

# UC Berkeley

## Precision Manufacturing Group

### Title

Geometric Approaches for Reducing Burr Formation in Planar Milling by Avoiding Tool Exits

### Permalink

<https://escholarship.org/uc/item/8x62k046>

### Authors

Chu, Chih-Hsing  
Dornfeld, David

### Publication Date

2008-09-12

Peer reviewed

# Geometric Approaches for Reducing Burr Formation in Planar Milling by Avoiding Tool Exits

**Chih-Hsing Chu**, Dept. of Industrial Engineering and Engineering Management, National Tsing-Hua University, Hsinchu, Taiwan

**David Dornfeld**, Laboratory for Manufacturing Automation, Dept. of Mechanical Engineering, University of California at Berkeley, Berkeley, California, USA

## Abstract

One of the most effective methods for reducing milling exit burrs is to prevent the tool from exiting the workpiece. *Exit* here refers the condition in which a cutting edge is moving out of the workpiece while removing material. Only entrance burrs can occur under this circumstance, which are usually considered burr-free. This study proposes a set of geometric algorithms for avoiding tool exits in planar milling. Two distinct approaches are developed for tool path planning of 2-D polygons. The first approach generates exit-free tool paths by offsetting the workpiece edges with appropriate widths of cut. The second one adjusts tool positions locally on given tool paths. In addition, a two-stage algorithm is designed for 2-D free-form contours. The cutter locations causing the tool to exit the workpiece are first detected; then a heuristic scheme is applied to generate new cutter locations with no tool exits. Experimental results show that edge quality is significantly improved using the proposed methods. This work provides a feasible way for suppressing burr formation in an automatic manner, and thus reduces the need for deburring.

**Keywords:** *Edge Finishing, Tool Exit, Geometric Algorithm, Burr Formation, Planar Milling*

## Introduction

Machining is one of the most common manufacturing processes used in industry; however, burrs often occur along workpiece edges during machining. The existence of burrs may reduce the fit and ease of assembly, jeopardize the safety of workers during handling, or cause product malfunction in operation; hence, burrs must be removed. Traditionally, a second finishing operation, known as deburring, is often used to assure that the edges produced meet tolerance specifications. There are substantial costs associated with the deburring operation. In addition, deburring is difficult to automate, and thus may become a bottleneck in a production line (Gillespie 1999). On certain occasions, deburring

simply cannot be employed, such as in micro-machining (Damazo et al. 1999) and precision manufacturing. Burrs, together with chips, have been among the most troublesome obstacles to high productivity and automation of machining processes (Gillespie and Blotter 1976). Burr technology is not simple, contrary to common beliefs that burr problems are not serious (Gillespie 1981).

Planar milling is generally utilized to remove a layer of material on workpiece surfaces for roughness specifications or dimensional requirements. For example, automobile components, such as engine blocks and housings, are first made by casting processes, with shapes roughly created. Then, planar milling operations are performed on certain faces that fit with other parts or serve as dimension datum in assembly. Exit burrs are commonly encountered in planar milling. To enable the elimination of a deburring step, milling burrs have to be reduced to an acceptable amount during primary cutting operations. Burr formation is determined by several factors, including material properties, tool engagement conditions, and cutting parameters. Previous experiments (Pekelharing 1978; Olvera and Barrow 1996, 1998) have shown that, for a specific material with certain ductility, geometric factors dominate burr formation. This observation has been obtained for a wide range of materials. Because exit burrs only occur at tool exits, an effective approach for reducing burr formation is to prevent the tool from exiting the workpiece during machining. When the tool always enters the workpiece, only entrance burrs—usually small—form and the edge is usually considered to be burr-free. Exit burrs will not occur when there are only tool entrances in a milling operation.

Previous studies (Narayanaswami and Dornfeld 1997; Chu 1999; Hassamontr 1998; Chu and Dornfeld 2000; Jayaram, Cramer, and Mathrubutham 1999) were focused on geometric approaches to reducing and preventing milling burr formation. Narayanaswami and Dornfeld (1997) proposed an algorithm that adjusts workpiece orientation for reducing deburring cost in face milling. The in-plane exit angle (Pekelharing 1978) has been used to predict the occurrence of the primary burr. Chu (1999) estimated the total length of the primary exit burr locations in the planar milling of 2-D polygons. The total burr length is used as a criterion for selecting tool path orientation to minimize exit burrs. Hassamontr (1998) adjusted widths of cut in window-framing tool path planning for eliminating tool exits, but his algorithm is not applicable to curved workpieces or parts with internal contours. Chu and Dornfeld (2000) developed a systematic procedure for avoiding tool exits by adjusting tool positions around the workpiece vertices. This procedure does not consider curved contours either. In practice, curved contours and internal features are common. Jayaram, Cramer, and Mathrubutham (1999) developed a CAD module for prediction of burr formation for machining process planning.

Tool path planning serves as an effective approach for improvement of machining quality with different aspects (Kramer 1992; Sarma 1999; Tarng and Chang 1993). Similarly, this study develops a set of geometric algorithms for avoiding tool exits in planar milling to overcome the deficiencies in previous work. First, the previous work (Hassamontr 1998) is extended for arbitrary 2-D polygons with circular edges and internal contours. Appropriate widths of cut are computed that allow the tool to always enter the part while traversing the workpiece edges. Exit-free tool paths are thus generated in a global manner. Second, given an existing tool path, algorithms are developed for automatic generation of special tool paths around the workpiece vertices. Only the tool positions in which the tool exits the workpiece are adjusted. A two-stage algorithm is developed for 2-D free-form contours. Tool exits are first identified on a given tool path. A heuristic method is then applied to move the cutter to a safe position with no tool exits. Note that the proposed methods are only applicable to flat-end cutters without ball nose.

This work serves as a special tool path planner for edge-precision process planning in a network-based

CAD/CAM system (Wright and Dornfeld 1998). Test examples are cut using both the proposed methods and the regular face-milling operation. Experimental results show that the edge quality is significantly improved with only entrance burr formation. The potential of reducing exit burr formation through CAD/CAM integration is successfully demonstrated. This work provides a feasible approach for burr formation control in addition to deburring.

## Tool Exits in Planar Milling

Tool entry and exit conditions are directly related to burr formation in the milling operation. *Exit* here refers specifically to the tool cutting edges moving out of the workpiece at an edge while removing material. *Entry* here refers specifically to the tool cutting edges moving into the workpiece at an edge while removing material. Tool exit is a necessary condition for exit burr formation. To produce exit burrs, this condition must be satisfied; otherwise, merely entrance burrs can occur. This fact has served as a basis for burr minimization tool path generation (Chu and Dornfeld 2000). *Figure 1* schematically shows the exit burr formation on a machined surface in planar milling.

Tool exits only occur in certain circumstances for a 2-D polygonal contour. The authors' previous work (Chu and Dornfeld 2000) described three distinct tool exit conditions, that is, (1) tool entering the workpiece, (2) tool moving along an edge, and (3) tool encountering an adjacent edge, as shown in *Figure 2*. This figure shows the top view of the tool moving on the machined surface. Corresponding methods have been developed for each tool exit condition. Tool exits only occur at one point when the tool enters the workpiece along a circular arc with the tool radius centered at the point  $p$  (see *Figure*

