ELSEVIER

Computer-Aided Design 38 (2006) 194–209

COMPUTER-AIDED DESIGN

www.elsevier.com/locate/cad

# Voronoi diagram-based tool path compensations for removing uncut material in 2½D pocket machining

M. Salman A. Mansor, S. Hinduja *, O.O. Owodunni

*School of Mechanical, Aerospace and Civil Engineering, Faculty of Engineering and Physical Sciences, The University of Manchester, P.O. Box 88, Manchester M60 1QD, UK*

Received 10 June 2005; accepted 15 September 2005

**Abstract**

This paper solves the problem of uncut areas, which can arise when 2½D pockets are machined with radial widths of cut greater than half the cutter diameter. Using the Voronoi diagram approach, three types of uncut areas are defined i.e., corner, centre and neck uncut regions. The corner uncut area is further subdivided into five different types, the centre uncut area into four and the neck uncut area into two. Techniques for detecting each type as well as algorithms for generating the tool paths for removing them are developed based on a singularity-free Voronoi diagram approach. These efficient and robust algorithms ensure that no uncut material is left behind even for complex-shaped pockets containing islands. The proposed algorithms even permit the radial width of cut to be increased to its limiting value of tool diameter. Three examples are included to illustrate the procedures for detection and removal of the different types of uncut areas.


*Keywords:* Pocket machining; Uncut material; Tool path compensations.

## 1. Introduction

Two types of well-established tool paths strategies for machining 2½D milled components are contour parallel offset and direction parallel. For the rough machining of features with hard boundaries such as moulds and dies, the machining volume is sliced into a number of horizontal cutting layers. Each layer is treated as a 2½D pocket, which is machined using a contour parallel machining strategy [1]. Direction parallel tool paths, also commonly referred to as zig-zag or zig machining, are not preferred for features with hard boundaries because cusps are left behind along the hard edges during rough machining; the removal of these cusps requires an extra pass thus increasing the total tool path length. However, direction parallel tool paths are ideally suited for features, some or all the boundaries of which are soft e.g., a face milling feature. If such features have two or more hard faces, then the possibility of cusps arises with direction parallel tool paths. Hence, contour parallel tool paths are gaining popularity and in recent years have received attention from several researchers [1–6].

Contour parallel tool paths are currently generated by pair-wise, pixel-based and Voronoi diagram approaches and the 2D boundary curves are represented either as: (i) parametric curves such as NURB-curves; (ii) lines and arcs; or (iii) a sequence of points [1].

Irrespective of the technique used for generating contour parallel tool paths or representing the boundary curves, one of the major problems is the identification and the subsequent removal of material left behind during machining. The occurrence of an uncut area depends upon the ratio of the radial width of cut ($b$) to tool diameter ($D$) and the geometry of the feature, the essential condition being that $b/D > 0.5$. Although the uncut material problem is well known to the machining community, apart from the proprietary approaches of commercial CAM systems, only few researchers [1–5] have addressed it.

As part of other issues related to the use of the Voronoi diagram for the generation of contour parallel tool paths, Held et al. [2] considered only the type of uncut area that can occur at a sharp corner formed by two pairs of tool path lines. Details of the detection and removal were not considered apart from determining the offset value (i.e. radial width of cut), which would prevent the occurrence of uncut areas at sharp corners. Their approach was to prevent the occurrence of these uncut

* Corresponding author. Tel.: +44 161 306 3808; fax: +44 161 306 3803.
*E-mail address:* sri.hinduja@manchester.ac.uk (S. Hinduja).

0010-4485//$ - see front matter © 2005 Published by Elsevier Ltd.
doi:10.1016/j.cad.2005.09.001

areas by reducing the offset value, which of course leads to an increase in the total tool path length.

Choi and Kim [1] addressed, as part of die-cavity pocketing, the classification, detection and removal of uncut regions. They identified a corner uncut area occurring at a sharp corner, a centre uncut area occurring at the centre of an innermost contour parallel offset curve, and a neck uncut area occurring in a region where the next level tool paths split. The detection of the uncut regions is based on a pixel 2D cut simulation method in which pixels not swept as the tool traverses the tool path, constitute uncut areas. Corner and neck uncut regions are removed by inserting simple linear tool path segments at the uncut break points while each centre uncut region is removed by appending the boundary of the uncut area as a clean-up tool path. Unfortunately, the procedure to remove the uncut regions is not described in detail and no systematic approach that may suggest completeness was adopted to arrive at the types of uncut area suggested.

Park and Choi [3] addressed the problem of detecting and removing uncut regions in generating contour parallel tool paths for pocketing based on the pair-wise approach. Their solution to the uncut problem is based on the expanded pair-wise interference detection offset algorithm for the three types of uncut areas (corner uncut, centre uncut and neck uncut), which were identified by Choi and Kim in an earlier research [1]. Normally detection of an uncut area requires determining the tool envelopes of the parent and child windows and then performing a Boolean intersection operation. Park and Choi describe a more efficient method in which the Boolean operation is avoided. Their work is based on an observation that an uncut area exists if the raw offset for the tool envelope is self-intersecting. The tool path for removing the uncut areas is essentially the boundary of the uncut areas with portions of the basic tool path extended to intersect this boundary in order to join the clean up curves to the rest of the tool path in a smooth manner. Though the technique suggested guarantees that no uncut material is left behind, the tool path length for covering the uncut area can be further reduced.

Recently, Choy and Chan [4] proposed strategies in which either a single loop or a double loop is appended to every sharp corner of the tool path which smoothens the tool path, reduces the load on the cutter as well as removes the corner uncut material. The loop essentially consists of a line and an arc, which together form a D-shape. Whilst these strategies, which are actually for improving the NC tool path, can be used to overcome the problems of uncut regions, the tool path length to remove the corner uncut area is longer than what is absolutely necessary. Furthermore, only corner uncut areas are considered and no logic to detect their occurrence is given.

Quite independently and at the same time as Choi and Kim [1], Philimis [5] proposed bridge, inner window and window node compensations. Whilst the type of uncut areas identified are similar to those by Choi and Kim [1], the novelty of Philimis's work lies in the fact that it was based on the Voronoi diagram which was used to identify the uncut areas. However, the length of the tool path compensation segments suggested by him is longer than that necessary especially for centre and neck uncut areas.

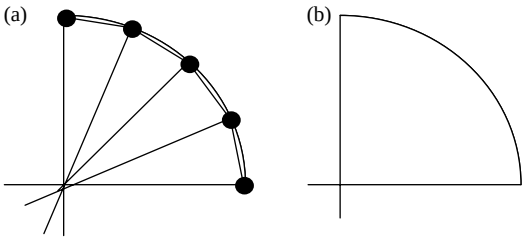


Fig. 1. Singularity problem. (a) Singularity problem with arc approximated by chords. (b) No singularity problem with a single arc.

The proprietary approaches of commercial CAM systems and the contributions of researchers to the uncut area problem have several shortcomings. Firstly, there is no scientific basis for the classes of uncut areas that have been proposed. The consequence of this is the ambiguity of the cases as well as lack of certainty whether all cases have been identified. Secondly, the algorithms for detecting the uncut areas and removing the material are not based on sound geometric premises. They either use the boundary of the uncut area or some approximation of it as the removal path for uncut areas. Thus, the results are not robust and the compensations derived are not optimal. If optimal tool paths to remove uncut areas are to be determined, there has to be a better understanding of the types of uncut areas that exist and the exact geometry of the paths required to remove them. This paper is a contribution to address this need.

In addressing the need identified above, this research carries out a systematic enumeration of the types of uncut areas using properties of the Voronoi diagram. Section 3 reports this after some preliminary definitions in Section 2. This is followed in Section 4 by Voronoi diagram-based algorithms for detecting the uncut areas enumerated in Section 3 and generating the exact tool paths for removing them. In Section 5 the computer implementation of the algorithms as well as results and discussion of tests carried out on realistic closed pockets are presented. Conclusions on the research reported in this paper are given in Section 6.

While researchers have identified the Voronoi diagram [2, 5–8,10] as a robust and mathematically sound approach, a reason for not exploiting the potential has been because of the so-called singularity problem. This problem, which is common to other methods, occurs as illustrated in Fig. 1a, when the boundary curves are represented by points [9]. The consequence of this is that the bisectors do not intersect one another at a point. The Voronoi diagram method adopted in this research represents the boundary curves by linear and circular curves and so avoids the singularity problem (Fig. 1b).

## 2. Preliminary definitions

A $2\frac{1}{2}$D feature can be defined by chains of lines and circular arcs, termed boundary curves, representing a section of the feature and the depth through which it is swept. The boundary curves, assumed to lie on the *XY*-plane, are represented in this

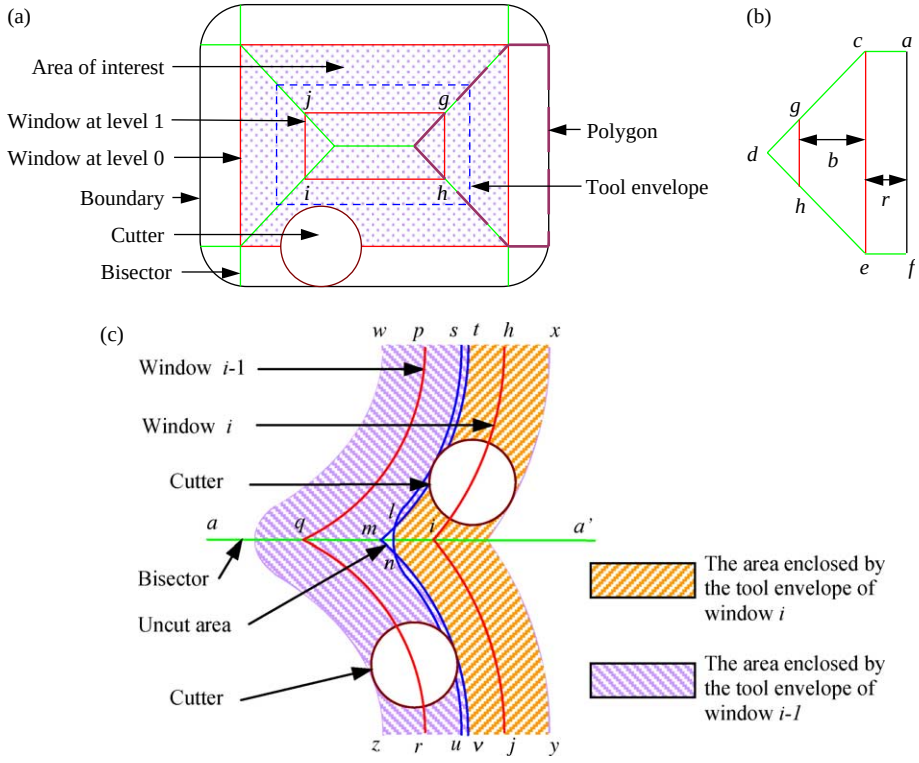


Fig. 2. Terminology associated with tool path generated using Voronoi diagram. (a) Voronoi diagram and tool path. (b) Polygon. (c) Occurrence of an uncut area.

