

Global Finish Machining Using CM²

Jianguo Wang¹, C. Greg Jensen² and W. Ed Red³

¹Brigham Young University, max.wang@byu.edu

²Brigham Young University, cjensen@byu.edu

³Brigham Young University, ered@et.byu.edu

ABSTRACT

Curvature matched machining (CM²) has been proved to be the most efficient five-axis machining algorithm. However, when CM² is used to generate tool paths for global machining of large regions of complex sculptured surface, e.g. the clay milling of a concept car, there will often be suboptimal CM² tool paths where the performance will be influenced greatly in some steep areas causing the lead angle of the tool to become larger to match the curvature and/or avoid gouging. By limiting the lead angle of the CM² tool while cutting these steep regions the benefits of speed, accuracy and surface finish can still be realized over large area of the surface, leaving only small uncut regions. This paper focuses on how to determine the uncut regions during the calculation of CM² tool paths and how to define the boundary of rework regions. Strategies for generating more efficiency CM² tool paths are also discussed. These methods were developed within the CATIA CAA environment and used to do finish global machining of a one-fourth scale Ford GT model.

Keywords: Five-axis, Curvature matched machining, CM², Global finish machining

1. INTRODUCTION

As the use of sculptured surface modeling grows, engineers will continue seeking better methods for sculptured surface machining. Some tool and die makers have found that, by changing from 3-axis to 5-axis milling, efficiency gains of 10 to 20 times could be achieved [1] [2]. If one can control an end mill to fit sculptured surface optimally, it would be possible to accomplish higher efficiency and better surface quality in finish sculptured surface machining [3] [4]. Flat-end mills are much more efficient at removing material than ball-end mill of the same size [5]. In recent years, many algorithms of tool path generation have been developed based on 5-axis flat-end milling.

For 5-axis machining, the strategy of tool axis orientation is chosen according to the part types and tool type. There are many types of cutters such as plain milling cutters, formed milling cutters, face milling cutters, and end milling cutters. However, end milling is often the only choice for sculptured surface machining (SSM). In 5-axis machining, the tool axis can be controlled by giving lead and tilt angle of the tool axis. For Sturz milling method, lead and tilt angle of the tool will be set to a constant value. For other tool axis strategies, these two angles are variable during the period of machining. The algorithm of curvature matched machining (CM²) serves to cater the curvatures of machining surface by changing the lead angle of the flat-end mill [3]. In addition, the method of CM² can reach the goals of improving surface finish and achieving design surface dimensions more accurately.

For the finish SSM, it is possible that some tool positions calculated using the algorithm of CM² interfere with the design surface. The algorithm used to generate NC tool paths must have the ability to avoid the interference between the tool and design surface. In addition, the tool position and orientation generated using the algorithm of CM² must be free from collisions between the spindle and machine tool fixtures. These two types of behaviors are respectively referred to as local and global gouging avoidance.

Gouging avoidance makes CM² impossible in some regions. When a large part is machined using CM² algorithm, gouging avoidance in some areas causing the lead angle to be extreme and the overall density of tool paths to increase greatly, thus causes the advantages of CM² to be minimized and CM² tool path to approach the density of a ball end mill. If the upper lead angle limit of the tool axis is restricted, it is anticipated that some material will not be removed and will require reworking these areas with smaller tools or other SSM methods. In a word, an effective solution for Global CM² is not currently available with today's CAD/CAM packages.

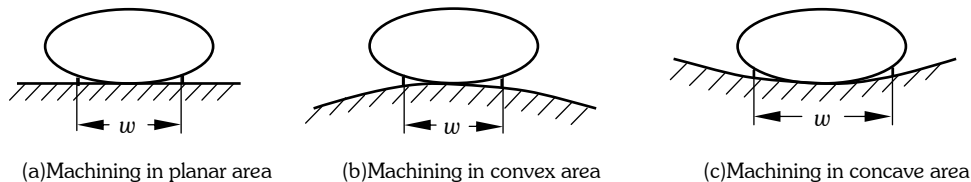
This paper reports on research focused on determining a global CM^2 machining method with improved local and global gouge avoidance. This global CM^2 method will have the added benefit of higher machining efficiencies over what is currently offered by the CAD/CAM vendors. An automated rework algorithm will also be developed that will more efficiently machine regions that were avoided by the global CM^2 . These rework areas will be machined using the most efficient tool path strategy. The main goal of this research is to find a method and algorithm for computing the rework boundary and its offset while applying optimal CM^2 methods to the largest possible region(s).