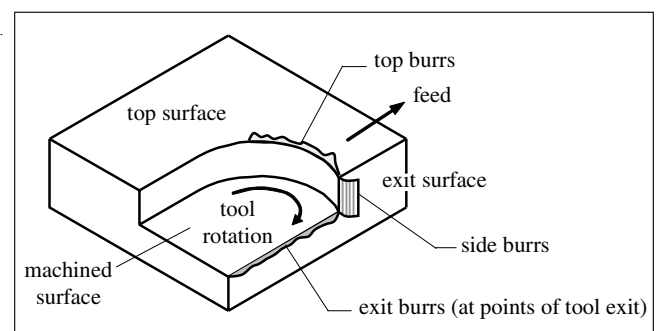


Figure 1  
Burr Produced in Planar Milling

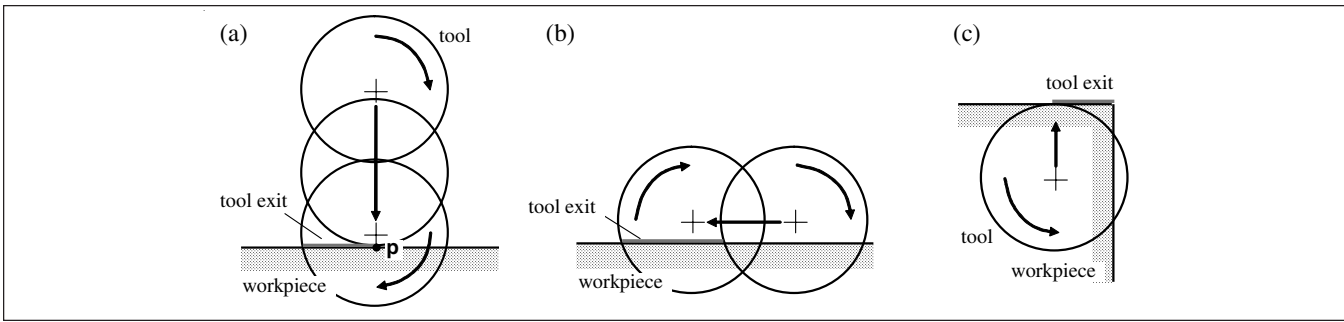


Figure 2  
 Three Tool Exit Conditions

(a) Tool enters workpiece along a straight line, (b) Tool moves along an edge, (c) Tool encounters an adjacent edge

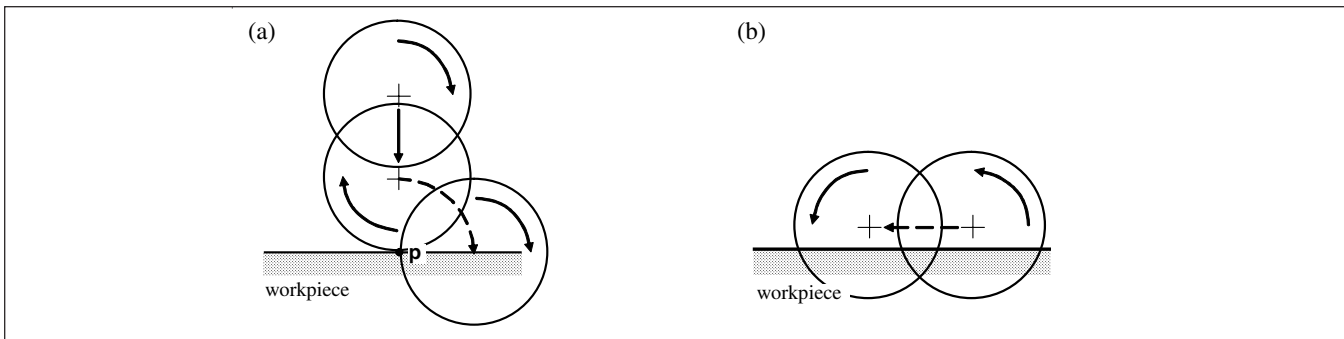


Figure 3

(a) Tool enters workpiece along a circular arc without tool exits,  
 (b) Down milling does not cause tool exits

3a). After entering the workpiece, the tool starts removing workpiece material along an edge. Down milling will not cause tool exits when the tool is traversing along the edge, as shown in *Figure 3b*. The third tool exit condition is more complex and thus is not completely solved. This paper is mainly focused on avoiding tool exits when the tool approaches an adjacent edge.

### Algorithms for Avoiding Tool Exits

Two distinct approaches are provided to prevent the tool from exiting the workpiece during machining. The first approach is to select appropriate widths of cut and corresponding offset distances, resulting in a global adjustment of tool paths. The second approach first detects when tool exits start to occur on an existing tool path. A special tool path that leads to only tool entry is then generated by locally modifying the cutter locations.

A set of five geometric algorithms is developed. They are designed for avoiding tool exits in various circumstances: (1) 2-D polygons by adjusting widths of cut, (2) 2-D circular edges by adjusting widths of cut, (3) 2-D polygons by adjusting tool

positions, (4) 2-D circular edges by adjusting tool positions, and (5) 2-D free-form contours. One major limitation is that the algorithms may not apply for instances where either short edges or non-neighboring edges in geometric proximity occurs. Other assumptions made to facilitate describing these algorithms include:

1. The tool always rotates clockwise unless specifically stated.
2. External profiles consist of a set of edges linked counterclockwise; internal profiles consist of a set of edges linked clockwise.
3. To offset an edge outward, the material side indicates a positive offset distance value, and the one toward the material side shows a negative value.
4. The arrow in front of each tool path indicates the tool feed direction.
5. WP represents workpiece, and TP represents tool path in all illustrations.
6. WP locates at the material side of the workpiece.

In planar milling, tool paths are generally obtained by offsetting the workpiece edges. Thus, the width depth of cut,  $w$ , normally a constant value

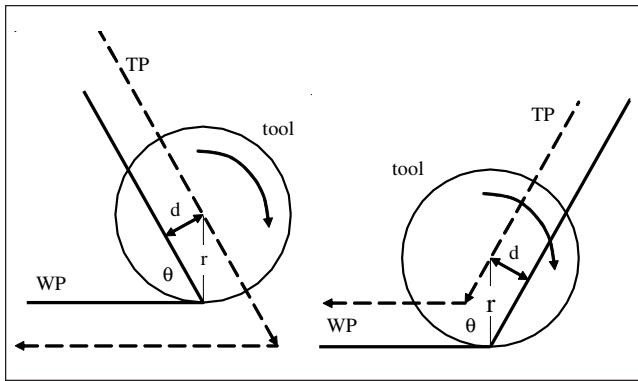


Figure 4  
 Tool Exits for 2-D Convex Polygonal Profiles

when the tool moves along an edge, determines the offset distance,  $d$ :

$$w = r - d \quad (1)$$

where  $r$  is the tool radius.

**Algorithm 1—  
 Avoiding Tool Exits for 2-D Polygons by  
 Adjusting Widths of Cut**

**Convex Contours**

Figure 4 shows the tool position in which the tool exit occurs exactly at the vertex shared with two straight edges. In this case, the offset distance is written as

$$d = r \cdot \cos \theta \quad (2)$$

where  $\theta$  is the extended angle between these two edges. The above equation applies for both acute and obtuse angles. Tool exits will start to form when the tool moves toward the workpiece from the critical position shown in the figure. Therefore, the offset distance range without tool exits is:

$$[d + e, r - e] \quad (3)$$

where  $e$  is a user-specified safety distance accounting for tool position deviations (for example, cutter run-out). Note that the offset distance cannot be larger than  $r$ ; otherwise, no material removal occurs during the machining.

**Concave Contours**

A similar analysis can be conducted for concave polygonal profiles. The offset distance that starts to cause tool exit at the vertex is expressed as:

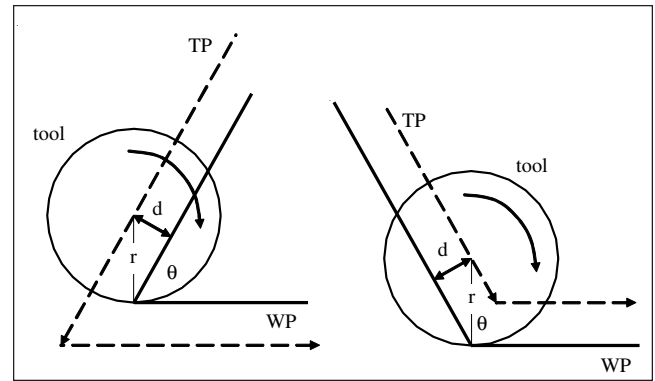


Figure 5  
 Tool Exits for Concave Polygonal Profiles

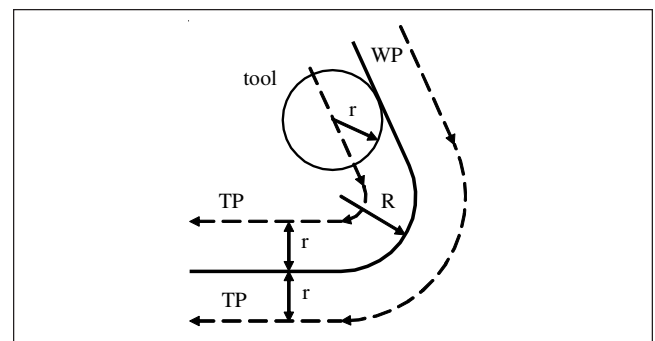


Figure 6  
 Tool Exits Do Not Occur When Tool Radius is Smaller Than Edge Radius

$$d = r \cdot \cos \theta \quad (4)$$

Notice that in Figure 5 the definition of  $\theta$  is not at the material side. The exit-free offset distances become:

$$[-r + e, -d - e] \quad (5)$$

**Algorithm 2—  
 Avoiding Tool Exits for 2-D Circular  
 Edges by Adjusting Widths of Cut**

**Convex Contours**

Tool exits do not appear when the tool radius is smaller than the radius of the circular edge that is,  $r < R$ , as shown in Figure 6. Hence, there are no limitations for the offset distance:

$$[-r + e, r - e] \quad (6)$$

Figure 7 shows the situation when tool exit starts to occur at the vertex shared with a circular edge and its adjacent straight edges. When the extended