research in the implicit forms $ax+by+c=0$ and $(x-a)^2+(y-b)^2=c^2$ for a straight line and circle, respectively. A network of curves, which are bisectors of the boundary curve is termed a Voronoi diagram. A bisector can either be a straight line, circular arc, parabola, ellipse or hyperbola and is represented by the following parametric Equations. [5–7]:

$$X(t) = x_1 - x_2 - x_3 \pm x_4\sqrt{(x_5 - x_6 t)^2 - (x_7 - x_8 t)^2} \quad (1)$$

$$Y(t) = y_1 - y_2 - y_3 \mp y_4\sqrt{(y_5 - y_6 t)^2 - (y_7 - y_8 t)^2} \quad (2)$$

where $t$ is the distance to the boundary curves corresponding to the bisector. The values of the variables $x_i$ and $y_i$ are given in Held's lecture notes [8].

The bisectors and the boundary curves can be subdivided into polygons [5], with each polygon containing one, and only one, boundary curve. In Fig. 2b, *acdef* is one of the eight polygons of the base face of the pocket, with *af* being the boundary curve (see Fig. 2b). The tool path elements are generated within each polygon by offsetting the boundary curves at distances of $r$, $r+b$, $r+2b$, etc. where $r$ is the tool radius and $b$ the radial width of cut. These offset elements (in polygon *acdef*, there are two offsets, *ce* and *gh*) are herein referred to as window lines. Both ends of a window line lie on bisectors. These window lines form the paths, which the centre of the cutter must follow. A closed chain of tool path elements is referred to as a window e.g. *ghij* (Fig. 2a). The window closest to the circuit formed by the outer boundary curves is referred to as being at level 0; the subsequent windows are said to be at level 1, level 2 etc. (as shown in Fig. 2a). A window at level $i+1$ is the child window of $i$. Alternatively, window $i$ is the parent of window $i+1$. Fig. 2a has only windows that are at levels 0 and 1.

Since uncut material occurs only inside the window at level 0, this research considers only the area enclosed by it, as shown in Fig. 2a.

## 3. Classification of uncut areas

In this section, a classification of uncut areas is carried out through a systematic enumeration of the topology and geometry of the bisectors and window lines. Two main properties are used to carry out this enumeration. The first property is that all uncut areas are centred on the Voronoi bisectors. A second property is that an uncut area is the result obtained by subtracting the areas enclosed by the tool envelopes of window $i$ (*sxyunls* in Fig. 2c) and window $i-1$ (*wtlmnvzw* in Fig. 2c) from the area enclosed by the outer envelopes of windows $i$-1 and $i$ (*wxyzw* in Fig. 2c). This second property means the adjacency of the window lines affects the type of uncut area that can occur.

Thus examination of the adjacencies of tool path windows and the bisector properties enables the enumeration of uncut areas. Two adjacency relationships that can exist between windows are parent–child relationship and child–child relationship within a multi-children parent window. In a parent–child relationship, one of three situations can arise: parent-no child, parent-one child and parent-several children relationships.

In a parent-no child relationship, it is possible to have an uncut area in the bisector network within the parent window. In

this case, the parent window becomes the innermost window. The uncut area is classified as a centre uncut area.

If there is a single child window in a parent–child relationship, uncut areas can occur only where the window lines intersect the bisectors. Therefore, the uncut areas are located at the corners of the windows. This is referred to as a corner uncut area. For the case of a parent–child relationship in a multi-children parent window, it is not possible to have a isector between the parent and child window and therefore the possibility of an uncut area does not arise.

The third type is a relationship between two child windows within a multi-children parent window. A bisector and hence an uncut area can exist in between these windows. This type is referred to as a neck uncut area.

The three cases of uncut areas identified above are equivalent to the centre uncut, corner uncut and neck uncut areas suggested by Choi and Kim [1]. The difference is that these have been arrived at by a systematic enumeration procedure using the Voronoi diagram, thus providing confidence that no case has been left out. These types are now considered further for special cases they contain.

The occurrence of a corner uncut area is a function of the geometry of two adjacent pairs of window lines. In Fig. 3, the node being considered for compensation is the corner window node $i$ formed by the intersection of window lines $hi$ and $ij$. The other pair is the parent window lines $pq_1$ and $rq_2$. Two different cases of the parent window lines can be identified. In the first case (Fig. 3), window lines $pq_1$ and $rq_2$ do not intersect the bisector $aa'$, which passes through node $i$; hence, these window lines are referred to as being unconnected. Bisector $aa'$ is referred to as the current bisector because it passes through node $i$. In the second case (Fig. 4), the parent window lines $pq$ and $qr$ meet the current bisector $aa'$ at $q$ and are said to be connected.

The boundary of the corner uncut area is formed by elements, which are segments of the tool envelops for two pairs of window lines. In Fig. 4, for example, $lm$ and $mn$ are segments of the tool envelope for the parent window lines $pq$ and $qr$, respectively, whilst $ln$ is a segment of the tool envelope for window lines $hi$ and $ij$. An uncut area is classified either as open or closed. In Fig. 4, since segments of the tool envelope for the parent window, $lm$ and $mn$, meet on the current bisector $aa'$, the uncut area is said to be closed. On the other hand, the uncut area $degf$ (Fig. 5) is open because $ed$ and $gf$, segments of the tool envelope for parent window lines $pq_1$ and $q_2r$, respectively, do not intersect the current bisector. Therefore an uncut area with connected parent window lines is always closed (Fig. 4) but one with unconnected parent window lines can either be closed (Fig. 3) or open (Fig. 5). In the latter case, segments of the tool envelope intersect two other bisectors. In a binary tree structure, these two bisectors would be the left and right bisectors of the current bisector. Another observation is that the left and right bisectors can belong to either the same polygon or to different polygons. In Fig. 6, $st$ and $uv$ intersect bisectors which belong to polygon 5, whereas in Fig. 5, segments of the tool envelope $ed$ and $fg$ intersect bisectors which belong to polygons 2 and 5.

The corner uncut area is also classified as being either small or big. The uncut area is small if the tool, when placed at any node on the boundary of the uncut area, covers the portion of the uncut area contained in the polygon in which the node lies. For example, in Fig.7, area $swv$ is said to be small because when the tool is centred at $s$, it completely covers the uncut area (shaded in the figure) contained in polygon 5. Note that it does

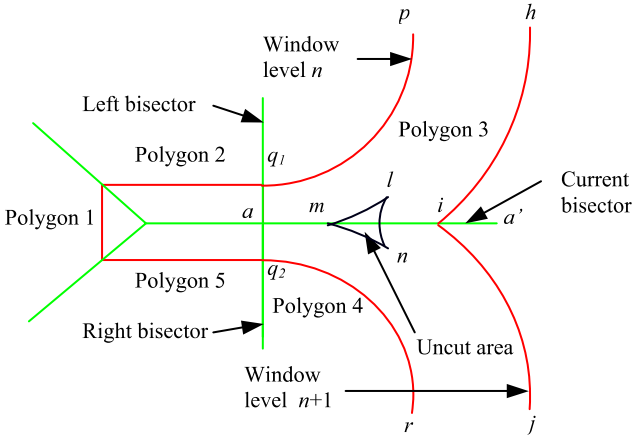


Fig. 3. Corner uncut area-type 1.

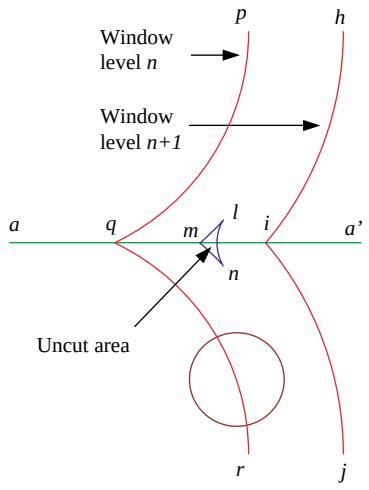


Fig. 4. Corner uncut area-type 2.

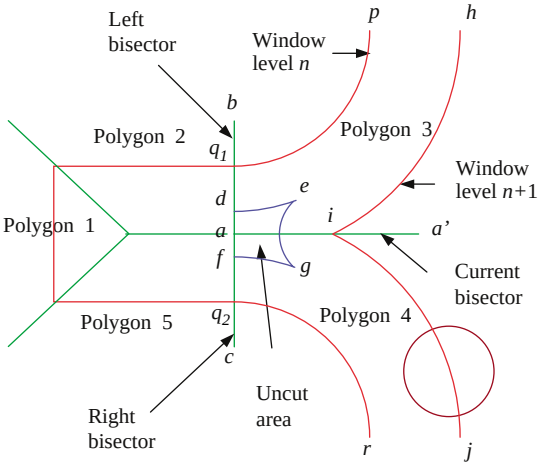


Fig. 5. Corner uncut area-type 3.