2. METHODS

As discussed previously the machining of large surface regions with complex shapes are divided in many different machining regions according to their different shape characteristics. The programming of these regions is generally finished region by region. Sometimes local optimization of these region-by-region tool paths is feasible to improve the overall efficiency. The drawback of this method is that the programming is not automated and is very time-consuming although computing the tool paths for this traditional region-by-region machining is less time consuming. For traditional global CM^2 machining, in order to completely finish the machining of the complex shape, the density of tool path for global machining is very high, making the overall machining time much longer than it needs to be. Programmers are in a dilemma wanting to use global machining strategy, but without the long programming and machining times. The main focus of this paper is on how to optimize tool paths for global machining. The following sections present the principles of global-local CM^2 .

2.1 Limits of Improving Machining Efficiency

Fig.1 shows that the same effective edge of an inclined tool has a different machining strip width when it is used for machining different areas. Fig.1 (c) is the case of CM^2 . So it has the largest strip width, that is, it has the best machining performance. The machining strip width is essentially determined by the diameter of the tool, scallop height, and the lead angle or local curvature of part surface at the cutter contact point. In convex region, it is also influenced by the geometry shape. Both Fig.1 (a) and (b) have smaller machining strip width.



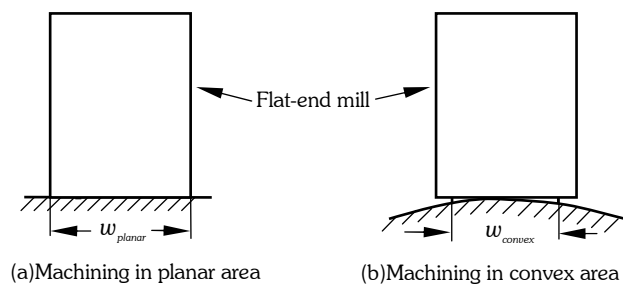
The direction of the tool motion is perpendicular to the paper

Fig.1. Different machining strips for the same effective edge of the tool bottom.

For CM^2 application, Fig.2 (a) and (b) are the best result for machining the regions that are planar or convex in the direction of the stepover of tool path. Even so, the strip width is very small when the machining region is steep convex in the direction of stepover. For the machining region that is planar in the direction of stepover, the larger the diameter of tool, the bigger is the strip width. CM^2 used in concave regions can achieve the largest machining strip. When there is a gouging avoidance in CM^2 , the situation of Fig.1 (a) and (b) will happen and the strip width in Fig.1 (c) will also be decreased.

From the points mentioned above, the factors that restrict the improvement of CM^2 efficiency are:

- large lead angle or gouging avoidance in the planar region or in the convex area with steep shape in the direction perpendicular to the tool motion
- steep convex shape in the stepover direction
- gouging avoidance in concave region



The direction of the tool motion is perpendicular to the paper

Fig. 2. The desired machining strip.

2.2 Determining the Tool's Diameter (D)

In order to find how much the diameter D of the appropriate tool is, the following two factors have to be considered.

a) The curvature distribution of part surfaces in the concave region for global-local CM^2

$$D \leq 2 \sin \alpha_M / k_{max} \tag{1}$$

where k_{max} is the maximum curvature in the concave machining area and α_M is the upper limit of lead angle

b) The geometry shape in the stepover direction if the shape is convex

$$D > w_{convex} \tag{2}$$

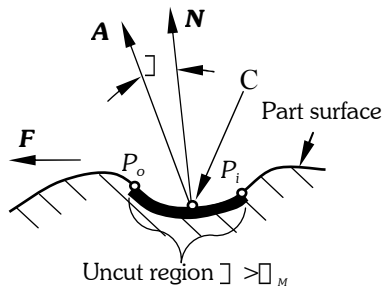
where w_{convex} is machining strip in the convex regions.

In general, increasing the diameter of the tool causes the increase of machining strip width. Based on formula (2), when the diameter of the mill is larger than a specific value (w_{convex}), increasing the diameter of the tool is invalid for expanding the machining strip width for machining the convex region. In addition, more uncut material may be left in the concave region with bigger curvature when the part is machined by a tool with too large diameter. For actual machining, the tool diameter is often determined by referring to formula (1). When $D > 2 \sin \alpha_M / k_{max}$, there will be uncut material left, and the larger D is, the more uncut material is left. The determination of D is based on the balance of cut and uncut regions. Adaptive balance can achieve a higher machining efficiency.

2.3 Elimination of the Limits

The geometry shape in the stepover direction should be as flat as possible if the shape is convex. In practice, it is difficult to judge what the best direction is. Most often, the direction with the longest segment of tool path is selected by programmers. For global-local CM^2 , the upper limit of lead angle should be set to be small enough so that the curvature of the effective edge of the tool bottom becomes the limits of lowering the density of tool path. This may result in uncut material left in the regions where a bigger lead angle should be given for gouging avoidance or curvature matching. If the method of determining the rework regions is obtained, the strategy of improving machining efficiency by limiting lead angle is feasible. Said another way, the global-local CM^2 is first implemented by the main global CM^2 with smaller upper limit of lead angle, and then by machining the uncut regions using the adaptive methods, such as ball-end milling or local CM^2 .