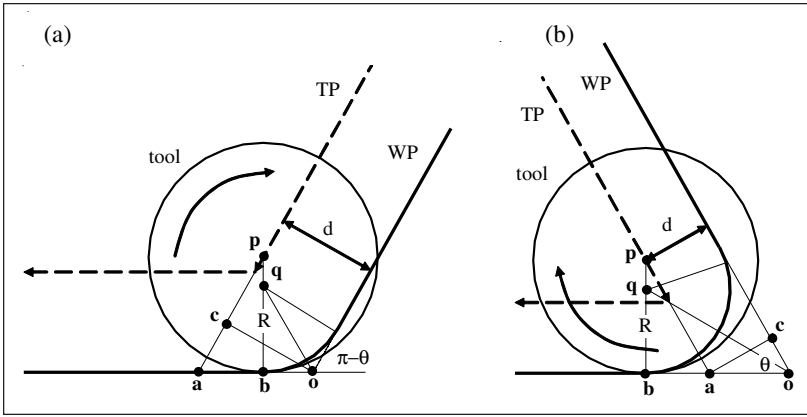


Figure 7  
 Tool Exits for 2-D Convex Profiles with Arcs

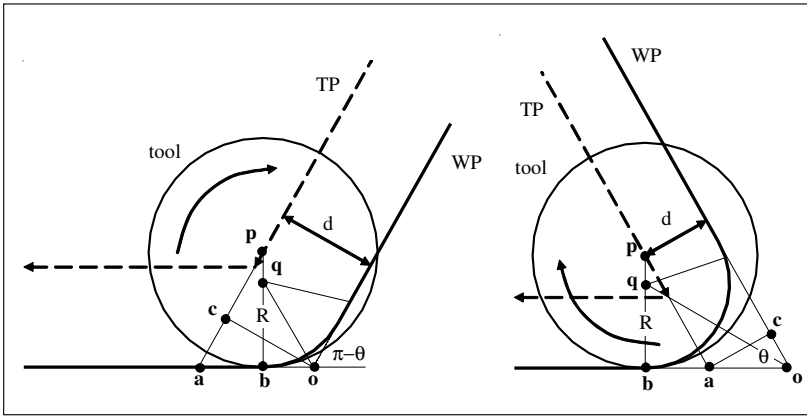


Figure 8  
 Tool Exits for Concave Profiles with Circular Arcs

angle between the two straight edges is obtuse (see Figure 7a), the line segment  $ao$  can be written as

$$oa = ab + bo$$

that is,

$$\frac{oc}{\sin(\pi - \theta)} = \frac{pb}{\tan(\pi - \theta)} + \frac{qb}{\tan(\theta/2)}$$

$$\frac{d}{\sin \theta} = \frac{-r}{\tan \theta} + \frac{R}{\tan(\theta/2)}$$

Expanding the above equation using trigometric relationships, the following is obtained:

$$d = 2R \cdot \cos^2(\theta/2) - r \cdot \cos \theta \quad (7)$$

The same derivation can be carried out for the acute case shown in Figure 7b:

$$ob = oa + ab$$

$$\frac{qb}{\tan(\theta/2)} = \frac{ac}{\sin \theta} + \frac{pb}{\tan \theta}$$

$$R/\tan(\theta/2) = d/\sin \theta + r/\tan \theta$$

The same expression,  $d = 2R \cdot \cos^2(\theta/2) - r \cdot \cos \theta$ , was obtained. Therefore, for the convex contours, the offset distance range without tool exits is:

$$[-d + e, r - e] \quad (8)$$

### Concave Contours

For the concave contours shown in Figure 8, the expression for computing  $d$  remains the same, but the offset distance range becomes:

$$[-r + e, d - e] \quad (9)$$

### Algorithm 3— Avoiding Tool Exits for 2-D Polygons by Adjusting Tool Positions

1. Assume a tool path is obtained by offsetting a workpiece edge with a constant distance. For each tool path, calculate the cutter location  $p_i$  at which the tool tangentially contacts the workpiece edge. The distance  $p_i p_i$  on the tool path is computed as (see Figure 9):

$$\angle p_i p_i q = \angle aoc = \theta, \text{ and } p_i c = p_i q + qc$$

that is,

$$r = p_i p_i \cdot \sin \theta + (-d_2)$$

Note that  $d_2$  has a negative value in the case shown in Figure 9 (offsetting toward material, see assumption 3). Therefore,

$$p_i p_i = \frac{r + d_2}{\sin \theta} \quad (10)$$

Assume the contacting point is  $c$ . Figure 10a shows that the tool will exit the next edge

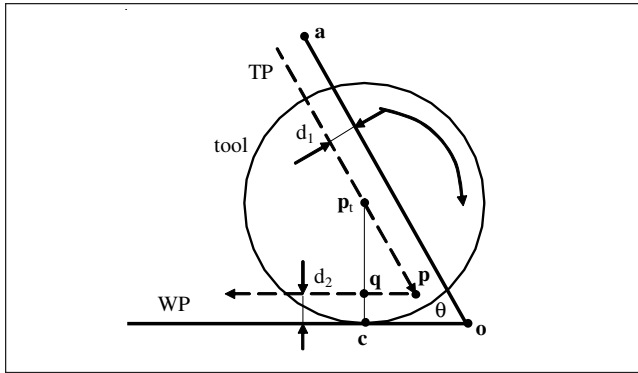


Figure 9

Tool Exit Point on Original Tool Path for 2-D Polygonal Profiles

along  $co$  on the current path, and exit burrs will thus form.

2. Find the *tool approaching boundary*, defined as the boundary beyond which tool exits take place. A straight adjacent edge is the approaching boundary, that is,  $co$ .
3. Tool moves along the direction  $d_e$  of the approaching boundary until it reaches a safe cutter location  $p_s$ . They are expressed as

$$d_e = \frac{co}{|co|}; \quad p_s = p_t + d_e \cdot |co|$$

In other words, the tool will move a distance  $|co|$  along  $d_e$ . It will merely contact the vertex  $o$  at the new cutter location.

4. The tool will move along a circular path  $p_s p_i$  and go past the second edge. Note that  $p_i$  is the intersection point between this circular path and the extension of the second edge.
5. Tool goes back to the original tool path, namely  $p_i p$ .

The resulting exit-free tool path around the vertex  $o$  consists of  $p_t \rightarrow p_s \rightarrow p_i \rightarrow p$ , in which  $p_s \rightarrow p_i$  is an arc centered at  $o$  with tool radius (see Figure 10b).

#### Algorithm 4— Avoiding Tool Exits for 2-D Circular Edges by Adjusting Tool Positions

Algorithm 3 can be modified for contours with circular edges. The procedures are described as follows:

1. Assume a tool path is obtained by offsetting a workpiece edge with a constant distance. For each tool path, calculate the cutter location  $p_t$  at which the tool tangentially contacts the workpiece edge. The tool starts to exit the next edge from the contacting point  $c$ , and exit burrs will thus form along  $co_2$ . Suppose the extensions of the two straight edges intersect at  $o$ . The distance  $p_t p$  on the tool path is computed as follows (see Figure 11):

$$p_t p = p_t e - p e = \frac{p_t c}{\sin(\pi - \theta)} - \frac{p h}{\sin(\pi - \theta)}$$

where  $\angle p e h = \pi - \theta$ . The above expression equals:

$$p_t p = \frac{r + d_2}{\sin \theta} \quad (11)$$

Note that  $d_2$  has a negative value in this case.

