Interference-free tool-path generation in the NC machining of parametric compound surfaces

J S Hwang

A method is presented for generating interference-free tool paths from parametric compound surfaces. A parametric compound surface is a surface that consists of parametric surface elements. The method is largely composed of two steps: points are obtained from a compound surface to be converted into a triangular polyhedron; tool paths are then generated from the polyhedron. An efficient algorithm is used in the calculation of cutter-location data, and planar tool paths, which are suitable for metal cutting, are produced. The time taken to obtain all the tool paths from a surface model that consists of a large number of parametric surfaces is short. Some real applications are presented.

compound surfaces, NC machining, cutter-location data

CAD/CAM systems are now used in various manufacturing applications, and the surface-modelling features of those systems make it possible to manufacture arbitrarily shaped objects without physical models. However, some problems still exist in the NC machining of complicated shapes that can be seen easily in the inner or outer panels of car bodies. This is because the surface models representing those shapes are usually composed of many surfaces, so that the tool paths for machining them cannot be generated with ease. This paper discusses several approaches to this issue, and presents a method that is numerically stable for relatively many surfaces.

A collection of topologically unrelated surface elements is often called a 'compound surface'[1], and it is called a 'parametric compound surface' if the elements are composite parametric surfaces[2]. Most surface modellers use rectangular parametric surfaces as their surface-modelling tools, and these are also used in this paper. An example of a parametric compound surface is shown in Figure 1a.

For machining parametric surfaces, the parametric machining method is often used (see, for example, Reference 3). This method makes the tool move along either u or v isoparametric curves, maintaining its contact with the surface. If this method is to be used in compound-surface machining, some operations should be carried out before tool-path generation. These are the surface-trimming or relimiting operations to eliminate tool interference that may occur within a surface (when the curvature radius of the surface is less than the radius of the ball endmill) or among surfaces. These operations require a great deal of user interaction, and interference may not be avoided if mistakes are made.

Several approaches have been proposed to avoid such interference automatically. One of them is to generate cutter-contact (CC) data (i.e. points or curves on surfaces), without worrying about interference, and then to convert them into interference-free tool paths by removing the gouging portions (as in References 4 and 5). Even though the method has some good features, such as computational efficiency and the automatic generation of cutter-location (CL) data for the subsequent removal of uncut volumes (as discussed in Reference 5), this method may bring about some serious problems in metal cutting, particularly in rough cutting. A tool path produced by this method is not planar, while the CC data, which are usually generated by sectioning the part surfaces with a plane, are planar, so that it is difficult to control NC-machine feed rates for variable cutting volumes along the curved path (in Reference 2, this path is termed 'CC Cartesian', and the planar tool

Department of Stamping Tool Manufacturing, Hyundai Motor Company, 700 Yangjung-dong, Ulsan 681-791, Korea
Paper received: 29 June 1991. Revised: 7 April 1992

 0010-4485/92/120667-10

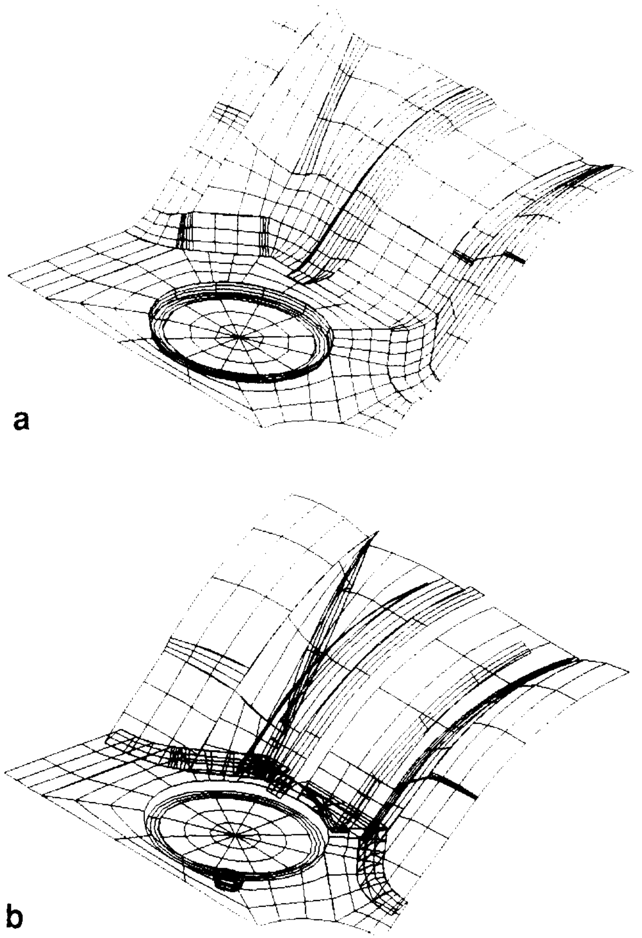

Figure 1. *Parametric compound surface; (a) surface, (b) its offset surface*

path is termed 'CL Cartesian' (see the difference in Figure 2)). Further, in rough cutting with tip-insert-type tools that are widely used for metal cutting, possible interference between the tool shank and materials may cause tool breakage or machine troubles. In Figure 2a, such interference may occur in the path joining the two marked points. For these reasons, planar tool paths are much preferred in metal cutting.

Duncan and Mair's approach[1] is similar to that of the author of this paper in that CL data are generated from a polyhedron model. In their method, however, CL points are calculated at every triangular facet of the polyhedron, so that it takes more time to generate all the machining data, and also the tool paths are not planar.

Another approach uses the Z buffer, sometimes called the Z map. A Z buffer is a collection of z-coordinate values, computed at the rectangular grid points on the xy plane, in a domain of interest. It can be obtained by the intersection of surfaces and vertical lines passing through the grid points. Tool paths are generated from the Z buffer by the 'inverse offset method'[6]. Possible disadvantages of this approach are that a great many points must be picked out to satisfy the machining tolerance, and much computation is also needed to determine one tool position. One merit of the approach is that tools other than ball endmills, such as filleted (torus) endmills, can be used easily.

