



All Theses and Dissertations

2007-02-14

Tessellated Surface Machining

Jiewu Lu

Brigham Young University - Provo

Follow this and additional works at: <https://scholarsarchive.byu.edu/etd>

 Part of the [Mechanical Engineering Commons](#)

BYU ScholarsArchive Citation

Lu, Jiewu, "Tessellated Surface Machining" (2007). *All Theses and Dissertations*. 1656.
<https://scholarsarchive.byu.edu/etd/1656>

This Dissertation is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in All Theses and Dissertations by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.

TESSELLATED SURFACE MACHINING

by

Jiewu Lu

A dissertation submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Mechanical Engineering

Brigham Young University

April 2007

Copyright © 2007 Jiewu Lu

All Rights Reserved

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a dissertation submitted by

Jiewu Lu

This dissertation has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

_____	_____
Date	C. Greg Jensen, Chair
_____	_____
Date	W. Edward Red
_____	_____
Date	Robert H. Todd
_____	_____
Date	Carl D. Sorensen
_____	_____
Date	Jordan J. Cox

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the dissertation of Jiewu Lu in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Date

C. Greg Jensen
Chair, Graduate Committee

Accepted for the Department

Matthew R. Jones
Graduate Coordinator

Accepted for the College

Alan R. Parkinson
Dean, Ira A. Fulton College of Engineering
and Technology

ABSTRACT

TESSELLATED SURFACE MACHINING

Jiewu Lu

Department of Mechanical Engineering

Doctor of Philosophy

Sculptured surface machining (SSM) is the end milling of sculptured surfaces with multi-axis machine tools. SSM consists of the tessellated model machining (TMM) and the parametric model machining (PMM). The former is gaining more and more attention over the past decade because it offers many advantages over the latter.

New methods for various machining stages of TMM are proposed in this dissertation. First, in the 5-axis finish machining stage, a 3-dimension Configuration-space (3D C-space) machining method is presented. Next, in clean-up machining stage, an effective and creative approach is introduced. Finally, a complete TMM system is developed.

The TMM system is developed with C++ on Windows platform. Benchmarks are used to test the methods proposed in this dissertation. The results show that methods are accurate and efficient.

ACKNOWLEDGMENTS

I would like to thank my graduate advisor, Dr. Jensen, for all his support and patience through the past three years. I would like to thank the other members of my graduate committee, Dr. Red, Dr. Todd, Dr. Sorensen, and Dr. Cox, for their excellent support and encouragement. I would also like to thank the members of the Direct Machining and Control research group and the ParaCAD research group at Brigham Young University.

I owe a special debt of gratitude to my parents and family. They have always been there to offer their support and encouragement. I would like to express appreciation to all of my friends who have given help and support during this endeavor.

Finally, I would like to thank Brigham Young University, especially the department of Mechanical Engineering. I have learned greatly in the past three years to be a better person. With the education I received from here, I have laid a good foundation for my future professional career.

TABLE OF CONTENTS

1	INTRODUCTION	1
1.1	Statement of problems	4
1.1.1	TMM.....	4
1.1.2	Geometry feature calculation.....	6
1.1.3	Gouging elimination	6
1.1.4	Five-axis finish machining.....	8
1.1.5	Clean-up machining.....	8
1.1.6	Accuracy and performance	9
1.2	Limitations	10
1.3	Research objectives.....	11
2	LITERATURE REVIEW	13
2.1	Overview.....	13
2.2	The data structure of the STL data.....	14
2.2.1	STL data.....	14
2.2.2	Calculation of local geometry information.....	15
2.3	Rough machining	16
2.4	Finish machining.....	17
2.4.1	Three-axis machining.....	17
2.4.2	Five-axis machining.....	18

2.5	Clean-up machining	22
3	METHODOLOGY	25
3.1	Data structure of the STL data	25
3.2	Calculation of the geometry feature.....	28
3.3	Frame transformation.....	30
3.4	Rough machining	32
3.4.1	Machining method determination and cutter selection	33
3.4.2	Cutting-depth determination	34
3.4.3	Offset surface determination.....	34
3.4.4	Justification of the end of a tool path.....	35
3.4.5	Tool path generation	36
3.5	Finish machining.....	36
3.5.1	Tool path topology selection.....	37
3.5.2	CC points generation.....	39
3.5.3	Gouging recognition	57
3.5.4	CL points generation of three-axis machining.....	60
3.5.5	Five-axis machining CL point generation.....	66
3.6	Clean-up machining	81
3.6.1	Clean-up strip recognition.....	83
3.6.2	CL points generation.....	84
4	IMPLEMENTATION.....	99
4.1	Architecture of the TMM system.....	99
4.1.1	Data flow diagram.....	99

4.1.2	User-case view	100
4.1.3	Sequence diagram	101
4.1.4	Class diagram	102
4.2	Hardware	103
5	RESULTS	105
5.1	Overview	105
5.2	Benchmarks	105
5.3	Rough machining	108
5.4	Three-axis finish machining	110
5.5	Five-axis finish machining	111
5.6	Clean-up machining	116
6	CONCLUSIONS	119
6.1	Overview	119
6.2	Summary	119
6.3	Contributions	121
6.4	Future research	122
	REFERENCES	125
	Appendix A	131
	Appendix B	141

LIST OF FIGURES

Figure 1-1 The process of SSM	1
Figure 1-2 Different gouging cases.....	7
Figure 2-1 C-space method.....	18
Figure 2-2 Gouging cases of the flat end cutter.....	18
Figure 2-3 A safe C-space.....	20
Figure 3-1 Map between real triangles and ordered triangle array.....	26
Figure 3-2 Calculation of common vertex normal.....	29
Figure 3-3 Frame Transformation.....	31
Figure 3-4 2.5-axis rough machining.....	32
Figure 3-5 Remaining material height in the ball end cutter machining process	33
Figure 3-6 Offset of tessellation data.....	35
Figure 3-7 Generation of CC points.....	37
Figure 3-8 Difference of the safe machining surface and the given tessellated surface...	40
Figure 3-9 Generation of CC points.....	42
Figure 3-10 A spindle surface.....	42
Figure 3-11 Cross section of the design surface at a CC point.....	45
Figure 3-12 Step-over distance determination.....	46
Figure 3-13 Bottom of the flat end cutter.	48
Figure 3-14 The generalized cutter ([54]).....	51

Figure 3-15 Tip of a generalized cutter.....	53
Figure 3-16 Expanded section of the generalized cutter.....	54
Figure 3-17 Remaining material height calculation.....	55
Figure 3-18 Different gouging cases.....	58
Figure 3-19 The yaw angle is needed in this situation.....	58
Figure 3-20 Different gouging cases.....	59
Figure 3-21 Gouging possible facets.	60
Figure 3-22 Gouging problems between a generalized cutter and a triangle.	61
Figure 3-23 Different sections of a generalized cutter.....	63
Figure 3-24 C-space construction.	66
Figure 3-25 Sample C-space for a facet.....	68
Figure 3-26 C-space construction processes for CC points	69
Figure 3-27 Gouging elimination methods.	71
Figure 3-28 The trajectory of the cutter holder.....	79
Figure 3-29 The cutter posture determined by the ideal C-space set.....	80
Figure 3-30 Clean-up strip.	82
Figure 3-31 Enlarged image of cross section of the clean-up strip.....	83
Figure 3-32 Position the cutter to CC points.	84
Figure 3-33 Lift the cutter to eliminate gouging.....	85
Figure 3-34 Edge contacts the taper of the cutter.	88
Figure 4-1 TMM software architecture.....	100
Figure 4-2 User-case view of TMM.	101
Figure 4-3 Operation sequence diagram of finishing.	102

Figure 4-4 Class diagram of TMM	103
Figure 4-5 Five-axis full-size Tarus mill	104
Figure 5-1 Benchmarks.....	107
Figure 5-2 Rough machining results.....	109
Figure 5-3 3-axis finish machining results.....	110
Figure 5-4 A sample 3D C-Space.....	111
Figure 5-5 Step-over distance of convex surface.....	112
Figure 5-6 5-axis finish machining results.....	115
Figure 5-7 Clean-up machining results.....	117

LIST OF TABLES

Table 3-1 Comparison between the point-based method and the curve-based method....	37
Table 5-1 Rough machining parameters.....	108
Table 5-2 3-axis finish machining parameters.....	110
Table 5-3 5-axis finish machining parameters.....	112
Table 5-4 5-axis finish machining parameters.....	117

1 INTRODUCTION

In the decade after the Second World War, the increased need for precision-machined aircraft parts led to the development of NC milling machines. As an ever-increasing variety of products are now being designed with sculptured surfaces, machining these surfaces efficiently has become an important role in the process of bringing new products, such as automobiles, ships, TV sets, etc., to the marketplace [11]. Sculptured Surface Machining (SSM) is a method that includes “information processing technology concerned with the efficient machining of sculptured surfaces by using NC machines.”

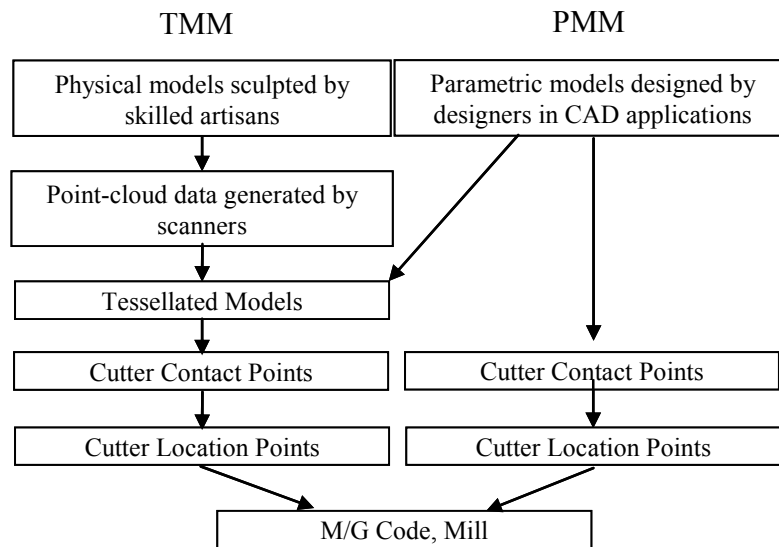


Figure 1-1 The process of SSM

The SSM process typically consists of tessellated model machining (TMM) and parametric model machining (PMM) as shown in Figure 1-1. The product design is embodied in either a physical model that can be sculpted from clay by a skilled artisan [35] or a parametric model created in a CAD software application. For clay models, white light scanning can be used to generate point cloud data, which can then be reconstructed into tessellated or parametric models (such as Bezier, B-spline, or NURBS). Tool path planning, that is vital to the whole SSM process, is based on either physical models that have been scanned and tessellated or parametric models. These two types of models lead to very different machining strategies, e.g. tool paths developed on tessellated models and tool paths associated with parametric models.

Since the introduction of CAD/CAM systems into engineering and manufacturing companies in the 60's and 70's, the vendors of these systems have been pushed to provide more complete tool path planning strategies for PMM. Little attention has been given to TMM by these vendors other than to plan a simple ball end mill path across the tessellated model. More recently, researchers have begun to investigate machining strategies for TMM. However they have been limited and have not addressed all the stages of machining.

Operations of both PMM and TMM can be grouped into four stages according to machining sequences: the rough machining stage, the semi-finish machining, the finish machining stage, and the clean-up machining stage.

The purpose of rough machining is to quickly remove material while leaving a small amount of uncut allowance for semi-finish machining and/or finish machining. The 2.5-axis machining method is widely used in the rough machining stage.

The semi-finish machining stage is an intermediate stage between the rough machining stage and the finish machining stage. Its purpose is to generate a physical surface that has a uniform stock allowance for the finish machining stage so that during the finish pass the cutter will have a more constant chip load.

The most time-consuming parts of developing a complete (rough through finished) tool path plan are the calculations and subsequent graphical representations of the final surface milling pass across a parametric or tessellated surface. In addition, the actual finish machining plan directly influences the quality and accuracy of the resulting machined surface. Consequently, most current SSM research is focused on the finish machining stage.

In addition to the rough, semi-finish, and finish stages, a clean-up machining stage is necessary if there are small regions of the surface where the finish cutter could not machine leaving uncut material behind. The most challenging parts of this stage includes the clean-up strip (region) recognition and the cutter location point determination.

A cutter-contact (CC) path is a series of CC points where the cutter is tangent to the surface being machined, and a cutter location (CL) path is defined as the locus of the CL points, typically at the center tip of the tool. Brief descriptions of the operations used to determine the CC and CL paths are as follows:

1. Tool path planning—the CC paths are obtained from the design surface.
2. CL data computation—the CC paths are converted to CL paths.
3. Cutting simulation—the work piece is “virtually” machined (many packages are capable of doing virtual machining).

4. Gouge detection—the simulated machined surface is compared against the design surface.

This dissertation focuses on the investigation of TMM operations 1 and 2. Operation 3 is out of the scope of this dissertation because it is commonly understood, while operation 4 is considered as a part of operation 2.

The objective of this dissertation is to suggest ways in which to improve the machining accuracy and efficiency in each operation. The following sections of this chapter will analyze current challenging problems in TMM and this dissertation's research objectives.

1.1 Statement of problems

This dissertation presents solutions to several machining problems. This section will introduce and discuss the main problems. First, the reader will be given a clear understanding of the advantages that TMM has over PMM and why it is necessary to improve TMM. Next, we will look at the TMM data source and discuss TMM data preparation. Finally, we will separately discuss each of three key problems in TMM: gouging elimination, 5-axis finish machining, and clean-up machining.

1.1.1 TMM

Currently, most CAD systems use parametric surfaces (or solids) to represent the geometry of a CAD model. To transfer models between different CAD/CAM systems for various designs or manufacturing processes, neutral data files such as Initial Graphics Exchange Specification (IGES) are used extensively. However, because most CAD systems use different internal representations, the translation of CAD models using IGES

is not always straightforward and error free. In contrast to IGES, the stereo lithography (STL) format is simple and easy to implement. Because of its simplicity and established use in various engineering fields, STL translation today is supported by most CAD/CAM systems.

Meanwhile, most CAM systems using PMM still need to locally tessellate the surface under the tool bottom or even internally triangulate the model in order to calculate a collision- and gouge-free tool position (Lauwers et al. 2003).

Today, with the advent of more powerful computers, the millions of calculations required to plan a flat end-mill tool path over TMM surfaces is not only feasible but can be performed in a reasonable amount of time.

Furthermore, the latest three-dimensional scanning technology also helps the rapid growth of reverse engineering applications in which very large and complex models are created and stored in STL files. The general consensus is that the use of triangulated surfaces and STL files for design and manufacturing applications will become increasingly popular.

According to the above analysis, both PMM and TMM have advantages and drawbacks. However, general commercial CAM systems do not have a full complement of tool path planning techniques and methods for rough, semi-finish, finish, or pencil tracing (clean-up) of TMM surfaces, even though TMM techniques in CAD/CAM areas will likely be applied more widely in the future. This dissertation will focus on the development of a complete TMM system. General computing algorithms for various machining stages will be presented in detail.

1.1.2 Geometry feature calculation

The triangle mesh format is a common representation for tessellated models. STL files can be used to record high-quality triangle mesh data for tessellated models, not only because most current CAD/CAM systems support the STL format, but also because the STL file is standardized and easy to read and interpret. Consequently, many researchers are considering the use of STL files as the topology skeleton on which to plan all tool paths from rough milling to clean-up. This dissertation basis all of its tool path planning algorithm development on STL files.

The disadvantage of STL files lies in the absence of topological information and local geometry information of the tessellated surface (such as the normal of the vertex, the curvature at a random point, etc.) from the file format. Only the normal information of every facet and the position information of every vertex are recorded in STL files. While the position information of every vertex is assured to be on the surface and the normals of the tessellation give only an approximation of the true surface normals.

Prior research that has been conducted in related areas (e.g. solid modeling, rendering techniques) will be applied to the Geometry Feature Calculation in this dissertation.

1.1.3 Gouging elimination

In the machining process, parts of the cutter (including the cutter head) may interfere with the design surface. If this occurs anywhere along the tool path, a gouging problem occurs. Gouging problems can be categorized into three cases: local gouging, global gouging, and cutter head gouging (Figure 1-2). Local gouging occurs when the cutter bottom interferes with the tessellated surface by more than the allowable tolerance. Global gouging occurs when the cutter shaft (or side) interferes with the tessellated

surface by more than the allowable tolerance. When cutter head gouging occurs, the collate, tool holder, or spindle interferes with the tessellated surface by more than an allowable tolerance. The method for recognizing cutter head gouging is the same as that of global gouging because the cutter head can be treated as a cylinder just like the cutter shaft.

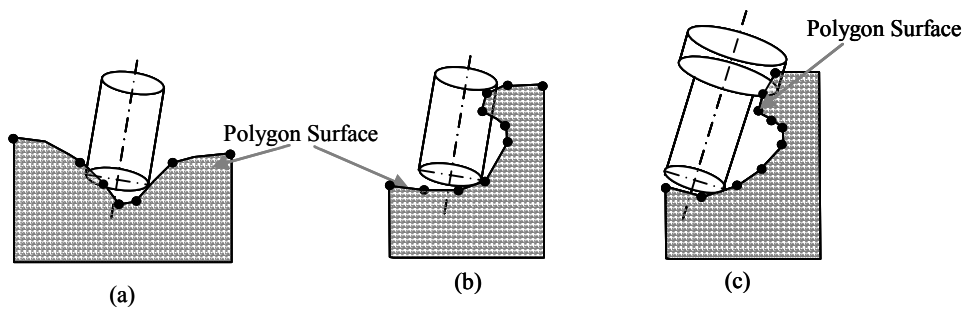


Figure 1-2 Different gouging cases.

The method for solving gouging problems can be divided into two steps: (1) Gouging detection, and (2) Gouging elimination. To detect gouging problems, all triangles in the triangle mesh are transformed into a tool coordinate system. This system is used to determine whether there is any cutter gouging. To eliminate gouging problems, detailed analyses for various machining stages are presented later according to various gouging elimination methods. For example, in 3-axis machining, the cutter is lifted along Z direction to eliminate the gouging. In 5-axis machining, the cutter is both rotated and lifted to eliminate the gouging. In clean-up machining, the cutter is translated along a vector to eliminate the gouging. Even if different cutters are used, the gouging elimination equations are different.

1.1.4 Five-axis finish machining

The tool path plan of 5-axis finish machining can be divided into roughly three steps: (1) Selection of the tool path topology; (2) Generation of CC (cutter contact) points including the determination of the step-over distance (i.e. the generation of the CC tool path); and (3) Generation of CL (cutter center or cutter location) points, (i.e. generation of the CL tool path). In this dissertation, we will focus our efforts on step 2 and step 3 because the techniques related to these two steps are critical to machining speed and accuracy.

In step 2, if the step-over distance is set to the largest possible value (with the precondition that the cup height is less than the machining tolerance), the number of tool paths is minimized. As a result, the machining time is decreased. In step 3, the cutter posture and position are described with four parameters: the position of the CC point, the inclination angle λ by which the cutter is rotated about ${}^L Y$, the yaw angle ω by which the cutter is rotated about ${}^L Z$, and the height δ by which the cutter is lifted along the normal of the surface. At a fixed CC point, the optimal set $(\delta, \lambda, \omega)$ describes a gouge-free cutter posture and has a cusp height less than the machining tolerance. Current research techniques and the machining algorithms found in popular CAM packages that are used to deal with either of these two steps still require a great deal of effort.

1.1.5 Clean-up machining

In some of the latest commercial CAM software, the pencil-cut function is provided for machining parametric surfaces and a few of these systems claim to do clean-up of triangulated surfaces. However, to the best of the author's knowledge, most of the commercial CAM systems suffer from one or more of the following problems: (1)

Failure to correctly identify the pencil-cut curves that are obvious to the human user; (2) Failure to generate pencil-cut tool paths correctly; or even worse, (3) Sometimes the software crashes during the process of generating the pencil-cut tool path of complex surfaces. Hence it is imperative to develop a more robust and accurate method for pencil-cut machining.

1.1.6 Accuracy and performance

Accuracy, efficiency and robustness characterize the ideal tool path. Research efforts on SSM are driven by the goal of higher accuracy and efficiency. Accuracy is guaranteed by two factors: (1) The tool path is gouge-free (there are no over-cut situations); and (2) The cusp height is less than the machining tolerance. A gouge-free tool path refers to a tool path that does not interfere with the design surface. The cusp height describes the distance between the post-form surface and the design surface. In this dissertation, accuracy is treated as a prerequisite condition because it is a critical requirement in SSM. All methods presented in this dissertation plan a gouge-free tool path and have a cusp height less than the machining tolerance.

Efficiency of a tool path is discussed in terms of the machining time consumed. It is determined by three factors: (1) The machining time along a single tool path; (2) The traversal time between adjacent tool paths; and (3) The number of tool paths. Factor 2 is not discussed in this dissertation. Efficiency is the objective in all methods of this research.

Theoretically, there is an *ideal* optimal tool path which consumes the least machining time with satisfactory machining accuracy. However, there are no reasonable methods to

find this *ideal* optimal tool path. The objective of current research is to find a tool path that is as close to the *ideal* as possible within a reasonable set of search criteria.

1.2 Limitations

Research on SSM has been conducted for more than 30 years. However, we cannot discuss all areas of SSM in one dissertation. Rather, this dissertation will focus on a few areas of SSM that still need work. The following issues will *not* be discussed:

1. PMM (The parametric model machining method).
2. The pocket rough machining method in the rough machining stage.
3. Semi-finish machining.
4. Automatic cutter selection.

PMM and TMM are two main machining methods in SSM. However, as mentioned in the introduction, the focus of our research is on TMM. To learn more about techniques related to PMM, see Choi's research on the subject [11].

The techniques related to pocket machining are relatively mature. In addition, pocket machining is not widely adopted in car body clay-machining which is the main application area of this research.

Semi-finishing techniques are not discussed because these techniques are similar to those in 3-axis finish machining. The only difference is the machining tolerance.

Cutter selection is a challenging problem that uses pattern recognition. Because the research presented here focuses on SSM problems and not pattern recognition problems, a discussion of cutter selection is outside the scope of this dissertation.

1.3 Research objectives

Improvements in TMM machining accuracy and efficiency can be made in the following four areas:

1. The step-over distance can be maximized based on local geometry features.
2. Machining speed can be increased and the machining tolerance can be guaranteed in the 5-axis machining process by using the 3-dimension configuration space method. In this method, a 3D C-space is built for each CC point. Every configuration set in this 3D C-space guarantees that the cutter is gouge-free and that the cusp height is less than the machining tolerance. The optimal cutter orientation is then obtained by determining the optimal configuration set in the corresponding 3D C-space via a special optimization process.
3. Machining speed and accuracy can also be increased using a new method, presented in this dissertation, for the clean-up machining process. In this method, the generalized cutter is moved along random vectors.
4. To unify the methods used to increase machining speed and accuracy, this dissertation constructs a complete TMM system. In this system, rough-machining methods, finish machining methods, and clean-up machining methods are developed with C++.

2 LITERATURE REVIEW

This chapter reviews research related to sculptured surface machining (SSM), including research in various fields such as geometry information calculations of tessellated surfaces, rough machining methods, finish machining methods and clean-up machining methods.

2.1 Overview

SSM techniques emerged in 1950. Duncan's work was the basis of most existing researchers' work [15]. Held [19] summarized the rough machining methods before 1991, in detail, in his publication. Choi and Jerald reviewed related SSM techniques prior to 1997 in their book [15]. Considerable research has been conducted on 5-axis machining and the clean-up machining stage (e.g., Choi, B.K., Jerald, R., Jensen, G., Lee, Y.S., Waterloo Univ., Jun, C.S, etc.). Other SSM-related techniques—such as tessellated model reconstruction algorithms, geometry feature calculations, and tessellation model data structures—have been developed in other areas (e.g., rapid prototyping, computer graphics, etc.).

Detailed information will be presented in this section as follows:

1. The data structure of the STL data.
2. Rough machining.

3. Finish machining.
4. Clean-up machining.

2.2 The data structure of the STL data

2.2.1 STL data

STL files were widely used as the data source from which to plan a tool path [1][35][50]. However, an STL file is like a bucket of random facets—the facet data in the STL file is unordered and no topological information is provided. First, an efficient data structure must be employed to organize the STL data because subsequent operations (such as data query and geometry features calculation, the tool path generation, etc.) query facet geometry information frequently. Rock proposed applying the balanced binary tree (ASL tree) to organize STL data [45]. However, there are difficulties in planning the tool path in multi directions if ASL is applied. In other words, querying an entity along different directions, such as the X direction or Y direction, which is frequently done in this dissertation, is a time-consuming process in an ASL tree. A more reasonable method comes from the multi-directional binary search tree (k-d tree) in Bentley's work [6]. This method has been further applied to TMM by Yau [54]. Rosenberg has compared the current popular data structures used in region operation areas [44]. However, to the best of my knowledge, there is no detailed performance analysis of different data structures being applied to TMM.

To represent the topology information of STL files, we will use current research in solid modeling. The winged edge structure is the most popular structure for a manifold polyhedron representation [7]. The radial edge structure is the most popular structure for

non-manifold objects [53]. Both of them can be applied to STL models in SSM. However, neither of them is specifically developed presented for STL files. As a result, there is a lot of redundant memory allocation in these data structures. To improve the performance of the data structure, a simplified version is needed.

Linear lists, although a poor structure for sorting and searching, is still a good choice in which to store the triangle entities because these triangles are not frequently searched. Actually, triangles are ordered only once for subsequent operations in SSM. There are no operations like insert, delete, etc. in SSM operations. Even the query operation is executed from one CC point to next one. Consequently, any data structures discussed above do not have an obvious advantage over link lists in SSM. In addition, the data structures above are complicated to implement. For the above reasons, linear lists are applied in this dissertation. A simplified version of the radial edge structure is used to represent the topology information.

2.2.2 Calculation of local geometry information

Only facet normals and vertex positions are recorded in STL files. Some research has been conducted to calculate other local geometry features for subsequent processes [22][41][42]. The common method of calculating the normal of a common vertex involves adding up the normal of the facets incident to this common vertex. Meyer has presented another method to calculate the curvature at a point and the line of curvature passing through this point [41]. The differences in these algorithms lie in the calculation of the weight of the incident facet normal. We will choose Meyer's method to calculate the local geometry feature because the theoretical model of his method is more reasonable than the common industry method.

2.3 Rough machining

Since the goal of the rough machining is to remove the material as fast as possible, most existing rough machining research is proposed to improve efficiency. To maintain efficiency, the cutter orientation should not change frequently in rough machining. As a result, most researchers apply 2.5-axis or 3-axis machining methods. Held presented a general approach for pocket machining in his book [19]. Both the contour-parallel machining pattern (mostly used for pocket machining) and the direction-parallel machining pattern (mostly used for plane-stepping machining) were surveyed and analyzed. Choi surveyed some popular approaches on rough machining in his book [11] in which rough machining methods for TMM were not introduced. For pocket rough machining, the main concern is the automatic recognition of the machining pattern, especially for complex surfaces. Arkin and Held treated the zigzag problem of the machining pattern as an NP-hard problem and managed to decrease the retraction time based on graph-theoretical equations. This method is useful for complex surfaces. Tang [49] and Tao [50] presented work to reduce the retraction time of pocket machining. These methods can be applied to complicated surfaces in plane-stepping machining.

Lee applied the 5-axis technique to a rough machining of a ruled surface [35]. From a practical point of view, Lee's work is not quite practical because ruled surfaces are not widely applied in the industry.

The cutting force is a function of various independent parameters. Current cutting force models include volumetric models and mechanistic models [11]. A simple volumetric model can be used to estimate spindle power requirements and average cutting forces. However, a more complete mechanistic model is needed if peak force

magnitudes and force directions are required. A good review of information on mechanistic models can be found in Smith and Tlusty's work [48].

There isn't much research on plane-stepping tessellated model machining even though a great deal of rough machining research has been conducted. My dissertation focuses on how to build an efficient, easy to implement, rough machining program for TMM.

2.4 Finish machining

2.4.1 Three-axis machining

Three-axis machining and 5-axis machining are two machining methods used during the finish machining stage. Choi has presented some detailed analyses for three different cutters on convex and concave regions in the 3-axis machining stage [11]. These analyses were the basis of most current 3-axis machining methods. Li and Jerald have presented a gouging justification method in their 5-axis machining work [35]. However, they did not apply this method to their earlier 3-axis machining research. Lo presented a two-stage method for 3-axis machining [38]. During the first stage, the larger cutter is utilized for efficient surface machining, but the cutter-interference region was skipped. At the second stage, the smaller cutter was utilized for the skipped residual region. The sizes of these two cutters were chosen so as to generate an interference-free path intersection and the total lengths of the machining paths were minimized. However, no detailed performance comparisons between this method and the conventional 3-axis machining methods were proposed in his research. This method can also be applied to the flat or the fillet cutter even though Lo's work is based on the ball end mill.

Yau has presented an algorithm to plan the tool path and eliminate the gouging with a generalized cutter [54]. The gouging elimination methods in Yau's work were based on the methods of Li and Jerald. This dissertation combines Yau's work and a new step-over distance determination method in 3-axis machining.

2.4.2 Five-axis machining

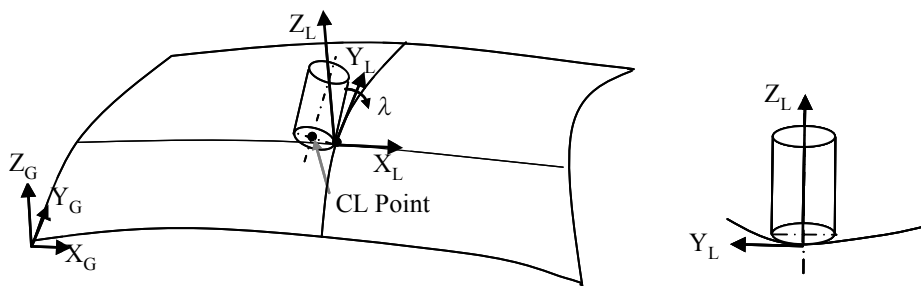


Figure 2-1 C-space method.

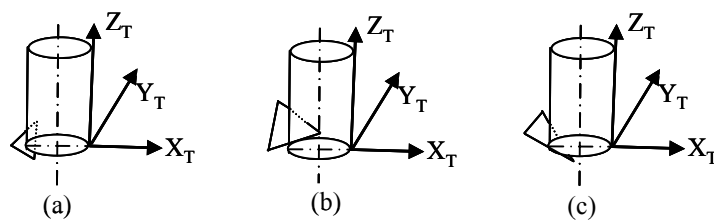


Figure 2-2 Gouging cases of the flat end cutter.

Some popular 5-axis machining methods are discussed in this section. Jensen [22] has presented the curvature matched machining (CM^2) method. By matching the curvatures of the silhouette of the cutter to the curvatures of the surface at a CC point, the

material can be cut locally at a high removal rate (Figure 2-1). Because the gouging problems are not eliminated, this method can be used to set the initial cutter posture and position for this dissertation. Li and Jerald have presented a robust algorithm to detect and eliminate all types of gouging in TMM [35]. They treated vertex gouging, edge gouging, and triangle gouging separately (Figure 2-2). Rotating the cutter with the inclination angle and lifting the cutter along the normal of the surface were techniques used to eliminate gouging. Lauwers applied CM^2 and the gouging elimination method presented by Li to tessellated model machining method [27][28]. CM^2 was implemented to determine the initial inclination angle and Li's work was used to eliminate the gouging. However, despite achieving a gouge-free tool path, none of the above research considered the yaw angle. It is a complex problem to find the optimal tool path with the known initial cutter posture and the current gouging elimination method, and there is no reasonable analysis in his research to explain whether the resulting gouge-free tool path is optimal.

The principle axis method (PAM) [43] method can be viewed as a type of CM^2 . The two principle curvatures of the cutter surface, which match the two principle curvatures of the model surface respectively, are simulated and analyzed. The difference between PAM and classical CM^2 is that PAM matches the minimum principle curvature of the cutter surface for the maximum principle curvature of the model surface, whereas CM^2 matches the cutter contour curvature to the curvature of the cross section curve on the surface. The direction of the minimum principle curvature differs a great deal along a single tool path. As a result, the cutter posture in PAM changes too much over a single

tool path. The cusp height generated by this tool path is large and the machining speed is low. Because of its inefficiency, the planned tool path is not an optimal one.