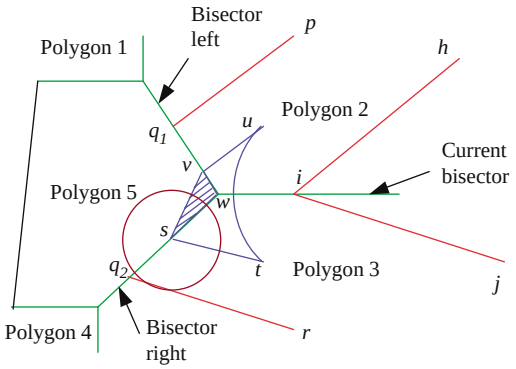


Fig. 6. Corner uncut area-type 4.

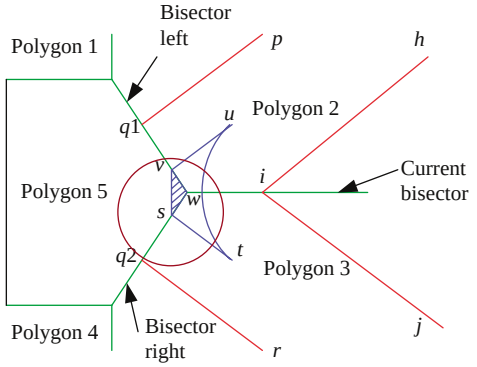


Fig. 7. Corner uncut area-type 5.

not cover the uncut areas in polygons 2 and 3. On the other hand, area *swv* in Fig. 6 is big because the tool when placed at *s* cannot completely cover the shaded area in polygon 5. As shown in Section 4, the classification of an uncut area as big or small is important because it helps in determining the most efficient compensation segment for removing it.

Hence, five types of corner uncut areas (CUA) can be identified:

(i) CUA in which the parent window lines are unconnected and the CUA is closed (Fig. 3);
(ii) CUA in which the parent window lines are connected (Fig. 4);
(iii) CUA in which the parent window lines are unconnected, the CUA is open and some of its end points are on bisectors which belong to different polygons (Fig. 5);
(iv) CUA in which the parent window lines are unconnected, the end points of the CUA are on bisectors which belong to the same polygon (Fig. 6) and the CUA is big; and
(v) CUA in which the parent window lines are unconnected, the end points of the CUA are on bisectors which belong to the same polygon and the CUA is small (Fig. 7).

A taxonomy of the corner uncut area is presented in Fig. 8.

A centre uncut area (CtUA) occurs inside an innermost window. The innermost window can be either closed (see Fig. 9a) or open (see Fig. 9c) and the uncut area can be either small or big. The definition for a small area is slightly different from that defined for the corner uncut area in the sense that when the cutter centre is placed at any one of the nodes of the uncut area, it must cover the entire uncut area.

Hence, the four different types of CtUA are:

(i) CtUA where the window is closed and the uncut area is big (Fig. 9a);
(ii) CtUA where the window is closed and the uncut area is small (Fig. 9b);
(iii) CtUA where the window is open and the uncut area is big (Fig. 9c); and
(iv) CtUA where the window is open and the uncut area is small (Fig. 9d).

A neck uncut area (NUA) occurs between at least two pairs of window lines (*ijk* and *lmn* in Fig. 10) and their parent lines (*opqr* and *stuv*). The two pairs of window lines (*ijk* and *lmn*) are at the same level and each pair subtends an acute angle. These two pairs of window lines are related in the sense that one window line from each pair, share the same parent window. A neck uncut area is a special case of a centre uncut area that has two or more open sides.

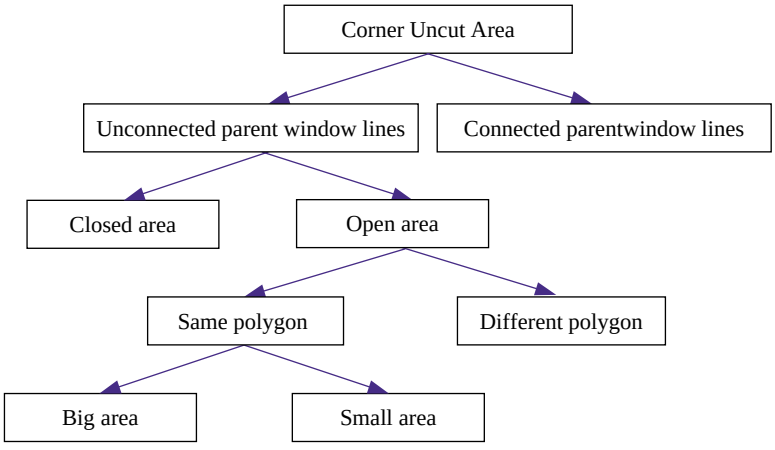


Fig. 8. Classification of corner uncut areas.

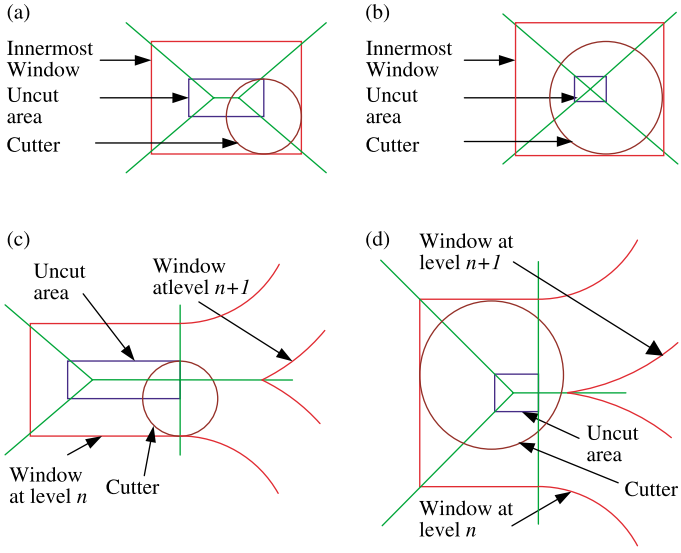


Fig. 9. Different types of centre uncut areas. (a) A big, closed centre uncut area. (b) A small, closed centre uncut area. (c) A open, big centre uncut area. (d) A open, small centre uncut area.

Neck uncut areas can be classified either as big or small and an example of each is shown in Fig. 10.

## 4. Algorithms for tool path compensations

In this section, algorithms that generate tool paths for removing the different types of uncut areas discussed in the previous section are presented. These tool paths, referred to as corner node, inner window and neck uncut compensations are described in Sections 4.1 and 4.2, while tool paths for compensating open uncut areas are presented in Section 4.3.

### *4.1. Algorithm for compensating corner node uncut area*

Fig. 11 shows a corner node (*wn*), which is analysed to determine if any compensation is required.

Step 1. The algorithm starts by calculating the parametric distance ($t_{wn}$) of node *wn* from its boundary curve. Also shown in Fig. 11 are the parent corner node (*pwn*), the parent window lines and the envelope traced by the cutter as it travels along the parent window lines. If the parametric distance to the parent window lines is $t_{pn}$, then the parametric distance to the envelope is given by

$$t_p = t_{pn} + x\,(\text{tool radius}) \tag{3}$$

Theoretically, $x$ should be equal to 1 but in practice, the value of $x$ is in the range 0.9–1. This has the effect of making the area bigger than its actual size, resulting in a longer compensation segment; this ensures that the area is completely removed.

Step 2. Next, the value of $t_p$ is compared with the start, end and minimum parametric values ($t_s$, $t_e$, and $t_{\min}$, respectively) of the current bisector *se*. For the tool envelope to intersect the current bisector, one of the following conditions must be satisfied. It is necessary to consider all the three conditions, (see Fig. 12) because in that way one does not have to determine as to which end of the bisector is the start (or end) node.

(i) $t_p \geqq t_s$ and $t_p < t_e$
(ii) $t_p < t_s$ and $t_p \geqq t_e$
(iii) ($t_p < t_s$ and $t_p < t_{e)}$ or ($t_p > t_s$ and $t_p > t_e$) and ($t_p \geqq t_{\min}$ and $t_{\min} > 0$)

Step 3. Assuming that one of the three conditions is satisfied, the Cartesian co-ordinates of the intersection point $p$ are determined using Eqs. (1) and (2).

Step 4. If the distance $|wn–p| < r$ then an uncut area does not exist at this corner node. If $|wn–p| > r$, then a circle, centred at $p$ and of radius $r$, is intersected with the current bisector. If the intersection point between the tool radius circle and the current bisector is denoted by $q$, then the compensation is given by the segment $wn–q$ of the bisector (Fig. 11).

Step 5. Whilst in Fig. 11 the tool envelope intersects the current bisector, this is not always the case. There is no intersection if none of the three conditions in step 2 is satisfied. In such cases, the bisector network is searched and the left and right bisectors of the current bisector are identified. Fig. 13 shows the tool envelope intersecting the left and right bisectors at $p_1$ and $p_2$, respectively. In some cases, the tool envelope intersects only the left or right bisector. For example, Fig. 14 shows the tool envelope intersecting only the left bisector.

Step 6. If there are two intersection points and they are represented by $p_1$ and $p_2$, then steps 3 and 4 are carried out for

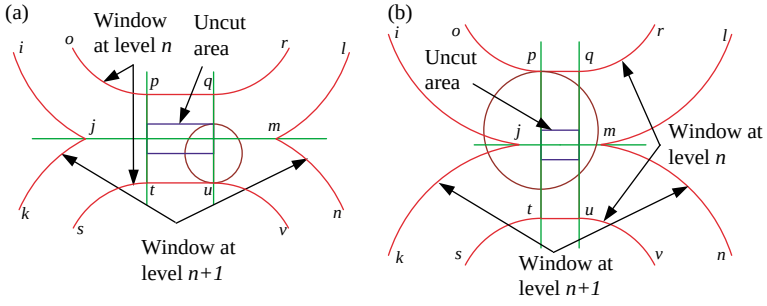


Fig. 10. Different types of neck uncut areas. (a) Big neck uncut area. (b) Small neck uncut area.