The offset-surface approach was seriously considered in the research described in this paper, because it is an easily conceivable way of generating CL data. However, the author found some problems in the adaptation of the method for practical compound-surface models. Figure 1b shows the offset surface of the compound-surface model in Figure 1a. As can be seen in Figure 1b, the elements of the offset surface are placed very irregularly in space. This means that it is not necessary for the practical applications to model surfaces for which geometrical-continuity conditions are satisfied among them (not patches), which requires a great deal of time. If tool paths are to be generated from the offset surface, it is necessary to find a good method of linking the spaced offset-surface elements, and trimming the overlapped elements often located in the interference regions. However, it was not possible to find a robust method for the irregular offset surface.

In this paper, a method of generating interference-free tool paths for machining parametric compound surfaces is presented. The method mainly consists of two steps: first, a 'point data' is obtained from a parametric compound surface, and CL data is then generated from the point data, which may be considered as a polyhedron by triangulation. The parametric compound surfaces given in this paper are modelled with CATIA, the CAD/CAM system used in the author's company. A patch of the CATIA parametric surface has a maximum of 15 degrees. In practice, the compound surface consists of the 'faces' of surfaces as well as the surfaces. A portion of a surface confined by several 'edges' is a face, and an edge of a surface is represented by two polynomial equations, $u = u(w)$ and $v = v(w)$, where u and v are the parameters of the surface[7]. Hence, a compound surface can be modelled without overlapping of its elements. The term 'not overlapped' means that any crosssectional curves generated by sectioning the surface elements with a vertical plane can be linked to become one curve without curve-trimming operations.

POINT-DATA GENERATION

A collection of point passes generated by 'digitizing' a compound surface is called 'point data'. The method

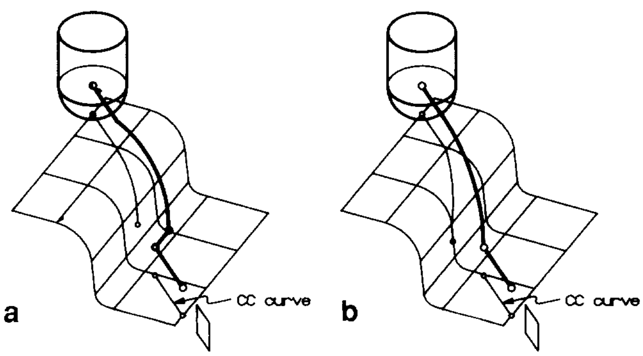

Figure 2. *Cartesian tool paths; (a) CC path, (b) CL path*

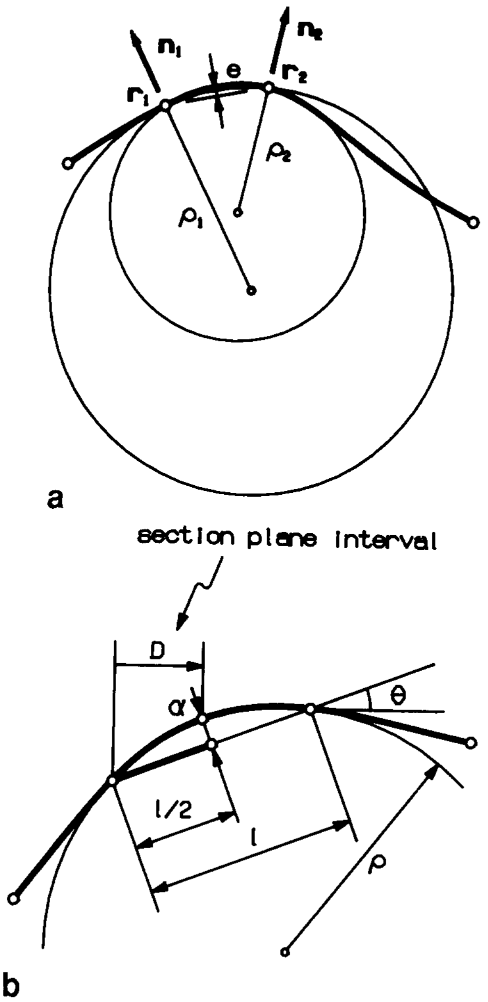

Figure 3. *Determination of step length and pass interval*

used by Scherrer and Hillberry[8] is used to generate the point data. By using the method, sequences of points can be obtained from the surface elements intersected by a section plane, and linked to become a pass of points across the compound surface. Thus, the point data can be obtained by sectioning the compound surface with parallel section planes.

The step lengths of points are determined by a well known method in which locally defined circles that approximate curve segments are used. Figure 3a shows the circles, which are defined by two points and the normal vector (see the Appendix for the details). In the calculation, two chordal deviations e_1 and e_2 are obtained, and the sampling error is estimated as $e = (e_1 + e_2)/2$.

Because two passes of points are to be converted into triangular facets in the next step, the pass interval must be determined properly to make the deviations of facets from the surface less than the point-sampling tolerance α. Figure 3b shows the way to compute the interval (a facet is shown by a heavy line). Let an edge of the facet be on the right-hand side point of the heavy line; then, the chordal deviation represents the deviation of the edge from the surface. Thus, if the deviation is set to α, its maximum allowable value, the interval can be calculated as $D = (1/2)l\cos\theta$, where l is the chord length, and θ is the angle between the horizontal line and the chord. The chord length l can be computed by using Equation 5 in the Appendix. In the implementation, D is computed at every sampled point with the help of the previous pass, and the smallest D becomes the next pass interval.

TOOL-PATH PLANNING METHOD

CL data is represented as the pair $(\mathbf{e}, \mathbf{u})$, where $\mathbf{e}$ is the coordinates of the tool-tip point, and $\mathbf{u}$ is the tool-axis vector[5]. For 3-axis NC machining, the tool-axis vector $\mathbf{u} = (0, 0, 1)$. This is also so in this case, and CL data can be obtained by calculating only a tip point $\mathbf{e}$, which is called a cutter-location point. As shown in Figure 4a, when a part surface exists, and the offset surface at a distance R, the radius of a ball endmill, from the surface is intersected by a vertical line at the point $\mathbf{p}$, the CL point is given by

$$\mathbf{e} = \mathbf{p} - R\mathbf{u} \tag{1}$$

The point $\mathbf{p}$ is actually the centre of the ball of the tool. In the interference region, an interference-free CL point can be obtained with the highest of the intersection points. The point $\mathbf{p}_h$ shown in Figure 4a is the highest point.