2.4 Determination of Rework Regions



- A:** Tool axis direction
- N:** Normal of the machining surface at contact points
- F:** Feed direction
- α : Lead angle of the tool
- C:** Cutter contact point
- P_o : The contact point on the boundary of the uncut region where the tool moves in the uncut region
- P_i : The contact point on the boundary of the uncut region where the tool moves out the uncut region

Fig. 3. Uncut regions.

By comparing the upper limit of lead angle (α_M) to the one (α) with which the effective edge of the tool bottom matches the curvature of the part surfaces at the cutter contact point, the uncut regions can be determined. All the regions where α_M is smaller than α are uncut (see Fig.3). α_M can also be calculated using the equation below. But in general, it should be determined by programmer.

$$\alpha_M = \arcsin(k_M D / 2) \tag{3}$$

where k_M is the maximum curvature which effective edge of the tool bottom can match for the actual machining because of the limit of the lead angle. It is determined by analyzing the curvature distribution in the region that is concave in the direction of stepover, and in the planes that are perpendicular to the direction of stepover. α can be calculated using equation below

$$\alpha = \arcsin(|\mathbf{c}|D/2) \tag{4}$$

where \mathbf{c} is all the possible curvature of part surfaces in the planes perpendicular to the tool motion at contact points (see Fig.4). \mathbf{c} can be calculated using equation below

$$|\mathbf{c}| = |\mathbf{k}_1| \cos^2 \theta + |\mathbf{k}_2| \sin^2 \theta \tag{5}$$

After substitute \mathbf{c} into Eqn. (4), the lead angle can be calculated if gouging avoidance is not considered

$$\alpha = \arcsin((|\mathbf{k}_1| \cos^2 \theta + |\mathbf{k}_2| \sin^2 \theta) \cdot D / 2) \tag{6}$$

Using the toolkit API of CAD packages, principle curvatures, \mathbf{k}_1 and \mathbf{k}_2 , may be easily obtained. Therefore, k_M can be determined by referring to the value of $|\mathbf{c}|$ in the machining region. The value of k_M should just be larger than $|\mathbf{c}|$ in the regions which are machined by the main global CM^2 , or less than $|\mathbf{c}|$ in the small uncut regions. Said another way, $|\mathbf{c}|$ is larger than k_M in uncut regions. Therefore, the larger k_M , the smaller the uncut regions. Fig.5 shows that, using CATIA V5, $|\mathbf{k}_1|$ can be directly obtained from the analysis result of -0.0341 and $|\mathbf{k}_2|$ is equal to -1/58.291. Now that the value of k_M is available, the radius of the flat-end mill used for finishing the main global CM^2 may be determined with the equation below.

$$D = \frac{2 \sin \alpha}{k_M} \tag{7}$$

Eqn. (7) also shows that the larger k_M is, the smaller the tool radius. Therefore, the smaller the tool radius, the smaller and fewer the resulting uncut regions, however, Lee, Y.S. [6] [7] and formula (2) suggest, the smaller the tool radius, the higher the density of global CM^2 tool path. In addition, if the tool radius is too large, the total area or number of uncut regions will be large, thus causing extensive and time consuming tool path planning. The best tool radius should be determined using optimization. The determination of the ideal tool radius is recommended as a future research topic. In this paper, the selection of tool radius is based on the number of uncut regions, where the number does not exceed 2. The tool path of CM^2 created by CATIA V5 does not contain the parameters of θ and \mathbf{c} , but contains the normal vector (\mathbf{N}) of part surfaces and the tool axis vector (\mathbf{A}) at each contact point (see Fig.3). Therefore, θ can be calculated in terms of the equation below

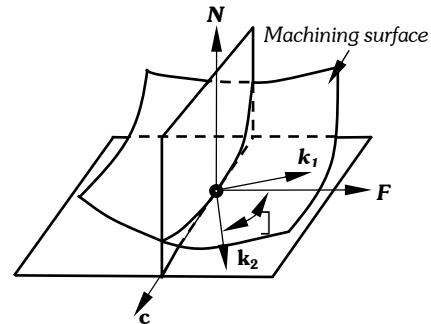
$$\theta = \arccos\left(\frac{\mathbf{A} \cdot \mathbf{N}}{|\mathbf{A}| |\mathbf{N}|}\right) \tag{8}$$

3. DEVELOPMENT

The method described above was fully implemented for the global-local CM^2 . To compute the boundary of the rework regions, tool path of assumed previous CM^2 (AP- CM^2) should be first generated using the method of CM^2 in CATIA V5 R13.

3.1 Tool Path Planning

Comparing to various other multi-axis tool path planning strategies, multi-axis sweeping strategy is more feasible for global machining of a quilt of complex surfaces. For this strategy, the tool paths are parallel to the plane that is defined by the programmer. In this paper, the tool path for both AP- CM^2 and actual global CM^2 are created using this strategy.



