a CAM package to generate NC code. Another goal was to use as input, an STL (Stereolithography) representation of the CAD boundaries, layer-based toolpath start and finish depths, stock diameter and length, sacrificial support geometry data, and most critical; the set of orientations about the axis of rotation model since the STL model has become the de-facto standard for rapid prototyping. Using only the slice geometry data from the STL file, the process planning method is able to determine tool length and diameter, toolpath containment required to machine all surfaces of the part. The steps of the process planning method are illustrated in Figure 3.1.

In the following sections, we will present an approach to calculating the set of orientations, and determining the parameters for layer-based toolpath planning. Finding the set of orientations is a problem of minimizing the number of *setups,* where a setup is an orientation about the axis of rotation. This is accomplished by generating a visibility map of the model with respect to the axis of rotation. In CAM we can generate layer-based toolpaths automatically, if a few critical parameters can be defined. For each orientation given in the visibility map solution, we analyze the slice file information to determine where the layer-based toolpaths should start and finish (depth), the dimensions of the layers (toolpath containment boundary), the size of the tool and the size of the stock material. To begin, we will discuss the current approach to visibility mapping and how the minimum set of orientations can be found.

3.1 Determining the Set of Orientations

The critical data required for processing a prototype using the method of CNC-RP is the number and orientation of 2½-D toolpaths necessary to machine all surfaces of the part. Other sufficiency conditions must be resolved, such as determining a proper tool length and diameter. In this research, the problem of *visibility* to the surface of a model that is rotated about the 4th axis is investigated. The problem is two-fold: 1) Determine whether all surfaces of the model can be reached with rotations about the selected axis, and if so, 2) Calculate the minimum number of orientations (index rotations) required to machine the entire surface.

Many approaches to machinability and visibility analyze the solid model using surface normal calculations (i.e. Guassian mapping). Notably, Chen and Woo [10] performed seminal work with visibility cones. In Tang *et al* [11] spherical visibility maps are used to find a 4th axis of rotation such that the maximal number of surfaces can be machined. Recently, Balasubramanium *et al* [12] use a tessellated representation of the model surface for generating toolpaths. The same authors note that visibility cones represent *likely* access directions, although obstruction from other surfaces may still prohibit tool access. These visibility maps are created for a section of the 3-D surface and therefore represent local visibility for that particular section of the part surface.

In CNC-RP, accessibility is limited to one rotation axis therefore it is much simpler to solve a series of 2-D visibility problems. Similar to rapid prototyping methods where models are created layer by layer, the visibility algorithm for our method analyzes the CAD model layer by layer.