Warkentin presented a geometry feature matching method named the multi-point matching algorithm (MPM) [51][52]. MPM matches the cutter contour with the geometry of the local part surface by placing the cutter with an orientation that maximizes the number of contact points between the cutter and the model surface. This work has not yet been applied to tessellated model surfaces and its implementation presents a significant challenge. Additionally, its basic idea to locate and position the cutter as near as possible to the part surface is similar to CM^2 .

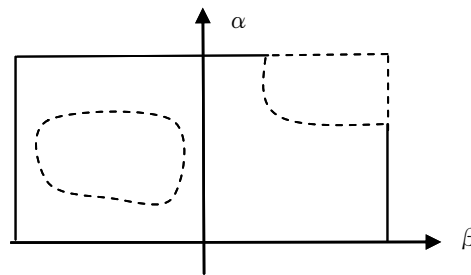


Figure 2-3 A safe C-space

Methods based on the configuration space (C-space) approach are relatively mature and robust. The basic steps of the C-space approach are: (1) Finding a safe C-space on a moving object without interfering with any obstacles; and (2) Searching for the best configuration in the C-space according to the objective functions of the problems. The C-space method is an optimization algorithm that is applied to engineering problems. Choi has applied the C-space approach to SSM [12]. Lee presented a detailed mathematic model for the C-space approach based on a 5-axis machining method with a fillet cutter

[31]. Jun described the C-space approach optimization principles based on the tessellated model [23]. A *safe* C-space was generated with an adaptive boundary search method (the region bounded by the dashed line in Figure 2-3 is the *safe* C-space). In this *safe* C-space, each configuration set describes a gouge-free cutter posture. The optimal tool path is then planned according to a simplified objective function that is not based on the kinematics analysis. The cutter lifting height is not considered in most of current C-space work. Consequently, these C-space approaches are all based on a 2-dimension C-space.

The haptic interface method is another method to plan the tool path [5][55]. However, this method is not discussed in detail in this research because it is quite different from the classic tool path plan methods. Furthermore, many of the current applications of the haptic interface method to machining are theoretical.

In the industry, the optimal CL tool path is planned based on the design surface topology, the shape of the cutter, and the kinematics of the machine tool. There are three factors (adjusting the cutter along the surface normal by δ , adjusting the cutter lift height, and rotating the cutter by the inclination angle λ and the yaw angle ω) that precisely determine the cutter posture and position at a CC point. Past researchers have focused their efforts on one or two of these factors but not all three. In addition, the calculation of the step-over distance in previous research was based on a simplified surface topology at a CC point.

All the methods above optimize the cutter posture for a CC point separately. For example, CM^2 and PAM match the curvature of the cutter and the curvature of the surface, MPM positions the cutter to be as close to the surface as possible, the C-space method searches for the optimal location for a CC point within the C-space set along the

boundary of the *safe* region, and the machining speed is based on the kinematics plan of the entire tool path. Consequently, the kinematics of the machine tool needs to be involved in planning the optimal tool path. The consideration of step-over distance, machine tool kinematics, and the three adjustment factors $(\lambda, \omega, \delta)$ will improve the machining performance, especially for complex models. The method presented in this dissertation considers all of these inputs when planning an optimal tool path.

In this dissertation, a 3D C-space is built for each CC point with three factors: $(\delta, \lambda, \omega)$. Every C-space set $(\delta, \lambda, \omega)$ inside this 3D C-space describes a gouge-free cutter posture. This research will integrate the C-space method, CM^2 , and machine tool kinematics analysis for the planning of optimal cutter postures and positions within the constructed 3D C-space.

2.5 Clean-up machining

Currently, most of the research on clean-up machining is based on 3-axis machining methods with the ball end cutter. Kim [25] has extended the guide plane to be the guide surface that generates a clean-up tool path parallel to the clean-up strip. This method is more efficient than classic methods. However, it cannot be applied to tessellated models because tessellated models lack the parametric geometry feature. Ren [46] treated the clean-up region as a V shape slot and applied multiple intermediate cutters to clean-up machining. A searching method was used to recognize the pencil curve. He further presented a method to refine the pencil-cut curve and smooth the pencil-cut tool path in his later work [47]. This method is reasonable if the shape of the cross section along the clean-up strip is simple. Kim and Jun applied their curve-based approach to the

generation of clean-up CL points [26] in the finish machining stage [44]. Zhu [55] applied the 5 DOF (degree of freedom) haptic interface to the 5-axis clean-up machining method.

To cut the clean-up strip, lifting the cutter along a vector determined by the shape of the strip is normally more efficient than lifting the cutter along the Z direction. However, none of the above methods consider this. In addition, this dissertation will derive equations for clean-up machining with a generalized cutter.

3 METHODOLOGY

This chapter is devoted to machining methods implemented in the TMM system proposed in this dissertation. A simple and efficient data structure for the STL data is presented first. The local geometry information of the design surface is then calculated. Next, rough machining methods and 3-axis machining methods are implemented based largely on existing methods. A new unique 3D C-space 5-axis machining method is presented. The 3D C-space is my most significant contribution and is demonstrated through searching the optimal tool path within the configuration space at each cutter contact point and considering the shape of the cutter, the local geometry feature of the TMM surface, and the machine kinematics. Finally, a creative clean-up machining method is proposed. In this method, a more efficient gouging elimination method than existing ones is developed. The 3D C-space method and the subsequent clean-up machining method are keys in this dissertation.

3.1 Data structure of the STL data

Facets in the STL file are not ordered, which means that a facet must be efficiently extracted from the STL text file when it is needed for path planning (see triangle 11 at the left-bottom corner of Figure 3-1(a)). When a tool path is planned, triangles are queried frequently. To reduce each triangle query time when dealing with a large STL

file, an efficient querying method is presented. This dissertation proposes an algorithm wherein three vector objects (a template class in the Standard Template Library of C++) are used to save all triangle objects. One vector saves the triangle entities read from STL files (see *arrTriangle* in Figure 3-1(b)). Each node of this vector is a triangle entity that is represented by the following C++ code segment:

Algorithm 1: Create ordered triangle arrays

```

class CTriangle
{
    //.....
    int      m_nId;
    double   dXMax, dXMin, dYMax, dYMin, dZMax, dZMin;
    CVector  m_vecNormal;
    CPoint   m_ptVex[VEXCOUNT];
    int      m_arrAdjPtIndex[VEXCOUNT][ADJCOUNT], m_arrAdjTriID[VEXCOUNT][ADJCOUNT];
}

class CPoint
{
    //.....
    double   m_dX, m_dY, m_dZ;
    CFrame   m_frameTransform;
    double   m_dInclinationAngle;
}

```

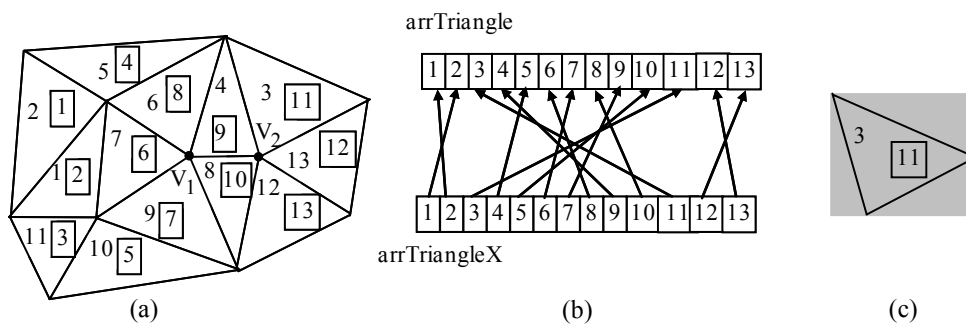


Figure 3-1 Map between real triangles and ordered triangle array.

Six variables (*dXMax*, *dXMin* ...) are used to record the bounding box of each triangle (see Figure 3-1(c)), and *IDs* for all adjacent points and triangles are recorded as well. The data structure of the triangle is a simplified version of the Radial Edge

Structure [53]. Both topology information and geometric information are recorded within the triangle object. Two vector objects save ordered pointers to triangles along the X and Y directions separately, every node uses m_nId to map the pointer to its original triangle in the STL file. For example, in Figure 3-1(b), node 3 (3 refers to its index in the array, not its ID) in *arrTriangleX* means its minimum X coordinate is less than that of triangle 4 but greater than triangle 2. Meanwhile, this node maps to triangle 11 (11 is its ID) in the *arrTriangle* array (Figure 3-1(a)). The point class *CPoint* records the point position, the point normal, the transformation information, etc. The following algorithm shows how to save and construct vector objects in STL files:

Algorithm 2: Create ordered triangle arrays

```
while (!END_OF_FILE) // until reach end of file
{
    ReadLines_BuildTriangle(); //Read lines for a triangle and build a triangle object
    AddTriangle(); //Calculate triangle's boundary box, and add this triangle to the end of arrTriangle
    OrderArrays(); // Insert this triangle's pointer into arrTriangleX and arrTriangleY according to its minimum
    // boundary values along X and Y direction separately
}
```

In Figure 3-1(a), the number in every triangle is the original position of the triangle recorded in STL files, while the number in the cell grid is the position of the corresponding triangle recorded in *arrTriangleX*. Figure 3-1(b) shows the map between two types of numbers. The bounding box for every triangle and the ordered vector object is the key to the entire algorithm's efficiency.

To improve query performance, great care must be taken when triangles are queried. The algorithm below shows how to calculate the normal of triangle vertices.

Algorithm 3: Find all adjacent triangles and vertices for vertices

```
//example where machine direction is +X direction
indexPrev =1;

for (i=0; i<arrTriangleX.size; i++) //all triangles in arrTriangleX
{
    for (j=0; j<3; j++) //every vertex of one triangle
    {
        bFirstTri = FALSE;
```

```

for (k=indexPrev; k< arrTriangleX .size; k++) //from the previous index to check whether the triangle' boundary box covers
//current CC point, imagine V1 is the current vertex (Fig. 1 (a))
{
  if (SameVertex(k)) //check whether this triangle has at least one vertex same with current vertex. for example, if k=1, then,
//SameVertex(k) = FALSE, and go to next loop, k++; When k=6, it will find that Triangle 6 in arrTriangleX
//has the same vertex with V1 , then, SameVertex(k) = TRUE
  {
    AddTriToAdjTriArr(k); //add triangle (for example, Triangle 6) into m_arrAdjTriIndex of V1
    AddPtToAdjPtArr(k); //add those common vertices (except V1, two other vertices of triangle 6) into m_arrAdjPtIndex of V1
    if (!bFirstTri)
    {
      indexPrev = k; //if this is the first triangle that has the same vertex with CC points, record this triangle's position as the
//index,
//this index could be used as next loop's index, k=6
      bFirstTri = TRUE;
    }
    if (LowerBoundary(k)) //if all X coordinate of triangle k is greater than CC points, other triangles are not necessary to justify
//any more, (k=11 for this situation in Fig. 1) in another word, //keep looping until k=11, then,
//current triangle's boundary box is greater than //V1.m_dX break;
  }
} //in the loop for V1, triangle 6, 7, 8, 9, 10 will be found to be adjacent triangles of this V1; in next loop for
//vertex V2, initial value of indexPrev will be 6. 9, 10, 11, 12, 13 will be adjacent triangles of V2
}
}
}
GetCCPointNormal(); //Use these adjacent points to calculate the normal

```

3.2 Calculation of the geometry feature

Before the tool path can be planned, the normal of each triangle vertex must be calculated. When compared to various algorithms, the “Mean Weighted by Angle” (MWA) algorithm and the “Mean Weighted by Areas of Adjacent Triangles” (MWAAT) algorithm are two of the best algorithms for sculptured surfaces [22]. MWA adds up the normal of all facets incident to the common vertex in question.

The weight of every facet normal is the angle under which a facet is incident to the vertex (Figure 3-2(a)).

$$N_v = \sum_{i=1}^n \alpha_i N_{fi} \quad (3.1)$$

MWAAT treats the area of each incident facet as the weight of the facet normal.

$$N_v = \sum_{i=1}^n A_{fi} N_{fi} \quad (3.2)$$

Another more reasonable algorithm is from Meyer's work [41]:

$$N_v = \frac{1}{2 \cdot A_{mixed}} \sum_{i=1}^n (\cot \beta_{ij} + \cot \gamma_{ij}) \cdot (V_i - V_j) \quad (3.3)$$

$$A_{mixed} = \sum_{i=1}^n A_{fi} \quad (3.4)$$

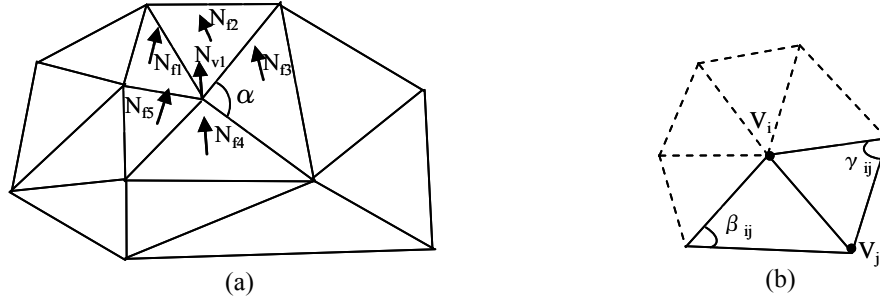


Figure 3-2 Calculation of common vertex normal

If the facet is a non-obtuse angle, A_{fi} is the incident facet's area. Otherwise, A_{fi} is half of the incident facet's area if the angle α_i at the common vertex is obtuse and one quarter of the facet's area if the angle α_i at the common vertex is not obtuse (Figure 3-2(b)). We recommend adopting Meyer's method if the tessellated surface is complex, such as those found from scanning a physical turbine blade that is being reversed engineered. Both MWA and MWAAT can be applied if the model tessellation is simple.

The normal of a point P on the facet edge V_1V_2 is calculated with the affine algorithm (Equation (3.5))

$$N_t = N_{v1} + (N_{v2} - N_{v1}) \cdot t \quad (3.5)$$

where t is the ratio of PV_1 to V_1V_2 .

3.3 Frame transformation

Frame transformation is an important technique used in SSM. There are various coordinate systems involved in the different machining stages. In this section, we discuss only discuss coordinate systems transformations used in 5-axis machining. The transformation techniques at other stages are simplified versions from those used in the 5-axis machining stage.

There are three coordinate systems involved in the CL point generation process at the 5-axis machining stage: the Global Coordinate System (GCS or the Global Frame, built with ${}^G X$, ${}^G Y$, ${}^G Z$), the Local Coordinate System (LCS or the Local Frame, built with ${}^L X$, ${}^L Y$, ${}^L Z$), and the Tool Coordinate System (TCS or the Cutter Frame, built with ${}^T X$, ${}^T Y$, ${}^T Z$) (Figure 3-3). GCS is used to record the coordinates of all triangles in the STL data source. The LCS is the local geometry feature frame constructed by ${}^L X$, ${}^L Y$ and ${}^L Z$. The vector ${}^L Z$, which follows along the normal (n) direction of the surface at the CC point, is calculated using Equation (3.6) and Equation (3.7). Vector ${}^L X$ is the feed direction (f), and is defined as the intersection line of the tangent plane at the CC point and the drive plane going through the CC point. Variable ${}^L Y$ is the cross product of ${}^L Z$ and ${}^L X$. TCS is the tool frame (Figure 3-3(b)). Variable ${}^T C$ is the cutter center point. Variable ${}^T X$ is the straight line connecting ${}^T C$ and CC point.

In 5-axis machining operations, the cutter can be translated along ${}^L Z$ by δ first, then rotated about ${}^L Y$ by λ , and finally rotated about ${}^L Z$ by ω as shown in Figure 3-3(a). After these ordered transformations, the relation of these three frames is derived as Equations (3.6), (3.7), (3.8), and (3.9).

$${}^G P = {}^G T_L \cdot {}^L P = \begin{bmatrix} {}^G X_{X_L} & {}^G X_{Y_L} & {}^G X_{Z_L} & {}^L X_{P_{cc}} \\ {}^G Y_{X_L} & {}^G Y_{Y_L} & {}^G Y_{Z_L} & {}^L Y_{P_{cc}} \\ {}^G Z_{X_L} & {}^G Z_{Y_L} & {}^G Z_{Z_L} & {}^L Z_{P_{cc}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot {}^L P \quad (3.6)$$

$${}^L P = {}^L T_G \cdot {}^G P = {}^G T_L^{-1} \cdot {}^G P \quad (3.7)$$

$${}^T P = {}^T T_L \cdot {}^L P = {}^T T_\delta \cdot {}^N T_\lambda \cdot {}^M T_\omega \cdot {}^L P$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\delta \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos \lambda & 0 & -\sin \lambda & 0 \\ 0 & 1 & 0 & 0 \\ \sin \lambda & 0 & \cos \lambda & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos \omega & \sin \omega & 0 & 0 \\ -\sin \omega & \cos \omega & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.8)$$

$$\cdot \begin{bmatrix} {}^L X_P \\ {}^L Y_P \\ {}^L Z_P \end{bmatrix} = \begin{bmatrix} \cos \lambda \cdot \cos \omega \cdot {}^L X_P + \cos \lambda \cdot \sin \omega \cdot {}^L Y_P - \sin \lambda \cdot {}^L Z_P \\ -\sin \omega \cdot {}^L X_P + \cos \omega \cdot {}^L Y_P \\ \sin \lambda \cdot \cos \omega \cdot {}^L X_P + \sin \lambda \cdot \sin \omega \cdot {}^L Y_P + \cos \lambda \cdot {}^L Z_P - \delta \end{bmatrix}$$

$${}^L P = {}^L T_T \cdot {}^T P = {}^T T_L^{-1} \cdot {}^T P \quad (3.9)$$

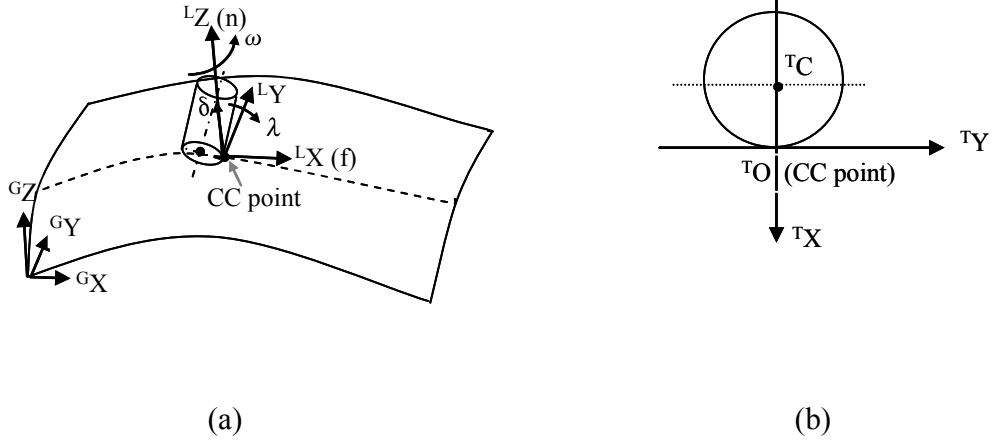


Figure 3-3 Frame Transformation.

where ${}^G P$ is the point coordinate with respect to the Global Frame, ${}^L P$ is the point coordinate with respect to the Local Frame and ${}^T P$ is the point coordinate with respect to the Tool Frame. Variable ${}^T T_L$ indicates the transformation of the frame L relative to the frame T.

The 3D C-space set $(\delta, \lambda, \omega)$ is specified with respect to the LCS. The cutter's posture at a CC point is uniquely determined by a fixed 3D C-space set.

3.4 Rough machining

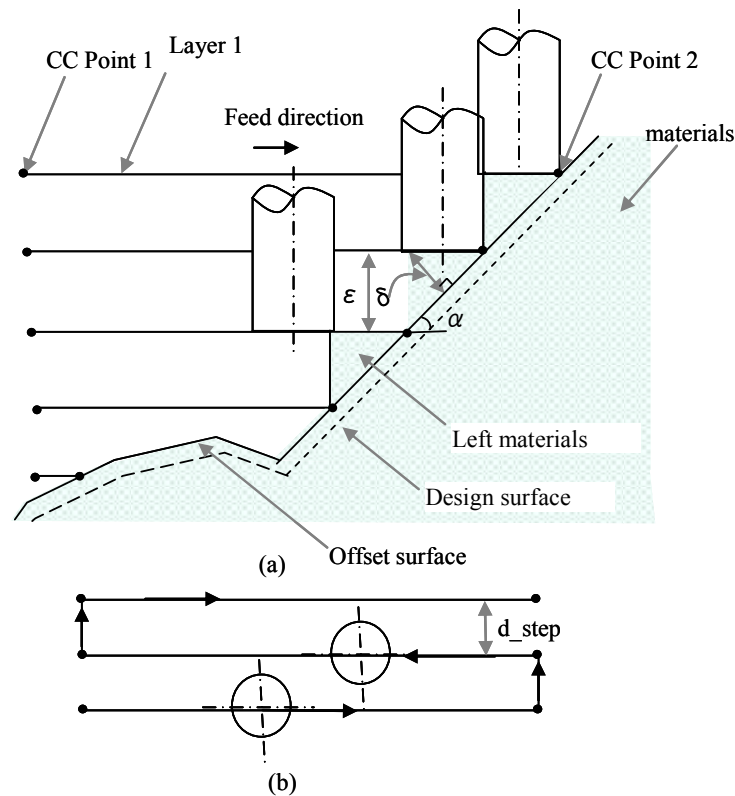


Figure 3-4 2.5-axis rough machining.

The purpose of rough machining is to quickly remove the correct material from the blank leaving a small amount of uncut allowance for semi-finish machining or finish machining. Figure 3-5 is an example of 2.5-axis rough machining. Operations of the rough machining stage can be divided into the 5 steps listed below and illustrated in Figure 3-5:

1. The machining method is determined, and the cutter—such as a ball end cutter, a fillet end cutter, or a flat end cutter—is selected.
2. The cutting depth is determined.
3. The offset surface on the tessellated surface is constructed.
4. The end point of the tool path is justified.
5. The CL tool path is generated.

These steps are all discussed in detail below.

3.4.1 Machining method determination and cutter selection

The “contour-map” approach is widely used at the rough machining stage of an SSM-process (2.5-axis rough machining method), where the cavity volume is sliced into equally spaced cutting layers as shown in Figure 3-5. The 2.5-axis rough machining method is applied to this dissertation based on shape analysis of tessellated surfaces machined. This method leads to a post-form surface with *stairs* (Figure 3-5). To decrease the height of the *stairs*, the ball end cutter is used as shown in Figure 3-5. Other cutters, such as the fillet end cutter or the flat end cutter, results in higher *stairs*.

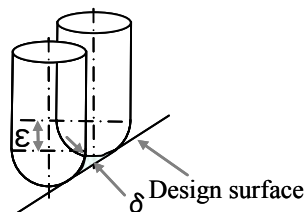


Figure 3-5 Remaining material height in the ball end cutter machining process

3.4.2 Cutting-depth determination

The remaining material height, which is determined by the cutting depth, is calculated with Equation (3.10).

$$\delta = R - \sqrt{R^2 - \left(\frac{\varepsilon}{2 \sin \alpha}\right)^2} \quad (3.10)$$

where ε is the z-offset cutting depth, δ is the remaining material height, α is the slope of the cross section of the surface, and R is the radius of the cutter. There are two factors that determine the cutting-depth: the remaining material height and the hardness of the material.

Equation (3.11), which is derived from Equation (3.10), is used to calculate the cutting depth according to the remaining material height (the rough machining tolerance).

$$\varepsilon = 2 \sin \alpha \sqrt{2\delta R - \delta^2} \quad (3.11)$$

In this dissertation, all benchmarks are made of clay or foam. As a result, the cutting depth, which is determined by the hardness of the material, is always a larger value than that calculated in Equation (3.11). Consequently, the cutting depth is determined by the remaining material height (machining tolerance). A detailed comparison of these two methods is needed if other materials (such as aluminum or steel) are used.

3.4.3 Offset surface determination

The *post-form* surface (the *output* state of the workpiece at the completion of an operation) in the rough-machining stage is not the design surface. Rather, it is the *offset-surface* of the design surface (Figure 3-5 (a)). The *offset-surface* here indicates a surface constructed by offsetting the design surface along a particular direction. Jun's work can be employed to offset the tessellated model surface. However, this method is computationally intensive and unnecessarily precise for rough machining (Figure 3-6 (a)).

A more reasonable method involving fewer calculations is to offset the vertex along the *average surface normal* (Figure 3-6 (b), and (c) as calculated in section 3.2). The accuracy of this method satisfies the rough machining tolerance requirements even though it is not as accurate as the method used in Jun's work.

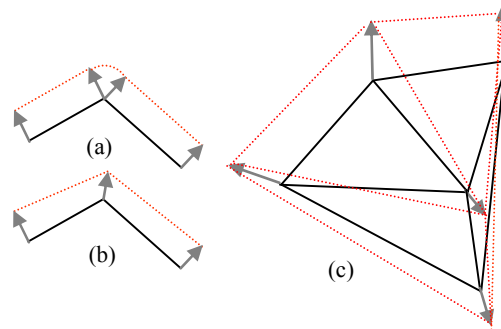


Figure 3-6 Offset of tessellation data

3.4.4 Justification of the end of a tool path

In the rough machining stage, each CC tool path is a straight line composed of two CC points: the start point and the end point (point 1 and point 2 in Figure 3-5 (a)). The start point of each tool path in Figure 3-5 is a boundary point on the raw stock, while more work is needed to determine the position of the end point of each gouge-free tool path.

Because the ball end cutter is re-parameterized from the generalized cutter, the method to find the position where the cutter contacts the offset surface at the end of the tool path will be covered later in section 3.5.4.

3.4.5 Tool path generation

The step-over distance d_{step} is the distance between two adjacent tool paths at the same cutting layer (Figure 3-5 (a)). In this case, it is set to a constant $c \cdot D$. The variable c is a safety factor. For best results, it is set to 0.95. The variable D is the diameter of the cutter. At the lowest cutting layer, the cusp height determined by the step-over distance may be larger than the rough machining tolerance, which means semi-finish machining is needed before finish machining operations are performed. In this dissertation, the 3-axis machining method is used for semi-finish machining. Finally, the CL tool paths are planned when all CL points are connected as shown in Figure 3-5.

3.5 Finish machining

Unlike rough machining, the purpose of finish machining is to mill the designed surface as precisely as possible. The point-based method and the curve-based method are two finish machining tool path planning methods. The comparison of these two methods is shown in Table 3-1. In the point-based method, the CL points, such as the center point of the cutter bottom, are generated based on CC points, which are the contact points between the cutter and the post-form surface. However, in the curve-based method, the CL points are generated based directly on the tessellated surface. The point-based method has no application limits while the curve-based method can only be applied to 3-axis machining. On the other hand, the point-based method can be used to eliminate gouging on CC points, while the curve-based method can be used to eliminate gouging along the whole tool path. No performance comparison between these two methods is currently available. As a result, both of these two methods exist in current TMM systems.

My research will focus on the point-based method because a combined 3-axis/5-axis TMM system is implemented in this dissertation.

Table 3-1 Comparison between the point-based method and the curve-based method.

Methods	CC points	CL Points	3-axis	5-axis	Robust
Point-based	Y	Y	Y	Y	N
Curve-based	N	Y	Y	N	Y

The tool path generation for the point-based machining method can be divided roughly into three steps: (1) Tool path topology selection, (2) Generation of CC points and (3) Elimination of gouging and the generation of CL points.

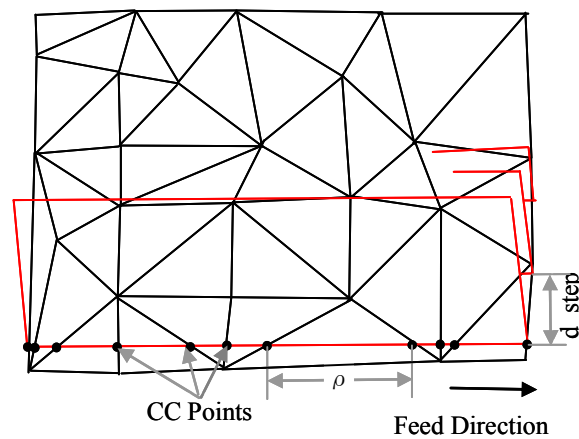


Figure 3-7 Generation of CC points

3.5.1 Tool path topology selection

There are four main tool path topologies used for most industry practices: (1) The serial pattern (e.g. ${}^G X^G Y$ -parallel, boundary curves parallel and boundary curves normal);

(2) The radial pattern (e.g. spiral and contour parallel offset); (3) The strip pattern (e.g. strip-parallel and strip-normal); and (4) The contour pattern (e.g. helical, z-constant and BC-parallel) [11]. This research uses the ${}^G X^G Y$ -parallel topology to plan the tool path. Both guide planes and drive surfaces can be used to generate the tool path. Since the drive surface method is the standard as established by APT, we use this method in our research.

CC points are the intersection points where the model surface meets the set of drive planes parallel with the ${}^G X^G Z$ or ${}^G Y^G Z$ plane (Figure 3-7). The cutter is placed on these CC points. The distance between two consecutive CC points is the step-forward distance ρ , which is determined by the triangle geometry feature and the drive plane position. More CC points need to be inserted between two consecutive CC points if the normal difference of these two CC points is larger than the threshold determined by the mill machine.

The step-over distance is the distance between two consecutive drive planes. The positions of the drive planes are determined by the cutter orientation, the local geometry features of the sculpture surface, and the mill allowance.

To automate the tool path pattern selection, a simple method is used—the chosen tool path may be parallel either to the ${}^G X^G Z$ plane or ${}^G Y^G Z$ plane. If δN along the direction parallel to ${}^G X^G Z$ is larger than it is along the direction parallel to ${}^G Y^G Z$ direction, the tool path parallel to ${}^G Y^G Z$ plane is chosen. The average normal change δN along a tool path is calculated with Equation (3.12):

$$\delta N = \sqrt{\frac{\sum_{i=2}^n \left(\frac{N_{V_i} - N_{V_j}}{d_{V_i V_j}} \right)^2}{n-1}} \quad (3.12)$$

where n is the number of CC points along one tool path, $d_{V_i V_j}$ is the distance between two consecutive CC points, and N_{V_i} is the normal of CC point i .

3.5.2 CC points generation

3.5.2.1 Identification of the safe machining region

CC points denote actual contact points where the cutter touches the post-form surface. In practice, CC points are planned based on the design surface because the post-form surface is not known until the machining process is completed. In tessellated surface machining, CC points have no relationship to the original design surface because the tessellation is typically taken from white light or laser scanning. This point cloud data is edited and the post-form surface is re-machined. In this dissertation, tessellated surfaces are generated from parametric design surfaces; the accuracy of these tessellations is controlled by the tessellation tolerance parameter. Figure 3-8 shows the difference of the given tessellated surface S and the design surface. The tessellation tolerance τ_t indicates the tolerance of the tessellation from the parametric design surface to the tessellated surface. The given polygon surface in this dissertation is tessellated from the parametric design surface. The variables τ_t^+ and τ_t^- are the *out* and *in* tessellation tolerances, respectively. In Figure 3-8, the surface S_1 is obtained by offsetting the design surface S τ_t^- upwards, and the surface S_2 is obtained by offsetting S τ_t^+ downwards. The design surface may be anywhere in the region between S_1 and S_2 . The variables τ_m^+ and τ_m^- are the *out* and *in* machining tolerances, respectively. The value for S_{11} is calculated by offsetting S_1 τ_m^- upwards, and S_{12} is calculated by offsetting S_1 τ_m^+

downwards. The value for S_{21} is calculated by offsetting S_2 τ_m^- upwards, and S_{22} is calculated by offsetting S_2 τ_m^+ downwards.