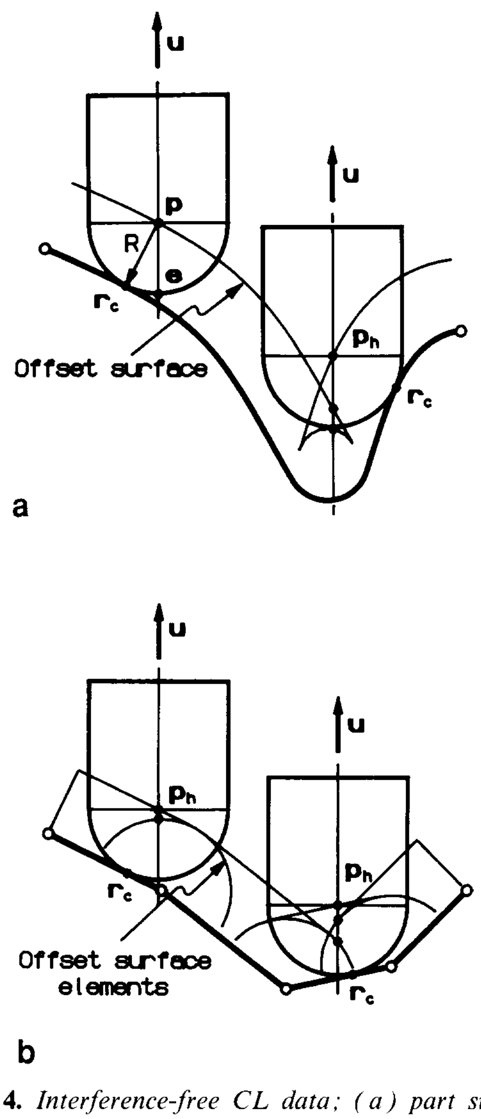

Figure 4. *Interference-free CL data; (a) part surface, (b) 2D polyhedron*

The surface of the polyhedron constructed with the point data is composed of triangular facets so that its offset surface consists of portions of spheres, cylinders and plane facets that are the offset surfaces of vertex points, edge lines and triangular facets, respectively. The 2D version of a polyhedron and its offset surface elements is shown in Figure 4b. As can be seen in Figure 4b, an interference-free CL point for the polyhedron can be obtained with the highest of all the intersection points of its offset-surface elements and a vertical line, whether it is in the interference region or not. Therefore, a pass of CL points can be produced with consecutive vertical lines, and, if these lines are contained in a plane, the generated tool path is planar. In the implementation, the plane containing the tool path is set to be parallel to the section plane of the point-data generation step for the sake of computational efficiency.

ANALYSIS OF RELATIONS BETWEEN TWO ADJACENT TOOL VOLUMES

In this section, some terminologies are introduced, and the relations between two adjacent tool volumes are analysed, which is necessary for the later discussion of the CL-point calculation algorithm.

Definitions

As shown in Figure 5, the point **p**, the centre of the half sphere of the ball endmill, is called a cutter middle (CM) point. The face of the half sphere is called the cutter face (CT) of the tool. Let R be the radius of the tool, S be a part surface, and **p** be a CM point on a vertical line (remember that the tool-axis vector $\mathbf{u} = (0, 0, 1)$). Then, a portion of the surface S, given by

$$S_{\mathbf{p}} = \{\mathbf{r} \mid |\mathbf{p} - \mathbf{r}|^2 - (\mathbf{u} \cdot (\mathbf{p} - \mathbf{r}))^2 < R^2, \mathbf{r} \in S\}$$

is called the cutter-contact region of **p**, and a point $\mathbf{r}_c$ in $S_{\mathbf{p}}$ is called the cutter-contact point of **p** if

$$|\mathbf{p} - \mathbf{r}_c| = R$$

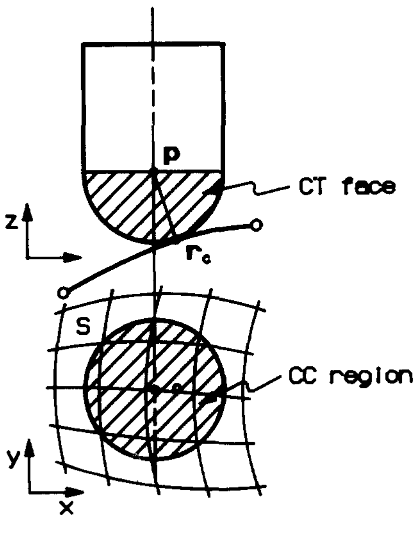

Figure 5. *CM point, CC point, CT face and CC region*

and

$$\mathbf{u} \cdot (\mathbf{p} - \mathbf{r}_c) > 0$$

For all points **r** in $S_{\mathbf{p}}$, if the expressions

$$|\mathbf{p} - \mathbf{r}| \geqq R$$

and

$$\mathbf{u} \cdot (\mathbf{p} - \mathbf{r}) > 0$$

are satisfied, the CM point **p** is called an interference-free CM (IFCM) point.

Relations between two adjacent tool volumes

If there is a CM point on a vertical line, where the tool interferes with the part surface, the IFCM point on that line must be at a position further up than the CM point. Remember that an interference-free CL point is obtained with the highest CM point. If there are an IFCM point $\mathbf{p}_1$ on a vertical line and a neighboring CM point $\mathbf{p}_2$ on another line, and the tool at $\mathbf{p}_2$ interferes with both the part surface and the volume of the tool at $\mathbf{p}_1$, as shown in Figure 6, three situations may occur, as follows:

- The shank (cylindrical portion) of the tool at $\mathbf{p}_2$ is gouged by the half sphere of the tool at $\mathbf{p}_1$ (see Figure 6a).
- The half sphere of the tool at $\mathbf{p}_2$ interferes with the half sphere of the tool at $\mathbf{p}_1$ (see Figure 6b).
- The half sphere of the tool at $\mathbf{p}_2$ interferes with the shank of the tool at $\mathbf{p}_1$ (see Figure 6c).

