Visibility from a Slice File for Rapid CNC Machining

Matthew C. Frank
_Iowa State University_, mfrank@iastate.edu

Richard A. Wysk
_The Pennsylvania State University_

Sanjay B. Joshi
_The Pennsylvania State University_

Follow this and additional works at: https://lib.dr.iastate.edu/imse_conf

Part of the Ergonomics Commons, Industrial Engineering Commons, and the Operational Research Commons

Recommended Citation
https://lib.dr.iastate.edu/imse_conf/163
Visibility from a Slice File for Rapid CNC Machining

Abstract
A methodology for using CNC machining as a rapid prototyping process is being developed. The method involves cutting complex parts using layer-based machining operations from a plurality of orientations about one axis of rotation. A critical step is to determine the number of and location of those orientations. This paper presents an approach to mapping the visibility of a model about an axis of rotation using a set of model slices taken orthogonal to the axis of rotation.

Keywords
Visibility, Rapid Prototyping, Machining

Disciplines
Ergonomics | Industrial Engineering | Operational Research

Comments

This conference proceeding is available at Iowa State University Digital Repository: https://lib.dr.iastate.edu/imse_conf/163
Visibility from a Slice File for Rapid CNC Machining

Matthew C. Frank
Department of Industrial and Manufacturing Systems Engineering
Iowa State University
Ames, IA 50011

Richard A. Wysk  Sanjay B. Joshi
Department of Industrial and Manufacturing Engineering
Penn State University
University Park, PA 16802

Abstract
A methodology for using CNC machining as a rapid prototyping process is being developed. The method involves cutting complex parts using layer-based machining operations from a plurality of orientations about one axis of rotation. A critical step is to determine the number of and location of those orientations. This paper presents an approach to mapping the visibility of a model about an axis of rotation using a set of model slices taken orthogonal to the axis of rotation. Keywords: Visibility, Rapid Prototyping, Machining

1. Introduction
The labor intensive and time consuming task of manual process planning is recognized as the main factor prohibiting CNC machining from being used as a rapid prototyping (RP) process (Wang, et al. 1999). Existing commercialized RP processes are capable of creating physical models from CAD with little human intervention. Likewise, if CNC machining is to be employed as a rapid prototyping method, one will need to automate the steps involved in creating process and fixture plans. A method for machining complex models using a 3-axis milling machine with a 4th axis indexer is being developed (Frank, et al., 2002, Frank, et al. 2003). A brief overview of the approach is illustrated in Figure 1. The method involves executing layer-based toolpaths from a plurality of orientations in order to machine the surfaces of a model. These toolpath orientations are about an axis of rotation and are indexed using a 4th axis on the milling machine. This method simplifies the problem of toolpath planning by taking a feature-free approach, whereby the goal is to simply machine the visible surfaces from each orientation rather than planning tool paths for each model feature. In addition, the problem of fixturing is simplified by borrowing from the concept of sacrificial supports, as used in other RP processes. Throughout the process, the model is secured to the remainder of the stock material by small cylinders attached to the ends of the model along the axis of rotation. The cylinders are cut in order to remove the model after machining.

The critical data required for processing a part using this method is the number and orientation of the 2½-D tool paths necessary to machine all the surfaces. It is our goal to automatically create these tool paths for machining, and eliminate the complex planning traditionally associated with CNC machining. The reachability of the surface for machining can be abstracted to the geometric problem of visibility. We require that any surface need not be completely visible from only one direction, but there must exist a set of orientations that make the surface completely visible.

Figure 1–Rapid Machining (a) Setup and (b) Process steps
visible. In order for the surfaces to be machined, they must be visible in the tool approach direction. Other sufficiency conditions must be resolved such as determining a proper tool length and diameter; however, these problems will not be addressed in this paper. In this paper, we consider the problem of visibility to the surface of a model that is rotated about a 4th axis. The problem is two-fold: 1) Determine whether all the surfaces of the model can be reached with rotations about the selected axis and if so, 2) Calculate the minimum number of orientations required to machine the part. An open problem is to determine the axis or multiple axes of rotation required to machine all surfaces. This problem will not be addressed in the current paper.