The region bounded by S_{21} and S_{12} is then the safe machining region, as calculated in Equation (3.15). The offset distance between S_{21} and S is τ^+ , while the offset distance between S_{12} and S is τ^- . The relationships between all tolerances are given by Equations (3.13), (3.14), and (3.15).

$$\tau_t = \tau_t^+ + \tau_t^- \quad (3.13)$$

$$\tau_m = \tau_m^+ + \tau_m^- \quad (3.14)$$

$$\begin{cases} \tau^+ = \tau_m^- - \tau_t^+ \\ \tau^- = \tau_m^+ - \tau_t^- \\ \tau = \tau^- + \tau^+ \end{cases} \quad (3.15)$$

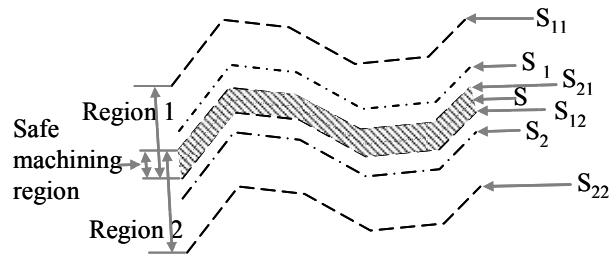


Figure 3-8 Difference of the safe machining surface and the given tessellated surface.

In this dissertation, the bottom boundary surface S_{12} of the safe machining region is treated as the design surface where CC points are planned. The region under S_{12} is the gouging prone region because the cutter may gouge on the original design surface if the CC point is set inside this region. The region above S_{21} is named the scallop or cusp region because there will be unwanted uncut material left if the CC point is set inside this

region. CC points can be adjusted anywhere inside the safe machining region to eliminate gouging or to improve the machining speed. Hereafter, the design tessellated surface in this dissertation refers to S_{I2} . The machining tolerance refers to τ which is set to a positive value.

3.5.2.2 Step-over distance determination

There are two methods to determine the step-over distance. The first method is to set the distance to a constant value. For example, the step-over distance can be set to one quarter of the cutter diameter. If the machining tolerance is known, the step-over distance will be the minimum value with which the desired surface quality is achieved. As a result, the cusp height determined by the first method may be unnecessarily low, and the machining speed of this method is not satisfactory. The second method is to determine the step-over distance according to the local geometry feature: the surface normal and the local curvature at a CC point. In this dissertation, the second method is used to decrease the machining time.

CC points are obtained by slicing the tessellated surface with a set of drive planes parallel to the ${}^G X^G Z$ or ${}^G X^G Y$ plane (Figure 3-9). The cutter is placed on these CC points. The step-over distance (the tool path interval) d_{step_i} is the distance between two consecutive drive planes DP_i and DP_{i+1} . The position of the drive planes is determined by the cusp height, which is based on the cutter orientation, the cutter shape, the topology of the surface, and the machining tolerance. For the purposes of this dissertation, the step-over distance was set to the largest possible value to decrease the machining time.

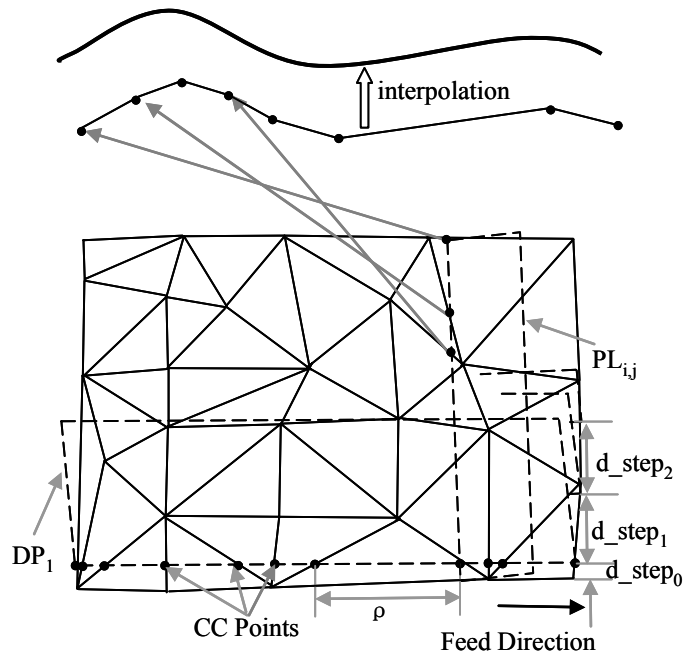


Figure 3-9 Generation of CC points.

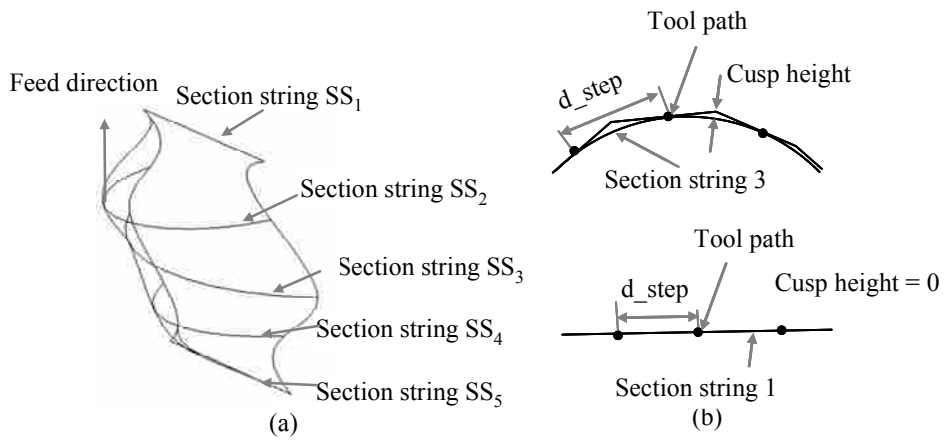


Figure 3-10 A spindle surface.

As shown in Figure 3-10, the through-curve surfaces are composed of five section strings ($SS_1, SS_2, SS_3, SS_4, SS_5$). The first section string (SS_1) and the final section string (SS_5) are both straight lines. Using regular methods, the step-over distance is determined according to the boundary curve (such as SS_1 or SS_5). In Figure 3-10, the step-over distance is set to D (the diameter of the cutter) since both SS_1 and SS_5 are straight lines where the cusp height is 0. However, the curvature of other section strings is not zero, as shown in Figure 3-10(b), which means the remaining material height at different section strings is larger than that at SS_1 or SS_5 along the same tool path. Consequently, the cusp height may be larger than the machining tolerance if the step-over distance is determined according to SS_1 or SS_5 . The cusp height may still be larger than the machining tolerance even if string 3 is used to determine the step-over distance because the curvature of string 2 and the shape of string 4 are not involved in determining the step-over distance. It will be challenging to determine the precise step-over distance for an even more complicated surface using existing methods. The method developed for this dissertation can be used to solve this problem.

Before we discuss the techniques used to determine the step-over distance, we must first introduce some equations and definitions that are used later:

The CC point surface, which is composed of all CC points, is calculated by:

$$S_CCTP = \{CCTP_i | i = 1..n\} \quad (3.16)$$

The CL point surface, which is composed of all CL points, is calculated by:

$$S_CLTP = \{CLTP_i | i = 1..n\} \quad (3.17)$$

The CC tool path, which is composed of all CC points ($CCPt_{i,j}$), generated by one drive plane DP_i , is calculated by:

$$CCTP_i = \{CCPt_{i,j} | j = 1..m\} \quad (3.18)$$

The CL tool path, which is composed of all CL points ($CLPt_{i,j}$) related to CC points ($CCPt_{i,j}$), is calculated by:

$$CLTP_i = \{CLPt_{i,j} \mid j = 1..m\} \quad (3.19)$$

The 3D C-space related to $CCTP_{i,j}$ is calculated by:

$$V_{i,j} = \left\{ (\lambda, \omega, \delta) \mid \lambda \in (0, \frac{\pi}{4}), \omega \in (-\frac{\pi}{4}, \frac{\pi}{4}), \delta \in (0, \tau) \right\} \quad (3.20)$$

For the remaining material height as shown in Figure 3-11:

The left side of the $CCTP_{i,j}$ is calculated using $\zeta_{i,j,L}(d_1)$, where d_1 is along LY of LCS.

The right side of the $CCTP_{i,j+1}$ is calculated using $\zeta_{i,j+1,R}(d_2)$, where d_2 is along LY of LCS.

The maximum remaining material height point (MRMH point) is calculated using:

$CCPt_{i,a_0}$, which is the point where the remaining material height is the maximum on the left side of $CCTP_i$.

$CCPt_{i,b_0}$, which is the point where the remaining material height is the maximum on the right side of $CCTP_i$.

The step over distance between $CCTP_i$ and $CCTP_{i+1}$ is calculated by:

$$d_step_i = d_step_{i,L} + d_step_{i+1,R} \quad (3.21)$$

where $d_step_{i,L}$ and $d_step_{i+1,R}$ are indicated in Figure 3-12(a).

$C_{i,j}$ is the intersection curve of the design surface and the plane perpendicular to the drive plane as shown in Figure 3-11(a).

$P_{i,j}(d)$ is a point on the curve $C_{i,j}$, while $P_{i,j}(\alpha)$ is a point at the cutter bottom.

$P_{i,j}(d)$ and $P_{i,j}(\alpha)$ have the same Y in LCS. The difference (Z) between $P_{i,j}(d)$ and

$P_{i,j}(\alpha)$ is the remaining material height.

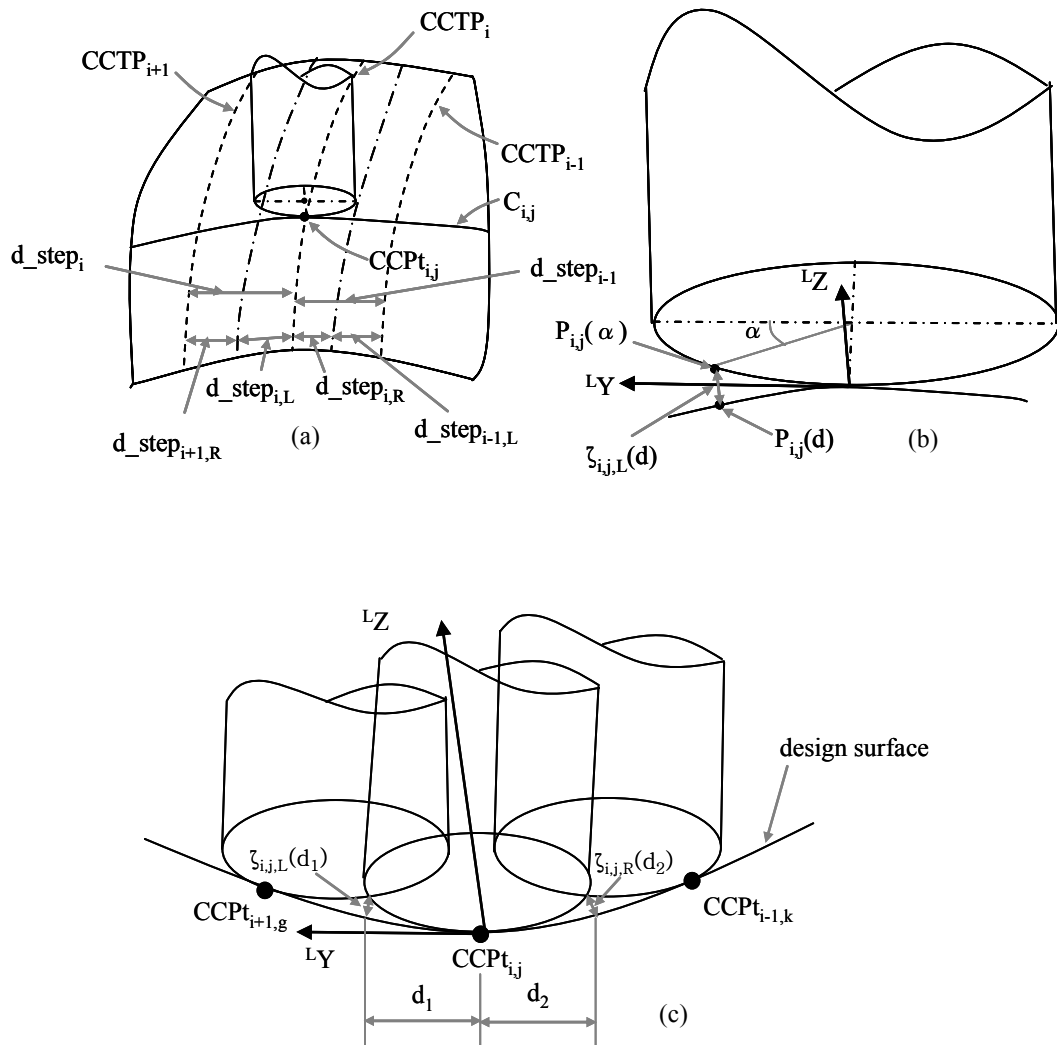


Figure 3-11 Cross section of the design surface at a CC point.

The method used to determine the step-over distance for 5-axis machining and 3-axis machining is different. Basically, the difference lies on the cutter direction in the machining process. In 5-axis machining, there is an inclination angle; while in 3-axis machining, the cutter posture is fixed.

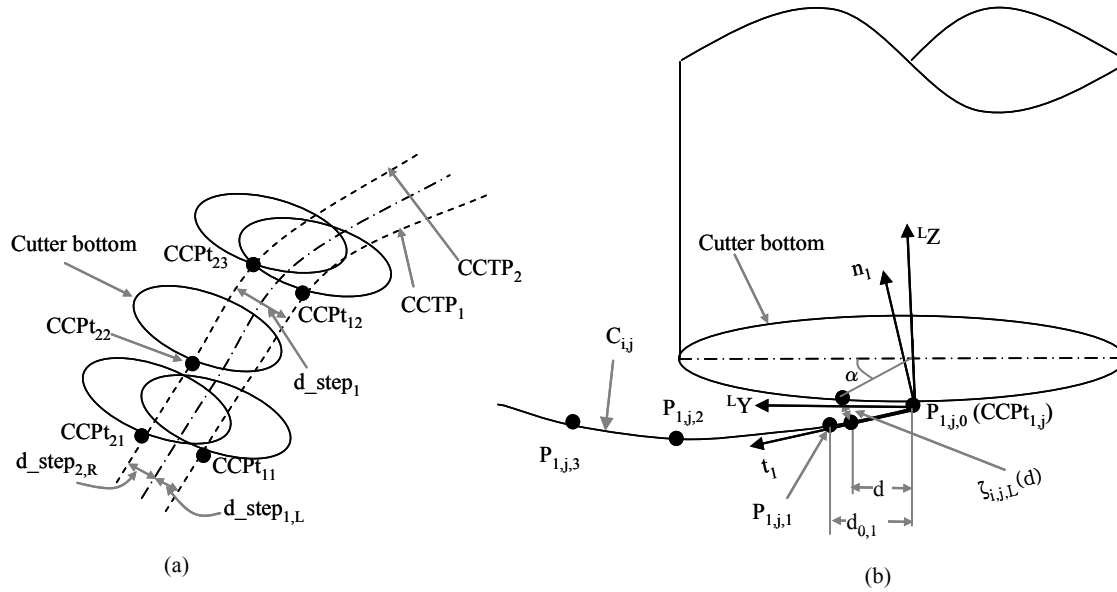


Figure 3-12 Step-over distance determination.

Here, we will discuss the step-over distance determination of 5-axis machining and 3-axis machining separately.

- **Deciding the step-over distance in five-axis machining**

1. Set the position (d_step_0) of the first CC point tool path ($CCTP_1$) as shown in Figure 3-9. d_step_0 is set to one quarter of the cutter radius as shown in Equation (3.22).

$$d_step_0 = 0.125 \cdot D \quad (3.22)$$

2. Find the maximum remaining material height $\zeta_{1,a_0,L}(0.25 \cdot D)_{\max}$ left of the first CC point tool path $CCTP_1$ using Equation (3.23).

$$\zeta_{1,a_0,L}(d)_{\max} = \max \{ \zeta_{1,j,L}(d), j = 1 \dots m \} \quad (3.23)$$

where $d = 0.25 \cdot D$

The CC point $CCPt_{1,a_0}$ is the maximum remaining material height point (MLMH point), i.e. the point where the remaining material height left of $CCTP_I$ is the maximum.

The following sub-steps are used to explain this step.

- At each CC point $CCPt_{1,j}$, a plane $PL_{1,j}$ perpendicular to the drive pane DP_I is derived. The intersection curve $C_{1,j}$ of $PL_{1,j}$ and the design surface S is obtained as shown in Figure 3-9 (Lauwers et al. 2003).

$$C_{1,j} = PL_{1,j} \cap S = \{P_{1,j,k} \mid k = -l..l\} \quad (3.24)$$

- Obtaining the cutter posture is the precondition to the remaining material height determination. In this dissertation, the cutter posture $(\delta^*, \lambda^*, 0)$ at CC point $CCPt_{1,j}$ is determined with the CM^2 method (ω^* is always 0). However, $(\delta^*, \lambda^*, 0)$ is not the final optimal cutter posture generated in this dissertation because the cutter posture cannot be determined exactly until the final optimal CL point is just an estimate. As a result, the remaining material height here is not precise. An ellipse E is obtained (Figure 3-12(b)) if the cutter bottom silhouette is projected to $PL_{1,j}$. The remaining material height $\zeta_{1,j,L}(0.25 \cdot D)$ for each CC point $CCPt_{1,j}$ is the distance along LZ direction from a point $P_{i,j,(d)}$ on $C_{1,j}$ to a point $P_{i,j,(\alpha)}$ on the cutter bottom as calculated using Equations (3.25) and (3.27), where α is set to $\pi/3$ (d is $0.25D$) and $P_{i,j,k}$ is the point $0.25D$ away from ${}^T O$.

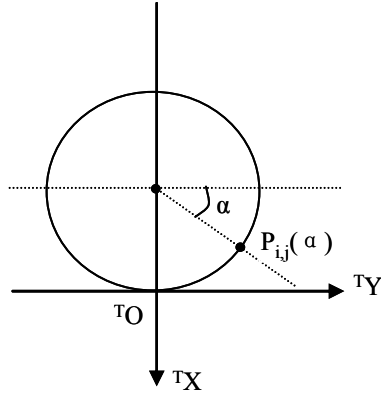


Figure 3-13 Bottom of the flat end cutter.

$$\begin{cases} {}^T X_{P_{i,j}(\alpha)} = -R + R \cdot \sin \alpha \\ {}^T Y_{P_{i,j}(\alpha)} = R \cdot \cos \alpha \\ {}^T Z_{P_{i,j}(\alpha)} = 0 \end{cases} \quad (3.25)$$

Substituting Equation (3.25) into Equation (3.9), yields ${}^L P_{i,j}(\alpha)$.

$P_{i,j}(d)$ in Figure 3-12(b) is a point on the curve C_{ij} that has the same ${}^L Y$ as

$P_{i,j}(\alpha)$

$${}^L Y_{P_{i,j}(d)} = {}^L Y_{P_{i,j}(\alpha)} \quad (3.26)$$

Since

$${}^L P_{i,j}(d) = t \cdot {}^L P_{i,j,k} + (1-t) \cdot {}^L P_{i,j,k+1} \quad (3.27)$$

where $P_{i,j}(d)$ is between $P_{i,j,k}$ and $P_{i,j,k+1}$

and t is

$$t = \frac{d - {}^L Y_{P_{i,j,k}}}{{}^L Y_{P_{i,j,k}} - {}^L Y_{P_{i,j,k+1}}} = \frac{R \cdot \cos \alpha - {}^L Y_{P_{i,j,k}}}{{}^L Y_{P_{i,j,k}} - {}^L Y_{P_{i,j,k+1}}} \quad (3.28)$$

the remaining material height is:

$$\zeta_{i,j,L}(d) = {}^L Z_{P_{i,j}(\alpha)} - {}^L Z_{P_{i,j}(d)} \quad (3.29)$$

Then, with ${}^L P_{i,j}(\alpha)$ and Equation (3.27), $\zeta_{1,j,L}(0.25 \cdot D)$ is obtained:

$$\begin{aligned} \zeta_{i,j,L}(d) = & -\sin \lambda \cdot (-R + R \cdot \sin \alpha) + \cos \lambda \cdot \delta - \left(\frac{R \cdot \cos \alpha - {}^L Y_{P_{i,j,k}}}{{}^L Y_{P_{i,j,k}} - {}^L Y_{P_{i,j,k+1}}} \cdot {}^L Z_{P_{i,j,k}} \right. \\ & \left. + \left(1 - \frac{R \cdot \cos \alpha - {}^L Y_{P_{i,j,k}}}{{}^L Y_{P_{i,j,k}} - {}^L Y_{P_{i,j,k+1}}} \right) \cdot {}^L Z_{P_{i,j,k+1}} \right) \end{aligned} \quad (3.30)$$

where α is set to $\pi/3$ and i is 1.

- The next step is to compare $\zeta_{1,j,L}(0.25 \cdot D)$ generated with Equation (3.30) and get $\zeta_{1,a_0,L}(0.25 \cdot D)_{\max}$ with Equation (3.23).
3. α is obtained from Equation (3.32), which is simplified from Equation (3.30).

$$\zeta_{1,a_0,L}(d) = \tau \quad (3.31)$$

$$\alpha = \sin^{-1} \left(\frac{C}{\sqrt{A^2 + B^2}} \right) \quad (3.32)$$

where

$$A = -R \cdot \sin \lambda$$

$$B = \frac{{}^L Z_{P_{i,a_0,k}} - {}^L Z_{P_{i,a_0,k+1}}}{{}^L Y_{P_{i,a_0,k}} - {}^L Y_{P_{i,a_0,k+1}}} \cdot R \cdot \cos \alpha$$

$$C = \tau + {}^L Z_{P_{i,a_0,k+1}} - R \cdot \sin \lambda - \cos \lambda \cdot \delta - \frac{{}^L Z_{P_{i,a_0,k}} - {}^L Z_{P_{i,a_0,k+1}}}{{}^L Y_{P_{i,a_0,k}} - {}^L Y_{P_{i,a_0,k+1}}} \cdot {}^L Y_{P_{i,a_0,k}}$$

After α is calculated using Equation (3.32), $d_{step_{1L}}$ is obtained by transforming $P_{i,j}(\alpha)$ to GCS with Equation (3.6).

Because the cutter posture that is used to find the MRMH point is not precise, the calculated step-over distance is also not accurate.

A safety factor $c=0.9$ is used for the step-over distance in this dissertation as in Equation (3.33).

$$d_step_{1,L} = c \cdot R \cdot \cos \alpha \quad (3.33)$$

4. Determine $d_step_{2,R}$, the value of $d_step_{2,R}$ which cannot be determined directly with the methods in step 3. However, using the methods in step 3 to calculate $d_step_{2,R}$, we can first determine the CC point $CCPt_{2,j}$. Obtaining $d_step_{2,R}$ is also the precondition to determine the CC point $CCPt_{2,j}$. The $d_step_{2,R}$ value is approximated here to solve this problem. The following sub-steps lead to our solutions.
 - Approximate $d_step_{2,R}$ ($d_step_{2,R}^*$) to determine the location of the second tool path. In this dissertation, $d_step_{2,R}^*$ is set as equal to $d_step_{1,L}$.
 - Generate CC points $CCPt_{2,j}^*$ with the approximated second tool path. $CCPt_{2,j}^*$ is not the same as $CCPt_{2,j}$ because $CCPt_{2,j}^*$ is the approximated CC point.
 - Approximate the remaining material height $\zeta_{2,j,R}(0.25 \cdot D)$ along the approximated second tool path $CCPt_{2,j}^*$ (Figure 3-12(b)) with the methods from step 3. Then calculate the MRMH CC point P_{1,b_0} for the approximated second tool path $CCPt_{2,j}^*$, where the remaining rightside material height right of the second tool path is the maximum. The distance $d_step_{2,R}$ is still approximated with methods in step 3. Again, a safety factor 0.9 is adopted.
5. Repeat step 3 and step 4 to generate other CC tool paths $CLTP_i$ ($i > 2$).

With the above five steps, the CC point tool path can be generated.

- **Determining the step-over distance in three-axis machining**

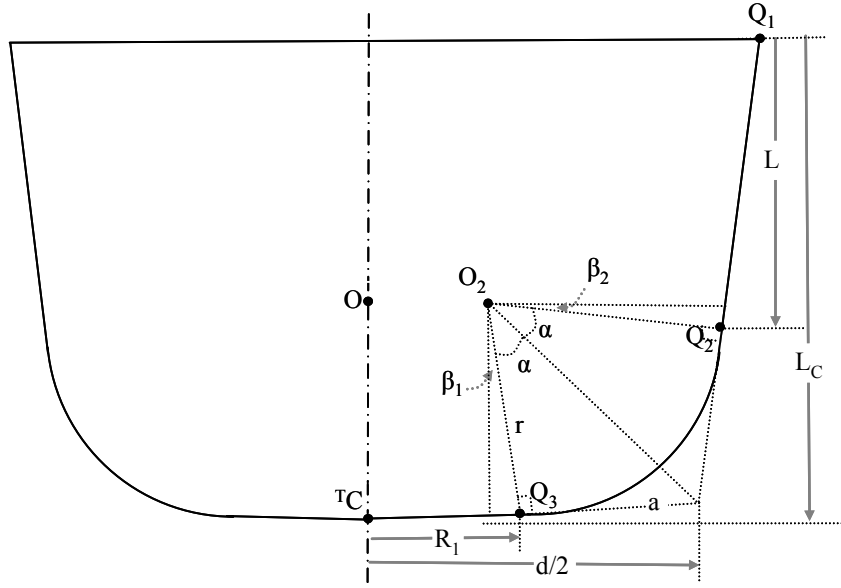


Figure 3-14 The generalized cutter ([54]).

In 3-axis machining, the determination of the step-over distance is similar to that used in 5-axis machining. However, one difference is that in order to eliminate the gouging at a CC point, the cutter can only be lifted along GZ direction. Another difference is that the generalized cutter, instead of the flat end cutter, is used in 3-axis machining. The shape of the generalized cutter is more complex than that of the flat end cutter.

The generalized APT cutter is shown in Figure 3-14. TC is the cutter center point. O_2 is the center point of the corner circle. O is the center point of the torus region. If the generalized cutter is re-parameterized to be a ball-end cutter, O and O_2 will be the same point. And if the generalized cutter is re-parameterized to be a flat end cutter, O and TC will be the same point. The cross section of the cutter can be divided into three regions:

the taper region (from Q_1 to Q_2), the torus region (from Q_2 to Q_3), and the cone region (from Q_3 to TC). All parameters of the generalized cutter can be derived from the followings ones:

L_C : height of the cutting edge as measured along the cutter axis.

β_1 : angle from a radial line through the cutter tip to the cutter bottom, $0^\circ \leq \beta_1 \leq 90^\circ$.

β_2 : taper angle between the cutter side and the cutter axis; it is positive when sloping outward and negative sloping inward from the cutter side, $-90^\circ \leq \beta_2 \leq 90^\circ$.

R : cutter radius.

r : corner radius.

- **Basic geometric parameters of the generalized cutter**

The generalized cutter parameters are calculated as follows:

$$(L_C - \frac{d}{2} \cdot \tan \beta_1) \cdot \tan \beta_2 + \frac{d}{2} = R \quad (3.34)$$

$$d = 2 \cdot \frac{(R - L_C \cdot \tan \beta_2)}{(1 - \tan \beta_1 \cdot \tan \beta_2)} \quad (3.35)$$

$$\alpha = \frac{(0.5 \cdot \pi - \beta_1 - \beta_2)}{2} \quad (3.36)$$

$$a = r \cdot \tan \alpha \quad (3.37)$$

$$L_C - L = a \cdot \sin \beta_2 + \frac{d}{2}$$

$$L = L_C - (a \cdot \cos \beta_2 + \frac{d}{2} \cdot \tan \beta_1) \quad (3.38)$$

$$R_1 = \frac{d}{2} - a \cdot \cos \beta_1 \quad (3.39)$$

$$f = (L_C - L) + r \cdot \sin \beta_2 \quad (3.40)$$

$$e = R_1 - r \cdot \sin \beta_1 \quad (3.41)$$

$$R_2 = R - L \cdot \tan \beta_1 \quad (3.42)$$

Assume point P is a point on the cross section of the cutter. R_P is the distance between the point P and the cutter axis. The equations of different regions of the cutter are listed as follows:

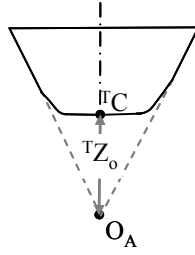


Figure 3-15 Tip of a generalized cutter.

- Taper region:

$$\frac{\sqrt{{}^T X_P^2 + {}^T Y_P^2}}{{}^T Z_P - {}^T Z_0} = \frac{R - R_2}{R} \quad (3.43)$$

where

$${}^T Z_0 = L_C - R \cdot c \tan \beta_2 \quad (3.44)$$

$$R_p = R - (L_C - {}^T Z_P) \cdot \tan \beta_2 \quad (3.45)$$

$${}^T Z_P = L_C - \frac{R - R_p}{\tan \beta_2} \quad (3.46)$$

and ${}^T Z_0$ is the distance from the apex (O_A) of the taper region to the cutter center point (${}^T C$) as shown in Figure 3-15.

- Torus region:

$$({}^T X_P^2 + {}^T Y_P^2 + {}^T Z_P^2 + e^2 - r^2)^2 - 4e^2({}^T X_P^2 + {}^T Y_P^2) = 0 \quad (3.47)$$

The center point of the torus is set to ${}^T C$ in Equation (3.47).

$$R_p = e + \sqrt{r^2 - (f - {}^T Z_P)^2} \quad (3.48)$$

$${}^T Z_P = f - \sqrt{r^2 - (R_p - e)^2} \quad (3.49)$$

- Cone region:

$$\frac{\sqrt{{}^T X_P^2 + {}^T Y_P^2}}{{}^T Z_P} = c \tan \beta_1 \quad (3.50)$$

$$R_p = \frac{{}^T Z_p}{\tan \beta_1} \quad (3.51)$$

$${}^T Z_p = R_p \cdot \tan \beta_1 \quad (3.52)$$

- **Determination of the step-over distance in five-axis machining**

Most of the steps to determine the step-over in 3-axis machining are the same as those in 5-axis machining. To avoid the confusion, detailed steps are listed as follows. The steps that are exactly the same as those in 5-axis machining are not discussed again.

1. Set the position (d_step_0) of the first CC point tool path ($CCTP_1$), similar to that in 5-axis machining. Again, here, we set the first tool path to be $0.25R$ where R is the cutter diameter. Basically, 0.25 is an approximated value which may not be accurate. In some cases, the scallop height right to the first tool path may be larger than the machining tolerance. If this happens, the factor (0.25) should be decreased manually. According to our experiences, 0.25 is a reasonable value for most cases.

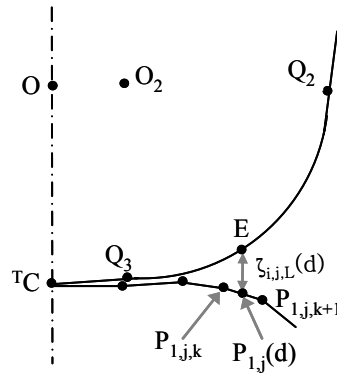


Figure 3-16 Expanded section of the generalized cutter

As shown in Figure 3-17, a straight line consisting of two end points can be represented with:

$$\frac{{}^T Y - {}^T Y_1}{{}^T X - {}^T X_1} = \frac{{}^T Y_2 - {}^T Y_1}{{}^T X_2 - {}^T X_1} = \eta \quad (3.54)$$

or

$$A^T X + B^T Y + c = 0 \quad (3.55)$$

where

$$A = \frac{\eta}{\sqrt{\eta^2 + 1}}$$

$$B = \frac{-1}{\sqrt{\eta^2 + 1}}$$

$$C = \frac{{}^T Y_1^2 - \eta \cdot {}^T X_1}{\sqrt{\eta^2 + 1}}$$

The distance between two points along ${}^T Y$ axis is here represented as d .

If δ is between d_1 and d_2 , then E is between B and Q . The value of E is obtained with

$$E = Q + (E - Q) \frac{\delta - d_1}{d_2 - \delta} \quad (3.56)$$

If E is between Q_2 and Q_3 and P is between $P_{1,j,k}$ and $P_{1,j,k+1}$, then E is found.

