5-2006

# Machinability Analysis for 3-Axis Flat End Milling

Ye Li
*Iowa State University*, yeli@iastate.edu

Matthew C. Frank
*Iowa State University*, mfrank@iastate.edu

# Machinability Analysis for 3-Axis Flat End Milling

**Abstract**
This paper presents a method for geometric machinability analysis. The implementation of the strategy determines the machinability of a part being processed using a plurality of 3-axis machining operations about a single axis of rotation for setup orientations. Slice file geometry from a stereolithography model is used to map machinable ranges to each of the line segments comprising the polygonal chains of each slice. The slices are taken orthogonal to the axis of rotation, hence, both two- and three- dimensional (2D and 3D) machinability analysis is calculated for perpendicular and oblique tool orientations, respectively. This machinability approach expands upon earlier work on 2D visibility analysis for the rapid manufacturing and prototyping of components using CNC machining.

**Keywords**
CNC machining, machinability, slice geometry, tool accessibility

# Machinability Analysis for 3-Axis Flat End Milling

**Ye Li**
e-mail: yeli@iastate.edu

**Matthew C. Frank**

Department of Industrial and Manufacturing Systems Engineering,
Iowa State University,
Ames, IA 50011

*This paper presents a method for geometric machinability analysis. The implementation of the strategy determines the machinability of a part being processed using a plurality of 3-axis machining operations about a single axis of rotation for setup orientations. Slice file geometry from a stereolithography model is used to map machinable ranges to each of the line segments comprising the polygonal chains of each slice. The slices are taken orthogonal to the axis of rotation, hence, both two- and three-dimensional (2D and 3D) machinability analysis is calculated for perpendicular and oblique tool orientations, respectively. This machinability approach expands upon earlier work on 2D visibility analysis for the rapid manufacturing and prototyping of components using CNC machining.* [DOI: 10.1115/1.2137748]

*Keywords: machinability, tool accessibility, CNC machining, slice geometry*

## 1 Introduction

Machinability analysis is taking an increasingly important role as complex surfaces are used in the design of a wide variety of parts. Current computer-aided manufacturing (CAM) software is readily capable of generating toolpaths given a set of surfaces of a part and a cutting orientation (3-axis machining). However, determining the setup orientation can be difficult and moreover, it may be very challenging to determine if the part can be created using machining at all. An appropriate setup orientation can guarantee an effective cutting of the surface, whereas an inappropriate one will leave too much material in certain regions. The advancement of 5-axis computer numerically controlled (CNC) milling machines seems to alleviate this situation; however, often the cost and/or difficulty of programming a 5-axis machine have limited their widespread use. Three-axis machines, as economical and technologically mature pieces of equipment, have been paid special attention with respect to complex surface machining if assisted with multisetup devices (e.g., a programmable indexer). Suh and Lee [1] used a 3-axis machine with a rotary-tilt-type indexer to provide an alternative to 5-axis ball end milling. Suh et al. [2] provided a theoretic basis for machining with additional axes. Recently, Frank et al. [3] employed a 3-axis milling center with a fourth axis indexer as an effective rapid prototyping machine. End mills have been shown to offer a better match to the part surface geometry, a higher material removal rate, and a longer tool life compared to ball-mills [4]. Ip and Loftus [5] demonstrated the competency of an inclined end mill machining strategy on 3-axis machines in producing low curvature surfaces. However, to machine a surface with large curvature variation, it is necessary to determine a set of machining orientations and carry out multiple 3-axis machining operations in a sequential manner with respect to each of those orientations. Therefore, an effective machinability analysis is of critical importance to the successful implementation of multiple orientation 3-axis machining for creating complex parts.

Many researchers have studied machinability analysis and its closely related workpiece setup problem. Most of the approaches are based on visibility, which is essentially line-of-light accessibility. Su and Mukerjee [6] presented a method to determine machinability of polyhedral objects. A convex enclosing object is constructed to make each face of the part orthogonally visible to the planes of the enclosing object. The part is then considered to be machinable from the normal-vector directions of the enclosing object planes. Later, computational geometry on the sphere was utilized to analyze visibility by Chen and Woo [7] who performed pioneering work on computational geometry algorithms that could be used for determining workpiece setup and machine selection. Tang et al. [8] formulated the problem of workpiece orientation as finding the maximum intersection of spherical polygons. Gan et al. [9] discussed the properties and construction of spherical maps and presented an efficient way to compute a visibility map from a Gaussian map. Chen et al. [10] partitioned the sphere by spherically convex polygons to solve the geometric problem of determining an optimal workpiece orientation for 3-, 4-, and 5-axis ball end milling. A visibility map is generated by using the normal vectors of a specified *portion* of the surface of a part; therefore, it cannot guarantee global accessibility. Yang et al. [11] computed visibility cones based on convex hull analysis, instead of relying on visibility maps. Yin et al. [12] defined complete visibility and partial visibility, and presented a C-space-based method for computing visibility cones. A sculptured surface is approximated by its convex hull [11], and the spherical algorithms [7,13] are used in the approach of Yin [12]. The convex hull may, in some cases, have a significant deviation from the true surface. Suh and Kang [14] constructed a binary spherical map to compute the point visibility cone in order to algebraically solve machining configuration problems, including workpiece setup orientation. The part surface is decomposed into triangular patches. An occupancy test of the patches is conducted on a triangular-represented unit sphere to generate global visibility. Dhaliwal et al. [15] presented a similar approach for computing global accessibility cones for polyhedral objects, but with exact mathematical conditions and algorithms. Balasubramaniam et al. [16] analyzed visibility by using computer hardware (graphics cards). Frank et al. [17] analyzed two-dimensional (2D) global visibility on stereolithography (STL) slices and searched the necessary machining orientations for fourth-axis indexable machining by executing a GREEDY search algorithm. All these visibility-based approaches determine the necessary condition for machinability; however, they ignore tool geometry and, therefore, true accessibility (machinability) is not guaranteed. Figure 1 shows that the accessibility cone $(\alpha, \beta)$ based on line-of-light visibility cannot guarantee the true accessibility using a sized tool in machining a segment $\overline{ij}$.

Su and Mukerjee [6] took into account the cutter information by constructing a new part model through offsetting the original part surface by the amount of the cutter radius. Machinability was further guaranteed by checking the topology of this offset part

Contributed by the Manufacturing Engineering Division of ASME for publication in the JOURNAL OF MANUFACTURING SCIENCE AND ENGINEERING. Manuscript received October 13, 2004; final manuscript received August 8, 2005. Review conducted by D.-W. Cho.

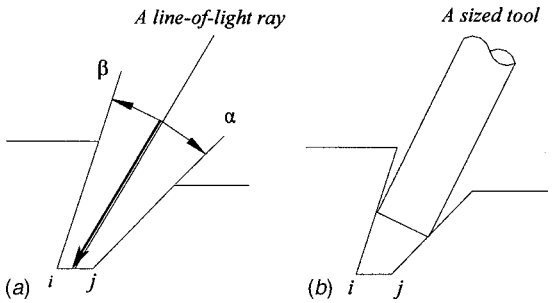



**Fig. 1 Accessibility based on light ray and a sized tool**

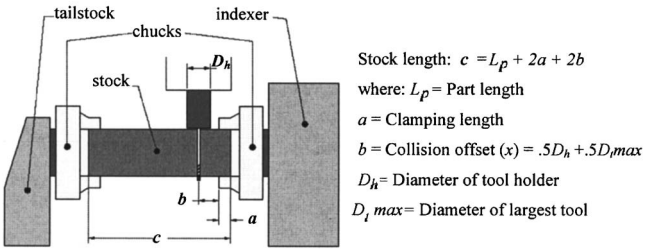



**Fig. 2 Setup for rapid machining**

surface. This method is effective for the machinability analysis of a ball end cutter, but not for that of a flat end cutter, because the effective radius of a flat end cutter is variable with the change of tool tilting angle. Haghpassand and Oliver [18] and Radzevich and Goodman [19] considered both part surface and tool geometry. However, tool size was not taken into account because Gaussian mapping does not convey any size information of the part surface and/or the tool. Balasubramaniam et al. [16,20] verified tool posture from visibility results by collision detection before interpolating the tool path for 5-axis machining.

Over the past years, feature-based technologies have been an active field among the manufacturing research community. Regli [21], Regli et al. [22], and Gupta and Nau [23] discussed feature accessibility and checked it by calculating the feature accessibility volume and testing the intersection of the feature accessibility volume with the part. Gupta and Nau [23] recognized all machining operations that could machine the part, generated operation plans, and checked and rated different plans according to design needs. A comprehensive survey paper on manufacturability by Gupta et al. [24] reviewed representative feature-based manufacturability evaluation systems. Shen and Shah [25] checked feature accessibility by classifying the feature faces and analyzing the degree of freedom between the removal volume and the workpiece. The MEDIATOR system reported by Gaines et al. [26] used the knowledge of manufacturing equipment to identify manufacturing features on a part model. Accessibility is examined by testing the intersection of removal volumes with the part. Faraj [27] discussed the accessibility of both 2.5-D positive and negative features. Other researchers presented featured-based approaches to determine workpiece setups [28–31].

Although feature-based approaches are capable tools to handle feature-based design, they cannot lend themselves to free-form surfaces where definable features may not exist. In addition, feature-based approaches suggest that all the geometric elements comprising of a feature are treated together as an entity. This actually imposes a constraint to the analysis of a part model. For example, it might be feasible to machine a portion of a part feature in one orientation and then finish the remaining surfaces of the feature in one or more successive orientations. The current problem that this paper addresses is based on a rapid machining strategy proposed by Frank et al. [3] whereby a part is machined with a plurality of 3-axis machining operations from multiple setup orientations about a single axis of rotation.