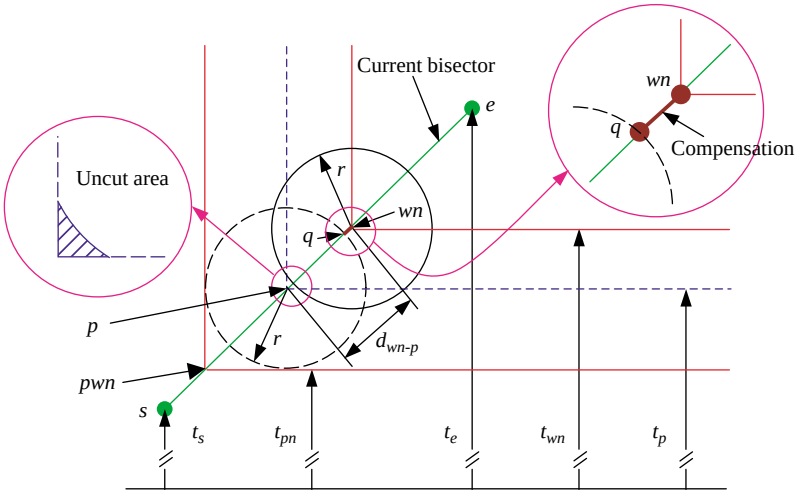


Fig. 11. Corner uncut area and its compensation.

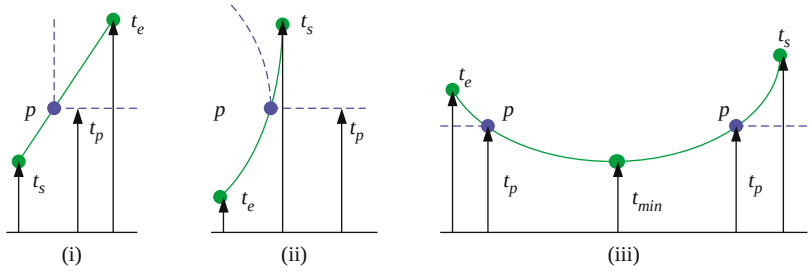


Fig. 12. Different cases of intersection between the tool envelope and bisectors.

each of them. Assuming that distances $|wn–p_1|$ and $|wn–p_2|$ are greater than $r$, then the longer of the two distances is retained. In Fig. 15, since $|wn–p_2| > |wn–p_1|$, the compensation is given by $wn–p_0–q$.

Step 7. When the tool envelope intersects the left and/or right bisectors and not the current bisector (Fig. 16), then a compensation segment is introduced from window node *wn* to *s*, the end of the current bisector. Fig. 17 shows the uncut area as well as the compensation segment *wn–s*. At this stage no analysis is performed as to whether the entire length or part of *wn–s* is necessary. This situation arises when the parent window lines are unconnected and there is an open centre uncut area. Later on, this compensation segment will be either merged with the compensations for the centre uncut area, shrunk or even discarded.

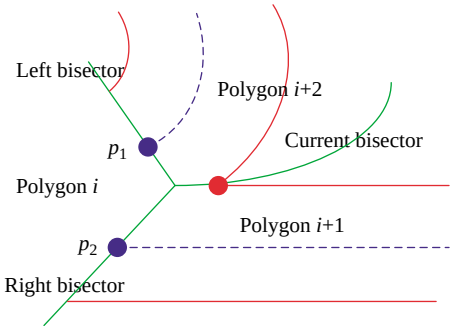


Fig. 13. Tool envelope intersecting bisectors belonging to the same polygon.

### 4.2. Algorithms for compensating centre uncut areas

The algorithm presented in this section detects the existence of an uncut area in the inner window and neck regions, and generates compensation tool path segments to remove them. The first four steps of the algorithm are executed for each polygon; steps 5–7 are executed after all the polygons have been considered.

Step 1. Within a polygon, the window line with the highest parametric value of $t$ is identified and if its parametric value is given by $t_{wlmax}$, then Eq. (3) is used to calculate the parametric distance of the tool envelope for this window line from the boundary curve (Fig. 18) where $x$ is between 0.9 and 1.0 as in Eq. (3).

Step 2. All the bisectors of this polygon are examined and the maximum parametric value $t_{max}$ is obtained by examining

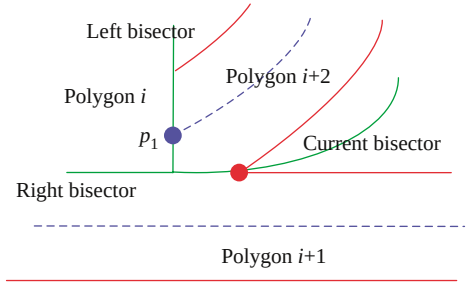


Fig. 14. Tool envelope intersecting only one bisector.

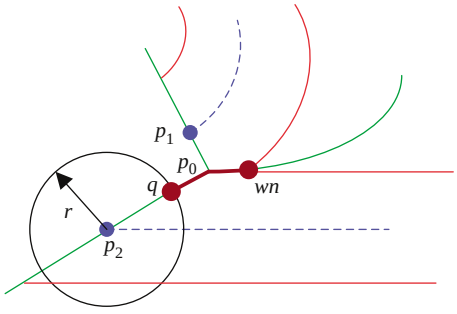


Fig. 15. Determination of the corner node compensation.

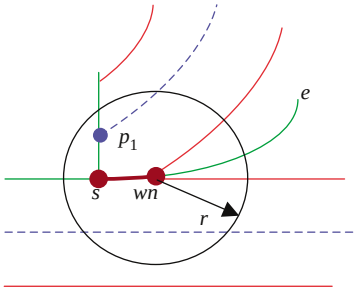


Fig. 16. Temporary corner node compensation for no uncut area.

the $t_s$ and $t_e$ values of each bisector in the polygon [5]. An inner window or neck uncut area exists in this polygon if $t_{\max} > t_p$.

Step 3. To determine the uncut area, the tool envelope is intersected with each bisector of the polygon. Normally, a bisector will have only one intersection point and a total of two intersection points in a polygon (Fig. 19a). However, if the central bisector is non-linear (see Fig. 19b), there can be a total of four intersection points. The Cartesian co-ordinates of these points are calculated, as before, using Eqs. (1) and (2).

Step 4. Bisectors that contain an intersection point are subdivided into two parts i.e. upper and lower parts. Points on the upper part of the bisector will have parametric values greater than $t_p$; the upper part belongs to the uncut area boundary. If both ends of a linear bisector have values greater than $t_p$, then the entire bisector is an upper bisector and is part of the boundary of the uncut area.

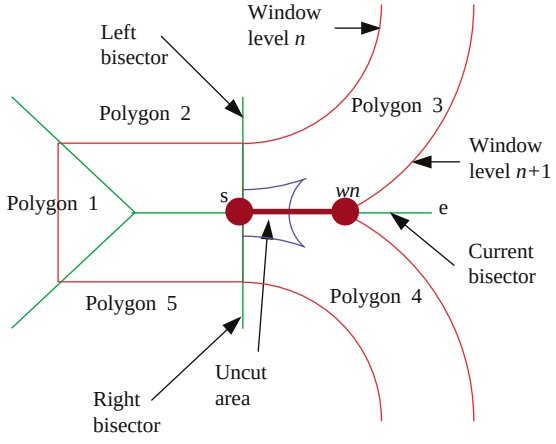


Fig. 17. Temporary corner node compensation for an open uncut area.

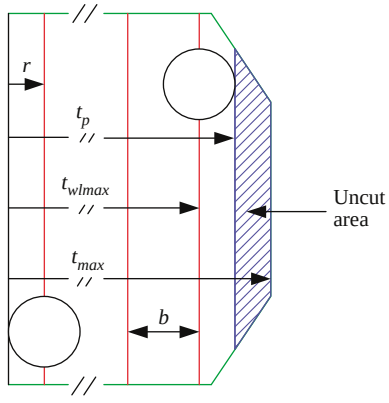


Fig. 18. Centre uncut area within a polygon.

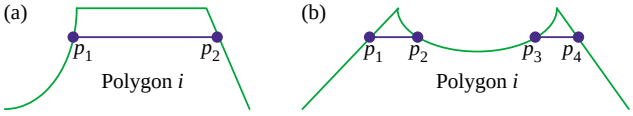


Fig. 19. Intersection of tool envelope with bisectors. (a) Two intersection points. (b) Four intersection points.

If the central bisector is non-linear, and there are more than two intersection points between the tool envelope and the bisector, then a portion of this bisector will belong to the lower part. Fig. 20 shows two disjointed uncut areas, $p_4$–$p_3$–$b$ and $p_2$–$p_1$–$c$, which are formed by the upper parts of bisectors $p_4$–$b$, $b$–$p_3$, $p_2$–$c$, $p_1$–$c$ and segments of the tool envelope $p_1$–$p_2$ and $p_3$–$p_4$. If the uncut area is classified as small, part of the lower bisectors $a$–$p_4$ and $d$–$p_1$ will be used to determine the compensation segments.

Step 5. After all the polygons have been considered, the upper bisectors and uncut areas computed for each polygon are merged together. In some cases, the merger results in an uncut area that is closed i.e. it is bounded on all sides by tool envelope lines as shown in Fig. 21. In other cases, the resulting uncut area is open on one or more sides, as in Figs. 22 and 23; in the former, the uncut area is open on one side

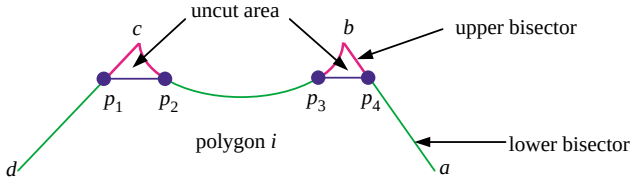


Fig. 20. Two disjointed uncut areas within one polygon.

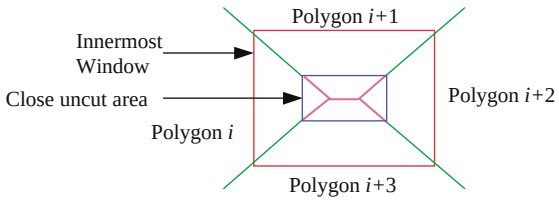


Fig. 21. Closed centre uncut area.