In the latter two situations, an interesting relationship is that any points on the CT face of $\mathbf{p}_2$ which are included in the volume of the tool at $\mathbf{p}_1$ cannot interfere with the part surface. This is because, if the points interfered with the part surface, $\mathbf{p}_1$ could not be an IFCM point. Therefore, if a CM point is obtained on a vertical line such that, at the point, the tool interferes with the part surface and the tool volumes of neighboring IFCM points, the IFCM point on that line can be calculated within the reduced CC region. Because it is common practice in sculptured-surface machining that most sets of two adjacent tool volumes intersect each other, in other words, that most step sizes are smaller than $2R$, an efficient algorithm calculating CL points can be developed that is based on these relations. Figure 6 shows the reduced CC region for each case, projected on the xy plane.

Actually, the second situation occurs in most cases, and the modified CC region is confined by a portion of an ellipse, which is created by projecting the circle at the intersection of two half spheres. The CC region in this case can be reduced, in fact, in two ways: reducing the CC region on the xy-plane domain, as mentioned above, and reduction on the CT-face domain, in which a plane between the two CM points partitions the region. The latter method is used in the author's research, because it is simpler than the former, which requires the calculation with an ellipse. The Appendix gives the details of the reduction method on the CT-face domain.

TOOL-PATH GENERATION

As mentioned above, a tool path is generated with consecutive vertical lines. This section outlines the algorithm for calculating a CL point with a vertical line, and it also describes the method of determining step sizes and tool-path intervals. Other issues relating to this tool-path-generation method are also discussed below.

CL-point calculation algorithm

Since an interference-free CL point is produced from an IFCM point with Equation 1, the algorithm presented below is actually for calculating an IFCM point. The calculation of CM points with the vertex points of the polyhedron is simpler than with the triangular facets. Thus, a CM point where the tool may slightly gouge facets is first calculated with vertex points in the CC region, and then the CC region is reduced, on the basis of the relationships discussed above, and the exact IFCM point is computed with the facets and edges in the reduced CC region. The overall procedure of the algorithm consists of the following steps.

(1) Reduce the CC region with neighboring IFCM points, if possible.

(2) Obtain an initial CM point with all the vertex points in the CC region.

(3) Reduce the CC region one more time, if possible.

(4) For all the facets in the rereduced CC region, do the following:
  (4.1) Select a facet.
  (4.2) If it is identified that the tool at the current CM point does not interfere with the infinite plane containing the facet, go to Step 4.1.
  (4.3) If the CC point is located within the bounds of the facet, update the z value of the CM point, and go to Step 4.1.
  (4.4) If it is identified that the tool may interfere with an edge of the facet, correct the z value of the CM point according to the comparison of the current and the new z value.

(5) Convert the IFCM point to an interference-free CL point.

The algorithm is discussed below in more detail. When a vertical line is set, and an IFCM point is to be calculated on that line, the inner area of a circle becomes the CC region on the xy plane domain, and, for the calculation of the highest CM point with vertex points, the points in that circle must be selected. In fact, the circle is the projected shape of the ball endmill. Incidentally, if there exists a neighboring IFCM point whose CC point is in the current CC region, the tool can be placed at a virtual CM point, touching the CC point. This may make it possible to reduce the CC region, and the number of vertex points to be selected may also be reduced. In the implementation, three neighboring IFCM points, one in the same path and two in the previous tool path, are chosen. If one or more of the three CC points are in the current CC region, the virtual CM point, the highest CM point calculated with the CC points, is obtained, and the CC region is reduced if possible. The calculation of a CM point with a vertex is simple. Let (x_p, y_p) be a 2D point where a vertical line penetrates the xy plane, and $\mathbf{r} = (x_r, y_r, z_r)$ be a vertex point in the CC region. Then, the z value of the CM point where the tool touches the point $\mathbf{r}$ is given by

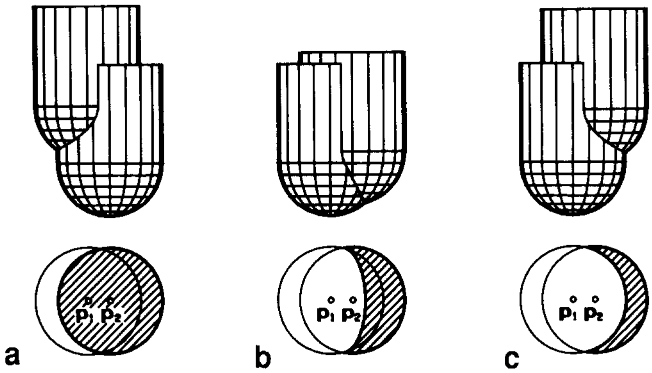

Figure 6. *Relationships between two adjacent tool volumes; (a) gouging, (b) interference with half sphere, (c) interference with shank*

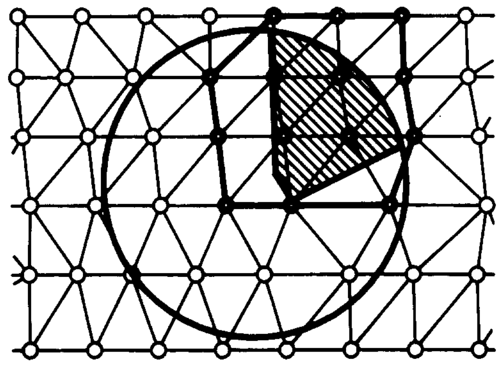

Figure 7. *Reduced CC region*

$$z_p = z_r + (R^2 - (x_p - x_r)^2 - (y_p - y_r)^2)^{1/2}$$

where R is the radius of the ball endmill. The highest of all the CM points calculated with the vertex points in the CC region becomes an initial CM point.