The strategy is implemented on a 3-axis CNC milling machine with a fourth-axis indexer (Fig. 2). Round stock material is fixed between two opposing chucks and rotated between operations using the indexer. For each orientation, all visible surfaces are machined using simple layer-based tool-path planning. By setting the collision offset ($b$) (shown in the Fig. 2) on each side of the workpiece, the implementation of rapid machining can avoid the risk of collision between tool holders and the holding chucks. The diameter of largest tool ($D_{\text{tmax}}$) used to calculate the collision offset ($b$) makes the setting of collision offset for each new part unnecessary. The *feature-free* nature of this method suggests that it is unnecessary to have any surface be completely machined in any particular orientation. The goal is to simply machine *all* surfaces after *all* orientations have been completed. The number of rotations required to machine a model is dependent on its geometric complexity. Figure 3 illustrates the process steps for creating a typical complex part using this strategy.

Currently, the necessary cutting orientations are determined by 2D visibility maps with tool access restricted to directions orthogonal to the rotation axis. Cross-sectional slices of the geometry from an STL model are used for 2D visibility mapping. The visibility of those slices approximates the visibility of the entire surface of the part along the axis of rotation since the slices are generated orthogonal to that axis. The above literature review suggests that existing approaches to machinability cannot calculate the set of orientations for setups such that one can machine *all machinable* surfaces after *all* orientations, because either (i) 2D or three-dimensional (3D) visibility cones employed by the

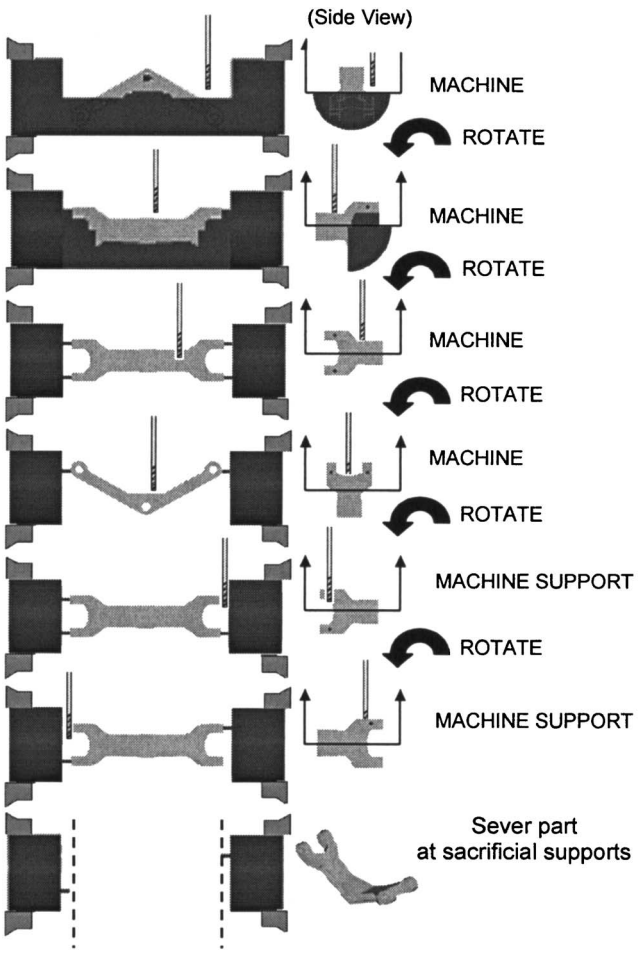



**Fig. 3 Process steps for rapid machining**

visibility-based approaches convey no size information of the tool and workpiece and, therefore, cannot guarantee true accessibility; or (ii) the feature-based approaches cannot cope with complex (free-form) surface machining because few traditional features can be identified on parts with free-form surfaces.

An effective machinability analysis method is a prerequisite to the successful implementation of multisetup 3-axis end milling in order to achieve the needs of 4- and perhaps 5-axis machining. An effective machinability analysis method will determine, given a machining orientation and an end mill of a particular size, how much of the part surface can be machined with respect to this machining orientation. The focus of this paper is to present a feature-free machinability analysis that can determine the number of setups required to completely machine the surfaces of a part with one-axis-of-rotation setups. The machinability analysis method presented in this paper is unlike any previous work in its completely feature-free treatment of the part geometry. We reduce the surfaces of the part down to simple line segments on the slices; therefore, any CAD model can be exported as an STL file and studied. This approach is done because we are only assuming that the part is machined about one axis of rotation; therefore, it is much simpler to simply analyze the 2D slices rather than 3D surface geometry.

The remainder of this paper is organized as follows. In Sec. 2, definitions that are used throughout this paper are presented. Section 3 discusses the machinability analysis method in further detail, and Sec. 4 presents the implementation of the machinability analysis approach. Last, conclusions and future research endeavors are provided.

## 2 Definitions

Although previous researchers have defined the concepts of *visibility* and *machinability* in their work, similar definitions are provided first in this section to clarify the difference between visibility and machinability. Next, the concepts of *tool space* (TS), *obstacle space* (OS), and *machinable range* (MR) are introduced. A condition to determine the existence of machinability is also derived. The definitions provided in this section are used for the subsequent discussion in the remainder of this paper.

- *Visibility*: A point $p$ on a surface $S(p \in S)$ is visible by a light ray emanated from an external point $q$ if $\overrightarrow{pq}$ suffices the condition of $\overrightarrow{pq} \cap (S-p) = \Phi$.
- *Machinability*: A point $p$ on a surface $S(p \in S)$ is machinable by a certain type and size of tool $T(\mathrm{CL}, \alpha)$ if $p \in T(\mathrm{CL}, \alpha)$ and $T(\mathrm{CL}, \alpha) \cap (S-p) = \Phi$. $T(\mathrm{CL}, \alpha)$ represents the tool surface at the cutter location CL, approaching from the orientation $\alpha$.

By definition, machinability shares the same concept of accessibility with visibility, but differs in the sense that machinability takes into account the size and shape of the cutting tool instead of treating it simply as a line of light. Therefore, machinability can guarantee true accessibility, whereas visibility is only a necessary condition of machinability. Hence, the aggregate of orientations satisfying machinability is a subset of that satisfying visibility. In other words, machinability can guarantee visibility, but not vice versa.

Unlike the expression of visibility in angular orientations, the bundle of which forms a cone, there are two parameters used to describe machinability. They are the cutter location and the approaching orientation, if the type and size of a cutter are specified. Machinability with respect to an approaching orientation $\alpha$ exists only if there is a cutter location that allows the cutting tool to approach and touch the point $p$ without intersecting any other part surface.

Similar to the concept of the visibility of a feature, the machinability of a feature (a line, a curve, or a patch of surface that is geometrically composed of a set of points) is the intersection of the machinability of each point belonging to that feature. Similar to the concept of *partial visibility* (PV), *partial machinability* (PM) of a feature can also be defined in addition to the concept of *complete machinability* (CM).

- *Partial Machinability*: A feature is partially machinable along an orientation $\alpha$ if there exists at least one point on that feature such that no cutter location CL exists for it to suffice the condition of $p \in T(\mathrm{CL}, \alpha)$ and $T(\mathrm{CL}, \alpha) \cap (S-p) = \Phi$.
- *Complete Machinability*: A feature is completely machinable along an orientation $\alpha$ if for each point on that feature at least one cutter location CL can be found to guarantee the condition of $p \in T(\mathrm{CL}, \alpha)$ and $T(\mathrm{CL}, \alpha) \cap (S-p) = \Phi$.

Note that *Complete Machinability* may exist for either a point or a feature, whereas *partial machinability* exists only for a feature, because a point can only be said to be either machinable or nonmachinable.

If machinability exists with respect to an approaching orientation $\alpha$, the number of feasible cutter locations CLs may vary with different points on a surface. Points with more feasible CLs translates to easier machining because the more possible CLs provide more options for tool-path and setup planning. The need to measure the space of cutter locations leads to the concept of *tool space*.

- *Tool Space*: The aggregate of all feasible cutter locations to cut a point $p$ from an orientation $\alpha$ forms a region called tool space, written as $\mathrm{TS}(p, \alpha) = \{\mathrm{CL} : p \in T(\mathrm{CL}, \alpha) \text{ and } T(\mathrm{CL}, \alpha) \cap (S-p) = \Phi\}$.

Tool space of a feature $F$ is the union of the tool space of every point belonging to $F$; that is, $\mathrm{TS}(F) = \{\cup \mathrm{TS}(p, \alpha) : p \in F\}$. A tool space reaches its maximum value *maximum tool space* (MTS) when there is no obstacle around the geometric entity. Here, we consider the entire part surface except the portion under consideration to be obstacles. Thus, the corresponding space for obstacles is defined as *obstacle space*.

- *Obstacle Space*: The aggregate of all unfeasible cutter locations with respect to an orientation $\alpha$ due to the existence of an obstacle $i$ ($Obi$) is called the obstacle space of obstacle $i$, written as $\mathrm{OS}(i, \alpha) = \{\mathrm{CL} : T(\mathrm{CL}, \alpha) \cap Obi \neq \Phi\}$.

The cutter cannot enter the domain of obstacle space because it will gouge into the obstacle.

Tool space can be computed by subtracting all the *obstacle spaces* from *maximum tool space*.

$$\mathrm{TS} = \mathrm{MTS} - \sum_i \mathrm{OS} \tag{1}$$

If the computed tool space using Eq. (1) is not empty, then machinability exists; otherwise, the geometric entity is nonmachinable. The machinability analysis method presented in this paper is based on Eq. (1). Tool space is actually a measure of machinability since it tells the existence of machinability and the magnitude of machinability, if it exists.

Once the tool space is determined, the machinable range resulting from it can be obtained.

- *Machinable Range*: The maximum machinable portion of a feature given the tool space is called machinable range of that feature, written as $\mathrm{MR} = \{p : p \in F \text{ and } \mathrm{TS}(p, \alpha) \neq \Phi\}$.

The above definitions will be used throughout the remainder of this paper.