Otherwise, go to the next case.

- E is in the torus region (Figure 3-17(b)).

Again, if δ is between d_1 and d_2 , then E is between B and Q . The value for α is obtained to calculate E :

$$\delta = (-r \sin \alpha - f) - \left[-\frac{A(r \cos \alpha - e) + C}{B} \right] \quad (3.57)$$

which can be simplified to

$$\alpha = \sin^{-1} \left(\frac{C - B(d + f) - Ae}{r\sqrt{A^2 + B^2}} \right) + \beta \quad (3.58)$$

where

$$\begin{cases} \sin \beta = \frac{A}{\sqrt{A^2 + B^2}} \\ \cos \beta = \frac{B}{\sqrt{A^2 + B^2}} \end{cases}$$

E can then be calculated using α .

If E is between Q_2 and Q_3 and P is between $P_{1,j,k}$ and $P_{1,j,k+1}$, E is found.

Otherwise, go to the next case.

- E is in the taper region.

The method in this case is the same as that of cone region, which is not discussed again in this section. The only difference is that the parameters of the taper and the cone are different.

The step-over distance is the distance between point E and the CC point along the ${}^G X$ direction.

4. Same as that of 5-axis machining.
5. Repeat step 3 and step 4 to generate other CC tool paths $CLTP_i(i>2)$.

The step-over distance can be calculated using the above five steps. Then you can generate CC point tool paths by slicing the tessellated surface with drive planes.

3.5.3 Gouging recognition

The cutter is placed on the CC points. In 3-axis machining, the tip (${}^T C$) of the generalized cutter is placed on the CC point, and the initial cutter direction is set parallel to ${}^G Z$. In 5-axis machining, a cutter bottom boundary point is placed on the CC point (Figure 3-18) and the initial cutter direction is set parallel to the CC point normal. With the initial cutter posture, the cutter may gouge the tessellated surface. Gouging

recognition methods and gouging elimination methods need to be employed to plan the optimal CL tool path.

Before building a 3D C-space, gouging problems must first be eliminated. As discussed in the introduction, gouging problems can be divided into three cases: local gouging, global gouging, and cutter head gouging (Figure 3-18).

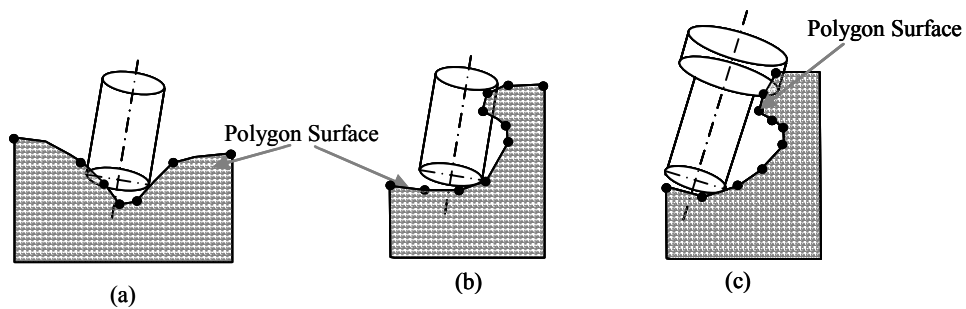


Figure 3-18 Different gouging cases.

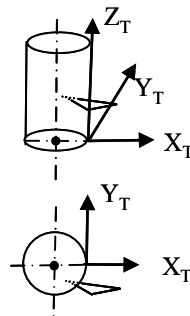


Figure 3-19 The yaw angle is needed in this situation.

The graphic shown in Figure 3-19 is a special gouging case. In this situation, a rotation about Z_T by the yaw angle is a more efficient way to eliminate the gouging than the rotation about Y_T by the inclination angle or the translation along Z_T by δ .

Consequently, it is necessary to involve the yaw angle in the gouging elimination method even though it has normally not been considered in most of the work in this area.

Gouging recognition methods are different for each cutter. In this section, only the flat end cutter is involved. Gouging recognition for a generalized cutter is discussed together with gouging elimination for a generalized cutter.

Li and Jerard have presented methods to recognize and eliminate different gouging situations (Figure 3-20, Li and Jerard 1994) for a triangle with respect to TCS. Equation (3.59) can be obtained from Li's work.

$$f({}^T\Delta P_1P_2P_3) = \begin{cases} a > 0 & \text{gouging} \\ a = 0 & \text{contact} \\ a < 0 & \text{separate} \end{cases} \quad (3.59)$$

where ${}^T\Delta P_1P_2P_3$ is a triangle with respect to TCS. If a is less than 0, the cutter gouges the triangle. Refer to Li's paper (Li and Jerard 1994) if details are needed.

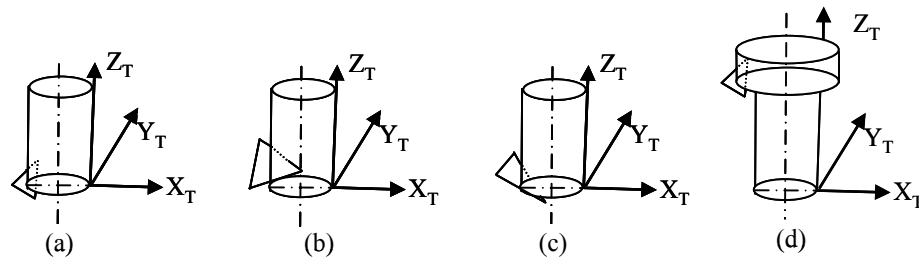


Figure 3-20 Different gouging cases.

If gouging may occur in a particular facet, that facet is considered gouging-possible. Not all facets are gouging-possible at a particular CC point in the tessellated model. Figure 3-21 shows the process of recognizing the gouging-possible facets.

First, a circle with a radius equal to the cutter length is drawn. All facets that intersect with this circle are gouging-possible facets. These facets are numbered in order (i.e., 1, 2, 3...N). The C-space $V_{i,j}$ at the CC point P is then obtained with Equation (3.60).

$$V_{i,j} = \bigcap_{k=1}^N v_k \quad (3.60)$$

where v_k is the C-space for facet k.

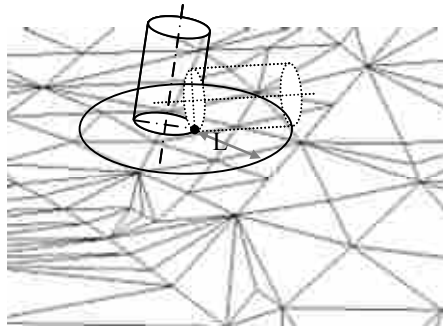


Figure 3-21 Gouging possible facets.

3.5.4 CL points generation of three-axis machining

In 3-axis machining, if a triangle is gouged by a cutter, then the only way to eliminate this gouging problem is to lift the cutter along the ${}^G Z$ direction. The generalized end cutter is the base model of all types of cutters and is used to machine the models in this dissertation. With the exception of a flat end cutter, various cutters—such as a fillet end cutter, a ball end cutter, etc.—can also be used in the 3-axis machining process. The main algorithms used to eliminate generalized end cutter gouging in 3-axis machining are discussed in Yau's work [54]. The method used in this work is a modified

version of the vertex gouging elimination method used in Yau's work. For clarity and completeness, we will also introduce the edge gouging elimination methods and triangle gouging elimination methods used in Yau's paper. These methods will also be used partly in clean-up machining. For more detailed information, refer to Yau's paper.

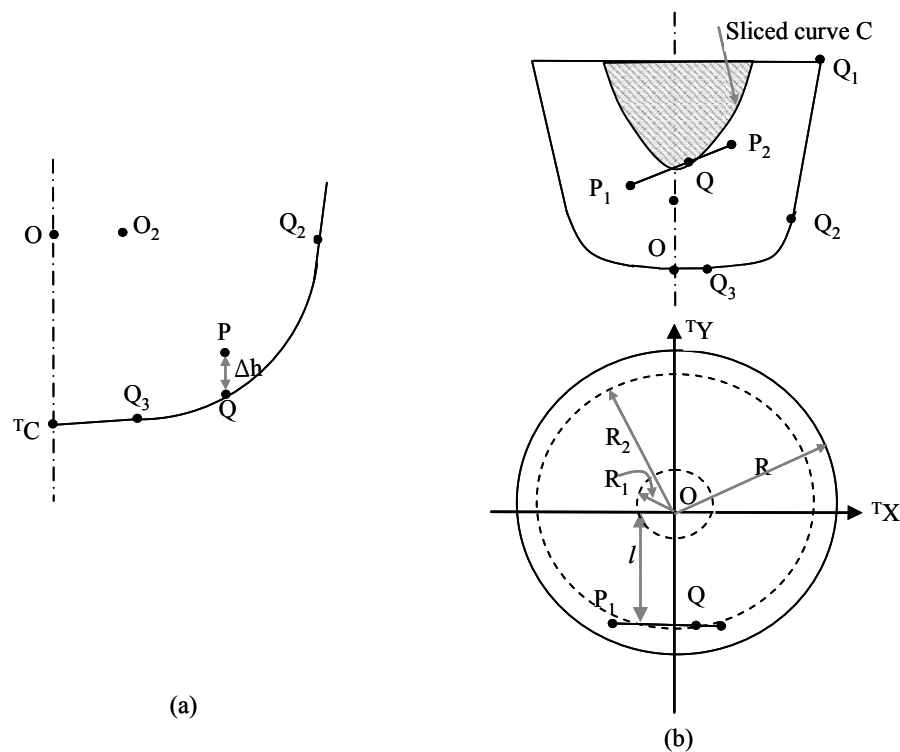


Figure 3-22 Gouging problems between a generalized cutter and a triangle.

Instances where cutter gouging occurs on a triangle can be divided into vertex gouging, edge gouging, and triangle gouging as in Li's work (Li and Jerard 1994). In each instance, the geometric entity (vertex, edge, or triangle) may contact the cutter in different regions.

1. Vertex Gouging (Figure 3-22(a)).

For a triangle that is being gouged by the cutter, we designate the triangle vertex as P .

The distance from the vertex P to the cutter axis is:

$$\rho = \sqrt{{}^T X_P^2 + {}^T Y_P^2} \quad (3.61)$$

Substitute ${}^T Z_V$ into Equations (3.45), (3.48), (3.51) to determine the region in which P is located. Point Q , which is a point on the cutter that has the same ${}^T X$ and ${}^T Y$ value with P . ${}^T Z_Q$, is obtained using Equations (3.46), (3.49), or (3.52) according to the location of P .

The distance from the vertex P to the point Q is given by:

$$\Delta h = {}^T Z_P - {}^T Z_Q \quad (3.62)$$

where Δh is the cutter lifting height needed to eliminate the vertex gouging. If Δh is larger than 0, this vertex is gouged by the cutter. If Δh is less than 0, there is no vertex gouging.

2. Edge Gouging (Figure 3-22(b)).

Each edge $\overline{{}^T P_1 {}^T P_2}$ of a triangle can be rotated about ${}^T Z$ to $\overline{{}^T P_1^* {}^T P_2^*}$ which is parallel to ${}^T X {}^T Z$ plane in TCS as shown in Figure 3-22(b). The plane parallel to the ${}^T X {}^T Z$ plane and containing $\overline{{}^T P_1^* {}^T P_2^*}$ is π . C is the intersection curve of π and the cutter. To eliminate the gouging, $\overline{{}^T P_1^* {}^T P_2^*}$ can be lifted along vector $(0, 0, l)$ until this edge is tangent to C . The distance l from point $(0, 0, 0)$ to plane π is equal to ${}^T Y_{P_1^*}$.

The slope of this edge is m :

$$m = \frac{{}^T Z_2 - {}^T Z_1}{{}^T X_2 - {}^T X_1} \quad (3.63)$$

As shown in Figure 3-22(b), if l is less than R_1 , π slices the cone region, the torus region, and the taper region. Curve C in this case, as shown in Figure 3-23(a), is not continuous at T_2 and T_3 . If l is larger than R_1 and less than R_2 , π slices the torus region and the taper region. Curve C in this case, as shown in Figure 3-23(b), is not continuous at T_2 . If l is larger than R_2 , π slices only the taper region. Curve C in this case is shown in Figure 3-23(c).

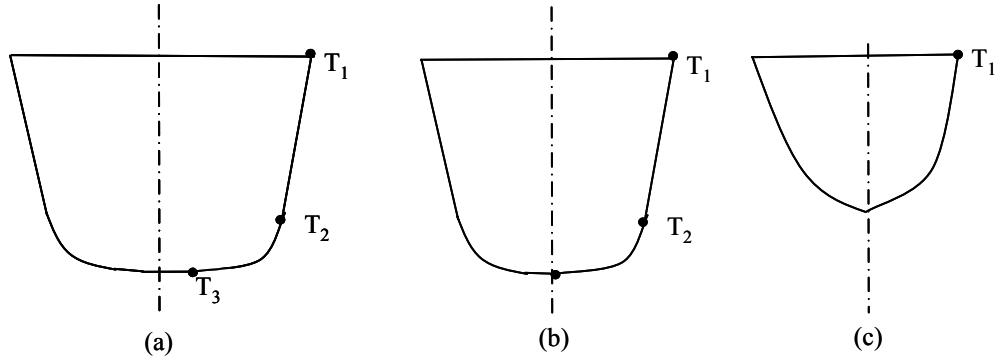


Figure 3-23 Different sections of a generalized cutter.

Curve C is a spline. Its slope is different at each spline section and is calculated as

follows:

$$m_{2^+} = \frac{L}{R - R_2} \frac{\sqrt{R_2^2 - l^2}}{R_2} \quad (3.64)$$

$$m_{2^-} = \sqrt{\frac{4e^2 + (1+l^2)(R_1^2 - l^2)^2 - 4e(R_1^2 - l^2)R_1}{R_1^2(4eR_1^2 - R_1^2 - e^2 + r^2)}} \quad (3.65)$$

$$m_{3^+} = \sqrt{\frac{4e^2 + (1+l^2)(R_2^2 - l^2)^2 - 4e(R_2^2 - l^2)R_2}{R_2^2(4eR_2^2 - R_2^2 - e^2 + r^2)}} \quad (3.66)$$

$$m_{3^-} = \frac{\sqrt{R_1^2 - l^2} \tan \beta_1}{R_1} \quad (3.67)$$

where m_{2^+} is the slope of the taper at point T_2 , m_{2^-} is the slope of the torus at point T_2 , m_{3^+} is the slope of the torus at point T_3 , and m_{3^-} is the slope of the torus at point T_3 .

In this instance, gauging is divided into 5 types according to the position where the triangle contacts with the cutter after the cutter is lifted to eliminate the gouging: the taper region ($m > m_{2^+}$), point T_2 ($m_{2^-} \leq m \leq m_{2^+}$), the torus region ($m_{3^+} < m < m_{2^-}$), point T_3 , ($m_{3^-} \leq m \leq m_{3^+}$), and the cone region ($m < m_{3^-}$).

- Taper region:

$${}^T X_Q = \text{sign}(m) \frac{Rml}{\sqrt{\left(\frac{RL}{R-R_2}\right)^2 - R^2 m^2}} \quad (3.68)$$

- Torus region:

$${}^T X_Q \cdot (e - \sqrt{{}^T X_Q^2 + l^2}) - \eta \cdot \sqrt{r^2 \cdot ({}^T X_Q^2 + l^2) - ({}^T X_Q^2 + l^2)} \cdot (\sqrt{{}^T X_Q^2 + l^2} - e) = 0 \quad (3.69)$$

- Cone region:

$${}^T X_Q = \frac{ml}{\sqrt{\tan^2 \beta_1 - m^2}} \quad (3.70)$$

- T_2

$${}^T X_Q = \text{sign}(m) \sqrt{R_2^2 - l^2} \quad (3.71)$$

- T_1

$${}^T X_Q = \text{sign}(m) \sqrt{R_1^2 - l^2} \quad (3.72)$$

After ${}^T X_Q$ is obtained, the lifting height is easily calculated.

3. Triangle Gouging

The triangle $\Delta P_1 P_2 P_3$ π is described as

$$a \cdot {}^T X + b \cdot {}^T Y + c \cdot {}^T Z + d = 0 \quad (3.73)$$

where the normal of π is normalized to

$$n = (a, b, c) \quad (3.74)$$

- Taper region:

The plane π can be lifted to contact the torus region at a point Q :

$$\begin{cases} {}^T X_Q = -\frac{R \cdot a}{\sqrt{a^2 + b^2}} \\ {}^T Y_Q = -\frac{R \cdot b}{\sqrt{a^2 + b^2}} \\ {}^T Z_Q = -\frac{a \cdot {}^T X_Q + b \cdot {}^T Y_Q + d}{\sqrt{a^2 + b^2}} \end{cases} \quad (3.75)$$

If both Equations (3.76) and (3.80) are satisfied, Q is a reasonable tangent point.

$$L_C - L < {}^T Z_Q \leq L_C \quad (3.76)$$

- Torus region:

The plane π can be lifted to contact the torus region at a point Q :

$$\begin{cases} {}^T X_Q = -\frac{l \cdot a}{\sqrt{a^2 + b^2}} \\ {}^T Y_Q = -\frac{l \cdot b}{\sqrt{a^2 + b^2}} \\ {}^T Z_Q = -\frac{a \cdot {}^T X_Q + b \cdot {}^T Y_Q + d}{\sqrt{a^2 + b^2}} \end{cases} \quad (3.77)$$

where

$$l = e + r \cdot \sqrt{1 - c^2} \quad (3.78)$$

- Cone region:

The cutter and the triangle only gouge each other at $Q(0, 0, 0)$ with the condition:

$$\|d\| < \sin \beta_1 \quad (3.79)$$

In the triangle gouging case, there is a point P at the plane π which has the same $({}^T X, {}^T Y)$ with Q .

$$P \in \Delta P_1 P_2 P_3 \quad (3.80)$$

In this case, the lifting height is

$$\Delta h = {}^T Z_P - {}^T Z_Q \quad (3.81)$$

If the lifting height is larger than 0, then this lifting height is used to eliminate the gouging. Otherwise, there is no gouging.

Using the results from the three gouging cases above, the cutter lifting height Δh can be obtained. The CL point is obtained if the cutter is lifted Δh along the ${}^G Z$ direction.

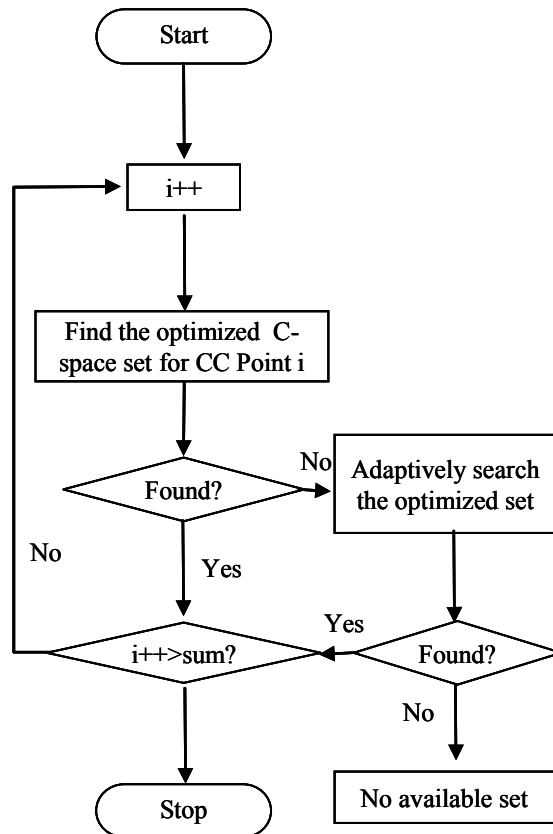


Figure 3-24 C-space construction.

3.5.5 Five-axis machining CL point generation

A 5-axis mill has three transaction motion axes: X , Y , Z , and two rotation axes: an inclination angle λ and a yaw angle ω . To eliminate gouging, three methods are

employed: lifting the cutter δ along Z_L , rotating the cutter about Y_L by the inclination angle λ , and rotating the cutter about Z_L by the yaw angle ω . CL points are then determined not only by CC points but also by λ , ω , and δ . The CL point $CLPt_{i,j}$ is determined by the set $(CCPt_{i,j}, \delta, \lambda, \omega)$. If the CC point $CCPt_{i,j}$ is fixed, the CL point $CLPt_{i,j}$ is only determined by the 3-dimension set $(\delta, \lambda, \omega)$. If the cutter posture as determined by the set $(\delta, \lambda, \omega)$ is gouge-free and the cusp height determined by this cutter posture is less than the machining tolerance, this set $(\delta, \lambda, \omega)$ is called an *available* 3D C-space set. The union of all *available* 3D C-space sets for a CC point $CCPt_{i,j}$ is used to build the 3D C-space $V_{i,j}$.

Figure 3-24 shows the general process to plan a CL tool path:

1. For each CC point, a corresponding C-space is first constructed.
2. The optimal C-space set is then determined within this constructed C-space set.
3. The CL point is generated based on the optimal C-space set.
4. If there isn't an optimal C-space set, this means that the CC point is inside a clean-up machining strip that will need subsequent clean-up operations.
5. Deal with all CC points using the operations above.
6. Connect all CL points to obtain the optimal tool path.

3.5.5.1 Construction of C-space

Figure 3-25 is a C-space sample for the gouging problem of a flat end cutter at a CC point. All C-space sets that are used to construct the *available* C-space satisfy two preconditions: the cutter posture, as determined by this set, is gouge-free; and the

remaining material height with this cutter posture is less than the machining tolerance. The top surface comprises the C-space sets with which the cusp height is equal to the machining tolerance. The bottom surface comprises the C-space sets with which the cutter path is gouge-free. The region between the top surface and the bottom surface is the 3D C-space for a CC point.

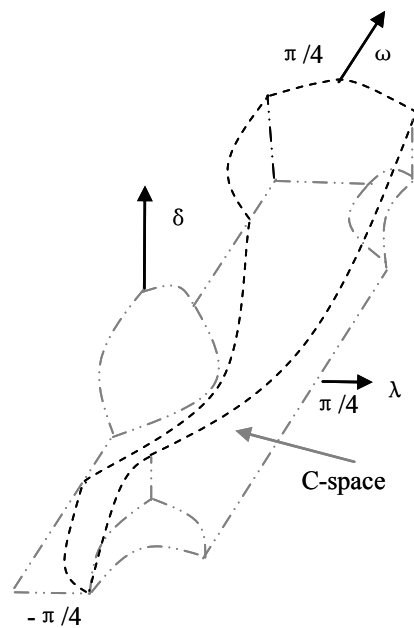


Figure 3-25 Sample C-space for a facet.

Figure 3-26 indicates the process to build the C-space for a CC point as follows:

1. Divide the gouging instances on a CC point P into 5 different cases. Calculate the bottom boundary surface S_2 of the C-space in each case, such as VTCBT, VTCBD, etc.
2. Intersect the bottom boundary surface S_2 with the results from each case to obtain the S_2 for a single facet on the CC point P .

3. Intersect the bottom boundary surface S_2 with all facets to obtain the C-space for CC point P .
4. Build the bottom boundary surface S_1 of the C-space for CC point P .

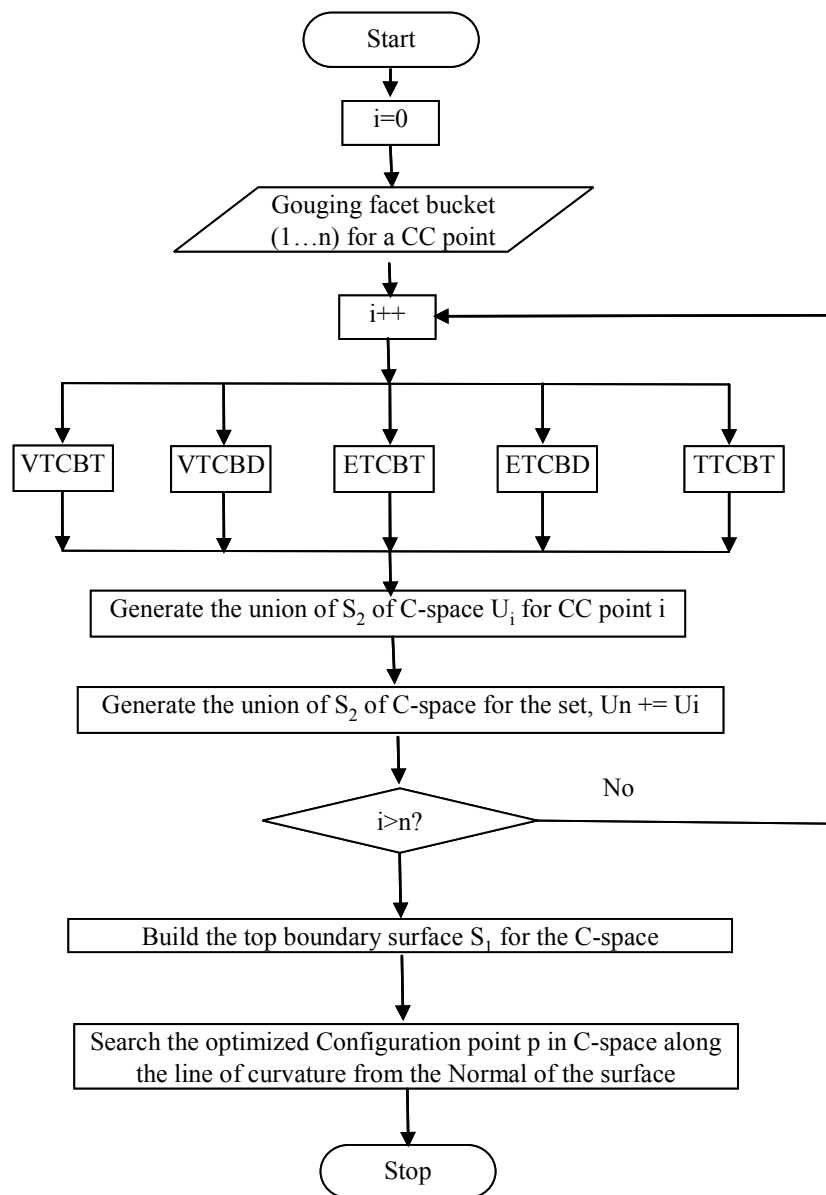


Figure 3-26 C-space construction processes for CC points

5. Build the C-space for point P .
6. Search the optimal configuration set within the region (C-space) between S_1 and S_2 .

The purpose of building the top boundary surface S_1 is to determine the boundary value (δ_1) of the cutter lifting height for each corresponding (λ, ω) set. The cusp height, determined by δ_1 , is equal to the machining tolerance τ as described in the following equations:

$$\begin{cases} {}^L d_{_step_{i,L}} = {}^L Y_{P_{i,j}(\alpha)} = {}^L Y_{P_{i,j}(d)} \\ \tau = {}^L Z_{P_{i,j}(\alpha)} - {}^L Z_{P_{i,j}(d)} \end{cases} \quad (3.82)$$

To obtain α and δ , substitute Equation (3.25) into Equation (3.9) and Equation (3.82), to derive

$$\alpha = \sin^{-1}\left(\frac{C}{\sqrt{A^2 + B^2}}\right) \quad (3.83)$$

and

$$\begin{aligned} B &= R \cdot \cos \omega \\ C &= d_{_step_{i,j}(\alpha)} - (\tau + {}^L Z_{P_{i,j}(d)} - R \cdot \sin \lambda) \cdot \sin \omega \cdot \tan \lambda + R \cdot \sin \omega \cdot \sin \lambda \\ \delta &= \frac{(\tau + {}^L Z_{P_{i,j}(d)} + \sin \lambda \cdot (-R + R \cdot \sin \alpha))}{\cos \lambda} \end{aligned} \quad (3.84)$$

Consequently, with a known (λ, ω) set, α and δ_1 can be obtained with Equations (3.83) and (3.84). The union of all $(\lambda, \omega, \delta_1)$ sets is S_1 .

From the analysis above, (λ, ω) sets are selected on a grid. The finer grids are, the higher quality of the surface can be obtained. However, the calculation time consumed will be longer. So, a reasonable grid density should be selected carefully. We select 3 degree according to our experiences.

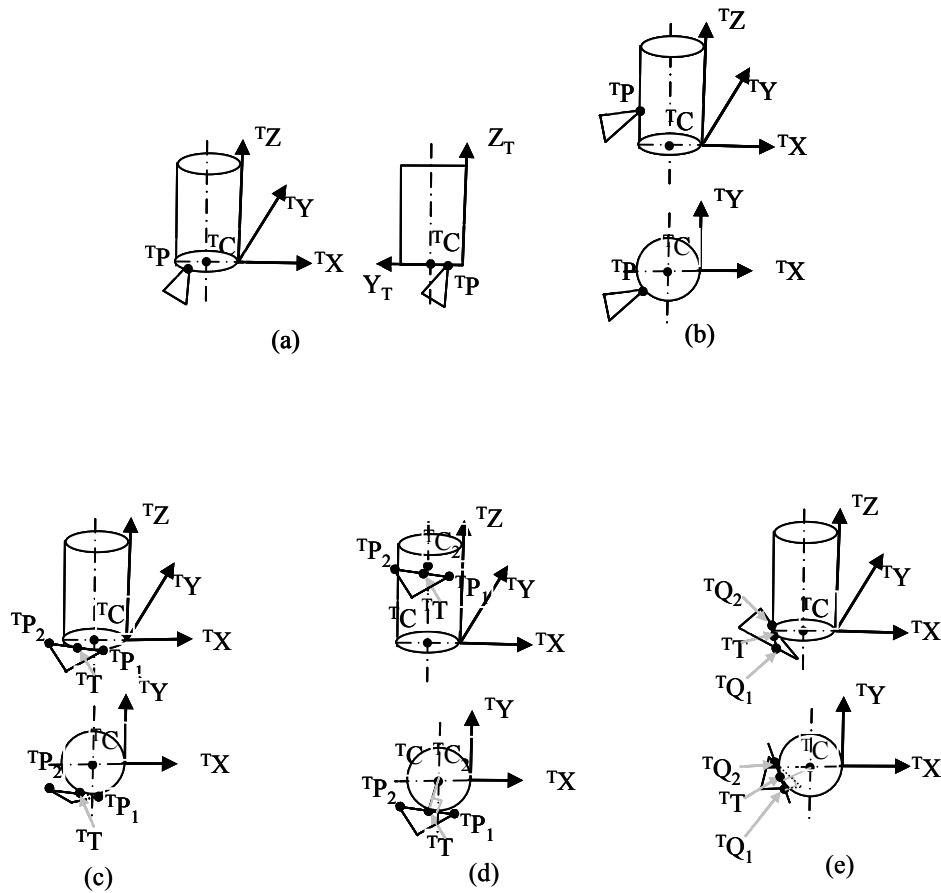


Figure 3-27 Gouging elimination methods.

To build the bottom boundary surface S_2 , different gouging cases needed be discussed separately (as follows).

1. Vertex Gouging

- The vertex touches the cutter bottom (VTCBT)

Referring to Figure 3-27(a), point ${}^T C$ is the cutter center point. Point ${}^T P$ is with respect to frame T. To calculate the C-space requires the calculation of ω with a given λ .

If ${}^T P$ contacts the cutter bottom, then:

$${}^T Z_p = 0 \quad (3.85)$$

and

$$\|{}^T P T_C\| = \sqrt{{}^T X_p^2 + {}^T Y_p^2} \leq R \quad (3.86)$$

Then, substitute Equation (3.8) into Equation (3.85) to get

$$\begin{aligned} \sin \lambda \cdot \cos \omega \cdot {}^L X_p + \sin \lambda \cdot \sin \omega \cdot {}^L Y_p + \cos \lambda \cdot {}^L Z_p - \delta &= 0 \\ \Rightarrow \delta &= \sin \lambda \cdot \cos \omega \cdot {}^L X_p + \sin \lambda \cdot \sin \omega \cdot {}^L Y_p + \cos \lambda \cdot {}^L Z_p \end{aligned} \quad (3.87)$$

Substitute Equation (3.8) into Equation (3.86), and an implicit equation of the *available* (λ, ω) region is obtained:

$$(\cos \lambda \cdot \cos \omega \cdot {}^L X_p + \cos \lambda \cdot \sin \omega \cdot {}^L Y_p - \sin \lambda \cdot {}^L Z_p)^2 + (-\sin \omega \cdot {}^L X_p + \cos \omega \cdot {}^L Y_p)^2 < R^2 \quad (3.88)$$

Given each (λ, ω) set in the *available* (λ, ω) region, if the corresponding δ_2 obtained by Equation (3.87) is less than the δ_1 obtained by Equation (3.84), then $(\lambda, \omega, \delta_2)$ is an *available* 3D C-space set for this vertex. The union of all *available* 3D C-space sets is the 3D C-space for VTCBT.