2. Review of Related Work

Many approaches to machinability and visibility analyze the model using surface normal calculations (i.e. Gaussian mapping). Notably, Chen and Woo (1992) performed seminal work with visibility cones. Gan et al. (1994) discuss the properties of spherical maps. Tang et al. (1992) and Chen et al. (1993) use spherical visibility maps to find a 4th axis of rotation such that the maximal number of surfaces can be machined. More recently, Balasubramanium et al. (2000) use a tessellated representation of the model surface for generating toolpaths. They 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 3D surface and therefore represent local visibility for that particular section of the part surface.

There is a large amount of published work in 2-D visibility problems. In particular, polygon visibility problems have received much attention (Lee, 1983; Shin and Woo, 1989; Ghosh and Mount, 1991; Gewali and Ntafos, 1998; Everett et al., 1999; Kapoor and Maheshwari, 2000). Others present work on the popular Art Gallery Problem, which looks at the minimum number of interior points with which all edges of a polygon can be viewed (Shin and Woo, 1989; Ntafos and Gewali, 1994; and Laurentini, 1999). A variant of the art gallery problem is the Fortress Guard Problem, in which 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 (O’Rourke, 1987). 2-D visibility cones can be created to represent the visible ranges for a point on a polygon. These cones can be created using Euclidean Shortest Path algorithms (Lee and Preparata, 1984). Guibas et al. (1987) presented several algorithms for visibility and shortest path problems.

3. General Methodology

Similar to rapid prototyping methods where models are created layer by layer, the algorithm presented in this paper analyzes the CAD model layer by layer. We assume that a proposed axis of rotation is given by the user, similar to choosing a build orientation in the current RP methods. Unlike other approaches, we do not require that all points on any arbitrary section of the surface are simultaneously visible. In other words, it is not a feature-based approach whatsoever. For example, consider the surface illustrated in Figure 2. Using an approach like Gaussian mapping, one would conclude that the surface is not visible since the 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 a surface can be visible after two orientations.

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. 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 many cross sections of the model will approximate visibility to the entire surface. For this simple model and the slice shown in Figure 3a, the chain of edges in the polygon can be “seen” from many different views. If the views in Figure 3b 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 lightening arrows are used, then only two rotations are needed. In this case, two rotations is the fewest number required. 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. 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. If a visible range exists for every segment on each chain, for all slices in the set, then the remaining problem is to determine the minimum set of polar orientations such that every segment is visible in at least one orientation.

Using an STL file for visibility mapping presents some practical challenges. Depending on the accuracy desired, the STL could have few or many triangular facets representing the surface. Therefore, each slice will have few or many segments for each polygonal chain. Suppose a coarse STL is used, and the slice geometry appears like the one in Figure 4. Notice how visibility does not exist to the segment \((uv)\) shown in Figure 4a; however, if a midpoint is added, then the new sub-segment \((uv')\) becomes visible (Figure 4b). For practical purposes, the approach to visibility for rapid machining will need to be able to handle problems such as STL granularity. In this manner, the visibility algorithm needs to be adaptive depending on the visibility conditions. The addition of midpoints to non-visible segments is an approach that can modify the chain representation dynamically such that a finer mapping of the visibility of the surface can be obtained. In other approaches, the assumption is that the surface representation (set of polygons) is fixed, and the algorithm continues whether visibility ranges are found or not.

4.0 Visibility Algorithms

It is appropriate to present the visibility mapping in two phases: 1) calculating the visible range for a segment with respect to the chain on which it resides, and 2) calculating the ranges blocked by obstacles (other chains) on the same slice plane. This is done to separate the visibility analysis into two steps; one that defines local visibility and one that defines the ranges of visibility blocked from obstacles, resulting in global visibility directions.

4.1 Visibility of a segment with respect to its own chain

Visibility to every segment on each surface slice chain is a necessary condition for the machining of all surfaces in Rapid Machining. Visibility to a point on the surface slice chain will first be presented. This formulation will then be extended to segments defined by consecutive endpoints on the polygonal chain. Consider the polygon \(P\) and its convex hull (CH), \(S\), in Figure 5. It can easily be seen that all points on the convex hull \(S\) are visible for a viewing range of at least 180º. For any point \(P_i\) not on \(S\), the visible range can be found by investigating points from the adjacent counter-clockwise (CCW) convex hull point to the adjacent clockwise (CW) convex hull point.