\mathbf{k}_1 and \mathbf{k}_2 are the principle curvature of the part surface at the contact point

Fig. 4. The relationship between a curvature and the principle curvature.

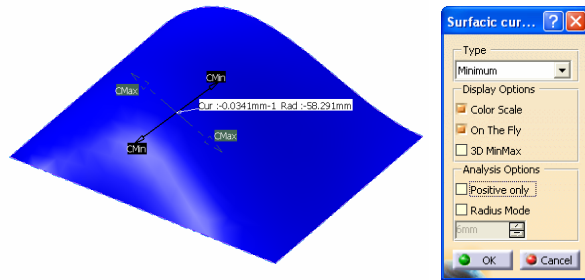


Fig. 5. Analysis of curvature distribution.

3.2 Data Structure and Tool Path Dumping

In this paper, the boundary points are extracted from tool path of AP-CM². They should sit on or be as close as possible to the machining surfaces. They are computed based on the information of the contact points, the normal of machining surfaces and the vector of tool axis at each contact point. To easily locate the individual contact point, an index should also be included. The *struct* type of *MyTotalToolPath* is defined in the program as the following.

```

struct MyTotalToolPath
{
    double LocalAxis[3];
    double ContactPoint[3];
    double ContactNormal[3];
    int ToolPathNb;
    int Start;
    int End;
};
    
```

The variable is declared as the type of *MyTotalToolPath* for holding all pertinent tool path information. *ToolPathNb* can be used for determining which trajectory a contact point belongs to. Tool axis is expressed by *LocalAxis*. Normal of machining surface is expressed by *ContactNormal*. Contact points are expressed by *ContactPoint*. *Start* and *End* are used to determining the number of the first and the last contact point of a trajectory. These items are very important for computing the rework region.

3.2 Data Screening

Each point (P_i or P_o) (see Fig.3) sitting on the boundary of uncut regions meets the condition below

- The lead angle is less than α_M .
- The lead angle at the contact point next to P_i is larger than α_M (see Fig.3).
- The lead angle at the contact point previous to P_o is larger than α_M (see Fig.3).

If the uncut region exists in the side of the machining surfaces (see Fig.6), in the case of Fig.6 (a), P_o will be on the boundary of the machining surfaces, and in the case of Fig.6 (b), P_i may be on the boundary of the machining surfaces. Substituting *LocalAxis* and *ContactNormal* into Eqn. (8), the lead angle can be calculated using the equation below

$$\square = \arccos\left(\frac{LocalAxis \cdot ContactNormal}{|LocalAxis| |ContactNormal|}\right) \quad (9)$$

In order to compute multiple uncut regions, P_i or P_o should be given index numbers. In general, the distribution of points of P_i and P_o is irregular. The boundary of rework regions cannot be generated by simply connecting the adjacent points of P_i or P_o . A rework region should be surrounded by smooth curves. All the points of P_i and P_o should be on the boundary or outside of the actual uncut region.

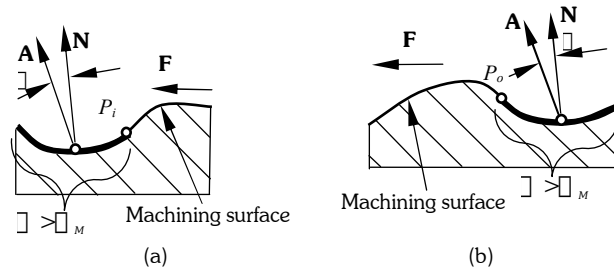


Fig. 6. The uncut regions in the side of the machining surfaces.

3.2.1 Determination of the Tool Motion Direction

To help optimize the boundary of rework regions in the future, each contact point on the boundary will be attached with a direction variable of *MotionDirection*. The value of this variable may be “Forward” or “Backward.” It is determined by the intersection angle (φ_b) between the direction of the cross product of the view direction (\mathbf{V}) and the machining direction (\mathbf{F}), and the one of the cross product of \mathbf{V} and vector \mathbf{M}_b (see Fig.7). Both \mathbf{V} and \mathbf{F} are defined by programmers. The planes which a tool path trajectory exists in are defined by \mathbf{V} and the machining direction of multi-axis sweeping tool path planning. Both \mathbf{F} and \mathbf{V} are set by programmers. \mathbf{V} can be extracted from tool path and \mathbf{F} can be obtained from the dialog box of Settings of Rework Region (see Fig.12). \mathbf{M}_b is the vector from point P_i to point P_o (see Fig.7).