2. Find the *tool approaching boundary*, defined as the boundary beyond which tool exits take place. The second straight edge of the circular edge is the approaching boundary, that is,  $o_2 c$  (see Figure 12a).
3. Tool moves along the direction  $d_e$  of the approaching boundary until it reaches a safe cutter location  $p_s$ , expressed as

$$d_e = \frac{co_2}{|co_2|}; \quad p_s = p_t + d_e \cdot |co_2|$$

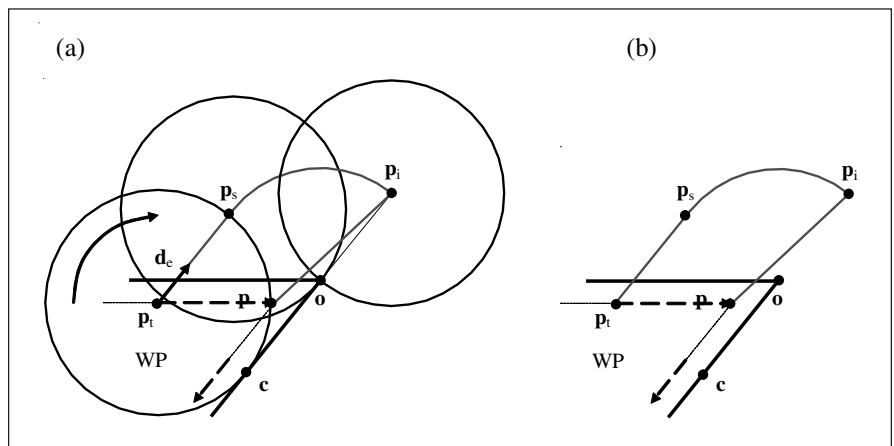


Figure 10

Avoid Tool Exits Around Vertex Shared with Two Straight Edges

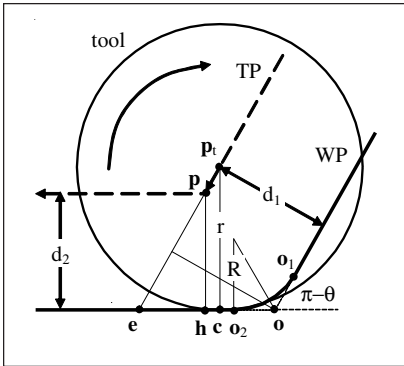


Figure 11

Tool Exit Point on a Given Tool Path for 2-D Contours with Arcs

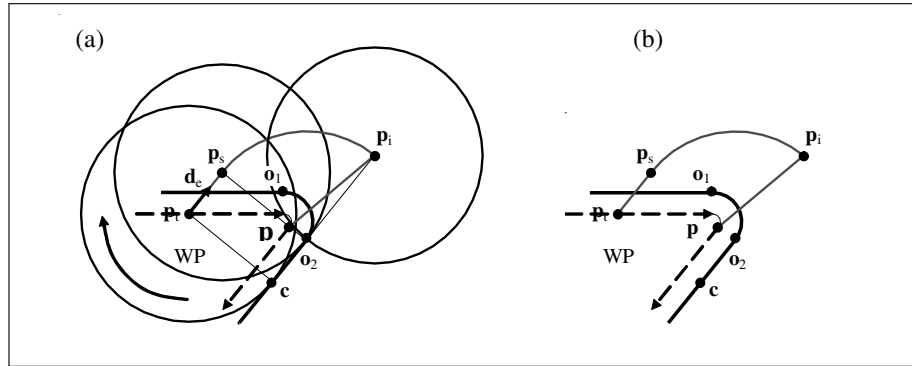


Figure 12

Avoid Tool Exits Around Vertex Shared with Circular Edge and Straight Edge

In other words, the tool will move a distance  $|co_2|$  along  $d_c$ . It will contact the vertex  $o_2$  at the new cutter location.

4. The tool will move along a circular path  $p_s p_i$  and go past the second edge. Note that  $p_i$  is the intersection point between this circular path and the extension of the second edge.
5. Tool goes back to the original tool path, namely  $p_i p$ .

The resulting exit-free tool path around the vertex  $o$  consists of  $p_t \rightarrow p_s \rightarrow p_i \rightarrow p$ , in which  $p_s \rightarrow p_i$  is an arc centered at  $o_2$  with tool radius  $r$  (see Figure 12b).

Although Algorithms 3 and 4 were derived for convex contours, they are applicable to concave ones, too. The only difference is that Eq. (10) becomes

$$p_t p = \frac{r - d_2}{\sin \theta} \quad (12)$$

In addition, the procedures described can be used for both conditions in which  $\theta$  is acute and obtuse.

### Algorithm 5— Avoiding Tool Exits for a 2-D Free-Form Contour

The algorithms previously proposed may not be applicable to 2-D free-form contours. First, there exist no explicit solutions for predicting the occurrence of tool exits, as the workpiece vertices or edges cannot be unambiguously defined. Moreover, the tool approaching boundary is difficult to compute. For this reason, distinct methods are needed for generating new tool positions without tool exit. A novel,

two-stage algorithm is developed for this purpose: it first detects tool positions that will induce tool exits along a given tool path. With this information, including the locations as well as the extents of tool exits, new tool positions can be automatically computed using a heuristic scheme. Figure 13 contains a flow chart of the algorithm. The detailed procedures are explained as follows:

#### 1. Determine tool entry position

- (1) Discretize the contour to obtain a set of points,  $P_i = \{p_i\}$ , according to a specified normal chordal deviation error,  $e_1$ . These points are recorded in the sequence of increasing curve parameter values in  $P_i$ .
- (2) For each point, make a circle centered at  $p_i$  with radius  $2r$  ( $r$  is the tool radius); then find the total number of points  $N_i$  in  $P_i$  within the circle that are not the adjacent points of  $p_i$ , that is,  $p_{i-1}$  or  $p_{i+1}$ .
- (3) Tool entry point  $t_i$  can be any point with  $N_i = 0$ .

#### 2. Generate tool entry path

- (1) Find the unit normal vector  $n_i$  at  $t_i$  on the free-form contour; the tool plunging position is then expressed as:

$$c_p = t_i + (r + d_{s1}) n_i$$

where  $d_{s1}$  is a safety distance to the workpiece boundary.

- (2) The tool moves along the  $-n_i$  direction until it tangentially contacts the workpiece at  $t_i$ . Assume the tool center position is  $c_i$  at this point.