Figure 5 – A point \(P_i\) and its adjacent convex hull points
These points will be denoted the left and right convex hull points of \( P_i, LCHP(P_i) \) and \( RCHP(P_i) \), respectively. If one considers the pocket in Figure 5 with the lid formed by the \( LCHP \) and \( RCHP \), visibility is not possible through any points CCW of \( LCHP \) or CW of \( RCHP \). For any point \( P_i \) not on the CH of \( P \), a line drawn through a point not in the set \( [LCHP, RCHP] \) would have to pass through the interior of \( P \). With that consideration, it is only necessary to calculate the polar angles from \( P_i \) to the points in the set \( [LCHP, RCHP] \), excluding \( P_i \). This set is divided into two sets, \( S_1 \) and \( S_2 \) where \( S_1 : [LCHP, P_{i-1}] \) and \( S_2 : [P_{i+1}, RCHP] \).

Now, the visible range for a point is bounded by the minimum polar angle from \( P_i \) to points in \( S_1 \) and the maximum polar angle from \( P_i \) to points in \( S_2 \). This is the visibility range for the point \( P_i \) with respect to the boundary of its own chain and is denoted \( V(P_i) = [\min_{x \in S_1} \angle P_iX, \max_{x \in S_2} \angle P_iX] \), where \( RV(P_i) = \max_{x \in S_2} (\angle P_iX, X) \), the “right” visible bound for \( P_i \) and \( LV(P_i) = \min_{x \in S_1} (\angle P_iX, X) \), the “left” visible bound for \( P_i \). Using this procedure, it is only necessary to analyze segments of each polygonal chain on the slice in order to determine visibility to the surface. If visibility to all segments exists and all polygonal chains are simple polygons, then visibility exists to the polygon. Likewise, visibility to all polygonal chains on all slices in the set approximates visibility to the entire surface of the 3D model.

Consider the segment \( \overrightarrow{uv} \) defined by points \( u \) and \( v \) in \( P \), where; \( u : P_i \) and \( v : P_{i+1} \). The intersection of visibility ranges for the points \( u \) and \( v \) and the 180° range about the segment define a feasible range of polar angles in which the segment could be reached. Intersecting the visibility ranges for each point with the 180° range about the segment is done since visibility to the segment obviously cannot exist from any direction “behind” the segment. The 180° range about the segment is the set of angles: \( [\angle uv, \angle vu] \). In Figure 6, the ranges are illustrated (\([RV_v, LV_v],[RV_u, LV_u],[\angle uv, \angle vu]\)) of \( uv \). The intersection of the visibility of \( u \) and the visibility of \( v \) will have bounds of \( RV_u \) and \( LV_v \) as: \( (RV_u \cap LV_v = [RV_u \cap LV_v]) \). The sets \( S_1 \) and \( S_2 \) are thus redefined: \( S_1 : [LCHP(u), (u-1)] \) and \( S_2 : [(v+1), RCHP(v)] \). The ends of the visibility range are denoted \( RV(\overrightarrow{uv}) \) and \( LV(\overrightarrow{uv}) \), the right and left visibility bounds of the segment \( \overrightarrow{uv} \), where: \( RV(\overrightarrow{uv}) = [\max_{n \in S_2} (\angle uv)] \) and \( LV(\overrightarrow{uv}) = [\min_{n \in S_1} (\angle uv)] \).

Visibility to the segment \( \overrightarrow{uv} \) is defined as: \( V(\overrightarrow{uv}) : [RV(\overrightarrow{uv}), LV(\overrightarrow{uv})] \). Since not all surfaces will have a simple open pocket as shown in Figure 6, it is necessary to investigate the characteristics of the pocket in order to determine proper bounds for the visibility range, if indeed one exists. There are cases where the minimum angle to points in \( S_1 \) or the maximum angle to points in \( S_2 \) is outside of 180° range above the segment. In this case, \( RV \) or \( LV \) is set to the extremes of \( [\angle uv, \angle vu] \), either \( (\angle uv) \) or \( (\angle vu) \), respectively. There is the possibility that no visibility exists as defined by the range \( [RV, LV] \) due to severe undercuts or overlapping surfaces above the segment. In each of the cases problems occur from naïve setting visibility to \([RV, LV]\). This can be avoided by investigating the characteristics of the pocket where the segment \( uv \) resides. The two points in \( S_1 \) and \( S_2 \) where the bounds \( RV \) and \( LV \) are calculated are used and denoted as \( I_1 \) and \( I_2 \), respectively. (See Figure 7) The geometric relationships between \( I_1, I_2, u \), and \( v \) can be used to determine if, 1) the entrance to the pocket has an overlapping rim that makes visibility impossible, 2) Whether \( RV \) and/or \( LV \), as calculated, are outside of the 180° range, and/or 3). Whether the range defined by \( RV \) and \( LV \) defines an opening that permits visibility to the entire segment from one orientation. Simple algorithms using vector cross products are used to verify the existence of a feasible visible range. Due to lack of space, these algorithms and examples are omitted.
4.2 Visibility blocked by obstacles on the slice plane