Once the initial CM point has been obtained, the CC region may be reduced one more time, and, for the calculation of an IFCM point with facets and edges, the triangular facets overlapped with the rereduced CC region are selected. Figure 7 shows an example of the reduced CC region with three neighboring IFCM points, where the crosshatched area is the reduced CC region, and the selected facets lie within the area enclosed by the heavy rules. The calculation process with a facet is discussed below. When a facet with three vertex points is selected, an implicit equation that represents the plane

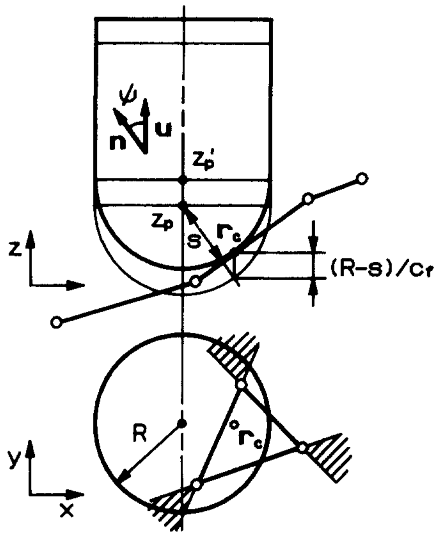

Figure 8. *CM-point calculation with facets*

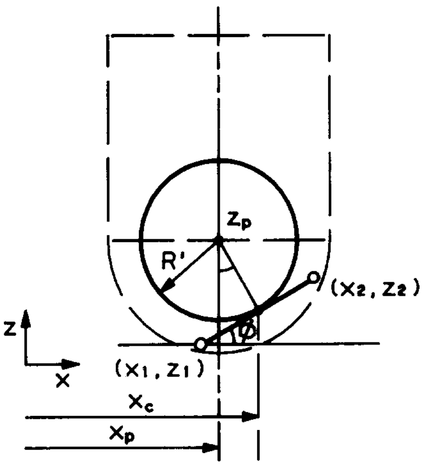

Figure 9. *CM-point calculation with edges*

containing the three points can be defined:

$$a_f x + b_f y + c_f z + d_f = 0 \tag{2}$$

The normal vector $\mathbf{n} = (a_f, b_f, c_f)$ of the plane is a unit vector whose direction points outwards from the polyhedron. With the current CM point $\mathbf{p} = (x_p, y_p, z_p)$ (the initial CM point for the first selected facet), the perpendicular distance from the plane to the CM point $\mathbf{p}$ is given by

$$s = a_f x_p + b_f y_p + c_f z_p + d_f$$

If the distance s is greater than or equal to the tool radius R, the tool at $\mathbf{p}$ cannot interfere with the facet, and another facet can be selected. Otherwise, the tool interferes with the 'infinite' plane, and it interferes with the facet if the CC point on that plane is within the triangular bounds of the facet. Figure 8 shows the 2D side view of a gouged facet. In this case, the CC point $\mathbf{r}_c = (x_c, y_c, z_c)$ on the plane is expressed in vector notation as

$$\mathbf{r}_c = \mathbf{p} - R\mathbf{n} + ((R - s)/c_f)\mathbf{u}$$

where c_f is the same as in Equation 2, and is the cosine of the angle between the plane normal vector $\mathbf{n}$ and the tool-axis vector $\mathbf{u}$. The test of whether the CC point $\mathbf{r}_c$ is within the triangular bounds is accomplished on the xy-plane domain. Let (x_1, y_1) and (x_2, y_2) be two projected vertex points of a facet, selected in anti-clockwise order. The line passing through the two points is defined as

$$fx + gy + h = 0$$

where $f = -(y_2 - y_1)$, $g = (x_2 - x_1)$ and $h = x_1 y_2 - x_2 y_1$. The definition ensures that the value t, given by

$$t = fx_c + gy_c + h \tag{3}$$

is greater than or equal to zero if the projected CC point (x_c, y_c) is located on the left-hand side of the line, or is on the line. Accordingly, as shown in Figure 8, the CC point $\mathbf{r}_c$ is within the bounds of the facet if the projected CC point (x_c, y_c) is located on the left-hand side of, or is on, every projected line of three edges (i.e. three of the values of t are greater than or equal to zero). If this is true, the z value of the CM point is updated as

$$z'_p = z_p + (R - s)/c_f$$

If one of the values of t in Equation 3 is negative, the tool may interfere with an edge of the facet. Suppose that there are two points (x_1, z_1) and (x_2, z_2) on the xz plane, and a circle of radius R' whose centre is on a vertical line $x - x_p = 0$, as shown in Figure 9. Then, the circle touches the 'infinite' line passing through the two points at a contact point whose x-coordinate value is given by

$$x_c = x_p + R' \sin \phi$$

where

$$\phi = \sin^{-1}((z_2 - z_1)/((x_2 - x_1)^2 + (z_2 - z_1)^2)^{1/2})$$

If $x_1 \leqq x_c \leqq x_2$, the circle touches the line segment joining the two points, and the z value of the circle centre point is determined as

$$z_p = z_1 + (x_c - x_1)\tan \phi + R' \cos \phi \tag{4}$$

In the case under discussion, if a vertical plane which contains the edge line is set and is assumed to be the xz plane of the above calculation, the z value of the CM point, where the tool touches the edge, can be computed. Here, the R' is the radius of the circle created by the intersection of the ball of the tool and the vertical plane, and it is given by

$$R' = (R^2 - (fx_p + gy_p + h)^2/(f^2 + g^2))^{1/2}$$

where the term $(fx_p + gy_p + h)^2/(f^2 + g^2)$ means the square of the normal distance from the projected edge line to the 2D CM point. If the new z_p of Equation 4 is greater than the current z_p, the z_p is updated.

If two values of Equation 3 are negative (in other words, if the projected CC point is in the marked area shown in Figure 8), the tool may interfere with a vertex point. However, the tool does not interfere with the vertex point in this case because the initial CM point was the highest point calculated with all the vertexes. There cannot be three negative values of Equation 3. With all the facets in the reduced CC region, the highest CM

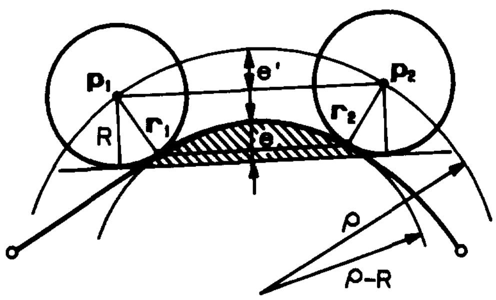

Figure 10. *Determination of step length of tool*

point, which is the IFCM point, is obtained and finally converted into an interference-free CL point.