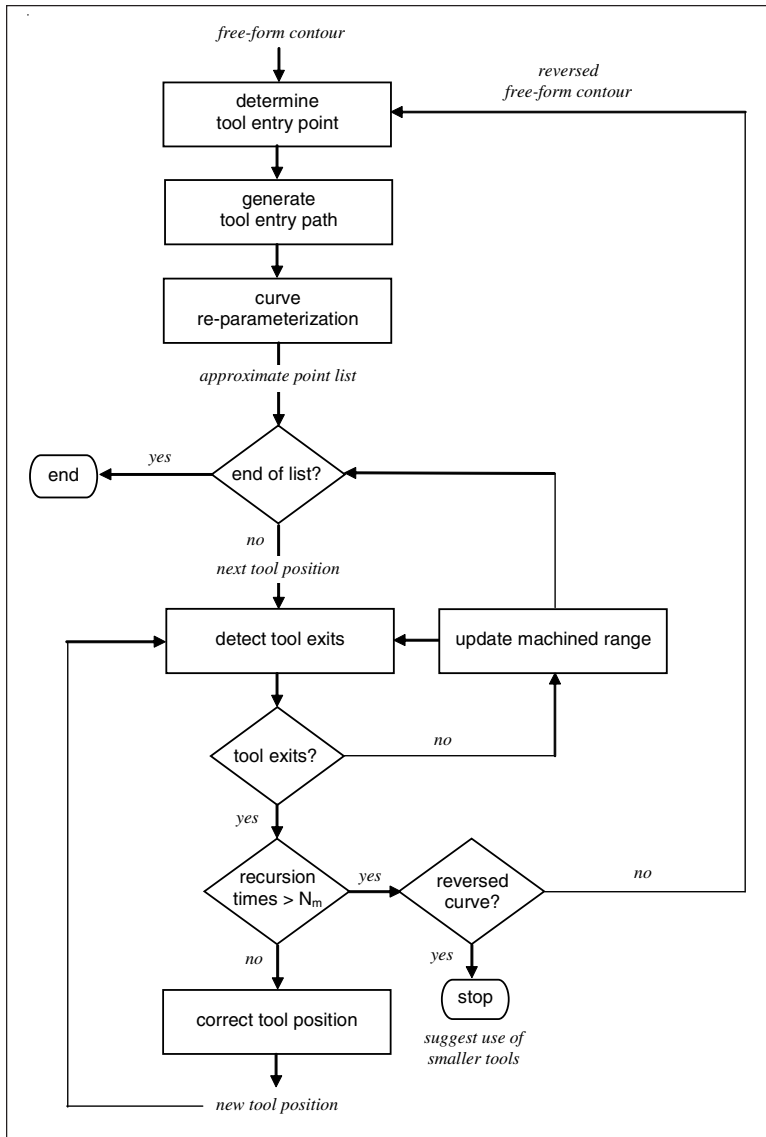


Figure 13

Flow Chart of Algorithm for Avoiding Tool Exits in a Free-Form Contour

- (3) The tool goes along a circular path until the tool center point reaches the workpiece boundary. Centered at  $c_i$  (see Figure 14), the tool intersects the contour at two points: the exit point  $t_i$  and the entry point  $i_s$ , respectively, when the tool rotates clockwise. Note that the tool exits the workpiece only at  $t_i$  when it is moving along the circular path.

### 3. Re-parameterization

- (1) Re-parameterize the contour so that its free-form representation starts and ends at  $t_i$  (assume the contour is described by one single, closed, free-form curve).

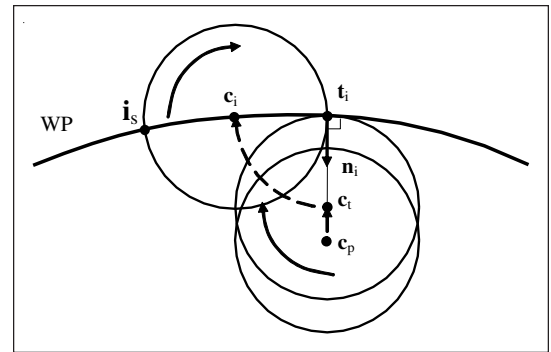


Figure 14

Tool Entry Path for a Free-Form Contour

- (2) The corresponding parameter values at  $t_i$  and  $i_s$  are 0 and  $u_s$ , respectively. Initialize the machined parameter range with  $[0, u_s]$ . The curve segment generated by parameter range  $[0, u_s]$  has been machined. An important fact is that exit burrs will not form within this curve segment, as the material along this edge segment has been removed. For this reason, tool exits occurring in this machined region are not taken into account.
- (3) Discretize the new free-form curve to a set of new points,  $P_i' = \{p_i'\}$ . Two criteria are used in this discretization process. First, the chordal deviation error between any two consecutive points is smaller than a specified chordal error,  $e_2$ . And, the Cartesian distance between any two consecutive points is smaller than a specified value,  $d_{p1}$ .

### 4. Detect tool exits

- (1) Assume the machined parameter range is represented as  $[0, u_m]$ . When the tool is at  $c_i$ ,  $u_m = u_s$  (see Figure 14).
- (2) For each point  $p_i'$ , make a circle centered at the point with radius  $r$ . Compute the intersections between the circle and the original free-form contour. Assume the number of intersection curves is  $M_i$ . If  $M_i > 2$ , arrange the sequence of the intersection curves according to increasing parameter values, that is, the first intersection curve has smaller parameter values than the second one, and so on.

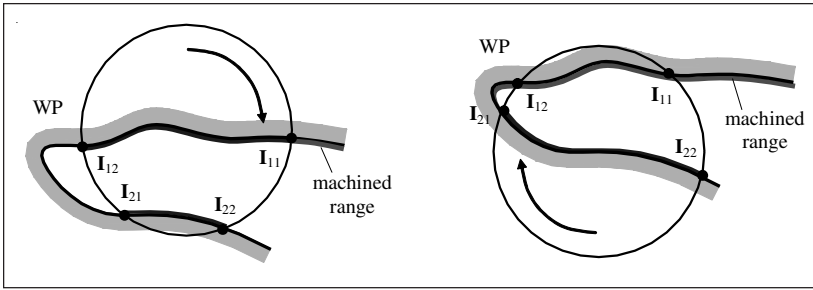


Figure 15  
 Check Tool Does Not Exit in Machined Range

- (3) Assume each intersection curve has two ending points,  $I_{i1}$  and  $I_{i2}$ , with corresponding parameter values,  $u_{i1}$  and  $u_{i2}$ , where  $u_{i2} > u_{i1}$  (see Figure 15). Suppose that tool exits do not occur within the first intersection curve. Starting from the second intersection curve, find the curves with the parameter value range that is not within the machined parameter range  $[0, u_m]$ . Assume  $N_i$  is the total number of the curves satisfying the above condition.
- (4) If  $N_i = 0$ , update the machined parameter range with  $[0, u_{i2}]$ ; otherwise, ( $N_i > 0$ ) and the tool exits the workpiece at  $\mathbf{p}_i'$ .

### 5. Correct tool positions

- (1) For each intersection curve with tool exits, discretize each curve segment into a set of points according to a specified approximate error,  $e_3$ , and add the set of points into  $\{\mathbf{S}\}$ . Thus,  $\{\mathbf{S}\} = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_n\}$  contain all the approximating points generated from these curves.
- (2) The *tool escaping direction*,  $\mathbf{D}_e$ , is the direction along which the tool moves out of tool exits in the workpiece, as shown in Figure 16.  $\mathbf{D}_e$  is the sum of each connecting vector from  $\mathbf{s}_i$  to the tool center point,  $\mathbf{p}_i'$ . The unit direction vector  $\mathbf{d}_e$  is defined as:

$$\mathbf{D}_e = \sum_1^n \mathbf{p}_i' \mathbf{s}_i; \quad \mathbf{d}_e = \frac{\mathbf{D}_e}{|\mathbf{D}_e|}$$

- (3) The *tool escaping distance*  $D_e$  is the distance for the tool to move along  $\mathbf{d}_e$  until it reaches a safe tool position. For each tool exit point  $\mathbf{s}_i$ , the problem can be formulated as (see Figure 17): given a fixed point

$\mathbf{s}_i$  and a line through  $\mathbf{p}_i'$  with a direction vector  $\mathbf{d}_e$ , find a position  $\mathbf{p}_i''$  in the line at which the distance to  $\mathbf{s}_i$  is equal to the tool radius  $r$ . The moving distance  $D_i$  along  $\mathbf{d}_e$  is  $\mathbf{p}_i' \mathbf{p}_i''$ . Hence the tool escaping distance at tool position  $\mathbf{p}_i'$  is the maximal distance among all the  $D_i$ 's, that is,  $D_e = \max(D_1, D_2, \dots, D_n)$ . A safety distance can be added into the escaping distance so that

$$D_e = \max(D_1, D_2, \dots, D_n) + d_{s2}$$

- (4) The candidate safe position with no tool exits is written as:

$$\mathbf{p}_s = \mathbf{p}_i' + D_e \mathbf{d}_e$$

However, the tool at this new position may exit the workpiece at points other than  $\{\mathbf{S}\}$ . Hence, tool exit detection must be recursively applied on new tool positions generated. This is indicated by the back arrow from "new tool position" in the flow chart.