The algorithms described in the previous section provide a necessary condition for the visibility criteria of rapid machining; that \( V(\overline{uv}) \) must exist for all segments. This is interpreted as the local visibility of the segment. Other geometric conditions also exist that must be taken into account. For instance, the range \( V(\overline{uv}) \) only considers the visible range with respect to the chain on which the segment resides. However, obstacles in the slice plane can also block visibility \( V(\overline{uv}) \). The problem is to define the set of ranges where a segment is visible in the presence of other chains on the slice. Each slice contains a set of chains \( J \), \( j \in J \) where \( J = \{ 1, ..., n \} \). For any segment on a slice containing \( n \) chains, there could be as many as \( n \) visible ranges for the segment. We will denote \( V(\overline{uv})_j \), as the visibility with respect to the chain \( j \) on which \( \overline{uv} \) resides, denoted \( j^* \). The set of ranges for which \( \overline{uv} \) is visible from the exterior will be called \( VIS(\overline{uv}) \) and represents the global visibility of the segment. It is calculated as the visibility of \( \overline{uv} \) with respect to chain \( j^* \) minus the set of ranges blocked by other chains on the slice. For all obstacle chains, the polar range blocked by the chain is denoted \( VB(\overline{uv})_j \); (Visibility blocked to the segment by another chain on the slice). This set of visible ranges for the segment \( \overline{uv} \) is defined: \( VIS(\overline{uv}) = V(\overline{uv})_j - \sum_j VB(\overline{uv})_j \), for \( j \in J \setminus \{ j^* \} \). Visibility blocked to the segment \( \overline{uv} \) by chain \( j \) is the union of the visibility blocked by chain \( j \) to point \( u \) and the visibility blocked by chain \( j \) to point \( v \), intersected with the range \( [\angle vuv, \angle vuv] \) about the segment \( \overline{uv} \) (Figure 8). The set of angles blocked to the segment \( \overline{uv} \) are: \( VB(\overline{uv})_j = \{ [\angle vuv] \cup [\angle uv] \} \). Considering the condition that blocked visibility is only valid within the range \( [\angle uv, \angle vuv] \) about the segment, then the union operation yields the following range: \( VB(\overline{uv}) = [\angle UV \cup \angle V] \cup [\angle RB \cup \angle LB] \). Calculating \( RB_u \) and \( LB_v \) is straightforward, as \( RB_u \) is simply the minimum polar angle from \( u \) to all points on the blocker chain and \( LB_v \) is the maximum polar angle from \( v \) to all points on \( P_j \), where \( P_j \) is the set of points for the blocker chain \( (RB_u = [\angle UV]\) and \( LB_v = [\angle V] \). At this point, all data is available for calculating the sets of global visibility ranges for each segment: \( VIS(\overline{uv}) = V(\overline{uv})_j - \sum_j VB(\overline{uv})_j \), for \( j \in J \setminus \{ j^* \} \). The output of the visibility algorithm is the collection of visible ranges for each segment on each chain on each slice, given in polar angle about the axis of rotation, as follows: \( VIS_{jk} = \{ \Theta_j, \Theta_{j+1}, \Theta_{j+1}, ..., \Theta_k \} \), where \( r_{\text{MAX}} = n \) (number of chains on slice \( k \)), \( i \) is the segment, \( j \) is the chain, and \( k \) is the slice. A necessary and sufficient condition for the visibility criteria of rapid machining is that visibility as defined by \( VIS_{jk} \) exists for all segments on all chains for all slices of the surface geometry. If this condition is not satisfied, then the entire surface of the part cannot be machined using the proposed method, or at least not using the axis of rotation selected.

4.3 Calculating the Minimum Set of Orientations

From the visibility information, a reverse mapping of the sets of segments visible from orientations about the rotation axis is calculated. This mapping is used to derive the minimum set of orientations, such that all surfaces are machined in at least one orientation. The problem of finding the set of orientations 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 \( [0^\circ, 360^\circ] \) such that, for every segment, at least one angle is contained in one of its visibility ranges. It is noted that the Minimum Set Cover problem is NP-Hard, so we used a Greedy Heuristic to achieve an approximately optimal solution quickly (Chvátal, 1979).