## 3 Machinability Analysis

The machinability analysis approach presented in this paper is based on the concept of configuration space (C-space). The concept of C-space first applied in robotic spatial motion planning was documented by the work of Lozano-Perez [32]. The basic idea of C-space is to find the aggregate of the valid spatial configurations for a moving mechanism in an environment with obstacles around it. Recently, C-space has been applied in tool-path planning for multiaxis machining. Choi et al. [33] presented a C-space-based approach to generate 3-axis numerically controlled (NC) tool paths for sculpture machining by transforming the designed part surface and stock-surface into elements in C-space and treating the cutter as a moving object in the safe space. The C-space is represented and computed using a Z-map model of the part. Choi and Ko [34] incorporated C-space into computer-automated process planning (CAPP) for freeform die-cavity machining. Morishige et al. [35] used C-space to generate tool paths for 5-axis ball end milling. Jun et al. [36] optimized tool orientations for 5-axis flat end milling by a search method in C-space. C-space of the cutting tool, which is defined as tool space in Sec. 2, provides the safe space for tool-path planning; therefore, tool paths based on C-space are always gouge-free and collision-free. The tool space can be seen as the aggregate of all possible tool paths. If tool space exists, then at least one tool path can be generated to machine the corresponding geometric point on the part surface, and hence, this point has machinability. Therefore, by testing the tool space of each point on a surface, the machinability of this surface can be theoretically determined.

The input for the machinability method of this paper is the slice file of an STL model of the part along the intended axis of rotation. An STL model is an approximation of the part surface by using triangular facets and is currently the de facto standard file format for rapid prototyping systems. The tessellation process to create an STL model can keep the approximation error—the deviation between the part surface and the tessellation triangles—within a specified tolerance. The size and shape of each triangle created is adaptive to its local region on the part surface. Therefore, compared to the Z-map model employed by Choi et al. [33], which is essentially a virtually equal-spacing sampled model, the slice file of an STL model is a more precise and efficient representation of the original part surface. The slicing process for the STL model of the part, which may be either a feature-based part or a free-form geometry, breaks the part surface into many line segments comprising the polygonal chains of each slice. These line segments are essentially the only representation of a "feature" in the method presented in this paper. In this manner, it is not important that any set of segments be machinable in any particular setup orientation. The intent is to map the segments or portions of segments machinable from each orientation $\alpha$, in order for a minimum set of orientations to be found such that all segments are machined after all setup orientations are completed.

Each line segment of the slice file is either perpendicular or oblique with respect to the tool approach orientation. If the perpendicular case occurs, the obstacles and their corresponding obstacle spaces for each point on that line segment are the same. This is actually a static-obstacle case. A two-dimensional C-space analysis can be performed for all those points at the same time, considering all the adjacent segments above the segment to be obstacles. Section 3.1 discusses the perpendicular case in more detail. However, if the line segment to be checked is oblique with respect to the tool approach orientation, the obstacles for each point on that segment are variable. This dynamic-obstacle environment creates difficulty for the machinability analysis process. The solution for this case will be presented in Sec. 3.2.

**3.1 Perpendicular Case.** The coordinate system used in this paper is consistent with that of a 3-axis milling center, whereby the slicing of the part occurs along the $x$-axis of the machine coordinate frame and each slice is in the $y$-$z$ plane (Fig. 4($a$)). We use $\overline{P_{i,j}P_{i,j+1}}$ to represent a segment that is currently undergoing the machinability analysis process, where $i$ denotes the number of the slice on which that segment resides, and $j$ and $j+1$ denote the two consecutive points forming that line segment. In the case where the segment $\overline{P_{i,j}P_{i,j+1}}$ is perpendicular to the tool-cutting orientation, all the segments on slice $i$ and its adjacent slices with a portion having a greater height than that of $\overline{P_{i,j}P_{i,j+1}}$ along the tool-cutting orientation are obstacles. The obstacle spaces associated with these obstacles remain unchanged for the analysis of every point on $\overline{P_{i,j}P_{i,j+1}}$. In this situation, the cutting tool is moving in an environment with static obstacles. The problem is simplfied as finding the tool space for segment $\overline{P_{i,j}P_{i,j+1}}$ as a geometric primitive on a two-dimensional plane, instead of analyzing each point separately. Figure 4($a$) shows that there are *left* obstacles and *right* obstacles on each side of segment $\overline{P_{i,j}P_{i,j+1}}$. Obstacles may also exist on the slice where $\overline{P_{i,j}P_{i,j+1}}$ resides. Plane $\pi$ contains segment $\overline{P_{i,j}P_{i,j+1}}$ and is perpendicular to the tool-cutting orientation $\alpha$. Any segment that has a portion above plane $\pi$ is considered an obstacle. Figure 4($b$) is the top view of Fig. 4($a$). The boundary of the maximum tool space and the obstacle spaces are constructed by offsetting segment $\overline{P_{i,j}P_{i,j+1}}$ and obstacle segments by the amount of the tool radius. The curves in the dashed-dotted line denote the boundaries of left obstacle space ($\Sigma_m \mathrm{OS}(L_m, \alpha)$) and right obstacle space ($\Sigma_n \mathrm{OS}(R_n, \alpha)$). The curves in the dashed line denote the boundaries of current slice obstacle space ($\mathrm{OS}(i, \alpha)$) and the closed curve in the solid line denotes the boundary of the maximum tool space (MTS). The actual tool space (TS), represented by the shaded region in Fig. 4($b$), is computed using Eq. (1) as follows:

$$\mathrm{TS} = \mathrm{MTS} - \sum_m \mathrm{OS}(L_m, \alpha) - \sum_n \mathrm{OS}(R_n, \alpha) - \mathrm{OS}(i, \alpha)$$

If tool space results in an empty set, then the segment $\overline{P_{i,j}P_{i,j+1}}$ contains no machinable portion. If tool space exists, then the machinable range (MR) boundary can be obtained by offsetting the boundary of tool space by the amount of the tool radius (Fig. 4($c$)). It should be noted that the tool space may consist of several subregions in practice (the tool space shown in Fig. 4($b$) consists of only one region). Therefore, the machinable range may also be separated into subsections. If the machinable range subsections cover the entire segment, $\overline{P_{i,j}P_{i,j+1}}$, that is, $\overline{P_{i,j}P_{i,j+1}} \subseteq \Sigma_\lambda MR_\lambda$, then segment $\overline{P_{i,j}P_{i,j+1}}$ has complete machinability; otherwise, it only has partial machinability. Figure 4($c$) shows that $\overline{P_{i,j}P_{i,j+1}}$ has partial machinability and the portions outside of the machinable range boundary are nonmachinable portions of segment $\overline{P_{i,j}P_{i,j+1}}$.

### 3.2 Oblique Case.

*3.2.1 Dynamic-Obstacle Environment*. For most cases, the cutting orientation is not perpendicular, but oblique to the line segment being analyzed for machinability. This makes the machinability analysis significantly different from the previously described perpendicular case that only had static obstacles. The fundamental reason is that the static-obstacle machinability analysis approach based on two-dimensional C-space does not work for the oblique case, which is characterized by dynamic obstacles. The maximum tool space for machining each point under the oblique case is invariantly a half-circle arc, which is shown in Fig. 5. However, the obstacle spaces that the points on $\overline{P_{i,j}P_{i,j+1}}$ are subject to are dynamically changing, as the point under analysis is moving along $\overline{P_{i,j}P_{i,j+1}}$. Figure 6 shows segment $\overline{P_{i,j}P_{i,j+1}}$ on slice $i$ and an obstacle segment $\overline{P_{i+n,k}P_{i+n,k+1}}$ on adjacent slice $i+n$. Polygon $P_{i+n,k}P_{i+n,k+1}P'_{i+n,k+1}P'_{i+n,k}$ is the obstacle polygon resulting from segment $P_{i+n,k}P_{i+n,k+1}$ with respect to the cutting orientation $\alpha$. It is clear that the effective obstacles affecting point $P_1$ and $P_2$, denoted by $\mathrm{Ob}_1$ and $\mathrm{Ob}_2$, respectively, are different, even though they are caused by the existence of the same obstacle segment $P_{i+n,k}P_{i+n,k+1}$. Hence, the obstacle spaces associated with