### 6. Reverse parameterization direction

- (1) The correction scheme described may fail when the free-form contour is highly complex (see Figure 18a for example). There-

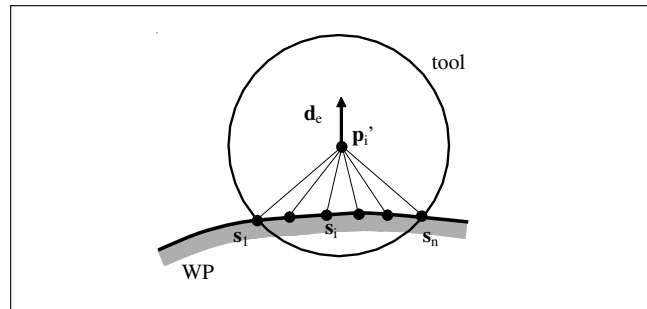


Figure 16  
 Compute Tool Escaping Direction from Tool Exits

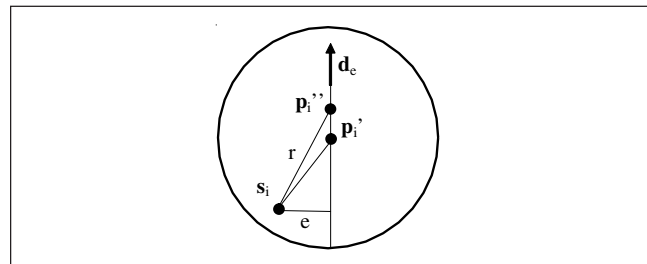


Figure 17  
 Computing Tool Escaping Distance for One Tool Exit

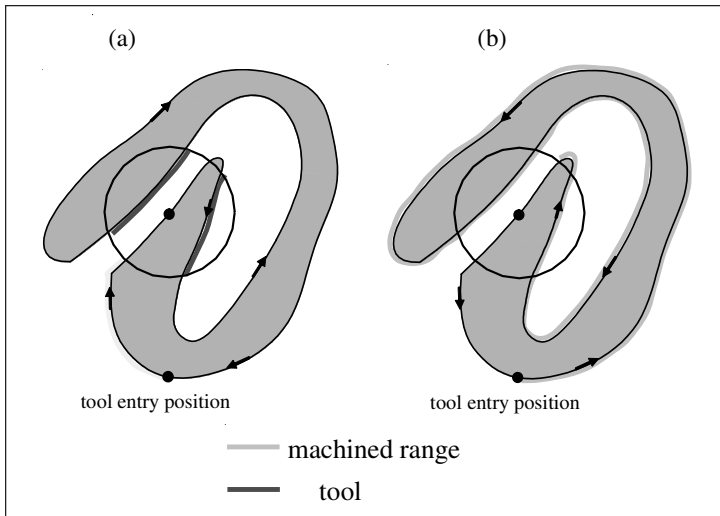


Figure 18

- (a) Complex case in which tool exits cannot be avoided,  
 (b) Reversing tool path direction helps avoid tool exits

fore, a specified recursion time,  $N_r$ , is used as a criterion in this algorithm to stop the program. Should this case happen, it is considered that the tool cannot be prevented from exiting the workpiece. Nevertheless, to change the tool entry point or to reverse the traversing direction of tool path may help solve the problem, as shown in *Figure 18b*.

- (2) When tool exits cannot be avoided, the free-form curve will be re-parameterized. The tool will traverse and cut the contour in the reverse direction. Certainly, the tool rotation must also be reverted. The entire algorithm is then applied again. If tool exits still occur, the program will terminate and suggest the use of a smaller tool.

This algorithm is not designed for any particular free-form scheme, such as Bézier, B-spline, or NURBS. Instead, it is applicable to curves in parametric representations. In addition, the detection-correction method can be used for both internal as well as external free-form contours.

### Test Results

This work has been implemented in C++ using ACIS (Spatial 1995) as a geometry engine. It has served as a special tool path planner in a networked CAD/CAM system (Dornfeld et al. 1999). The user can specify the contours in a 3-D CAD model for planar milling, the tool geometry used, the entry workpiece edge, and the depth of cut. NC code will be automatically generated for a three-axis milling machine.

*Figure 19* shows a test example consisting of straight and circular edges. Both an internal contour and an external contour are to be cut in planar milling. An offset range that results in exit-free tool paths is computed for each edge using the corresponding algorithms, depending on the edge types (linear or circular). Generally this offset extent varies from edge to edge. The overlap range for all the edges is calculated. The tool will not exit the workpiece on the tool paths generated by offsetting the edges with a distance within this overlap range. *Figure 19a* shows the tool paths generated with a 12.7 mm (0.5 in.) diameter cutter. The arrow along each tool path represents the tool feed direction. Second, given the workpiece edges as original tool paths, special tool paths are generated for avoiding tool exits around the vertices. *Figure 19b* shows the tool paths com-

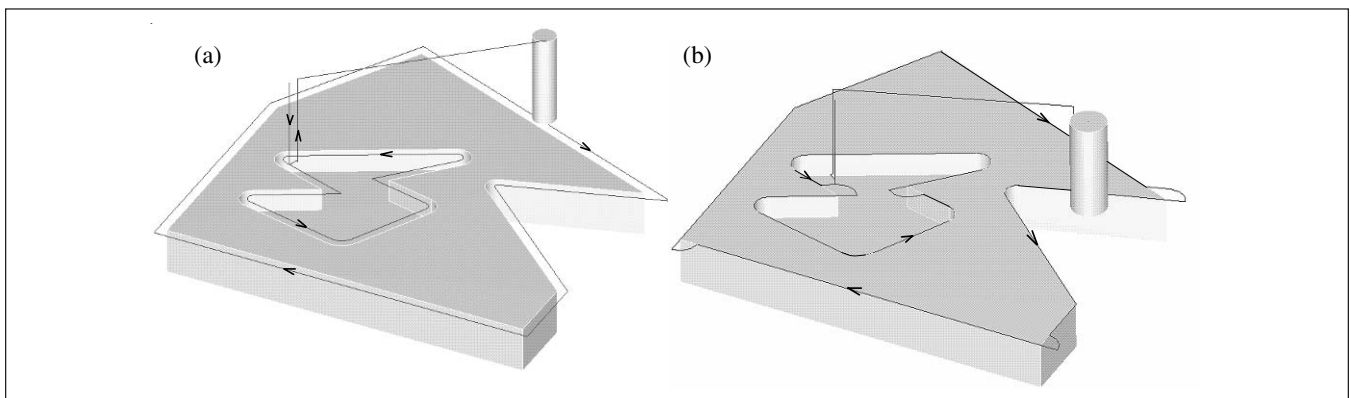


Figure 19

Test Example Consisting of Straight and Circular Edges

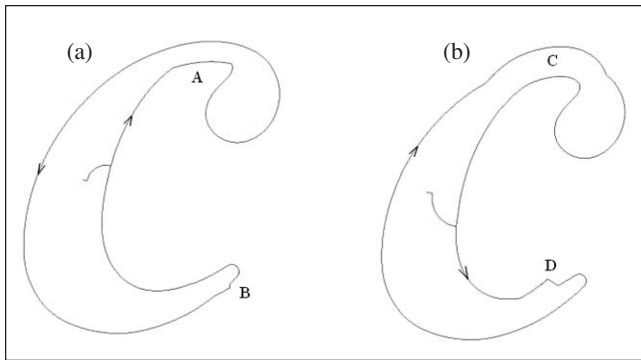


Figure 20  
 Regular and Special Tool Paths for C-Shaped Test Part

puted from Algorithms 3 and 4 with a 19.05 mm (0.75 in.) diameter cutter. Note that additional tool paths are needed for removal of workpiece material away from the edges.

A second test part with a C-shaped pocket is used to test the algorithm for free-form machining. Given the contour as the original tool path, *Figure 20a* shows the corrected tool path using a 12.7 mm (0.5 in.) diameter tool. Tool exits occur around the narrow regions, denoted as A and B in this figure. Original tool positions near these regions must be adjusted so that the tool does not cut the workpiece material on the other side, or cause tool exits. In the second case, shown in *Figure 20b*, the tool diameter is increased to 19.05 mm (0.75 in.). Certainly, the tool still exits the part near narrow regions (C and D). The extent of tool exits becomes so large that they cannot be avoided when the tool rotates clockwise (refer to the step 6 of this algorithm). As a result, the traversing direction along the tool path is reversed; the whole procedures are then applied again. The bump-shape curves in region C and D indicate the corrected tool positions. Note that both the tool feed and rotational direction have been reversed.