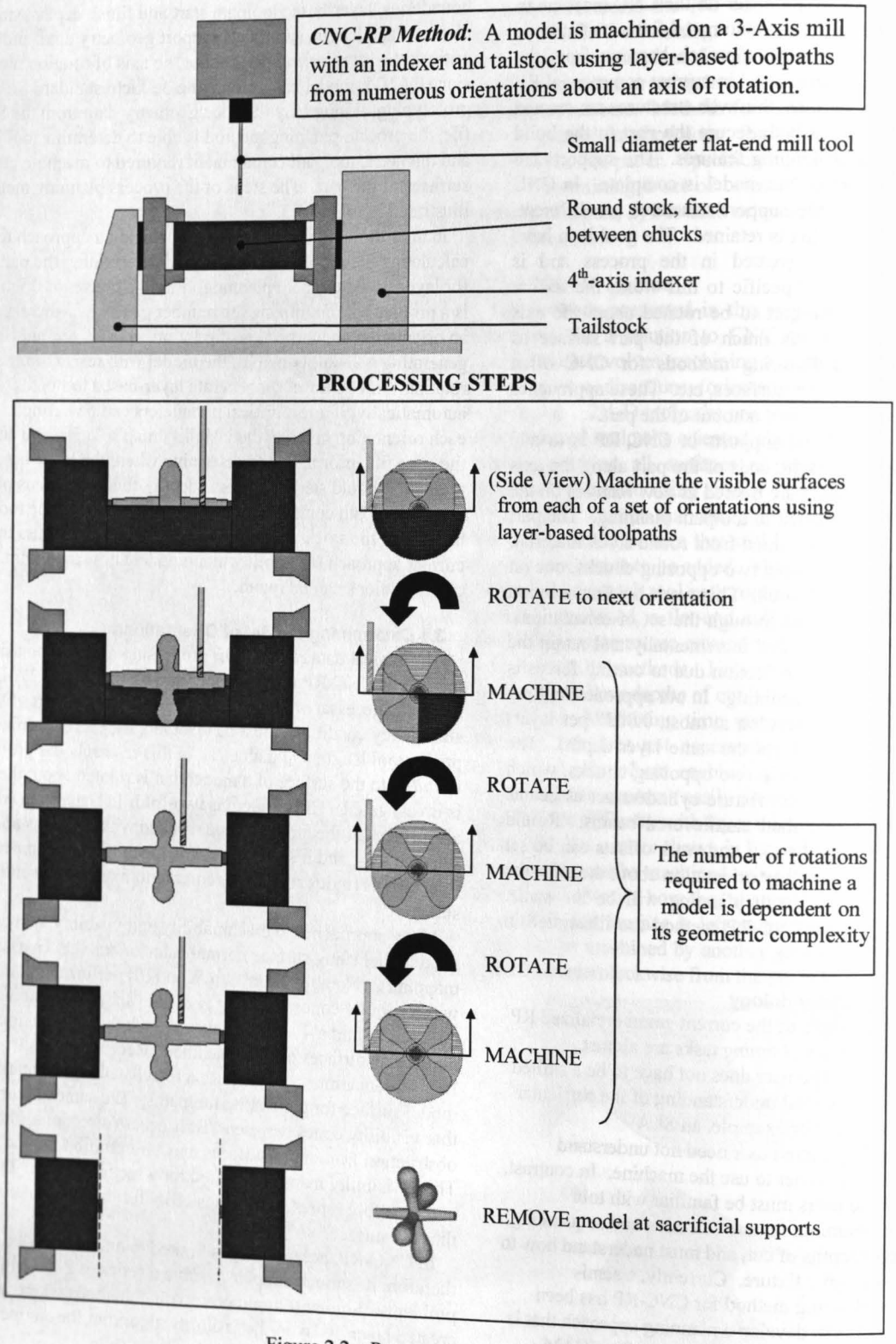
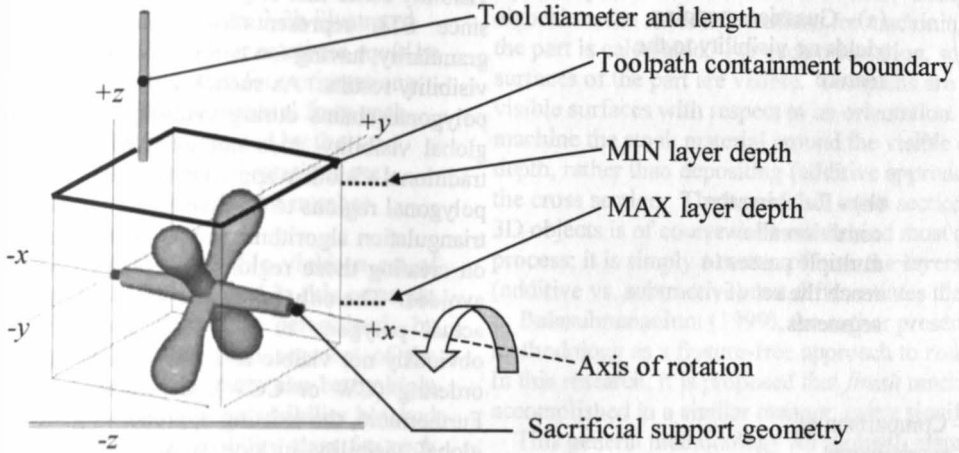


Figure 2.2 – Process steps for CNC-RP

Process Planning Method: An STL model, oriented by the user along a proposed axis of rotation, is analyzed in order to calculate the necessary data that will enable automated layer-based toolpath generation in CAM.