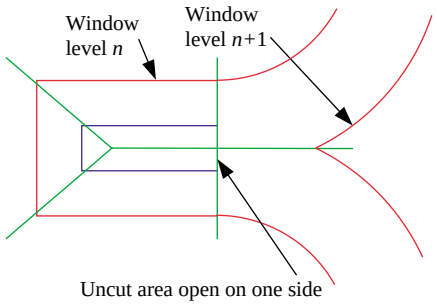


Fig. 22. Uncut area open on one side.

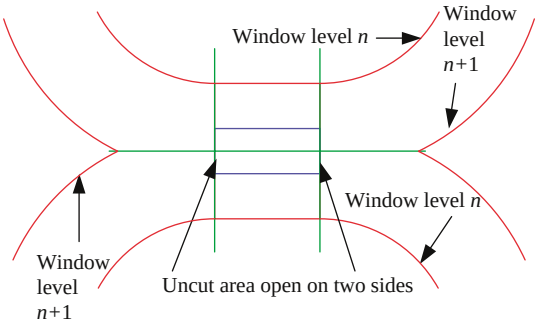


Fig. 23. Uncut area open on two sides.

and in the latter on two opposite sides. The latter is an example of a neck uncut area.

Step 6. The uncut area is classified as either small or big. If the area is small, then it can be machined by the cutter travelling some distance along one of the lower bisectors. Assuming that a particular bisector ($b$) is selected, then to determine the compensation segment, the tool radius circle is placed at each of the corners of the uncut area and the corresponding intersection points with bisector $b$ determined. To ensure complete coverage by the tool, the compensation segment is given by $wn$–$m$, where $wn$ is the window node lying on $b$ and is nearest to the uncut area and $m$ is the furthest intersection point from $wn$.

In Fig. 24a the centre uncut area is given by $p_1p_2p_3$. If compensation is required along $b_1$, the tool radius circle is successively placed at $p_1$, $p_2$ and $p_3$ and the intersection points $m_1$, $m_2$ and $m_3$ on bisector $b_1$ are determined, respectively. Hence, the compensation segment which should be appended to $wn$ is $wn$–$m_1$, since $m_1$ is the furthest point from $wn$.

Similarly, compensations segments are determined for the other two bisectors connected to the small-uncut area. Fig. 24b shows the three possible compensations from which one has to be selected.

Step 7. The procedure for obtaining the compensation segments to remove a big uncut area essentially involves obtaining the positions on the bisector network to which the cutter centre should travel to in order to cover the corners of the uncut area. Determination of the circle centres is described in step 7.1; steps 7.2–7.4 correspond to the three stages of achieving the compensation. The first stage involves determining the subset of the bisector that will remove each corner. In the second stage, the paths for removing each corner are merged together to obtain a set of paths, which removes all the corners. In the last stage, one of the merged paths is selected.

Step 7.1. For a big uncut area, the tool-radius circle is placed at each corner of the uncut area and its intersections with the upper part of the bisectors are determined. In Fig. 25, the intersections with the bisectors when the tool-radius circle is placed at corners 1, 2, 3, 4 are $b$, $d$, $c$ and $e$, and $a$, respectively.

Step 7.2. To determine the tool path that would cover corner $i$, the paths from each of the other corners $j$ ($j=1$, $n$ and $j\neq i$) along the bisector network to each intersection point $k$ ($k=1,m$) are determined for those cases where the distance between corners $i$ and $j$ is greater than the tool radius. This yields a set of paths $\{path_{i,j,1}, path_{i,j,2}, \cdots path_{i,j,k}, \cdots path_{i,j,m}\}$ each starting from corner $j$ and correspond to the $m$ intersections of the circle placed at corner $i$. Thus, in Fig. 25, the possible paths to cover corner 3 of the uncut area, starting from corners 1, 2 and 4 are

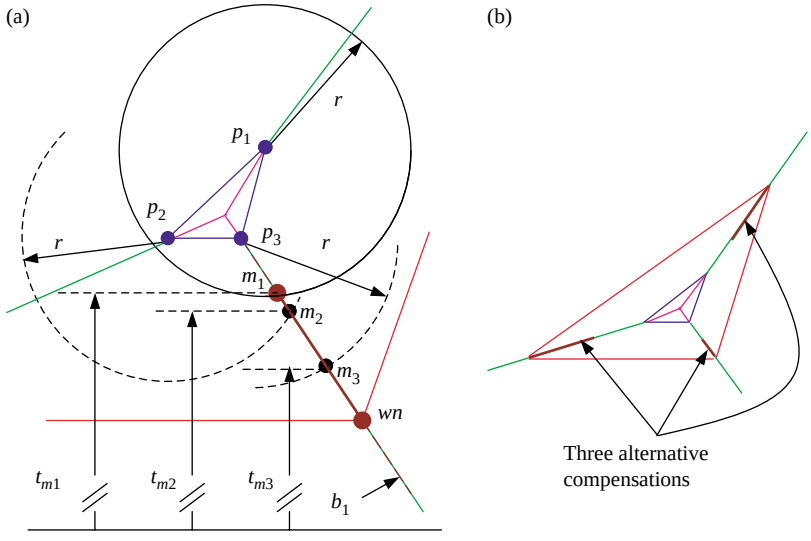


Fig. 24. Alternative compensations to machine small uncut area. (a) Compensation at node $wn$. (b) Three alternative compensation.

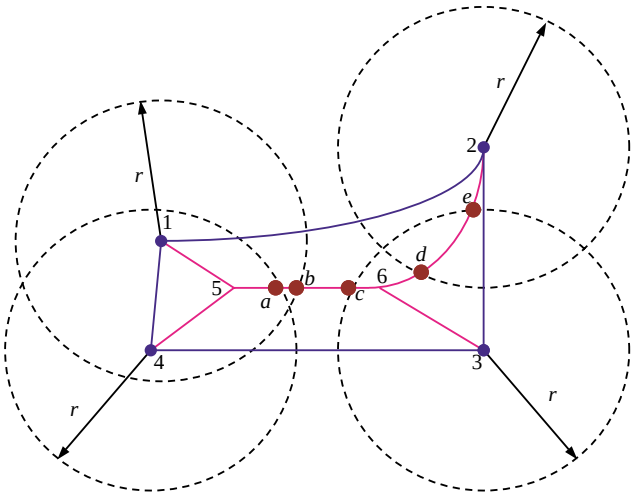


Fig. 25. Intersections of the tool-radius circle with the upper bisectors.

{1-5-*c*, 1-5-6-*e*}, {2-*e*, 2-6-*c*}, and {4-5-*c*, 4-5-6-*e*}, respectively.

From the set of paths for each pair of corners *i* and *j*, only one path whose length is shorter than the others is required. This required $path_{i,j}$ can be represented as follows:

$$path_{i,j} = minimum\ length\{path_{i,j,1}, path_{i,j,2}, \cdots path_{i,j,k}, \cdots path_{i,j,m}\} \quad (4)$$

Applying this rule in the example of Fig. 26a and b, it means that of the paths 1-5-*c* and 1-5-6-*e*, $path_{3,1}$ = 1-5-*c* is the required path to cover corner 3 starting from corner 1. This procedure leads to the determination of a single path $path_{i,j}$ to cover every corner *i* of the uncut area starting from other corners *j* ($j \neq i$). Thus for each corner *i*, there would a set of paths $\{path_{i,1}, path_{i,2} \cdots path_{i,j} \cdots, path_{i,n}\}$ each of which corresponds to starting from a corner *j* ($j \neq i$) of the uncut area.

Step 7.3. The path, *path*, which covers the entire uncut area would be the union of the paths which cover each corner. Thus *path* can be written as in Eq. (5).

$$path = \{path_{1,2}\ OR\ path_{1,3} \cdots OR\ path_{1,n}\} \cup \cdots \cup \{path_{n,1}\ OR\ path_{n,2}\ OR\ path_{n,n-1}\} \quad (5)$$

The expansion of Eq. (5) would yield the expression of Eq. (6). The first *n* terms of Eq. (6) represent the paths from each of the corners of the uncut areas, which would cover the entire uncut area. The subsequent terms represent the paths, which would require starting from more than one corner to cover all the uncut area. These latter terms are not efficient and can be neglected as unfeasible solutions. Each of the former terms can be represented as the path $path_j$, which is the union of all the paths $path_{i,j}$ which start from corner *j* and cover each corner *i* of the uncut area as stated in Eq. (7).

$$path_j = path_{1,j} \cup path_{2,j} \cup \cdots\cdots \cup path_{n,j} \quad (7)$$

The union operation for determining each path $path_j$ is not geometrically involved and only requires checking the segments of the paths, $path_{i,j}$. A segment is part of the union if it is the only one or longer than or equal in length to any other segment lying on the same bisector. In Fig. 27 the union of the paths 1-5-6-*d* and 1-5-*c* starting from corner 1 leads to $path_1$ = 1-5-6-*d*. The final result of this step would be the path $path = \{path_1\ OR\ path_2$ OR $\cdots path_j\ OR\ \cdots OR\ path_n\}$ which are alternate paths corresponding to the number of nodes, *n*, of the uncut area.

Step 7.4. The final path is obtained as an optimal of the possible paths through each corner. Of the four possible paths, 1-5-6-*d*, 2-6-*a*, 3-6-*d*-*a* and 4-5-6-*d*, shown in Fig. 28 to remove the uncut area, 3-6-*d*-*a* would be the best if the length of the tool path is the optimisation criterion.

### 4.3. Algorithm for compensating open centre uncut areas

When the centre uncut areas from different polygons are synthesized, the resulting uncut area may be open on one or more sides, an open side being one which is bounded by a bisector(s). Associated with each open side, as shown in Figs. 29a and 30a, is a corner window node *i* to which, as described earlier in Section 4.1, a temporary compensation segment *i*–*h* was attached.