$$\varphi_b = \cos^{-1}\left(\frac{(\mathbf{V} \otimes \mathbf{F}) \cdot (\mathbf{V} \otimes \mathbf{M}_b)}{|\mathbf{V} \otimes \mathbf{F}| |\mathbf{V} \otimes \mathbf{M}_b|}\right) \quad (10)$$

where “ \otimes ” denotes the cross product of two vectors and “ \bullet ” denotes the dot product of two vectors. φ_b can be calculated in terms of Eqn. (10). If φ_b is less than 90 degrees, the tool motion direction at the point of P_i and P_o is “Forward.” Otherwise, it is “Backward.” In addition, each sub trajectory is also assigned a direction variable of *PathDirect*. The value of this variable is determined by the intersection angle (φ) between the direction of the cross product of the view direction (\mathbf{V}) and the machining direction (\mathbf{F}), and the one of the cross product of \mathbf{V} and vector \mathbf{M} (see Fig.8). \mathbf{M} is the vector from the point of “Start point” to the point of “End point” of a sub trajectory (see Fig.8). φ can be calculated in terms of Eqn. (11). If φ is less than 90 degrees, the tool motion direction of a sub trajectory is “Forward.” Otherwise, it is “Backward.”

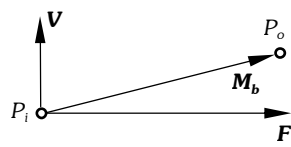


Fig. 7. The tool motion direction of a pair of contact points on the boundary.

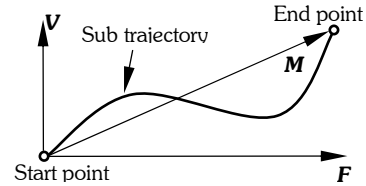
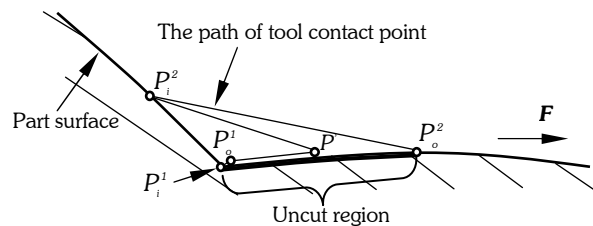


Fig. 8. The tool motion direction of a sub trajectory.

$$\varphi = \cos^{-1} \left(\frac{(\mathbf{V} \otimes \mathbf{F}) \bullet (\mathbf{V} \otimes \mathbf{M})}{|\mathbf{V} \otimes \mathbf{F}| |\mathbf{V} \otimes \mathbf{M}|} \right) \quad (11)$$

3.2.2 Pairing point P_i and P_o

P_i and P_o always show up as pairs (see Fig.3). They are related to the direction of tool motion that a contact point on the boundary of the uncut regions is which of P_i and P_o . For an ideal case, there should always be a point P_i and P_o on a sub trajectory that goes across an uncut region. If the tool motion direction at a contact point on the boundary of the uncut region points into this region, this contact point is P_i . If the tool motion direction at a contact point on the boundary of the uncut region points out of this region, this contact point is P_o .



The tool moves along point P_i^1, P_o^1, P_i^2 and P_o^2

Fig. 9 Uncut regions.

In the actual uncut region of a sub trajectory, there is often more than one pair of P_i and P_o . Fig.9 shows that there are two pairs of P_i and P_o . One is a pair of P_i^1 and P_o^1 , and the other is a pair of P_i^2 and P_o^2 . Program is capable of detecting if they belong to two different rework regions. Obviously, there are abundant points of P_i and P_o in Fig.9. They should be deleted.

A pair of P_i and P_o has not only the same tool motion direction, but also the same tool path index. All the pairs of P_i and P_o where the tool motion direction is different from the one of sub trajectory, said another way, where both φ_b and φ are not less or larger than 90 degrees at the same time, will be considered as irregular points. These points will be deleted and not be used to create the boundary curves of the rework regions. The ideal pairs of P_i and P_o should have the same tool motion direction as the sub trajectory which they are sitting on.

3.3 Determination of the Effective Boundary Points

Although the paired points of P_i and P_o sit around the uncut region, the boundary curve created by connecting them directly is wavy. Fig.10 shows that just connecting the effective points of P_o and P_i is a good solution. A rework region is encircled by two major curves. If the points of P_i at both ends of two major curves are included, the boundary curves with such a connection will be better. If the end point of one major curve is P_i , the end point of the other curve should be P_o , and vice versa (see Fig.10). Another issue is, beginning from one end point, how to find the subsequent points of P_o . How can the broken polylines of major boundary curves shown in Fig.10 be created automatically? How can the major boundary curves be generated in the case of