PROCESS PLANNING STEPS

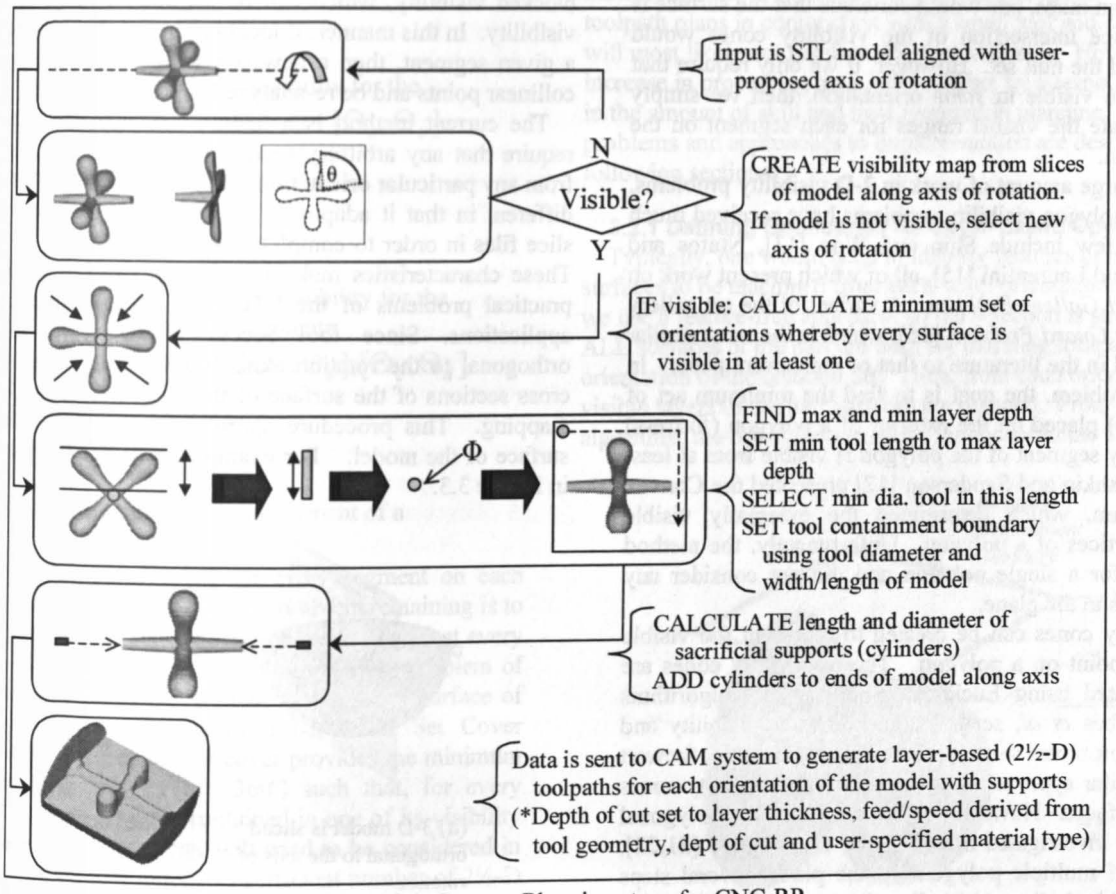


Figure 3.1- Process Planning steps for CNC-RP

We do not require that any arbitrary sections of the surface are simultaneously visible. In other words, it is not a feature-based approach. For example, consider the surface illustrated in Figure 3.2.

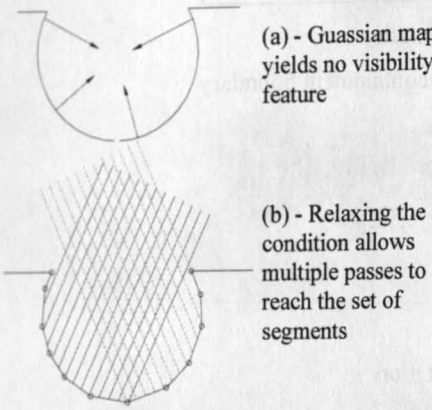


Figure 3.2 – Comparison of visibility requirements

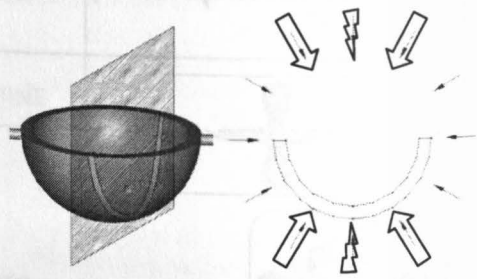
Using Guassian maps, one would conclude that the surface is not visible since intersection of the visibility cones would obviously yield the null set. However, if we only require that *all* surfaces are visible in *some* orientation, then we simply need to calculate the visible ranges for each segment on the polygonal chain.

There is a large amount of work in 2-D visibility problems. In particular, polygon visibility problems have received much attention. A few include Shin and Woo [13], Ntafos and Gewali [14], and Laurentini [15], all of which present work on the popular *Art Gallery Problem.* A variant of the art gallery is the *Fortress Guard Problem* [16], which is the most similar problem found in the literature to that of the current paper. In the fortress problem, the goal is to find the minimum set of points (guards) placed on the exterior of a polygon (fortress) such that every segment of the polygon is visible from at least one point Peshkin and Sanderson [17] presented the Convex Rope algorithm, which determined the externally visible ranges for vertices of a polygon. Unfortunately, the method was suitable for a single polygon and did not consider any other obstacles in the plane.

2D visibility cones can be created to represent the visible ranges for a point on a polygon. These visibility cones are typically created using Euclidean Shortest Path algorithms [18]. In Guibas *et al*, several algorithms for visibility and shortest path problems were presented [19]. Recently, Stewart [20] used similar approaches to determine 2D visibility cones for folded surfaces. However, in his approach, the polygonal regions to be investigated must first be triangulated [21,22]. In the case of multiple polygons in the plane, several steps were involved including; 1) Finding the convex hull of the set of polygons, 2) Determining the set of connected regions (polygons) within the convex hull, 3) Triangulating this set of polygonal regions, 4) Computing the Euclidean shortest paths to each point on the polygons, and 5) Separately dealing with holes.