- The vertex touches the cutter body (VTCBD).

Referring to Figure 3-27(b), ${}^T P$ is the contact point with the cutter body, where

$${}^T Z_p \leq [0, H] \quad (3.89)$$

and

$$\|{}^T P T_C\| = R \quad (3.90)$$

Then

$$\begin{aligned} \sqrt{({}^T X_p + R)^2 + {}^T Y_p^2} &= R \\ \Rightarrow (\cos \lambda \cdot \cos \omega \cdot {}^L X_p + \cos \lambda \cdot \sin \omega \cdot {}^L Y_p - \sin \lambda \cdot {}^L Z_p + R)^2 + (-\sin \omega \cdot {}^L X_p + \cos \omega \cdot {}^L Y_p)^2 &= R^2 \\ \Rightarrow \omega &= \theta - \arccos\left(\frac{-B \pm \sqrt{B^2 - 4 \cdot A \cdot C}}{2 \cdot A}\right) \end{aligned} \quad (3.91)$$

where

$$A = -{}^T\rho^2 \cdot \sin^2 \lambda \quad (3.92)$$

$$B = 2 \cdot \rho \cdot \cos \lambda \cdot (-\sin \lambda \cdot {}^LZ_p + R) \quad (3.93)$$

$$C = \sin^2 \lambda \cdot {}^LZ_p^2 + {}^T\rho^2 - 2 \cdot R \cdot \sin \lambda \cdot {}^LZ_p \quad (3.94)$$

$$\sin \theta = {}^LY_p / \rho, \tan \theta = {}^LY_p / {}^LX_p \quad (3.95)$$

with the precondition that

$${}^TZ_p \leq [0, H] \quad (3.96)$$

The union of all *available* 3D C-space sets $(\lambda, \omega, 0)$ (obtained from Equation (3.91)) that satisfy Equation (3.96) is the 3D C-space for VTCBD. In this case, it is a 3D C-space shell, not a C-space volume.

2. Edge Gouging

- The edge touches the cutter bottom (ETCBT)

Referring to Figure 3-27(c), TT is the intersection point of an edge ${}^TP_1{}^TP_2$ and the cutter bottom plane, as well as a point on the contour of the cutter bottom.

Since

$${}^TZ_T = 0 \quad (3.97)$$

and

$$\|{}^TZ_T T_C\| = R \quad (3.98)$$

then,

$${}^TZ_{P_2} + ({}^TZ_{P_1} - {}^TZ_{P_2}) \cdot t = 0 \quad (3.99)$$

and

$$[{}^TX_{P_2} + ({}^TX_{P_1} - {}^TX_{P_2}) \cdot t + R]^2 + [{}^TY_{P_2} + ({}^TY_{P_1} - {}^TY_{P_2}) \cdot t]^2 = R^2 \quad (3.100)$$

Solve for t in Equation (3.99) and substitute it into Equation (3.100).

$$[{}^T X_{P_2} + ({}^T X_{P_1} - {}^T X_{P_2}) \cdot \frac{{}^T Z_{P_2}}{{}^T Z_{P_2} - {}^T Z_{P_1}} + R]^2 + [{}^T Y_{P_2} + ({}^T Y_{P_1} - {}^T Y_{P_2}) \cdot \frac{{}^T Z_{P_2}}{{}^T Z_{P_2} - {}^T Z_{P_1}}]^2 = R^2 \quad (3.101)$$

An implicit equation is then obtained by substituting Equation (3.101) into Equation (3.8).

$$\delta = \frac{-B \pm \sqrt{B^2 - 4 \cdot A \cdot C}}{2 \cdot A} \quad (3.102)$$

where

$$\left\{ \begin{array}{l} A = k_1^2 + k_2^2 \\ B = D - 2AF \\ C = E + A(F^2 - D) \\ D = 2({}^T X_{P_2} + R)k_1 + k_2 {}^T Y_{P_2} \\ E = {}^T X_{P_2}^2 + 2{}^T X_{P_2} R + {}^T Y_{P_2}^2 \\ F = \sin \lambda \cdot \cos \omega \cdot {}^L X_p + \sin \lambda \cdot \sin \omega \cdot {}^L Y_p + \cos \lambda \cdot {}^L Z_p \\ k_1 = \frac{{}^T X_{P_1} - {}^T X_{P_2}}{{}^T Z_{P_1} - {}^T Z_{P_2}} \\ k_2 = \frac{{}^T Y_{P_1} - {}^T Y_{P_2}}{{}^T Z_{P_2} - {}^T Z_{P_1}} \end{array} \right. \quad (3.103)$$

Again, given each (λ, ω) set, if the corresponding δ_2 obtained by Equation (3.103) is less than the δ_1 obtained by Equation (3.84), then $(\lambda, \omega, \delta_2)$ is an *available* 3D C-space set for this vertex. The union of all *available* 3D C-space sets is the 3D C-space for ETCBT.

- The edge touches the cutter body (ETCDB).

Referring to Figure 3-27(d), ${}^T T$ is the tangent point of ${}^T P_1 {}^T P_2$ and the cutter body. ${}^T C_2$ has the same Z coordinate as ${}^T T$, and is on the axis of cutter.

Since

$$\overrightarrow{{}^T P_1 {}^T P_2} \times \overrightarrow{{}^T T {}^T C_2} = 0 \quad (3.104)$$

and

$$\|{}^T Z_T T_C\| = R \quad (3.105)$$

then substitute Equation (3.8) into Equation (3.104) and Equation (3.105) to derive

$$\frac{{}^T Y_T - 0}{{}^T X_T + R} \cdot \frac{{}^T Y_{P_2} - {}^T Y_{P_1}}{{}^T X_{P_2} + {}^T X_{P_1}} = -1 \quad (3.106)$$

and

$$\sqrt{({}^T X_T + R)^2 + {}^T Y_T^2} = R \quad (3.107)$$

Then solve Equation (3.106) for t :

$$t = \frac{{}^T Y_{P_1} \cdot ({}^T Y_{P_2} - {}^T Y_{P_1}) + ({}^T X_{P_1} + R) \cdot ({}^T X_{P_2} - {}^T X_{P_1})}{({}^T X_{P_2} - {}^T X_{P_1})^2 + ({}^T Y_{P_2} - {}^T Y_{P_1})^2} \quad (3.108)$$

And change Equation (3.107) to:

$$[{}^T X_1 + t({}^T X_2 - {}^T X_1) + R]^2 + [{}^T Y_1 + t({}^T Y_2 - {}^T Y_1)]^2 = R^2 \quad (3.109)$$

An implicit equation about λ and ω is then obtained by substituting Equation (3.109)

into Equation (3.108) and Equation (3.8). The union of all *available* 3D C-space sets $(\lambda, \omega, 0)$ obtained from Equation (3.109) that satisfy Equation (3.96) is the 3D C-space for ETCBD. In this case, it is a 3D C-space shell, not a C-space volume.

3. Triangle Gouging (TTCBT)

Referring to Figure 3-27(e), ${}^T Q_1 {}^T Q_2$ is an intersection line in the triangle and the ${}^T X^T Y$ plane. ${}^T T$ is the tangent point of ${}^T Q_1 {}^T Q_2$ and the contour of the cutter bottom.

Since

$$\|{}^T Q_1 {}^T Q_2 \cdot {}^T T T_C\| = 0 \quad (3.110)$$

$$\|{}^T T T_C\| = R \quad (3.111)$$

Then, the normal of plane ${}^T P_1 {}^T P_2 {}^T P_3$ (the facet plane) is

$$N = \overrightarrow{{}^T P_1 {}^T P_2} \times \overrightarrow{{}^T P_1 {}^T P_3} = \begin{vmatrix} i & j & k \\ {}^T X_{P_2} - {}^T X_{P_1} & {}^T Y_{P_2} - {}^T Y_{P_1} & {}^T Z_{P_2} - {}^T Z_{P_1} \\ {}^T X_{P_3} - {}^T X_{P_1} & {}^T Y_{P_3} - {}^T Y_{P_1} & {}^T Z_{P_3} - {}^T Z_{P_1} \end{vmatrix} \quad (3.112)$$

and the normal of plane ${}^T P_1 {}^T P_2 {}^T P_3$ is

$$\begin{aligned} \overline{N_x} \cdot \overline{P^T P_1} &= 0 \\ \Rightarrow \overline{N_x} \cdot (X - X_{P1}) + \overline{N_y} \cdot (Y - Y_{P1}) + \overline{N_z} \cdot (Z - Z_{P1}) &= 0 \end{aligned} \quad (3.113)$$

Then, line ${}^T Q_1 {}^T Q_2$ is:

$$N_x \cdot (X - X_{P1}) + N_y \cdot (Y - Y_{P1}) - N_z \cdot Z_{P1} = 0 \quad (3.114)$$

and line ${}^T T T_C$ is

$$\frac{{}^T Y - 0}{{}^T Y + R} \cdot \left(-\frac{N_x}{N_y}\right) = -1 \quad (3.115)$$

Substitute Equation (3.114) and Equation (3.115) into Equation (3.110):

$$\begin{cases} {}^T X_T = \frac{N_x \cdot X_{R1} + N_y \cdot Y_{R1} + N_z \cdot Z_{R1}}{N_x \cdot N_x + N_y \cdot N_y} \\ {}^T Y_T = \frac{N_x}{N_y} \cdot ({}^T X_T + R) \end{cases} \quad (3.116)$$

Then substitute Equation (3.116) into Equation (3.111):

$$d = \frac{N_x \cdot (-R - X_{R1}) + N_y \cdot (0 - Y_{R1}) - N_z \cdot Z_{R1}}{\sqrt{N_x^2 + N_y^2}} = R \quad (3.117)$$

The implicit equation is then obtained by substituting Equation (3.117) with Equation (3.8).

$$\delta = F - \frac{R \sqrt{N_x^2 + N_y^2} + N_x \cdot (R + X_{R1}) + N_y \cdot Y_{R1}}{N_z} \quad (3.118)$$

Again, given each (λ, ω) set, if the corresponding δ_2 obtained by Equation (3.103) is less than the δ_1 obtained by Equation (3.84) and point ${}^T T$ is inside the given triangle, $(\lambda, \omega, \delta_2)$ is an *available* 3D C-space set for this vertex. The union of all *available* 3D C-space sets is the 3D C-space for ETCBT.

3.5.5.2 CL Tool Path Generation

Using the method just discussed, a 3D C-space $V_{i,j}$ for each corresponding CC point $CCPt_{i,j}$ is built. The generation of the CL tool path $CLTP_i$ in this dissertation is divided into three steps: (1) Searching the optimal C-space set $OCS_{i,j}$ with a search policy (i.e. an optimization method) for each CC point; (2) Determining the CL point $CLPt_{i,j}$ with this optimal C-space set $OCS_{i,j}$; (3) Connecting the CL points to generate the local optimal CL tool path $CLTP_i$. The first step plays the key role in this process and is closely related to mill machine kinematics.

The optimization method used in this dissertation can be divided into two types according to the kinematics involved: the global optimization process, and the local optimization process. In the global optimization process, the process for $CCPt_{i,j}$ is affected by the optimal C-space sets of other CC points (e.g. $CCPt_{i,j-1}$, $CCPt_{i,j-2}$...), while in the local optimization process, the process for each CC point $CCPt_{i,j}$ is independent of any other point.

The CL tool path generated by the global optimization process is called the global CL tool path. The global optimization process involves a detailed kinematics analysis, including the generation of the manipulator trajectory, the position control of the manipulators, the force control of the manipulators, etc. All CL points are optimal concurrently. Planning such ideal global optimal tool path is currently too complex to implement.

In this dissertation, the global optimal tool path is simplified to the local optimal tool path. The optimization process of each CL point is independent. The speed of all joints of the mill machine is assumed to be constant (uniform motion). No acceleration, machine jerk, or force analysis is considered in this dissertation.

Every 5-axis machine has different kinematics features. The so-called local optimal tool path is only optimal for one type of 5-axis machine. Consequently, an ideal CAM system generates different CL tool paths for different mill machines. The 5-axis Tarus Mill Machine (Model: TPI-2000) is used in this dissertation. The tilt head (C-axis) is on the top of the rotation head (A-axis). The five joints move independent of each other, i.e. the five joints move concurrently. The speed of each joint is as follows: the X joint: 170 mm/s, the Y joint: 180 mm/s, the Z joint: 240 mm/s, the C joint: 0.3 rad/s, and the A joint: 0.5 rad/s.

Searching the optimal C-space set $OCS_{i,j}$ for each CC point in the local optimization process can be divided into three steps: (1) Determine the ideal C-space set $ICS_{i,j}$; (2) Determine the initial C-space set; (3) Search for the C-space set $OCS_{i,j}$ closest to the ideal one $ICS_{i,j}$ in the 3D C-space. The machining process that is determined by the ideal C-space set has the best performance. However, the ideal C-space set may not be within the constructed 3D C-space. The final optimal C-space set will be the *available* C-space set *closest* to the ideal C-space set. *Closest* indicates the minimum cutter motion time between two C-space sets, which will be explained later.

- **The ideal C-space set**

As shown in Figure 3-28, $CCPt_{i,j-1}$, $CCPt_{i,j}$, and $CCPt_{i,j+1}$ are three adjacent CC points on the same tool path. $CHPt_{i,j-1}$, $CHPt_{i,j}$, and $CHPt_{i,j+1}$ are center points of the cutter holder (*CH* points). The connection of *CH* points is the translation trajectory of the mill machine. As shown in Figure 3-28, the length of the line connecting $CCPt_{i,j}$ and $CHPt_{i,j}$ is a constant (L). $CHPt_{i,j}$ is obtained with the known location of $CHPt_{i,j}$ as shown in Equation (3.119).

$$CHPt_{i,j} = CCPt_{i,j} + H_{i,j} \quad (3.119)$$

$$\|H_{i,j}\| = L \quad (3.120)$$

where $H_{i,j}$ is the vector connecting $CCPt_{i,j}$ and $CHPt_{i,j}$.

The length of the cutter holder trajectory is the shortest (the same as the length of the CC tool path) if $H_{i,j}$ is along a fixed direction as shown in Figure 3-28(c). The CH point is offset along the normal of the CC point in Figure 3-28(b). It is obvious that the cutter holder trajectory in Figure 3-28(b) is longer than that in Figure 3-28(c). Therefore, to obtain the ideal C-space set in this dissertation, $V_{i,j}$ is set to a constant. We set it to $(0, 0, 1)$, the same as Z direction as Equation (3.120).

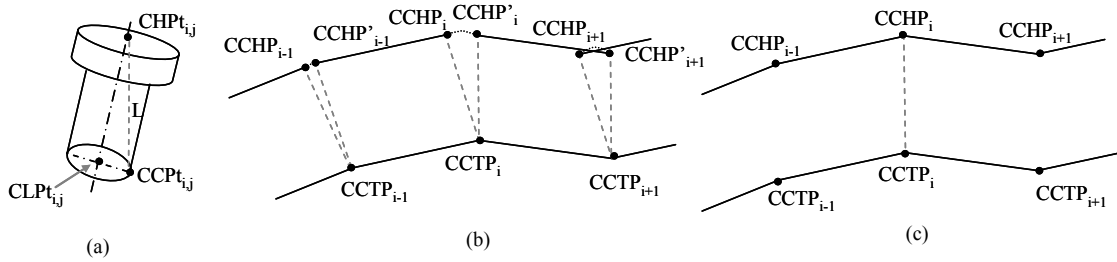


Figure 3-28 The trajectory of the cutter holder.

- **The initial C-space Set**

The initial C-space set is the set where the optimization process starts. The selection of the initial C-space set affects the optimization process performance. To promote the optimization process speed, the initial C-space set should be set as close to the final optimal set as possible.

The cutter posture is determined by the ideal C-space set as shown in Figure 3-29. As a result, the cusp height is normally much larger than the machining tolerance if the

cutter motion is upward (Figure 3-29(a)), and the cutter seriously gouges the design surface if the cutter motion is downward (Figure 3-29(b)). According to the benchmark result, most of the final optimal C-space sets are close to the normal of the design surface. Consequently, in this dissertation the C-space set that determined by the CM^2 method is applied to be the initial C-space set because the CM^2 method positions the cutter based on the surface normal and the local curvature. The final optimal set will be searched with a special search policy based on this initial set. With this idea, the 3D C-space method in this dissertation can also be called an advanced CM^2 method.

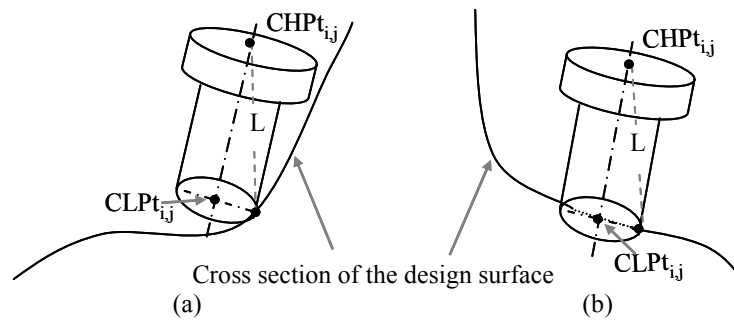


Figure 3-29 The cutter posture determined by the ideal C-space set.

- **Optimization process**

As discussed above, the term *close* for two C-space sets indicates the motion time distance between a C-space set and the ideal C-space set. For example, $(\delta, \lambda, \omega)$ and $(\delta^*, \lambda^*, \omega^*)$ are two C-space sets for $CCPt_{i,j}$. Point $CLPt_{i,j}$ and point $CLPt_{i,j}^*$ are two CL points determined by these two C-space sets. CH and CH^* are corresponding cutter holder points. The motion time between these two cutter postures are calculated in Equation (3.123).

$$\begin{cases} d_x \\ d_y \\ d_z \end{cases} = \begin{cases} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{cases} \cdot \overline{CHCH^*} \quad (3.121)$$

$$t_x = d_x/v_x, t_y = d_y/v_y, t_z = d_z/v_z$$

$$t_\lambda = |\lambda - \lambda^*|/\omega_C \quad (3.122)$$

$$t_\omega = |\omega - \omega^*|/\omega_A$$

$$t = \max(t_x, t_y, t_z, t_\lambda, t_\omega) \quad (3.123)$$

where $v_x, v_y,$ and v_z respectively designate the linear velocity of prismatic joints $x, y, z,$ and the velocity ω_C and ω_A designate the angular velocity of rotational joints C and $A.$

If $(\delta^*, \lambda^*, \omega^*)$ is the ideal C-space set, and $(\delta, \lambda, \omega)$ is an auxiliary C-space set, the objective function is the motion time distance between these two C-space sets as calculated in Equation (3.123). To maintain efficiency, the search policy is to minimize this objective function.

The boundary of the 3D C-Space determines the boundary functions of the optimization process as shown in Equation (3.124).

$$\begin{cases} \delta_1(\lambda, \omega) \leq \delta \leq \delta_2(\lambda, \omega) \\ 0 \leq \lambda \leq \frac{\pi}{4} \\ -\frac{\pi}{4} \leq \omega \leq \frac{\pi}{4} \end{cases} \quad (3.124)$$

Various optimization approaches can be applied to the tool path generation. The penalty function method [16] is adopted in this dissertation because of its robustness and efficiency. The penalty coefficient r is set at 10000 to guarantee the convergence speed.

3.6 Clean-up machining

Some uncut areas may be left along concave regions after the finish machining stage. Because clean-up machining uses a smaller cutter than that used in finish machining, the

process needs to be supplemented if the heights of the uncut areas are larger than the finish machining tolerance. Hereafter, this type of uncut material is called *scallop or cusp*. The strip containing *over-high material* is the clean-up strip (Figure 3-30(b)).

Clean-up machining is processed in two steps: (1) Clean-up strip recognition (Figure 3-30(b)), and (2) CL point generation. CC points are not planned because the CL point generation method here is a curve-based method.

In the first step, the clean-up strip is recognized. In most of past literatures, the strip is recognized according to the curvature change. This method will also be applied in this dissertation. In some cases, the strip may be in another shape (like a circle, a polygon, etc.) instead of a strip. The same recognition method can still be applied.

After the clean-up strip is determined, CC points can be deployed on it. Since the shape of the clean-up machining strip is different with the design part, CC points planning methods are also different.

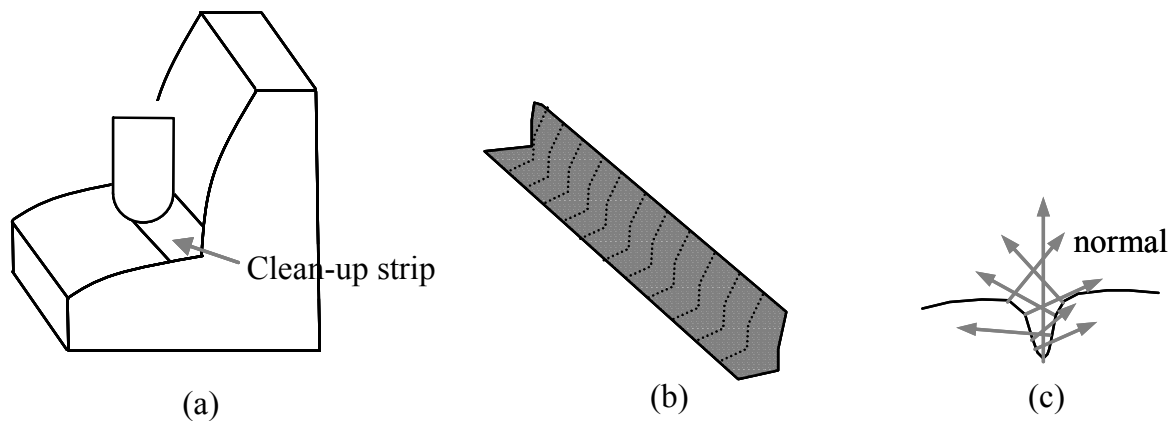


Figure 3-30 Clean-up strip.

3.6.1 Clean-up strip recognition

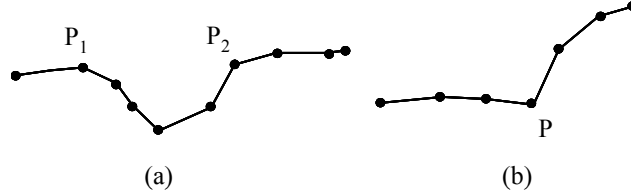


Figure 3-31 Enlarged image of cross section of the clean-up strip.

The clean-up strip is the region that needs to be machined in the clean-up machining stage. It is challenging to recognize the clean-up strip in tessellated models because tessellated models lack geometry features. The clean-up strip is divided into two types based roughly according to its *machining features*. As shown in Figure 3-31, the cross section of the clean-up shape is composed of CC points. CC point normals change sharply at the various CC points in type (a). In Figure 3-31(a), point P_1 is the first CC point where the normal changes sharply, and P_2 is the final one. The region between P_1 and P_2 is the clean-up strip. In type (b), the CC point normals change sharply only at one particular CC point: P . The clean-up strip in type (b) cannot be determined until the CL points are planned.

To determine whether the normal at a CC point changes sharply, a threshold δ_{max} is set. If δ_N , given by Equation (3.125), is greater than δ_{max} , then one point is found. Threshold δ_{max} is set according to the cutter size, the shape of the cutter, and the local geometry features of the design surface. In this dissertation, we set it manually.

$$\delta N = (N_{i+1} - N_i) / d_{i,i+1} \quad (3.125)$$

where N_i is the normal of CC point i , and $d_{i,i+1}$ is the distance between two consecutive CC points along a single tool path.

3.6.2 CL points generation

The process of planning the CL points in clean-up machining is divided into two steps: (1) Position the cutter to touch the CC points; (2) Determine the optimal CL points. In step 1, the cutter is positioned so to be gouge-free on the CC tool path. The generated cutter positions in step 1 lead the cutter to gouge as few triangles as possible, which contributes to the decrease in the number of computations in step 2. The optimal CL points are determined in section 3.6.2.2.

3.6.2.1 Position the cutter to touch CC points

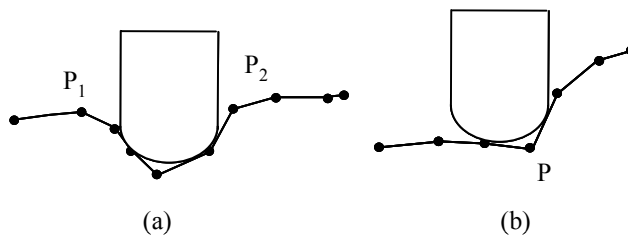


Figure 3-32 Position the cutter to CC points.

The graphics shown in Figure 3-32 are two examples of a clean-up strip. The cutter is positioned to touch the CC tool path with a curve-based method. There has been a great deal of research done about this [24][26], however this research will not be discussed.

3.6.2.2 Plan optimal CL points

After the cutter positions are roughly planned using the method in 3.6.2.1, a more accurate method should be employed to prevent possible gouging problems around the planned cutter position. To plan CL points precisely and optimally, the cutter is usually lifted along the normal of the CC point to eliminate gouging as shown in Figure 3-33. N_1 and N_2 are two offset vectors. N is the normal vector of the CL point. (For detailed information on the algorithms used to determine N , refer to [24][26].) However, for this dissertation, the gouging elimination approach is to translate triangles along the opposite direction of N to eliminate gouging while the position and the posture of the cutter are fixed. All the following calculations use this opposite direction.

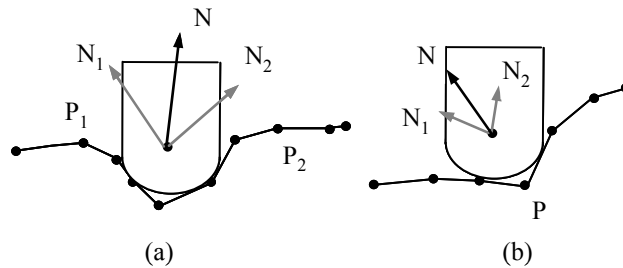


Figure 3-33 Lift the cutter to eliminate gouging.

At a CC point, the translation vector is assumed to be:

$$\vec{V} = \begin{Bmatrix} X_{\vec{V}} \\ Y_{\vec{V}} \\ Z_{\vec{V}} \end{Bmatrix} \quad (3.126)$$

To eliminate the gouging, a point of a triangle can be translated along this vector:

$${}^T P^* = {}^T P + t \cdot \vec{V} = \begin{cases} {}^T X + \mu \cdot X_{\vec{v}} \\ {}^T Y + \mu \cdot Y_{\vec{v}} \\ {}^T Z + \mu \cdot Z_{\vec{v}} \end{cases} \quad (3.127)$$

The gouging elimination problems can still be divided into the three cases as discussed in the sections above: vertex gouging, edge gouging, and triangle gouging.

1. Vertex gouging

Methods related to the vertex gouging recognition are the same as with those in section 3.5.4. If the vertex is gouging with the cutter, the vertex is translated along the vector until it contacts the cutter at point Q . Vertex gouging situations are categorized into three areas according to the position of point Q : the taper region, the torus region, and the cone region.

- Taper region

Assume point Q is in the taper region of the cutter. Then substitute Equation (3.127) into Equation (3.43) to obtain Equation (3.128), and calculate μ .

$$({}^T X + \mu \cdot X_{\vec{v}})^2 + ({}^T Y + \mu \cdot Y_{\vec{v}})^2 = \left(\frac{R - R_2}{L}\right)^2 ({}^T Z + \mu \cdot Z_{\vec{v}} - {}^T Z_0)^2 \quad (3.128)$$

This equation can be simplified:

$$A\mu^2 + B\mu + C = 0 \quad (3.129)$$

where

$$A = X_{\vec{v}}^2 + Y_{\vec{v}}^2 - \left(\frac{R - R_2}{L}\right)^2 Z_{\vec{v}}^2 \quad (3.130)$$

$$B = 2X_{\vec{v}} {}^T X + 2Y_{\vec{v}} {}^T Y + \left(\frac{R - R_2}{L}\right)^2 2Z_{\vec{v}} ({}^T Z - {}^T Z_0) \quad (3.131)$$

$$C = {}^T X^2 + {}^T Y^2 - \left(\frac{R - R_2}{L}\right)^2 ({}^T Z - {}^T Z_0)^2 \quad (3.132)$$

After the two roots (μ_1 and μ_2) in Equation (3.129) are obtained, then:

$$\mu = \min(\max(0, \mu_1), \max(\mu_2, 0)) \quad (3.133)$$

Substitute μ into Equation (3.127) to get ${}^T Z^*$. If Equation (3.134) is satisfied, then Q is in the taper region and the vertex gouging elimination is done. Otherwise, taper region gouging does not occur and next case (the torus region) should be tested.

$$Lc - L \leq {}^T Z^* \leq Lc \quad (3.134)$$

- Torus region

Assume point Q is in the torus region of the cutter. Substitute Equation (3.127) into Equation (3.47) to obtain Equation (3.135), and calculate μ :

$$\begin{aligned} & [({}^T X + \mu \cdot X_{\bar{v}})^2 + ({}^T Y + \mu \cdot Y_{\bar{v}})^2 + ({}^T Z + \mu \cdot Z_{\bar{v}})^2 + e^2 - r^2]^2 \\ & - 4e^2 [({}^T X + \mu \cdot X_{\bar{v}})^2 + ({}^T Y + \mu \cdot Y_{\bar{v}})^2] = 0 \end{aligned} \quad (3.135)$$

After the two roots (μ_1 and μ_2) in Equation (3.135) are obtained, then

$$\mu = \min(\max(0, \mu_1), \max(\mu_2, 0)) \quad (3.136)$$

Substitute μ to Equation (3.127) to get ${}^T Z^*$.

If Equation (3.137) is satisfied, Q is in the torus region and the vertex gouging elimination is done. Otherwise, torus region gouging does not occur and next area (the cone region) should be tested.

$$R_1 \tan \beta_1 \leq {}^T Z^* \leq Lc - L \quad (3.137)$$

- Cone region

Assume point Q is in the cone region of the cutter. Then substitute Equation (3.127) into Equation (3.50) to obtain Equation (3.138) and calculate μ :

$$({}^T X + \mu \cdot X_{\bar{v}})^2 + ({}^T Y + \mu \cdot Y_{\bar{v}})^2 = c \tan \beta_1^2 ({}^T Z + \mu \cdot Z_{\bar{v}})^2 \quad (3.138)$$

This equation can be simplified to

$$A\mu^2 + B\mu + C = 0 \quad (3.139)$$

$$A = X_{\bar{v}}^2 + Y_{\bar{v}}^2 - c \tan \beta_1^2 Z_{\bar{v}}^2 \quad (3.140)$$

$$B = 2X_{\bar{v}} {}^T X + 2Y_{\bar{v}} {}^T Y - 2c \tan \beta_1^2 Z_{\bar{v}} {}^T Z \quad (3.141)$$

$$C = {}^T X^2 + {}^T Y^2 - {}^T Z^2 c \tan \beta_1^2 \quad (3.142)$$

After the two roots (μ_1 and μ_2) in Equation (3.138) are obtained, then,

$$\mu = \min(\max(0, \mu_1), \max(\mu_2, 0)) \quad (3.143)$$

Substitute μ into Equation (3.127) to get ${}^T Z^*$.

If Equation (3.144) is satisfied, then Q is in the cone region. Otherwise, this case does not occur and next case will be tested.

$$0 \leq {}^T Z^* \leq R_1 \tan \beta_1 \quad (3.144)$$

2. Edge gouging

Given an edge, this edge can be translated along a vector until it contacts the cutter at point Q to eliminate the gouging, similar to the method used for vertex gouging. The gouging elimination problem is to find μ .

Again, according to the position of Q , the gouging cases can be divided into three areas: the taper region, the torus region, and the cone region. The algorithms used to calculate μ are discussed in the following sections.