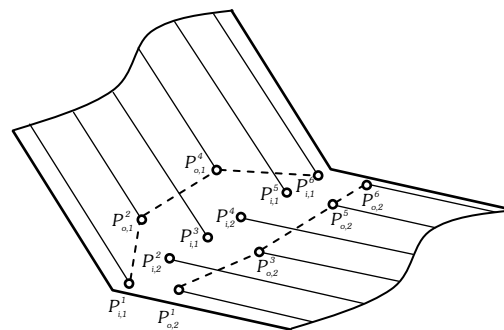


Fig. 10 Major boundary curves.

multiple rework regions? $P_{i,1}^1$ is the first point of the first polylines. $P_{o,2}^1$ is the first point of the second polylines. If the tool motion at the paired points of $P_{i,1}^1$ and $P_{o,2}^1$ is “Forward” (or “Backward”), the subsequent points of the first polylines except the last point should be the points of P_o with the tool motion direction of “Backward” (or “Forward”) and the following points of the second polylines except the last point should be the points of P_i with the tool motion direction of “Forward” (or “Backward”). If the tool motion at the last paired points of $P_{i,1}^n$ and $P_{o,2}^n$ for the first rework region is “Forward” (or “Backward”), $P_{i,1}^n$ (or $P_{o,2}^n$) is the last point of the first polylines and $P_{o,2}^n$ (or $P_{i,1}^n$) is the last point of the second polylines.

3.4 Offset of the Boundary of Uncut Regions

An adaptive outward offset of the boundary of rework regions ensures that there will be no uncut material left after the machining of rework regions. The points on the offset of the boundary should be as close as possible to the machining surface. They may be found on the polylines that are connected by the contact points on the same sub trajectory. The polyline length from the original effective boundary points (P_i or P_o) to the point P_b on the offset boundary is equal to the value of offset (see Fig.11 and Eqn. (12)). P_b can be calculated using Eqn. (13) and (14).

$$|P_i P_{i-1}| + \dots + |P_{i-k} P_b| = |P_o P_{o+1}| + \dots + |P_{o+k} P_b| = offset \tag{12}$$

In the case of Fig.11 (a),

$$P_b = P_{i-k} + (P_{i-k-1} - P_{i-k}) \frac{offset - (|P_i P_{i-1}| + \dots + |P_{i-k+1} P_{i-k}|)}{|P_{i-k-1} P_{i-k}|} \tag{13}$$

In the case of Fig.11 (b),

$$P_b = P_{o+k} + (P_{o+k+1} - P_{o+k}) \frac{offset - (|P_o P_{o+1}| + \dots + |P_{o+k-1} P_{o+k}|)}{|P_{o+k+1} P_{o+k}|} \tag{14}$$

Now, the boundary curves of rework regions can be generated by connecting these P_b points.

4. RESULTS

The purpose of this research is to implement the boundary computation of rework regions for global-local CM². The experimental results of global-local CM² and the computation of rework regions are presented below.

4.1 Tool Path Generation for AP-CM²

In order to compute the rework regions, the tool path of AP-CM² with large lead angle should be created first. In this example, there are two parts that are used for the test of global-local CM². For part A, the tool path of AP-CM² was generated with the lead angle of 45 degrees, the scallop height of 0.1mm and the machining tolerance of 0.01mm. For part B, the tool path of AP-CM² was created with the lead angle of 8 degrees, the scallop height of 0.2mm and the machining tolerance of 0.3mm.

4.2 The Computation of the Boundary of Rework Regions

When the function of parameter settings is activated, the dialog box of Settings of Rework Region will be opened (see Fig.12). In the box of Lead angle of previous CM², the lead angle for actual machining, α_m , should be entered. The value of offset should be entered in the box of Offset of rework boundary and the machining direction vector for the global-local CM² should be typed in the box of Machining Direction. The parameters set above will be used in the computation of rework boundary. To compute the boundary of rework regions, the function of computation should be activated. Then, based on the tool path of AP-CM² and the parameters set above, the boundary of rework regions will be computed. Fig.13 (a) and (d) show the rework boundary for the global-local CM² of part A and the rework boundary for the global-local CM² of part B.

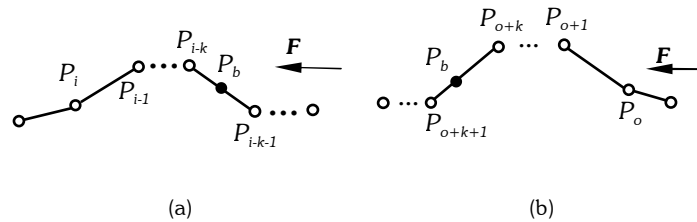


Fig. 11. The offset boundary of the uncut regions.

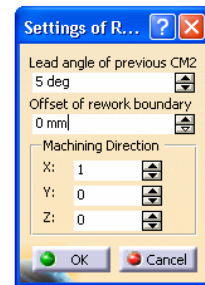


Fig. 12. The dialog box of Settings of Rework Region.