Determination of step length and tool-path interval

The step lengths of CL points are determined by the same method as for the computation of the step sizes of points in the point-data generation step, which uses locally defined circles. The slight difference is that it uses two IFCM points ($\mathbf{p}_1$ and $\mathbf{p}_2$ in Figure 10) and their projected normal vectors on the plane containing the tool path to define the circles. The machining tolerance β is given. The e shown in Figure 10 is the cutting error. Note that e', the chordal deviation from the circle, has the same value as the cutting error e; this can easily be proved.

Path intervals are determined in two ways, one using the scallop height, and the other the constant intervals. In compound-surface machining, the latter is frequently used, because, with the former method, a small cliffy surface element may cause the generation of a large number of tool paths in a given range.

Other issues

In the author's tool-path-generation method, the tool may not be positioned exactly in the interference region, so that the part surface may not be cut completely in that region. For this case, a repeated midpoint-sampling scheme is used. The scheme is discussed briefly below (see Figure 11a). When the cutting error between two IFCM points ($\mathbf{p}_1$ and $\mathbf{p}_2$ in Figure 11) is identified as being abnormally great, a midpoint $\mathbf{p}_2^{(1)}$ is obtained, and the cutting error between $\mathbf{p}_1$ and $\mathbf{p}_2^{(1)}$ is computed. If the new cutting error is also unacceptable, a midpoint $\mathbf{p}_2^{(2)}$ between $\mathbf{p}_1$ and $\mathbf{p}_2^{(1)}$ is sampled. Otherwise, a midpoint between $\mathbf{p}_2^{(1)}$ and $\mathbf{p}_2$ is sampled, and so on. Sampling is stopped if the horizontal distance between the latest two points is less than a small value ε, and the last IFCM point can be taken for the exact tool position. An estimation method is used to place the tool precisely in the surface boundary, where the tool may easily either overcut the neighboring surfaces not defined in the current surface model, or not cut the part surface, as shown in Figure 11b.

EXAMPLES OF REAL APPLICATIONS

Two examples of real applications are provided in this section. Both are those of machining stamping dies for car-body panels. The first, shown in Figure 12, is the machining of the die of a fender panel. Figure 12a shows the compound-surface model of the punch and the blank holder of the die. It consists of 494 composite parametric surfaces, and the size is approximately 1430 × 1190 mm^2. The plotting of the point data generated from the surface is shown in Figure 12b, and the tool paths with a radius R of 15 are shown in Figure 12c. In fact, for a better rendering, only one of every four passes is drawn in the plottings of Figures 12b and c. The point data was generated with a sampling tolerance α of 0.01, and there were 285 340 points. It took 3 min 56 s (CPU time) for the generation on an IBM ES/9000 host computer. The tool paths have 229 651 CL points, produced with a machining tolerance β of 0.01 and a constant path interval of 2.5, and it took 10 min 12 s on the same computer.

Figure 13 shows the second example, which is an application to inner-panel dies. This consists in machining the dies of the fuel-tank panel of a car. The surface model, the plottings of the point data and the tool paths are shown in Figures 13a–c, respectively. The size is about 1260 × 820 mm^2. It took 2 min 21 s for the generation of the point data, and 7 min 46 s for the tool paths, under the same conditions as those for the first