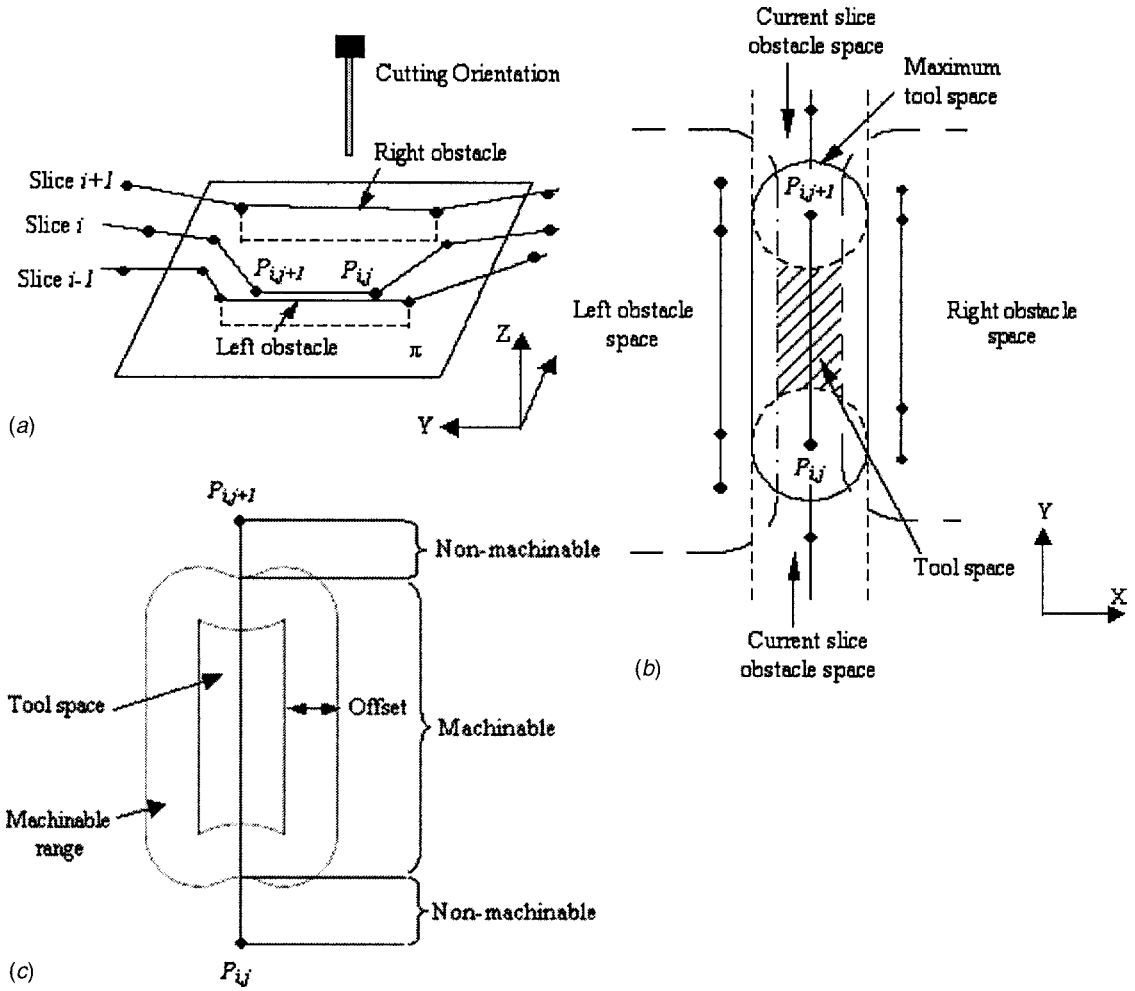



**Fig. 4 Illustration of machinability for perpendicular case**

these two obstacles are also different. The variation of obstacle spaces is because the heights of the points on $\overline{P_{i,j}P_{i,j+1}}$ with respect to the cutting orientation $\alpha$ are different from one another. The obstacle spaces of a certain height are determined by the projections of the obstacle segments above that height on the plane perpendicular to the cutting orientation $\alpha$ at that height. This is actually a problem of three-dimensional C-space. The tool space for $\overline{P_{i,j}P_{i,j+1}}$ could be computed by constructing a three-dimensional C-space. However, since the "part surfaces" being worked on consist of segments from STL slice geometry, there is no information about what kind of feature $\overline{P_{i,j}P_{i,j+1}}$ resides on and/or the local surface description in the vicinity of $\overline{P_{i,j}P_{i,j+1}}$. Therefore, testing machinability by constructing a three-dimensional C-space for each segment is inappropriate.

*3.2.2 Relative Movement of Effective Obstacles*. Although the entire three-dimensional C-space will not be constructed for each line segment, the effective obstacle spaces of each obstacle segment at different heights can be considered to have relative movement with the point under analysis if one sets a convention for machinability analysis. Referring to the example in Fig. 6, if one analyzes the point along the direction traversing the line segment $\overline{P_{i,j}P_{i,j+1}}$, the relative linear movement of the effective obstacles shows three stages where machinability is affected. They include stage 1, where the obstacle space begins to gouge into the maximum tool space (Fig. 7(*a*)) and the gouged tool space increases (Fig. 7(*b*)) until it reaches its maximum gouged tool space arc, $\theta_{mL}\theta_{mU}$ (Fig. 7(*c*)); stage 2, where the obstacle space maintains its maximum gouged tool space arc $\theta_{mL}\theta_{mU}$ (Figs. 7(*c*) and 7(*d*)); and stage 3, where the obstacle space begins to move away from $O_m$, the point being examined. Gouged tool space then begins to decrease (Fig. 7(*e*)) until the obstacle space does not affect machinablity (Fig. 7(*f*)). In Fig. 7, $\Delta$ denotes the slicing spacing and $k\Delta$ is the distance from the slice $i$ to slice $i+k$, where the obstacle segment is located. The variable $d$ is the relative distance of the effective obstacle segment to the segment under analysis. *EF* represents the effective obstacle, point $F$ moves along edge $\overline{P_{i+n,k+1}P_{i+n,k}}$ and $\overline{P_{i+n,k}P'_{i+n,k}}$, and point $E$ moves along edge $\overline{P_{i+n,k+1}P'_{i+n,k+1}}$ of the polygon $P_{i+n,k}P_{i+n,k+1}P'_{i+n,k+1}P'_{i+n,k}$ in Fig. 6.

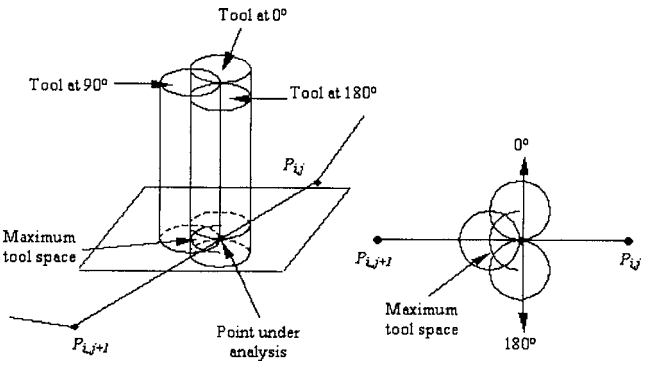



**Fig. 5 Illustration of maximum tool space under oblique cutting**

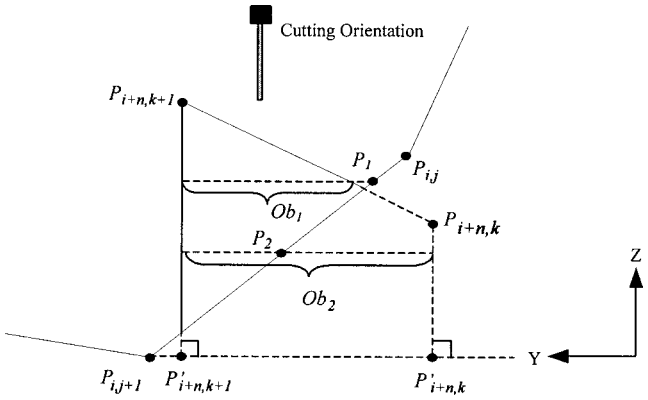



**Fig. 6 Variation of effective obstacles**

The variable $d_s$ denotes the distance from $F$ to $O_m$ along the $y$-axis, at which tool space begins to be gouged by the obstacle space of $EF$; while $d_m$ denotes the distance from $F$ to $O_m$ along the $y$-axis, at which the maximum gouged tool space is reached (Fig. 7($c$)), and also denotes the distance from $E$ to $O_m$ along the $y$-axis, at which gouged tool space begins to reduce (Fig. 7($d$)). From Fig. 7. it can be seen that if there exists a point $p \in EF$ with its distance to $O_m$ along the $y$-axis equal to $d_m$, then the maximum gouged tool space can be attained. The variable $d_e$ denotes the distance from point $E$ to $O_m$ along the $y$-axis, at which the tool space becomes unaffected by the obstacle space of $EF$. Note that each obstacle segment may have all or only one or two of these three typical stages in practice depending on the relative location of the obstacle segment and the segment under analysis. This will be discussed further in Sec. 3.2.3.

For the above case, once the stage the effective obstacle belongs to and the value of $d$ is known, the gouged tool space arc $(\theta_{dL}, \theta_{dU})$ can be computed as follows:

For stage 1 where $d_m < d < d_s$

$$\theta_{dL} = \arccos\left[\frac{k\Delta(k^2\Delta^2 + d^2) + \delta}{2(k^2\Delta^2 + d^2)R}\right]$$

$$\theta_{dU} = \arccos\left[\frac{k\Delta(k^2\Delta^2 + d^2) - \delta}{2(k^2\Delta^2 + d^2)R}\right]$$

For stage 2 where $d = d_m$

$$\theta_{dL} = 0$$

$$\theta_{dU} = \arccos\left(\frac{k\Delta - R}{R}\right)$$

For stage 3 where $d_e < d < d_m$

when $d_e < d < 0$

$$\theta_{dL} = 0$$

$$\theta_{dU} = \arccos\left[\frac{k\Delta(k^2\Delta^2 + d^2) + \delta}{2(k^2\Delta^2 + d^2)R}\right]$$

when $0 \leqslant d < d_m$

$$\theta_{dL} = 0$$

$$\theta_{dU} = \arccos\left[\frac{k\Delta(k^2\Delta^2 + d^2) - \delta}{2(k^2\Delta^2 + d^2)R}\right]$$

$$d_s = \sqrt{(2R)^2 - (k\Delta)^2}$$

$$d_m = \sqrt{R^2 - (k\Delta - R)^2}$$

$$d_e = -\sqrt{R^2 - (k\Delta - R)^2}$$

$$\delta = \sqrt{k^2\Delta^2(k^2\Delta^2 + d^2)^2 - (d^2 + k^2\Delta^2)^3 + 4d^2R^2(d^2 + k^2\Delta^2)}$$

$R$ is the tool radius.

*3.2.3 Characterization of the Relative Movement*. The above discussion demonstrates that the relative movement of the effective obstacle with respect to the point under analysis provides the information necessary to calculate the gouged tool space resulting from the corresponding effective obstacle space, and thus the tool space can be computed. Therefore, if the relative movement of each effective obstacle with respect to the point under analysis can be precisely determined, then machinability can be computed without constructing a three-dimensional C-space.