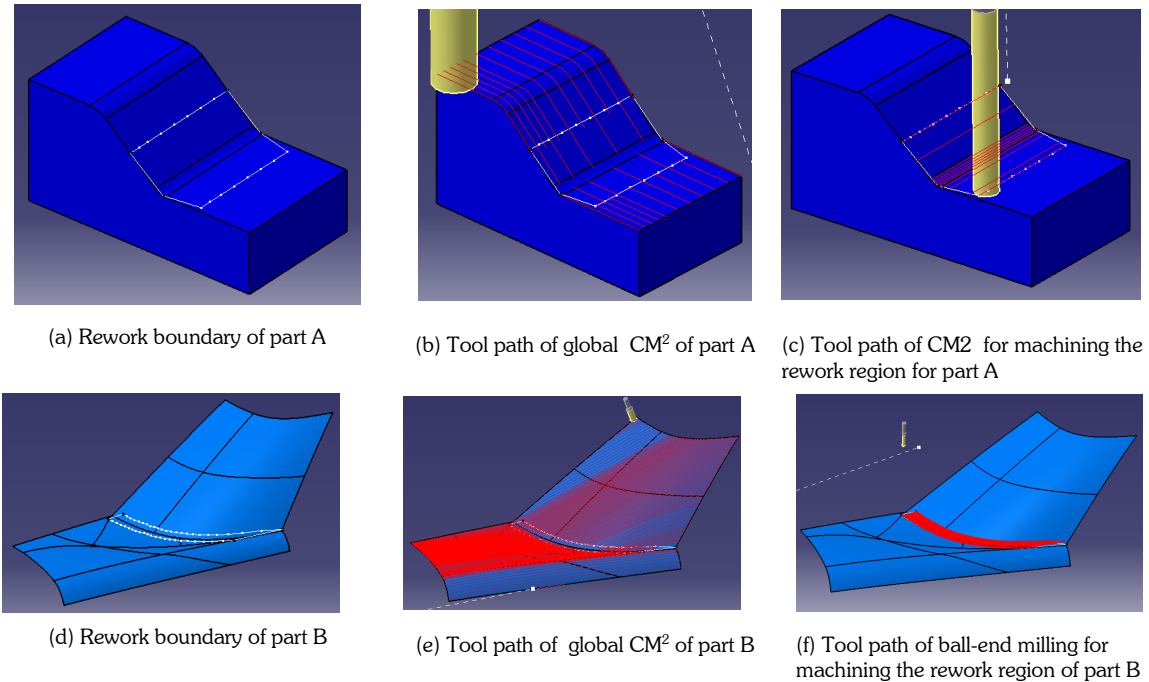


Fig. 13. The boundary of rework regions and the tool path of global-local CM^2 .

4.3 Tool Path Generation of Global-local CM^2

Now that the boundary of rework region has been created, the tool path of global CM^2 machining with the limit of lead angle may be generated with the lead angle of α_m (see Fig.13 (b) and (e)). For part A, the rework region can be machined using CM^2 with a smaller flat mill (see Fig.13 (c)). For part B, the rework region can be machined with a ball mill (see Fig.13 (f)). For both parts, the machining direction is the same as the one of their AP- CM^2 .

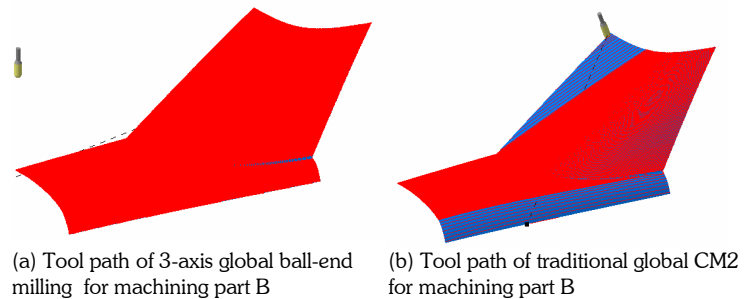


Fig. 14. Tool path of traditional global machining.

4.4 The Comparison of Different Global Machining Strategies

In order to compare global-local CM^2 to other traditional global machining strategies, the tool path of global ball-end milling and traditional global flat-end mill CM^2 of part B is generated using CATIA V5 (see Fig.14 (a) and (b)). All of the specifications used are listed in Table.1. This table also clearly shows that global-local CM^2 has the overall best machining performance.

4.5 Final Test—Machining One Quarter Scale of GT40 Model

To demonstrate the robustness of our method, we applied global-local CM^2 to a $\frac{1}{4}$ scale model of the GT40. Fig.15 shows the global-local CM^2 tool paths. Fig.16 shows the rework tools and Fig.17 shows the resulting clay model. The total time for machining the top portion of the GT40 was 94 minutes.

Table. 1. Comparison of different global machining.