CNC-RP requires visibility to the surface of the model, so we must explicitly work with the segments of the polygonal chains, not points. Therefore, we wish to create somewhat different visibility *cones* that originate at a line rather than at a cusp. Also, since STL representations can be generated with varying granularity, having too coarse of a slice file may lead to erroneous visibility results. As such, we wish to add collinear points to the polygonal chains during visibility calculations if either local or global visibility does not exist to a particular segment. If a traditional visibility approach is used, then for every point added, polygonal regions to be investigated need to be recreated and the triangulation algorithms rerun. Since this approach does not rely on creating those regions, the problem of dealing with holes is avoided. The only holes, per se, would be any islands within the actual polygons of the slice geometry. These islands are obviously not visible and can be detected simply by their chain ordering (CW or CCW), depending on the slicing algorithm. Furthermore, the existing approaches generate visibility cones for global visibility in one step. In our approach, we wish to separately generate local visibility "cones" and then calculate blocked visibility, which can be subtracted to determine global visibility. In this manner, if local but not global visibility exists to a given segment, then a new sub-segment can be created using collinear points and be re-analyzed.

The current method is a feature-free approach that does not require that any arbitrary section of the surface must be visible from any particular orientation. Moreover, the current approach is different in that it adaptively modifies the representation of the slice files in order to completely investigate visibility conditions. These characteristics make it uniquely capable in handling the practical problems of the STL files used in rapid prototyping applications. Since tool access is restricted to directions orthogonal to the rotation axis, 2-D visibility maps for a set of cross sections of the surface of the model are used for visibility mapping. This procedure approximates visibility to the entire surface of the model. For example, consider the part illustrated in Figure 3.3.



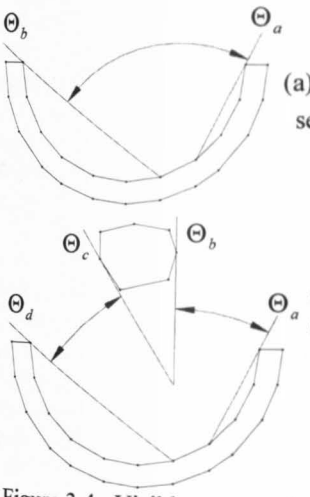
(a) 3-D model is sliced orthogonal to the axis of rotation

(b) Tool access directions are restricted to the slice plane

Figure 3.3 – Model with sample cross-section used for visibility mapping

Cross sectional slices of the geometry from an STL model provide polygonal chains that are used for 2-D visibility mapping. A simultaneous visibility solution for the set of cross sections of the model will approximate visibility to the entire surface. For this simple model and the slice shown in 3.2a, the chain of edges in the polygon can be "seen" from many different views. If the views in Figure 3.2b illustrated by the block arrows are chosen (⇨), four rotations *could* be used to machine the part. This implies that four orientations (index rotations) are used and all visible material from each view is removed. If the two orientations noted by the lightning arrows ( ) are used, then two rotations are needed. In this case, two rotations is the fewest number required.

Each slice (cross section) of the STL file yields a set of polygonal chains. For the method developed in this research, visibility for each polygonal chain is determined by calculating the polar angle range that each segment of the chain can be seen (Figure 3.4a). Since there can be multiple chains on each slice, one must consider the visibility blocked by all other chains. Therefore, the visibility data for each segment can be a set of ranges (Figure 3.4b).



(a) Visibility for the segment= $[\Theta_a, \Theta_b]$

(b) Visibility for the segment= $[\Theta_a, \Theta_b], [\Theta_c, \Theta_d]$

Figure 3.4– Visible ranges of one segment of a polygonal chain

If a visible range(s) exists for every segment on each chain, for all slices in the set, then the problem remaining is to determine the minimum set of polar directions such that every segment is visible in at least one direction. The problem of finding the set of rotations sufficient to see every surface of the model can be formulated as a Minimum Set Cover problem. The solution of the set cover provides the minimum set of angles from the set [0°, 360°) such that, for every segment, at least one angle is contained in one of its visibility ranges. However, other criteria will need to be considered in order to determine a minimum, yet sufficient number of 2½-D toolpaths necessary to machine all surfaces of the part. Tool diameter and length, and the processing sequence for the indexing operations would need to be considered. Furthermore, one would need to determine the axis or axes of rotations necessary to machine all the surfaces.

3.2 Planning Layer-Based Toolpaths

The method is based on the concept of layered machining from a plurality of orientations about an axis. Using the visibility algorithm, the set of orientations for machining all the surfaces of the part is calculated. For each orientation, some but not all of the surfaces of the part are visible. Toolpaths are generated for all visible surfaces with respect to an orientation. The approach is to machine the stock material around the visible cross section at each depth, rather than depositing (additive approach) the material for the cross section. The use of thin cross sectional layers to create 3D objects is of course the basis behind most all of the current RP process; it is simply a matter of how the layers are formed (additive vs. subtractive) that differentiates the current approach. In Balasubramanium (1999), the author presented this methodology as a feature-free approach to *rough* machining [23]. In this research, it is proposed that *finish* machining can be accomplished in a similar manner, using significantly thin layers.