*3.2.3.1 Geometric transformation*. The relative movement of each effective obstacle with respect to the point under analysis can be precisely determined by performing a geometric transformation to both the segment to be analyzed and the obstacle segment.

Consider an inclined line segment $\overline{P_{i,j}P_{i,j+1}}[P_{i,j}(y_{i,j}, z_{i,j})P_{i,j+1}(y_{i,j+1}, z_{i,j+1})]$ on slice $i$, and suppose that $z_{i,j}$ is greater than $z_{i,j+1}$ $(z_{i,j} > z_{i,j+1})$ (Fig. 8). The parameter $t$ is from the parametric representation of $P_{i,j}P_{i,j+1}$. The parameter $t$ is 0 at the end point with smaller $z$ coordinate and is 1 at the end point with greater $z$ coordinate. For the segment $\overline{P_{i,j}P_{i,j+1}}$ shown in Fig. 8, $t$ is 0 at $P_{i,j+1}$ and is 1 at $P_{i,j}$. Point $E(y_E, z_E)$ is an

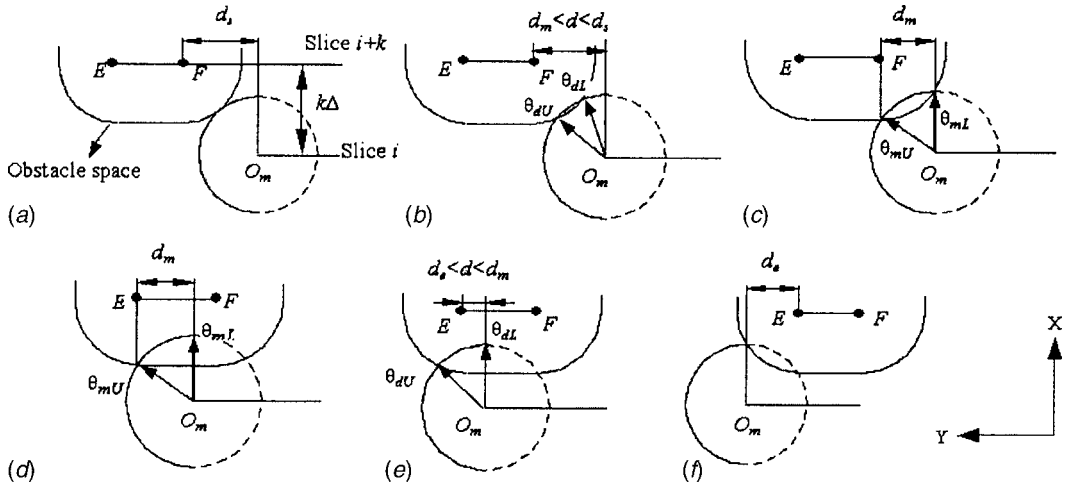



**Fig. 7 Variation of machinable range due to the existance of an obstacle segment**

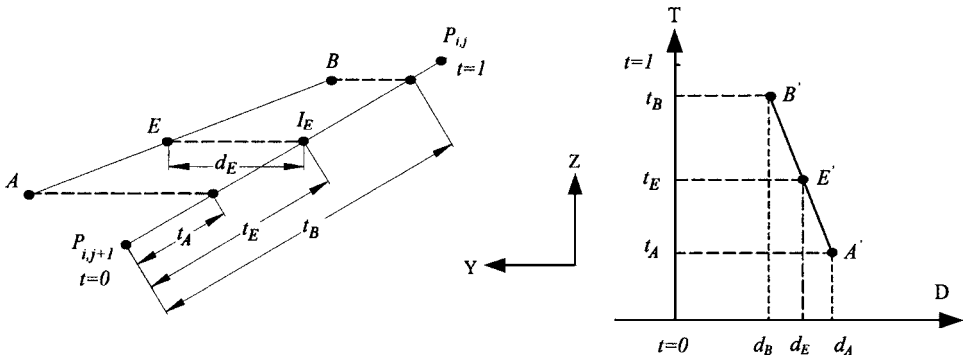



**Fig. 8 Geometric Transformation**

arbitrary point in the plane $y$-$z$ and lies on line $AB$, determined by point $A(y_A,z_A)$ and $B(y_B,z_B)$. Point $E$ can be transformed and mapped uniquely as a point $E'(d_E,t_E)$ in plane $D$-$T$ as shown in Fig. 8. The $d_E$ is the horizontal distance from $E$ to line $\overline{P_{i,j}P_{i,j+1}}$ along the $y$-axis. It is the distance from $E$ to $I_E(y_{iE},z_E)$, the intersection point of line $Z=z_E$ and the line where segment $\overline{P_{i,j}P_{i,j+1}}$ lies. The $t_E$ is the parametric value at the intersection point $I_E$. Similarly, $A$ and $B$ are mapped as $A'$ and $B'$ in plane $D$-$T$. The transformation from a 2D plane $y$-$z$ to the 2D plane $D$-$T$ ($D$-$T$ transformation) can be represented as a map $F$: $R^2 \rightarrow R^2$ of the form

$$F(p) = M \cdot p + n$$

for all point $p \in R^2$, where

$$M = \begin{bmatrix} 1 & \dfrac{y_{i,j+1}-y_{i,j}}{z_{i,j}-z_{i,j+1}} \\ 0 & \dfrac{1}{z_{i,j}-z_{i,j+1}} \end{bmatrix} \quad N = \begin{bmatrix} \dfrac{z_{i,j+1}y_{i,j}-z_{i,j}y_{i,j+1}}{z_{i,j}-z_{i,j+1}} \\ -\dfrac{z_{i,j+1}}{z_{i,j}-z_{i,j+1}} \end{bmatrix}$$

for line segments with

$$z_{i,j+1} < z_{i,j}; \quad M = \begin{bmatrix} -1 & \dfrac{y_{i,j+1}-y_{i,j}}{z_{i,j+1}-z_{i,j}} \\ 0 & \dfrac{1}{z_{i,j+1}-z_{i,j}} \end{bmatrix} \quad N = \begin{bmatrix} \dfrac{z_{i,j+1}y_{i,j}-z_{i,j}y_{i,j+1}}{z_{i,j+1}-z_{i,j}} \\ -\dfrac{z_{i,j}}{z_{i,j+1}-z_{i,j}} \end{bmatrix}$$

for those with $z_{i,j+1} > z_{i,j}$.

*3.2.3.2 Mapping an obstacle polygon on D-T plane*. Based on the $D$-$T$ transformation, an obstacle segment on adjacent slice $i+m$ $Q_{i+m,k}Q_{i+m,k+1}$ and its associated obstacle polygon $Q_{i+m,k}Q_{i+m,k+1}Q'_{i+m,k+1}Q'_{i+m,k}$ (Fig. 9) can be mapped onto $D$-$T$ plane as segment $Q^*_{i+m,k}Q^*_{i+m,k+1}$ and obstacle polygon $Q^*_{i+m,k}Q^*_{i+m,k+1}Q'^*_{i+m,k+1}Q'^*_{i+m,k}$ (Fig. 10). The coordinates of $Q^*_{i+m,k}, Q^*_{i+m,k+1}, Q'^*_{i+m,k+1}, Q'^*_{i+m,k}$ are $(d_{i+m,k},t_{i+m,k})$, $(d_{i+m,k+1},t_{i+m,k+1})$, $(d_{i+m,k+1},0)$, and $(d_{i+m,k},0)$, respectively. Any line segment formed by truncating a horizontal line in the $D$-$T$ plane by the boundary of the obstacle polygon $Q^*_{i+m,k}Q^*_{i+m,k+1}Q'^*_{i+m,k+1}Q'^*_{i+m,k}$ is actually the effective obstacle of $Q_{i+m,k}Q_{i+m,k+1}$ at the height $t$ along the $T$-axis, as shown in Fig. 10. The end point with a smaller $D$ value can be considered as a *front* point and the one with a larger $D$ value can be considered as a *rear* point. The front points and rear points of all the effective segments resulting from $Q_{i+m,k}Q_{i+m,k+1}$ form the front point trajectory and the rear point trajectory. In Fig. 10, the front point trajectory is $Q^*_{i+m,k}Q'^*_{i+m,k}$ and the rear point trajectory is $Q^*_{i+m,k}Q^*_{i+m,k+1}$ and $Q^*_{i+m,k+1}Q'^*_{i+m,k+1}$.

The three stages discussed in Sec. 3.2.2 can also be reflected clearly on the obstacle polygon in plane $D$-$T$, as illustrated in Fig. 10. $\overline{JK}$, along the front point trajectory, truncated by the line $d=d_s$ and $d=d_m$, corresponds to stage 1, $\overline{KG}$, along the line $d=d_m$ truncated by the front point trajectory and the rear point trajectory corresponds to stage 2, and $\overline{GH}$, along the trajectory of the rear point, truncated by line $d=d_m$ and $d=d_e$, corresponds to stage 3. As mentioned previously, not all the obstacle segments possess all of these three stages. How many stages an obstacle segment may have depends on the shape of the obstacle polygon in the $D$-$T$ plane. The mapping of these three stages onto the $D$-$T$ plane can provide a precise calculation for the gouged tool space because, given a $t$ value, the corresponding $d$ value, which denotes the relative distance of the segment under analysis and one obstacle segment, can be easily mapped on plane $D$-$T$ and, therefore, the gouged tool space can be precisely calculated with the $d$ value, using the method described in Sec. 3.2.2.