Since a neck is a special case of a centre uncut area, the algorithm described in the previous section would have determined several feasible compensation segments to machine it. For example (see Fig. 29b), for the open uncut area *abcd*, there are four possible compensation segments (*c*–*h*–*g*, *b*–*h*–*g*, *d*–*e*–*f* and *a*–*e*–*f*) from which one has to be selected.

In Fig. 29a, to remove the corner uncut area *bcef*, the temporary segment *i*–*h* should be shrunk to *i*–*k*. However, it would be unwise to select one of the four solutions shown in Fig. 29b and the compensation segment *i*–*k* as it would result in a disjointed solution. Instead, the solution adopted herein is to discard the portion of the segment labelled *h*–*c* or *b*–*h* and to

$$\begin{aligned} path = \{ & path_{2,1} \cup path_{3,1} \cup \ldots\ldots \cup path_{n,1} \} && OR \\ & \{ path_{1,2} \cup path_{3,2} \cup \ldots\ldots \cup path_{n,2} \} && OR \\ & \ldots\ldots\ldots\ldots\ldots\ldots\ldots && \\ & \{ path_{1,j} \cup path_{2,j} \cup \ldots\ldots \cup path_{n,j} \} && OR \\ & \ldots\ldots\ldots\ldots\ldots\ldots\ldots && \\ & \{ path_{1,n} \cup path_{2,n} \cup \ldots\ldots \cup path_{n,n-1} \} && \\ & . && \\ & . && \\ & \ldots\ldots\ldots\ldots\ldots\ldots\ldots && \end{aligned}$$

First *n* terms; Subsequence terms (6)

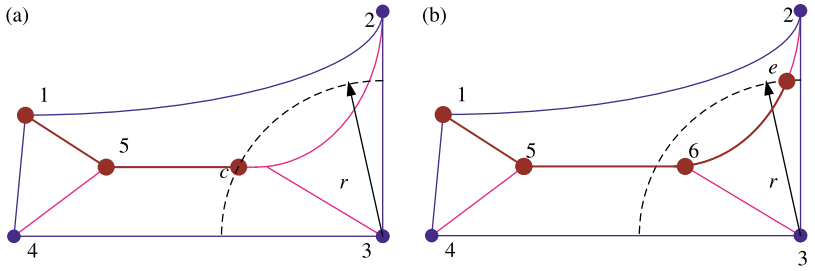


Fig. 26. Two possible compensation paths starting from node 1 to cover node 3.

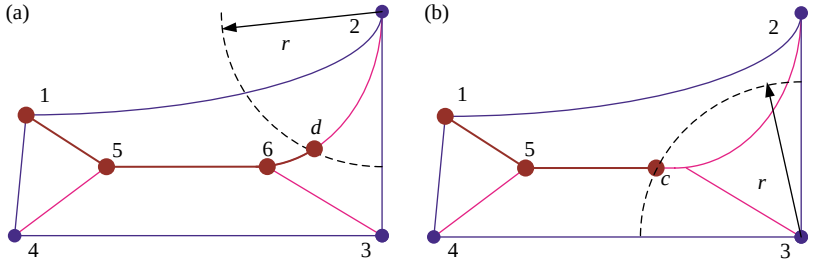


Fig. 27. Compensation paths starting from 1 to cover nodes 2 and 3.

extend the temporary segment *i–h* to *i–g* to form a continuous segment (Fig. 29a).

If the uncut area is open on two sides resulting in a neck (Fig. 30a), then there are at least two corner nodes, each with temporary compensation segments (i.e. *i–h* and *j–k*) attached to them. Again, there are four alternative compensations for the neck uncut area (Fig. 30b). If, for example, the first of the four solutions is selected, then *c–h* is discarded, *j–k* is extended to meet *i–l* (Fig. 30b), thus forming a bridge between the two corner nodes *i* and *j*. Although parts of the segments *j–k* and *i–h* are strictly not required, their inclusion provides an efficient solution.

In some cases, the open uncut area is small and the initial compensation segment *i–h*, instead of being extended, is shrunk to size *i–k*; in spite of this reduction in length, the tool is able to cover the small uncut area (Fig. 31). In a very few cases, the entire compensation segment is dispensed with because the uncut area is very small and is covered when the tool is at *i*.

## 5. Results and discussions

The algorithms described in Section 4 are part of a CAM system called TechMill for milled components that is currently under development. Three examples demonstrating these algorithms are presented and discussed in this section. In each of the examples the construction of the Voronoi diagram and the tool paths are considered to have been completed and

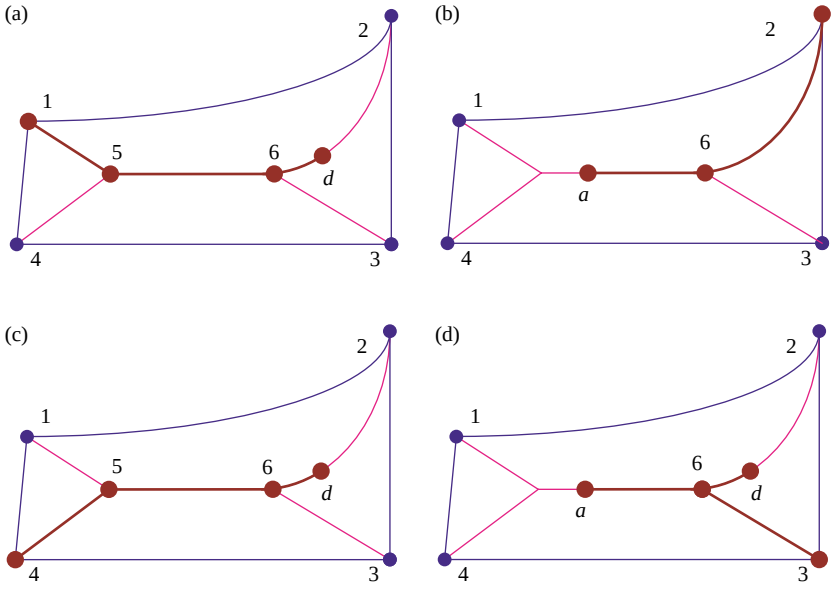


Fig. 28. Four alternative compensations to machine the centre uncut area 1–2–3–4.

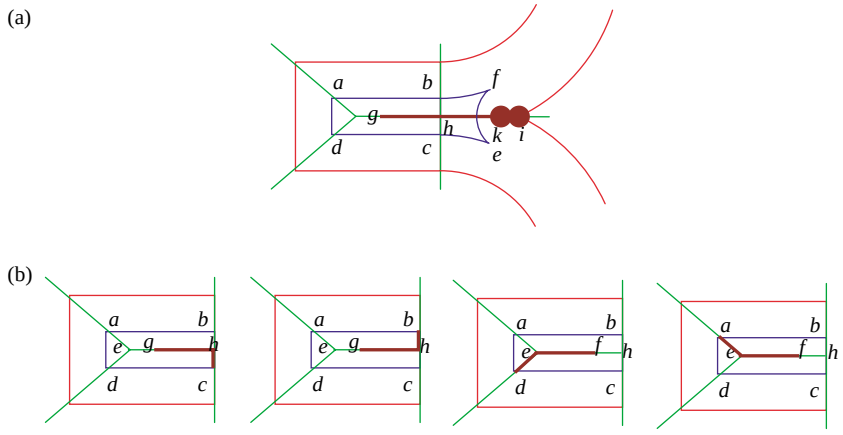


Fig. 29. (a) Synthesised compensation for centre and corner uncut areas. (b) Four alternative compensations to remove the open centre uncut area.

only the generation of the tool path compensation segments to remove the uncut area is considered.

### 5.1. Example I

The first example is a 2½D closed pocket the base face of which is shown in Fig. 32. Although the geometry of the pocket appears to be simple, the example is interesting as it gives rise to seven uncut areas (Fig. 33), one of which is a centre uncut area (*A*), four are corner uncut areas (*B*, *C*, *E* and *F*), and two are open uncut areas (*D* and *G*). The tool paths shown in Fig. 33 have been obtained using a tool of 18 mm diameter with a radial width of cut of 14.4 mm. The tool paths consist of two sets of windows, the first set containing two windows i.e. *abcde* and *fghij* and the second set contains only window *klm*. The two sets have a common parent window.

Window *fghij* gives rise to a centre uncut area which has two attributes: it is closed and big. Of the five feasible compensation segments shown in Fig. 34, the compensation shown in Fig. 34e is preferred as it results in the shortest tool path length of 21.23 mm. Other researchers would remove this uncut area by introducing another window coincident with the boundary of the uncut area; this would result in a compensation segment being 72.42 mm long. If

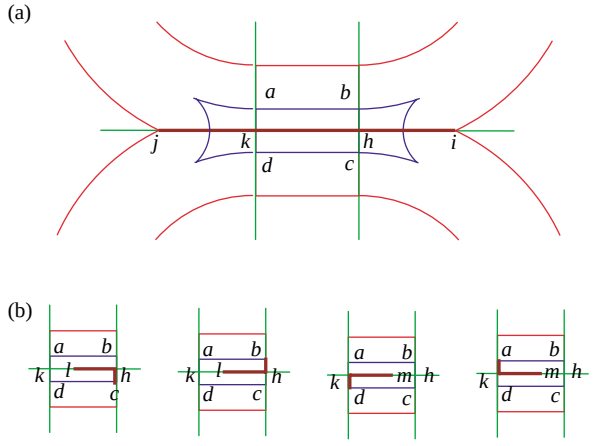


Fig. 30. (a) Synthesised compensation for neck and two corner uncut areas. (b) Four alternative compensation segments for the neck uncut area *abcd*.

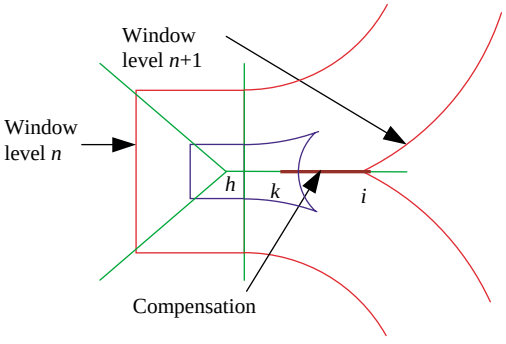


Fig. 31. Initial corner node compensation segment shrunk from *i–h* to *i–k*.