		Traditional ball mill global machining	Traditional flat mill CM^2 global machining without the limit of lead angle	Global-local CM^2 machining
Tool (Diameter): (mm)	Global Rework	Ball mill: 25.4 Ball mill: 6.35	Flat mill: 25.4	Flat mill: 25.4 Ball mill: 6.35
Machining tolerance: (mm)		0.03	0.03	0.03
Scallop height: (mm)		0.05	0.05	0.05
Feed: (mm/minute)		1000	1000	1000
Spindle speed: (RPM)		3000	3000	3000
Machining time: (minute)	Global Rework	549 Neglected	256	76 25
Total machining time: (minute)		> 549	256	101

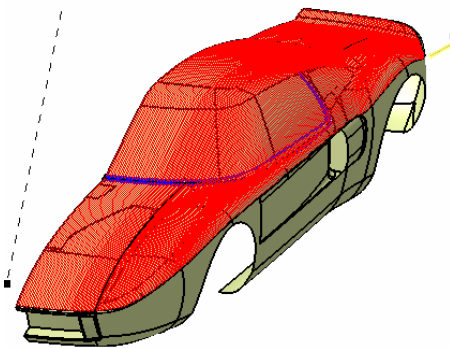
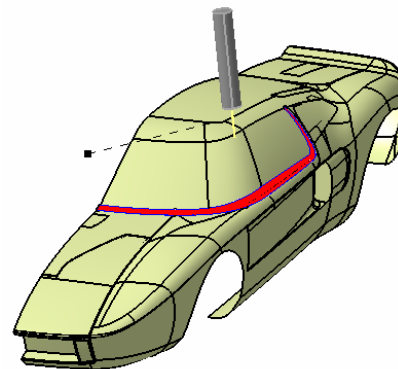
Fig. 15. Main tool path of global-local CM^2 Fig. 16. Rework tool path of global-local CM^2 **Main global-local CM^2 machining****Machining Tolerance:** 0.03 mm**Scallop height:** 0.05 mm**Lead Angle range:** 7 degrees**Flat mill diameter:** 6 mm**Feed:** 150 inches/minute**Machining time:** 90 minutes**Rework machining****Machining Tolerance:** 0.03 mm**Scallop height:** 0.05 mm**Ball mill diameter:** 3 mm**Feed:** 200 inches/minute**Machining time:** 4 minutes

Fig. 17. Actual machining of the top portion of GT40.

5. CONCLUSIONS

This paper has shown that the rework regions can be determined for CM^2 , and that global-local CM^2 is a feasible strategy for improving the efficiency of machining a quilt of sculptured surfaces. The concept and principle for global-local CM^2 were presented. The methodology about how to determine the parameters used in computation were introduced. The strategies for determining the boundary points of rework regions and obtaining the offset of the rework boundary were developed. The application for extracting AP- CM^2 tool path and a GUI for setting the parameters for global-local CM^2 were coded using CATIA V5 CAA. The comparison of different global machining was also given along with the results of global-local CM^2 milling applied to a 1/4 scale GT40 model.

The main advantages of global-local CM^2 are:

- Programming time for CM^2 tool path planning has been significantly reduced when machining a quilt-work of sculptured surfaces.
- Determining the uncut regions for global CM^2 vehicle is now completely automated.
- Machining efficiency is significantly improved.

Future work will focus on making our algorithms more robust so that CM^2 tool paths that generate highly irregular rework regions can easily and correctly be machined during the rework phase. Currently, we handle this problem by partitioning the regions or changing the machining parameters.

6. REFERENCES

- [1] Kim, B. H., et al, Effect of Cutter Mark on Surface Roughness and Scallop Height in Sculptured Surface Machining, *Computer Aided Design*, Vol. 26, No. 3, 1994, pp 179-188.
- [2] Vickers, G. W., and Quan, K. W., Ball-mills versus End-mills for Curved Surface Machining, *Journal of Engineering for Industry*, Vol. 111, 1989, pp 22-26.
- [3] Jensen, C. G., et al, Accurate Tool Placement and Orientation for Finish Surface Machining, *Proceedings of the Symposium on Concurrent Engineering*, 1992.
- [4] Marciniak, K., *Geometry Modeling for Numerically Controlled Machining*, Oxford University Press, 1991.
- [5] Loverton, T., et al, On the Use of Computer Based Simulation to Reduce Die Manufacturing Lead Times, IBEC'93, Body Assembly and Manufacture, pp 99-105, 1993.
- [6] Lee, Y. S., Adaptive Tool Path Planning for 5-Axis Sculptured Surface Machining by Machining Strip Evaluation, *Sculptured Surface Machining Conference IFIP TC5 WG5.3*, 1998.
- [7] Lee, Y. S., Non-isoparametric Tool Path Planning by Machining Strip Evaluation for 5-axis Sculptured Surface Machining, *Computer Aided Design*, Vol. 30, No. 7, 1998, pp 559-570.