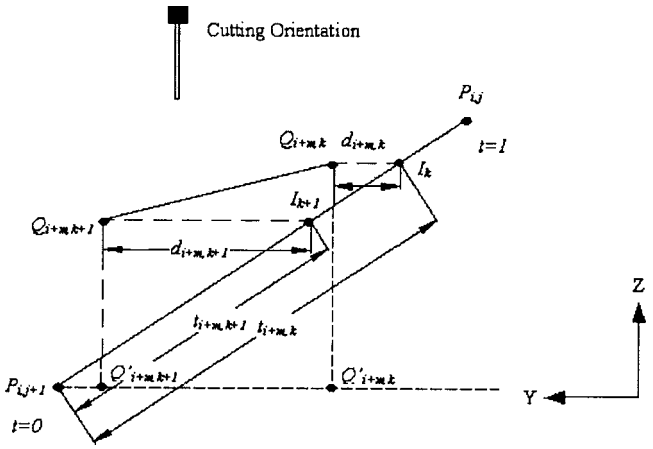



**Fig. 9 An obstacle polygon in *y-z* plane**

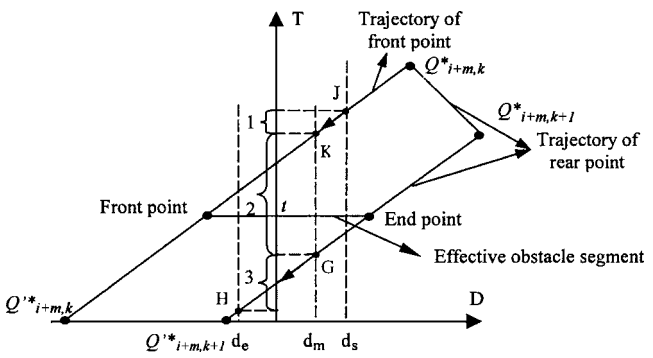



**Fig. 10 An obstacle polygon in *D-T* plane**

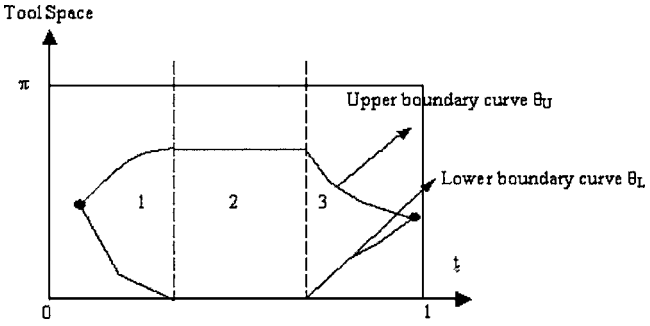



**Fig. 11 Machinability graph**

Thus far, we have shown that $d$ is a function of $t$ and gouged tool space is a function of $d$. Therefore, the gouged tool space can also be considered as a function of $t$. This leads to the construction of a graph of machinability versus the parameter $t$ for each line segment to be examined.

*3.2.4 Machinability Graph*. Figure 11 illustrates the composition of the machinability graph for line segment $\overline{P_{i,j}P_{i,j+1}}$ under the interference of one obstacle $Q_{i+m,k}Q_{i+m,k+1}$. It consists of the same three stages as those shown in Fig. 10 and is bounded by two curves: Upper boundary curve $\theta_U$ and lower boundary curve $\theta_L$. The region bounded within these two curves is the gouged tool space obstructed by the obstacle $Q_{i+m,k}Q_{i+m,k+1}$ and is denoted as $OS_{i+m,k,k+1}$. The rectangular frame shown in Fig. 11 with length 1 and width $\pi$ is the maximum tool space (MTS). Once the gouged tool space for each obstacle segment is obtained, the machinability analysis can be conducted by computing MTS $-\Sigma_n\Sigma_k OS_{i+n,k,k+1}$. If the subtraction operation yields an empty result, then the segment $\overline{P_{i,j}P_{i,j+1}}$ is not machinable with respect to the machining orientation. Otherwise, this segment is at least partially machinable.

Since the upper boundary curve $\theta_U$ and lower boundary curve $\theta_L$ are not of regular shapes, the analytic computation of tool space is not feasible. In this paper, a sweeping line method was used to incrementally check the existence of tool space. The steps to the method are as follows:

**Step 1**. Assign $t_s$ to $t$. ($t_s$ can be either 1 or 0, depending on the incremental direction).

**Step 2**. Check and store the intersections of line $t=t_s$ with the upper and lower boundary curves of each obstacle segment $Q_{i+m,k}Q_{i+m,k+1}$.

**Step 3**. If there is no intersection, then $t=t_s$ is machinable. Go to Step 8. If there are intersections, sort all the intersections in decreasing order into a sequence.

**Step 4**. If the maximum intersection is $<180$ deg, or the minimum intersection is greater than 0 deg, then $t=t_s$ is machinable. Go to Step 8. Otherwise, search the reverse order pair in the intersection points list. The reverse order pair is defined as a lower intersection point (LIP, an intersection with a lower boundary curve) immediately in front of an upper intersection point (UIP, an intersection with an upper boundary curve) in the sorted intersection points sequence.

**Step 5**. If there is no reverse order pair, then $t=t_s$ is nonmachinable. Go to step 8. Otherwise, store the reverse order pairs.

**Step 6**. For each reverse order pair $j$, check the number of net upper intersection points (Num_NUIP) before reverse order pair $j$ (denoted by $<j$) and the number of net lower intersection points (Num_NLIP) after reverse order pair $j$ (denoted by $>j$) in the intersection points sequence.

$$\text{Num\_NUIP}(<j) = \text{number of UIP}(<j) - \text{number of LIP}(<j)$$

$$\text{Num\_NLIP}(>j) = \text{number of LIP}(>j) - \text{number of UIP}(>j)$$

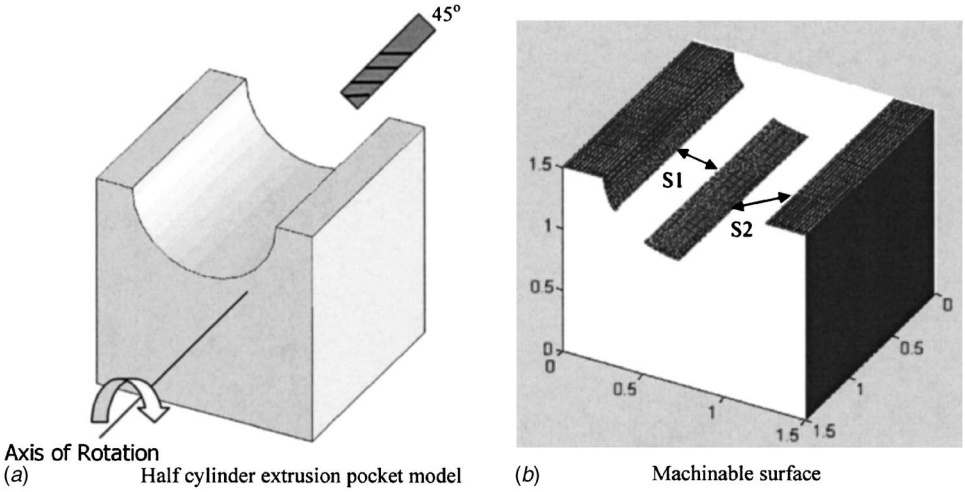
**Step 7**. If Num_NUIP=1 and Num_NLIP=1, then $t=t_s$ is machinable. If Num_NUIP$>1$ and Num_NLIP$>1$, then $t=t_s$ is nonmachinable.

**Step 8**. Update $t$ by an increment and check if stopping criterion is met. If not met, go to Step 1.

## 4 Implementation

To validate the approach proposed in this paper, the machinability algorithms were implemented in the C programing language on a Pentium IV, 3.06 GHz PC running *Windows XP*. The machinability software uses slice-visibility data, the size of the flat-end tool chosen and a specified cutting orientation as inputs. It generates the machinable portion of each slice segment with respect to the cutting orientation as output. We present two example part surfaces to verify the machinability analysis approach. The first part was chosen to be quite simple so that the nonmachinable regions should seem intuitive to the reader. The second part is a more complex part, a toy "jack," and this part is evaluated with 3D inspection software that is used for reverse engineering.

*Example 1*. Figure 12(*a*) shows a block with a half cylindrical extruded cut. We chose the direction of 45 deg on a plane orthogonal to the axis of rotation as the cutting orientation. The flat end tool diameter is set to be 0.25 in. (6.35 mm). The slice spacing is 0.01 in. (0.254 mm). Results from the machinability analy-




(*a*) Half cylinder extrusion pocket model (*b*) Machinable surface

**Fig. 12 Machinability of a half cylinder extrusion pocket**

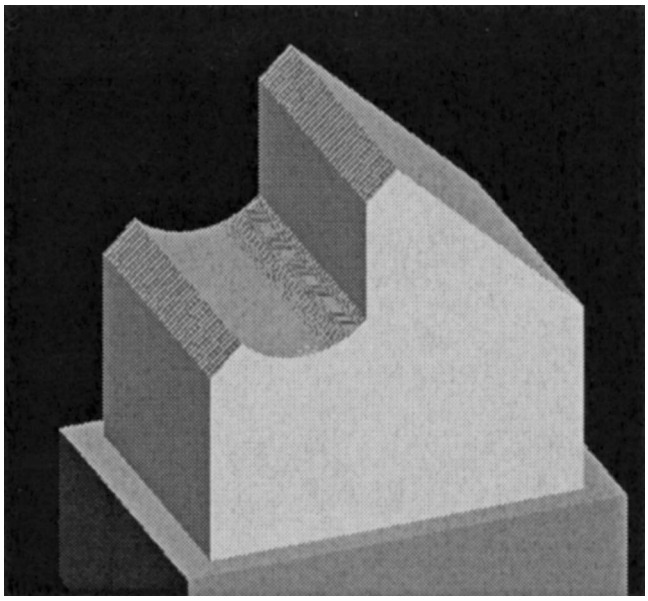


**Fig. 13 2D views of machinable profiles**

sis are displayed in Fig. 12(*b*), which indicates that there are two nonmachinable regions, denoted as S1 and S2. Since this model is an extrusion, the geometric shape along the axis of rotation does not change. Therefore, the results of the machinability analysis of this part should be the same on every slice along the axis of rotation. The machinable profile of one slice is displayed in Fig. 13(*a*). *O-A-B*, *C-D* and *E-F-G* are machinable regions while *B-C* and *D-E* are nonmachinable regions. This part was also virtually machined in Mastercam. A screen shot of the part being virtually machined from a block of material is shown in Fig. 14. The virtually machined volume was saved as an STL file and imported into RapidForm 2004 (reverse engineering software). Figure 13(*b*) shows a cross-sectional profile of this STL by using the inspection function of RapidForm. The orientation of the profile displayed in Fig. 13(*b*) is 45 deg with respect to the horizon, the same direction as the machining setup shown in Fig. 14. $O'$-$A'$-$B'$, $C'$-$D'$ and $E'$-$F'$-$G'$ are the machinable regions, while