- Taper region

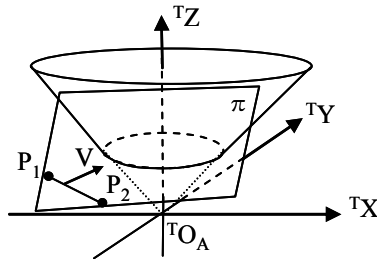


Figure 3-34 Edge contacts the taper of the cutter.

As shown in Figure 3-34, P_1P_2 is an edge of a triangle. V is the vector. A plane π is built with P_1P_2 and V . The intersection curve of the taper and the plane is C .

There are three contacting cases for the taper and the edge: (1) P_1 contacts to the taper; (2) P_2 contacts to the taper; (3) The edge contacts to the taper. Case (1) and (2) normally do not exist in the real machining process. Case (3) is discussed as follows.

The taper can be represented with Equation (3.43).

The plane π is represented with Equation (3.145).

$$A^T X + B^T Y + C^T Z + D = 0 \quad (3.145)$$

where

$$\begin{cases} A = {}^T X / \sqrt{{}^T X_n^2 + {}^T Y_n^2} \\ B = {}^T Y / \sqrt{{}^T X_n^2 + {}^T Y_n^2} \\ C = {}^T Z_n / \sqrt{{}^T X_n^2 + {}^T Y_n^2} \\ D = -({}^T X_1 {}^T X_n + {}^T Y_1 {}^T Y_n + {}^T Z_1 {}^T Z_n) / \sqrt{{}^T X_n^2 + {}^T Y_n^2} \end{cases} \quad (3.146)$$

$$\begin{cases} {}^T X_n = ({}^T Y_2 - {}^T Y_1) {}^T Z_V - ({}^T Z_2 - {}^T Z_1) {}^T Y_V \\ {}^T Y_n = -({}^T X_2 - {}^T X_1) {}^T Z_V + ({}^T Z_2 - {}^T Z_1) {}^T X_V \\ {}^T Z_n = ({}^T X_2 - {}^T X_1) {}^T Y_V - ({}^T Y_2 - {}^T Y_1) {}^T X_V \end{cases} \quad (3.147)$$

$({}^T X_n, {}^T Y_n, {}^T Z_n)$ is the normal vector of plane π .

The curve C is represented with Equation (3.148).

$$({}^T X^2 + {}^T Y^2) = \tan^2 \beta_2 \left(-\frac{A^T X + B^T Y + D}{C} - {}^T Z_0 \right)^2 \quad (3.148)$$

$P_1 P_2$ can be offset along the V direction until it contacts C at point Q . The offset line is given by Equation (3.149).

$$\overline{P_1^* P_2^*} = \overline{P_1 P_2} + \mu V \quad (3.149)$$

The equation of $\overline{P_1^* P_2^*}$ is represented with Equation (3.150) if $\overline{P_1^* P_2^*}$ is projected to ${}^T X^T Y$ plane.

$${}^T Y = k {}^T X + b \quad (3.150)$$

where

$$k = \frac{{}^T Y_2 - {}^T Y_1}{{}^T X_2 - {}^T X_1} \quad (3.151)$$

and

$$b = {}^T Y_1 + \mu {}^T Y_V - k({}^T X_1 + \mu {}^T X_V) \quad (3.152)$$

Substitute ${}^T Y$ in Equation (3.148) with Equation (3.149) to get Equation (3.153). This equation is then used to get the corresponding b , which is subsequently used to obtain Q .

$$\begin{aligned} & [k^2 + 1 - (\tan \beta_2 \frac{A+Bk}{C})^2] {}^T X^2 + [2kb - 2\frac{bk}{C} \tan \beta_2^2 (\frac{bB+A+D}{C} + z_0)] {}^T X \\ & + [b^2 - \tan \beta_2^2 (\frac{bB+A+D}{C} + z_0)^2] = 0 \end{aligned} \quad (3.153)$$

There is only one root in this 2 degree equation because there is only one tangent point Q between C and $\overline{P_1^* P_2^*}$. We can then get an equation describing b (Equation (3.154)).

$$A_4 b^4 + A_3 b^3 + A_2 b^2 + A_1 b + A_0 = 0 \quad (3.154)$$

where

$$\begin{cases} A_4 = 4E \\ A_3 = 8E \frac{B}{C} \tan \beta_2^2 \\ A_2 = 4E \{ [\frac{B}{C} \tan \beta_2^2]^2 - \tan \beta_2^2 (\frac{D}{C} + {}^T Z_0) \} \\ A_1 = 4E \frac{B}{C} \tan \beta_2^4 (\frac{D}{C} + z_0) - 2k + 2\frac{B}{C} \frac{A+Bk}{C} \tan \beta_2^2 \\ A_0 = \tan \beta_2^4 (\frac{D}{C} + {}^T Z_0)^2 + 2\frac{A+Bk}{C} \tan \beta_2^2 (\frac{D}{C} + {}^T Z_0) \\ E = [k^2 + 1 - (\tan \beta_2 \frac{A+Bk}{C})^2] \end{cases} \quad (3.155)$$

Substitute Equation (3.152) into Equation (3.154) to get a 4-degree equation that includes μ . The bi-section method is used to solve this equation. The two initial values of μ are set to 0 and μ_0 , where μ_0 is the value that guarantees the edge is gouge-free (the distance d from the cutter axis to the edge is larger than R).

$$d = R = \frac{({}^T Y_1 + \mu_0 {}^T Y_V) - k_{xy} ({}^T X_1 + \mu_0 {}^T X_V)}{\sqrt{k_{xy}^2 + 1}} \quad (3.156)$$

where

$$k_{xy} = \frac{{}^T Y_2 - {}^T Y_1}{{}^T X_2 - {}^T X_1} \quad (3.157)$$

$$\mu_0 = \frac{R\sqrt{k_{xy}^2 + 1} - {}^T Y_1 + k_{xy} {}^T X_1}{{}^T Y_V - k_{xy} {}^T X_V} \quad (3.158)$$

Then, μ_0 is calculated using Equation (3.158). If μ is not found, go to the next area.

If μ is positive, substitute it into Equation (3.153) to calculate ${}^T X_Q$, which is the location of the contact point. If μ is negative or zero, go to next case area.

${}^T Z_Q$ is discovered with the equation of $\overline{P_1^* P_2^*}$. If both Equations (3.159) and (3.160) are satisfied, edge gouging is eliminated and triangle gouging should be checked next. Otherwise, check the torus region next.

$${}^T X_1^* \leq {}^T X_Q \leq {}^T X_2^* \quad (3.159)$$

$$Lc - L \leq {}^T Z^* \leq Lc \quad (3.160)$$

- Torus region

The method to eliminate torus gouging is similar to that of taper gouging.

The intersection curve C of the torus and the plane π built with $P_1 P_2$ and V is given by Equation (3.161).

There is only one root in Equation (3.161) because the edge and the cutter are tangent to each other.

$$({}^T X^2 + {}^T Y^2 + (-\frac{A^T X + B^T Y + D}{C})^2 + e^2 - r^2)^2 - 4e^2({}^T X^2 + {}^T Y^2) = 0 \quad (3.161)$$

Simplify Equation (3.161) into to Equation (3.162).

Then, we have

$$A_4 {}^T X^4 + A_3 {}^T X^3 + A_2 {}^T X^2 + A_1 {}^T X + A_0 = 0 \quad (3.162)$$

where

$$\begin{cases} A_4 = 4E \\ A_3 = 8E \frac{B}{C} \tan \beta_2^2 \\ A_2 = 4E \left\{ \left[\frac{B}{C} \tan \beta_2^2 \right]^2 - \tan \beta_2^2 \left(\frac{D}{C} + z_0 \right) \right\} \\ A_1 = 4E \frac{B}{C} \tan \beta_2^4 \left(\frac{D}{C} + z_0 \right) - 2k + 2 \frac{B}{C} \frac{A+Bk}{C} \tan \beta_2^2 \\ A_0 = \tan \beta_2^4 \left(\frac{D}{C} + z_0 \right)^2 + 2 \frac{A+Bk}{C} \tan \beta_2^2 \left(\frac{D}{C} + z_0 \right) \\ E = \left[k^2 + 1 - \left(\tan \beta_2 \frac{A+Bk}{C} \right)^2 \right] \end{cases} \quad (3.163)$$

$\overrightarrow{P_1^* P_2^*}$ is tangent to the torus. As a result, there is only one contact point between $\overrightarrow{P_1^* P_2^*}$ and the torus, which means there is only one root in Equation (3.162). Equation (3.164) is obtained from Equation (3.162). Equation (3.165) is expanded from Equation (3.164).

$$({}^T X - {}^T X_0)^4 = 0 \quad (3.164)$$

$${}^T X^4 + 4 {}^T X_0 {}^T X^3 + 6 {}^T X_0^2 {}^T X^2 + 4 {}^T X_0^3 {}^T X + {}^T X_0^4 = 0 \quad (3.165)$$

Then, the coefficients in Equation (3.162) will satisfy the relationships in Equation (3.166).

$$A_4 : A_3 : A_2 : A_1 : A_0 = 1 : 4 {}^T X_0 : 6 {}^T X_0^2 : 4 {}^T X_0^3 : {}^T X_0^4 \quad (3.166)$$

Now, Equation (3.167) is obtained.

$$\frac{A_3}{A_4} = \frac{8A_2}{3A_3} \quad (3.167)$$

Next, b is obtained with Equation (3.168), which is derived from Equation (3.167).

$$(P + QB^2 + RB)b^2 + (2QBD + RD)b + [QD^2 - 8e^2(k+1)] = 0 \quad (3.168)$$

where

$$\begin{cases} P = 4k^2(2 - 3E) \\ Q = \frac{2}{C^4}(2 - 3E)(A + kB)^2 + \frac{2E}{C^2} \\ R = \frac{8k}{C^2}(2 - 3E)(A + kB) \end{cases} \quad (3.169)$$

If b is obtained, calculate μ using Equation (3.152). If b is not found, go to the next case.

If μ is positive, substitute it into Equation (3.153), and calculate the value of the ${}^T X_Q$ contact point. If μ is negative or zero, go to the next case.

${}^T Z_Q$ is calculated with the equation for $\overline{P_1^* P_2^*}$. If both Equation (3.170) and Equation (3.171) are satisfied, the edge gouging is eliminated and we should next check for triangle gouging. Otherwise, go to the next area (the cone region).

$${}^T X_1^* \leq {}^T X_Q \leq {}^T X_2^* \quad (3.170)$$

$$R_1 \tan \beta_1 \leq {}^T Z^* \leq L_C - L \quad (3.171)$$

- Cone region

The cone can be represented with Equation (3.50).

The only difference between the taper and the cone is that the top points and the slope for each shape are different. The equation for cone gouging elimination is obtained by substituting β_2 and Z_0 in Equation (3.154) with β_1 and 0 .

- Gouging at the intersection curve of the taper and the torus

The intersection curve of the taper region and the torus region is a circle (C_{TT}) as given by Equation (3.172). The offset edge may contact the cutter at C_{TT} , which is given by Equation (3.172).

Assume the contact point is Q .

Then,

$$\begin{cases} {}^T X = R_2 \cos \theta \\ {}^T Y = R_2 \sin \theta \\ {}^T Z = L_c - L \end{cases} \quad (3.172)$$

Substitute Equation (3.172) into Equation (3.145) to obtain θ_Q .

$$\theta_Q = \sin^{-1} \left(\frac{-C(L_c - L) - D}{R_2 \sqrt{A^2 + B^2}} \right) - \alpha \quad (3.173)$$

where

$$\begin{cases} \sin \alpha = \frac{A}{\sqrt{A^2 + B^2}} \\ \cos \alpha = \frac{B}{\sqrt{A^2 + B^2}} \end{cases} \quad (3.174)$$

Then μ is determined using Equation (3.175).

$$\mu = \frac{k^T Y_1 - {}^T X_1 - k \sin \theta + \cos \theta}{{}^T X_v - k^T Y_v} \quad (3.175)$$

Again, if Q is at the edge and the circle, this gouging case occurs. Otherwise, go to the next case.

- Intersection curve of cone and torus

The intersection curve of the cone region and the torus region is a circle (C_{CT}). The equation to solve this problem is similar to the one above. The only difference is that the radius of circle C_{CT} is R_1 instead of R_2 .

3. Triangle gouging

- Taper region

There is no possibility that the offset triangle will contact the taper. We do not give detailed explanation here.

The reason can be found directly.

- Torus region

The methods used in the torus region gouging case in section 3.5.4 can be applied here. The difference is that the triangle here is translated along the vector V instead of ${}^G Z$.

The triangle $\Delta P_1 P_2 P_3$ is offset along a vector until it contacts to the torus at point Q . The plane parallel to $\Delta P_1 P_2 P_3$ and consisting of point Q is represented by Equation (3.176):

$$a \cdot {}^T X + b \cdot {}^T Y + c \cdot {}^T Z + d^* = 0 \quad (3.176)$$

where

$$d^* = d - \mu(a^T X_v + b^T Y_v + c^T Z_v) \quad (3.177)$$

In this case, the contact point that is calculated with Equation (3.77) is modified to Equation (3.178).

$$\begin{cases} {}^T X_Q = -\frac{l \cdot a}{\sqrt{a^2 + b^2}} \\ {}^T Y_Q = -\frac{l \cdot b}{\sqrt{a^2 + b^2}} \\ {}^T Z_Q = -\frac{a \cdot {}^T X_Q + b \cdot {}^T Y_Q + d^*}{\sqrt{a^2 + b^2}} \end{cases} \quad (3.178)$$

This point is also on the torus. Therefore,

$$({}^T X_Q^2 + {}^T Y_Q^2 + {}^T Z_Q^2 + e^2 - r^2)^2 - 4e^2({}^T X_Q^2 + {}^T Y_Q^2) = 0 \quad (3.179)$$

Substitute Equation (3.178) into Equation (3.179). Then an implicit equation of μ is obtained:

$$\mu = \frac{(l - rc)\sqrt{a^2 + b^2} - d}{a^T X_v + b^T Y_v + c^T Z_v} \quad (3.180)$$

Using this implicit equation, we can solve for μ . We can then determine the location of the contact point Q using Equation (3.178). If point Q is in the torus region and within the triangle, edge gouging is eliminated. Otherwise, go to the next case.

To determine whether point Q is within the triangle, Equation (3.181) is applied.

$$\begin{cases} P = uP_1 + vP_2 + \omega P_3 \\ u + v + \omega = 1 \end{cases} \quad (3.181)$$

Solving it will yield

$$\begin{cases} u = \frac{({}^T X - {}^T X_3)({}^T Y_2 - {}^T Y_3) - ({}^T Y - {}^T Y_3)({}^T X_2 - {}^T X_3)}{({}^T X_1 - {}^T X_3)({}^T Y_2 - {}^T Y_3) - ({}^T Y_1 - {}^T Y_3)({}^T X_2 - {}^T X_3)} \\ v = \frac{({}^T X - {}^T X_3) - u({}^T X_1 - {}^T X_3)}{({}^T X_2 - {}^T X_3)} \\ \omega = 1 - u - v \end{cases} \quad (3.182)$$

If, u , v and ω are all in the region of $[0, 1]$, the point Q is inside the triangle.

Otherwise, it is outside of the triangle.

- Cone region

The only possibility that the cutter would contact the cone is when the triangle contacts the tip of the cutter. Or in other words, when the cutter center is inside the translated triangle $\Delta P_1 P_2 P_3$.

There are two conditions to determine whether a 3D point is inside a triangle: (1) The point is on the triangle plane; or (2) The projection of the point onto the ${}^T X^T Y$ plane is inside the projection of the triangle on the ${}^T X^T Y$ plane as given by Equation (3.181).

To determine whether condition 1 is satisfied, substitute the tip $(0,0,0)$ of the cutter into Equation (3.176) to check if d^* is 0. If d^* is 0, the center point is on the triangle plane. Then, solve for μ .

$$\mu = \frac{d}{a^T X_v + b^T Y_v + c^T Z_v} \quad (3.183)$$

Now, with μ , you can determine the offset triangle. Equation (3.181) is then applied to determine whether condition 2 is satisfied.

- Gouging at the intersection curve of the taper and the torus (C_{TT})

The plane consisting of the circle C_{TT} is π_2 . The intersection line of π_2 and the offset triangle $\Delta P_1 P_2 P_3$ is l , which is given by Equation (3.184).

$$ax + by + c(L_C - L) + d^* = 0 \quad (3.184)$$

If the distance from the cutter axis to line l is R_2 , the triangle is tangent to circle C_{TT} as given by Equation (3.185).

$$\frac{c(L_C - L) + d^*}{\sqrt{a^2 + b^2}} = R_2 \quad (3.185)$$

Then, μ is obtained using Equation (3.186), which is derived from Equation (3.185).

$$\mu = \frac{d - c(L_C - L) - R_2 \sqrt{a^2 + b^2}}{ax_v + by_v + cz_v} \quad (3.186)$$

Substitute Equation (3.172) into Equation (3.184) to obtain θ_Q .

$$\theta_Q = \sin^{-1} \left(\frac{-c(L_C - L) - d^*}{R_2 \sqrt{a^2 + b^2}} \right) - \alpha \quad (3.187)$$

where

$$\begin{cases} \sin \alpha = \frac{a}{\sqrt{a^2 + b^2}} \\ \cos \alpha = \frac{b}{\sqrt{a^2 + b^2}} \end{cases} \quad (3.188)$$

Again, the offset triangle is obtained with μ and Q is obtained with θ_Q . Equation (3.181) is applied to determine whether Q is inside the offset triangle.

- Gouging at the intersection curve (C_{CT}) of the cone and the torus

The only difference of this case and the former case is that the radius of circle C_{CT} is R_1 instead of R_2 .

With the detailed analysis performed in this section, the cutter position is accurately decided. The optimal clean-up tool path, then, is the connection of all cutter center points.

4 IMPLEMENTATION

This chapter shows the implementation of the TMM system. First, the software architecture is explained. Computer-Aided Software Engineering (CASE) tools (such as Rational Rose software) are used to design a robust and expandable system. The hardware involved in this dissertation is also introduced.

4.1 Architecture of the TMM system

4.1.1 Data flow diagram

The architecture of the TMM system is built using software engineering methods. The data flow diagram of this TMM system is shown in Figure 4-1. Given Parametric Model files, we first must convert them to a tessellated STL representation as the basis for sample data to use in the testing of our methods and algorithms. The TMM process begins when my method orders the triangles within the facet buckets. Next, the four machining stage operations are executed, either in order or separately. If the operations are performed in order, the final optimal tool path is generated in the clean-up machining stage. Otherwise, the optimal tool path may be planned at each stage. With this optimal tool path, the simulation process is applied to check the machining result, and finally the planned tool path is executed on the Tarus 5-axis mill creating the physical design model.

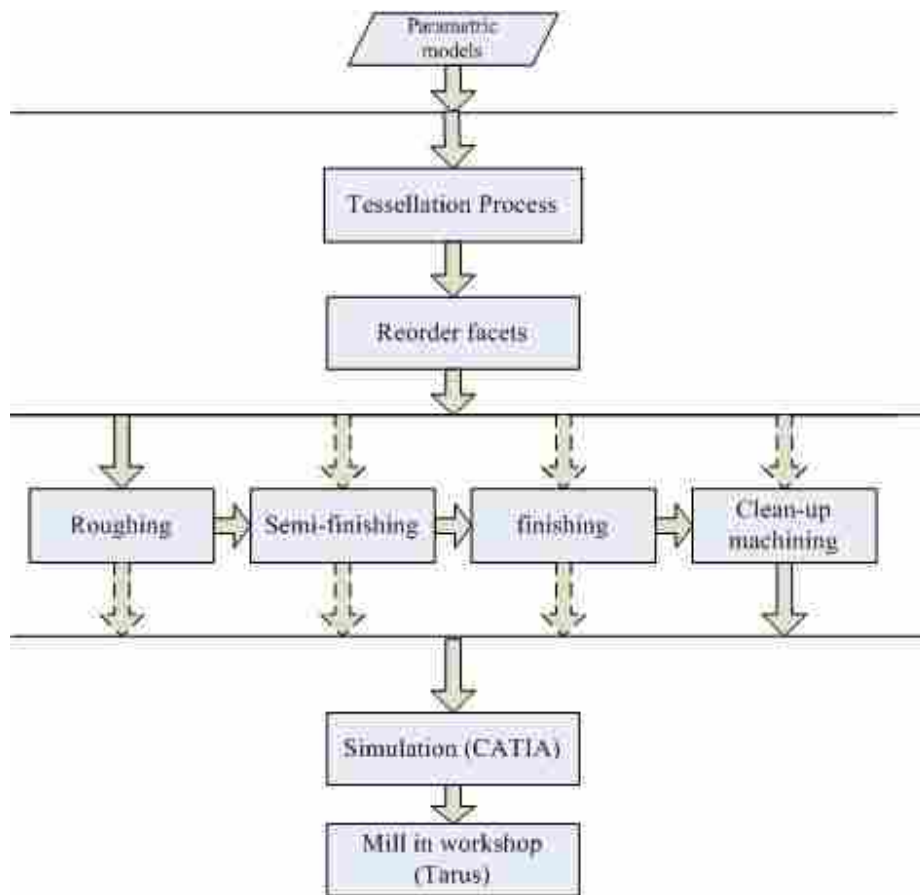


Figure 4-1 TMM software architecture.

4.1.2 User-case view

UML (unified modeling language) is an efficient tool used to help specify, visualize, and document software system models, including the models' structure and design, in a way that meets all of the above requirements [1]. Views and diagrams are two important UML tools that are used to construct the TMM system in this dissertation.

Figure 4-2 is the user-case view of this TMM system. This figure shows the relation between users (operators of the TMM system) and cases (execution modules of the TMM system) more clearly than the data flow diagram in Figure 4-1. 2.5D machining

and 3D machining are two routines in the rough machining module. The finish machining module consists of 3-axis machining and 5-axis machining routines, while the 5-axis machining routine is composed of CM² and 3D C-space sub-routines.

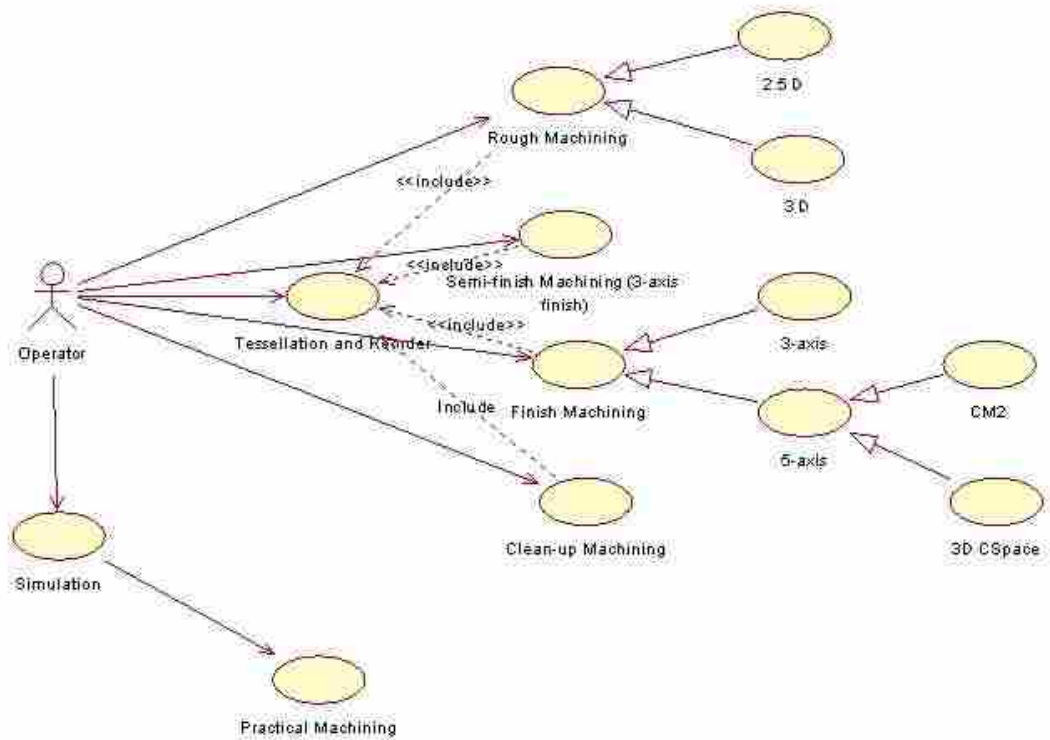


Figure 4-2 User-case view of TMM.

4.1.3 Sequence diagram

A sequence diagram is used to represent the sequence of operations in UML. The operations in each machining stage of SSM are similar. Figure 4-3 is the sequence diagram of the finish-machining stage. Each operation in Figure 4-3 corresponds to a method of class *C3DCSpaceMachining*, which is the key class for the 3D C-space method in this dissertation.

In Figure 4-3, the first operation (*CCTPath planning*) is to plan CC points. Then, the *Gouging elimination* operation is executed. Next, the *Generate the CLTPath* algorithm runs creating CL tool paths. Finally, the *Output CLTPath* operation shows the optimal CL points to users.

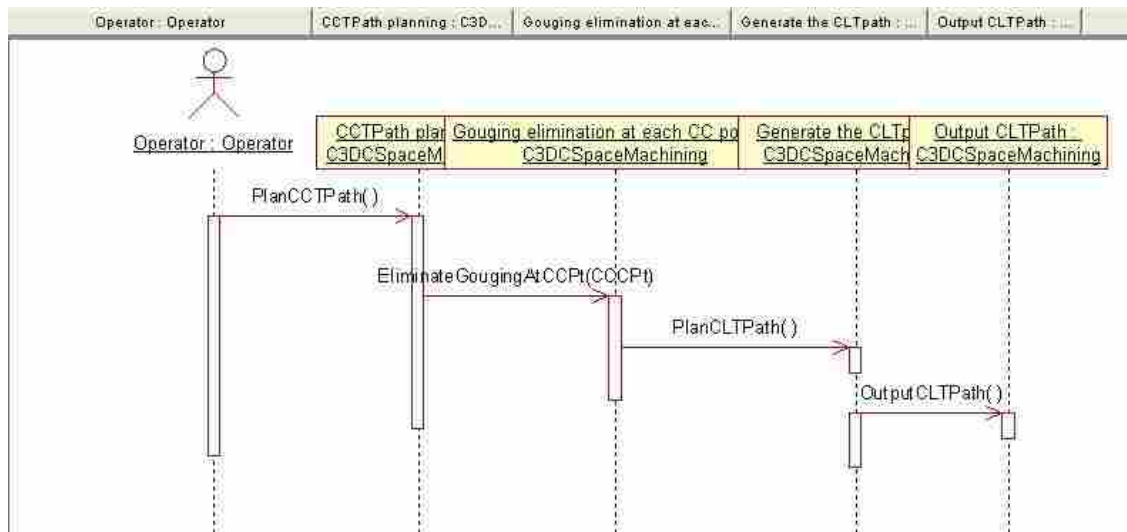


Figure 4-3 Operation sequence diagram of finishing.

4.1.4 Class diagram

Figure 4-4 is a class hierarchy chart indicating the key data structures of the TMM system. Each individual class (or *machining class* such as CRouging, C3DFinishing, etc.) is designed to handle one machining method. Each class has similar operations, as shown in Figure 4-3: CC tool path planning, gouging elimination, and CL tool path generation. As a result, we can use class inheritance to design some operations in a base class and build on these operations in individual classes. CTMM is the base class in this instance. This class is where several methods that are common to most classes are built. However,

a method like *PlanCCTPath()* is considered as a *virtual* function because it is different in each *machining class* and would be overloaded in derived classes. The keyword *virtual* is used to define a virtual method. Other methods, such as *SetSTLFile()*, are common methods used by all derived classes and are implemented in the base class.

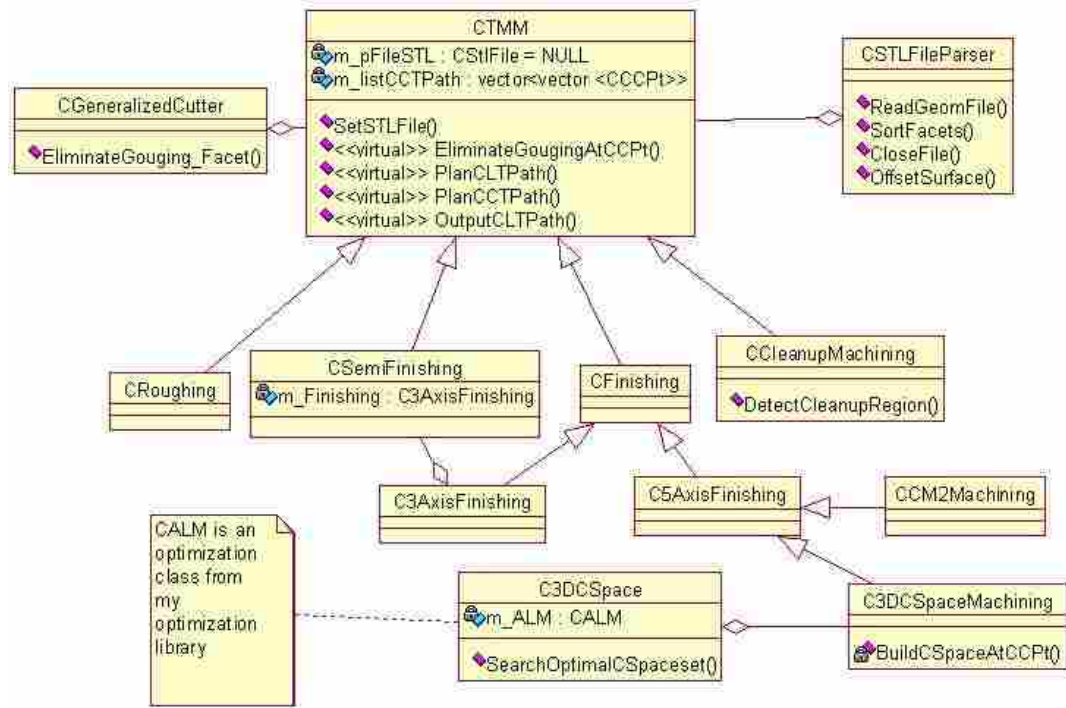


Figure 4-4 Class diagram of TMM.

Other details of the TMM system have been discussed in different sections above.

4.2 Hardware

The TMM system is developed on a HP xw4300 Workstation with the WindowsXP operating system.

The machining kinematics of this dissertation is with respect to the 5-axis Tarus mill machine as shown in Figure 4-5. This mill machine has five tool axes. The tilt head (C-axis) is on the top of the rotation head (A-axis). The speed of each joint is as follows: the X joint: 170 mm/s, the Y joint: 180 mm/s, the Z joint: 240 mm/s, the C joint: 0.3 rad/s, and the A joint: 0.5 rad/s. The tool path file generated with this TMM system can be machined directly with the Tarus.



Figure 4-5 Five-axis full-size Tarus mill

5 RESULTS

5.1 Overview

This chapter covers the machining results of this dissertation. It begins by describing the four surfaces that were tested with various types of curvatures: a parabolic surface, an elliptic surface, a hyperbolic surface, and an A-pillar surface. (Every sculptured surface in practice can be composed of the first three testing surfaces.)

Next, the machining processes are executed individually in each machining stage. In each machining stage, the machining parameters are listed first. Machining results are then presented. Detailed analyses are performed for the 5-axis machining results because 5-axis machining is the key part of this dissertation.

5.2 Benchmarks

To test the methods in this dissertation, the benchmarks shown in Figure 5-1 were machined with the developed TMM system. Benchmark (a), (b), and (c) are *through-curve* surfaces which are built with several section strings. The shape of the section strings of Benchmark (a), (b), and (c) are strictly concave, strictly convex and a combination of both concave and convex curvatures. In other words, surface (a) is parabolic, surface (b) is elliptic, while surface (c) is a free-form surface that includes

hyperbolic parts. The size of these three benchmarks is 10 inches x 10 inches. Benchmark (d) is a the A-pillar model of the Ford GT. Parabolic points, elliptical points, and hyperbolic points exist on different segments of the A-pillar surface.