This general methodology for toolpath planning is based on a goal of automation, rather than the typical goals of minimizing machining time or creating a desired surface finish. The resulting toolpath plans in conjunction with a small tool and depth of cut, will most likely not result in *efficient* toolpaths. However, the increase in processing time will be offset by a respective decrease in the amount of skill and time required in planning. Specific problems and approaches to implementation are described in the following sections.

3.2.1 Defining Boundaries for Layer-Based Operations

Typically, one would need to identify features or particular surfaces to be machined from each setup orientation. However, we use a feature-free approach, so the selection is straightforward; ALL surfaces of the part are used for planning toolpaths for every orientation of the solution set. Then, from each orientation, visible layers of the part surfaces are planned. From the visibility algorithm, the orientation to rotate the model in the CAM

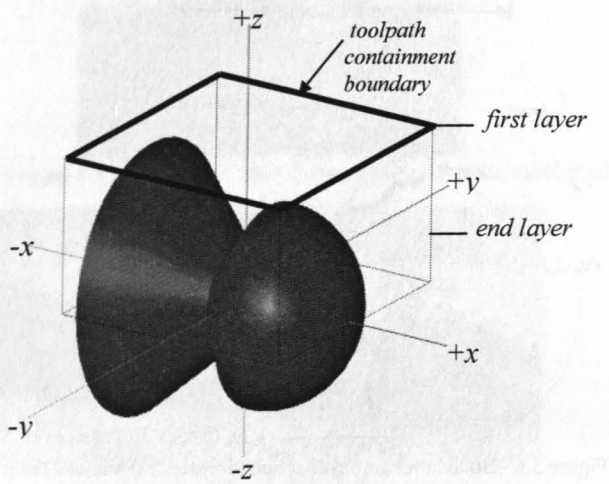


Figure 3.5 – Boundaries for layer-based toolpath planning

environment can be applied. Next, the containment boundaries for creating the layer-based toolpaths must be defined. Assuming the tool is oriented in the z-direction, the containment boundary from above the part is specified by a rectangle (x-y). The other information required is the depth of the maximum and minimum z-level layers. (Figure 3.5)

The length of the rectangle (x) must be greater than the part length along the axis of rotation, while the width of the rectangle (y) must be greater than the maximum part diameter. Specifically, the containment boundary must be greater in both length and width of the part by at least the diameter of the tool (all four sides). This is necessary since the tool will at least require a path around the part, in order to machine around the visible boundaries of the part. The next problem is to determine the depths (z), at which the layer-based toolpaths should begin and end. The assumption is that the first layer must begin at or above the stock radius. The radius of the stock must only be larger than the maximum swept radius of the part, which is simply the maximum distance from the axis to all segment endpoints from all slices. The visibility algorithm yields the set of segments visible from each orientation in the solution set ($\Theta_{s'}$). It is only necessary to machine as *deep* as the deepest visible surface for each orientation. This depth can be calculated as the maximum