**Table 1 Boundary point coordinates of nonmachinable regions**

| Point | Result of machinability analysis (in.) | Result of virtual machining (in.) |
|---|---|---|
| $B,B'$ | $B(0.3173,-0.2481)$ | $B'(0.3170,-0.2506)$ |
| $C,C'$ | $C(0.4973,-0.4294)$ | $C'(0.4956,-0.4291)$ |
| $D,D'$ | $D(0.7500,-0.4983)$ | $D'(0.7513,-0.4982)$ |
| $E,E'$ | $E(1.2500,0.0000)$ | $E'(1.2496,0.0000)$ |

$B'$-$C'$ and $D'$-$E'$ are the nonmachinable regions in Fig. 13(*b*). To verify the results from the machinability analysis, two local coordinate systems are set up on point $O$ ($X$-$O$-$Y$ in Fig. 13(*a*)) and $O'$ ($X'$-$O'$-$Y'$ in Fig. 13(*b*)), respectively. Coordinates of the nonmachinable boundary points $B$, $C$, $D$, and $E$ are computed from the results of the machinability analysis software we developed and then the coordinates of $B'$, $C'$, $D'$, and $E'$ are measured in RapidForm 2004. The coordinates of the points from each approach are shown in Table 1. The data are very close, and the error is within that which can be expected using an STL approximation and a line-sweeping algorithm for finding machinable regions.

*Example 2*. The second part model used as an example is a toy "jack" that is analyzed first for visibility and then for machinability using the approach of this paper. Figure 15(*a*) shows the STL-model of the "jack." The visibility software developed previously [17] first processed this model and gave the result that the model is 100% visible through four orientations {50, 155, 228, 335 deg}. To demonstrate the deficiency of visibility analysis, the "jack" model was machined with these four orientations using a 0.125 in. dia (3.175 mm dia) flat-end tool. Figure 15(*b*) shows the machined "jack" with nonmachinable regions indicated by the four rectangles.

Since the shape of slices of the "jack" model varies along the axis of rotation, the nonmachinable regions change accordingly and, therefore, are not regular shapes. To demonstrate that the machinability analysis approach can predict the nonmachinable regions, we virtually machined the "jack" from a 50 deg orientation in MasterCAM and saved the result as an STL file after virtual machining. The file was imported into RapidForm 2004 and was overlapped with the original CAD model. Using the inspection function of RapidForm, we are able to map the deviations of

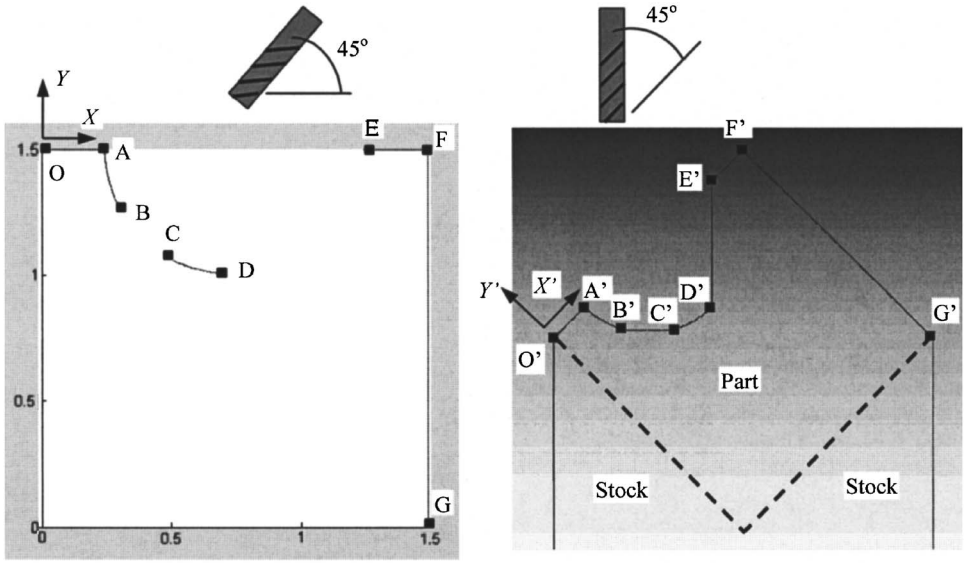



**Fig. 14 Screen shot of example part virtually machined in MasterCAM**

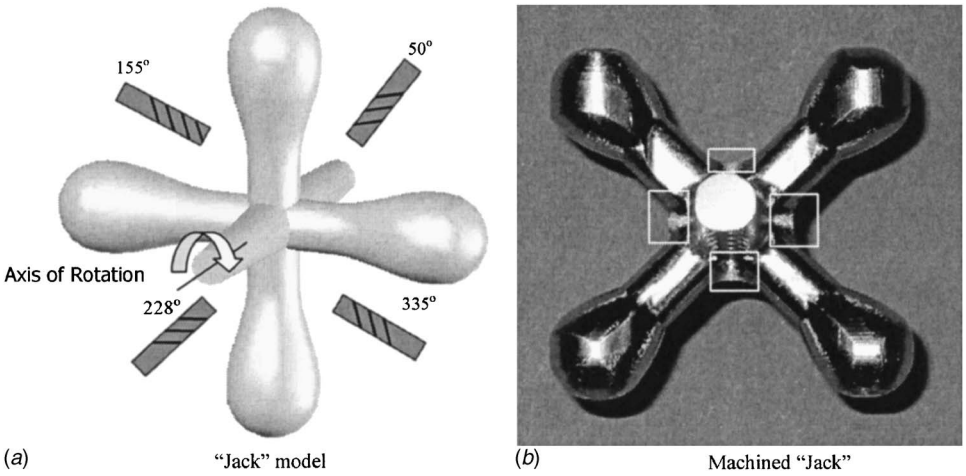



**Fig. 15 Machining result of a "jack" model**

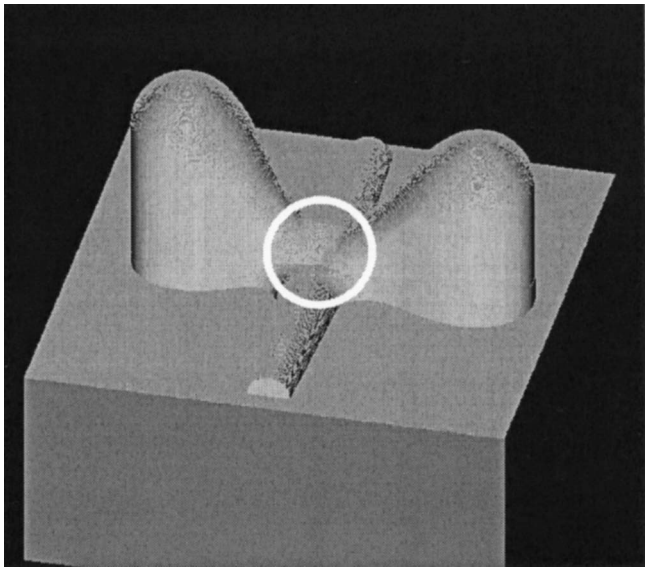


**Fig. 16 Machined "jack" surface in MasterCAM**

the machined STL model to the original part model. Figure 16 illustrates the virtually machined surface (in MasterCAM) by rotating the part model to a 50 deg orientation. The region marked by the circle contains the nonmachinable surfaces. Figure 17(*b*) illustrates the deviation results from RapidForm 2004. The part model is displayed in point shading mode to indicate the surface profile. Figure 17(*a*) illustrates the results from our machinability analysis software, which corresponds well with the graphical display in Fig. 17(*b*).

The above examples indicate that the machinability software not only predicts machinable and nonmachinable regions for a sliced STL model, but also determines the exact coordinate locations of its machinable and nonmachinable portions. Therefore, given a cutting orientation and a sized tool, the precise locational information of the machinable and nonmachinable regions can be obtained, thus, providing a tool for product designers in evaluating the manufacturability of a particular design. Another application could be found in determining the minimum number of setups for machining a part using an indexable 4-axis machine with respect to one rotation axis. This could be realized by first running the machinability software for cutting orientations from 0 up to 360 deg at the interval of a specified incremental angle, and then searching for the minimum number of setup orientations using an optimization algorithm. In this manner, the machinability analysis

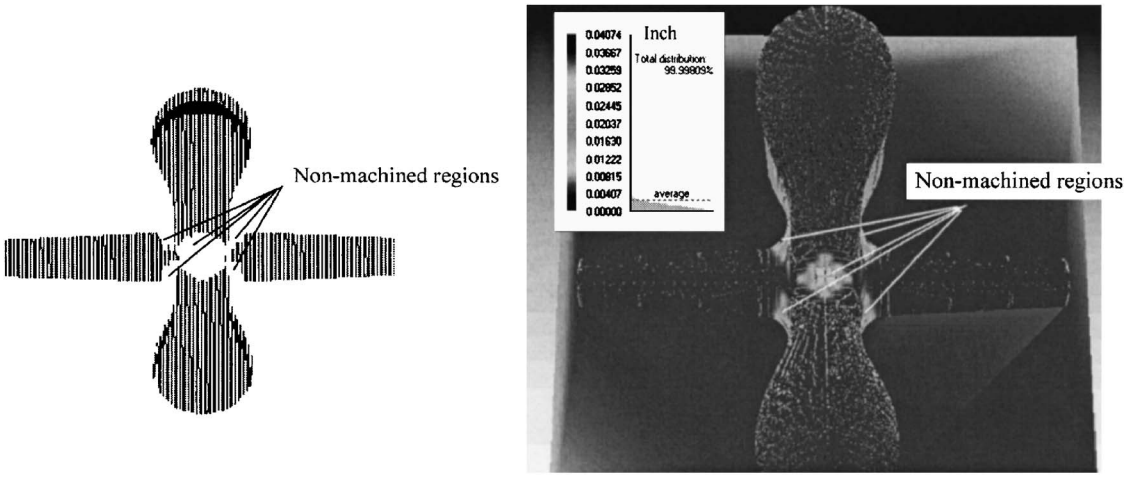



**Fig. 17 Identification of nonmachinable regions for "jack"**

presented in this paper is not relegated to a *verification* tool, rather, the information it provides can be used to calculate machining setup information and for tool-path planning.