Cutting experiments are conducted on Al 6061-T6 to compare the edge quality using the regular and the special tool paths. Here “regular” refers to zig-zag or unidirectional tool paths that normally are used in planar milling. A C-shaped part has been cut prior to the tests. Then a layer of material is to be removed in the milling experiments. A 76.2 mm (3 in.) diameter face milling cutter is used in the first cut. The second cut consists of two sets of tool paths. A first set of tool paths, as shown in *Figure 20a*, is generated for removing the material along the workpiece edges. The remaining material away from the edges is then removed using a second set of zig-

Table 1  
 Cutting Conditions for Free-Form Machining

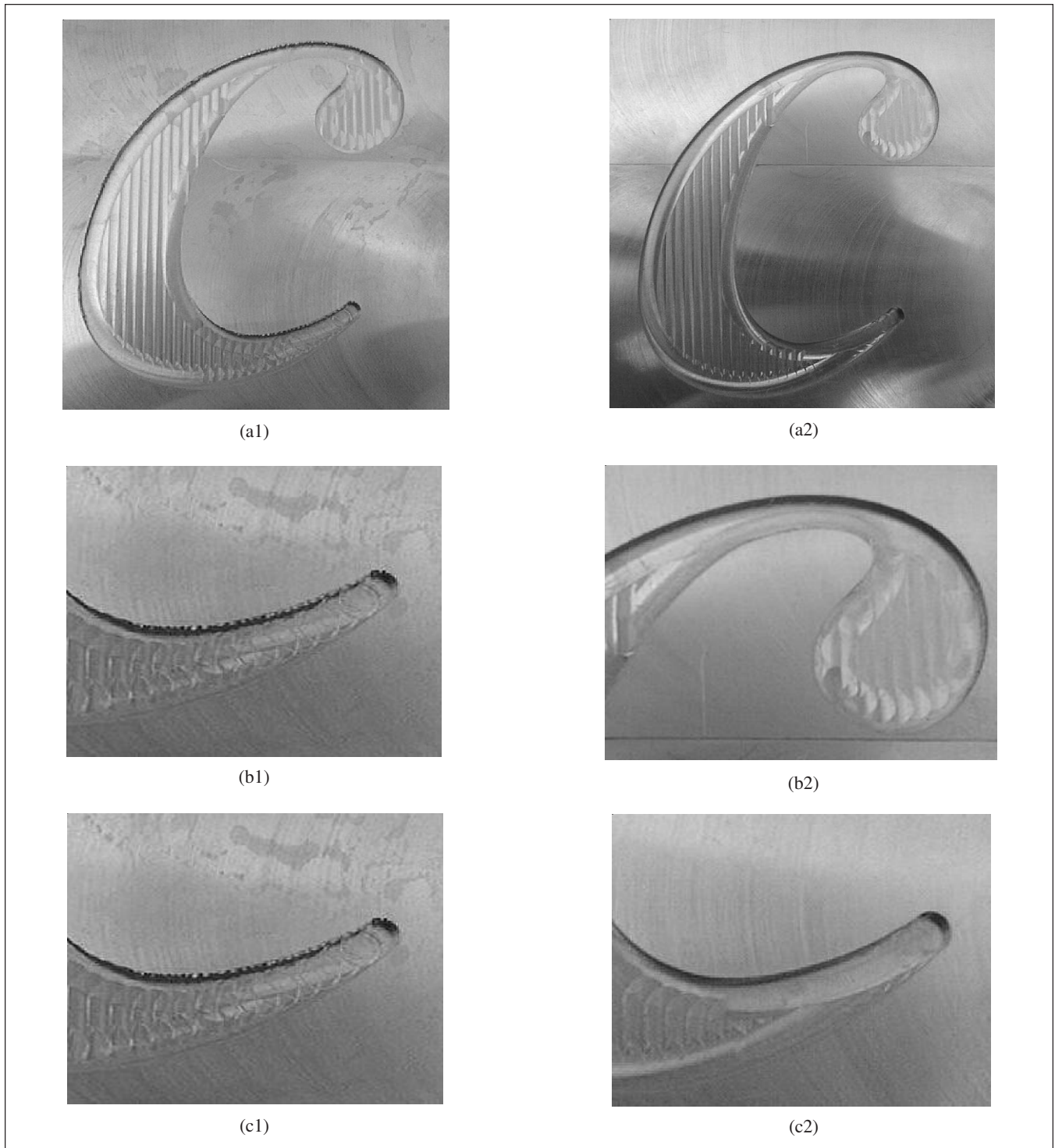
	First	Second
Tool type	Valenite V40CT-S100-20 insert; coated carbide	HSS, two-flute end milling cutter
Tool diameter	76.2 mm (3 in.)	12.7 mm (0.5 in.)
Feed per tooth	0.05 mm	0.05 mm
Depth of cut	2.032 mm	2.032 mm
Rotational speed	636 rpm	2292 rpm

zag tool paths. The selection of cutting speed and feed is based on the *Machining Data Handbook* (TechSolve 1980). *Table 1* lists the cutting parameters for both experiments. Note that the feed per tooth and the depth of cut remain the same in both cuts. In this case, the difference in edge quality can be compared. Previous studies (Pekelharing 1978; Park 1996) have shown that these two factors dominate exit burr formation under normal cutting conditions.

*Figures 21(a1)* and *21(a2)* show the machined part after the first and second cuts, respectively. Exit burrs are readily recognized in *Figure 21(a1)*. *Figure 21(b1)* shows an enlarged view of region A. The irregular protrusions along the edge represents the wavy burr formation (Park 1996). In contrast, the second part has better edge quality with no discernible burr formation, as shown in *Figure 21(b2)*. The enlarged region B indicates similar results (see *Figures 21(c1)* and *21(c2)*). In addition, burr size has been measured using an optical microscope. *Figure 22* shows the captured images under the same magnification factor for both parts (the same location in region A). The burr size in the part cut using regular tool paths is on the order of 1 mm. A deburring operation is required for removing a burr of such size. On the other hand, the burr size is much smaller in the second part, as only entrance burrs occur. From the experimental results, it is concluded that the edge quality is significantly improved by avoiding tool exits in planar milling. The geometric approaches proposed in this paper effectively achieve this.

## Discussion

A better solution than secondary finishing operations is to stop burr formation during primary cutting operations, or to reduce burr size to an acceptable amount. This study has demonstrated such an in situ burr reduction approach. The special tool paths remove the material along the workpiece edges



**Figure 21**  
(a) Machined parts after experiments, (b) Comparison of edge qualities in region A for both parts,  
(c) Comparison of edge qualities in region B for both parts

prior to regular zig-zag tool paths to complete surface generation. As a result, burrs cannot form while the workpiece material away from the edges is being cut. Due to the additional edge-cleaning tool

paths, the total machining time increases. However, if the cost of deburring is significant, the additional machining time may be economically justified (Wright et al. 2000). This in situ burr reduction ap-

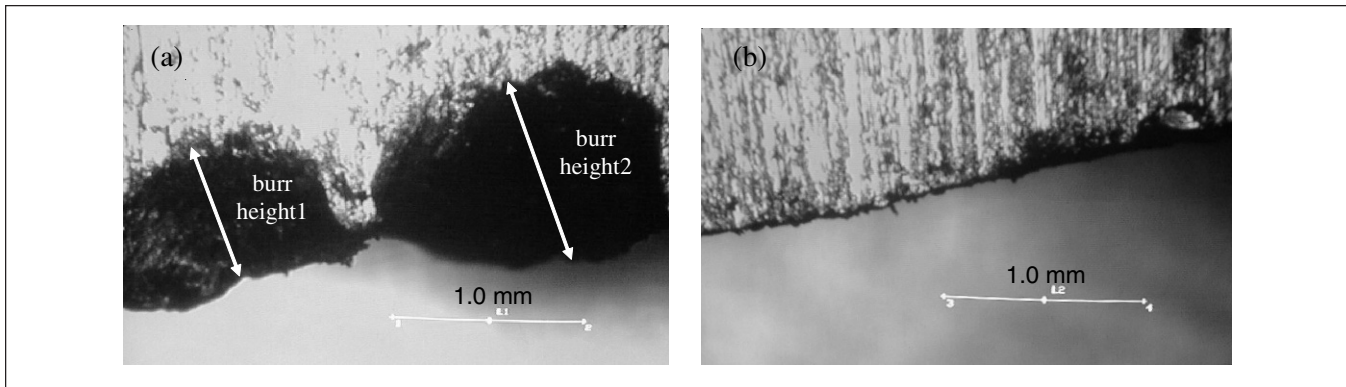


Figure 22  
(a) Burr size on the order of 1 mm formed in first part, (b) Only entrance burr formed in second part

proach is mostly applicable when deburring operations are difficult to perform. For example, the manufacture of miniature parts generally does not allow secondary finishing operations because of fixturing or precision problems. Also, the deburring work for intricate automobile parts (such as engine components) is conducted by sophisticated robot systems. To accommodate these systems, original production lines need to be reconfigured, considerably increasing the total production cost. Deburring may not be economically feasible. One urgent need is to develop a quantitative measure for characterizing edge defects.