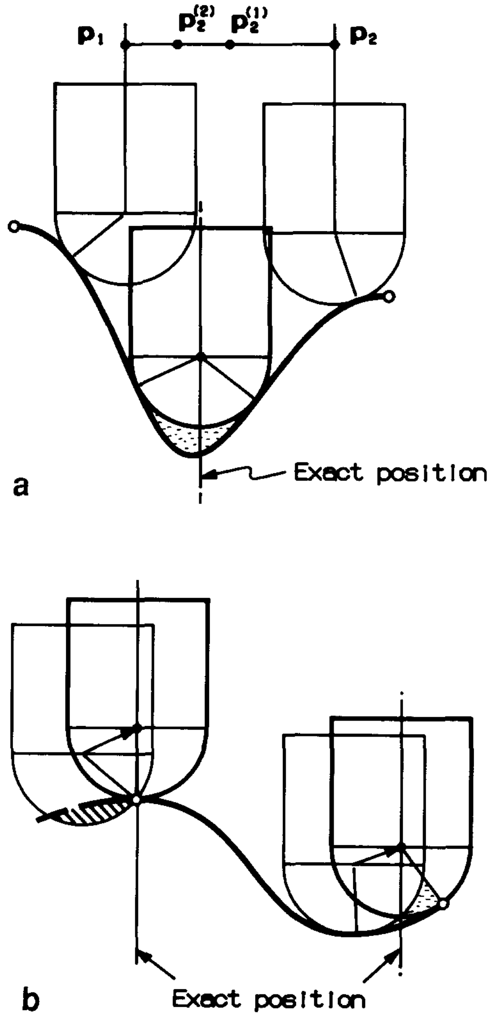

Figure 11. *Finding of exact tool positions*

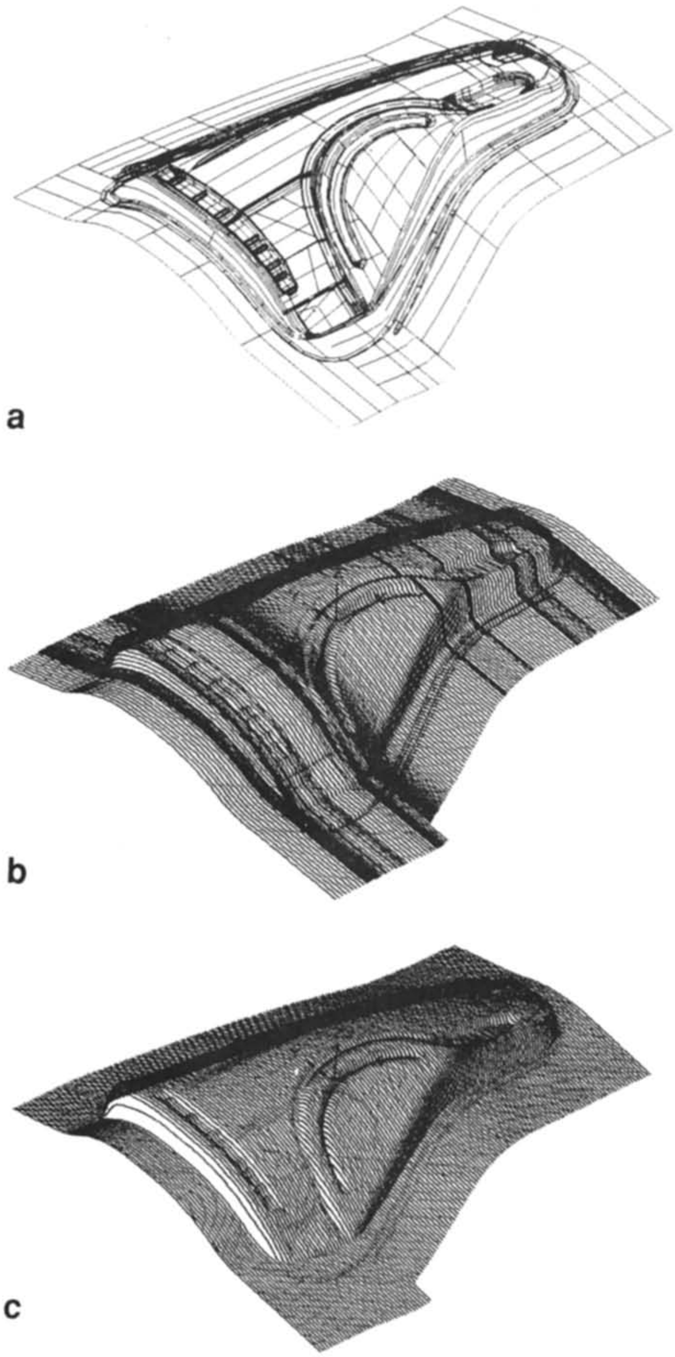

Figure 12. *Machining of fender die; (a) compound-surface model, (b) plotting of point data, (c) tool paths*

example. Figure 13d shows the tool paths for machining the upper part of the die. They can be generated from the same point data, but the coordinate transformation of the point data is needed first.

CONCLUSIONS

The method of machining parametric compound surfaces presented in this paper has been developed to meet practical needs. It is a robust method that can be used in the machining of any parametric compound surface, regardless of its complexity, and any type of interference can be avoided, such as in caved shapes. As is shown in

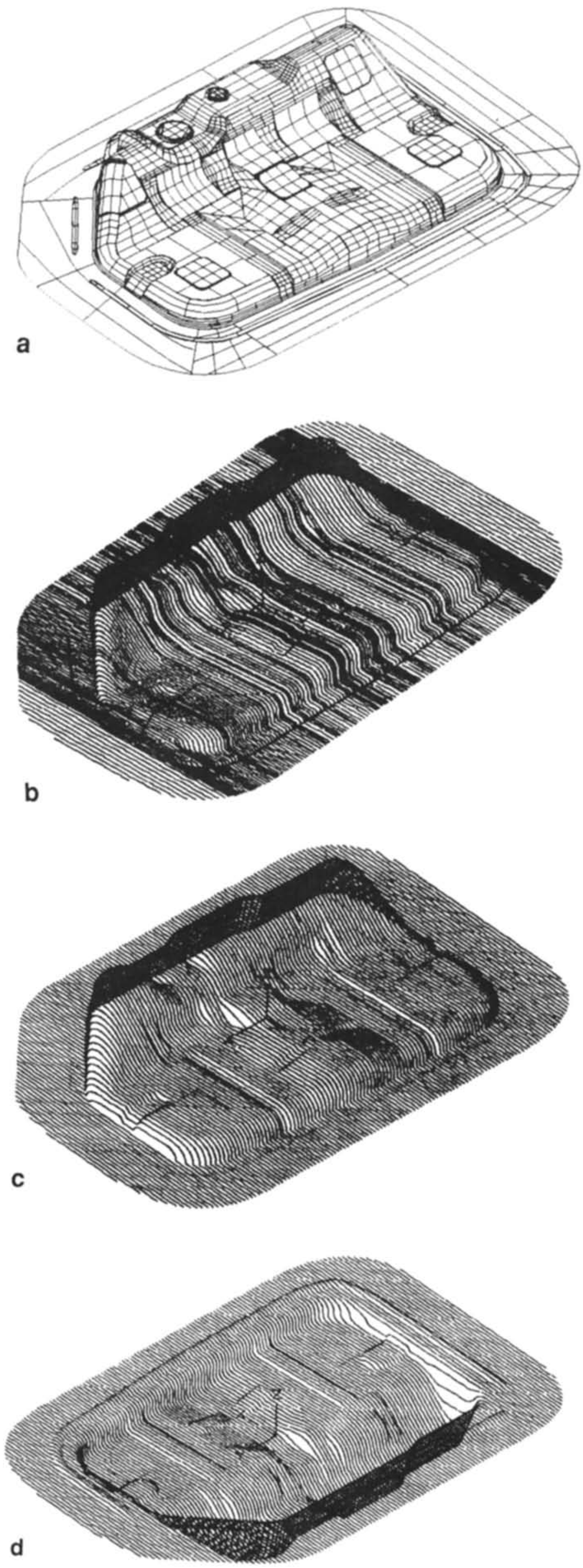

Figure 13. *Machining of fuel-tank dies; (a) surface model, (b) plotting of point data, (c) tool paths, (d) tool paths for machining upper part of die*

the above examples, the computing time is also short for a relatively large surface model with complicated shapes. Another possible advantage of the method is that the generated tool paths, which are planar, are appropriate for metal cutting.

The proposed algorithm for calculating CL points automatically prevents the part surface (the polyhedron) from gouging, in that the amount of gouging of the original surface cannot exceed $\alpha + \beta$, the sum of the point-sampling tolerance and the machining tolerance for the polyhedron. Therefore, given the machining tolerance γ for the original surface, the desired result can be obtained by setting α and β in accordance with the condition $\alpha + \beta < \gamma$.

This method (generating tool paths from the polyhedron) may easily be extended to machining using flat endmills or filleted endmills, which are often used in the rough cutting of metal to improve cutting efficiency. The author has developed a similar method for these types of tool, and this will be discussed when it has been fully implemented.