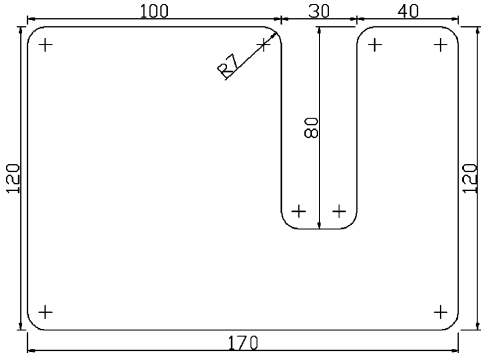


Fig. 32. Base face of a 2½D closed pocket.

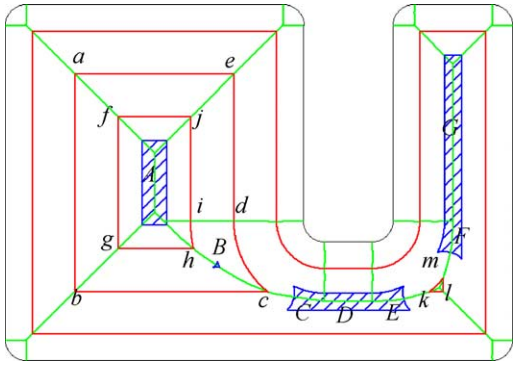


Fig. 33. Voronoi diagram, tool paths and uncut areas.

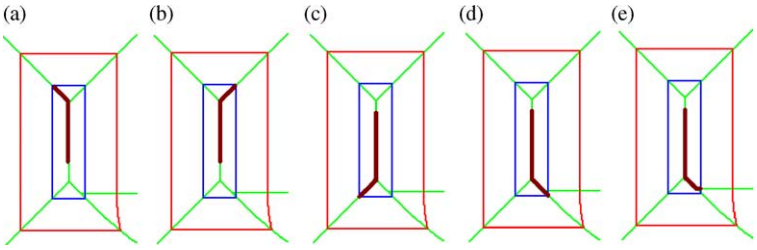


Fig. 34. Alternative compensations to remove the centre uncut area *A*.

one were to overmill the compensation segment suggested herein to account for the tool travelling back to the window *fghij*, the total travel would still be 2/3rd of that suggested by other researchers.

The second uncut area lies on bisector *hc* and the compensation is a small segment *h–n* appended to corner window node *h* (Fig. 35). The length of the compensation segment is 1.74 mm.

Corner node *c* also gives rise to uncut area $a'b'c'd'$ (Fig. 36a) which is open because the tool envelope lines $a'b'$ and $d'c'$ do not intersect the current bisector *co*. One of the tool envelope lines ($a'b'$) intersects the left bisector *op* but there is no intersection between the right bisector *oq* and the tool envelope line $d'c'$. Since this area is open, initially a compensation segment *c–o* is attached at node *c*.

Nodes *k* and *m* (Fig. 36b and c) give rise to similar types of uncut areas *E* and *F*; temporary compensation segments *k–q* and *m–r* are attached at corner nodes *k* and *m*, respectively.

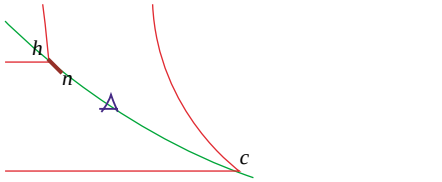


Fig. 35. Compensation *h–n* to remove corner uncut area *B*.

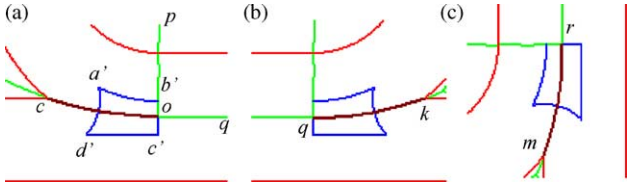


Fig. 36. Open corner uncut areas and their compensations. (a) Area *C*. (b) Area *E*. (c) Area *F*.

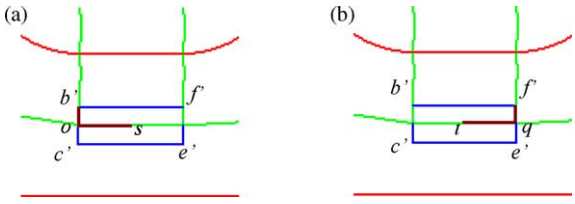


Fig. 37. Alternative compensations to remove the neck uncut area *D*.

Area *D* (Fig. 37), as defined by $b'c'e'f'$, is an example of a big neck uncut area open on sides $b'c'$ and $e'f'$. To remove the neck uncut area, two compensations segments ($b'$–*o*–*s* and $f'$–*q*–*t*) are initially determined by the system. But since there are two corner node compensations on either side of the neck, these compensations are discarded. Instead, compensation segments *c–o* and *k–q* are extended to meet each other, thus forming a bridge compensation between nodes *c* and *k* (Fig. 39).

Area *G* (Fig. 38), as defined by $g'h'i'j'$, is a centre uncut area and was obtained by synthesising the uncut areas in polygons 1, 2 and 3. The other attributes of this area are that it is big and open. The three alternative compensation segments that can be used are shown in Fig. 38. Since there is a corner uncut area adjacent to this area, the compensation *m–r* for the corner uncut area is extended to *u*, resulting in a composite segment *m–r–u* (Fig. 39).

Merger of the tool path compensations as discussed above with the tool path windows results in a total tool path length of

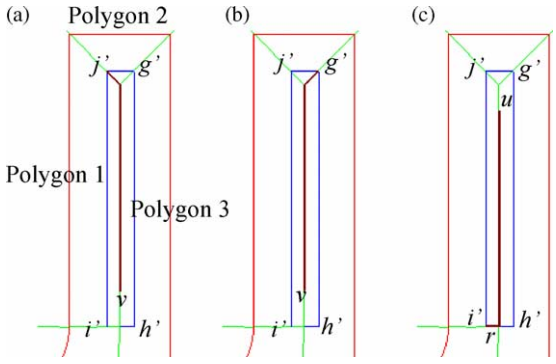


Fig. 38. Alternative compensations to remove the open centre uncut area *G*.

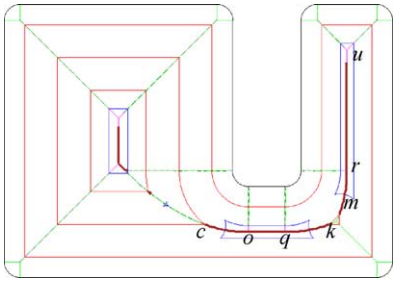


Fig. 39. Tool path compensations as calculated by the TechMill system.

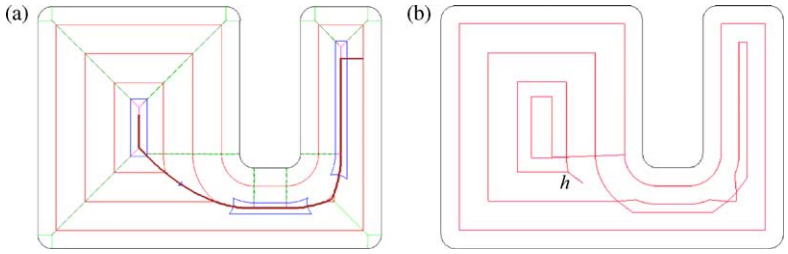


Fig. 40. Comparison of tool paths from different systems. (a) TechMill: length = 1258 mm. (b) Pro-Engineer: length = 1481 mm.

1258 mm (see Fig. 40a). This example has also been analysed using three commercial systems i.e. Pro-Engineer, EdgeCAM and DEPOCAM. The results obtained from Pro-Engineer are shown in Fig. 40b. The compensation segments are very different from those suggested by the authors. To start with, the compensation for uncut area $A$ is given by a loop. The corner compensation at node $h$ is 9.06 mm which is 5 times that in Fig. 35. Similarly the compensations for uncut areas $C$, $D$, $E$, $F$ and $G$ are not line segments but loops. Hence, it is not surprising that the total tool path with Pro-Engineer is 1481 mm which is 17.8% greater than that shown in Fig. 40a. The tool path and compensation segments obtained with EdgeCAM were very similar to those obtained with Pro-Engineer and the total tool path length was about 1457 mm. With DEPOCAM, however, the uncut areas had to be machined in a separate operation and the recorded tool path length was 1688 mm.

The proposed algorithm can cater for uncut areas even when the radial width of cut is equal to the cutter diameter. Fig. 41 show the tool paths and the compensations resulting when $b/D = 1$. There is only set of window frames and both the inner windows have compensations at each of the corner nodes. Attached to node $c$ is a composite compensation segment to machine the uncut area.

### 5.2. Example II

The second example is a closed pocket with a triangular island, the feature base of which is shown in Fig. 42a. To generate the tool paths, a tool radius of 9 mm and a $b/D$ of 0.95 have been assumed. There are five uncut areas and their tool path compensations are shown in Fig. 42b. Areas 1–4 are of the same type i.e., open corner uncut areas. Of particular interest is the fifth uncut area. Although it appears to be a centre uncut area, it is in fact a neck uncut area; moreover, since it is small, the compensation for it is a small segment attached to corner node $i$. The compensation is unusual in the sense that the segment is towards the centre, unlike other corner node compensations, which are towards the outer boundary.