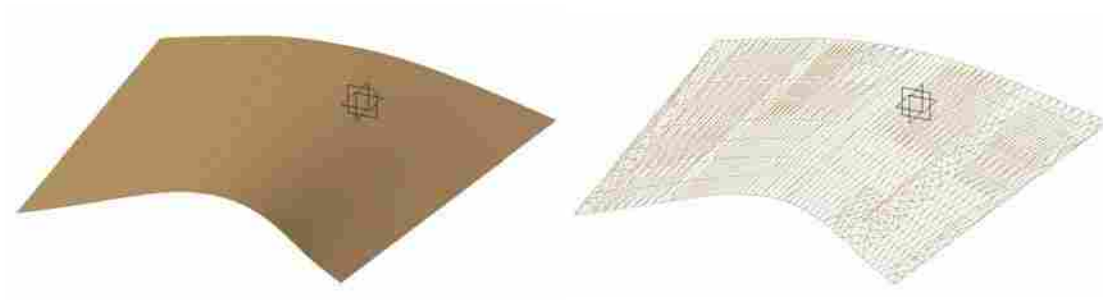
The surfaces in Figure 5-1 are tessellated from parametric models built inside a CAD application. The tessellation tolerance is 0.002 inch. The facet amount of the hyperbolic surface is about 15 times larger than that of a parabolic surface or an elliptic surface even at the same size. The density of facets around a point on a surface indicates the curvature around that point. Normally, the density of facets around a point on the surface is high if the corresponding curvature is large.

The tool path in this dissertation is planned with an HP personal workstation with a 3.4GHZ CPU and 3 gigabytes of memory.

In this section, the inch is used as the length unit. Different cutters, such as the ball end cutter (BEC) or the flat end cutter (FEC), are used in this section. So, results in this section can show methods in this dissertation well.



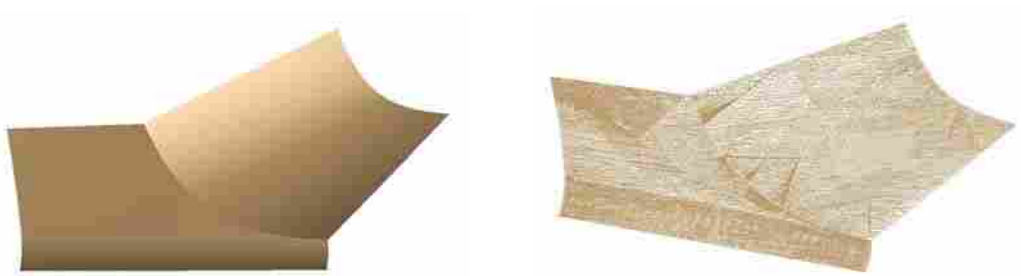
(a) Concave surface.



(b) Convex surface.



(c) Complex surface.



(d) A-pillar.

Figure 5-1 Benchmarks.

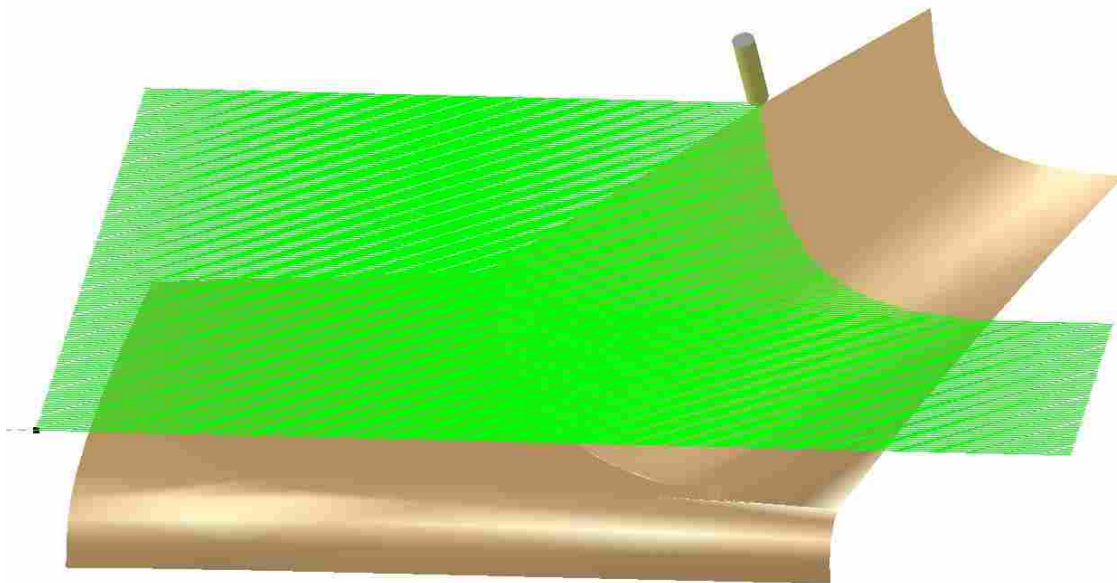
5.3 Rough machining

As what is discussed, much material is cut at a high rate in rough machining. So, a benchmark with much material should be selected as the test part in rough machining.

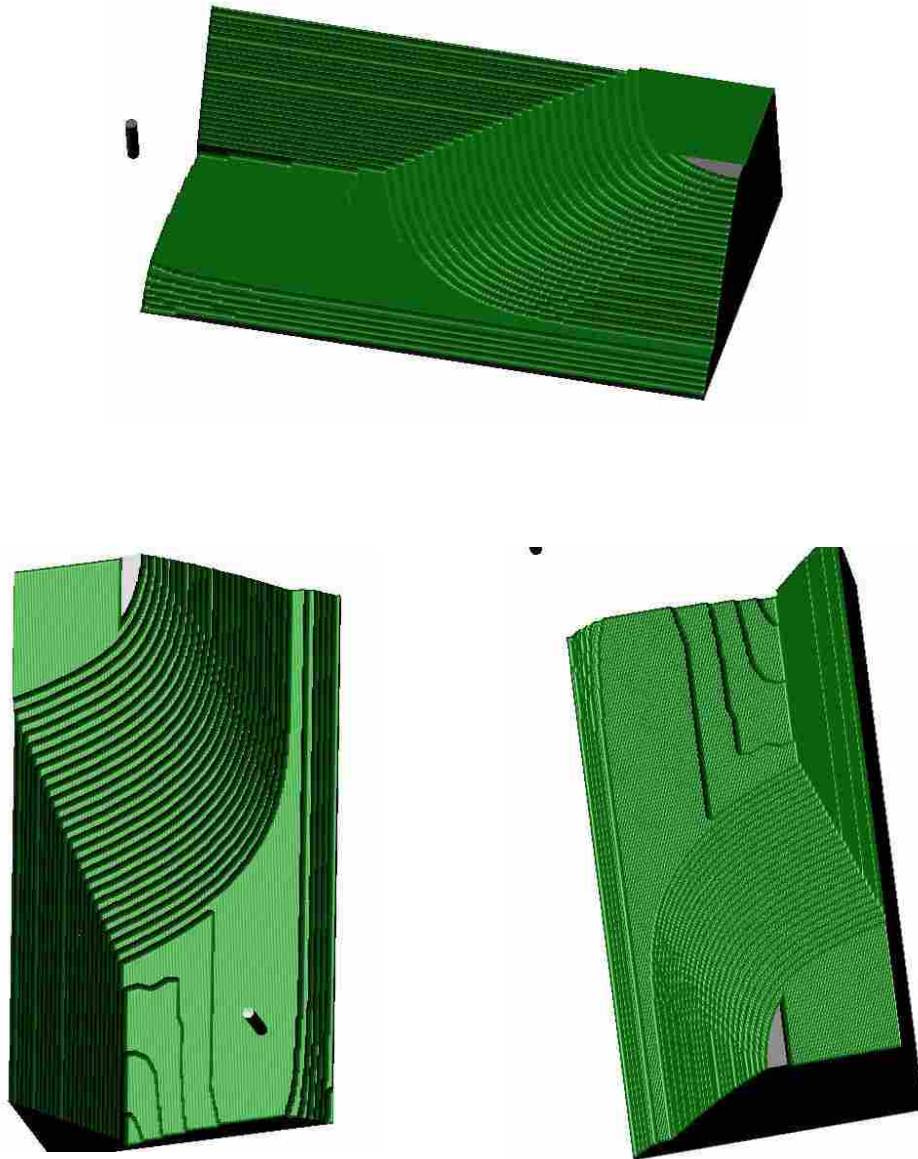
In this dissertation, we used 2.5 axis machining method to machine the part in rough machining. This method works well for most of parts. However, if machining surfaces do not have a whole bulk of material, 3-axis machining method or even 5-axis machining with a fixed cutter posture can be applied, too.

Table 5-1 Rough machining parameters.

Benchmark	Cutter	Parameters	Machining Tolerance
A-pillar	BEC	D: 1	0.4



(a) One layer of tool path.



(b) Different views of machining results

Figure 5-2 Rough machining results.

The A-pillar is the only benchmark tested in rough machining because the rough machining methods presented in this dissertation do not work well for the shapes of other testing surfaces. Figure 5-2 is the rough machining result. The machining tolerance is

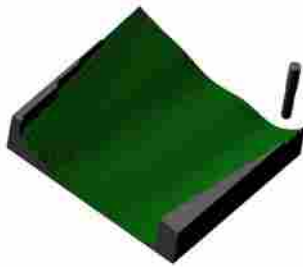
low if the stock material is metal. In this dissertation, the rough machining tolerance is set to 0.4 inches since the stock material is foam.

The 3-axis machining method with a high machining tolerance can be used if other testing surfaces need to be machined in the rough machining stage.

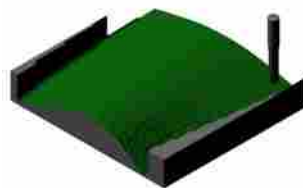
5.4 Three-axis finish machining

Table 5-2 3-axis finish machining parameters.

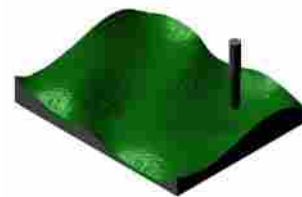
Benchmark	Cutter	Parameters	Machining Tolerance
Concave	BEC	D: 0.5	0.002
Convex	BEC	D: 0.5	0.002
Complex	BEC	D: 0.5	0.002



(a) Concave surface.



(b) Convex surface.



(c) Complex surface.

Figure 5-3 3-axis finish machining results.

Figure 5-3 shows the 3-axis finish machining results. In Chapter 3: Methodology, the generalized cutter (GC) is used in 3-axis finish machining. Here, a special cutter (BEC) that has been parameterized from the GC is used because the BEC is more popular than other cutters in the 3-axis finish machining industry.

5.5 Five-axis finish machining

Figure 5-4 is a 3D C-space built in the tool path planning process. The boundary curves of the top surface indicate the cutter posture used to generate the designed machining tolerance. When λ is 0.25π , the maximum δ occurs while ω is 0. When ω is constant, δ decreases while λ increases. These conclusions verify common sense.

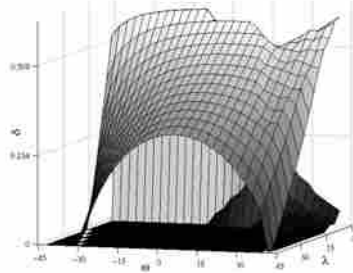


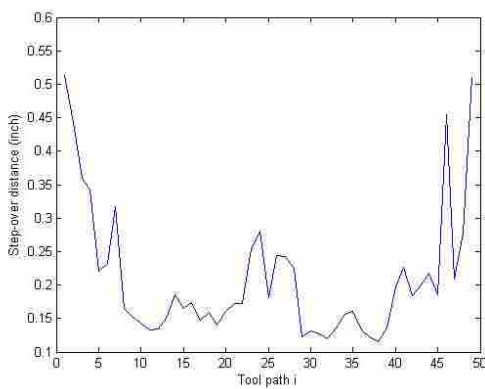
Figure 5-4 A sample 3D C-Space.

Figure 5-5 is the step-over distance of the convex surface. In Figure 5-5(a), the step-over distance is determined by the local geometric information just as discussed in section 3.5.2.2. There are a total of 49 tool paths. The step-over distance is less when the local curvature is large.

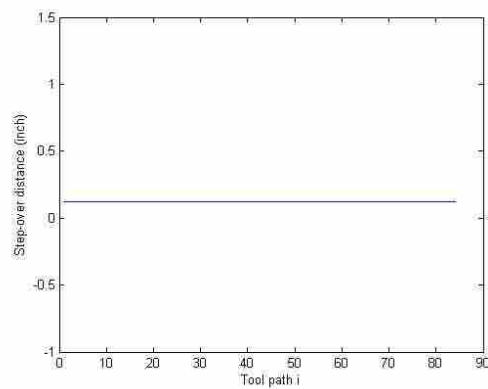
In Figure 5-5(b), the step-over distance is set to a constant. To eliminate gouging, the step-over distance should be set equal to the smallest possible step-over distance (0.12112 inch) in Figure 5-5(a). As a result, the value of the tool path in Figure 5-5(b) is 84, which is much larger than that of Figure 5-5(a). If the machining time along each tool path is assumed to be the same, the total machining time of the process using the step-

over distance of Figure 5-5(b) is 1.714 times longer than that of Figure 5-5(a). The machining results of Figure 5-5(a) and (b) are shown in Figure 5-6(b) and (c).

Also, at some regions of the machined part in (b), the machining quality will be unnecessarily high. More seriously, the height of the scallop height may change sharply in (b). As a result, the machining quality in (b) may be poor.



(a) Step-over distance is determined by the local geometry information.

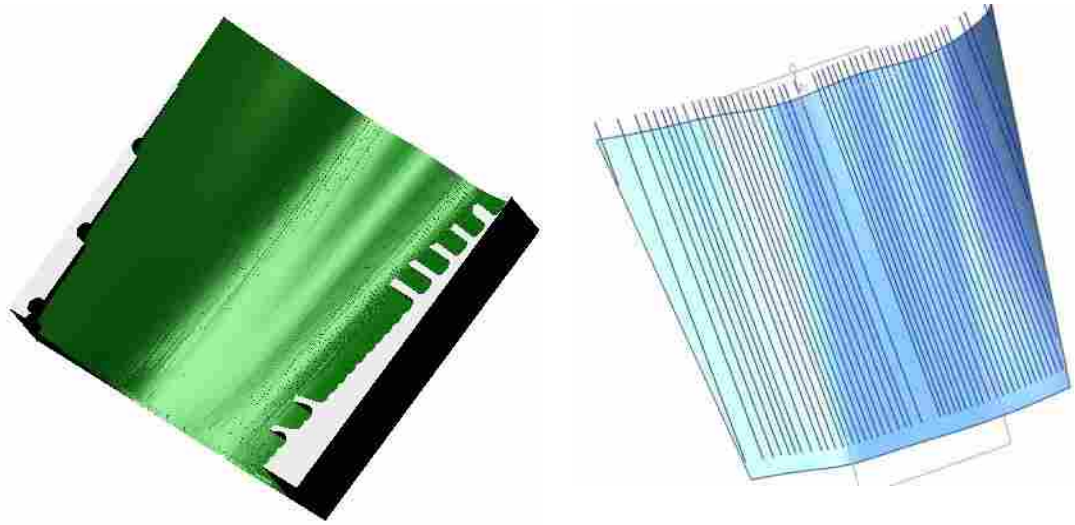


(b) Step-over distance is a constant

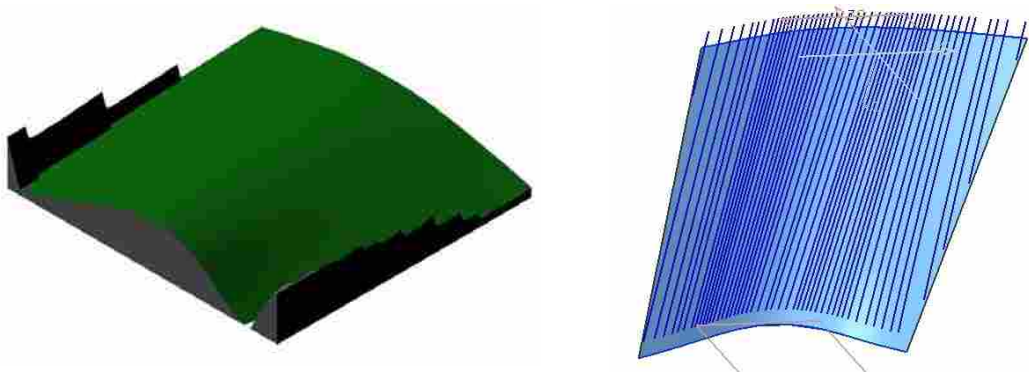
Figure 5-5 Step-over distance of convex surface.

Table 5-3 5-axis finish machining parameters.

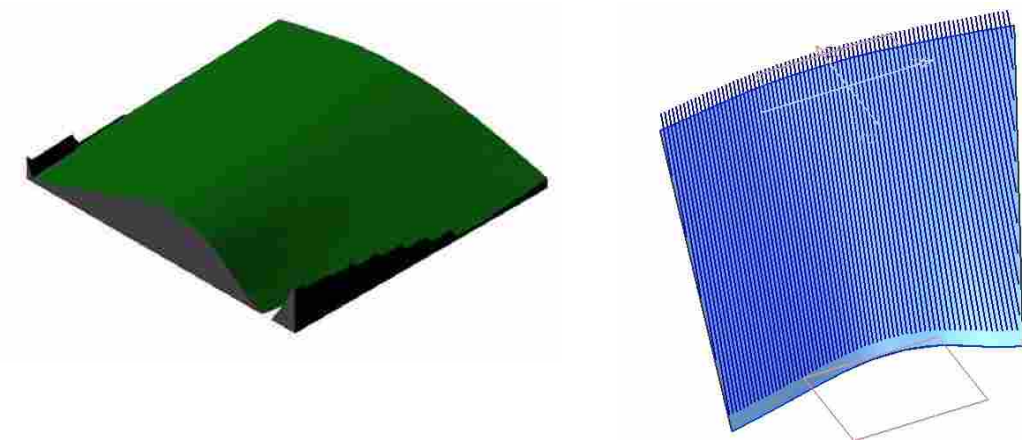
Benchmark	Cutter	Parameters	Machining Tolerance
Concave	FEC	D: 0.5	0.004
Convex	FEC	D: 0.5	0.004
Complex	FEC	D: 0.5	0.004



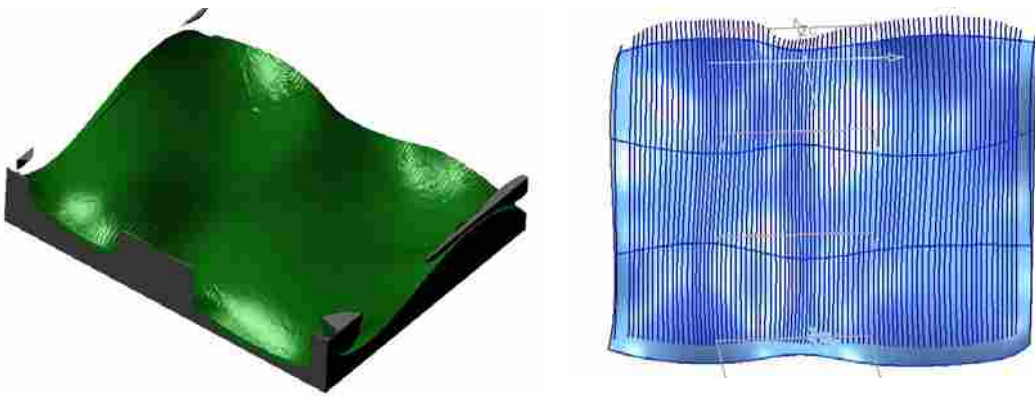
(a) Concave surface machined with the 3D C-space method.



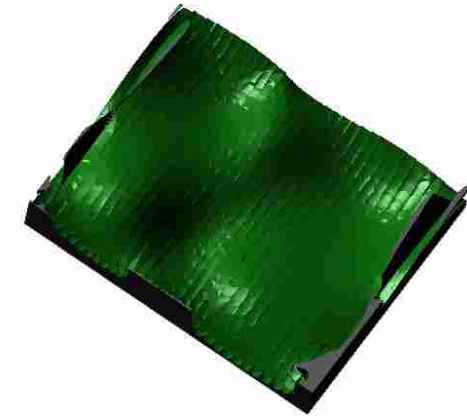
(b) Convex surface machined with the 3D C-space method.



(c) Convex surface with constant step-over distances.



(d) Complex surface machined with the 3D C-space method.



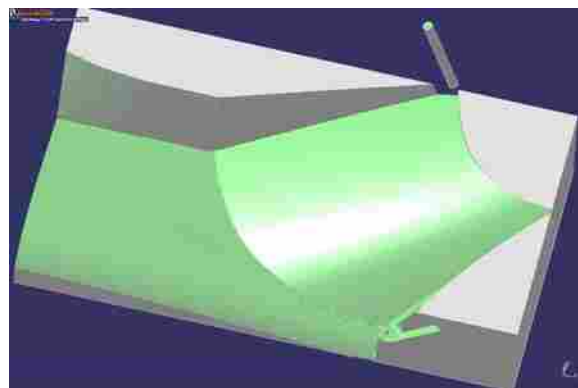
(e) Complex surface machined with the 3D C-space method.



(f) Complex surface machined in Application A.



(g) Complex surface machined in Application B.



(h) A-pillar machined with the 3D C-space method.

Figure 5-6 5-axis finish machining results.

Figure 5-6 displays the simulation results. The CL tool path was planned with 3D C-space methods, except for the results in Figure 5-6(f) and (h). If a 3D C-space is not found at a CC point, this CC point will be marked as a clean-up point and be machined in the next machining process (clean-up machining). Surfaces shown in Figure 5-6(a), (b),

(c), (d), and (h) are machined with a machining tolerance of 0.004 inches, and the surfaces shown in Figure 5-6(e), (f), and (g) are machined with a machining tolerance of 0.03 inches.

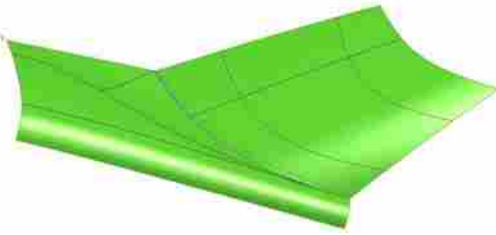
There are 30 tool paths generated in Figure 5-6(e). The complex surface is machined in two popular CAD applications with the same machining tolerance. With CAD Application A, the machining operation with a machining tolerance of 0.03 inches crashes for unknown reasons. Figure 5-6(f) is the machining result with 40 tool paths. The step over distance is set to a constant (0.25 inches). The comparison shows that the machined surface in Figure 5-6(e) is more even than that in Figure 5-6(d). With the fixed step-over distance, there is no way to verify whether the cusp height is larger than the machining tolerance using CAD Application A. In CAD Application B, the result with a machining tolerance of 0.03 inches is shown in Figure 5-6(e). A total of 182 tool paths are generated, which leads to a time-consuming machining process. Similarly, the machining result has an unnecessarily high quality.

5.6 Clean-up machining

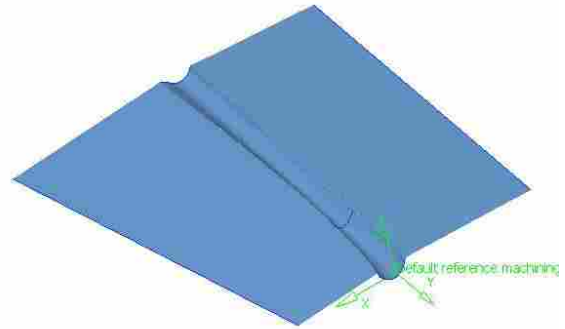
The A-pillar and a surface with a gap are tested in clean-up machining. Figure 5-7(a) is the CL tool path generated with the clean-up machining methods introduced in this dissertation. Figure 5-7(b) is another test surface. Figure 5-7(c) is the CL tool path and (d) is the finished result. In Figure 5-7(a), the cutter is lifted along the vector of the points in the clean-up strip to eliminate gouging. In Figure 5-7(b), the cutter is lifted along the Z direction to eliminate gouging.

Table 5-4 5-axis finish machining parameters.

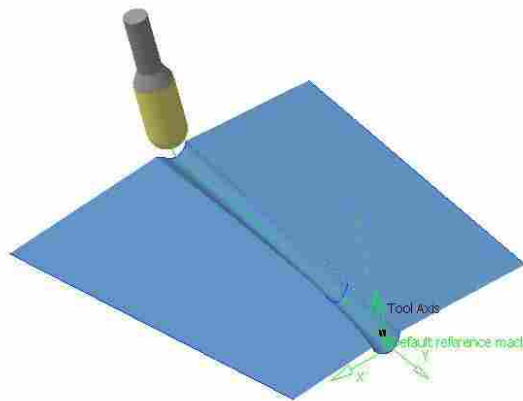
Benchmark	Cutter	Parameters	Machining Tolerance
A-pillar	BEC	D: 1	0.002
Surface b	BEC	D: 1	0.002



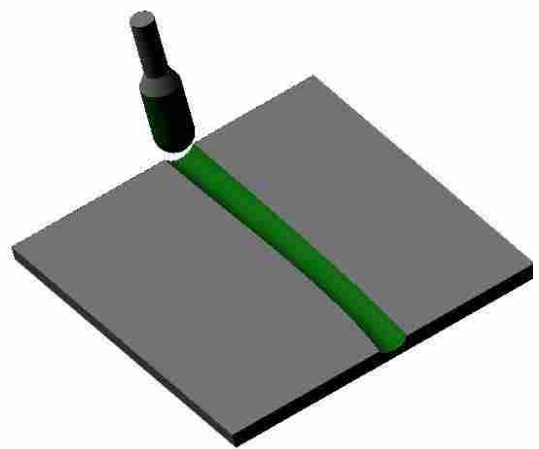
(a) CL tool path of A-pillar.



(b) A benchmark with a gap.



(c) A CL tool path of clean-up machining.



(d) Clean-up machining result.

Figure 5-7 Clean-up machining results.

6 CONCLUSIONS

6.1 Overview

This chapter summarizes the discoveries and developments of this research first. Then a comprehensive discussion of the contribution of this dissertation is presented. Next, conclusions of this dissertation are discussed in detail and finally, recommendations are given to future research in the field of sculptured surface machining (SSM).

6.2 Summary

This dissertation developed a complete TMM system. In this system, rough machining methods, finish machining methods, and clean-up machining methods are implemented. In each machining stage, mature or original methods are used as an initial benchmark and then further developed.

In the rough machining stage, a 2.5-degree machining approach is applied. In this approach, the stock material is cut layer by layer. Each single tool path in one layer is a straight line composed of two CL points. With this approach, the rough machining process in this research has a high machining speed. For some sculptured models, the remaining material height in the rough machining stage may be stair stepped and non-

uniform. In this case, a semi-finish machining operation is needed. The 3-axis machining method is used as the semi-finish machining operation in this TMM system.

Finish machining methods can be categorized into two types: 3-axis machining methods and 5-axis machining methods. We applied and modified existing 3-axis machining methods in this dissertation. A generalized cutter is used in 3-axis machining. CC points are planned by slicing the tessellated surfaces with drive planes. Then, CL points are derived based on these CC points using gouge elimination methods. Lifting the cutter along the Z direction is the approach that was used to eliminate gouging in 3-axis machining.

In 5-axis machining, this dissertation presents a new 3D C-space method for this process. Because the step-over distance is determined by the geometric features around the MRMH CC point, the generated step-over distance is set as large as possible. Moreover, the cutter posture at each CC point is chosen from within the 3D C-space with an optimization method that uses improved boundary conditions and objective functions. In addition, we introduced machine tool kinematics to the plan of improved tool paths. As a result, the tool path in this dissertation is better optimal than other existing methods.

In clean-up machining, a generalized cutter is used. The clean-up region is detected based on CC points generated in 3-axis machining. The clean-up region can be recognized according to the slope change of each segment of the tool path. The process of planning the CL points for clean-up is divided into two steps in this dissertation, which decreases the CPU computation time. To eliminate gouging problems efficiently, the cutter is offset along a vector determined by the shape of the clean-up strip instead of along the ${}^G Z$ direction.

Efficiency is the objective function of each machining method in this TMM system with the precondition of satisfactory accuracy.

6.3 Contributions

The principle contribution of this research is that we present a 3D C-space method for 5-axis machining. This method combines several existing 5-axis machining methods and considers more factors (such as the local geometry information of the surface, the machine tool kinematics, etc.) than prior research to plan a truly optimal tool path. The machining benchmarks show that the methods in this dissertation are robust and efficient.

The 3D C-space can also be applied to other machining methods. For example, the tool path generated in this dissertation are composed of piecewise straight lines, which can be interpolated to spline curves to improve the machining speed or to eliminate machine jerk ([29]). With regular machining methods, such as those presented in [29], it is challenging to guarantee that the cusp height determined by the spline tool path is less than the machining tolerance, and that the spline tool path is gouge-free. If the spline tool path is determined by the C-space set within the 3D C-space, this path will be gouge-free and the cusp height will be less than the machining tolerance if the optimal CL tool path is determined by a C-space set. Meanwhile, the optimization method used in this work can easily be changed to other search policies.

Finally, the CC points where 3D C-space does not exist will need further processing (clean-up machining). The intersections of these CC points delineates the clean-up

machining strip. In other words, the 3D C-space method provides a new method to recognize the clean-up machining strip.

The second contribution is that we present a more complete gouging elimination model for the generalized cutter. We applied the generalized cutter to the clean-up machining stage and the 3-axis finish machining. In the clean-up machining stage, we built a mathematic model for the gouging elimination process in which the generalized cutter is lifted along a vector instead of the ${}^G Z$ direction.

The third contribution is that we built a complete TMM system that includes all necessary machining stages. This system can be used as a platform for future TMM research because it provides most of the modules needed by TMM research.

Furthermore, our work can be applied not only to STL files but also other data sources (such as VRML, GT, etc.). It can also act as a base for machining via a global network.

6.4 Future research

The rough machining methods in this dissertation are not complete. Future research can be conducted to plan the CC tool path based on the 3-axis machining method. The CL tool path planned with the 3-axis machining method can be interpolated to be B-Splines or piecewise straight lines which can then be smoothly and quickly machined.

In 3-axis machining, a similar idea to our 3D C-space method can be presented in the future. A safe region that employs the cutter lifting height could be built at each CC point. A B-Spline CL tool path could be planned within this region based on the CL tool path generated in this dissertation.

In 5-axis machining, with the 3D C-space method in this dissertation, the scale for the time to generate the CC points, the time to build the bottom surface (S_2) of the 3D C-space, the time to build the top surface of the 3D C-space (S_1) and the time of the optimization process is 0.1:8:1:0.5. Therefore, the bottleneck of the tool path planning speed is in building the bottom surface of the 3D C-space.

This tool path planning strategy is computationally intensive and thus very time-consuming. Our future research will focus on improved methods for calculating the 3D C-space at each CC point. We will also focus on modified optimization methods and more efficient program data structures. The kinematics model applied in this dissertation is a simplified model and needs more work in order to consider such factors as machine jerk, accelerations, and decelerations. Finally, the 3D C-space method can be further applied to the generalized cutter.

In clean-up machining, a more detailed clean-up strip detection method is needed, especially for complicated surfaces. The 5-axis machining method can be used to machine the clean-up strip, instead of the 3-axis machining method, to improve the machining quality.