## 5 Conclusions and Future Research

This paper presents a machinability analysis approach for 3-axis flat end milling based on the concept of C-space. The approach is intended to find the true accessibility, which is often approximated by the visibility of line of light. The input information is the STL slice file of a geometric model with restriction to one axis of rotation. The model can be either feature based or feature free, as the machinability analysis approach presented in this paper does not require feature recognition. Instead, it analyzes basic geometric primitives, line segments from the STL slice file. In addition to checking 2D machinability, the analysis approach can handle 3D machinability by constructing a machinability graph. The results of this analysis yields the portions of the surface segments machinable from a given orientation and tool. The expectation is that the machinability results can be used to determine the minimum set of setup orientations necessary to machine all surface segments using simple 3-axis milling operations. Currently, our approach is restricted to 3-axis flat end milling. The machinability analysis of ball-end mills and fillet-end mills may be the focus of future research. In addition, how to extend the current research to the application of 4-axis or perhaps 5-axis milling will likely be another consideration.

## References


[1] Suh, S. H., and Lee, J. J., 1998, "Five-Axis Part Machining With Three-Axis CNC Machine and Indexing Table," ASME J. Manuf. Sci. Eng., **120**(1), pp. 120–128.

[2] Suh, S. H., Lee, J. J., and Kim, S. K., 1998, "Multiaxis Machining With Additional Axis NC System: Theory and Development," Int. J. Adv. Manuf. Technol., **14**(12), pp. 865–875.

[3] Frank, M. C., Wysk, R. A., and Joshi, S. B., 2004, "Rapid Planning for CNC Machining—A New Approach to Rapid Prototyping," J. Manuf. Syst., **23**(3), pp. 242–255.

[4] Vickers, G. W., and Quan, K. W., 1989, "Ball-Mills Versus End-Mills for Curved Surface Machining," ASME J. Eng. Ind., **111**, pp. 22–26.

[5] Ip, W. L. R., and Loftus, M., 1993, "The Application of An Inclined End Mill Machining Strategy on 3-Axis Machining Centers," Int. J. Mach. Tools Manuf., **33**(2), pp. 115–133.

[6] Su, C. J., and Mukerjee, A., 1991, "Automated Machinability Checking for CAD/CAM," IEEE Trans. Rob. Autom., **7**(5), pp. 691–699.

[7] Chen, L. L., and Woo, T. C., 1992, "Computational Geometry on the Sphere With Application to Automated Machining," J. Mech. Des., **114**, pp. 288–295.

[8] Tang, K., Woo, T., and Gan, J., 1992, "Maximum Intersection of Spherical Polygons and Workpiece Orientation for 4- and 5-Axis Machining," J. Mech. Des., **114**, pp. 477–485.

[9] Gan, J. G., Woo, T. C., and Tang, K., 1994, "Spherical Maps: Their Construction, Properties, and Approximation," J. Mech. Des., **116**, pp. 357–363.

[10] Chen, L. L., Chou, S. Y., and Woo, T. C., 1993, "Separating and Intersecting Spherical Polygons: Computing Machinability on Three-, Four-, and Five-Axis Numerically Controlled Machines," ACM Trans. Graphics, **12**(4), pp. 305–326.

[11] Yang, W., Ding, H., and Xiong, Y., 1999, "Manufacturability Analysis for A Sculpture Surface Using Visibility Cone Computation," Int. J. Adv. Manuf. Technol., **15**(5), pp. 317–321.

[12] Yin, Z. P., Ding, H., and Xiong, Y. L., 2000, "Visibility Theory and Algorithms With Application to Manufacturing Processes," Int. J. Prod. Res., **38**(13), pp. 2891–2909.

[13] Gan, J. G., 1990, "Spherical Algorithms for Setup Orientations of Workpiece With Sulptured Surfaces," Ph.D. thesis, University of Michigan, Ann Arbor, MI.

[14] Suh, S. H., and Kang, J. K., 1995, "Process Planning for Multi-axis NC Machining of Free Surfaces," Int. J. Prod. Res., **33**(10), pp. 2723–2738.

[15] Dhaliwal, S., Gupta, S. K., Huang, J., and Priyadarshi, A., 2003, "Algorithms for Computing Global Accessibility Cones," ASME J. Comput. Inf. Sci. Eng., **3**(3), pp. 200–209.

[16] Balasubramaniam, M., Laxmiprasad, P., Sarma, S., and Shaikh, Z., 2000, "Generating 5-Axis NC Roughing Paths Directly From A Tessellated Representation," Comput.-Aided Des., **32**(4), pp. 261–277.

[17] Frank, M. C., Wysk, R. A., and Joshi, S. B., 2005, "Determining Setup Orientations From the Visibility of Slice Geometry for Rapid CNC Machining," ASME J. Manuf. Sci. Eng., **128**, pp. 228–238.

[18] Haghpassand, K., and Oliver, J. H., 1995, "Computational Geometry for Optimal Workpiece Orientation," J. Mech. Des., **117**(2A), pp. 329–335.

[19] Radzevich, S. P., and Goodman, E. D., 2002, "Computation of Optimal Workpiece Orientation for Multi-Axis NC Machining of Sculptured Part Surfaces," J. Mech. Des., **124**(2), pp. 201–212.

[20] Balasubramaniam, M., Sarma, S. E., and Marciniak, K., 2003, "Collision-Free Finishing Toolpaths From Visibility Data," Comput.-Aided Des., **35**(4), pp. 359–374.

[21] Regli, W. C., 1995, "Geometric Algorithms for the Recognition of Features From Solid Models," Ph.D. thesis, The University of Maryland, College Park, MD.

[22] Regli, W. C., Gupta, S. K., and Nau, D. S., 1995, "Extracting Alternative Machining Features: An Algorithmic Approach," Res. Eng. Des., **7**(3), pp. 173–192.

[23] Gupta, S. K., and Nau, D. S., 1995, "Systematic Approach to Analyzing the Manufacturability of Machined Parts," Comput.-Aided Des., **27**(5), pp. 323–342.

[24] Gupta, S. K., Regli, W. C., Das, D., and Nau, D. S., 1997, "Automated Manufacturability Analysis: A Survey," Res. Eng. Des., **9**(3), pp. 168–190.

[25] Shen, Y., and Shah, J. J., 1998, "Recognition of Machining Features Based on HSPCE Decomposition, Feature Composition, and Process Centered Classification," J. Mech. Des., **120**(4), pp. 668–677.

[26] Gaines, D. M., Castano, F., and Hayes, C. C., 1999, "MEDIATOR: A Resource Adaptive Feature Recognizer that Intertwines Feature Extraction and Manufacturing Analysis," J. Mech. Des., **121**(1), pp. 145–158.

[27] Faraj, I., 2003, "Manufacturing Features: Verification Interaction Accessibility and Machinability," Int. J. Prod. Res., **41**(10), pp. 2249–2272.

[28] Ferreira, P. M., and Liu, C. R., 1988, "Generation of Workpiece Orientations for Machining Using A Rule-Based System," Rob. Comput.-Integr. Manufact., **4**(3/4), pp. 545–555.

[29] Demey, S., Brussel, H. V., and Derache, H., 1996, "Determining Set-ups for Mechanical Workpieces," Rob. Comput.-Integr. Manufact., **12**(2), pp. 195–205.

[30] Wu, H. C., and Chang, T. C., 1998, "Automated Setup Selection in Feature-Based Process Planning," Int. J. Prod. Res., **36**(3), pp. 695–712.

[31] Ong, S. K., Ding, J., and Nee, A. Y. C., 2002, "Hybrid GA and SA Dynamic Set-up Planning Optimization," Int. J. Prod. Res., **40**(18), pp. 4697–4719.

[32] Lozano-Perez, T., 1981, "Automatic Planning of Manipulator Transfer Movements," IEEE Trans. Syst. Man Cybern., **SMC-11**(10), pp. 681–698.

[33] Choi, B. K., Kim, D. H., and Jerard, R. B., 1997, "C-space Approach to Tool-Path Generation for Die and Mound Machining," Comput.-Aided Des., **29**(9), pp. 657–669.

[34] Choi, B. K., and Ko, K., 2003, "C-Space Based CAPP Algorithm for Freeform Die-Cavity Machining," Comput.-Aided Des., **35**(2), pp. 179–189.

[35] Morishige, K., Takeuchi, Y., and Kase, K., 1999, "Tool Path Generation Using C-Space for 5-Axis Control Machining," ASME J. Manuf. Sci. Eng., **121**(1), pp. 144–149.

[36] Jun, C. S., Cha, K. D., and Lee, Y. S., 2003, "Optimizing Tool Orientations for 5-Axis Machining by Configuration-Space Search Method," Comput.-Aided Des., **35**(6), pp. 549–566.