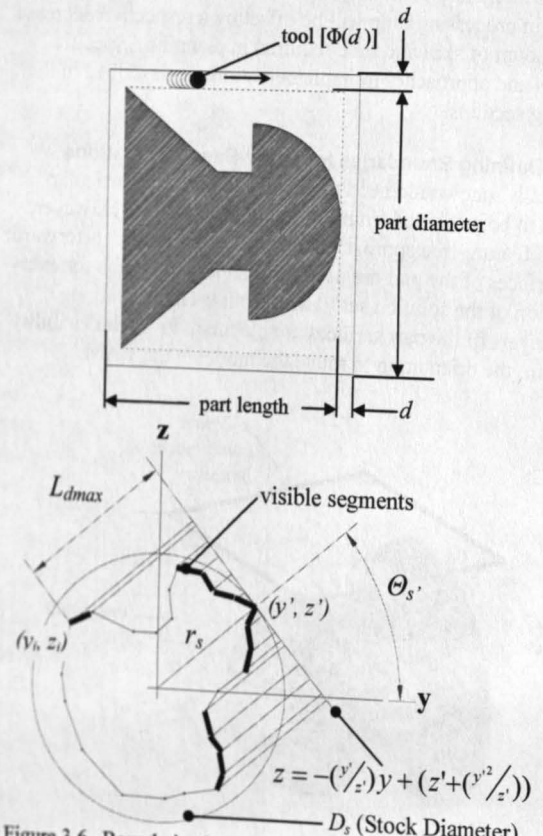


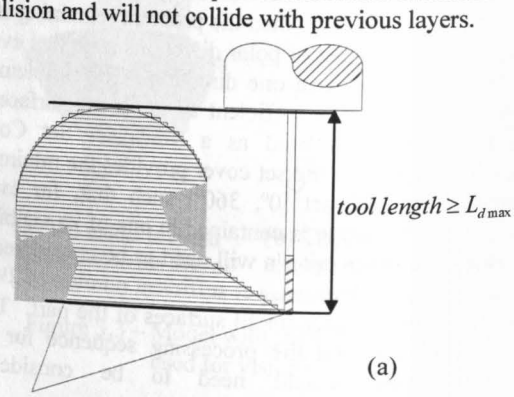
Figure 3.6 –Boundaries for layer-based toolpaths (a) Viewed from tool axis direction, illustrating the layer boundary, (b) Arbitrary slice and maximum depth to visible segments (end layer)

distance from each visible segment endpoint to a tangent line at the solution orientation. The last layer is generated through this maximum distance. (Figure 3.6) Given this information, layer based toolpaths are generated for each rotation of the model in the CAM environment. The layer thickness is set by the *maximum stepdown* parameter in CAM.

The strategy in this section provides the necessary information for setting the boundaries defining the region where layer based toolpaths are generated for a given orientation. Using this approach, the user does not have to identify surfaces or features to be machined. From the data available in the visibility slice files, layer based toolpaths can be "projected" along a solution orientation using the toolpath containment boundaries. The remaining piece of information needed is the tool diameter, which is discussed in the next section.

3.2.2 Selecting Tools

A desired goal is to choose tools that will be capable of machining a variety of complex surfaces. The current approach is to select the smallest tool diameter available in the necessary length that is specified.Since only 2½-D are to be machined, a flat-end tool is most appropriate. Whereas a ball-end (spherical) tool is able to machine smaller radii surfaces in some cases, the diminishing diameter of the cutter contact area is a problem since very shallow depths of cut are used in RP.Typically, RP machines run unattended for several hours, or even days. For an RP process using CNC machining to execute unattended, we must ensure collision-free machining for any model complexity. One condition to ensure collision free toolpaths is that the tool length must be greater than or equal to the distance to the deepest layer to be machined. In the previous section, we specified this distance for the location of the last layer (L_{dmax}, in Figure 3.6). In this manner, one is assured that even on the deepest layer, the tool holder will not collide with the stock. Also, to ensure that no portion of the tool will collide with any previously machined layers, the tool shank diameter must be less than or equal to the flute diameter. This criterion unfortunately makes a tool more susceptible to deflection and possible breakage. Typically, long tools are designed with large shank diameters and only have the length of the cutting portion (flutes) at the prescribed diameter. Figure 3.7 illustrates how a tool with the proposed characteristics can reach to the furthest z-depth for a solution orientation without tool collision and will not collide with previous layers.



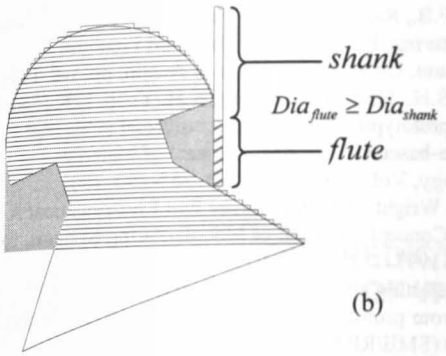


Figure 3.7 – Tool characteristics for (a) Tool length, (b) Tool shank and flute diameters

The active constraint in this approach is the tool length. For any orientation, we must choose a tool that can reach to the visible surfaces. We are left with the task of defining a diameter. Typically, we would analyze the part features and choose the largest tool diameter, in an attempt to minimize machining time (maximize material removal). However, in the context of rapid prototyping we do not assume that feature information is available. The current approach is to choose the smallest diameter flat-end mill, in the required length. We propose that custom tools will be designed for CNC-RP applications, since the depths of cut for this method are so shallow (<=0.005").

Given the length criteria for each orientation, we can choose different strategies for selecting the *set* of tools for the entire model-building process. It would be appropriate to assume that a CNC RP machine would be setup with a finite set of tools by a technician, and then would be periodically serviced as tools wear. For example, a tool changer with a capacity of 20 tools could be loaded with tools ranging from ½" to 10" long. One approach is to use a different length tool for each orientation. This may require as many different tools as there are orientations in the solution set. Another approach is to use one tool for all operations, where the length of the tool is set to the maximum depth for ALL orientations in the solution set ($MAX(L_{dmax})$). (Figure 3.8)