REFERENCES

- [1] Stereolithography Interface Specification, 3D Systems, Inc., June 1988.
- [2] <http://www.uml.org/>
- [3] Arkin E., Held M., and Smith C. L., Optimization problems related to zigzag pocket machining, *Algorithmica*, Vol. 26, No. 2, 2000, pp197–236.
- [4] Atlan, T.; Lilly, B. W.; Kruth, J. P.; Konig, W.; Tonshoff, H. K.; Khairy, A. B., Advanced techniques for die and mold manufacturing, *Annals of the CIRP*, 1993, 42, 707–716.
- [5] Balasubramaniam, M.; Ho, S.; Sarma, S. E.; and Adachi, Y., Generation of collision-free 5-axis tool paths using a haptic surface, *Computer-Aided Design*, Vol. 34, No. 4, 2002, pp 267–279.
- [6] Bentley, J. L., Multidimensional binary search trees used for associative searching, *Communications of the ACM*, Vol. 18, No. 9, 1993, pp509–517.
- [7] Baumgart, B. G., Winged edge polyhedron representation, Stanford University, Stanford, CA, 1972.
- [8] Chaine, R., A geometric convection approach of 3-D reconstruction, *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, 2003, pp218–229.
- [9] Choi, B.K.; Park, J.W.; Chung, Y.C., Variable radius blending by ball position sampling, *Proc. Of Pacific Conf. on Computer Graphics and Appl.*, World Scientific Pub. Co., 1993, pp221–234.
- [10] Choi, B.K.; Kim, Dae H; Jerard, Robert B., C-space approach to tool-path generation for die and mould machining, *Computer-Aided Design*, Vol. 29, No. 9, 1997, pp 657–669.

- [11] Choi, B. K. and Jerard, Robert B., *Sculptured Surface Machining, Theory and Applications*, USA: Kluwer Academic Publishers, 1998.
- [12] Choi, B.K.; Park, J.W.; and Jun, C.S., Cutter-location data optimization in 5-axis surface machining, Vol. 25, No. 6, 1993, pp 377–386.
- [13] Drysdale, R. L.; Jerard, R. B.; Schaudt, B.; and Hauck, K., Discrete simulation of NC machining, *Proceedings of the third annual symposium on Computational geometry*, 1987, pp 126–135.
- [14] Drysdale, R. L.; Ozair, K.; and Jerard, R. B., Discrete surface representations for simulation, verification, and generation of numerical control programs, *Proceedings of the 1991 NSF Design and Manufacturing Systems Conference*, Jan 1991, pp431–438.
- [15] Duncan, I. P.; and Mair, S. G., *Sculptured, Surfaces in Engineering and Medicine*, Cambridge University Press, Cambridge, MA, 1983.
- [16] Fletcher, R., *Practical Methods of Optimization*, USA: John Wiley & Sons, 1987.
- [17] De Floriani, L.; Magillo, P.; Puppo, E.; and Sobrero, D., A Multi-Resolution Topological Representation for Non-Manifold Meshes, *Proceedings Solid Modeling 2002*, 2002 pp 159–170.
- [18] Hwang, J.S., and Chang, T.C., Three-axis machining of compound surfaces using flat and filleted endmills, *Computer-Aided Design*, Vol. 30, No. 8, 1998, pp 641–647.
- [19] Held, M., *On the Computational Geometry of Pocket Machining*, Springer-Verlag, 1991.
- [20] Jensen, C.G., and Anderson, D.C., Accurate tool placement and orientation for finish surface machining, *Journal of Design and Manufacturing*, No. 3, 1993, pp 251–261.
- [21] Jensen, C.G.; Red, W. E.; and Pi, J., Tool selection for five-axis curvature matched machining, *Computer-Aided Design*, Vol. 34, No. 3, March 2002, pp 251–266.
- [22] Jin, S.; Lewis, R.B.; and West, D.; A Comparison of Algorithms for Vertex Normal Computation, www.tricity.wsu.edu/~bobl/personal/mypubs/2003_vertnorm_tvc.pdf.
- [23] Jun, C.S.; Cha, K.; and Lee, Y.S., Optimizing Tool Orientations for 5-Axis Machining by Configuration-Space (C-Space) Search Method, *Computer-Aided Design*, Vol. 35, No. 6, 2003, pp 549–566.

- [24] Jun, C.S.; Cha, K.; and Park, S., A new curve-based approach to polyhedral machining, *Computer-Aided Design*, Vol. 34, No. 5, 2002, pp 379–389.
- [25] Kim, B.H., and Choi, B.K., Guide surface based tool path generation in 3-axis milling: an extension of the guide plane method, *Computer Aided Design*, Vol. 32, No. 3, 2000, pp 191–199.
- [26] Kim, D.S.; Jun, C.S.; and Park, S., Tool path generation for clean-up machining by a curve-based approach, *Computer-Aided Design*, In Press.
- [27] Kiswanto, G.; Lauwers, B.; Kruth, J. P., Tool path generation for 5-axis milling based on faceted models, *Machine Engineering*, Vol. 2, 2002, p.99–109.
- [28] Lauwers, B.; Kiswanto, G.; Kruth, J.P., Development of a Five-axis Milling Tool Path Generation Algorithm based on Faceted Models, *CIRP Annuals*, Vol. 52, No. 1 2003, p.85.
- [29] Lee, C.H., Smooth tool-taxis variation in 5-axis surface machining, *Proceedings of IMECE2004*, 60276.
- [30] Lee, Y.S., and Chang, T.C., Two-Phase Approach to Global Tool Interference Avoidance in 5-axis Machining, *Computer Aided Design*, Vol. 27, No. 10, 1995, pp 715–729.
- [31] Lee, Y.S., Admissible Tool Orientation Control of Gouging Avoidance for 5-Axis Complex Surface Machining, *Computer Aided Design*, Vol. 29, No. 7, 1997, pp 507–520.
- [32] Lee, Y.S., Mathematical modeling using different endmills and tool placement problems for 4- and 5-axis NC complex surface machining, *International Journal of Production Research*, Vol. 36, No. 3, 1998, pp 785–814.
- [33] Lee, Y. S., and Koc, B., Ellipse Offset Approach and Inclined Zig-Zag Method for Multi-Axis Roughing of Ruled Surface Pockets, *Computer-Aided Design*, Vol. 30, No. 12, 1998, pp 957–971.
- [34] Li, S.X., and Jerard, R.B., Non-parametric Three Axis Tool Path Generation for Finish Machining of Sculptured Surfaces, *Geometric Modeling for Product Realization*, North-Holland, 1993, pp 251–256.
- [35] Li, S.X., and Jerard, R.B., 5-axis machining of sculptured surfaces with flat-end cutter, *Computer Aided Design*, Vol. 26, No. 3, 1994, pp 165–178.
- [36] Li, W.; Davis, T.; Jensen, C.G.; Red, W.E., Rapid and Flexible Prototyping through Direct Machining, *Computer-Aided Design and Applications*, Vol. 1, Nos. 1-4, CAD'04, 2004, pp. 91–100.

- [37] Lin, H.W.; Tai, C.L.; Wang, G. J., A mesh reconstruction algorithm driven by an intrinsic property of a point cloud, *Computer-Aided Design*, Vol. 36, No. 1, 2004, pp 1–9.
- [38] Lo, C-C, Two-stage cutter-path scheduling for ball-end milling of concave and wall-bounded surfaces, *Computer-Aided Design*, Vol. 32, No. 10, 2000, pp 597–603.
- [39] McMains, S.; Hellerstein, J.M.; Sequin, C.H., Out-of-Core Build of a Topological Data Structure from Polygon Soup, *Proceedings of the 6th ACM Symposium on Solid Modeling and Applications*, 2001, pp171–182.
- [40] Mencl, R., and Muller, H., Interpolation and Approximation of Surfaces from Three-Dimensional Scattered Data Points, *State of the Art Reports, Eurographics '98*, , 1998, pp 51–67.
- [41] Meyer, M.; Desbrun, M.; et al., Discrete Differential-Geometry Operators for Triangulated 2-Manifolds, *In Visualization and Mathematics III*, 2003, pp 35–57.
- [42] Nelson, M., Weights for Computing Vertex Normals from Facet Normals, *Journal of Graphics Tools*, Vol. 4, No 2, 1999, pp 1–6.
- [43] Rao, N.; Ismail, F.; Bedi, S., Tool path planning for five-axis machining using the principle axis method. *Int. J. Mach. Tools*, Vol. 37, No. 7, 1997, pp 1025–1040.
- [44] Rosenberg, J. B., Geographical Data Structures Compared: A study of Data Structures Supporting Region Queries, *IEEE Transactions on Computer-Aided Design*, Vol. CAD-4, No. 1, Jan. 1985, pp 53–67.
- [45] Rock, S. J., and Wozny, M. J., Generating Topological Information from a “Bucket of Facets”, *Solid Freeform Fabrication Symposium Proceedings*, 1992, pp. 251–259.
- [46] Ren, Y.; Yau, H.; Lee, Y.S., Clean-up Tool Path Generation by Contraction Tool Method for Machining Complex Polyhedral Models, *Computers in Industry*, Vol. 54, No. 1, 2004, pp. 17–33.
- [47] Ren, Y.; Zhu, W.; Lee, Y.S., Material side tracing and curve refinement for pencil-cut machining of complex polyhedral models, *Computer-Aided Design*, In Press.
- [48] Smith, S., and Tlustý, J., An overview of modeling and simulation of the milling process, *Trans. Of ASME, Jr. of Engr. for Industry*, Vol. 113, May 1991, pp169–175.
- [49] Tang, K.; Chou, S.-Y.; Chen, L.-L., An algorithm for reducing tool retractions in zigzag pocket machining, *Computer-Aided Design*, Vol. 30, No. 2, 1998, pp 123–129.

- [50] Tao, S., and Ting, K.-L., Unified rough cutting tool path generation for sculptured surface machining, *Computer-Aided Design*, Vol. 39, No. 13, 2003, pp 2973–2989.
- [51] Warkentin, A.; Ismail, F.; Bedi, S., Intersection approach to multi-point machining of sculptured surface, *Computer Aided Geometric Design*, Vol. 15, No. 6, 1998, pp 567–584.
- [52] Warkentin, A.; Ismail, F.; Bedi, S., Multi-Point Tool Positioning for Sculptured Surface Machining, *Computer Aided Geometric Design*, 2000, Vol. 17, pp. 83–100.
- [53] Weiler, K., The Radial Edge Structure: a Topological Representation for Non-manifold Geometric Boundary Modeling, M. J. Wozny, H. W. McLauhlin, and J. L. Encarnacao, editors, *Geometric Modeling for CAD Applications*, North-Holland, 1988, pp. 3–36.
- [54] Yau, H.T.; Chuang, C.M.; Lee, Y.S., Numerical control machining of triangulated surfaces in a stereo lithography format with a generalized cutter, *Int. J. Prod. Res.*, Vol. 42, No. 13, 2004, pp 2573–2598.
- [55] Zhu, W., and Lee, Y.S., Five-Axis Pencil-Cut Planning and Virtual Prototyping with 5-Degree-of-Freedom Haptic Interface, *Computer-Aided Design*, Volume 36, No. 13, 2004, pp 1295–1307.

APPENDIX A

Surface Data of Benchmark

Concave surface:

Two B-splines are built first. The concave surface is constructed with a through-curve method based on these two B-splines.

Control points of Spline 1:

Pole Number	1				
Weight	=	1.00000000000			
Coordinates	XC =	-5.31285703876	X =	0.00000000000	
	YC =	2.12375344346	Y =	-5.31285703876	
	ZC =	0.00000000000	Z =	2.12375344346	
Pole Number	2				
Weight	=	1.00000000000			
Coordinates	XC =	-4.78535414379	X =	0.00000000000	
	YC =	1.90646118910	Y =	-4.78535414379	
	ZC =	0.00000000000	Z =	1.90646118910	
Pole Number	3				
Weight	=	1.00000000000			
Coordinates	XC =	-3.80191222906	X =	0.00000000000	
	YC =	1.51159819394	Y =	-3.80191222906	
	ZC =	0.00000000000	Z =	1.51159819394	
Pole Number	4				
Weight	=	1.00000000000			
Coordinates	XC =	-2.08577960890	X =	0.00000000000	
	YC =	1.19894959495	Y =	-2.08577960890	
	ZC =	0.00000000000	Z =	1.19894959495	
Pole Number	5				
Weight	=	1.00000000000			
Coordinates	XC =	-0.89487840783	X =	0.00000000000	
	YC =	0.45060791096	Y =	-0.89487840783	
	ZC =	0.00000000000	Z =	0.45060791096	
Pole Number	6				
Weight	=	1.00000000000			
Coordinates	XC =	0.50415108187	X =	0.00000000000	

	YC =	0.48375602458	Y =	0.50415108187
	ZC =	0.00000000000	Z =	0.48375602458
Pole Number		7		
Weight	=	1.00000000000		
Coordinates	XC =	1.66808289465	X =	0.00000000000
	YC =	0.82895078376	Y =	1.66808289465
	ZC =	0.00000000000	Z =	0.82895078376
Pole Number		8		
Weight	=	1.00000000000		
Coordinates	XC =	3.38334878251	X =	0.00000000000
	YC =	0.11726632576	Y =	3.38334878251
	ZC =	0.00000000000	Z =	0.11726632576
Pole Number		9		
Weight	=	1.00000000000		
Coordinates	XC =	4.83002830859	X =	0.00000000000
	YC =	1.08557556228	Y =	4.83002830859
	ZC =	0.00000000000	Z =	1.08557556228
Pole Number		10		
Weight	=	1.00000000000		
Coordinates	XC =	5.26003777973	X =	0.00000000000
	YC =	1.58919644164	Y =	5.26003777973
	ZC =	0.00000000000	Z =	1.58919644164
Pole Number		11		
Weight	=	1.00000000000		
Coordinates	XC =	5.32325478899	X =	0.00000000000
	YC =	2.01809670345	Y =	5.32325478899
	ZC =	0.00000000000	Z =	2.01809670345

Control points of Spline 2:

Pole Number		1		
Weight	=	1.00000000000		
Coordinates	XC =	-4.62864577498	X =	10.00000000000
	YC =	1.08819913378	Y =	-4.62864577498
	ZC =	0.00000000000	Z =	1.08819913378
Pole Number		2		
Weight	=	1.00000000000		
Coordinates	XC =	-4.19939037727	X =	10.00000000000
	YC =	0.92959273666	Y =	-4.19939037727
	ZC =	0.00000000000	Z =	0.92959273666
Pole Number		3		
Weight	=	1.00000000000		
Coordinates	XC =	-3.27768154476	X =	10.00000000000
	YC =	0.61857372322	Y =	-3.27768154476
	ZC =	0.00000000000	Z =	0.61857372322
Pole Number		4		
Weight	=	1.00000000000		
Coordinates	XC =	-2.21778311551	X =	10.00000000000
	YC =	-0.12902186244	Y =	-2.21778311551

	ZC =	0.00000000000	Z =	-0.12902186244
Pole Number		5		
Weight	=	1.00000000000		
Coordinates	XC =	-0.52178263366	X =	10.00000000000
	YC =	-0.08723116013	Y =	-0.52178263366
	ZC =	0.00000000000	Z =	-0.08723116013
Pole Number		6		
Weight	=	1.00000000000		
Coordinates	XC =	0.81300482478	X =	10.00000000000
	YC =	-0.31172142360	Y =	0.81300482478
	ZC =	0.00000000000	Z =	-0.31172142360
Pole Number		7		
Weight	=	1.00000000000		
Coordinates	XC =	2.40149245577	X =	10.00000000000
	YC =	-0.29406723564	Y =	2.40149245577
	ZC =	0.00000000000	Z =	-0.29406723564
Pole Number		8		
Weight	=	1.00000000000		
Coordinates	XC =	3.35965360789	X =	10.00000000000
	YC =	-0.13408504800	Y =	3.35965360789
	ZC =	0.00000000000	Z =	-0.13408504800
Pole Number		9		
Weight	=	1.00000000000		
Coordinates	XC =	3.84727413810	X =	10.00000000000
	YC =	-0.01058633345	Y =	3.84727413810
	ZC =	0.00000000000	Z =	-0.01058633345

Convex surface:

Two B-splines are built first. The convex surface is constructed with a through-curve method based on these two B-splines.

Control points of Spline 1:

Pole Number		1		
Weight	=	1.00000000000		
Coordinates	XC =	-5.41253707665	X =	0.00000000000
	YC =	0.66931884577	Y =	-5.41253707665
	ZC =	0.00000000000	Z =	0.66931884577
Pole Number		2		
Weight	=	1.00000000000		
Coordinates	XC =	-4.66999839947	X =	0.00000000000
	YC =	0.85095927803	Y =	-4.66999839947
	ZC =	0.00000000000	Z =	0.85095927803
Pole Number		3		
Weight	=	1.00000000000		
Coordinates	XC =	-3.37812452597	X =	0.00000000000
	YC =	1.13458559497	Y =	-3.37812452597
	ZC =	0.00000000000	Z =	1.13458559497

Pole Number	4			
Weight	=	1.00000000000		
Coordinates				
	XC =	-2.05635851355	X =	0.00000000000
	YC =	1.40085249865	Y =	-2.05635851355
	ZC =	0.00000000000	Z =	1.40085249865
Pole Number	5			
Weight	=	1.00000000000		
Coordinates				
	XC =	-0.71137408750	X =	0.00000000000
	YC =	1.46125334161	Y =	-0.71137408750
	ZC =	0.00000000000	Z =	1.46125334161
Pole Number	6			
Weight	=	1.00000000000		
Coordinates				
	XC =	0.78619320700	X =	0.00000000000
	YC =	1.44891527757	Y =	0.78619320700
	ZC =	0.00000000000	Z =	1.44891527757
Pole Number	7			
Weight	=	1.00000000000		
Coordinates				
	XC =	2.53682697608	X =	0.00000000000
	YC =	1.16640726706	Y =	2.53682697608
	ZC =	0.00000000000	Z =	1.16640726706
Pole Number	8			
Weight	=	1.00000000000		
Coordinates				
	XC =	4.22840002199	X =	0.00000000000
	YC =	0.77585007737	Y =	4.22840002199
	ZC =	0.00000000000	Z =	0.77585007737
Pole Number	9			
Weight	=	1.00000000000		
Coordinates				
	XC =	5.20720866556	X =	0.00000000000
	YC =	0.51314611426	Y =	5.20720866556
	ZC =	0.00000000000	Z =	0.51314611426

Control points of Spline 2:

Pole Number	1			
Weight	=	1.00000000000		
Coordinates				
	XC =	-4.79796934492	X =	10.00000000000
	YC =	0.38940870290	Y =	-4.79796934492
	ZC =	0.00000000000	Z =	0.38940870290
Pole Number	2			
Weight	=	1.00000000000		
Coordinates				
	XC =	-4.26415509035	X =	10.00000000000
	YC =	0.55815957213	Y =	-4.26415509035
	ZC =	0.00000000000	Z =	0.55815957213
Pole Number	3			
Weight	=	1.00000000000		
Coordinates				
	XC =	-3.17383443511	X =	10.00000000000
	YC =	0.91126209640	Y =	-3.17383443511
	ZC =	0.00000000000	Z =	0.91126209640
Pole Number	4			

Weight	=	1.00000000000		
Coordinates	XC =	-1.79596436598	X =	10.00000000000
	YC =	1.44257759491	Y =	-1.79596436598
	ZC =	0.00000000000	Z =	1.44257759491
Pole Number		5		
Weight	=	1.00000000000		
Coordinates	XC =	0.14571281761	X =	10.00000000000
	YC =	2.06483273820	Y =	0.14571281761
	ZC =	0.00000000000	Z =	2.06483273820
Pole Number		6		
Weight	=	1.00000000000		
Coordinates	XC =	1.84532303700	X =	10.00000000000
	YC =	1.78653258887	Y =	1.84532303700
	ZC =	0.00000000000	Z =	1.78653258887
Pole Number		7		
Weight	=	1.00000000000		
Coordinates	XC =	2.92687188765	X =	10.00000000000
	YC =	0.74767965910	Y =	2.92687188765
	ZC =	0.00000000000	Z =	0.74767965910
Pole Number		8		
Weight	=	1.00000000000		
Coordinates	XC =	3.62911822007	X =	10.00000000000
	YC =	0.34843638503	Y =	3.62911822007
	ZC =	0.00000000000	Z =	0.34843638503
Pole Number		9		
Weight	=	1.00000000000		
Coordinates	XC =	3.90244845487	X =	10.00000000000
	YC =	0.19199241571	Y =	3.90244845487
	ZC =	0.00000000000	Z =	0.19199241571

Complex surface:

Four B-splines are built first. The convex surface is constructed with a through-curve method based on these four B-splines.

Control points of Spline 1:

Pole Number		1		
Weight	=	1.00000000000		
Coordinates	XC =	-4.83452923734	X =	0.00000000000
	YC =	0.83117426975	Y =	-4.83452923734
	ZC =	0.00000000000	Z =	0.83117426975
Pole Number		2		
Weight	=	1.00000000000		
Coordinates	XC =	-4.23359412867	X =	0.00000000000
	YC =	1.22852571944	Y =	-4.23359412867
	ZC =	0.00000000000	Z =	1.22852571944
Pole Number		3		

Weight	=	1.00000000000		
Coordinates	XC =	-3.17572336319	X =	0.00000000000
	YC =	1.63254247132	Y =	-3.17572336319
	ZC =	0.00000000000	Z =	1.63254247132

Pole Number	4			
Weight	=	1.00000000000		
Coordinates	XC =	-0.94905101320	X =	0.00000000000
	YC =	1.46408074758	Y =	-0.94905101320
	ZC =	0.00000000000	Z =	1.46408074758

Pole Number	5			
Weight	=	1.00000000000		
Coordinates	XC =	-0.55878264161	X =	0.00000000000
	YC =	0.38887242279	Y =	-0.55878264161
	ZC =	0.00000000000	Z =	0.38887242279

Pole Number	6			
Weight	=	1.00000000000		
Coordinates	XC =	1.88282986778	X =	0.00000000000
	YC =	0.93340485851	Y =	1.88282986778
	ZC =	0.00000000000	Z =	0.93340485851

Pole Number	7			
Weight	=	1.00000000000		
Coordinates	XC =	2.68592925279	X =	0.00000000000
	YC =	1.34201837238	Y =	2.68592925279
	ZC =	0.00000000000	Z =	1.34201837238

Pole Number	8			
Weight	=	1.00000000000		
Coordinates	XC =	4.21725080203	X =	0.00000000000
	YC =	1.32887093210	Y =	4.21725080203
	ZC =	0.00000000000	Z =	1.32887093210

Pole Number	9			
Weight	=	1.00000000000		
Coordinates	XC =	5.47757285099	X =	0.00000000000
	YC =	0.93515361287	Y =	5.47757285099
	ZC =	0.00000000000	Z =	0.93515361287

Pole Number	10			
Weight	=	1.00000000000		
Coordinates	XC =	6.21390834899	X =	0.00000000000
	YC =	0.63441403312	Y =	6.21390834899
	ZC =	0.00000000000	Z =	0.63441403312

Control points of Spline 2:

Pole Number	1			
Weight	=	1.00000000000		
Coordinates	XC =	-4.90234051572	X =	3.00000000000
	YC =	1.92682808671	Y =	-4.90234051572
	ZC =	0.00000000000	Z =	1.92682808671

Pole Number	2			
Weight	=	1.00000000000		

Coordinates	XC = -4.31868302157	X = 3.00000000000
	YC = 1.46328658447	Y = -4.31868302157
	ZC = 0.00000000000	Z = 1.46328658447
Pole Number	3	
Weight	= 1.00000000000	
Coordinates	XC = -3.29601199362	X = 3.00000000000
	YC = 0.94499320514	Y = -3.29601199362
	ZC = 0.00000000000	Z = 0.94499320514
Pole Number	4	
Weight	= 1.00000000000	
Coordinates	XC = -1.62130604026	X = 3.00000000000
	YC = 0.56743027860	Y = -1.62130604026
	ZC = 0.00000000000	Z = 0.56743027860
Pole Number	5	
Weight	= 1.00000000000	
Coordinates	XC = -0.06810139837	X = 3.00000000000
	YC = 1.59908032762	Y = -0.06810139837
	ZC = 0.00000000000	Z = 1.59908032762
Pole Number	6	
Weight	= 1.00000000000	
Coordinates	XC = 1.53984681152	X = 3.00000000000
	YC = 0.37286896354	Y = 1.53984681152
	ZC = 0.00000000000	Z = 0.37286896354
Pole Number	7	
Weight	= 1.00000000000	
Coordinates	XC = 3.88950280147	X = 3.00000000000
	YC = 0.07234185862	Y = 3.88950280147
	ZC = 0.00000000000	Z = 0.07234185862
Pole Number	8	
Weight	= 1.00000000000	
Coordinates	XC = 5.38694414337	X = 3.00000000000
	YC = 0.84131249677	Y = 5.38694414337
	ZC = 0.00000000000	Z = 0.84131249677
Pole Number	9	
Weight	= 1.00000000000	
Coordinates	XC = 6.25167971058	X = 3.00000000000
	YC = 1.70120947278	Y = 6.25167971058
	ZC = 0.00000000000	Z = 1.70120947278
Control points of Spline 3:		
Pole Number	1	
Weight	= 1.00000000000	
Coordinates	XC = -4.84125201982	X = 6.00000000000
	YC = 1.09443077247	Y = -4.84125201982
	ZC = 0.00000000000	Z = 1.09443077247
Pole Number	2	
Weight	= 1.00000000000	
Coordinates	XC = -4.43696319710	X = 6.00000000000
	YC = 1.30713283670	Y = -4.43696319710

	ZC =	0.00000000000	Z =	1.30713283670
Pole Number	3			
Weight	=	1.00000000000		
Coordinates	XC =	-3.62745102168	X =	6.00000000000
	YC =	1.57886959732	Y =	-3.62745102168
	ZC =	0.00000000000	Z =	1.57886959732
Pole Number	4			
Weight	=	1.00000000000		
Coordinates	XC =	-2.31465871192	X =	6.00000000000
	YC =	1.82575599182	Y =	-2.31465871192
	ZC =	0.00000000000	Z =	1.82575599182
Pole Number	5			
Weight	=	1.00000000000		
Coordinates	XC =	-1.17487972906	X =	6.00000000000
	YC =	1.14838038410	Y =	-1.17487972906
	ZC =	0.00000000000	Z =	1.14838038410
Pole Number	6			
Weight	=	1.00000000000		
Coordinates	XC =	0.20455008416	X =	6.00000000000
	YC =	0.09112469148	Y =	0.20455008416
	ZC =	0.00000000000	Z =	0.09112469148
Pole Number	7			
Weight	=	1.00000000000		
Coordinates	XC =	1.96811951810	X =	6.00000000000
	YC =	1.06366475449	Y =	1.96811951810
	ZC =	0.00000000000	Z =	1.06366475449
Pole Number	8			
Weight	=	1.00000000000		
Coordinates	XC =	4.01891138412	X =	6.00000000000
	YC =	1.10082739513	Y =	4.01891138412
	ZC =	0.00000000000	Z =	1.10082739513
Pole Number	9			
Weight	=	1.00000000000		
Coordinates	XC =	5.33380956787	X =	6.00000000000
	YC =	1.04158492551	Y =	5.33380956787
	ZC =	0.00000000000	Z =	1.04158492551
Pole Number	10			
Weight	=	1.00000000000		
Coordinates	XC =	6.14637447872	X =	6.00000000000
	YC =	0.90265495062	Y =	6.14637447872
	ZC =	0.00000000000	Z =	0.90265495062
Control points of Spline 4:				
Pole Number	1			
Weight	=	1.00000000000		
Coordinates	XC =	-4.91872042710	X =	9.00000000000
	YC =	1.66788109113	Y =	-4.91872042710
	ZC =	0.00000000000	Z =	1.66788109113

Pole Number	2		
Weight	=	1.00000000000	
Coordinates			
	XC =	-4.50806485641	X = 9.00000000000
	YC =	1.31978766736	Y = -4.50806485641
	ZC =	0.00000000000	Z = 1.31978766736
Pole Number	3		
Weight	=	1.00000000000	
Coordinates			
	XC =	-3.69569968219	X = 9.00000000000
	YC =	0.83943936563	Y = -3.69569968219
	ZC =	0.00000000000	Z = 0.83943936563
Pole Number	4		
Weight	=	1.00000000000	
Coordinates			
	XC =	-2.38801615698	X = 9.00000000000
	YC =	0.54689214573	Y = -2.38801615698
	ZC =	0.00000000000	Z = 0.54689214573
Pole Number	5		
Weight	=	1.00000000000	
Coordinates			
	XC =	-1.16405179513	X = 9.00000000000
	YC =	0.67508618677	Y = -1.16405179513
	ZC =	0.00000000000	Z = 0.67508618677
Pole Number	6		
Weight	=	1.00000000000	
Coordinates			
	XC =	-0.23741060782	X = 9.00000000000
	YC =	1.10763923085	Y = -0.23741060782
	ZC =	0.00000000000	Z = 1.10763923085
Pole Number	7		
Weight	=	1.00000000000	
Coordinates			
	XC =	0.77475516450	X = 9.00000000000
	YC =	0.87802558212	Y = 0.77475516450
	ZC =	0.00000000000	Z = 0.87802558212
Pole Number	8		
Weight	=	1.00000000000	
Coordinates			
	XC =	1.48534746713	X = 9.00000000000
	YC =	0.36426627421	Y = 1.48534746713
	ZC =	0.00000000000	Z = 0.36426627421
Pole Number	9		
Weight	=	1.00000000000	
Coordinates			
	XC =	2.68554847294	X = 9.00000000000
	YC =	0.05610475524	Y = 2.68554847294
	ZC =	0.00000000000	Z = 0.05610475524
Pole Number	10		
Weight	=	1.00000000000	
Coordinates			
	XC =	3.99087060784	X = 9.00000000000
	YC =	0.04910304933	Y = 3.99087060784
	ZC =	0.00000000000	Z = 0.04910304933
Pole Number	11		
Weight	=	1.00000000000	
Coordinates			
	XC =	5.44371156916	X = 9.00000000000

	YC =	0.23208672233	Y =	5.44371156916
	ZC =	0.00000000000	Z =	0.23208672233
Pole Number		12		
Weight	=	1.00000000000		
Coordinates	XC =	5.96379566955	X =	9.00000000000
	YC =	0.86095850260	Y =	5.96379566955
	ZC =	0.00000000000	Z =	0.86095850260
Pole Number		13		
Weight	=	1.00000000000		
Coordinates	XC =	6.13659165562	X =	9.00000000000
	YC =	1.16023920978	Y =	6.13659165562
	ZC =	0.00000000000	Z =	1.16023920978

APPENDIX B

A sample post-processing file output from the TMM system.

```
$$ -----
$$      Generated on Monday, September 04, 2006 6:16:27 PM
$$      CATIA APT VERSION 1.0
$$ -----
$$ Manufacturing Program.1
$$ Part Operation.1
$$*CATIAO
$$ Manufacturing Program.1
$$      1.00000      0.00000      0.00000      0.00000
$$      0.00000      1.00000      0.00000      0.00000
$$      0.00000      0.00000      1.00000      0.00000
PARTNO PART TO BE MACHINED
COOLNT/ON
CUTCOM/OFF
$$ OPERATION NAME : Tool Change.1
$$ Start generation of : Tool Change.1
TLAXIS/ 0.000000, 0.000000, 1.000000
$$ TOOLCHANGEBEGINNING
CUTTER/  1,  0.5,  0.000000,  0.5,  0.000000,$
        0.000000,  1.969000
TOOLNO/1,  0.394000
TPRINT/T1 End Mill D 0.394
LOADTL/1
$$ TOOLCHANGEEND
$$ End of generation of : Tool Change.1
$$ OPERATION NAME : Multi-Axis Sweeping.1
$$ Start generation of : Multi-Axis Sweeping.1
$$ Cutter: D: 1, corner R Rc: 0.5, cutting length: 1.969
FEDRAT/ 1000.0000,MPPM
SPINDL/  70.0000,RPM,CLW
GOTO/ 0.000000,-5.250357,4.115815,0.000000,0.000000,1.000000
GOTO/ 0.000000,-5.250357,2.115815,0.000000,0.000000,1.000000
GOTO/ 0.277169,-5.250357,2.077252,0.000000,0.000000,1.000000
GOTO/ 0.522732,-5.250357,2.058816,0.000000,0.000000,1.000000
GOTO/ 0.913461,-5.250357,2.029160,0.000000,0.000000,1.000000
.....
GOTO/ 1.895911,5.039643,1.626905,0.000000,0.000000,1.000000
GOTO/ 1.897385,5.039643,1.626777,0.000000,0.000000,1.000000
GOTO/ 1.921513,5.039643,1.628283,0.000000,0.000000,1.000000
GOTO/ 1.897385,5.039643,3.626777,0.000000,0.000000,1.000000
$$ End of generation of : Multi-Axis Sweeping.1
```

SPINDL/OFF
REWIND/0
END