5. Implementation

The visibility algorithms were implemented in C and tested on a Pentium IV, 2.0Ghz PC, running Windows XP. The following table presents data on the processing time for numerous sample models of a model of a toy “Jack”. The STL model granularity, presented as a range from “extra coarse” to “extra fine”, was generated by adjusting the chord height parameter. Slice intervals were taken from 0.0025” between each slice, up to 0.040”. In the table, the number of facets, chord height (CH), and total number of segments in the model are listed, along with the corresponding computation times for the visibility algorithm. (See Table 1)
Using the visibility data from the algorithms, a Greedy solution gave a minimum set of orientations required to machine the Jack. The solution is illustrated in Figure 9. Using these angles for setup, a prototype of the Jack was machined on a Haas VF-0 3-axis machining center. The part was created in approximately 3 hours. In Figure 10, the Jack is shown after being cut from the stock at the sacrificial supports once all orientations were machined.

6. Conclusions
The visibility method presented performs a critical function in automated process planning for rapid machining. The approach provides the data necessary for determining the minimum number of 2½-D toolpaths oriented about an axis of rotation needed to machine the entire surface of a model. Using slice file information as input, the method avoids the problems of feature extraction and identification; an area that has not yielded the automated, robust, solutions we require for rapid machining. This method is also an improvement over existing methods because it modifies the representation of the slice geometry in an effort to seek a feasible visibility solution.

Reference:

Table 1 – Process times for visibility algorithm

<table>
<thead>
<tr>
<th>STL Resolution</th>
<th>coarse</th>
<th>medium</th>
<th>fine</th>
</tr>
</thead>
<tbody>
<tr>
<td>C.H. Facets</td>
<td>0.0075</td>
<td>0.0025</td>
<td>0.000625</td>
</tr>
<tr>
<td>1</td>
<td>865</td>
<td>1990</td>
<td>6578</td>
</tr>
<tr>
<td>#sgmts time(s)</td>
<td>19.566</td>
<td>22.750</td>
<td>36.199</td>
</tr>
<tr>
<td>2</td>
<td>9.772</td>
<td>11.230</td>
<td>18.178</td>
</tr>
<tr>
<td>#sgmts time(s)</td>
<td>14.671</td>
<td>14.671</td>
<td>34.458</td>
</tr>
<tr>
<td>3</td>
<td>4.850</td>
<td>5.687</td>
<td>9.054</td>
</tr>
<tr>
<td>#sgmts time(s)</td>
<td>7.405</td>
<td>7.405</td>
<td>17.306</td>
</tr>
<tr>
<td>4</td>
<td>2.375</td>
<td>2.875</td>
<td>4.597</td>
</tr>
<tr>
<td>#sgmts time(s)</td>
<td>3.907</td>
<td>3.907</td>
<td>8.683</td>
</tr>
<tr>
<td>5</td>
<td>1.182</td>
<td>1.453</td>
<td>2.159</td>
</tr>
<tr>
<td>#sgmts time(s)</td>
<td>2.032</td>
<td>2.032</td>
<td>4.123</td>
</tr>
<tr>
<td>6</td>
<td>0.0200</td>
<td>0.0002</td>
<td>0.000625</td>
</tr>
<tr>
<td>#sgmts time(s)</td>
<td>14.671</td>
<td>11.230</td>
<td>18.178</td>
</tr>
<tr>
<td>7</td>
<td>0.0400</td>
<td>0.0100</td>
<td>0.0200</td>
</tr>
<tr>
<td>#sgmts time(s)</td>
<td>1.182</td>
<td>2.875</td>
<td>4.597</td>
</tr>
<tr>
<td>8</td>
<td>0.0075</td>
<td>0.0200</td>
<td>0.0100</td>
</tr>
<tr>
<td>#sgmts time(s)</td>
<td>14.671</td>
<td>22.750</td>
<td>36.199</td>
</tr>
<tr>
<td>9</td>
<td>0.000625</td>
<td>0.0025</td>
<td>0.0075</td>
</tr>
</tbody>
</table>

Figure 9 - Rotations required to machine the “jack”

Figure 10 – Example prototype