REFERENCES

1 **Duncan, J P and Mair, S G** *Sculptured Surfaces in Engineering and Medicine* Cambridge University Press, USA (1983)

2 **Choi, B K, Lee, C S, Hwang, J S and Jun, C S** 'Compound surface modelling and machining' *Comput.-Aided Des.* Vol 20 No 3 (1988) pp 127–136

3 **Loney, G C and Ozsoy, T M** 'NC machining of free form surfaces' *Comput.-Aided Des.* Vol 19 No 2 (1987) pp 85–90

4 **Bobrow, J E** 'NC machine tool path generation from CSG part representations' *Comput.-Aided Des.* Vol 17 No 2 (1985) pp 69–76

5 **Choi, B K and Jun, C S** 'Ball-end cutter interference avoidance in NC machining of sculptured surfaces' *Comput.-Aided Des.* Vol 21 No 6 (1989) pp 371–378

6 **Kondo, T, Kishinami, T and Saito, K** 'Machining system based on scanning data' *J. Jap. Soc. Precision Eng.* Vol 54 No 6 (1988) pp 1070–1075 (in Japanese)

7 *CATIA 3D Design – Geometry Interface Reference Manual* IBM, USA (1988)

8 **Scherrer, P K and Hillberry, B M** 'Determining distance to a surface represented in piecewise fashion with surface patches' *Comput.-Aided Des.* Vol 10 No 5 (1978) pp 320–324

BIBLIOGRAPHY

Hwang, J S 'Development of software for removing cutter interference in machining of free form surfaces' *Proc. Korea Soc. Automotive Eng. KSAE '90 Conf.* (Sep 1990) pp 27–31 (in Korean)

Ikeda, H, Amano, Y, Kawai, M and Sakuta, T 'Development of a complex free surface NC milling system for stamping dies' *Toyota Eng.* Vol 38 No 1 (1988) pp 64–75 (in Japanese)

Jun, G C 'A study of CL data generation method for Cartesian machining of sculptured surfaces' *MS Thesis* Korea Advanced Institute of Science and Technology, Korea (1987) (in Korean)

Lu, H, Miyoshi, T and Saito, K 'A CAD/CAM system based on P-map modelling method – concept and properties of P-map' *J. Jap. Soc. Precision Eng.* Vol 56 No 8 (1990) pp 1411–1416 (in Japanese)

APPENDIX

Estimation of deviation and determination of step length

As shown in Figure 14, let $\mathbf{r}_1$ and $\mathbf{r}_2$ be two consecutive points on a surface, and let $\mathbf{n}_2$ be the unit surface normal vector at the point $\mathbf{r}_2$. Then, a circle can be defined on the plane containing the two points and the vector. The radius of the circle is given by

$$\rho = (1/2)ts^2$$

where $s = |\mathbf{r}_2 - \mathbf{r}_1|$ and $t = |\mathbf{n}_2 \cdot (\mathbf{r}_2 - \mathbf{r}_1)|$, and the maximum deviation of the chord joining the two points from the circle is given by

$$e = \rho - (\rho^2 - s^2/4)$$

Given the tolerance α, the maximum allowable deviation, the next step length l is determined by

$$l = 2(2\rho\alpha - \alpha^2)^{1/2} \tag{5}$$

and the tangent distance for the next step is given by

$$d = l(\rho - \alpha)/\rho$$

Reduction of CC region

Let $\mathbf{p}_1$ be an IFCM point followed by a CM point $\mathbf{p}_2$. Then, the equation of a middle plane between the two

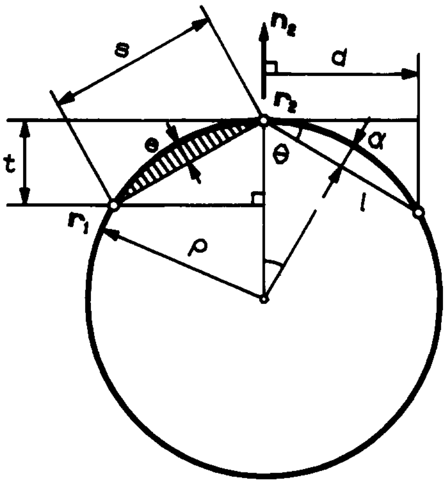

Figure 14. *Determination of step length*

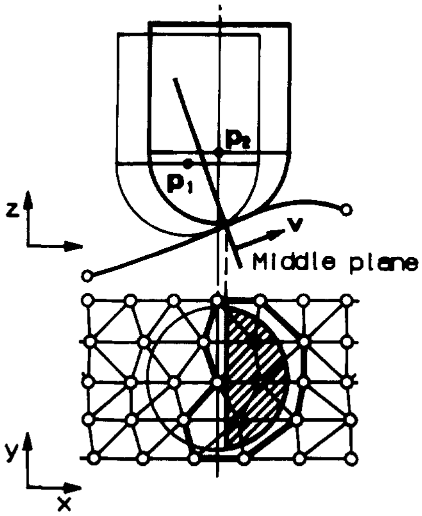

Figure 15. *Reduction of CC region*

points is given by

$$a_t x + b_t y + c_t z + d_t = 0$$

where a_t, b_t and c_t are the components of the vector $\mathbf{v} = \mathbf{p}_2 - \mathbf{p}_1$, and $d_t = -(1/2)\mathbf{v} \cdot (\mathbf{p}_1 + \mathbf{p}_2)$. If the two half spheres of tool volumes at the points intersect each other, the plane contains a circle created by the intersection of two half spheres, as shown in Figure 15. If a point $\mathbf{r} = (x_\mathbf{r}, y_\mathbf{r}, z_\mathbf{r})$ in the original CC region (the inner area of the circle on the xy plane) satisfies the expression

$$a_t x_\mathbf{r} + b_t y_\mathbf{r} + c_t z_\mathbf{r} + d_t > 0$$

it is in the reduced CC region on the CT face domain. The crosshatched area in Figure 15 is the reduced CC region, and the points marked by heavy circles are the points in that region. For the calculation of the IFCM point, the facets within the boundary marked by heavy rules are selected.