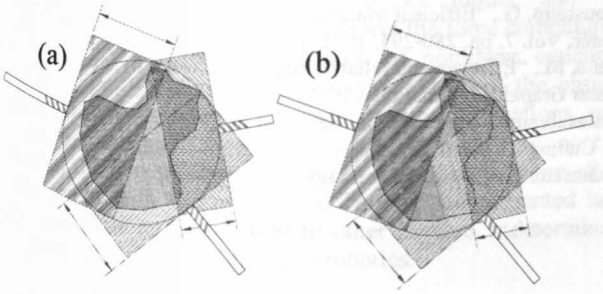


Figure 3.8 – Setting tool length: (a) Tool length = L_{dmax} for each orientation, (b) Tool length = $MAX(L_{dmax})$ for all orientations in the solution set

The tool selection criteria in this section is based on the need to ensure collision free machining for all orientations. We do not suggest that this approach is efficient. However, the approach is capable of handling a wide range of model complexity. From a practical standpoint, and the fact that an STL representation is the expected input, the approach is appropriate for rapid prototyping.

4. Example

In order to illustrate our methodology, a prototype has been created using the CNC-RP method. The prototype was machined from 1.375" 6061 aluminum round stock using a 1/8" flat end mill cutter on a 3-axis machining center. Layer thickness was set at 0.005", although greater or shallower depths of cut could have been used. The complexity of the model required 4 sets of toolpaths rotated 90° with respect to each other. Initial process planning for this model was completed in approximately 15 minutes and the total machining time was 3 hours. We only expect these times to decrease with further development.

The engineering process that was used consisted of creating "The Jack" using the MasterCAM drawing editor. An STL model of the jack was exported and a slice file created. The slice file was analyzed using the visibility algorithm, which yielded the set of orientations. Cylinders were created on the extreme ends of the part model in MasterCAM. The drawing file was then manually rotated in MasterCAM to each of the 4 orientations, and a step of 0.005" was defined within MasterCAM for surface milling. MasterCAM generated the CLData files automatically given our user input specifics (set to maximum feed and speed of 350ipm, and 7500rpm). The part was then machined on a Haas VF-0 3-axis machining center. The part was fixtured as described in the previous sections using a manual 4th axis indexer and a tailstock. Figure 4.1 shows the prototype ("The Jack") after two of four rotations have been completed, while Figure 4.2 is the finished prototype after the fixture cylinders have been removed.

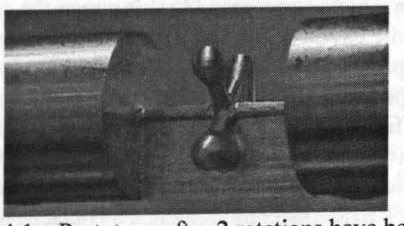


Figure 4.1 – Prototype after 2 rotations have been machined

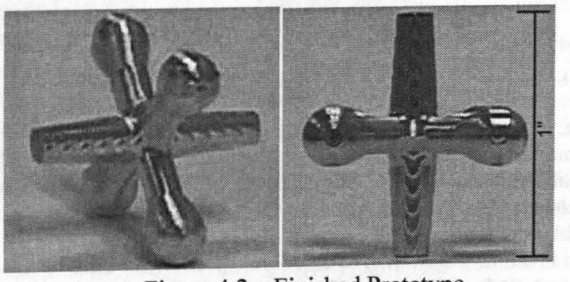


Figure 4.2 – Finished Prototype

5. Conclusions

A novel method for creating rapid prototypes using CNC machining is presented in this paper. Although some of the techniques used are not completely new, it is the systematization of these approaches that has yielded an exciting new method. This method for CNC-RP enables the automation of process and fixturing planning by using a simple set of rotated 2½-D toolpaths from different orientations about an axis of rotation. The method was based on an overall goal of automating the time-consuming process planning tasks that are typically performed by an experienced machinist. The method illustrates an interesting approach when one considers the problem of automating CNC process and fixture planning. The problem is particularly interesting if one assumes that feature information is not available. For example, we describe the use of a minimum diameter tool, given a required length for an operation. Obviously, one would rarely try to minimize a tool diameter, however we are forced to since we cannot assume anything about the geometric complexity of the part. Consider this problem in the context of an additive process like SLA: One could speed up the process if a laser with a larger spot diameter were used, however, what diameter would that be?

There remain many challenges to fully realize this methodology in an automated system, hopefully as an integrated module for an existing CAM package. The method will of course not work for all part geometries. It is assumed that an axis of rotation exists, such that all surfaces are visible when rotated about that axis. Even if an axis exists, the distance to the surfaces may preclude a tool of sufficiently small diameter to be able to create a surface shape. Finally, if a part feature requires a specialized tool (threads, dovetail, etc.) then this method will need to be followed with a post-processing step.

CNC-RP has obvious advantages over other RP systems, since it can produce models in numerous and more appropriate materials, and to a higher accuracy. In this paper, we illustrated how the CNC-RP technique can be used to produce a reasonably sophisticated component ("The Jack"). In many ways, our jack presented difficult accessibility problems as well as very difficult fixturing problems. In spite of this geometric sophistication, our technique performed quite well. Prototypes created by CNC-RP will be readily usable for testing a design's functional requirements, a characteristic not typically available in other RP models.