## Conclusion

This work has developed a set of geometric approaches for reducing exit burr formation in planar milling. Depending on the edge types, corresponding algorithms were designed for generation of exit-free tool paths. Experimental results have shown that the edge quality is significantly enhanced by avoiding tool exits, as only entrance burrs can form under this circumstance. These algorithms served as a basis in a tool path planner for the edge-precision process planning. Burr formation problems have been resolved in an automatic manner through CAD/CAM integration. This work has demonstrated that burr reduction can be effectively achieved by the automatic process planning, contrary to the conventional belief that deburring is the only solution.

The approaches proposed in this paper would likely be more practical if they were designed to reduce tool exiting as much as possible, rather than eliminate it completely. In addition, the future work should be focused on shortening the machining time induced by the extra tool paths. One feasible method

is to prioritize the workpiece edges so that the exit-free tool paths only need to be employed at critical locations. Other geometric approaches for burr reduction also need further investigation, such as the application of feature technology and design for burr minimization.

## Acknowledgments

This work is supported by the Consortium on Deburring and Edge Finishing (CODEF, <http://lma.berkeley.edu>) at the University of California at Berkeley. The authors also wish to thank Professor Paul Wright and Dr. F.-C. Wang in the NSF CyberCut project (<http://cybercut.berkeley.edu>) for their assistance and financial support of this work.

## References

- Chu, C.H. (1999). "Tool path planning for exit burr minimization by estimating the total length of primary burrs." Research report. Berkeley, CA: Laboratory for Mfg. Automation, Univ. of California at Berkeley.
- Chu, C.H. and Dornfeld, D.A. (2000). "Tool path planning for avoiding exit burrs." *Journal of Manufacturing Processes* (v2, n2), pp116-123.
- Damazo, B.N.; Davies, M.A.; Dutterer, B.S.; and Kennedy, M.D. (1999). "A summary of micro-milling studies." 1st Int'l Conf. and General Meeting of the EUSPEN, Bremen, Germany (v1), pp322-325.
- Dornfeld, D.A.; Wright, P.K.; Wang, F.C.; Sheng, P.S.; Stori, J.A.; Sundararajan, V.; Krishnan, N.; and Chu, C.H. (1999). "Multi-agent process planning for a networked machining service." *Transactions of NAMRI/SME* (v27), pp191-196.
- Gillespie, L.K. (1981). *Deburring Technology for Improved Manufacturing*. Dearborn, MI: Society of Manufacturing Engineers.
- Gillespie, L.K. (1999). *Deburring and Edge Finishing Handbook*. Dearborn, MI: Society of Manufacturing Engineers.
- Gillespie, L.K. and Blotter, P.T. (1976). "The formation and properties of machining burrs." *ASME Journal of Engg. for Industry* (v98, n1), pp66-74.
- Hassamontr, J. (1998). "Edge finishing planning in milling." PhD dissertation. Berkeley, CA: Dept. of Mechanical Engg., Univ. of California at Berkeley.

- Jayaram, U.; Cramer, D.; and Mathrubutham, N. (1999). "Representation of burr information in CAD models." ASME Design Engg. Technical Conf., Las Vegas, pp1-5.
- Kramer, T.R. (1992). "Pocket milling with tool engagement detection." *Journal of Manufacturing Systems* (v11, n2), pp114-123.
- Narayanaswami, D. and Dornfeld, D.A. (1997). "Burr minimization in face milling: a geometric approach." *ASME Journal of Mfg. Science and Engg.* (v119, May 1997), pp170-177.
- Olvera, O. and Barrow, G. (1996). "An experimental study on burr formation in square shoulder face milling." *Int'l Journal of Machine Tools & Manufacture* (v36, n9), pp1005-1020.
- Olvera, O. and Barrow, G. (1998). "Influence of exit angle and tool nose geometry on burr formation in face milling operations." *Proc. of the Institution of Mechanical Engineers, Part B* (v212), pp59-72.
- Park, I. (1996). "Modeling of burr formation process in metal cutting." PhD dissertation. Berkeley, CA: Dept. of Mechanical Engg., Univ. of California at Berkeley.
- Pekelharing, A.J. (1978). "The exit failure in interrupted cutting." *Annals of the CIRP* (v27), pp5-10.
- Sarma, S.E. (1999). "The crossing function and its application to zig-zag tool paths." *Computer Aided Design* (v31), pp881-890.
- Spatial Corp. (1995). *ACIS Geometric Modeler Application Guide*. Westminster, CO.
- Tarn, Y.S. and Chang, W.S. (1993). "Dynamic NC simulation of milling operations." *Computer Aided Design* (v25, n12), pp769-775.
- TechSolve Inc. (1980). *Machining Data Handbook*<sup>TM</sup>, 3rd ed. Cincinnati.
- Wright, P.K. and Dornfeld, D.A. (1998). "CyberCut: a networked manufacturing service." *Transactions of NAMRI/SME* (v26), pp281-286.
- Wright, P.K.; Dornfeld, D.A.; Wang, F.C.; and Chu, C.H. (2000). "Decision making in a multiple-criterion agent-based process planning system." *Transactions of NAMRI/SME* (v28), pp293-298.

## Authors' Biographies

Chih-Hsing Chu attended National Taiwan University and received his BS and MS degrees from the Dept. of Mechanical Engineering. He received his PhD in mechanical engineering at the Laboratory for Manufacturing Automation, University of California at Berkeley. His project work at Berkeley was focused on Internet-based design and manufacturing. He worked as a web applications engineer at RedSpark, an Autodesk Venture, on development of web-based collaboration software. His past work experiences include research intern at DaimlerBenz AG, Stuttgart, Germany, and technical consultant for Standford Machinery, Taoyuan, Taiwan. Prior to joining National Tsing Hua University in 2002, he was on the faculty of the Industrial and Systems Engineering Dept. at Virginia Tech. He also worked as a visiting researcher for one year at the Laboratory for Machine Tools and Production Engineering (WZL), RWTH Aachen, Germany. He was an invited scholar at CREDITS Center, Sungkunkwan University, Korea, in 2005. His research interests include product development, collaborative design, design chain management, and CAD/CAM. He is a member of SME, ASME, PDMA, and SCC.

David Dornfeld received his BS, MS, and PhD degrees in mechanical engineering from the University of Wisconsin-Madison. His PhD was in the area of production engineering. He joined the faculty of the University of California at Berkeley in the Mechanical Engineering Dept. in 1977 and is presently professor of manufacturing engineering as well as the Will C. Hall Family Chair in Engineering. He is the past director of the Engineering System Research Center in the College of Engineering. In 1982 and 1992, he was Directeur de Recherche Associe, Ecole Nationale Supérieure des Mines de Paris, and Invited Professor, Ecole Nationale Supérieure d'Arts et Metiers-ENSAM, Paris, respectively. Dr. Dornfeld's research activities are in several fields of manufacturing engineering and flexible automation: acoustic emission monitoring and analysis of manufacturing processes; burr formation and edge finishing (leads an industry consortium supporting work in this area); precision manufacturing; green manufacturing; intelligent sensors and signal processing for process monitoring and optimization.