### 5.3. Example III

The third example (Fig. 43a) is a 2½D closed pocket containing a human-shaped protrusion similar in shape to that given by Held [8]. A tool radius of 10 mm and a $b/D$ ratio of 0.85 have been assumed to generate the tool paths. There are 17 uncut areas and their tool path compensations are shown in Fig. 43b. Areas 1 and 2 are corner uncut areas the parent windows lines of which are connected; areas 3 and 4 are also closed corner uncut areas but the parent window lines are unconnected; area 5 is also a corner uncut area with the parent window lines unconnected and tool envelope lines open; moreover, the tool envelope lines intersect the left and right bisectors which belong to the same polygon, and the area is big. Area 6 has the same characteristics as 5 except that the left and right bisectors belong to different polygons. In area 7, the parent window lines are unconnected and the uncut area is open, the left and right bisectors belong to the same polygon and the area is small. Area 8 is a closed centre uncut area, the area being small. Lastly, area 9 is an example of a small neck uncut area. Since the feature is symmetrical, areas 10–17 are identical to areas 1–8.

Of particular interest are the 2nd, 11th, 8th and 17th uncut areas. Since uncut areas 8 and 17 are small and of the closed centre uncut type, the tool path compensations for them are small segments attached to corner nodes $i$ and $j$ (Fig. 43b), and the compensations are towards the centre of the innermost windows. On the other hand, for uncut areas 2 and 11, the compensations attached to corner node $i$ and $j$ are towards the outer boundary. Hence, there are two compensations attached to node $i$, one of which is a corner node compensation and the other an inner window compensation. Similarly, two different types of compensations are attached to corner node $j$.

## 6. Conclusions

Although the types of uncut areas that have been identified are similar to those identified by other researchers, the result has been obtained systematically using the

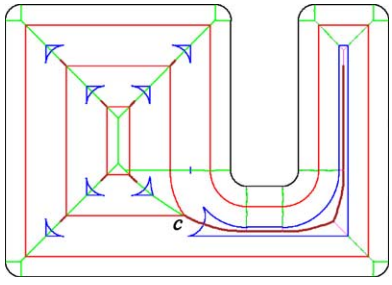


Fig. 41. Uncut areas, tool paths and compensations when $b/D = 1$ and $r = 9$ mm.

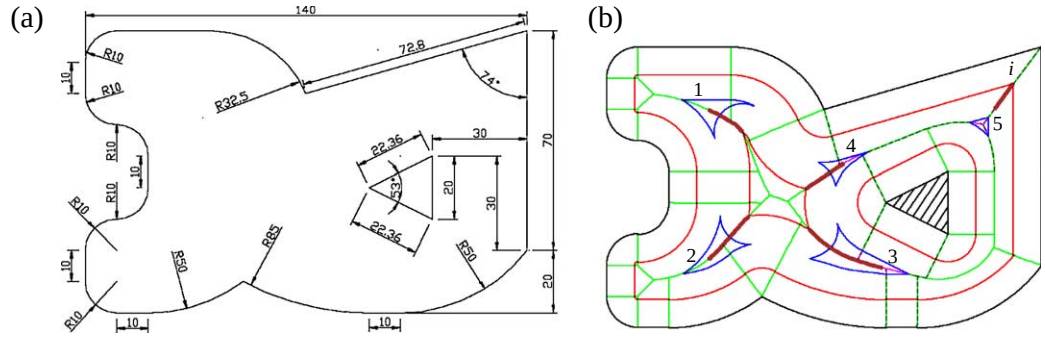


Fig. 42. Example II. (a) Base face of a pocket with one protrusion. (b) Voronoi diagram, tool paths, uncut areas and their compensations with $b/D=0.95$ and $r=9$ mm.

Voronoi diagram approach; moreover, this work has utilised the properties of the uncut areas and their parent window lines to classify these uncut areas into more specific types. This sub-classification has resulted in novel and efficient tool path compensations.

The compensation are optimised for minimum length; moreover, in the case of centre uncut areas, several alternative solutions are obtained so that at the time of linking two neighbouring windows, a particular compensation can be selected to reduce the amount of over milling. As a result of the efficient compensations, the total tool path lengths are considerably shorter than those generated by commercial CAM systems and other researchers.

The strategy for generating the tool compensation is robust since the system is able to generate compensation segments even when the ratio of $b/D$ approaches unity.

Although there are four different types of compensations (corner, inner window, neck and bridge compensations), only two basic algorithms, one for corner uncut areas and the other for centre uncut areas, have been developed. The neck algorithm is an adaptation of that for the centre uncut area. The bridge compensation is obtained by synthesising the compensations for the corner and neck compensations, thus resulting in a continuous compensation.

The examples have shown that the compensations suggested in this research are much shorter than those generated by leading commercial CAM systems. However, the proposed compensations have disadvantages—they often result in overmilling and abrupt changes in direction. The latter may result in jerky motions but this can be overcome if loops are constructed around the suggested compensations resulting in a smoother tool motion.

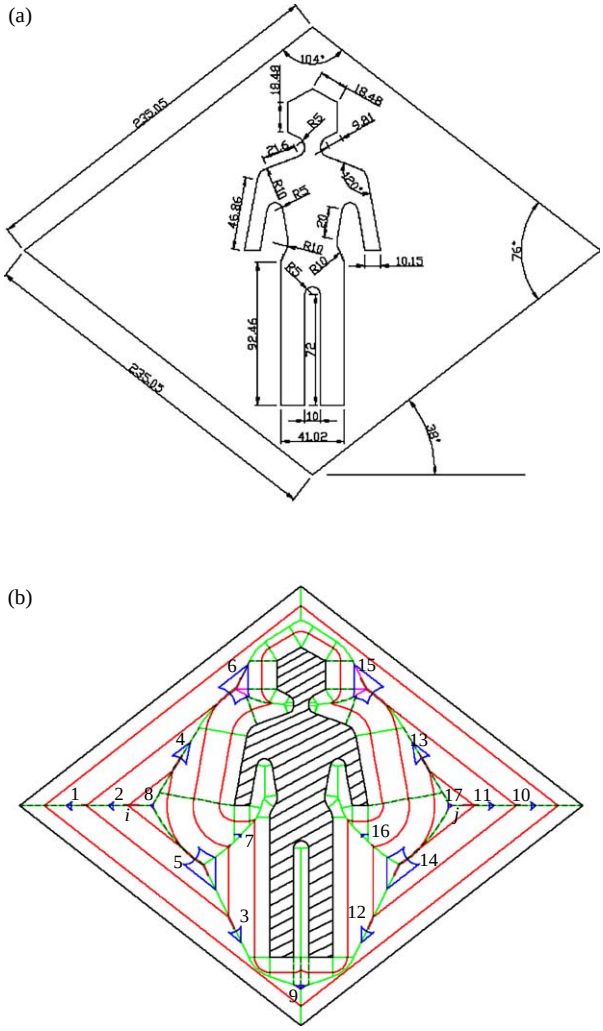


Fig. 43. Example III. (a) Base face of a pocket with a human-shaped protrusion. (b) Uncut areas and tool path compensations with $b/D=0.85$ and $r=10$ mm.

## Acknowledgements

M. Salman A. Mansor would like to thank the Malaysian Government and the Universiti Sains Malaysia for sponsoring his PhD studies at The University of Manchester; Dr. O.O. Owodunni would like to acknowledge the financial support from the EU for the I*PROMS project.

## References

[1] Choi BK, Kim BH. Die-cavity pocketing via cutting simulation. Comput-Aided Des 1997;29(12):837–46.
[2] Held M, Lukacs G, Andor L. Pocket machining based on contour-parallel tool paths generation by means of proximity maps. Comput-Aided Des 1994;26(3):189–203.
[3] Park SC, Choi BK. Uncut-free pocketing tool-paths generation using pair-wise offset algorithm. Comput-Aided Des 2001;33(10):739–46.
[4] Choy HS, Chan KW. A corner-looping based tool path for pocket milling. Comput-Aided Des 2003;35(2):155–66.

[5] Philmis P. Optimum toolpaths for 2.5D pockets. PhD Thesis. University of Manchester Institute of Science and Technology, Sackville Street, Manchester, United Kingdom; 1999.
[6] Sandiford D. Feature volume construction and determination of optimum toolpaths and tool sets for 2½D milling. PhD Thesis. University of Manchester Institute of Science and Technology, Sackville Street, Manchester, United Kingdom; 2002.
[7] Royaldi A. Voronoi diagram construction and optimum tool selection for 2½D milling features. PhD Thesis, University of Manchester Institute of Science and Technology, Sackville Street, Manchester, United Kingdom; 1998.
[8] Held M. On the computational geometry of pocket machining. Berlin: Springer; 1991.
[9] Choi BK, Park SC. A pair-wise offset algorithm for 2D point-sequence curve. Computer-Aided Des 1999;31(12):735–45.
[10] Person H. NC machining of arbitrary shaped pockets. Comput-Aided Des 1978;10(3):169–74.

**Mohd Salman Abu Mansor** is currently a PhD student at The University of Manchester, UK. He graduated with a BEng (Hons) Degree in Mechanical Engineering from the Universiti Sains Malaysia (USM), Pulau Pinang, Malaysia in 2001 and MSc Degree in Advanced Manufacturing Technology and Systems Management from the University of Manchester Institute of Science and Technology (UMIST), Manchester, UK in 2002. His research interests are in the area of Computer Aided Process Planning, CAD/CAM and Reverse Engineering.

**Srichand Hinduja** is a Professor at the School of Mechanical, Aerospace and Civil Engineering (MACE), The University of Manchester, UK. He graduated with a Bachelor of Engineering Degree from the College of Engineering, Guindy, Madras, India in 1965. After a spell of 2½ years in Germany, he continued with his postgraduate studies and received his MSc and PhD from the University of Manchester in 1968 and 1971, respectively. He lectures on finite elements, numerical control engineering and CAD/CAM. His two major current research interests are modelling the pressure die casting process using finite and boundary elements, and the process planning of turned and milled components. He is the head of the Manufacturing Group in the School of MACE.

**Oladele Owodunni** graduated with a BSc (Mechanical Engineering) and MSc (Mechanical Engineering) from the University of Lagos, Nigeria. After some years of lecturing in Nigeria, he studied for his PhD at University of Manchester Institute of Science and Technology, Manchester (UMIST), UK. He is currently involved in postdoctoral research at the University of Manchester. His research interests include feature recognition, CAD/CAPP/CAM, reverse engineering, layered manufacturing and reconfigurable tooling.