References

[1] Wang, F.C., L. Marchetti, P.K. Wright, "Rapid Prototyping Using Machining", SME Technical Paper, PE99-118, 1999

[2] Song, Y., Y.H. Chen, "Layer Based Robot Machining for Rapid Prototyping", Proceeding of the ASME Design Engineering Technical Conference, pp. 215-223, 2000

[3] Vouzelaud, F.A., Bagchi, A., Sferro, P.F., "Adaptive Laminated Machining for Prototyping of Dies and Molds", Proceedings of the 3rd Solid Freeform Fabrication Symposium, pp. 291-300, 1992

[4] Walczyk, D.F., Hardt, D.E., "Rapid tooling for sheet metal forming using profiled edge laminations-design principles and demonstration", Journal of Manufacturing Science and Engineering, Transactions of the ASME, vol. 120, no. 4, pp. 746-754, 1998

[5] Merz, R., Prinz, F.B., Ramaswami, K., Terk, M., Weiss, L.E., *Shape Deposition Manufacturing*, Proceedings of the Solid Freeform Fabrication Symposium, University of Texas at Austin, pp. 1-8, 1994

[6] Choi, D.S., Lee, S.H., Shin, B.S., Whang, K.H, Yoon, K.K., Sarma, S.E., " A new rapid prototyping system using universal automated fixturing with feature-based CAD/CAM", Journal of Materials Processing Technology, Vol. 113, No. 1-3, pp. 285-290, 2001

[7] Sarma, S.E., P.K Wright, "Reference Free Part Encapsulation: A New Universal Fixturing Concept", Journal of Manufacturing Systems, vol. 16, no. 1, pp. 35-47, 1997

[8] Lennings, L., "Selecting either layered manufacturing or CNC machining to build your prototype", Proceedings: Rapid Prototyping and Manufacturing 2000 (SME/RPA), Chicago, April 2000

[9] Delft Spline Systems, PO Box 2071, 3500 GB Utrecht, The Netherlands, Url:www.deskproto.com

[10] Chen, L.L., Woo, T.C., "Computational Geometry on the Sphere with Application to Automated Machining, Transactions of the ASME, Vol. 114, pp. 288-295, June 1992

[11] Tang, K., Woo, T.C., Gan, J., "Maximum Intersection of Spherical Polygons and Workpiece Orientation for 4- and 5-Axis Machining", Journal of Mechanical Design, Vol. 114, pp. 477-485, September 1992

[12] Balasubramanium, M., Laxmiprasad, P., Sarma, S., Shaikh, Z., "Generating 5-Axis NC Roughing Paths Directly from a Tessellated Representation", Computer Aided Design, Vol. 32, pp. 261-277, 2000

[13] Shin, S.Y., T.C. Woo, "An Optimal Algorithm for Finding all Visible Edges in a Simple Polygon", IEEE Transactions on Robotics and Automation, vol. 5, no. 2, April 1989

[14] Ntafos, S., Gewali, L., "External Watchman Routes", The Visual Computer, Vol. 10, pp. 474-483, 1994

[15] Laurentini, A., "Guarding the Walls of an Art Gallery", The Visual Computer, Vol. 15, pp. 265-278, 1999

[16] O'Rourke, J., Art Gallery Theorems and Algorithms, Oxford University Press, New York, 1987

[17] Peshkin, M.A., Sanderson, A.C., "Reachable Grasps on a Polygon: The Convex Rope Algorithm", IEEE J. of Robotics and Automation, Vol. RA-2, No. 1, March 1986

[18] Lee, D. T., Preparata, F. P., "Euclidean Shortest Paths in the Presence of rectilinear Barriers", Networks, Vol. 14, No. 3, pp. 393-410, 1984.

[19] Guibas, L. J., Hershberger, J., Leven, D., Sharir, M., Tarjan, R. E., "LinearTime Algorithms for Visibility and Shortest Path Problems Inside Triangulated Simple Polygons", Algorithmica, Vol. 2, pp. 209-233, 1987

[20] Stewart, J.A., "Computing visibility from folded surfaces", Computers and Graphics, Vol. 23, No. 5, pp. 693-702, 1999

[21] Toussaint, G., "Efficient triangulation of simple polygons", Visual Computer, Vol. 7, pp. 280-295, 1991

[22] Held, M., "Efficient and reliable triangulation of polygons", Computer Graphics International 1998", pp. 633-643, June 1998

[23] Balasubramaniam, M., "Tool Selection and Path Planning for 3-Axis Rough Cutting", Thesis, Department of Mechanical Engineering, The Massachusetts Institute of Technology, June 1999